



Possibilities Unlimited

AssistEdge Community Edition 20.1 Automation Studio User Manual



COPYRIGHT NOTICE

©2023 EdgeVerve Systems Limited, Bangalore, India. All Rights Reserved. This documentation is the sole property of EdgeVerve Systems Limited ("EdgeVerve"). EdgeVerve believes the information in this document or page is accurate as of its publication date; such information is subject to change without notice. EdgeVerve acknowledges the proprietary rights of other companies to the trademarks, product names and such other intellectual property rights mentioned in this document. This document is not for general distribution and is meant for use solely by the person or entity that it has been specifically issued to and can be used for the sole purpose it is intended to be used for as communicated by EdgeVerve in writing.

Except as expressly permitted by EdgeVerve in writing, neither this documentation nor any part of it may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, printing, photocopying, recording, or otherwise, without the prior written permission of EdgeVerve and/ or any named intellectual property rights holders under this document.

EDGEVERVE SYSTEMS LIMITED

Electronics City, Hosur Road, Bangalore 560 100, India.

www.edgeverve.com

AE-CE-AUTOMATION STUDIO USER MANUAL

AUTHORIZED BY: EDGEVERVE SYSTEMS LIMITED

DATE: NOV, 18, 2023.

REVISION HISTORY

Product Version	Revision Version	Revision Date	Change Description
20.1	1.0	18/11/2023	Final Version

Contents

- 1 Introduction 11
 - 1.1 About AssistEdge Community Edition..... 11
 - 1.2 Scope of the Manual 11
 - 1.3 Naming Conventions..... 11
- 2 Automation Studio 12
 - 2.1 Capabilities..... 12
 - 2.2 Automation Process Lifecycle..... 13
- 3 Getting Started 14
 - 3.1 Prerequisite and Installation..... 14
 - 3.2 Login to Automation Studio 14
 - 3.2.1 Studio Welcome Screen 15
 - 3.2.2 Other Studio Options..... 17
- 4 My Processes..... 25
 - 4.1 Filter Processes 26
 - 4.2 Rename and Delete a Process 28
- 5 Admin Capabilities 29
 - 5.1 Applications 30
 - 5.1.2 Configure Application 31
 - 5.2 Deployment 62
 - 5.2.1 Deploy a Process..... 63
 - 5.2.2 Decommission and Delete a Process..... 64
 - 5.2.3 Add as Processbot..... 65
 - 5.2.4 Restore a Process..... 66
 - 5.3 Migration 68
 - 5.3.1 Export a Process..... 68
 - 5.3.2 Import a Process..... 71
 - 5.4 Bots 76
 - 5.4.1 Processbots..... 76
 - 5.4.2 Microbots..... 85
 - 5.5 Process Backups 89
 - 5.6 Marketplace..... 90
 - 5.6.1 Downloading Skill form AssistEdge Marketplace..... 90

5.6.2 Publishing Skills in AssistEdge Marketplace..... 91

6 Manage Studio.....93

6.1 The Tool Bar.....93

6.2 The Canvas.....94

6.3 The Canvas Tools Panel94

 6.3.1 The Activity..... 97

6.4 The Flowchart Designer98

 6.4.1 Additional Features..... 99

6.5 The Properties & Suggestions Grid.....99

6.6 The Outline Grid..... 100

6.7 The Parameter Bar 101

 6.7.1 Variables in Automation Studio..... 101

 6.7.2 Arguments in Automation Studio 103

6.8 The Status Bar 105

 6.8.1 The Watch Console..... 106

 6.8.2 The Studio Console Output..... 106

 6.8.3 The Warnings Console 106

 6.8.4 The Error Console 107

 6.8.5 The Execution Stats Console 107

 6.8.6 The Validation Error Console..... 108

7 Prerequisite Checker..... 109

7.1 Chrome Browser Compatibility Check..... 109

7.2 Internet Explorer Browser Compatibility Check..... 111

7.3 Firefox Browser Compatibility Check..... 113

7.4 Microsoft Edge Browser Compatibility Check..... 114

7.5 SAP Thick Client Compatibility Check..... 116

7.6 Screen Resolution Compatibility Check..... 117

7.7 Display Scaling Compatibility Check 118

7.8 Microsoft Excel Compatibility Check..... 119

7.9 Microsoft Outlook Compatibility Check 120

7.10 Operating System Compatibility Check 121

7.11 Adobe Acrobat Reader Compatibility Check 121

7.12 Java Compatibility Check 123

7.13 Health Report and Warning Console 124

- 8 Process Management 125
 - 8.1 Automation Process Workflow 125
 - 8.2 Auto-Save Process Workflow 125
 - 8.2.1 Auto-Save Interval 125
 - 8.2.2 Enable or Disable Auto-Save Option 126
 - 8.3 Types of Automation Processes 126
 - 8.3.1 Process 126
 - 8.3.2 Sign In Process 128
 - 8.3.3 Reset Process 136
 - 8.4 Process Input/Output 138
 - 8.4.1 Using Process Input 138
 - 8.4.2 Process Outputs 141
 - 8.5 Setup Environment 142
 - 8.5.1 Reset Environment 143
 - 8.5.2 Close Environment 143
 - 8.6 Test Run Process 143
 - 8.6.1 Debug 144
 - 8.7 Publish Process 145
 - 8.8 Deploy Process 147
- 9 Canvas Tools 148
 - 9.1 Computer Vision 149
 - 9.1.1 NIA Vision 149
 - 9.1.2 NIA Vision Activity Properties - NIA provided OCR 152
 - 9.1.3 Output Consumption 153
 - 9.2 Contextual Intelligence 154
 - 9.2.1 Decision Model 154
 - 9.3 Cognitive Services 167
 - 9.3.1 Prerequisites 167
 - 9.3.2 Language 168
 - 9.3.3 Vision 175
 - 9.3.4 Speech 179
 - 9.3.5 Generic Cognitive Service 185
 - 9.4 Assign and Writeline 191
 - 9.4.1 Assign 191
 - 9.4.2 Writeline 193

- 9.4.3 Write to Log files..... 194
- 9.4.4 Add Log Fields..... 195
- 9.4.5 Remove Log Fields..... 198
- 9.5 Flow Controls 199
 - 9.5.1 If 200
 - 9.5.2 Parallel..... 201
 - 9.5.3 DoWhile 202
 - 9.5.4 Sequence..... 203
 - 9.5.5 While..... 204
 - 9.5.6 Decision Node..... 205
 - 9.5.7 Delay..... 206
- 9.6 Process Components..... 208
 - 9.6.1 Applications..... 208
 - 9.6.2 Using Reuse Process..... 317
 - 9.6.3 Using SignIn Process..... 321
 - 9.6.4 Using Reset Process..... 323
 - 9.6.5 Process Kill..... 325
 - 9.6.6 Application Host..... 328
 - 9.6.7 Microbot..... 376
 - 9.6.8 Code Editor..... 380
 - 9.6.9 Validation..... 393
 - 9.6.10 Popup Handler 393
 - 9.6.11 API Activity..... 397
 - 9.6.12 SOAP API Activity 397
 - 9.6.13 Rest API Activity..... 403
 - 9.6.14 JSON Activity..... 413
 - 9.6.15 XML Activity..... 433
 - 9.6.16 FTP Activity..... 446
 - 9.6.17 Execute DB Query..... 465
 - 9.6.18 Execute Java Script..... 466
 - 9.6.19 Key Events..... 468
 - 9.6.20 Focus Window..... 474
 - 9.6.21 Dock Window..... 477
 - 9.6.22 Print Screen 479
 - 9.6.23 Win Popup Handler..... 480

- 9.6.24 *Navigate To*.....496
- 9.6.25 *Regex Validator*.....499
- 9.6.26 *Python Script Automation*.....502
- 9.6.27 *Disable Retry*.....510
- 9.6.28 *Launch and SignIn Application*.....512
- 9.6.29 *Close Application*.....513
- 9.7 *Complex Iterators*.....515
 - 9.7.1 *Advanced Loop*.....515
 - 9.7.2 *Preview & Filter*.....528
 - 9.7.3 *Step-By-Step Guide to Use Advanced Loop to Evaluate the Price of Items in Invoice*.....531
- 9.8 *DataTable*537
 - 9.8.1 *Create DataTable*.....537
 - 9.8.2 *Add Row*.....540
 - 9.8.3 *Remove Row*.....541
 - 9.8.4 *Update Data*.....543
 - 9.8.5 *Clear DataTable*.....545
 - 9.8.6 *Get Data*.....546
 - 9.8.7 *Import DataTable*.....547
 - 9.8.8 *Export DataTable*.....551
 - 9.8.9 *Sort DataTable*.....553
 - 9.8.10 *Count Row Column*.....554
 - 9.8.11 *Example of Create DataTable, Add Row, Sort DataTable and Export DataTable*.....556
- 9.9 *Image* 561
 - 9.9.1 *Image Control*.....561
 - 9.9.2 *Text Extractor*.....566
 - 9.9.3 *Prerequisites*.....566
- 9.10 *Files*577
 - 9.10.1 *Write To Excel*.....578
 - 9.10.2 *Excel Loop*.....581
 - 9.10.3 *Break Loop*.....585
 - 9.10.4 *Use Case*.....590
 - 9.10.5 *Virus Scan*.....595
 - 9.10.6 *File Splitter*.....599
 - 9.10.7 *File Merger*.....603
 - 9.10.8 *File Operations*.....607

- 9.10.9 File Read.....625
- 9.10.10 File Search.....628
- 9.11 Word634
 - 9.11.1 Prerequisites.....635
 - 9.11.2 Replace Text.....636
 - 9.11.3 Replace Image.....638
 - 9.11.4 Save As.....639
 - 9.11.5 Add Hyperlink.....640
 - 9.11.6 Add Image.....643
 - 9.11.7 Add Data Table.....646
- 9.13 Exception Handling650
 - 9.13.1 TryCatch.....650
 - 9.13.2 Throw.....653
 - 9.13.3 Ignore Error Support.....654
- 9.14 E-Mail655
 - 9.14.1 SMTP Mail.....656
 - 9.14.2 Outlook Mail.....666
- 9.15 SAP Thick Client685
 - 9.15.1 Prerequisites.....685
 - 9.15.2 Addition of SAP Application.....686
 - 9.15.3 SAP Recorder.....686
 - 9.15.4 GUI Field Extractor.....691
 - 9.15.5 Attach Applications.....695
- 9.16 SharePoint Activities697
 - 9.16.1 Using Sharepoint Operations Activity.....698
 - 9.16.2 Upload.....699
 - 9.16.3 Download.....700
 - 9.16.4 Copy/Move.....701
 - 9.16.5 Delete.....702
 - 9.16.6 SharePoint Operations Properties.....703
- 9.17 PDF Template Creator.....703
- 9.18 Excel Recorder737
 - 9.18.1 Prerequisites.....737
 - 9.18.2 Using the Excel Recorder.....737
 - 9.18.3 Excel Scripts Properties.....742

- 9.18.4 Step-By-Step Guide to Use Excel Recorder to Calculate the Invoice Total 743
- 9.19 Process Recorder 745
 - 9.19.1 Prerequisites 745
 - 9.19.2 Process Recorder Usage 753
 - 9.19.3 Using Process Recorder 755
 - 9.19.4 Web Mode Properties 762
 - 9.19.5 Windows Mode Properties 767
 - 9.19.6 Process Recorder Properties 771
 - 9.19.7 Handling Dynamic Controls 772
 - 9.19.8 Step-By-Step Guide to Use Process Recorder to Print the Population of a City in Notepad From a Web page 773
- 9.20 Peoplesoft 775
- 9.21 Siebel 775
 - 9.21.1 Navigation 777
 - 9.21.2 Siebel Operations 779
 - 9.21.3 Drilldown 781
 - 9.21.4 Set Field Value 784
 - 9.21.5 Get Field Value 788
 - 9.21.6 Fetch Grid Data 791
 - 9.21.7 Open Task 794
 - 9.21.8 Step-By-Step Guide to Use Siebel to Search for Service Request on the Oracle Siebel CRM HI Application 797
- 9.22 Automation of Oracle E-Business Suite using Micro-bots 802
 - 9.22.1 Oracle EBS Open Form 803
 - 9.22.2 Oracle EBS Menu Options 806
 - 9.22.3 Oracle EBS Toolbar Options 808
 - 9.22.4 Oracle EBS Set Text 811
 - 9.22.5 Oracle EBS Set Radio Button 813
 - 9.22.6 Oracle EBS Get Text 814
 - 9.22.7 Oracle EBS Get Radio Button 816
 - 9.22.8 Oracle EBS Select Tab 819
 - 9.22.9 Oracle EBS Click Hyperlink 820
 - 9.22.10 Oracle EBS Switch Form 821
 - 9.22.11 Click a Button 822
 - 9.22.12 Oracle EBS Start Mapping Server 823
 - 9.22.13 Oracle EBS Stop Mapping Server 823
- 9.23 Automation of Siebel HI Application using Microbots 824

9.23.1 SiebelHINav.....	824
9.23.2 SiebelHIUIButtonClick.....	826
9.23.3 SiebelHIUIGetGridCount.....	827
9.23.4 SiebelHIUIUpdateGridCell.....	828
9.23.5 SiebelHIWebDialogUpdate.....	830
9.23.6 SiebelHIReadWebDialogElement.....	831
9.24 Oracle EBS Codeless Configuration.....	832
9.24.1 Launch an Oracle EBS Form through Web Navigator.....	832
9.24.2 Application Host - Setting Up the Oracle EBS Window for Automation.....	833
9.24.3 Application Control – Basics.....	835
9.24.4 Launch an Oracle EBS Form using Java-based Navigator.....	837
9.24.5 Close, Minimize or Maximize the Form.....	840
9.24.6 Restore Form.....	841
9.24.7 Set Text.....	842
9.24.8 Get Text.....	843
9.24.9 Select Value from the Choice box/Drop-down.....	844
9.24.10 Read Selected Choice-box Value.....	845
9.24.11 Activate an Oracle Form.....	846
9.24.12 Click.....	847
9.24.13 Select Toolbar Item by Value.....	848
9.24.14 Select Toolbar Item by Index.....	849
9.24.15 Select Tab by Value.....	850
9.24.16 Read Selected Tab Name.....	851
9.24.17 Set Check-box.....	852
9.24.18 Get Check-box Status.....	853
9.24.19 Get Check-box Text.....	854
9.24.20 Click Menu Options.....	855
9.24.21 Set Radio Button.....	856
9.24.22 Get Selected Radio Button Text.....	857
9.24.23 Check Component State.....	858
9.24.24 Read Status Bar Message.....	859
9.24.25 Select List-view Item by Index.....	860
9.24.26 Scroll Operations.....	861
9.24.27 Select Grid Row.....	862
9.24.28 Get/Set Grid Cell Data.....	864

9.24.29 *Other Grid Operations*.....866

9.25 Appendix.....866

9.25.1 *Arguments in Automation Studio*866

1

Introduction

1.1 About AssistEdge Community Edition

AssistEdge Community Edition is a lighter version of AssistEdge RPA designed for you! Community users such as students, developers and professionals can design automations and run them on their in-built personal robot with the AssistEdge Community Edition. AssistEdge Community Edition spans across the automation continuum from deterministic through intelligent to human-empowered automation. By seamlessly unifying the human and digital co-worker, AssistEdge enables the conceptualization and delivery of future workforce with greater ease.

1.2 Scope of the Manual

This document lists the various features and functionality of AssistEdge Community Edition Automation Studio.

1.3 Naming Conventions

Acronym	Expansion
AE	AssistEdge
RPA	Robotic Process Automation
CE	Community Edition

2 Automation Studio

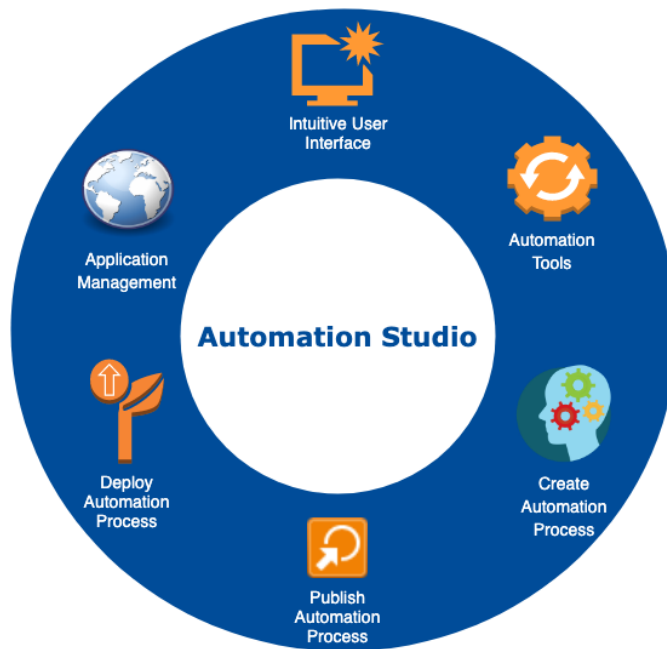
Automation Studio is the core of AssistEdge that empowers you to configure automation processes. The Automation Studio encompass configuration capabilities for a number of technology applications.

A robot, forming a digital workforce for your enterprise, executes the process workflow configured in Automation Studio. The process workflow in collaboration with human in the loop engagement can also be defined in Automation Studio.

2.1 Capabilities

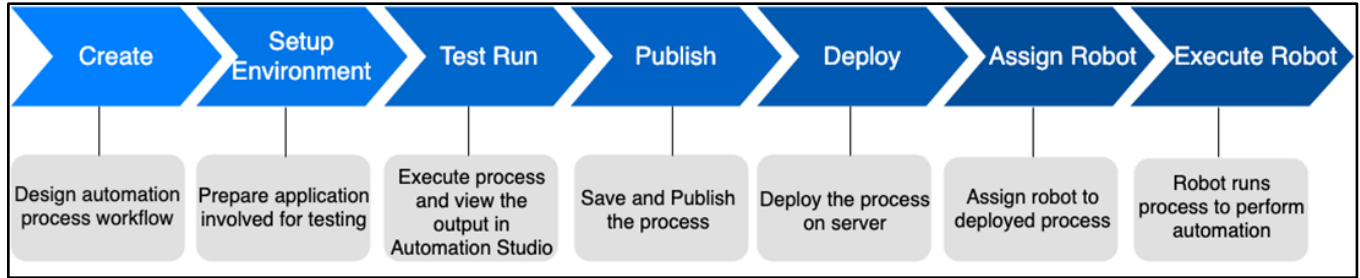
Automation Studio is a powerful tool that fulfills your business needs by offering numerous configurable activities for automation processes. It offers a user friendly and intuitive interface for the ease of workflow configuration.

Additionally, Automation Studio offers you to integrate custom code to automate advanced automation scenarios.



2.2 Automation Process Lifecycle

The life cycle of an automation process workflow in AssistEdge is designed considering key elements of SDLC. Following is the high level automation process life cycle:



A process workflow is designed in Automation Studio which is then deployed on the server. From server the process workflow is assigned to a robot to perform automation.

3 Getting Started

You can start working with Automation Studio after the product is installed using the product package.

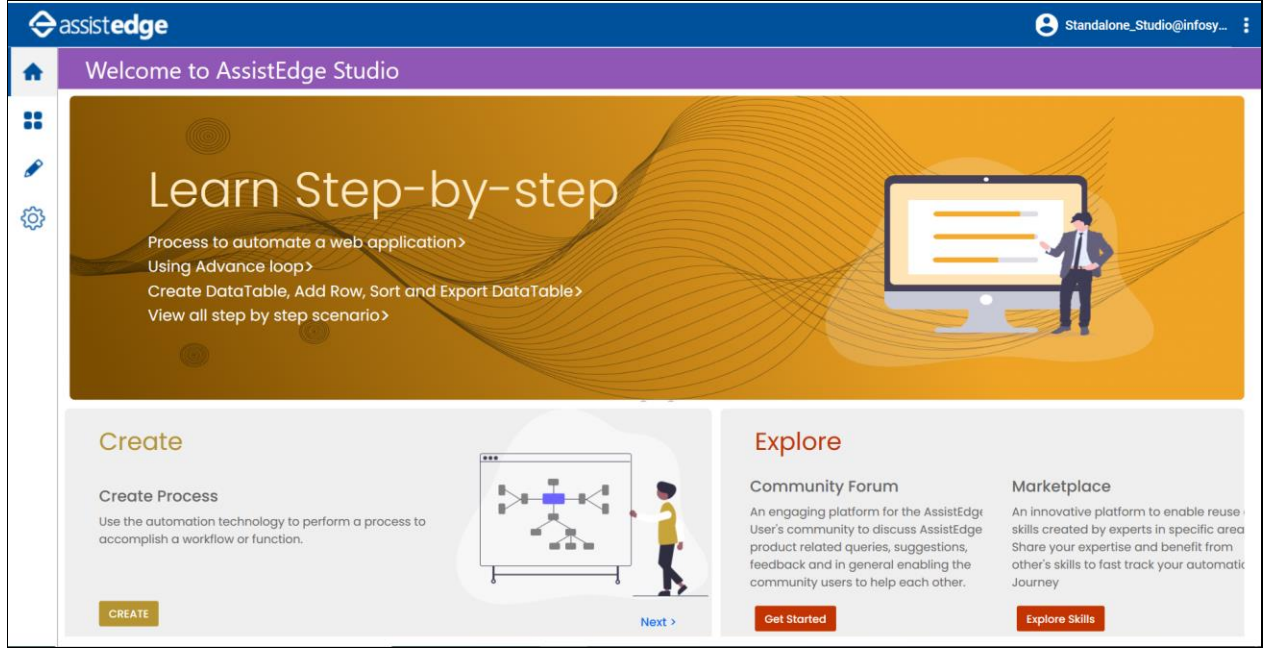
3.1 Prerequisite and Installation

For detailed instructions, refer the AE -CE-CustomizationGuide.pdf.

3.2 Login to Automation Studio

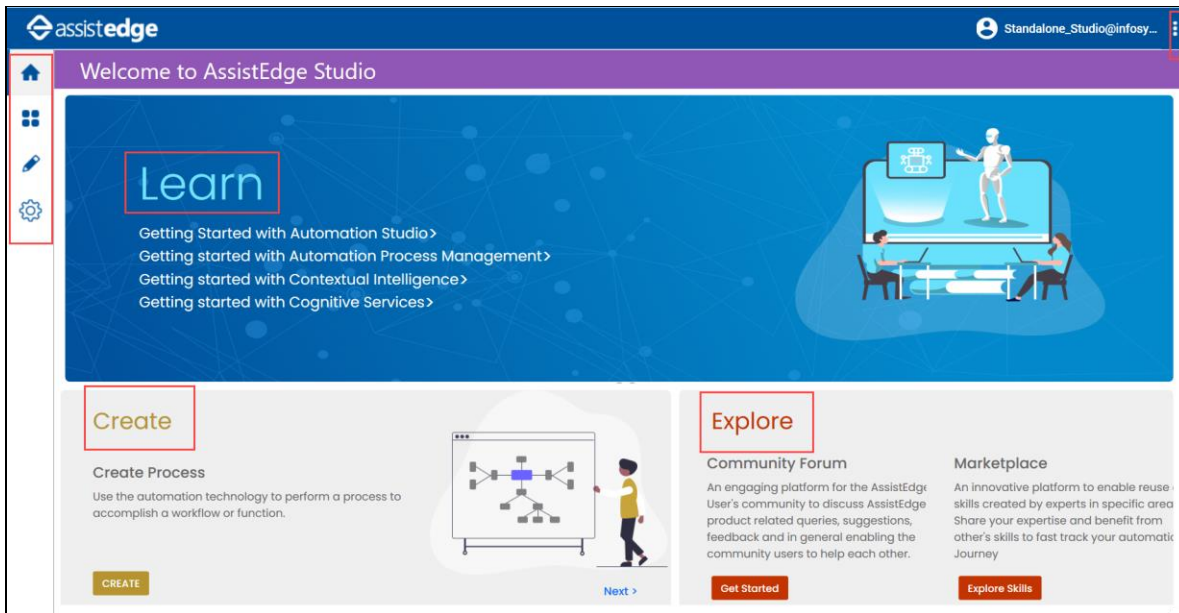
1. Launch **Automation Studio** using the executable file.

2. Click **Close** to go to **Welcome Screen** of Automation Studio.



3.2.1 Studio Welcome Screen

When you login to Studio, the Welcome to AssistEdge Studio page appears. The Studio Welcome page directs you to the knowledge base where you can access information about Studio, and other components of AssistEdge.



The Welcome page includes three panels:

- [Learn](#)
- [Create](#)
- [Explore](#)

Learn

Access the most commonly used knowledge base topics for you to get started with Studio. It includes:

- **Learn:** Provides the **Getting Started** links of the most searched/used topics.
- **Learn Step by Step:** Provides links to scenario-based step by step guide to use studio activities.
- **Helpful Links:** Provides links to the other knowledge base topics for the product.

Create


This section provides the link to the studio page where you can, create, test, publish, and manage the automation processes. The functionality available through the Studio menu is explained in [Manage Studio](#) section.

Explore

This section provides links to:

- **Community:** An engaging platform for the AssistEdge user's community to discuss AssistEdge product related queries, suggestions, feedback and in general enabling the community users to help each other.
- **Marketplace:** An innovative platform to enable reuse of skills created by experts in specific areas. Share your experience and benefit from other's skills to fast track your automation journey.

3.2.2 Other Studio Options

The other Studio options  available in the vertical ellipsis at the top right corner of your Studio welcome page, help you with the overall information about Studio. Navigate to the given options as per your requirement.

About
View Logs
Health Report
Check For Update
Show the tour
Preferences
Configurations
Community Forum

- [About](#)
- [View Logs](#)
- [Health Report](#)
- [Check for Updates](#)
- [Show the Tour](#)
- [Preferences](#)
- [Configurations](#)
- [Community Forum](#)

About

View the Release and Patch versions and other Copyright related information.

View Logs

View the Automation Studio logs.

Health Report

View the auto generated summary of the data accumulated by Prerequisite Checker which consists of the following details:

- Evaluated Prerequisite's name.
- Version installed on the system.


- Is the evaluated prerequisite supported by the product?
- The range of versions supported by the product.
- Name of the driver.
- Version of the driver detected.
- Is the driver supported by the installed version of the component?

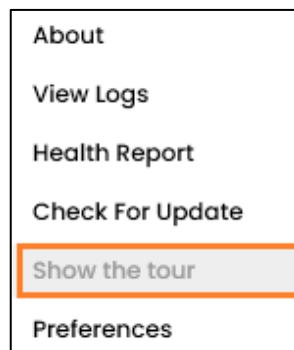
This summary is auto created when the user uses any of the components evaluated by the Prerequisite Checker.

Check for Updates

Check if there are any new updates for you to get the application upgraded to the latest version.

Show the Tour

The first time when you login to Studio, you will be presented with a quick visual tour of the interface. The Guided Tour provides an overview presentation of features that can be accessed within Studio. In case you missed watching the Guided Tour, click  and select Show the tour from the options that appear.

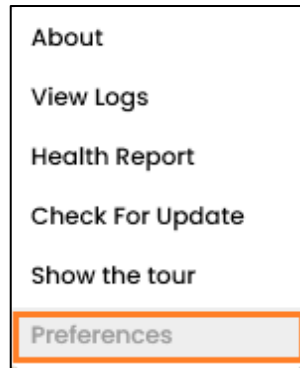


Preferences

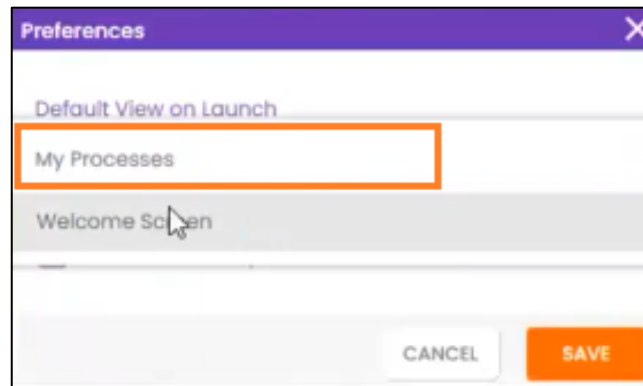
By default, the landing page is set to the Home page. However, you can change the preferences and set the Studio page (where you can create/manage the processes) as your landing page.

To change the default settings for the landing page:

1. Click the  icon. Click **Preferences**.



The below window appears.




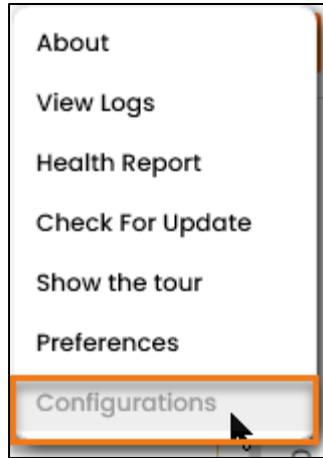
2. Select **My Processes**. Click **Save**. The next time you launch Studio, you will be presented the **My Processes** page. To switch back to the default settings, select the preference as **Welcome Screen** under **Default View on Launch** and save it.

Configurations

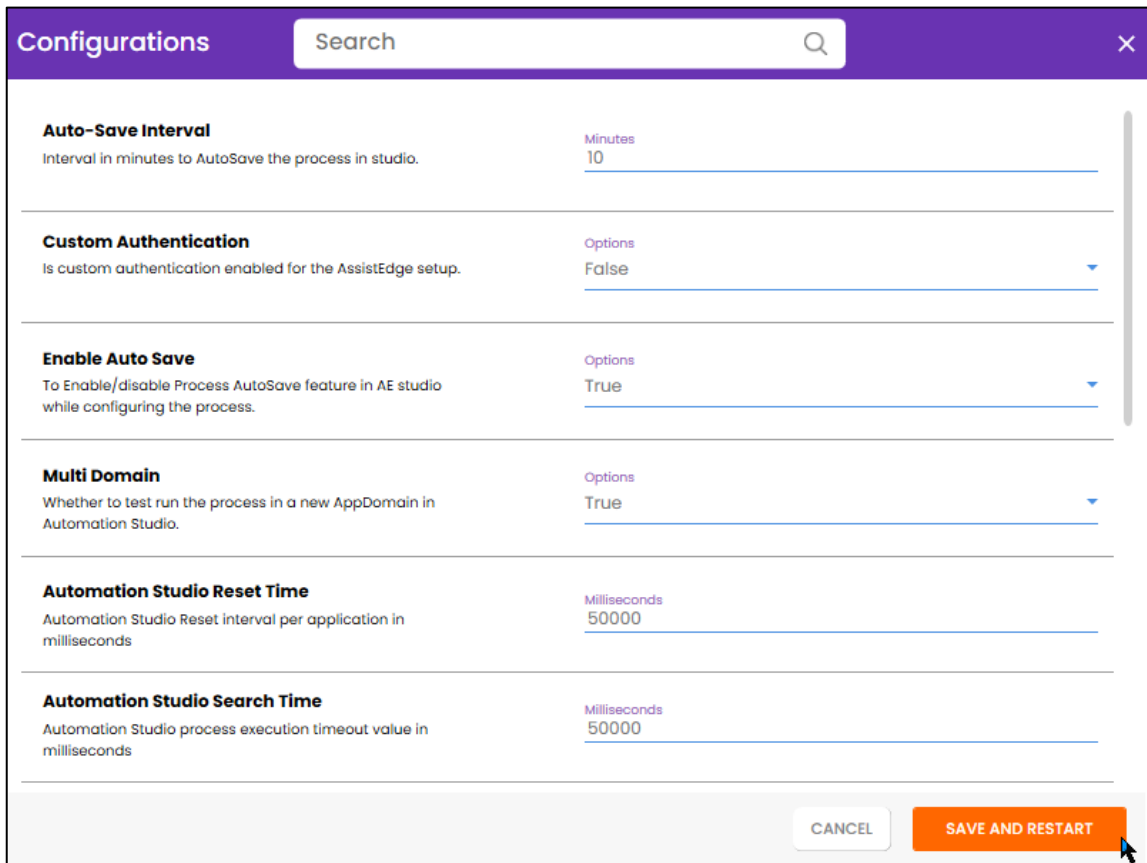
Using this feature, you can edit the configuration file values in Automation Studio.

To configure the values:

1. Click the  icon. Click **Configurations**.



The below window appears.



2. Edit the values as per the requirement.

- Left side pane indicates the configuration keys/settings along with the description and
- Right-side pane indicates the respective values, and you can update the values as required.
- You can search required configurations using search box. It will search the configuration keys matching with the search criteria.

The properties of configurations are listed in the following table.

Fields	Descriptions
Auto-Save Interval	Enables you to set the value for the interval in minutes to Auto save the process in studio.
Custom Authentication	Indicates whether the custom authentication is enabled for the AssistEdge setup. Possible values are: <ul style="list-style-type: none"> True False
Enable Auto Save	Enables you to Enable or Disable the process auto save feature in AE Studio while configuring the feature. <p>Possible values are:</p> <ul style="list-style-type: none"> True False
Multi Domain	Indicates whether you can run the process in a new AppDomain in Automation Studio. <p>Possible values are:</p> <ul style="list-style-type: none"> True False
Automation Studio Reset Time	Enables you to set the Automation Studio Rest interval per application in milliseconds. For example, 50000.
Automation Studio Search Time	Enables you to set the Automation Studio process execution timeout value in milliseconds. For example, 50000.
Automation Studio Sign-in Time	Enables you to set the Automation Studio sign-in interval per application in milliseconds. For example, 50000.
General Log Level	Enables you to update the logging level in AE Studio for general logs. <p>Possible values are:</p> <ul style="list-style-type: none"> Information All Debug
JSON Log Level	Enables you to update the logging level in AE Studio for JSON log level. <p>Possible values are:</p> <ul style="list-style-type: none"> Information All

Fields	Descriptions
	<ul style="list-style-type: none"> ▪ Debug
Automation Way	<p>If a web application contains java components and field config do not work while screen scrapping automation way can be changed for the concerned application.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> ▪ Java_Options_Tools ▪ Java_Options
Stop Patch Notification	<p>To stop the auto checking of the latest available patches while launching the AE Studio.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> ▪ True ▪ False
Show Pre-requisites Alert	<p>To allow the alerts for prerequisites check to be shown.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> ▪ True ▪ False
Chrome Browse Version	<p>Provide the Version of the chrome browser which is being used on the system. Eg.91.0.4472.101.</p>

3. Click **SAVE AND RESTART** to reflect the latest configurations changes. This option enables you to update the configuration values and restart the studio.

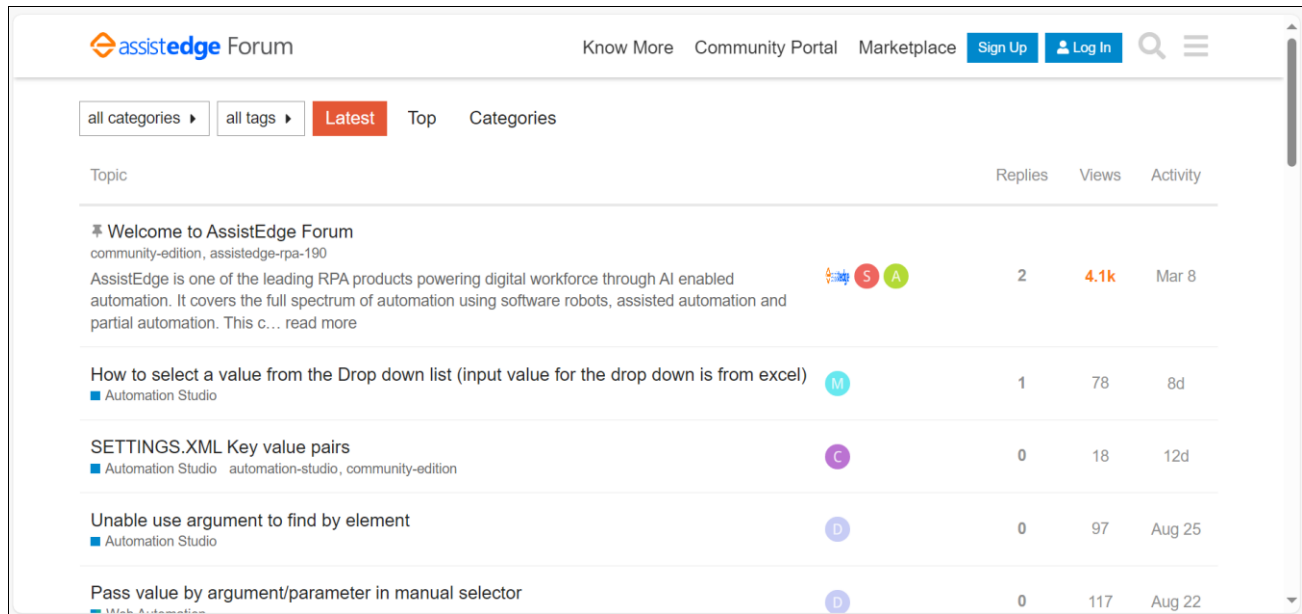
Community Forum

Using this feature, you can access the community forum page.

To access the community forum page:

1. Click the  icon. Click **Community Forum**.




The below page appears.




2. You can view the forum page with details of Topic of discussion, Replies to the question if any, Views of the particular topic along with date the activity was posted on.

Navigating Through Automation Studio

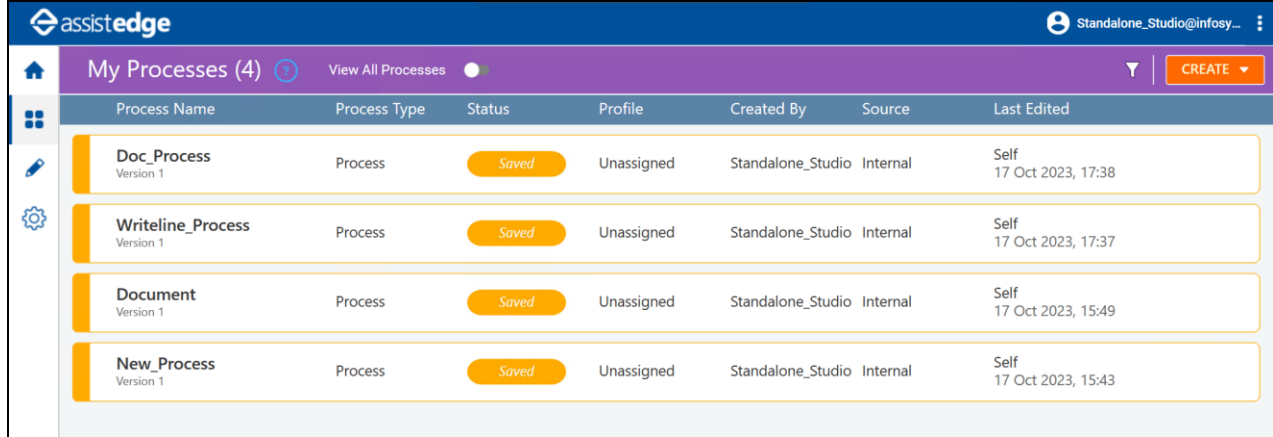
Automation Studio includes three main menus, as described below, that allows you to design a workflow and configure processes as per your business requirements. Additionally, you can create and manage settings of applications, deploy a process, manage process bots, and connect with AssistEdge Marketplace.

- [My Processes](#) 
- [Studio](#) 
- [Admin Menu](#) 

The My Processes Menu

The processes created in Automation Studio can be viewed on the My Processes window . Additionally, the My Processes menu provides an option to view the processes created by other users. The functionality that can be performed on this page is


explained in [My Processes Page](#) section.




The screenshot shows the 'My Processes' page in the AssistEdge interface. The page title is 'My Processes (4)' with a 'View All Processes' link. A 'CREATE' button is visible in the top right. The table below lists four processes, each with a 'Saved' status and 'Unassigned' profile.

Process Name	Process Type	Status	Profile	Created By	Source	Last Edited
Doc_Process Version 1	Process	Saved	Unassigned	Standalone_Studio	Internal	Self 17 Oct 2023, 17:38
Writeline_Process Version 1	Process	Saved	Unassigned	Standalone_Studio	Internal	Self 17 Oct 2023, 17:37
Document Version 1	Process	Saved	Unassigned	Standalone_Studio	Internal	Self 17 Oct 2023, 15:49
New_Process Version 1	Process	Saved	Unassigned	Standalone_Studio	Internal	Self 17 Oct 2023, 15:43

The Studio Menu

The Studio menu  helps you to manage, create, test, and publish the automation processes. The functionality available through the Studio menu is explained in [Manage Studio](#) section.

The Admin Menu

The Admin menu  helps you to create an application, profile and deploy an automation process. You can also import and export an existing process or a bot. The administrative capabilities that can be performed is explained in the [Admin Menu](#) section.

4 My Processes

The My Processes page provides access to the processes available in Automation Studio. It is the landing page of where you can perform actions related to a process such as rename or delete. It provides insight to details related to the processes.

Process Name	Process Type	Status	Profile	Created By	Source	Last Edited
Doc_Process Version 1	Process	Saved	Unassigned	Standalone_Studio	Internal	Self 17 Oct 2023, 17:38
Writeline_Process Version 1	Process	Saved	Unassigned	Standalone_Studio	Internal	Self 17 Oct 2023, 17:37
Document Version 1	Process	Saved	Unassigned	Standalone_Studio	Internal	Self 17 Oct 2023, 15:49
New_Process Version 1	Process	Saved	Unassigned	Standalone_Studio	Internal	Self 17 Oct 2023, 15:43

There is a color code scheme assigned to each status of a process that is available wherever status of process is displayed such as the Homepage, Deployment tab and Process Backups tab:

Color Code Scheme	Description
	Displays the saved status of a particular version of the process.
	Displays the published status of a particular version of the process.
	Displays the deployed status of a particular version of the process.
	Displays the decommissioned status of process. If a version of a process is decommissioned, all the other versions also remain in the decommissioned state, irrespective of their status.

In the Homepage menu, you can perform following functionality related to a process:

1. **Help** – Connects to the Automation Studio online help content.
2. **View All Processes** - Displays the list of processes created in Automation Studio. It helps you to change the view from My Processes to All Processes and vice versa.
3. **Filter** - Filters the processes based on different criteria such as name, type, status, of the processes.
4. **Create** - Creates different types of automation process workflow. For more information, see [Process Management](#).
5. **Process Dependency** - Provides an insight if the process is re-used in any other process.
6. **Application Dependencies** - Provides an insight if the process is dependent upon any application.
7. **More** - Renames or deletes the selected process.

4.1 Filter Processes

You can set filter to list the processes based upon your requirement. You can also combine multiple criteria to filter required processes for the ease of selection. Filter option changes based on the My Processes or All Processes view you select. The available filter criteria are:

- **Process Name** - Filters saved processes based on the name of the process selected. Select the name of the process to set the filter.
- **Process Type** - Filters saved processes based on the type of the process created. Select the type of the process to set the filter. Available sub filters are:
 - **Process** - Filters out the process workflow of **Process** type.
 - Load Generator Process - Filter out the process workflow of Load Generator Process type.
 - **Reset Process** - Filter out the process workflow of **Reset Process** type.
 - **SignIn Process** - Filter out the process workflow of **Sign In Process** type.
 - **Status** - Filters processes based on its status like if the process is saved or deployed. Select the status of the process to set the filter. Available sub filters are:
 - **Decommissioned** - Filters out the decommissioned processes.
 - **Deployed** - Filters out the deployed processes.
 - **Published** - Filters out the published processes.
 - **Saved** - Filters out the saved processes.
 - **Profile** - Filters processes based on the profiles they are assigned to. Select the profile assigned to the process to set the filter.
 - **Created By** - Filters processes based on the name of the process creator. Select the name of the user to set the filter.
 - **Last Edited By** - Filters processes based on the name of the user who last edited the process. Select the name of the user to set the filter.
 - **Edited Date** - Filters the list of processes based on dates range when the processes were edited. Select a **Start**

Date and an **End Date** to set the filter.

- **Application Data** - Filters the processes based on the selected application. Additionally, you can enter a search text to apply the filter. Select name of the application from the drop-down list to set the filter.

The sub filter supports fuzzy search criteria that implies absolute process name is not required to search for the desired process. You can also filter out processes using wildcards - * (asterisk) and? (question mark).


The * wildcard can be used to specify any number of characters, for example searching for Process* yields processes with all possible combination to the root process name string such as Process, Processor, Processed, Processing, Precess12 and, so on.

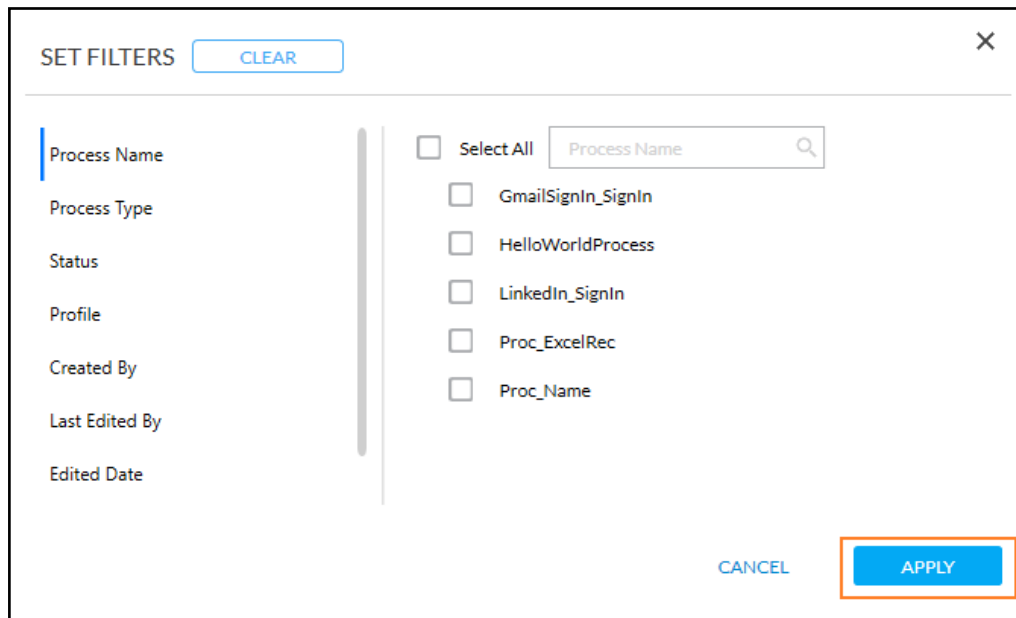
The ? wildcard can be used to specify a single character, for example- searching for a Process?? yields processes with all possible combination to the root process name string such as Process12, Process23, Processed and, so on.

Note:

- Wildcards can be placed at the start, at the end or in between the process name string. Both the wildcards can be used together in combination for the search as well.
- Wildcards are not applicable to **Application Data** filter.

To filter a process:


1. In the **Home** page, click the  (**Filter**) icon. The **SET FILTERS** dialog box appears.



2. Select the filter you want to use and check the respective sub-filters.
3. Click **APPLY** to apply the set filters. If you do not want to apply the set filters, click **CANCEL**.

The filtered process is available based on the My Processes or All Processes view. The filter changes to .

To remove applied filter:

1. In the **Home** page, click the  (**Filter**) icon. The **SET FILTER** dialog box appears.
2. Click **CLEAR** to remove the set filters.

The list of process based on the My Processes/All Processes view is displayed.

4.2 Rename and Delete a Process

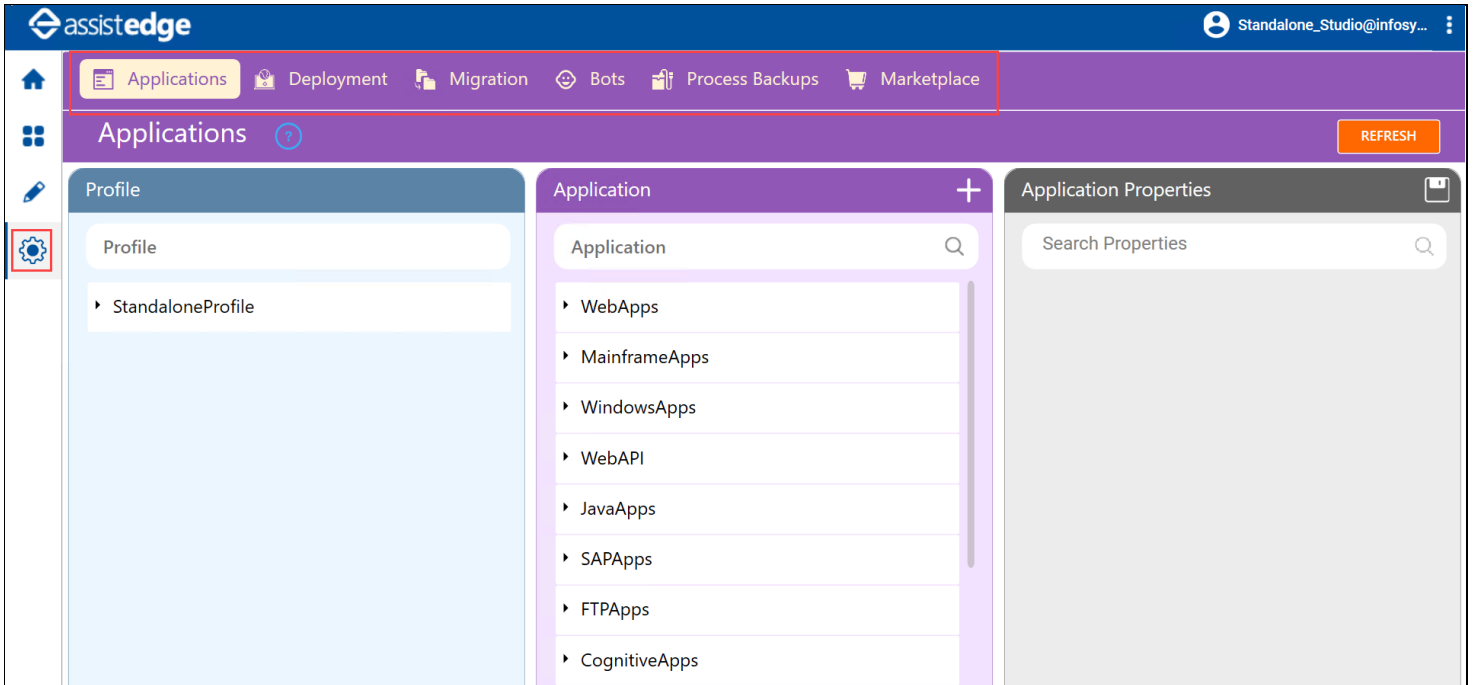
1. In the **Homepage** menu, select the process you want to delete.
2. Click **More**, and then click either **Rename** or **Delete**, depending on your requirement.

The selected process is renamed or deleted as per the chosen option.

Note: A deployed process cannot be deleted.

5 Admin Capabilities

In the Admin menu, you can perform various administrative capabilities such as create applications for a process, decommission a deployed process, import, or export a process, and even connect with AssistEdge Marketplace.

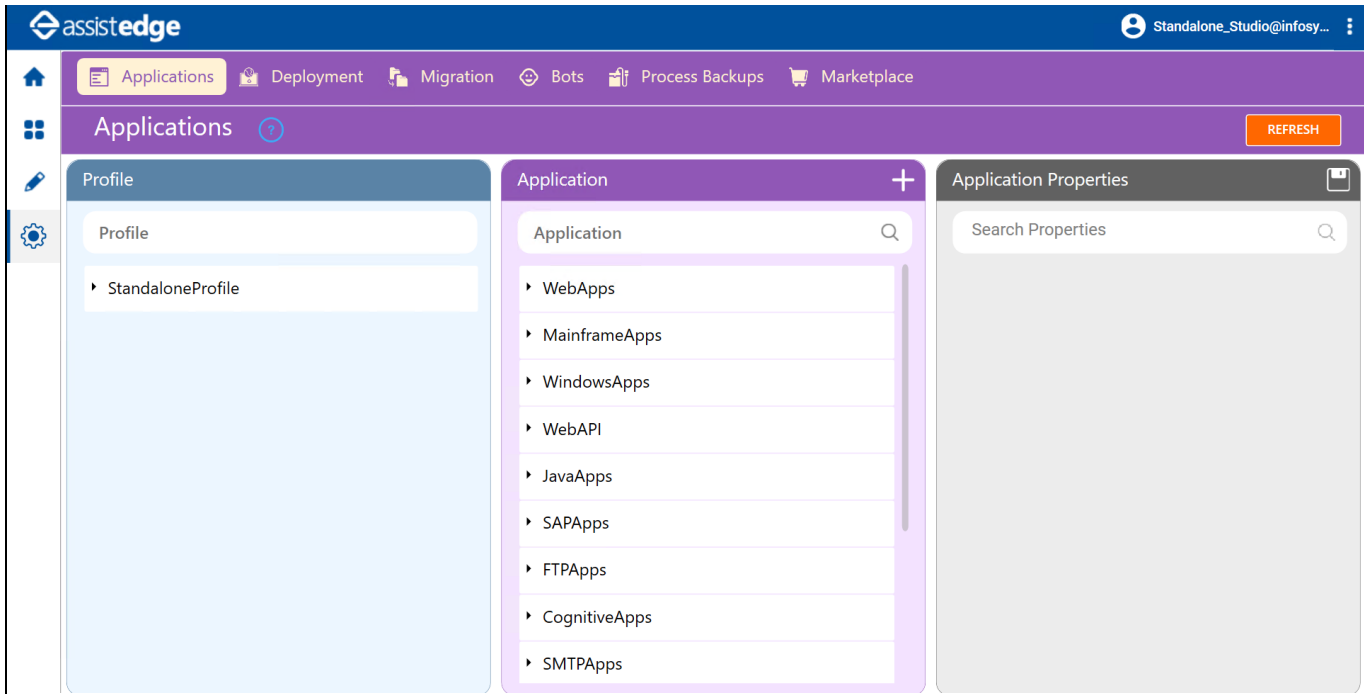


Following are the options available in the Admin menu:

- **Applications** - Manage profiles and applications related to the processes. See [Applications](#) for more details.
- **Deployment** - View the deployed processes and perform actions post deployment such as decommissioning, version control of the process. See [Deployment](#) for more details.
- **Migration** - Import or export processes from one environment to another. See [Migration](#) for more details.
- **Bots** - Creates, import, or export processbots and microbots. See [Bots](#) for more details.
- **Process Backups** - View the processes whose backups have been taken. Process backups for published as well as deployed version of the same process is listed. See [Process Backups](#) for more details.
- **Marketplace** - Search for the bots that are available in AssistEdge Marketplace. Additionally, it lists the downloaded bots from AssistEdge Marketplace. See [Marketplace](#) for more details.

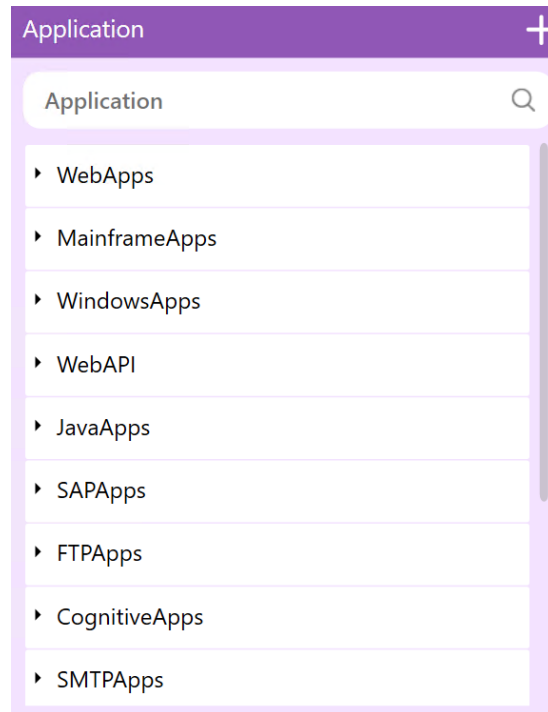
5.1 Applications

In the Applications tab, you can configure application which lets you instantiate the application at the time of setting environment for automation. The properties related to the configured application can be defined and updated in the Application Properties pane. You can search for the required property in the Search Properties search box.




5.1.2 Configure Application



You must configure the application such as web, excel, windows, webAPI, SAP, cognitive and others to create automation process workflow in Automation Studio.



The configuration of various applications help you establish connection with the target application through Automation Studio. Additionally, configuration of application lets you instantiate the application while setting up the environment in Automation Studio.

Once the application is added, you can expand the application type to view the list of configured applications under respective application types. Use the  (View Application Dependency) icon to view the name of the processes where the selected application is used, including the profile assigned to the processes. Additionally, you can rename and delete the configured application in this pane itself.

Rename and Delete an Application

Select the application you want to rename or delete and use the  (Rename Application) icon and  (Delete Application) icon to rename and delete the application respectively.

Note: Applications that are used in any of the processes cannot be deleted.

The application types supported are:

- [WebApps](#)
- [MainframeApps](#)
- [WindowsApps](#)
- [WebAPI](#)
- [JavaApps](#)
- [SAPApps](#)
- [FTPApps](#)
- [CognitiveApps](#)
- [SMTPApps](#)
- [DatabaseApps](#)
- [NIAApps](#)
- [Contextual Intelligence Apps](#)
- [ExcelApps](#)

WebApps

It allows you to configure web applications using browsers- Internet Explorer (IE), Firefox, Chrome, Internet Explorer Selenium, Microsoft Edge, and MicrosoftEdgeWithIECompatibility in Automation Studio.

For old use cases running on Internet Explorer browser, instead of manually reconfiguring the automation steps, Assist Edge's IE to Edge process converter utility enables you to perform this conversion in quick and easy steps with minimal manual efforts. You are required to convert the IE process workflow in Edge process using the IE -Edge process converter utility. The converted process workflow must be mapped to new web application created for the MicrosoftEdgeWithIECompatibility.

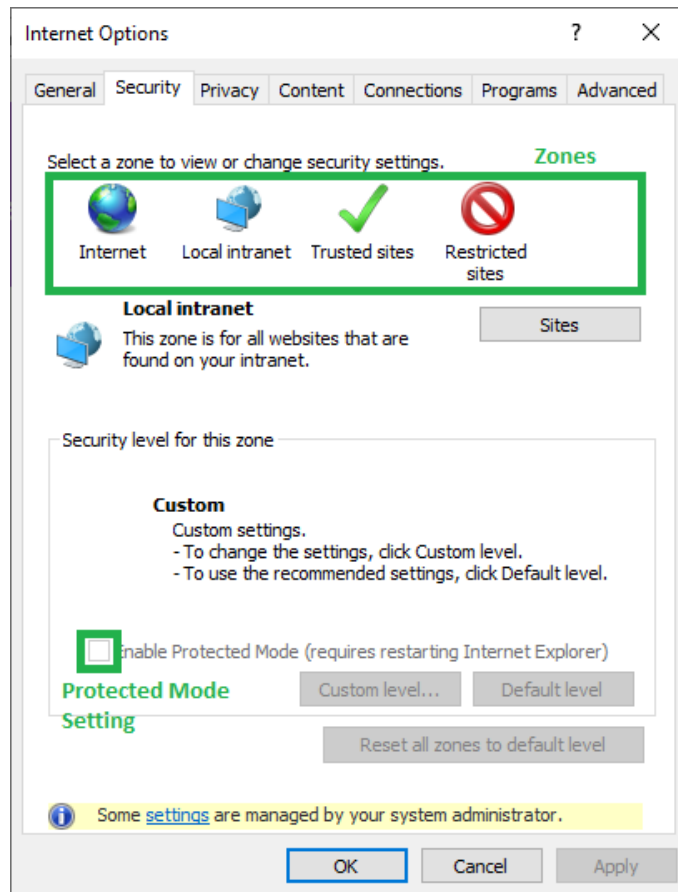
Note: Internet Explorer (IE) applications with document mode 5/7/8 are supported by the IE plugin. Selenium-based plugin InternetExplorerSelenium uses the IE web driver to automate the web applications utilizing IE.

IE Prerequisites for MicrosoftEdgeWithIECompatibility

- Internet Explorer Selenium Driver is used to support this feature. Note that this IE Selenium Driver is known to have some issues. Ensure to validate your automation thoroughly in lower environment before moving to production.

IE prerequisites available in AE 20.0 Troubleshooting guide is applicable here.

- On IE 7 or higher on Windows Vista, Windows 7, or Windows 10, you must set the Protected Mode settings for each zone to be the same value. The value can be on or off if it is the same for every zone.
- To set the **Protected Mode** settings, choose **Internet Options..** from the **Tools** menu, and click on the **Security** tab. For each zone, there will be a check box at the bottom of the tab labeled **Enable Protected Mode**.



- This is the recommended approach provided by Selenium. However, in case its not possible to set the Protected mode setting to same value across all zones, then add the below configuration key within the config file of client tools.

Under BrowserPluginSettings/Parameters

```
<add key="IgnoreProtectedMode" value="true" />
```

- Additionally, "Enhanced Protected Mode" must be disabled for IE 10 and higher. This option is found in the Advanced tab of the Internet Options dialog.
- The browser zoom level must be set to 100% so that the native mouse events can be set to the correct coordinates.
- You also need to set "Change the size of text, apps, and other items" to 100% in display settings.
- In some cases of Remote Desktop connection, scaling value cannot be set on the connected Virtual Machine. In this case, Resolution or Scaling value can be passed to the Remote Desktop Client. Below are the steps mentioned

to set Scaling of the Virtual Machine to 100% using Windows Remote Desktop Connection application.

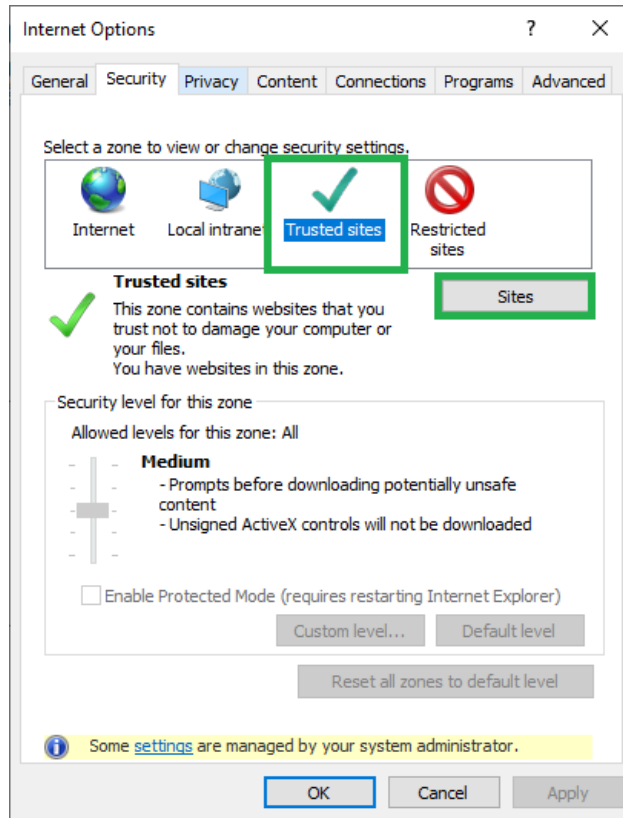
- Go to Advanced options.
- Go to Display tab.
- Use the slider to set the resolution. Scaling is auto set to the recommended value. Set the resolution same as the option in current system resolution setting so the 100% scaling is the recommended option.
 - For IE 11 only, you will need to set a registry entry on the target computer so that the driver can maintain a connection to the instance of Internet Explorer it creates.
- For 32-bit Windows installations, the key you must examine in the registry editor is HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\Main\FeatureControl\FEATURE_BFCACHE.
- For 64-bit Windows installations, the key is HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Internet Explorer\Main\FeatureControl\FEATURE_BFCACHE. Please note that the FEATURE_BFCACHE.
 - FEATURE_BFCACHE subkey may or may not be present and should be created if it is not present. Inside this key, create a DWORD value named iexplore.exe with the value of 0.

Note: Only applications that are launched in Internet Explorer document mode 11 are fully supported.

Known Limitations for MicrosoftEdgeWithIECompatibility

There are some known limitations with Selenium while using Microsoft Edge with IE Compatibility. These limitations along with their workaround are listed below.

- Shift Click interaction only does Click operation on the control. Unlike Chrome and Edge, Shift button won't be clicked due to which the application should launch a new window with URL that is to be attached and automated using JavaScript window.open() on click event.
- It is a known issue with Selenium in Microsoft Edge with IE mode, that handles of browser windows opened using JavaScript window.open() are sometimes not captured. Due to this, the window that is launched during the failed Shift Click interaction needs to be closed using Win Popup Handler activity and Shift Click interaction needs to be retried. Control Exists interaction can be used to detect if new window handle has been captured by Selenium or not. This retry mechanism can be implemented within a While loop for a certain number of tries or till Control Exists interaction returns true.
- If JavaScript is blocked on a machine due to Enhanced Security Configuration, then please enable it by connecting with your IT team. If this cannot be done, then add all application URLs that are to be automated in Trusted site list.




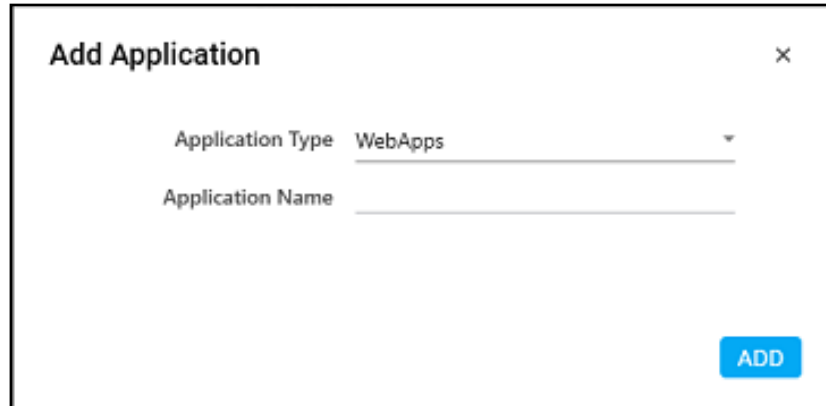
- Some applications may have an embedded iframe (such as Google Ads) which may inject 3rd party tracking scripts which may throw frequent JavaScript errors. When launched using Selenium in Microsoft Edge with IE Compatibility, these scripts within the iframe may cause the application to never attain a ready state. Due to this, Selenium is unable to perform any automation on such sites.

Note: Support for any of these limitations will be provided using Support Tickets

Configuring a Web Application

To configure a web application:

1. In the **Admin** menu, click the  (**Add Application**) icon to add an application. The **Add Application** dialog box appears.



2. In the Application Type list, select WebApps.
3. In the **Application Name** field, enter a desired name for the application you want to add. The name must not contain any special character or space.
4. Click **ADD**. A message for the successful addition of the application is displayed.
5. Close the displayed message and enter the following details in the **Application Properties** pane. The mandatory fields are highlighted in a red box.

Property Name	Usage
Details	
Login URL	The login URL of the web application you want to access. This is a mandatory field. <div style="background-color: #0070c0; color: white; padding: 2px; margin-top: 5px;">Note: The login URL can consist maximum of 256 characters.</div>
Search URL	The URL of the page that appears immediately post login. This is a mandatory field.
Display	
Application Description	A small description of the added web application.
Display Name	The desired name of the web application entered by the user. This is a mandatory field.
General	
Application Name	Name of the application entered at the time of adding the application. It auto-populates as per the name of the application entered while configuring the application.

Property Name	Usage
Application Type	Type of application selected at the time of adding the application. It auto-populates as per the application type selected while configuring the application.
Misc	
Is Application Tracked	It is applicable only for AssistEdge Engage. Signifies if the application is tracked for the time spent by Engage during the process execution. It is used for activity tracking reports and tracking time spent on each application in the execution dashboard reports. By default, it is selected.
Misc Java	
Environment Type	Determines the type of the application configured. Available options are: <ul style="list-style-type: none"> ▪ WINDOW_CLASS_BASED - is platform-dependent which implies web components are displayed based upon the view of the windows operating system. ▪ EVENT_BASED - is event-dependent which implies web components respond based on the actions of the user or the system. By default, WINDOW_CLASS_BASED is set. This field is editable only if Has Java Component check box is selected.
Has Java Component	Signifies if the application has Java components.
Navigation URL	
URL1/URL2/URL3	Enables navigation of a process flow from the current URL to the specified URL in this field. You must use Navigate To activity inside the Application activity to use this URL for navigation.
Preferences	
Is Launch Minimised	Signifies if the application window is minimized.

Property Name	Usage
Preferred Browser	<p>The browser you prefer to launch the web application. By default, preference is set to InternetExplorer. Available options are:</p> <ul style="list-style-type: none"> ▪ InternetExplorer ▪ FireFox ▪ Chrome ▪ InternetExplorerSelenium ▪ MicrosoftEdge ▪ MicrosoftEdgeWithIECompatibility <div style="background-color: #0070c0; color: white; padding: 10px; margin-top: 10px;"> <p>Note:</p> <p>Internet Explorer (IE) applications with document mode 5/7/8 are supported by the IE plugin. Selenium plugin InternetExplorerSelenium uses the IE web driver to automate the web applications utilizing IE.</p> </div>
Reset	
Is Reset Required	Signifies if the application reset needs to be done after each automation is executed. If it is not selected, the application does not reset.
Is Sequential Reset	Signifies if the application reset needs to be done one after another in no particular order. If it is selected, it is mandatory to have a Reset Process for that particular application created and used inside the application.
Sequential Launch/SignIn	
Is Sequential SignIn	Signifies if signing in to the application is required by the automation process in a sequence in which the application is used in the process. If it is selected, it is mandatory to have a SignIn Process for that particular application created and used inside the application.

6. Click the  (**Save Properties**) icon to save the application.


The web application is configured.

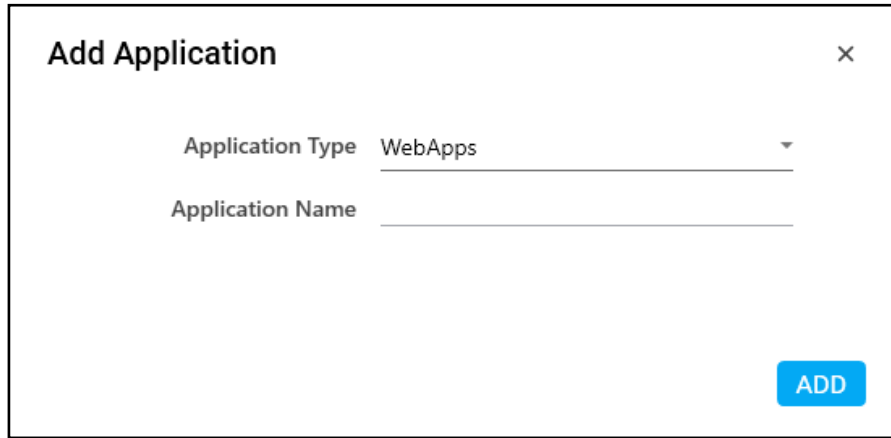
MainframeApps

It allows you to configure mainframe applications in Automation Studio.

In addition to the Mainframe emulators, Attachmate Extra Xtreme, Passport, Attachmate Reflection and EHLLAPI, Studio supports the Rumba Emulator 9.5 series.

To configure a mainframe application:

1. In the **Admin** menu, click the  (**Add Application**) icon to add an application. The **Add Application** dialog box appears.



2. In the Application Type list, select MainframeApps.
3. In the **Application Name** field, enter a desired name of the mainframe application you want to add. The name must not contain any special character or space.
4. Click **ADD**. A message for successful addition of the application is displayed.
5. Close the displayed message and enter the following details in the **Application Properties** pane. Mandatory fields are highlighted in a red box.

Property Name	Usage
Details	
Emulator	The software to access mainframe application and emulates the mainframe system on your desktop. Supported emulators are: <ul style="list-style-type: none"> ▪ AttachmateExtra ▪ Passport ▪ ReflectionUnixVMS ▪ AttachmateReflectionIBM ▪ ReflectionOpenSystem ▪ Rumba ▪ EHLLAPI

Property Name	Usage
SessionPath	The path of the session file that contains all the necessary details required to connect to the mainframe terminal.
This field appears only if Emulator selected is Rumba or EHLAPI .	
Session Short Name	<p>The short name of the session, as set in the emulator. To retrieve session short name in various emulators:</p> <ul style="list-style-type: none"> ▪ Rumba- Open Rumba emulator and navigate to Menu > Options > API ▪ Reflection- Open Reflection workspace and navigate to File > Settings > Document Settings > Configure Advanced Connection Settings ▪ Extra X-treme – Open Extra X-treme and navigate to Option > Global Preferences under Advanced tab ▪ Bluezone – Open Bluezone emulator and navigate to Session > Configure > API
Display	
Application Description	A small description of the mainframe application added.
Display Name	The desired name of the mainframe application entered by the user. This is a mandatory field.
General	
Application Name	Name of the mainframe application entered at the time of adding the application. It auto populates as per the name of the application entered while configuring the application.
Application Type	Type of application selected at the time of adding the application. It auto populates as per application type selected while configuring the application.
Misc	
Is Application Tracked	<p>It is applicable only for AssistEdge Engage.</p> <p>Signifies if the application is tracked for the time spent by Engage during the process execution. It is used for activity tracking report and tracking time spent on each application in the execution dashboard reports. By default, it is selected and can be edited.</p>
Is multiInstance allowed	<p>It has been decommissioned now.</p> <p>It was applicable only for AssistEdge SE (until version 8.6) which is AssistEdge Engage now. Signifies if multiple instances of SE (now Engage) processes can be initiated on a single user machine.</p>
Reset	

Property Name	Usage
Is Reset Required	Signifies if the application reset needs to be done after each automation is executed. If it is not selected, the application does not reset.
Is Sequential Reset	Signifies if the application reset needs to be done one after other in no particular order. If it is selected, it is mandatory to have a Reset Process for that particular application created and used inside the application.
Sequential Launch/SignIn	
Is Sequential SignIn	Signifies if signing in to the application is required by the automation process in a sequence in which the application is used in the process. If it is selected, it is mandatory to have a SignIn Process for that particular application created and used inside the application.


- Click the  (**Save Properties**) icon to save the application.

The mainframe application is configured.

WindowsApps

It allows you to configure Windows thick client (.exe files, the executable file for Microsoft Windows) such as Notepad, Command Prompt and others in Automation Studio.

To configure a windows application:

- In the **Admin** menu, click the  (**Add Application**) icon to add an application. The **Add Application** dialog box appears.

Add Application ×

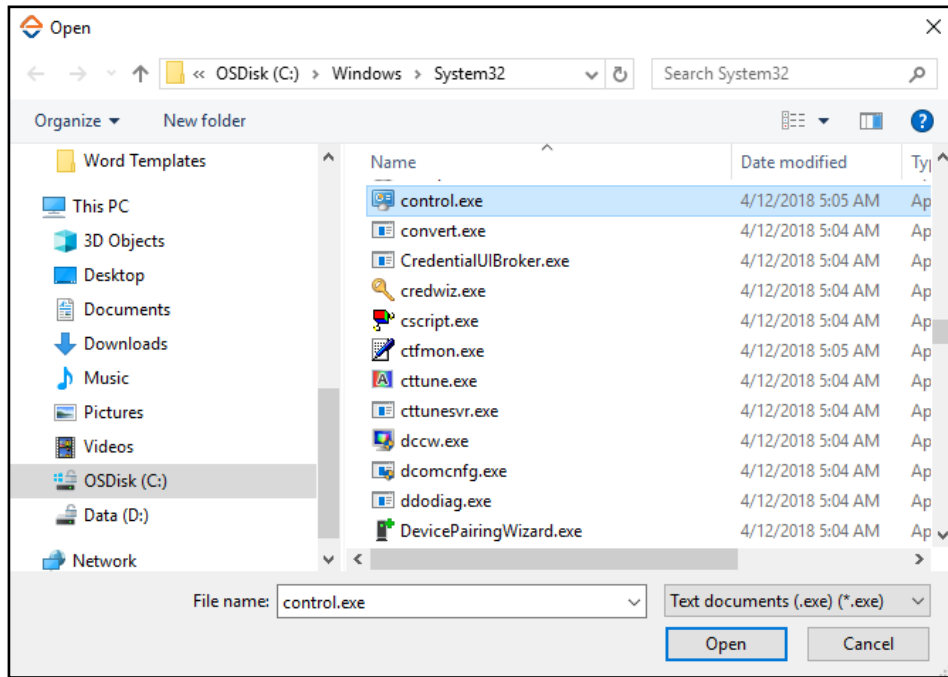
Application Type ▼

Application Name

ADD

- In the Application Type list, select **WindowsApps**.
- In the **Application Name** field, enter a desired name of the EXE application you want to add. The name must not contain any special character or space.

- Click **ADD**, navigate to the file location and the Click **Open**.



A message for successful addition of the application is displayed.

- Close the displayed message and enter the following details in the **Application Properties** pane. Mandatory fields are highlighted in a red box.

Property Name	Usage
Details	
Path	The path of the EXE file available on the system.
Display	
Application Description	A small description of the EXE file added.
Display Name	The desired name of the application, entered by the user. This is a mandatory field.
General	
Application Name	Name of the application entered at the time of adding the application. It auto populates as per the name of the application entered while configuring the application.
Application Type	Type of application selected at the time of adding the application. It auto populates as per the application type selected while configuring the application.

Property Name	Usage
Misc	
Additional Parameters	Enter the path of the file (along with its name) that you want to open within the application. Specify the file name in double quotes, if there are spaces in the file name.
Launch Time Out(in ms)	Signifies the maximum time out value for the launch of the application. The time out time must me in milliseconds. Default value is set to 5000ms.
Reset	
Is Reset Required	Signifies if the application reset needs to be done after each automation is executed. If it is not selected, the application does not reset.
Is Sequential Reset	Signifies if the application reset needs to be done one after other in no particular order. If it is selected, it is mandatory to have a Reset Process for that particular application created and used inside the application.
Sequential Launch/SignIn	
Is Sequential SignIn	Signifies if signing in to the application is required by the automation process in a sequence in which the application is used in the process. If it is selected, it is mandatory to have a SignIn Process for that particular application created and used inside the application.
Windows 10	
Launching Window Caption	The title of the window launched.


6. Click the  (**Save Properties**) icon to save the application.

The window application is configured.

WebAPI

It allows you to configure REST, SOAP, and BAPI APIs (Application Programming Interface) in Automation Studio.

To configure a windows application:

1. In the **Admin** menu, click the  (**Add Application**) icon to add an application. The **Add Application** dialog box appears.

Add Application ✕

Application Type WebAPI

Application Name

ADD

2. In the **Application Type** list, select **WebAPI**.
3. In the **Application Name** field, enter a desired name of the windows application you want to add. The name must not contain any special character or space.
4. Click **ADD**. A message for successful addition of the application is displayed.
5. Close the displayed message and enter the following details in the **Application Properties** pane. Mandatory fields are highlighted in a red box.

Property Name	Usage
Details	
Login URL	The login URL of the API you want to access. This is a mandatory field. <div style="background-color: #007bff; color: white; padding: 5px; margin-top: 5px;">Note: Do not provide the input path or the query parameter to the URL of the REST API added to Automation Studio.</div>
SignIn	Signifies if sign in for the application is required.
Type	The type of API added. Available options are: <ul style="list-style-type: none"> ▪ BAPI ▪ REST ▪ SOAP
Display	
Application Description	A small description of the application added.

Display Name	The desired name of the application entered by the use. This is a mandatory field.
General	
Application Name	Name of the API entered at the time of adding the application. It auto populates as per the name of the application entered while configuring the application.


- Click the  (**Save Properties**) icon to save the application.

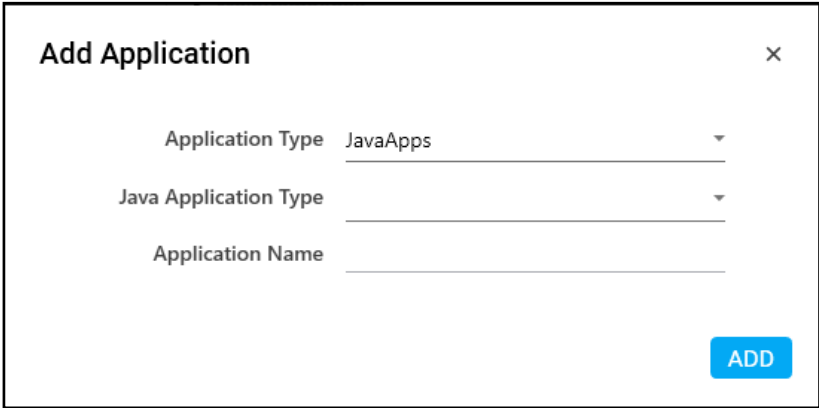
The API is configured.

JavaApps


It allows you to configure Java application in Automation Studio.

To configure a Java application:

- In the **Admin** menu, click the  (**Add Application**) icon to add an application. The **Add Application** dialog box appears.



- In the Application Type list, select JavaApps.
- In the **Java Application Type** list, select the type of Java application. Available options are:
 - Java
 - JNLP Application
 - Applet
- In the **Application Name** field, enter a desired name of the Java application you want to add. The name must not contain any special character or space.


5. Choose the environment file path as either windows class-based or event based.
6. Click the  (**Save Properties**) icon to save the application.

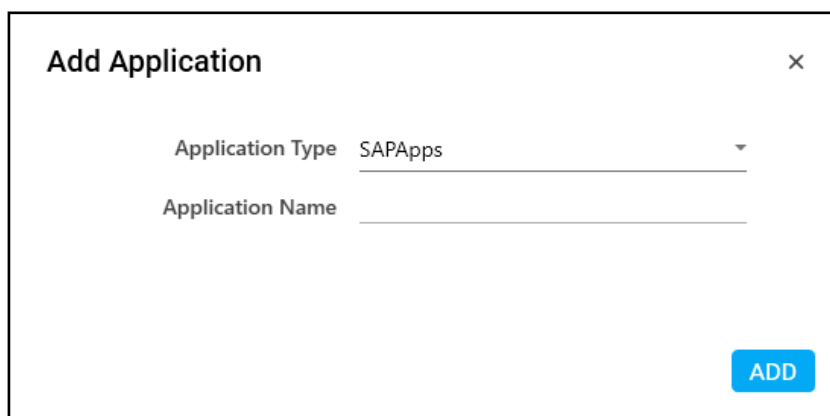
The Java application is configured.

SAPApps

It allows you to configure SAP thick client applications in Automation Studio. A thick client provides full functionality of the application without having a server communication.

To configure a SAP application:

1. In the **Admin** menu, click the  (**Add Application**) icon to add an application. The **Add Application** dialog box appears.



The screenshot shows a dialog box titled "Add Application" with a close button (X) in the top right corner. Inside the dialog, there are two input fields: "Application Type" with a dropdown menu currently showing "SAPApps", and "Application Name" with a text input field. A blue button labeled "ADD" is positioned at the bottom right of the dialog.

2. In the Application Type list, select **SAPApps**.
3. In the **Application Name** field, enter a desired name of the SAP application you want to add. The name must not contain any special character or space.
4. Click **ADD**. A message for successful addition of the application is displayed.
5. Close the displayed message and enter the following details in the **Application Properties** pane. Mandatory fields are highlighted in a red box.

Property Name	Usage
Details	
Connection Name	The name of the active enterprise SAP connection.
Display	
Application Description	A small description of the SAP application added.
Display Name	The desired name of the SAP application entered by the user. This is a mandatory field.
General	
Application Name	Name of the web application entered at the time of adding the application. It auto populates as per the name of the application entered while configuring the application.
Application Type	Type of application selected at the time of adding the application. It auto populates as per the application type selected while configuring the application.
Launch	
Is Startup Launch Required	Signifies if the SAP application must be launched separately and not to attach to an already launch application.
Reset	
Is Reset Required	Signifies if the application reset needs to be done after each automation is executed. If it is not selected, the application does not reset.
Is Sequential Reset	Signifies if the application reset needs to be done one after other in no particular order. If it is selected, it is mandatory to have a Reset Process for that particular application created and used inside the application.
Sequential Launch/SignIn	
Is Sequential SignIn	Signifies if signing in to the application is required by the automation process in a sequence in which the application is used in the process. If it is selected, it is mandatory to have a SignIn Process for that particular application created and used inside the application.

6. Click the  (**Save Properties**) icon to save the application.

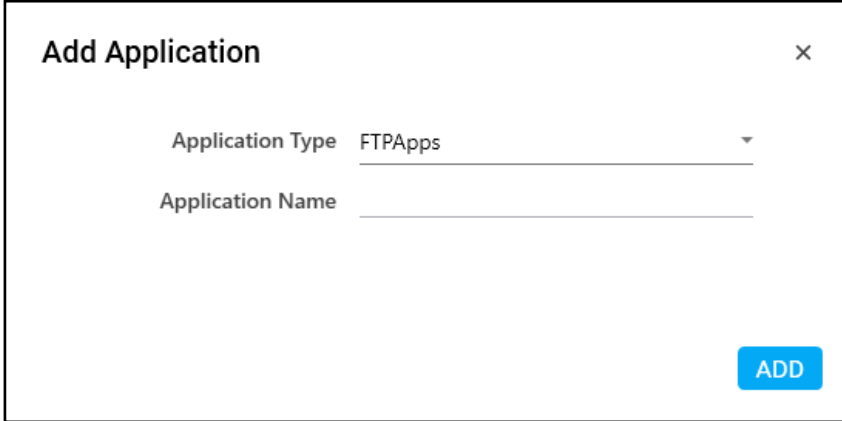
The SAP application is configured.

FTPApps

It allows you to configure FTP and SFTP server in Automation Studio.

To configure an FTP or a SFTP server:

1. In the **Admin** menu, click the  (**Add Application**) icon to add an application. The **Add Application** dialog box appears.



2. In the Application Type list, select **FTPApps**.
3. In the **Application Name** field, enter a desired name of the FTP server you want to add. The name must not contain any special character or space.
4. Click **ADD**. A message for successful addition of the application is displayed.
5. Close the displayed message and enter the following details in the **Application Properties** pane. Mandatory fields are highlighted in a red box.

Property Name	Usage
Display	
Application Description	A small description of the FTP server added.
Display Name	User entered, desired name of the FTP server to display. Is a mandatory field.
General	
Application Name	Name of the FTP server entered at the time of adding the application. It auto populates as per the name of the application entered while configuring the application.
Application Type	Type of application selected at the time of adding the application. It auto populates as per the application type selected while configuring the application.
Launch	
Is Startup Launch Required	Signifies if the FTP application must be launched separately and not to attach to an already launch application.

Property Name	Usage
Reset	
Is Reset Required	Signifies if the application reset needs to be done after each automation is executed. If it is not selected, the application does not reset.
Is Sequential Reset	Signifies if the application reset needs to be done one after other in no particular order. If it is selected, it is mandatory to have a Reset Process for that particular application created and used inside the application.
Sequential Launch/SignIn	
Is Sequential SignIn	Signifies if signing in to the application is required by the automation process in a sequence in which the application is used in the process. If it is selected, it is mandatory to have a SignIn Process for that particular application created and used inside the application.
ServerDetails	
Server IP	The FTP server IP address.
Server Port	The FTP port number (if the FTP port is using a different port other the default port, 21).
Transfer Type	The type of network protocol used for transfer of computer files between a client and a server on a network. Available options are: <ul style="list-style-type: none"> ▪ FTP- standard protocol used for file transfer. ▪ SFTP- protocol used for secure file transfer over secure shell.
SignIn	
SignIn	Signifies if sign in to the server is required. By default, it is set to true and cannot be edited.


- Click the  (**Save Properties**) icon to save the application.

The FTP server is configured.

CognitiveApps

It allows you to configure cognitive services in Automation Studio.

To configure a cognitive service:

- In the **Admin** menu, click the  (**Add Application**) icon to add an application. The **Add Application** dialog box appears.

2. In the Application Type list, select CognitiveApps.
3. In the **Application Name** field, enter a desired name of the cognitive service you want to add. The name must not contain any special character or space.
4. Click **ADD**. A message for successful addition of the application is displayed.
5. Close the displayed message and enter the following details in the **Application Properties** pane. Mandatory fields are highlighted in a red box.

Property Name	Usage
Display	
Application Description	A small description of the web application added.
Display Name	The desired name of the application entered by the user. This is a mandatory field.
General	
Application Name	Name of the application entered at the time of adding the application. It auto populates as per the name of the application entered while configuring the application.
Application Type	Type of application selected at the time of adding the application. It auto populates as per the application type selected while configuring the application.
Launch	
Is Startup Launch Required	Signifies if the cognitive application must be launched separately and not to attach to an already launch application.
License Type	The type of license used for the cognitive service. Available options are: <ul style="list-style-type: none"> ▪ BYOL - Buy Your Own License ▪ EVOL - EV Owned License

Property Name	Usage
Resource Group	Contains the cognitive service resources such as the database, virtual machines and so on.
Reset	
Is Reset Required	Signifies if the application reset needs to be done after each automation is executed. If it is not selected, the application does not reset.
Is Sequential Reset	Signifies if the application reset needs to be done one after other in no particular order. If it is selected, it is mandatory to have a Reset Process for that particular application created and used inside the application.
Sequential Launch/SignIn	
Is Sequential SignIn	Signifies if signing in to the application is required by the automation process in a sequence in which the application is used in the process. If it is selected, it is mandatory to have a SignIn Process for that particular application created and used inside the application.
SignIn	
Login Type	Signifies the way sign-in to the cognitive service must be done. Available options are: <ul style="list-style-type: none"> ▪ Key - If sign-in must be done using the key ▪ Credentials - If sign-in must be done using the sign-in credentials
SignIn	Signifies if sign in to the cognitive service is required. Be default, it is set to True and cannot be edited.


- Click the  (**Save Properties**) icon to save the application.

The cognitive service is configured.

SMTPApps

It allows you to configure the SMTP server in Automation Studio.

To configure a SMTP server:

- In the **Admin** menu, click the  (**Add Application**) icon to add an application. The **Add Application** dialog box appears.

Add Application ✕

Application Type SMTPApps ▼

Application Name

ADD

2. In the Application Type list, select SMTPApps.
3. In the **Application Name** field, enter a desired name of the SMTP server you want to add. The name must not contain any special character or space.
4. Click **ADD**. A message for successful addition of the application is displayed.
5. Close the displayed message and enter the following details in the **Application Properties** pane. Mandatory fields are highlighted in a red box.

Property Name	Usage
Details	
Domain	The domain name of the SMTP server.
Host Name	The hostname or IP address of the SMTP server.
IMAP Port No.	Port number of the IMAP protocol.
POP3 Port No.	Port number of the POP3 protocol.
Single Attachment Size Limit (MB)	The maximum file size of attachment allowed in MB.
SMTP Port No.	The port used to connect to the SMTP server.
SSLEnabled	Signifies if SSL is enabled. By default, it is set to False .
Total Attachment Size Limit (MB)	Total file size of the attachment allowed in MB.
Validate Channel	If user id used to connect to the server is same as the from email address in Send mail activity, then set this value to true, else to false. By default, it is set to False.

Property Name	Usage
Display	
Application Description	A small description of the FTP or the SFTP application added.
Display Name	The desired name of the application entered by the user. This is a mandatory field.
General	
Application Name	Name of the application entered at the time of adding the application. It auto populates as per the name of the application entered while configuring the application.
Application Type	Type of the application selected at the time of adding the application. It auto populates as per the application type selected while configuring the application.
Reset	
Is Reset Required	Signifies if the application reset needs to be done after each automation is executed. If it is not selected, the application does not reset.
Is Sequential Reset	Signifies if the application reset needs to be done one after other in no particular order. If it is selected, it is mandatory to have a Reset Process for that particular application created and used inside the application.
Sequential Launch/SignIn	
Is Sequential SignIn	Signifies if signing in to the application is required by the automation process in a sequence in which the application is used in the process. If it is selected, it is mandatory to have a SignIn Process for that particular application created and used inside the application.
SignIn	
SignIn	Signifies sign in to the SMTP server is required. Be default, it is set to True and cannot be edited.


6. Click the  (**Save Properties**) icon to save the application.

The SMTP server is configured.

DatabaseApps

It allows you to configure database application to Automation Studio.

To configure a database application:

1. In the **Admin** menu, click the  (**Add Application**) icon to add an application. The **Add Application** dialog box appears.

Add Application ×

Application Type DatabaseApps ▼

Application Name

ADD

2. In the Application Type list, select **DatabaseApps**.
3. In the **Application Name** field, enter a desired name of the database application you want to add. The name must not contain any special character or space.
4. Click **ADD**. A message for successful addition of the application is displayed.
5. Close the displayed message and enter the following details in the **Application Properties** pane. Mandatory fields are highlighted in a red box.

Property Name	Usage
Details	
DB Type	The type of database application configured. Supported types are: <ul style="list-style-type: none"> ▪ SQL ▪ Oracle ▪ MySQL ▪ Excel
Authentication Type(SQL)	The type of authentication configured in the SQL database. Appears when DB Type is set to SQL . Available options are: <ul style="list-style-type: none"> ▪ SQL ▪ Windows
DB Server	The server's name where database is hosted for database storage and retrieval.
DB Port	The port number to connect to the database server.

Property Name	Usage
DB Name	The name of the database created.
SID(Oracle)	The unique name of the Oracle database instance. Appears when DB Type is set to Oracle .
Excel Directory Path (Example - D:\folder)	The path of the excel application saved on the system. For example, D:\folder.
Display	
Application Description	A small description of the web application added.
Display Name	The desired name of the application entered by the user. This is a mandatory field.
General	
Application Name	Name of the application entered at the time of adding the application. It auto populates as per the name of the application entered while configuring the application.
Application Type	Type of the application selected at the time of adding the application. It auto populates as per the application type selected while configuring the application.
Reset	
Is Reset Required	Signifies if the application reset needs to be done after each automation is executed. If it is not selected, the application does not reset.
Is Sequential Reset	Signifies if the application reset needs to be done one after other in no particular order. If it is selected, it is mandatory to have a Reset Process for that particular application created and used inside the application.
Sequential Launch/SignIn	
Is Sequential SignIn	Signifies if signing in to the application is required by the automation process in a sequence in which the application is used in the process. If it is selected, it is mandatory to have a SignIn Process for that particular application created and used inside the application.
SignIn	
SignIn	Signifies if sign in to the database application is required. Be default, it is set to True and cannot be edited.


6. Click the  (**Save Properties**) icon to save the application.

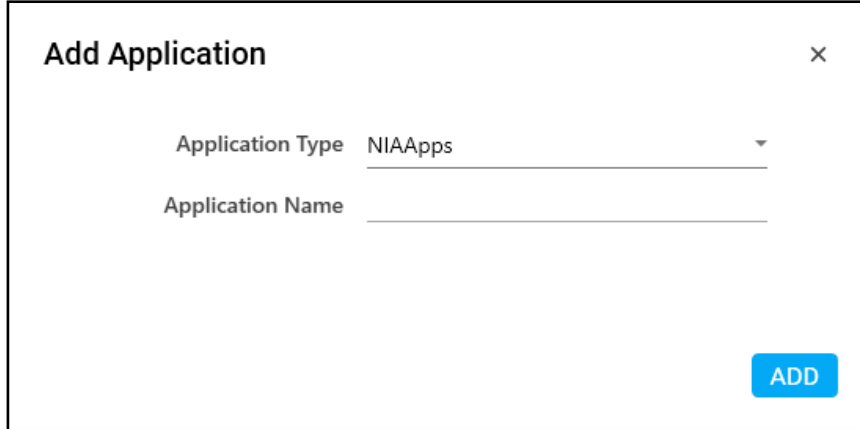
The database application is configured.

NIAApps

It allows you to configure NIA Vision application.

To configure a NIA application:

1. In the **Admin** menu, click the  (**Add Application**) icon to add an application. The **Add Application** dialog box appears.



2. In the Application Type list, select **NIAApps**.
3. In the **Application Name** field, enter a desired name of the NIA application you want to add. The name must not contain any special character or space.
4. Click **ADD**. A message for successful addition of the application is displayed.
5. Close the displayed message and enter the following details in the **Application Properties** pane. Mandatory fields are highlighted in a red box.

Property Name	Usage
Details	
Url	EdgeVerve hosted URL where API is called to use the NIA services.
Display	
Application Description	A small description of the NIA Vision application added.
Display Name	The desired name of the application entered by the user. This is a mandatory field.
General	

Property Name	Usage
Application Name	Name of the application entered at the time of adding the application. It auto populates as per the name of the application entered while configuring the application.
Application Type	Type of the application selected at the time of adding the application. It auto populates as per the application type selected while configuring the application.


- Click the  (**Save Properties**) icon to save the application.

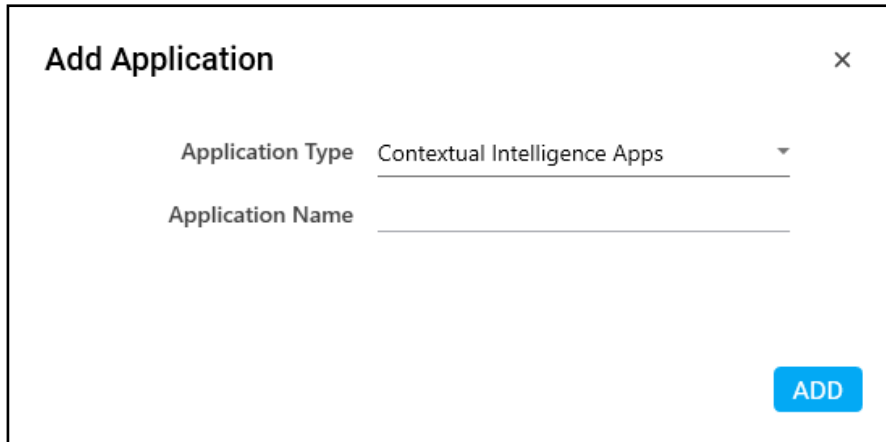
The NIA Vision application is configured.

Contextual Intelligence Apps

It allows you to configure server where machine learning models are hosted.

To configure a contextual intelligence:

- In the **Admin** menu, click the  (**Add Application**) icon to add an application. The **Add Application** dialog box appears.



- In the Application Type list, select **Contextual Intelligence Apps**.
- In the **Application Name** field, enter a desired name of the contextual intelligence application you want to add. The name must not contain any special character or space.
- Click **ADD**. A message for successful addition of the application is displayed.
- Close the displayed message and enter the following details in the **Application Properties** pane. Mandatory fields are highlighted in a red box.

Property Name	Usage
Details	
Http Method	The http request method that uses the web services corresponding to the machine learning model that needs to be called. Available options are: <ul style="list-style-type: none"> ▪ GET- retrieves data from the server ▪ POST-sends data to the server creating new resource
Server URL	URL of the server where the machine learning models are deployed.
SignIn	Signifies if sign in to the server is required. By default, it is cleared and can be edited.
Display	
Application Description	A small description of the web application added.
Display Name	The desired name of the application entered by the user. This is a mandatory field.
General	
Application Name	Name of the application entered at the time of adding the application. It auto populates as per the name of the application entered while configuring the application.
Application Type	Type of the application selected at the time of adding the application. It auto populates as per the application type selected while configuring the application.
Headers	
Accept	Defines the media type format of the output such as application/JSON. This information flows to the header of the Decision Model activity using this application.
Content-Type	Defines the media type format of the input such as application/XML. This information flows to the header of the Decision Model activity using this application.


6. Click the  (**Save Properties**) icon to save the application.

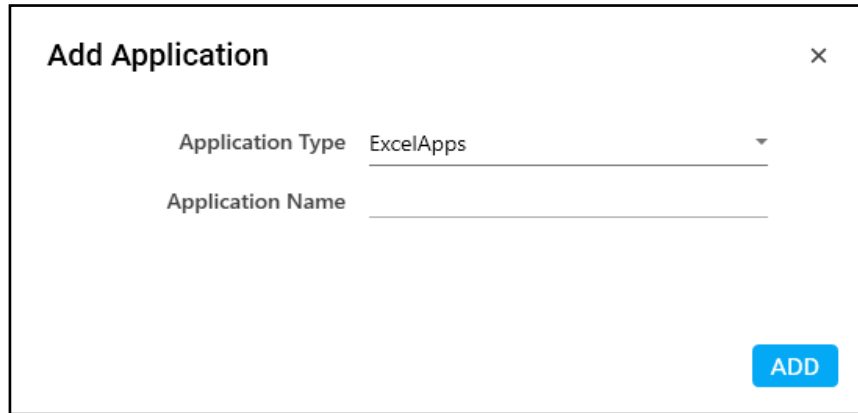
The cognitive intelligence application is configured.

ExcelApps

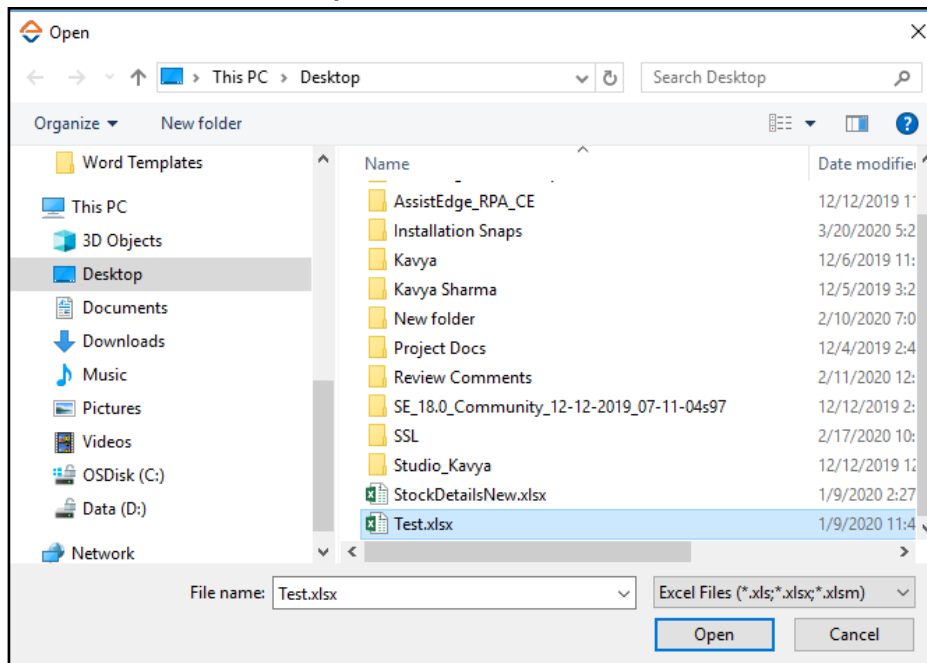
It allows you to configure excel application (excel file) in Automation Studio.

To configure an excel application:

1. In the **Admin** menu, click the  (**Add Application**) icon to add an application. The **Add Application** dialog box appears.



2. In the Application Type list, select **ExcelApps**.
3. In the **Application Name** field, enter a desired name of the excel application you want to add. The name must not contain any special character or space.
4. Click **ADD** and browse for the excel file. Click **Open**.



A message for successful addition of the excel application is displayed.

5. Close the displayed message and enter the following details in the **Application Properties** pane. Mandatory fields are highlighted in a red box.

Property Name	Usage
Details	
Path	The path of the excel file available on the system along with the file name.
Display	
Application Description	A small description of the excel application added.
Display Name	The desired name of the application entered by the user. This is a mandatory field.
General	
Application Name	Name of the application entered at the time of adding the application. It auto populates as per the name of the application entered while configuring the application.
Application Type	Type of the application selected at the time of adding the application. It auto populates as per the application type selected while configuring the application.
Misc	
Excel Visibility	Signifies if the configured Excel application needs to be loaded when automation is executed. If not selected, automaton is executed without loading configured Excel file on the screen.
Reset	
Is Reset Required	Signifies if the application reset needs to be done after each automation is executed. If it is not selected, the application does not reset.
Is Sequential Reset	Signifies if the application reset needs to be done one after other in no particular order. If it is selected, it is mandatory to have a Reset Process for that particular application created and used inside the application.
Sequential Launch/SignIn	
Is Sequential SignIn	Signifies if signing in to the application is required by the automation process in a sequence in which the application is used in the process. If it is selected, it is mandatory to have a SignIn Process for that particular application created and used inside the application.

6. Click the  (**Save Properties**) icon to save the application.

The excel application is configured.

5.2 Deployment

An automation process workflow must be deployed to send it to the server. From server, the deployed process is assigned to a robot for execution.

The Deployment tab allows you to deploy the published processes in Automation Studio. It also enables you to decommission the current version, restore the previous version and add a process as a processbot. Published as well as deployed processes are available in this tab. You can view details related to the process such as name of the process, its type, assigned profile, version and so on.


The version of the process displayed in the Deployment tab is the version that gets created while publishing the process. If you edit a deployed process, save, and publish the deployed process again. The version of the saved and published process post deployment gets incremental by one. Every version of the saved process that you publish is available for deployment.

Note:

- Only a published process can be deployed.
- A deployed process cannot be deleted.

Process Name	Process Type	Status	Profile	Published	PQM Score	Assisted
AutomobileCD	Process	Deployed	HRProfile	Version 1 by nutanankush.shinde @ 10:04 12 Aug 2021	Version 1 by nutanankush.shinde @ 10:05 12 Aug 2021	
TrainingEV	Process	Deployed	Training	Version 1 by nutanankush.shinde @ 13:19 11 Aug 2021	Version 1 by nutanankush.shinde @ 13:14 11 Aug 2021	
Simple1	Process	Deployed	HRProfile	Version 1 by nutanankush.shinde @ 12:42 11 Aug 2021	Version 1 by nutanankush.shinde @ 12:43 11 Aug 2021	
VCSDemo	Process	Published	TestPro	Version 1 by kavya.sharma @ 18:05 06 Aug 2021	n/a	
HelloWorld_VCS	Process	Published	EVDoc	Version 1 by kavya.sharma @ 18:05 06 Aug 2021	n/a	
WebTableExtractio...	Process	Published	Demo	Version 1 by kavya.sharma @ 18:05 06 Aug 2021	n/a	
HelloWorldVCS	Process	Published	Demo	Version 1 by kavya.sharma @ 18:05 06 Aug 2021	n/a	

5.2.1 Deploy a Process

1. In the **Deployment** tab, select the process you want to deploy and then click the  (Deploy) icon. The dialog box appears where you can enter details related to the deployment.

SAP Create PO Transpose ×

Deployed Version
n/a

Select Version
1 ▼

Manual Process Time			SLA		
Hour(s)	Min(s)	Sec(s)	Hour(s)	Min(s)	Sec(s)
00	00	00	00	00	00

Expected Execution Time

Hour(s)	Min(s)	Sec(s)
00	00	00

Comments

sample deployment

The process will now be deployed and will be available for automation to all robots and Attended Automation for the respective profile. Process backup will be stored in database. Please click confirm to continue.

Please note: You are about to deploy a process which uses an entity downloaded from AssistEdge Marketplace.


CANCEL
CONFIRM

2. In the **Select Version** list, select the version of the process to deploy. By default, only 1 is available for deployment for the first time you are deploying the process.
3. Enter the following details:
 - **Manual Process Time** - when a request in the process is executed manually, enter the expected process execution time in HH:MM:SS format. Hour(s) is in the range of 0-99 while Min(s) and Sec(s) are in the range 0-59. **Manual Process Time** is the average time taken by the process to be manually executed. This information is utilized at the reporting platform level, as a baseline, to gauge the saving on the process execution time due to automation.
 - **SLA** - when a process is executed by a robot, enter the expected SLA for the process in HH:MM:SS format. Hour(s) is in the range of 0-99 while Min(s) and Sec(s) are in the range 0-59. This information is used to calculate the service level (proportion of automation qualified requests which get executed within the SLA duration limit) of each process.
 - **Expected Execution Time** - the time that is expected for a robot to complete each transaction of the process. This information is used to notify the configured e-mail ID in case robot takes more time than expected execution time to execute the transaction.
4. In the **Comments** box, provide additional information related to the deployment.

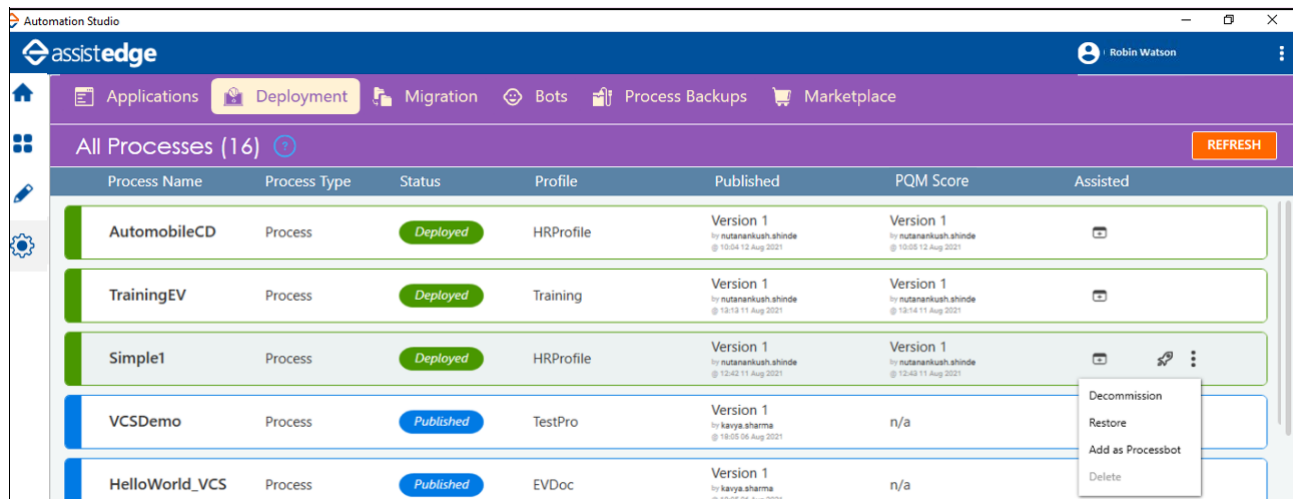
5. Click **CONFIRM**.





A message for successful deployment is displayed. The status of the published process changes to Deployed. While status of the deployed process remains unchanged.

A process backup is created automatically after deployment and appears in the Process Backups tab of Admin menu. It helps to retrieve data or review previous versions. It lets you save time taken in creating manual backups.

Use the  (More) icon to do one of the following:

- **Decommission** - decommissions the selected process
- **Add as a Processbot** - creates a processbot
- **Restore** - restores a previous version of the published or deployed process
- **Delete** - deletes the selected process




Process Name	Process Type	Status	Profile	Published	PQM Score	Assisted
AutomobileCD	Process	Deployed	HRProfile	Version 1 by mutanankush.shinde @ 10:04 12 Aug 2021	Version 1 by mutanankush.shinde @ 10:05 12 Aug 2021	
TrainingEV	Process	Deployed	Training	Version 1 by mutanankush.shinde @ 13:13 11 Aug 2021	Version 1 by mutanankush.shinde @ 13:14 11 Aug 2021	
Simple1	Process	Deployed	HRProfile	Version 1 by mutanankush.shinde @ 12:42 11 Aug 2021	Version 1 by mutanankush.shinde @ 12:43 11 Aug 2021	 
VCSDemo	Process	Published	TestPro	Version 1 by kavya.sharma @ 18:05 06 Aug 2021	n/a	
HelloWorld_VCS	Process	Published	EVDoc	Version 1 by kavya.sharma @ 18:05 06 Aug 2021	n/a	

5.2.2 Decommission and Delete a Process

Decommissioning a process temporarily or permanently removes a process from the server. Once you decommission a process, you can delete the process to permanently remove it. Deleting a used or irrelevant process helps to maintain sanctity of data in the system.


To decommission a process:

1. In the **Deployment** tab, select the process you want to decommission, and then click the  (**More**) icon.
2. Click **Decommission**. A confirmation message to decommission the process appears.
3. Click **OK**.

Status of the process changes to Decommissioned in the Deployment tab. In the Home tab, a decommissioned process appears with a red dot.

You can click the  (Deploy) icon again to make a decommissioned process active and available for use.

To delete a decommissioned process:

1. In the **Deployment** tab, select the deployed process you want to delete, and then click the  (**More**) icon .
2. Click **Delete**. A confirmation message to delete the process appears.
3. Click **OK**.

The process is permanently deleted from the database.


5.2.3 Add as Processbot

It allows you to create a processbot of the selected process that helps you to migrate the processbot from one environment to another. This is another way of migrating a process.

The processbot created in the Deployment tab appears in the Bots tab under Processbots pane. See [Processbots](#) section in the Bots tab to know more about processbots.

Note: Only a deployed process can be added as a processbot.

To add as processbot:

1. In the **Deployment** tab, select the process you want to add as a processbot, and then click the  (**More**) icon.
2. Click Add as Processbot. The Add as Processbot dialog box appears.

3. In the **Author** field, enter a name of the creator of the processbot.
4. In the **Description** box, enter information related to the processbot.
5. In the **Readme File Path**, browse the file that includes instruction related to usage of the processbot.
6. In the **Tags** field, enter a label of the processbot for identification. You can assign multiple processbot under same tag to keep them under one group for the ease of identification and access.
7. Click **SAVE**.


The process along with other sub-processes that are re-used in the primary process are added as a processbot.

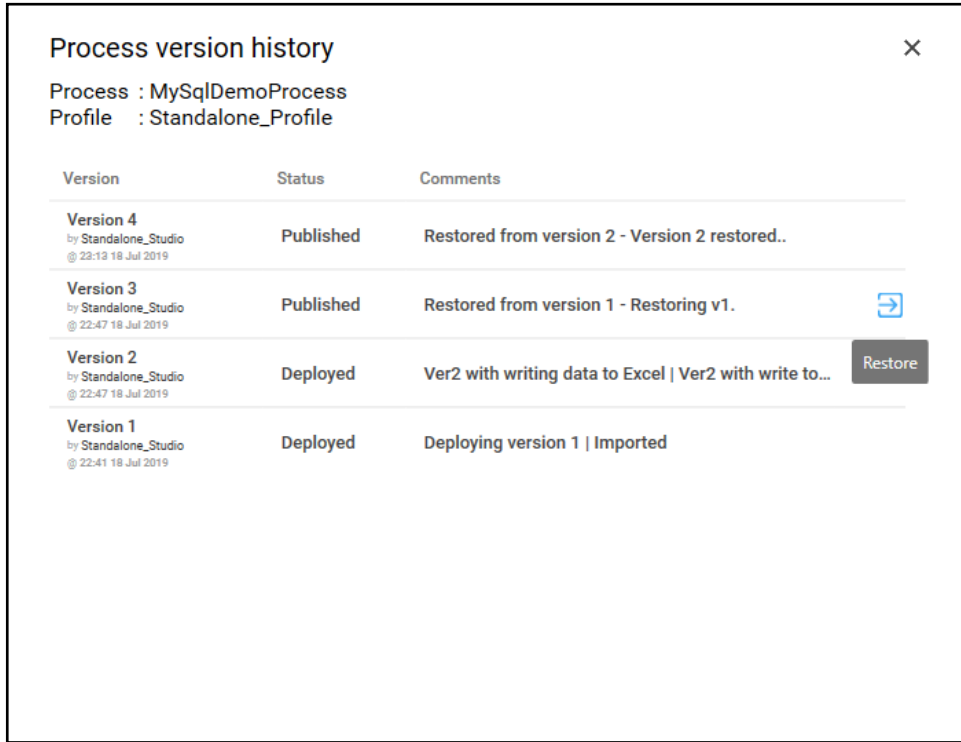
5.2.4 Restore a Process

You can roll back a published or a deployed process to one of the earlier versions in the Deployment tab.

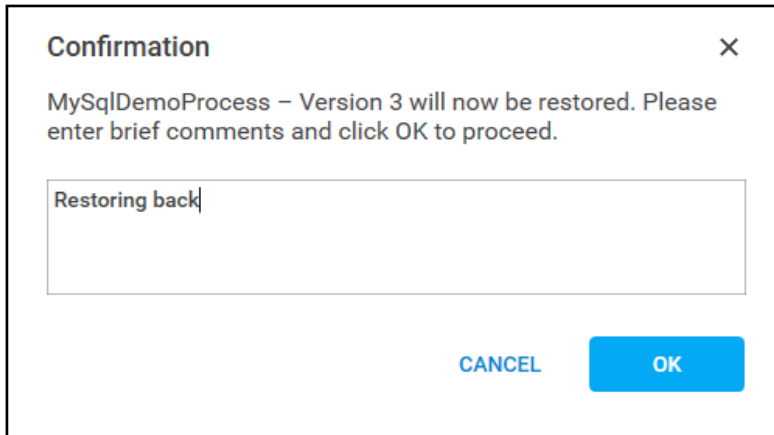
Note: The latest version of the published or deployed process cannot be restored.

To restore a process to previous version:

1. In the **Deployment** tab, select the process you want to restore to its previous, and then click the  **(More)** icon.
2. Click **Restore**. The **Process version history** dialog box appears.



3. Click the ↩ (**Restore**) icon. A confirmation box appears.



4. Enter a comment and the click **OK**.

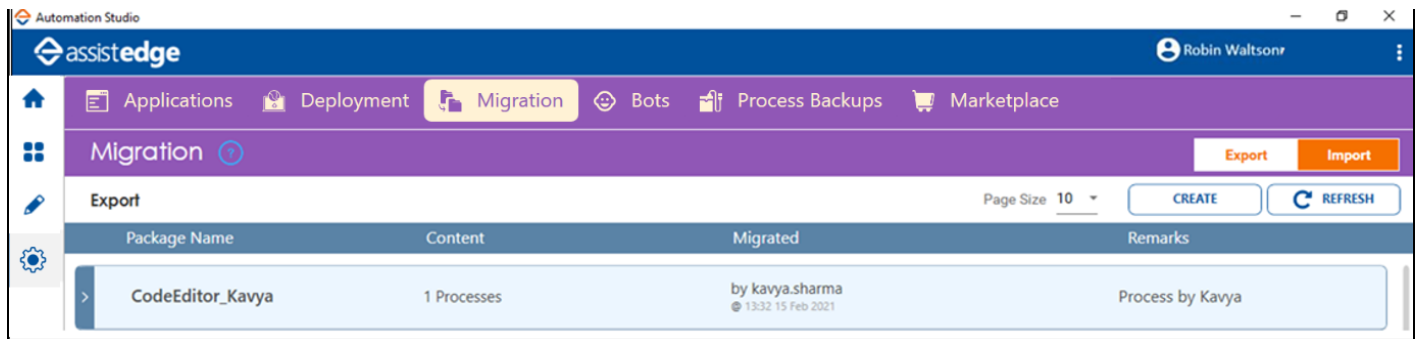
The process is restored to the selected previous version.

5.3 Migration

The Migration tab allows you to create packages of deployed processes and move from one environment to other. By default, Export is selected.

Note:

- Only a deployed process can be migrated.
- This option must be used to export and import processes from one environment to other. Process of one version must not be migrated to other version using this option. It is recommended to migrate the database in such scenarios where version of the product changes.
- **Backward compatibility** : In Automation studio, the ignore error will be set to NA by default when the processes are imported from older version than 17.5 to maintain the compatibility to process execution flow of older version. While importing the process from 17.5.2, you are required to add **ProcessImportVersion** key under **primarysettings** to enable the compatibility mode run for 17.5.2 processes.



The system allows you to do the following:

- [Export a Process](#)
- [Import a Process](#)

5.3.1 Export a Process

Process in Automation Studio can be exported from one environment to other environment as a zip file package. You can export one or multiple processes in a single package. The file package contains details related to the process like the dependent applications, code editors, Microbots, Test Cases, search criteria and others.

Note: Dependent processes exist for the DW activity, the Reuse activity, and the Trigger activity in the Automation studio.

These dependent processes are bundled with the parent process and are exported automatically. However, to ensure that the export process feature work as expected for the dependent processes of the DW activity and the Trigger activity, users must deploy the dependent process before executing the parent process.

To export a process:

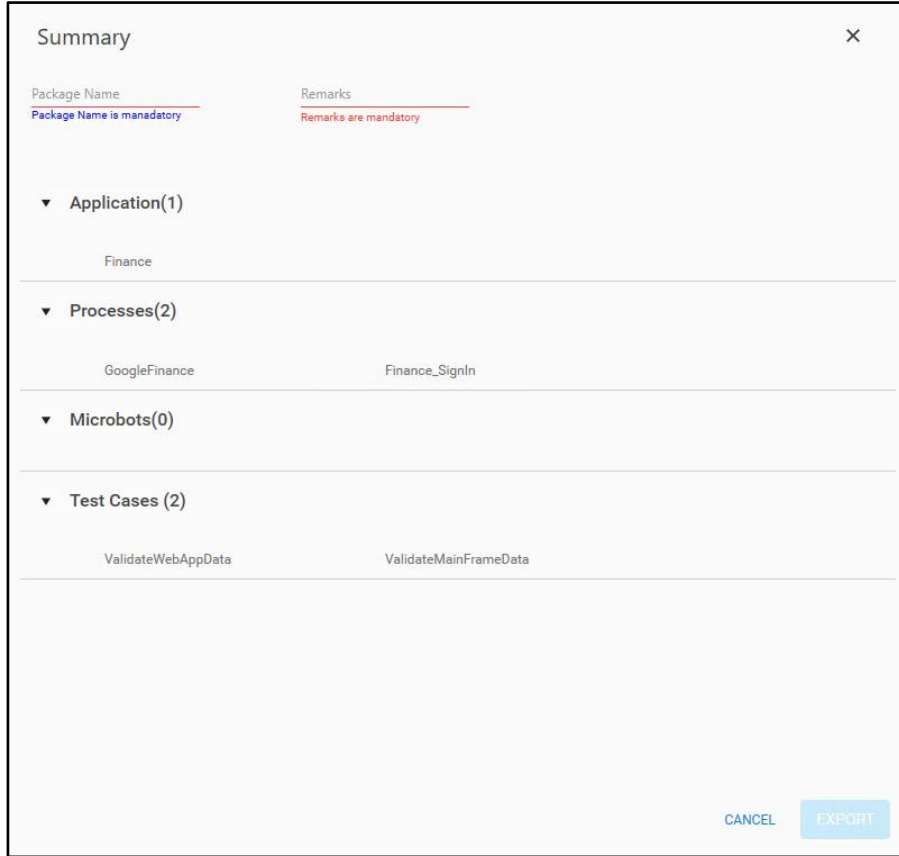
1. In the **Migration** tab, click **Export**.
2. Click **Create**. The **Create New Package** page opens displaying the deployed processes.

The screenshot shows the 'Automation Studio' interface with the 'Migration' tab selected. The 'Create New Package' page is displayed, showing a table of deployed processes. The table has the following columns: Process Name, Process Type, Profile, Deployed, and Add To Package. The data rows are as follows:

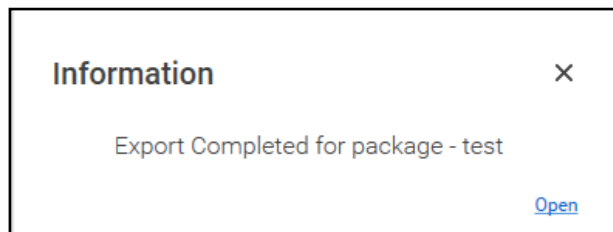
Process Name	Process Type	Profile	Deployed	Add To Package
Salarygraph	Process	EVDoc	Version 1 by Nutan Ankush Shinde(nutanankushshinde) @ 18:17 23 Apr 2021	<input type="checkbox"/>
UseCasedemo	Process	EVDoc	Version 1 by Parul Jain07 @ 17:40 16 Apr 2021	<input type="checkbox"/>
PDFdata	Process	EVDoc	Version 1 by Anupama Thombare @ 17:46 08 Apr 2021	<input type="checkbox"/>
WeatherReport	Process	EVDoc	Version 1 by Anupama Thombare @ 20:48 16 Mar 2021	<input type="checkbox"/>
LoadGen	Load Generator Process	LoadGenerator	Version 1 by Archana Muley(archana.muley) @ 23:17 05 Mar 2021	<input type="checkbox"/>
AdvLoopDataTable_LoadGen	Process	AEFinalProcesses	Version 1 by Archana Muley(archana.muley) @ 23:12 05 Mar 2021	<input type="checkbox"/>

At the bottom of the table, there are 'CANCEL' and 'EXPORT' buttons. A 'REFRESH' button is also visible at the top right of the table area.

- 3. Select one or more process, and then click **EXPORT**. The **Summary** dialog box appears displaying the information such as total number and name of the processes, application, and microbots to be exported.



- 4. In the **Package Name** field, enter a desired name of the package where the process gets exported.
- 5. In the **Remarks** field, enter information related to the package.
- 6. Click **EXPORT**.



A message for completion of the export appears along with a link where it gets exported and stored on the local machine.

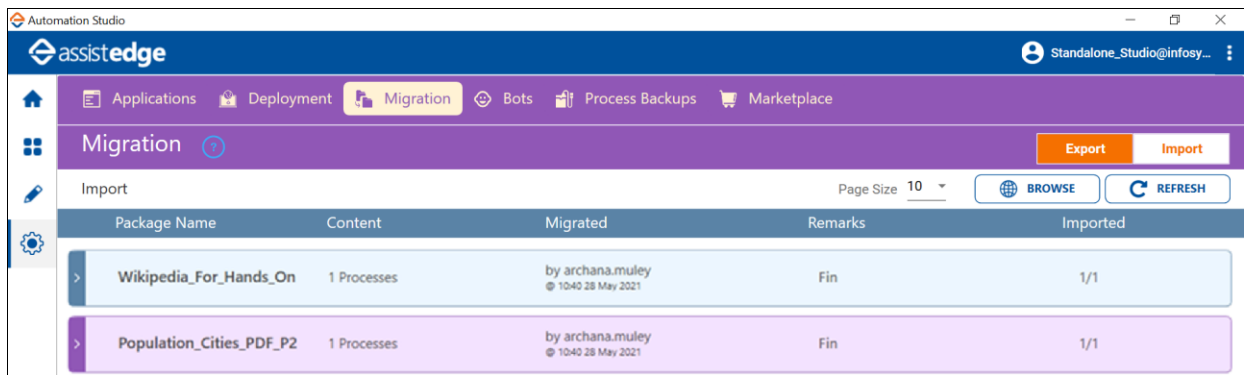
The package exported starts appearing in the Migration tab. You can expand the package and view the details of processes exported in the package. Additionally, you can view the associated Application Dependencies, Process Dependencies, Microbots, and Test Cases of the expanded package.

5.3.2 Import a Process

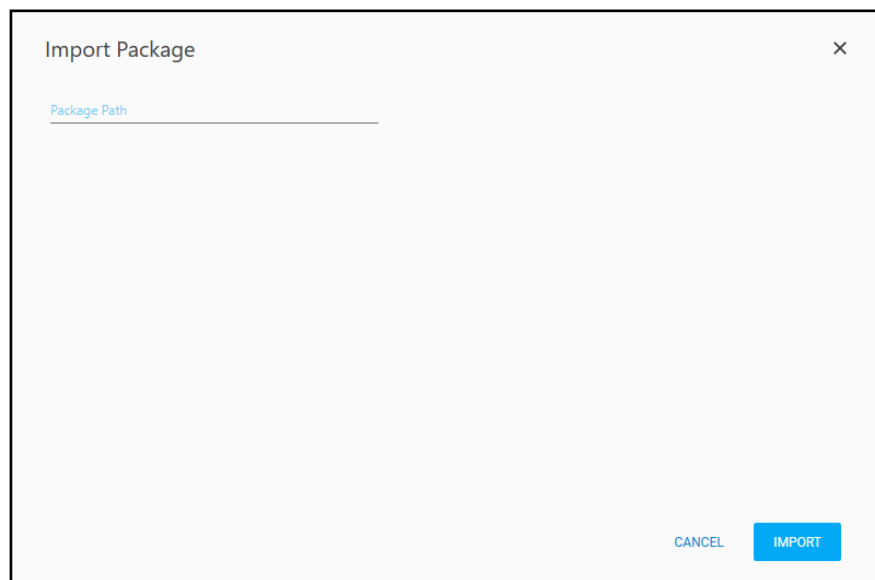
Process in Automation Studio can be imported from one environment to other environment as a zip file package. You can import one or multiple processes in a single package. The file package contains details related to the process like the dependent applications, code editors, Microbots, Test Cases, search criteria, and others.

To import a process:

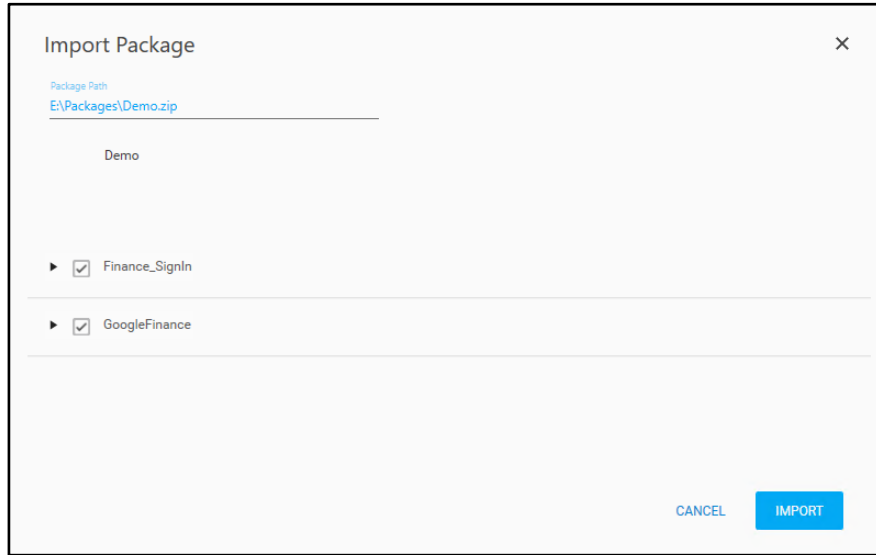
1. In the **Migration** tab, click **Import**.



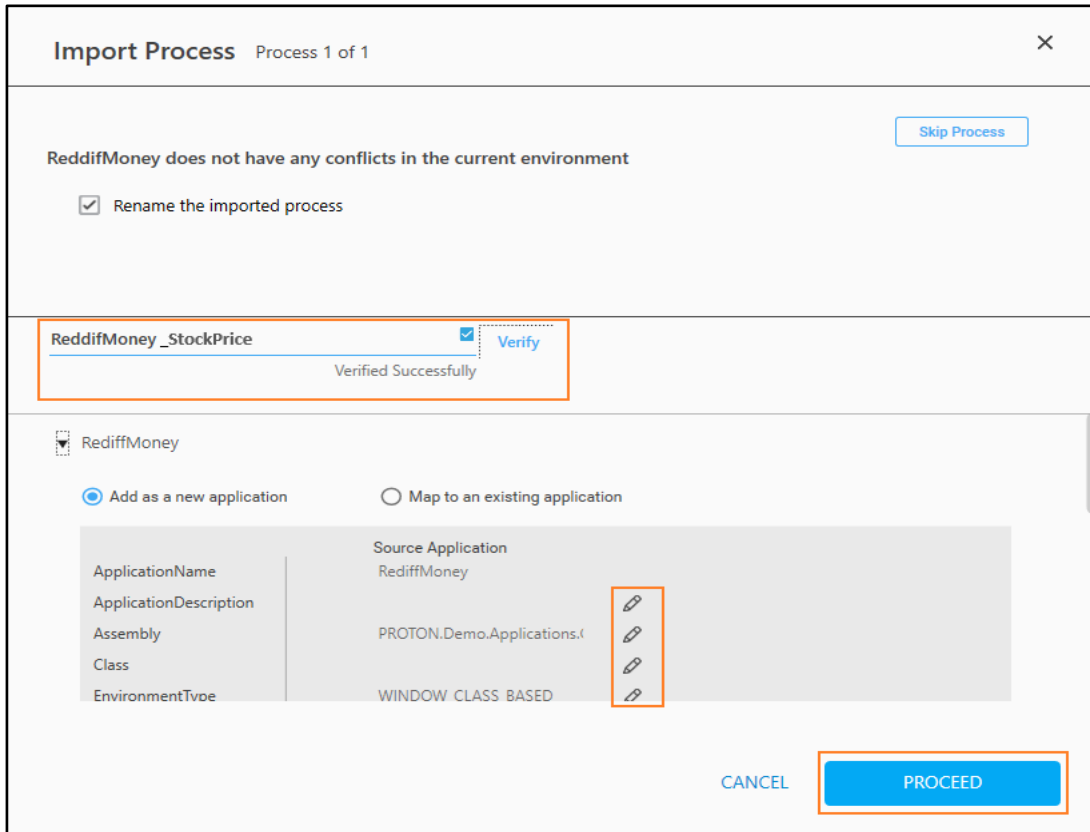
2. Click **Browse**. The **Import Package** dialog appears.



3. Click the **Package Path** field. Browse and select the process package zip folder to import.
4. Click **Open**. Select the process to be imported.
5. Click **IMPORT**. The **Import Process** dialog box appears.
6. Select the process to be imported. You can import single or multiple processes contained in the process zip folder. You can click the arrow beside the process to view the details related to the process selected. If multiple processes are being imported inside the zip folder, following options and screens appear for each process.




- 7. Click **IMPORT**. The **Import Process** dialog box appears. Following options appear based on the duplicity of the process:
 - If the process does not exist in the current environment, you can either rename the process (if required) or proceed with the same name. To rename the process:
 - i. Select the **Rename the imported process** check box. The **Verify** field appears.



- ii. Enter a name of your choice and click **Verify** to check duplicity of the process name. If no process exists in the environment with name entered, a message for successful verification appears.
 - iii. Click **PROCEED**.
8. If a process already exists in the current environment, you can select any of the available options as per your requirement:
- **Overwrite in the existing profile** - creates a copy of the existing process.
 - **Rename in the existing profile** - renames the process, and then imports the package.
 - **Re-map process to different profile** - maps the process with another profile.

The screenshot shows a dialog box titled "Import Process" with a close button (X) in the top right corner. The main text reads: "ReddiffMoney already exists in the current environment under EVDoc. How would you like to resolve this conflict?". There are three radio button options: "Overwrite in the existing profile" (selected), "Rename in the existing profile", and "Re-map process to different profile". A "Skip Process" button is in the top right. Below the options, the application name "ReddiffMoney" is displayed. Underneath, there are two radio button options: "Add as a new application" (selected) and "Map to an existing application". A table shows application details with edit icons (pencil icons) next to the "Source Application", "Assembly", and "EnvironmentType" fields. The "PROCEED" button is highlighted with an orange box.

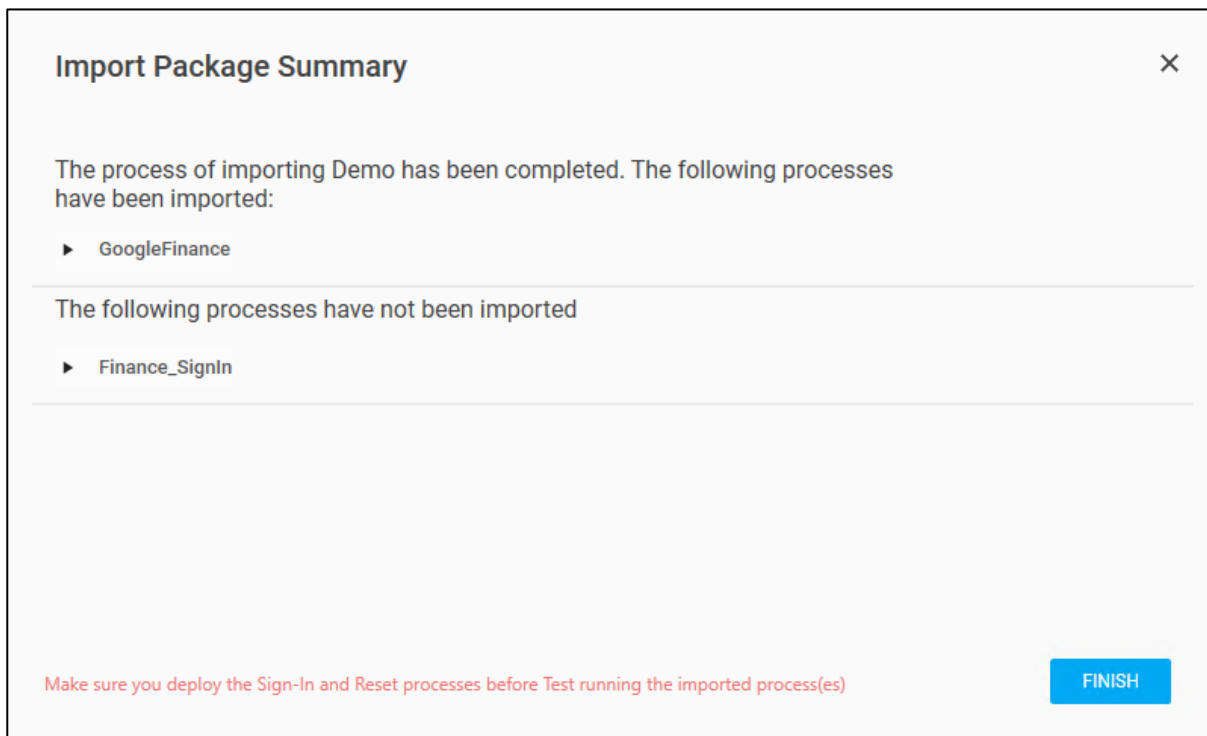
Property	Value	Edit
ApplicationName	ReddiffMoney	✎
ApplicationDescription		
Assembly	PROTON.Demo.Applications.1	✎
Class		✎
EnvironmentType	WINDOW CLASS BASED	✎

9. Click the arrow next to application name to view the details of the process being imported. Available options are **Add a new application** and **Map to an existing application**.
- **Add as a new application** - This option lets you add the application involved in the process to be imported to the current environment as a new application. Additionally, you can edit the properties of the associated application. To edit the properties:
 - Click the  (pencil) icon and edit the properties of the application involved.
 - **Map to an existing application** - This option lets you map the application involved in the process to be imported with the application that already exists in the current environment. To align the existing application with the process being

imported:

- In the **ApplicationName** list, select the existing application.

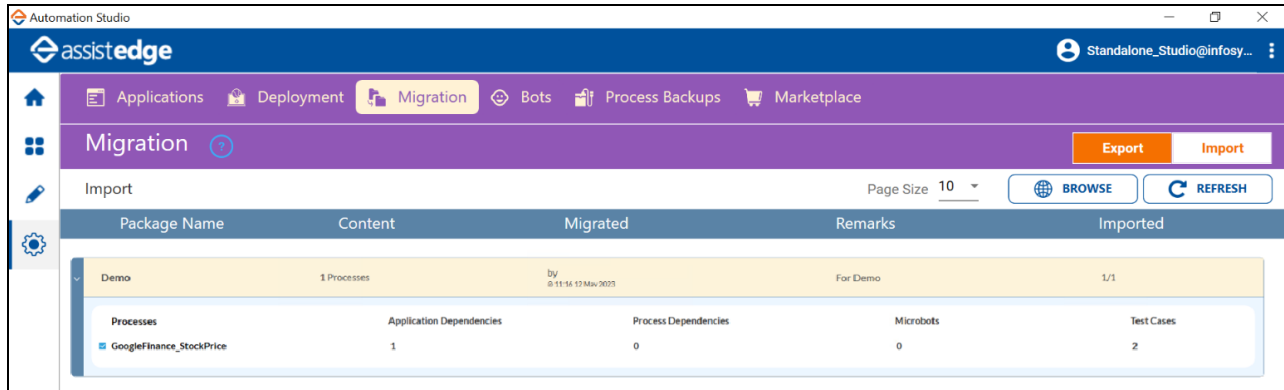
10. You can click **Skip Process** if you do not want to proceed with the specified details. Repeat steps 2 through 7 to initiate a fresh process import or skip this step to proceed with the specified details.
11. Click **PROCEED**. The **Import Package Summary** with details process imported is displayed.



12. Click **FINISH**.

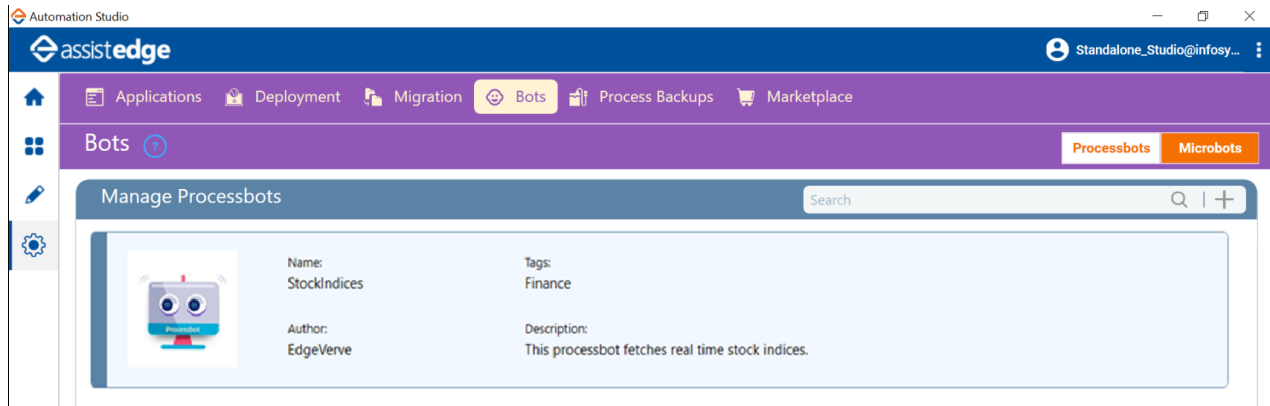
The process package is imported.

Once process is imported, you can expand the process and view the number of processes imported in the package. Additionally, you can view the associated applications, Microbots, Test Case,s and process dependencies of the expanded process.



5.4 Bots

The Bots tab allows you to manage processbots and microbots in Automation Studio. By default, Processbots is selected.



Click any of the following links to know the details:

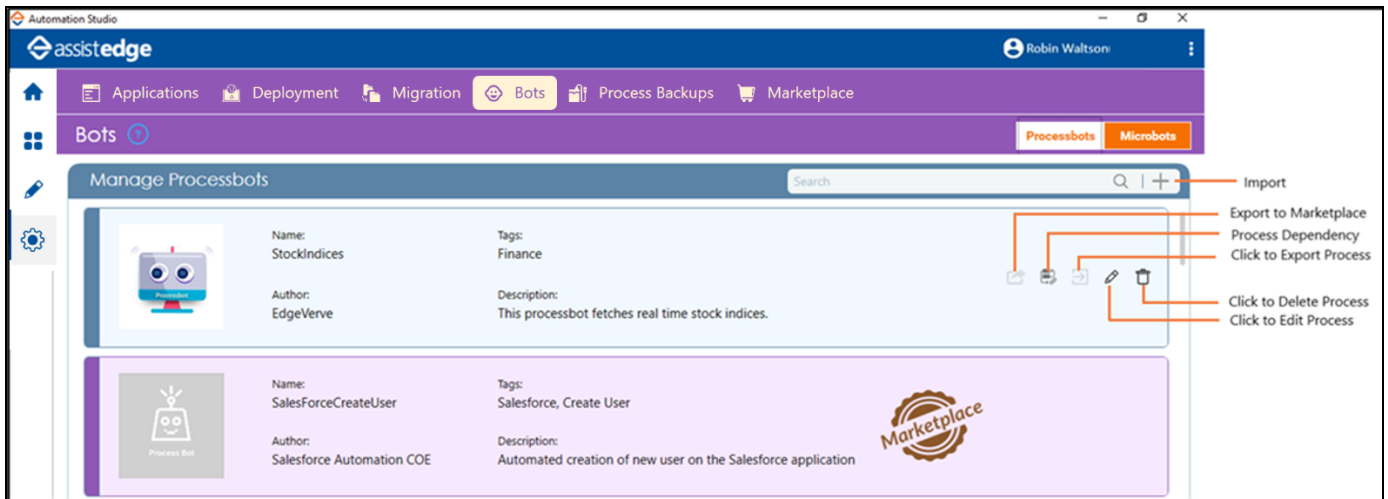
- [Processbots](#)
- [Microbots](#)

5.4.1 Processbots

Processbots are package of a process that contains the metadata related to the process such as its name, author, tags, descriptions and so on. Packaging a process with its metadata helps to know the utility and other usage details of the process.

A processbot is another way of migrating a process from one environment to another environment. All the processbots that are

created or imported appear in the Processbots tab. A processbot imported from AssistEdge Marketplace appears with the Marketplace symbol.




Following are the available functionality:

- **Export to Marketplace**- Creates a processbot package for exporting to AssistEdge Marketplace and saves it in the local machine.
- **Process Dependency**- Provides an insight if the process is re-used in any other process.
- **Click to Export Process**- Exports the process to .zip file and saves in the local machine.
- **Click to Edit Process**- Edits the metadata details entered while creating the processbot.
- **Click to Delete Process**- Deletes the processbot.

Create a Processbot

1. In the **Deployment** tab, select the process you want to add as a processbot, and then click the **(More)** icon.
2. Click **Add as Processbot**. The Add as Processbot dialog box appears.

Add as Processbot ×



Name
StockIndices

Author
EdgeVerve

Description
This processbot fetches real time stock indices.

Readme File Path
D:\Sample Data\Readme.txt

Tags
Finance

SAVE

3. In the **Author** field, enter a name of the creator of the processbot.
4. In the **Description** box, enter information related to the processbot.
5. In the **Readme File Path**, browse and select the file that includes instruction related to usage of the processbot.
6. In the **Tags** field, enter a label of the processbot for identification. You can assign the same tag to multiple processbot to keep them under one group for the ease of identification and access.
7. Click **SAVE**.


The processbot is created. It starts appearing in the Processbots tab.

Import a Processbot

1. In the **Bots** tab, click **Processbots** and then click the  (**IMPORT**) icon.

Import Processbot

Browse Processbot



Name

Description:

Author:

Tags:

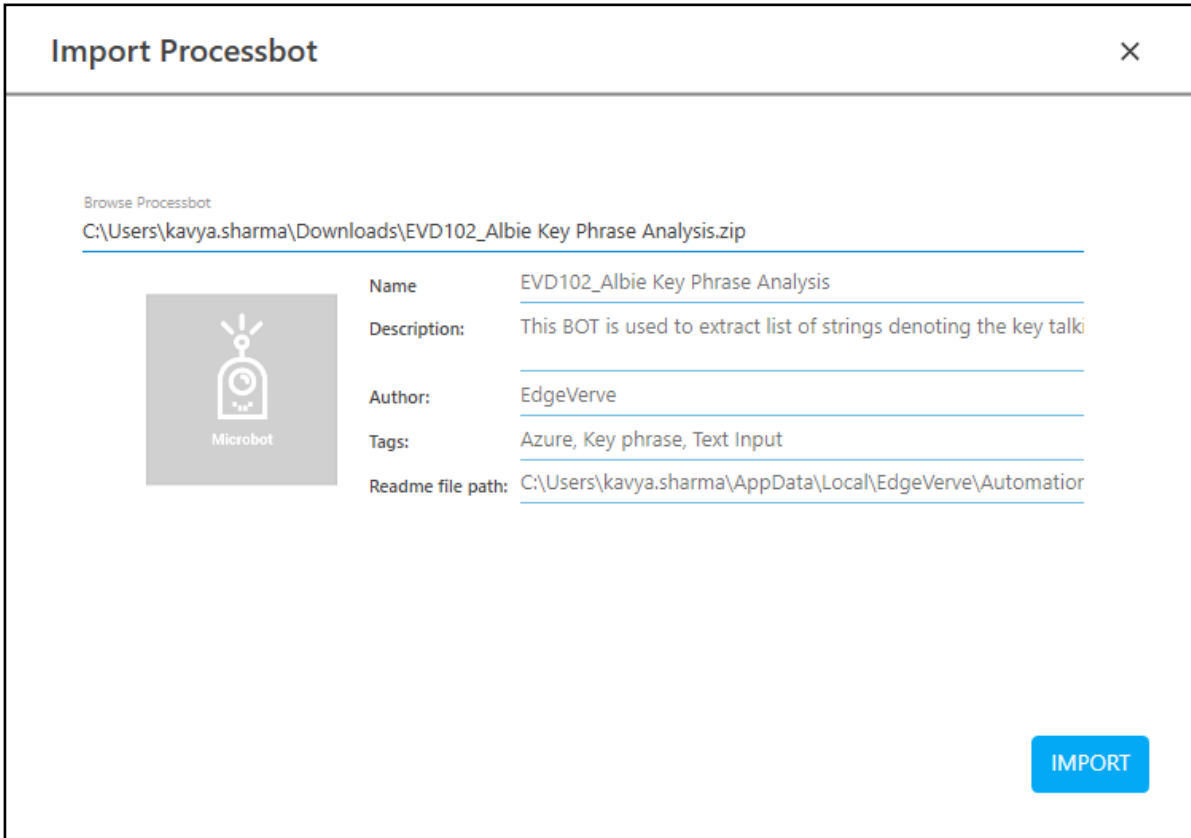
Readme file path:

Name, Description, Author and ReadMe are mandatory

IMPORT

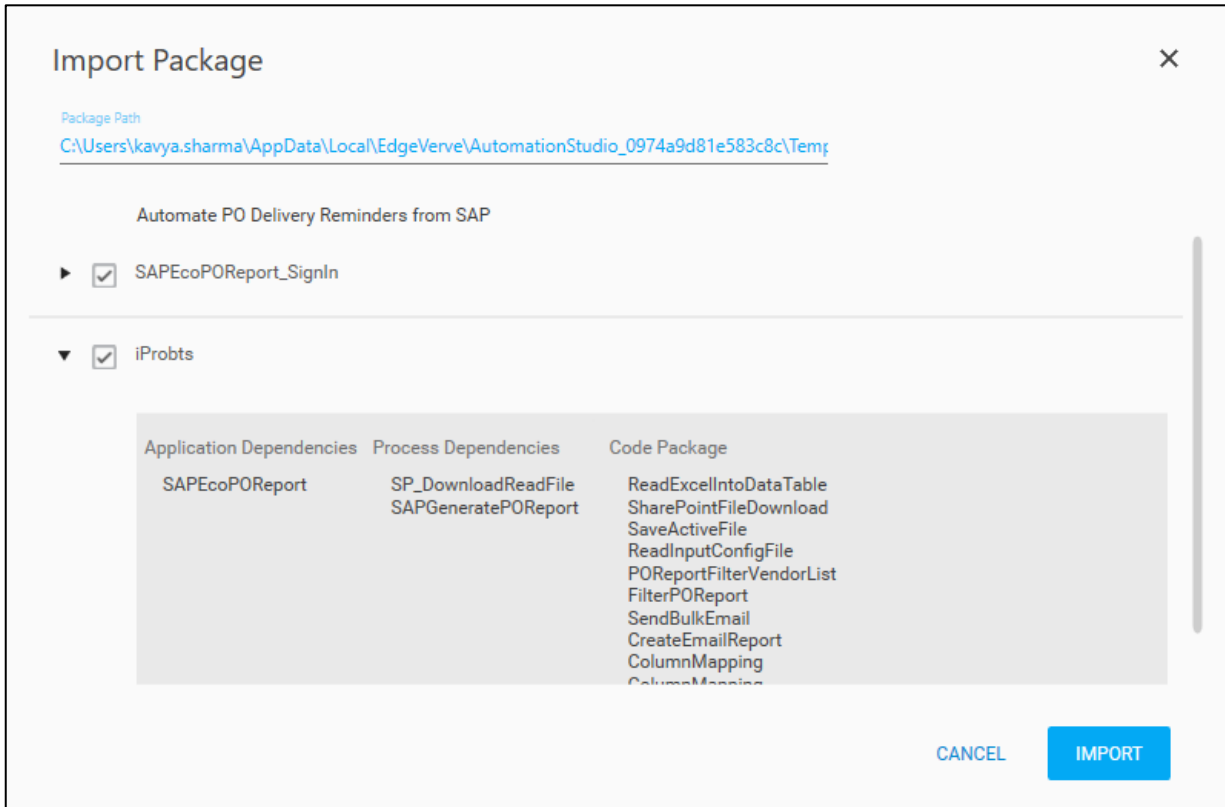
2. Click the **Browse Processbot** field and then browse, select, and open the process package zip folder to import to the current environment.

- 3. Click **Open**. Select the processbot to be imported. The relevant details appear in the respective fields of the processbot.



- 4. Click **IMPORT**. The **Import Process** dialog box appears.

- Select the process to be imported. You can import single or multiple processes contained in the processbot zip folder. You can click the arrow beside the process to view the details related to the process selected. If multiple processes are being imported inside the zip folder, following options and screens appear for each process.



- Click **IMPORT**. The **Import Process** dialog box appears. Following options appear based on the duplicity of the process: Select the **Rename the imported process** check box. The **Verify** field appears.
 - If the process does not exist in the current environment, you can either rename the process (if required) or proceed with the same name. To rename the process:

- a. Select the **Rename the imported process** check box. The **Verify** field appears.

Import Process Process 2 of 2

iProbts does not have any conflicts in the current environment

Rename the imported process

iProbts_SAP_PO Verify

Verified Successfully

▼ SAPEcoPOReport

Add as a new application Map to an existing application

	Source Application	
ApplicationName	SAPEcoPOReport	
ApplicationDescription	PO Report	<input type="text" value=""/>
Assembly	PROTON.Demo.Applications.(<input type="text" value=""/>
Class		<input type="text" value=""/>
ConnectionName	ECO	<input type="text" value=""/>

- b. Enter a name of your choice and click **Verify** to check duplicity of the process name. If no process exists in the environment with name entered, a message for successful verification appears.
- c. Click **PROCEED**.

- If a process already exists in the current environment, you can select any of the available options as per your requirement:
 - **Overwrite in the existing profile**- creates a copy of the existing process.
 - **Rename in the existing profile**- renames the process, and then imports the package.
 - **Re-map process to different profile**- maps the process with another profile.

Import Process

Process 1 of 2
✕

SAPEcoPOReport_SignIn already exists in the current environment.
How would you like to resolve this conflict?

Skip Process

Overwrite in the existing profile

Rename in the existing profile

Re-map process to different profile

SAPEcoPOReport_SignIn

▼ SAPEcoPOReport

Add as a new application Map to an existing application

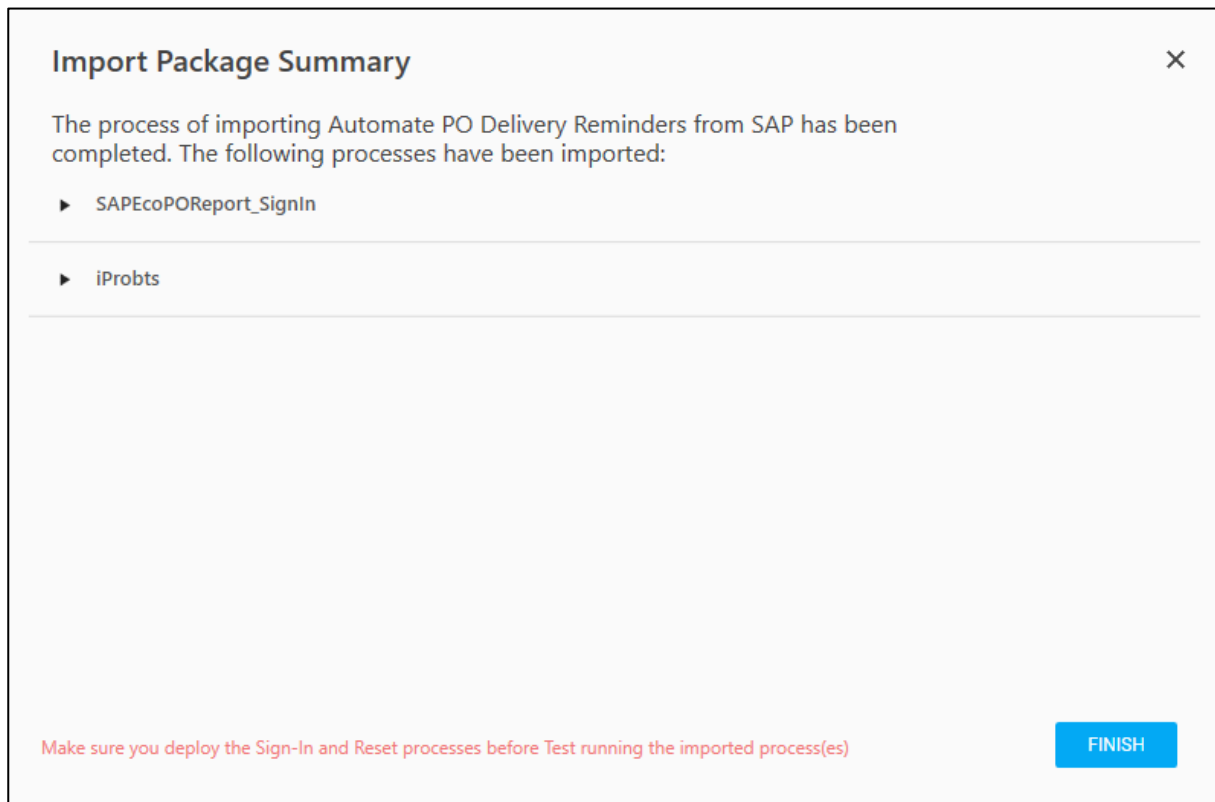
ApplicationName	Source Application	
ApplicationDescription	SAPEcoPOReport	
Assembly	PO Report	
Class	PROTON.Demo.Applications.t	
ConnectionName	ECO	

CANCEL

PROCEED

7. Click the arrow next to application name to view the details of the process being imported. Available options are - **Add a new application** and **Map to an existing application**.
 - **Add as a new application** - This option lets you add the application involved in the process to be imported to the current environment as a new application. Additionally, you can edit the properties of the associated application. To edit the properties:
 - Click the (pencil) icon and edit the properties of the application involved.

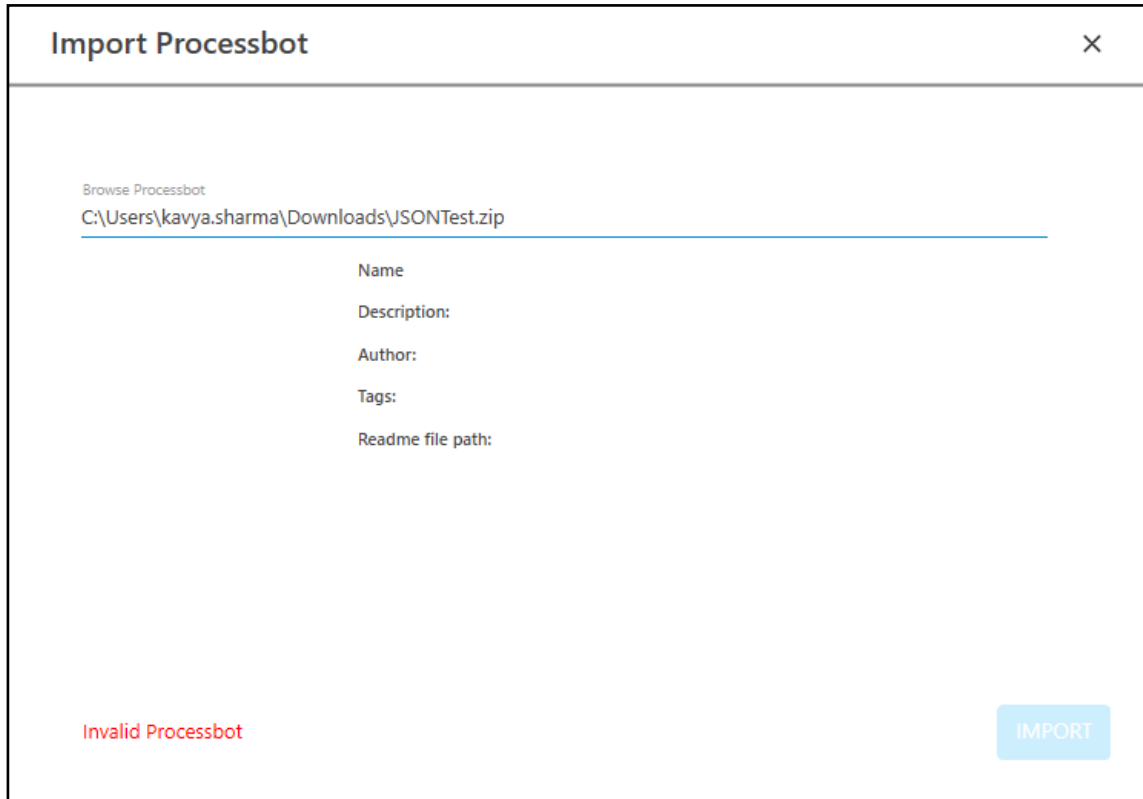
- **Map to an existing application** - This option lets you map the application involved in the process to be imported with the application that already exists in the current environment. To align the existing application with the process being imported:
 - In the **ApplicationName** list, select the existing application.
8. You can click **Skip Process** if you do not want to proceed with the specified details. Repeat steps 2 through 7 to initiate a fresh processbot import or skip this step to proceed with the specified details.
 9. Click **PROCEED**. The **Import Package Summary** with details process imported is displayed.



10. Click **FINISH**.

The imported processbot starts appearing in the Manage Processbots page.

If the processbot zip folder is corrupted or is not a valid processbot, a message stating Invalid Processbot is displayed, as shown in the below screen shot. Repeat steps 1 through 9 to import a valid processbot.



5.4.2 Microbots

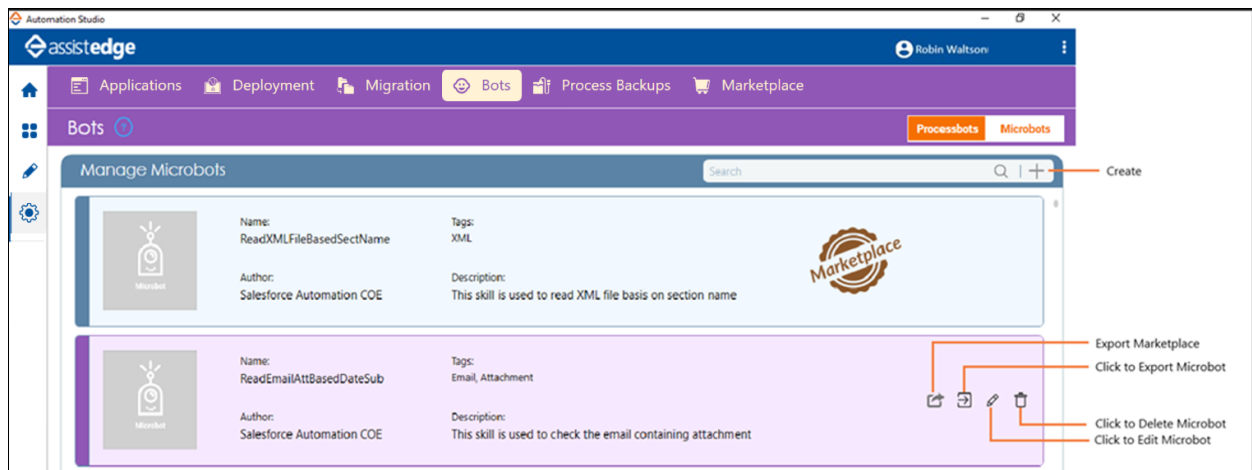
Microbots offer advanced capabilities where you can create small reusable custom code to perform automation. Microbots are reusable and can be used across automation processes. Tasks that cannot be performed using activities available in the Automation Studio, can be achieved with the help of a microbot such as capture screen shots and save in MS Excel, validate user in Active Directory.

Codes for a microbot are written in Microsoft Visual Studio and imported to Automation Studio.

The Microbots tab allows you to either create a new microbot or import an existing microbot available in your system. A microbot imported from AssistEdge Marketplace appears with the Marketplace symbol.

Note:

- The default size of the Microbot is 4 MB and if the Microbot contained in a process exceeds this limit the process does not get saved.
- To work with larger size of Microbots you must change the values of **maxAllowedContentLength** and **maxRequestLength** fields available in the **web.config** file. To update the **web.config** file, navigate to **Product folder > app > Modules > Workflow** folder, and make the changes as per your business need.



Following are the available functionality:

- **Export to Marketplace**- Creates a microbot package for exporting to AssistEdge Marketplace and saves it in the local machine.
- **Click to Export Microbot**- Exports the microbot to .zip file and saves in the local machine.
- **Click to Edit Microbot**- Edits the metadata details entered while creating the microbot.
- **Click to Delete Microbot**- Deletes the microbot.

Prerequisite

Create the custom code in Visual Studio and add the references of Automation Studio to your project. The DLL file created is imported in the Automation Studio that forms the intended microbot.

Create a Microbot

1. In the **Bots** tab, click **Microbots** and then click the **+** (**CREATE**) icon.
2. In the **Add New Microbot** dialog box, select **Create microbot**. The **Add New Microbot** dialog box appears.

Add New Microbot ×

Create microbot Import microbot

Name

Author

Description

Microbot Path

Readme File Path

Tags

Name, Description, Author, Microbot path and ReadMe are mandatory

SAVE

3. In the **Name** field, enter a name of the microbot.
4. In the **Author** field, enter the name of the creator of the microbot.
5. In the **Description** field, enter a description of the microbot stating its functionality.
6. In the **Microbot Path** field, browse and select the location of the saved DLL file.
7. In the **Readme File Path**, browse and select the file that includes instruction related to usage of the microbot such as the type of input to be used, the kind of output expected and so on.
8. In the **Tags** field, enter a label for the microbot identification. You can assign the same tag to multiple microbots to keep them under one group for the ease of identification and access.
9. Click **SAVE**.

The microbot is created.

Importing an Existing Microbot:


1. In the **Microbots** tab click **CREATE**.
2. Select **Import microbot**. The **Add New Microbot** dialog box appears.

Add New Microbot [X]

Create microbot Import microbot

Microbot Name
 D:\Build\EV_279_278 distributed\client-tools\AutomationStudio\Exports\Microbots\ReadInputCustomConfigX

ReadInputCustomConfigXMLFile

 Microbot

Description: Read Input Custom Configuration XML File

Author: Edge Verve

Tags: XML file

Microbot path: C:\Users\kavya.sharma\AppData\Local\EdgeVerve\Automation

Readme file path: C:\Users\kavya.sharma\AppData\Local\EdgeVerve\Automation

[SAVE]

3. In the **Microbot Name** field, browse and locate the existing microbot zip file.

The details of the selected microbot are automatically filled in the Description, Author, Tags, Microbot path and Readme file path. You can change these details as per your requirement.

4. Click **SAVE**. The imported microbot starts appearing in the **Manage Microbots** page.

The microbot is imported.


5.5 Process Backups

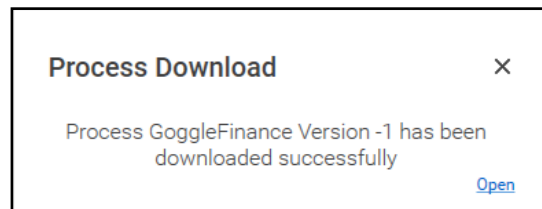
The Process Backups tab allows you to view the list of process backups. When a process is published, back up gets automatically created and saved in the database.

The backup is created only for published and deployed processes and is available for download. The downloaded backup is stored as .zip file in the local machine.

Process Name	Process Type	Profile	Status	Version	Comments
processwrite	Process	StandaloneProfile	Published	Version 1 by Standalone_Studio @ 15:55:20 Oct 2023	Imported
chrome	Process	StandaloneProfile	Published	Version 1 by Standalone_Studio @ 15:55:20 Oct 2023	Imported

To download a process backup:

1. In the **Process Backups** tab, select the backup process you want to download, and then click the  (**Download**) icon. A confirmation message to download the process backup appears.
2. Click **OK**.



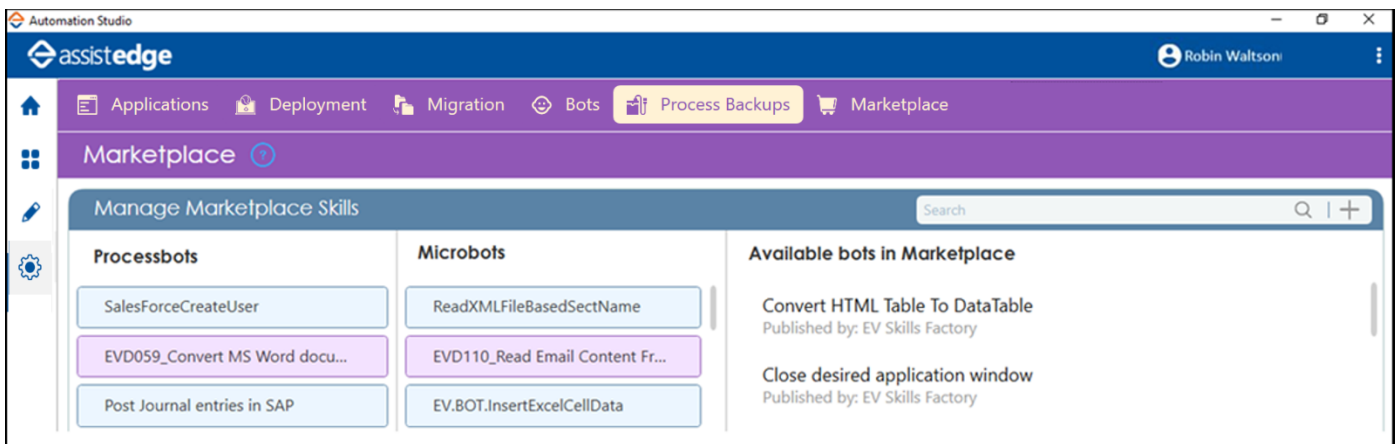
A message with successful download appears, including the version downloaded and link to the process backup zip file appears.

5.6 Marketplace

The Marketplace tab displays a list of comprehensive featured skills from AssistEdge Marketplace in the Available bots in Marketplace pane. Search for the required bot, download and import in automation Studio to start using it.

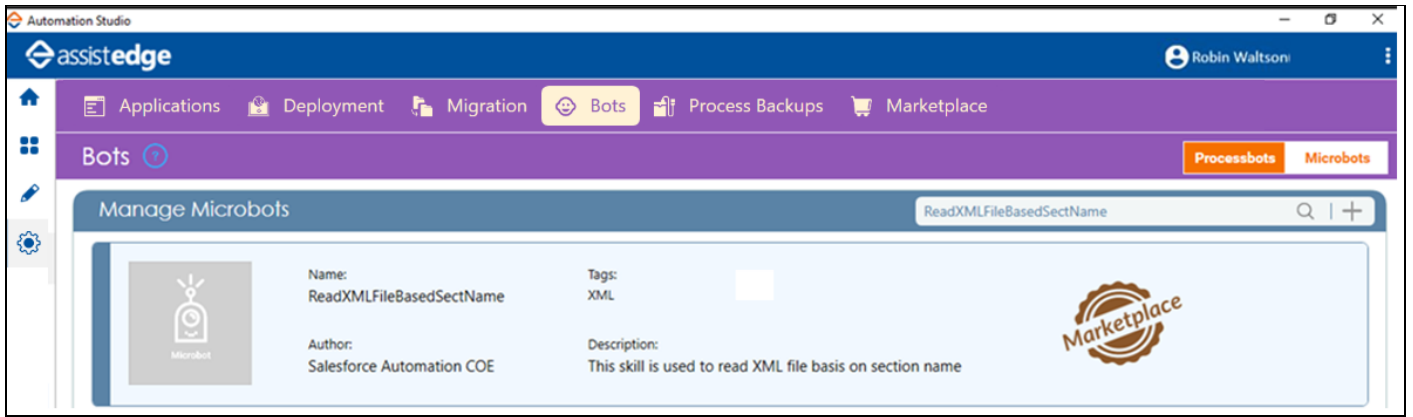
You can view the list of already imported processbots and microbots downloaded from AssistEdge Marketplace in the Processbots and Microbots pane respectively.

Search for the required skill by entering a string in the Search field. The string entered is searched in AssistEdge Marketplace and returns the search results in the Available bots in Marketplace pane. Additionally, the list in the Processbots and Microbots pane gets filtered based on the search string. A message is displayed in case no matching skill is found.



5.6.1 Downloading Skill form AssistEdge Marketplace


1. In the **Marketplace** tab, search and click the required bot from the list of available bots. The AssistEdge Marketplace link opens.
2. Login with appropriate credentials and download the processbot or microbot you want to import to Automation Studio. The bot gets downloaded as a zip file.
3. Import the downloaded zip file in Automation Studio. Refer [Importing a Processbot](#) and [Importing a Microbot](#) sections in the **Bots** tab to know how to import the downloaded bot.

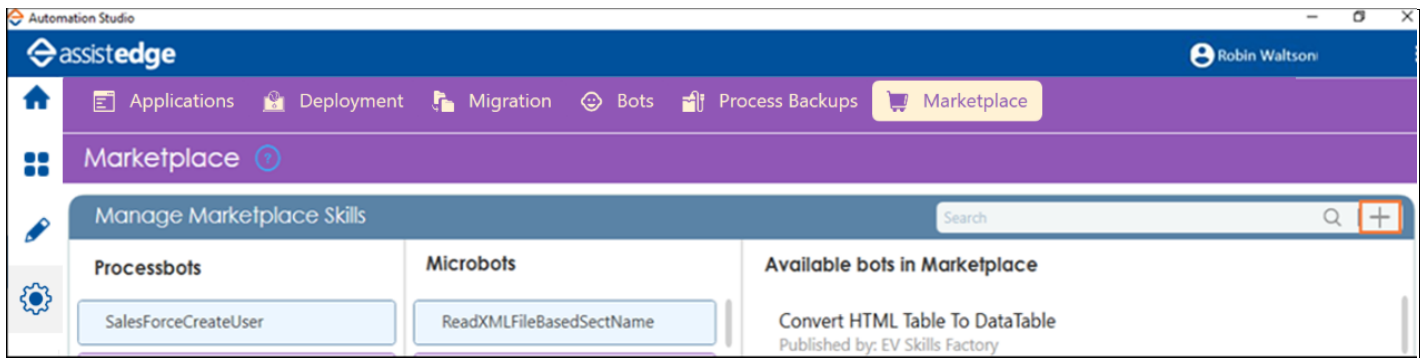


The imported bot starts appearing in the Marketplace tab. The bot is also available in the Bots tab.

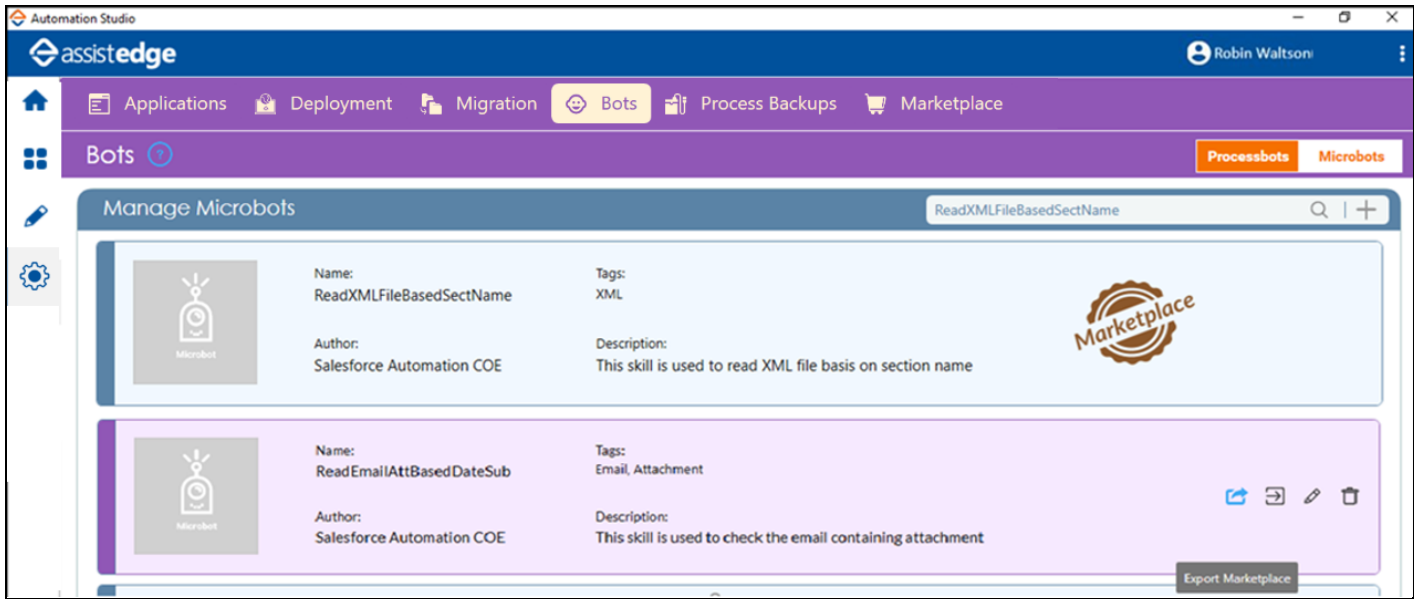
Bots downloaded from AssistEdge Marketplace are identified with the Marketplace tag against each bot.

5.6.2 Publishing Skills in AssistEdge Marketplace

Use the  (Upload to Marketplace) icon to upload your own bot to the AssistEdge Marketplace by exporting the reusable component. Once the Marketplace URL opens, you can publish the bot by becoming a publisher in AssistEdge Marketplace.



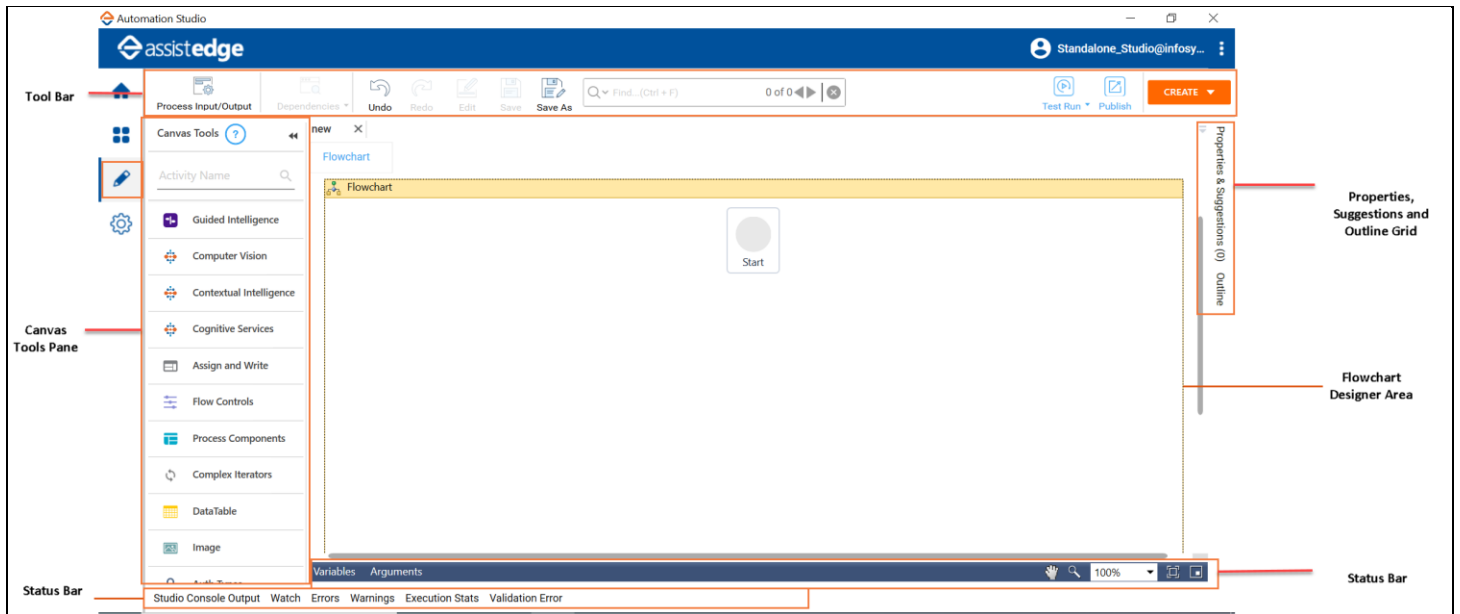
You need to export the zipped package to Automation Studio using Export to Marketplace option available in the Bots tab and then upload to AssistEdge Marketplace.



Note: A process that is exported to a zipped package using Export to Marketplace option can only be uploaded as a bot in the AssistEdge Marketplace. Any other process zip folder cannot be uploaded to the Marketplace.

6 Manage Studio

You can create, test, and publish process workflow in the Studio menu. It offers a rich interface that varies from automating a simple step of visiting a website to automating a complex process involving complex enterprise applications.



6.1 The Tool Bar

The Tool bar consists of functionality such as creating or editing a process, providing inputs to a process and so on. You can perform the following actions:

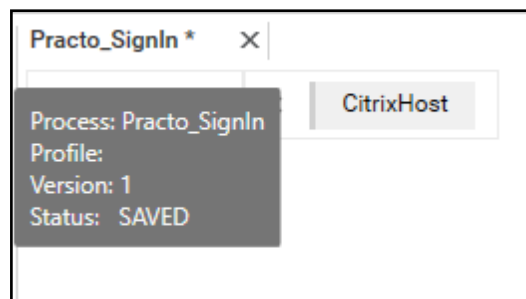


- **Process Input/Output-** Allows to select or add an argument as an input into the process. Additionally, you can mark the parameters that contain the outputs of the process workflow to view in the Control Tower. For more information, see Process Input/Output section.
- **Dependencies** - Allows to view details of the application/s used in the process. additionally, you can view if the selected process is re-used in any other process or not. This option is enabled only after a process is saved.
- **Undo** - Allows to reverse the last action performed. This option is enabled only after the process is edited.
- **Redo** - Allows reverse the action that was previously undone. This option is enabled only after an edit in the process is reversed.

- **Edit** - Allows to make changes in an existing process workflow. Only a published or a deployed process can be edited.
- **Save** - Saves the latest content in the process workflow.
- **Save As** - Saves a new process workflow by providing a name.
- **Find Options** - Allows to search a word in the process in the **Flowchart** designer area. The search is similar to MS Word and allows to search any text, an argument or property within the process. You can select the option to match case or match whole word while finding a word. The interface shows the number of occurrences for that search, along with its location at activity level within the process. Use the arrows to move to the previous and the next word found.
- **Test Run** - Prepares the environment and applications for testing. For more information, see [Test Run Process](#).
- **Publish** - Creates a backup of the process and prepares it for deployment. This option is enabled only after the process is saved. For more information, see [Publish Process](#).
- **Create** - Creates different types of automation process workflow. For more information, see [Process Management](#).

6.2 The Canvas

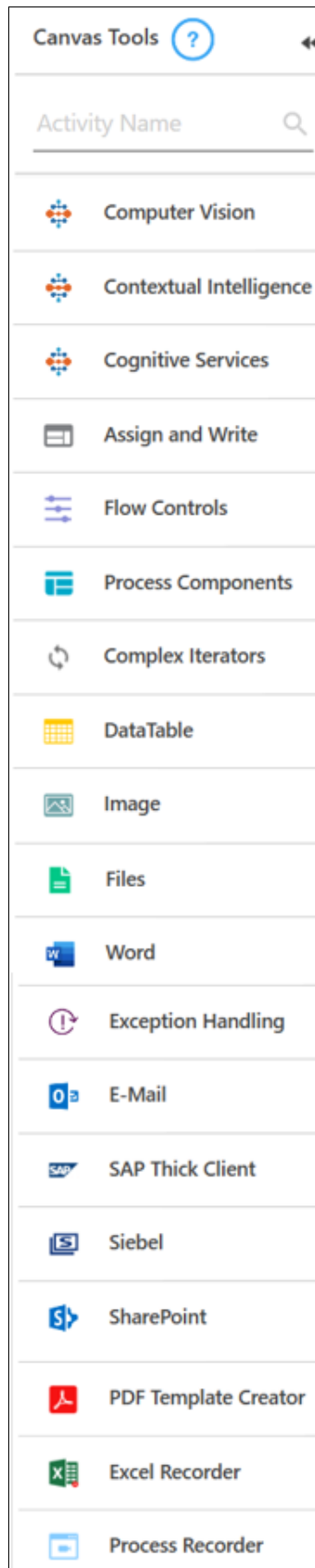
Canvas acts as the custom workflow designer area where RPA sequences are created. The center of the screen is the canvas where each tab displays name of the process that you open. You can hover over the process tab to view the details like name of the process, its status and current version. Additionally, the navigation trail of the selected process is displayed.



6.3 The Canvas Tools Panel

The Canvas Tools panel is the interface where you can find pre-defined automation tools to suit your business needs.

Use the ◀◀ to collapse the Canvas Tools panel and ▶▶ to expand it.



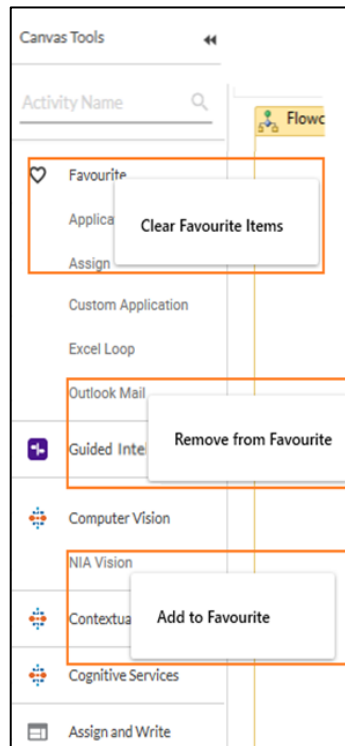
6.3.1 The Activity

Activities are pre-defined, out-of-box operators in Automation Studio. They are the building blocks that enable you to create the automation process workflow. Similar activities are clubbed under one tool category for an easy navigation.

Mark an Activity as Favourite

You can mark the frequently used activities in Canvas Tools as favorite. This enhances the ease of access to the activities marked.

Right click the selected activity and then click Add to Favourite. The activities marked as favorite are listed under Favourite Activity.



Right click the activity added to Favourite to remove from the list. Additionally, you can clear the entire list of favorites in a go by using Clear Favourite Items.






Using an Activity

You can drag an activity from the Canvas Tool pane and drop on to the Flowchart designer to add to a process workflow.

When you add an activity, details like the name of the activity, its properties, and the outline view of the selected activity appears.

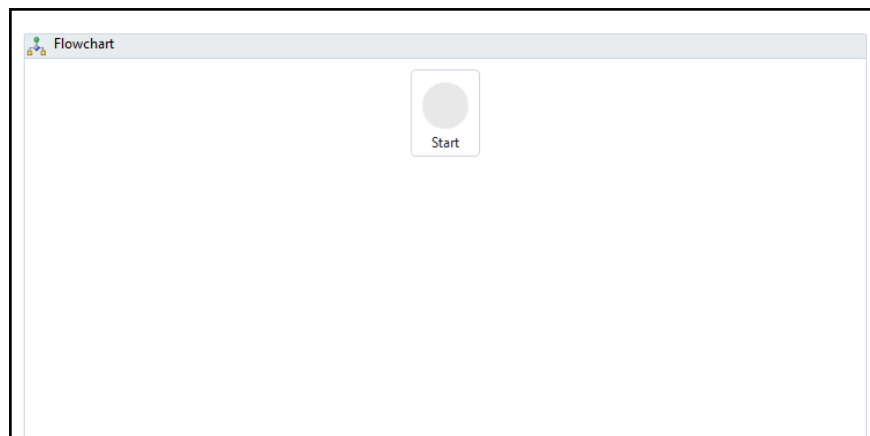
Activity Symbols

Below are the functionalities of an activity:

Symbol	Symbol Name	Description
	Validation Error or Warnings	Signifies either there is a validation error or the required inputs for the activity to proceed is currently missing. It disappears when required action is performed.
	Help	Help that is available within Automation Studio itself regarding the particular activity such as its definition or usage.
	Annotations	Add a note or a comment for additional information regarding the activity.
	Settings	Settings that you can use to define features of an activity such as capturing data from a web application or to provide input/output parameters.
	Compiler Error	Signifies the error that occurs when the compiler fails to compile the source code.

6.4 The Flowchart Designer

The Flowchart designer is the area on the Canvas where you can add activities to build workflow of the automation process. It appears when you start creating a process.



6.4.1 Additional Features

To enhance usability of Automation Studio, additional features are available on the Flowchart designer are:

Auto Parameter Rename

If the parameter name is changed, it gets auto reflected at activity level at all the instances where it is used. The parameter name, used inside scripts and the code, must be changed manually.

6.5 The Properties & Suggestions Grid

Click Properties & Suggestions to open and use **X** (close) to close the grid. With the help of **📌** (pin), you can auto hide the grid.

The fields specific to the Canvas, Flowchart designer and the selected activity are displayed in the Properties & Suggestions grid. It helps you to view and edit properties that describes how they are use in the flowchart.

In the Suggestions pane, the list of prerequisites with the selected activity is available. The Suggestions pane is divided in 2 sections:

- **Mandatory Pre-Requisites-** In this section, you can view the list of required prerequisites associated with the selected activity.
- **Recommended Pre-Requisites-** In this section, you can view the list of recommended prerequisites associated with the selected activity. These prerequisites are optional, you can use as per your preference.



Additionally, you can perform following actions:

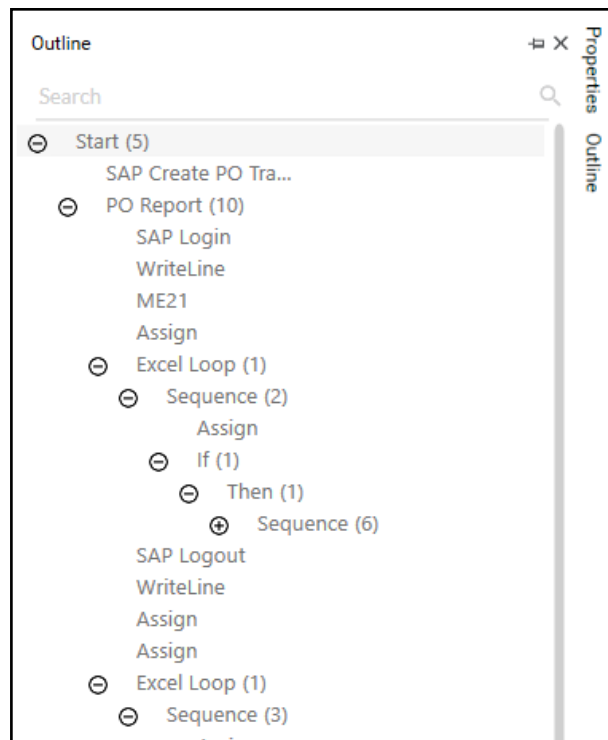
- **Categorized** - Views properties of the selected activity based on the category of the properties.
- **Alphabetical** - Views properties of the selected activity based on the alphabetical order of the properties.
- **Search** - Allows to search for a specific property of the selected activity.
- **Clear** - Clears the search box.

6.6 The Outline Grid

Click Outline to open the and use ✕ (close) to close the grid. With the help of 📌 (pin), you can auto hide the grid.



The Outline grid provides a view to the flow of the entire process workflow. You can view all the nodes of a process including the tools and activities used.



Use this grid to navigate to the desired section of the selected automation process.



6.7 The Parameter Bar

Variables and arguments are defined in the Parameter bar of the Studio menu. You can even re-size the Flowchart designer using this bar.

Click the  icon to activate the Toggle Pan Mode. Hold the space key or press mouse middle button to pan on demand. There are instances when you want to zoom in or zoom out a section of your process. Use the zoom drop-down to set the percentage of the layout and the  to reset it to 100%.

You can use the  icon to fit the entire content of the Flowchart designer in a single view-able screen. This helps you to have a glance at the entire process as a whole. The  icon helps you to get an overview of the entire process.

6.7.1 Variables in Automation Studio

A variable in Automation Studio is used to store data. It can be a string, integer, Boolean, null or other data type depending on the requirement. You can view the list of available variable type in the Variables pane. They have limited scope and can be used within an activity.

Variable Field Descriptions

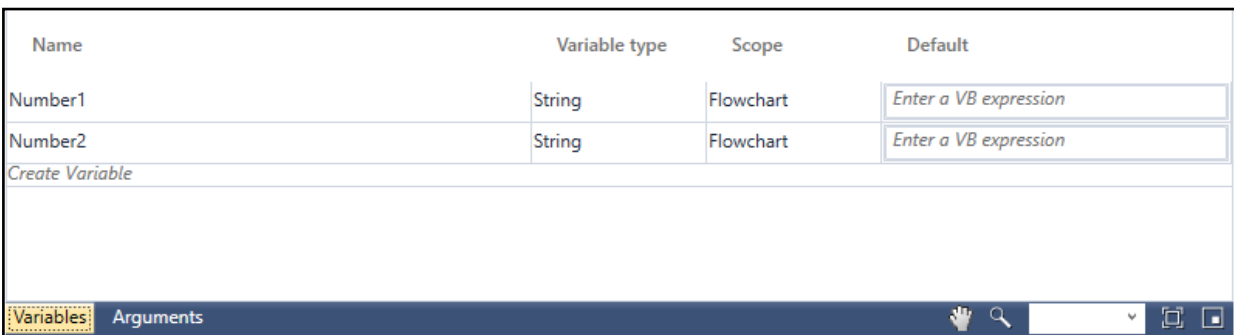
The fields of a variable in Studio are listed in the following table:

Field Name	Description
Name	The name of the variable created. If you do not assign a name, variable with a default name is created.
Variable Type	<p>Defines the type of variable created. Available options are:</p> <ul style="list-style-type: none"> ▪ String- a sequence of character, either as a constant or a variable. ▪ Boolean-a datatype used for making decision. Can have only two values- true or false. ▪ Int32-an integer type that range from negative through positive. ▪ Object- an abstract datatype that holds data of any kind. ▪ Array of [T]- a group or collection of same datatypes. ▪ Browse for Types...- other types of data. Automation studio supports, .Net

	variable datatype. Default value is set as String .
Scope	Determines the range of visibility and validity of the variable inside Automation Studio. Variables can be used only within an activity or a workflow. Variables created within an activity or a process workflow is available for selection only within that workflow by other activities. It would not be available to use within another workflow.
Default	The value that you want to assign to the variable. If you do not provide a value then the predefined value of the selected variable type is set, for example, null for Boolean, 0 for integer and so on.

To create a variable:

1. In the **Parameter** bar, click **Variables** to open the **Variables** pane.



2. Click the **Create Variable** field and enter a name of the variable.
3. From the **Variable Type** drop-down list, select the type of variable you want to create.
4. In the **Default** field, enter a value that you want to assign to the variable.

The variable is created.

To delete a variable:

1. In the **Parameter** bar, click **Variables** to open the **Variables** pane.
2. Right click the variable and then click **Delete**.

The variable is deleted.

6.7.2 Arguments in Automation Studio

An argument in automation Studio is also used to stores data but the scope of an argument is not limited. It can be used to pass data across different automation process workflow. This implies that an argument created within one workflow is available for selection within another workflow. It can be a string, integer, Boolean, null or other data type depending on the requirement. You can view the list of available variable type in the Arguments pane.

The Argument Fields

The fields of an argument in Studio are listed in the following table:

Field Name	Description
Name	The name of the argument created. If you do not assign a name, argument with a default name is created.
Direction	Defines the direction of the argument. Available options are: <ul style="list-style-type: none"> ▪ In- stored input data ▪ OutArgument - stores output data. ▪ InOutArgument - can be used to store input data as well as output data. ▪ Property – this option is not used in Automation Studio currently. It displays an error if used in the process workflow.
Argument Type	Defines the type of argument created. Available options are: <ul style="list-style-type: none"> ▪ String - a sequence of character, either as a constant or a variable. ▪ Boolean -a datatype used for making decision. Can have only two values- true or false. ▪ Int32 -an integer type that range from negative through positive. ▪ Object - an abstract datatype that holds data of any kind. ▪ Array of [T] - a group or collection of same datatypes. ▪ Browse for Types... - other types of data. Automation studio supports, .Net variable datatype. <p>Default value is set as String.</p>
Default Value	The value that you want to assign to the argument.

To create an argument:

1. In the **Parameter** bar, click **Arguments** to open the **Arguments** pane.

Name	Direction	Argument type	Default value
FirstName	In	String	<i>Enter a VB expression</i>
LastName	In	String	<i>Enter a VB expression</i>
Create Argument			

Variables Arguments

2. Click the **Create Argument** and enter a name of the argument.
3. From the **Direction** drop-down list, define the direction of the argument.
4. From the **Argument Type** drop-down list, select the type of argument you want to create.
5. In the **Default value** field, enter a value of the argument created.

The argument is created.

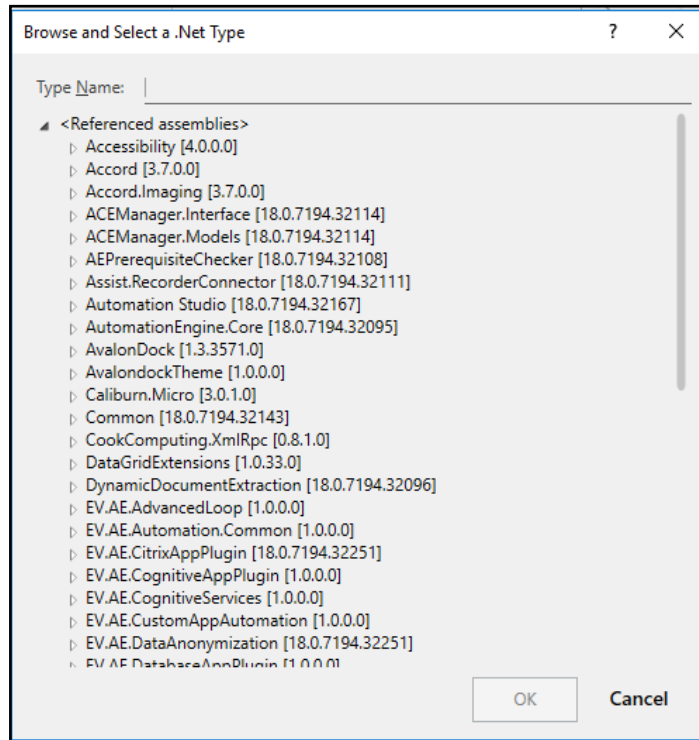
The process to delete an argument is same as deleting a variable.

[.Net Variable/Argument Types](#)

Automation Studio supports .Net variable type apart from default variable or argument types such as System.Data.DataRowView. You can browse for the .Net variable type using Browse for Types... option available in the Variable panel or the Argument panel.

To browse for the .Net Variable Type:

1. In the Variable/Argument panel, from the Variable type/Argument type drop-down, select **Browse for Types**.



The Browse and Select a .Net Type dialog box appears.

2. In the **Type Name** field, enter the keyword of the variable type you want to use.
3. Select the desired variable type and then click **OK**.

A variable with selected variable type is created.

6.8 The Status Bar

In the Status bar, you can view status and other stats of the process you executed along with the errors, if any.

The stats available can help the configurator to understand the CPU and memory utilization during the execution of the process. The configurator can also track the time taken by the steps involved in the process without actually deploying the process to a bot.

Studio Console Output Watch Errors Warnings Execution Stats Validation Error

6.8.1 The Watch Console

It allows you to add argument or variable to track the changes in the value assigned while debugging the process.



6.8.2 The Studio Console Output

The console of Automaton Studio where you can view the output.



6.8.3 The Warnings Console

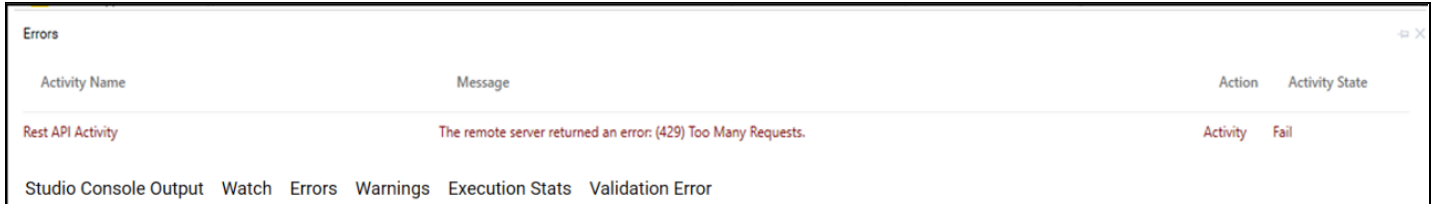
The console where you can view details about unexpected behavior of the process, if any. Warning provides a message related to possible future issue. These warning messages are shown for the applications which use any of the components covered by prerequisite checker even if the application has been imported or recorded by the PbE.

The console shows up when the environment is set or the application is run.



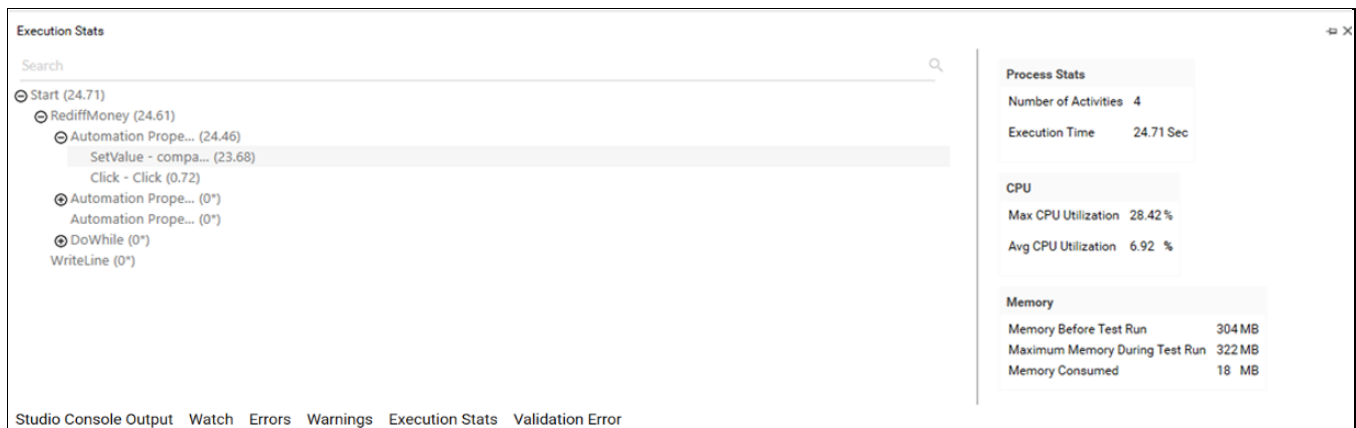
6.8.4 The Error Console

The console where you can view details of the error that might occur during testing. It provides quick issue identification during process configuration and execution.



6.8.5 The Execution Stats Console

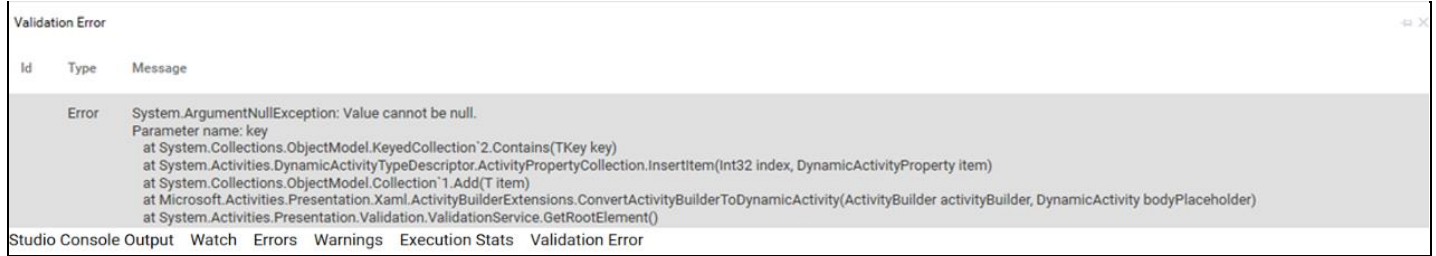
It provides an overview process execution. After a process is executed in Automation Studio, details of the workflow along with all the nodes involved can be viewed in this pane. Additionally, you can view:



- Process Stats** - Displays the total number of activities involved in the process and the total time taken to complete the last execution cycle. In the process outline, available on the left side of the pane provides the break-up of time taken by each activity in the process.
- CPU Utilization** - Displays the maximum and average CPU utilization in the last test run.
- Memory** - Displays details related to the memory available before and after the test run. It also displays the total memory consumed during execution. The memory consumed is the difference of the before and after stats of the memory occupied.

6.8.6 The Validation Error Console

It provides a window to the error that occurs due to validation.



7 Prerequisite Checker

The prerequisite checker works in accordance with the accuracy asserted by the ApplicationAutomationMetadata.json file which maintains the details required for checking the compatibility of the component.

This JSON file is available in: - ../AutomationStudio/ApplicationAutomationMetadata.json

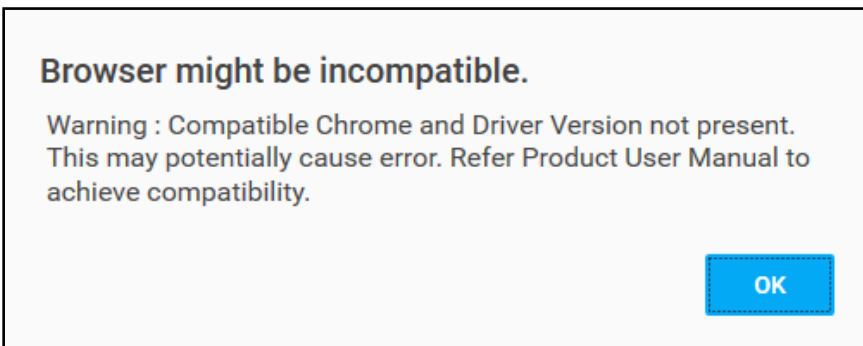
When studio is started, this file is copied to the "ProtonFiles" folder in local app data location which can be identified from the key "RootDirectory" present in the config file of the studio. This location will have user specific json file.

7.1 Chrome Browser Compatibility Check

Chrome automation in AE works on the basis of the chrome driver. Since chrome browser version is subject to change, its driver must be compatible with the new or changed version of the browser.

Automation Studio provides an OOB (Out of box) compatibility alert for Browser and Driver mismatch.

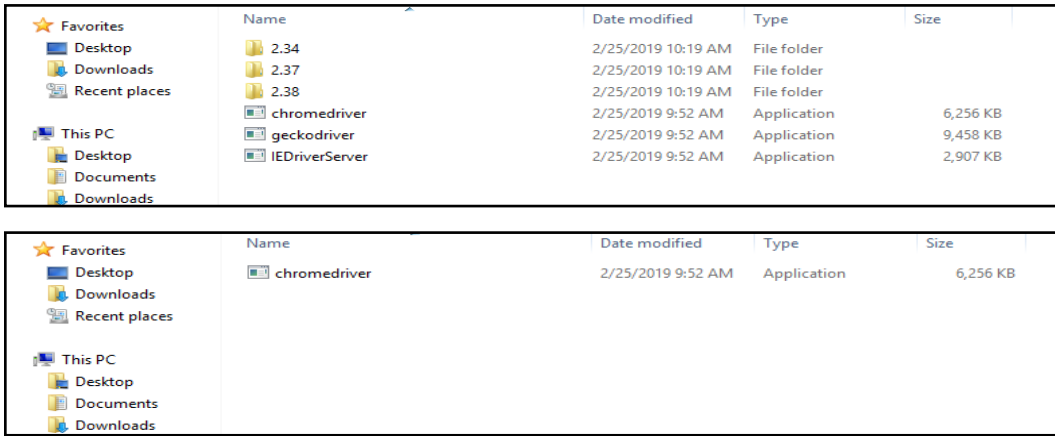
Following alerts are displayed notifying the compatibility of chrome driver and the browser:



Automation Studio displays an alert if the compatible browser and driver version are not present.

In case of such alerts, follow the below steps.

1. Verify the browser version of Chrome.
2. Verify the compatible web driver on the web.
3. To achieve compatibility for Studio, navigate to \AutomationStudio\SeleniumDrivers path.
4. Create a folder with driver version under '**SeleniumDrivers**' folder and place the driver in this new folder.



2.38 is the version of the chrome driver. Place the driver inside this folder.

5. Update the configuration file with the browser and driver entries. The file path is given below.

Automation Studio path: \AutomationStudio\ApplicationAutomationMetadata.json

For example: If we are adding an entry for driver 2.41 then we need to make a ComponentSpecifics json entry under the ComponentName "chrome". We need to specify following details:

- a. "ThirdPartyVersion": "2.41"
- b. "CompatibleComponentWithThirdParty": {"MinVersion": "67","MaxVersion": "69"}
- c. We also need to add the chrome version supported by added driver in the section SupportedVersions
- "SupportedVersions": ["61","62","63","64","65","66","67","68","69"]

```

{
  "ThirdPartyVersion": "2.38",
  "CompatibleComponentWithThirdParty": {
    "MinVersion": "65",
    "MaxVersion": "68"
  }
},
{
  "ThirdPartyVersion": "2.41",
  "CompatibleComponentWithThirdParty": {
    "MinVersion": "67",
    "MaxVersion": "69"
  }
}
],
"SupportedVersions": [
  "61",
  "62",
  "63",
  "64",
  "65",
  "66",
  "67",
  "68",
  "69"
]

```

6. Find the user specific RootDirectory folder which can be located from the key "RootDirectory" present inside PrimarySettings-->IODiskPaths.

7. Inside this folder, delete the SeleniumDrivers folder and ApplicationAutomationMetaData.json file and restart the studio.

Note:

- The browser and driver versions mentioned in the JSON file are not sacrosanct. The combination of application, browser version, and the driver version must be taken into consideration.
- This feature gives a warning to the end user and does not affect the rest of the execution.
- This feature does not replace the driver provided with the product unless the version is mentioned in the JSON file.

If the logs display that chromedriver is in use or access denied, kill the chromedriver.exe from the task manager.

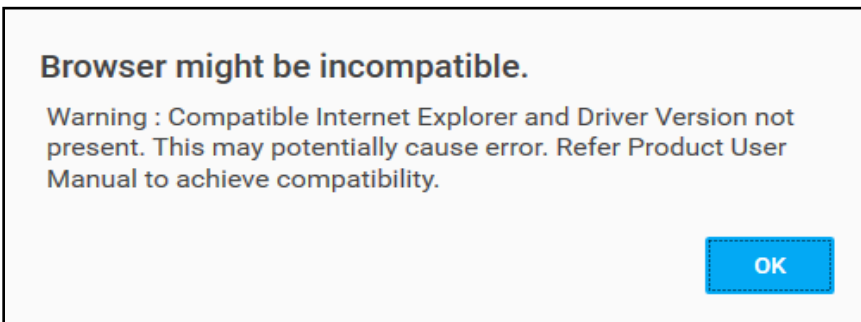
7.2 Internet Explorer Browser Compatibility Check

Internet Explorer automation in AE works on the basis of the IEDriverServer.exe driver. Since IE browser version is subject to change, its driver must be compatible with the new or changed version of the browser.

Automation Studio provides an OOB (Out of box) compatibility alert for Browser and Driver mismatch.

Following alerts are displayed notifying the compatibility of IEDriverServer and the browser:

- Automation Studio displays an alert if the compatible browser and driver version are not present.



In case of such alerts, follow the below steps:

1. Verify the browser version of Internet Explorer.
2. Verify the compatible web driver named **IEDriverServer** on the web and download it.
3. To achieve compatibility for Studio, navigate to \AutomationStudio\SeleniumDrivers automation studio path.
4. Replace the downloaded driver in this folder.

	Name	Date modified	Type	Size
★ Favorites				
Desktop	2.34	2/25/2019 10:19 AM	File folder	
Downloads	2.37	2/25/2019 10:19 AM	File folder	
Recent places	2.38	2/25/2019 10:19 AM	File folder	
This PC				
Desktop	chromedriver	2/25/2019 9:52 AM	Application	6,256 KB
Documents	geckodriver	2/25/2019 9:52 AM	Application	9,458 KB
Downloads	IEDriverServer	2/25/2019 9:52 AM	Application	2,907 KB

- Update the configuration file with the browser and driver entries. The file path is given below.

Automation Studio path:

\\AutomationStudio\ApplicationAutomationMetadata.json

```
{
  "ComponentName": "InternetExplorer",
  "ComponentSpecifics": [
    {
      "ThirdPartyVersion": "3.12.0",
      "CompatibleComponentWithThirdParty": {
        "MinVersion": "10",
        "MaxVersion": "11"
      }
    }
  ],
  "SupportedVersions": [
    "9", "10", "11"
  ]
}
```

- Find the user specific RootDirectory folder which can be located from the key RootDirectory present inside PrimarySettings-->IODiskPaths.
- Inside this folder, delete the SeleniumDrivers folder and ApplicationAutomationMetaData.json file and restart the studio.
- Check the browser compatibility with the new driver.

Note:

- The browser and driver versions mentioned in the JSON file are not sacrosanct. The combination of application, browser version, and the driver version must be taken into consideration.
- This feature gives a warning to the end user and does not affect the rest of the execution.

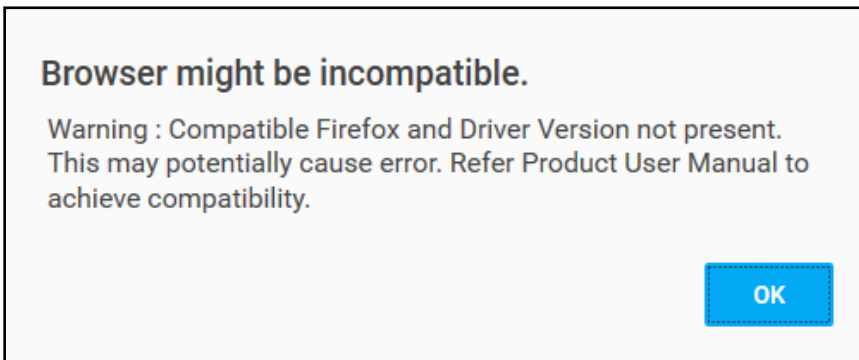
7.3 Firefox Browser Compatibility Check

Firefox automation in AE works on the basis of the geckodriver.exe, since Firefox browser version is subject to change, its driver must be compatible with the new or changed version of the browser.

Automation Studio provides an OOB (Out of box) compatibility alert for Browser and Driver mismatch.

Following alerts are displayed notifying the compatibility of geckodriver and the browser:

Automation Studio displays an alert if the compatible browser and driver version are not present.



In case of such alerts, follow the below steps.

1. Verify the browser version of Firefox.
2. Verify the compatible web driver on the web.
3. To achieve compatibility for Studio, navigate to:
 - \AutomationStudio\SeleniumDrivers automation studio path
4. Place the driver in this folder:

	Name	Date modified	Type	Size
★ Favorites				
Desktop	2.34	2/25/2019 10:19 AM	File folder	
Downloads	2.37	2/25/2019 10:19 AM	File folder	
Recent places	2.38	2/25/2019 10:19 AM	File folder	
This PC	chromedriver	2/25/2019 9:52 AM	Application	6,256 KB
Desktop	geckodriver	2/25/2019 9:52 AM	Application	9,458 KB
Documents	IEDriverServer	2/25/2019 9:52 AM	Application	2,907 KB
Downloads				

5. Update the configuration file with the browser and driver entries. The file path is given below.

Automation Studio path:

AutomationStudio\ApplicationAutomationMetadata.json

```
"ComponentName": "Firefox",
"ComponentSpecifics": [
  {
    "ThirdPartyVersion": "0.24.0",
    "CompatibleComponentWithThirdParty": {
      "MinVersion": "59",
      "MaxVersion": "66"
    }
  }
],
"SupportedVersions": [
  "59", "60", "61", "62", "63", "64", "65", "66"
]
```

6. Find the user specific RootDirectory folder which can be located from the key RootDirectory present inside PrimarySettings-->IODiskPaths.
7. Inside this folder, delete the SeleniumDrivers folder and ApplicationAutomationMetaData.json file and restart the studio.
8. Restart the studio and check the browser compatibility with the new driver.

Note:

- The browser and driver versions mentioned in the JSON file are not sacrosanct. The combination of application, browser version, and the driver version must be taken into consideration.
- This feature gives a warning to the end user and does not affect the rest of the execution.

7.4 Microsoft Edge Browser Compatibility Check

Microsoft Edge, Chromium-based browser automation in AE works using the msedgedriver.exe driver. The Microsoft Edge browser version is subject to change; hence, the required driver must be compatible with the updated version of the browser.

The following warning is displayed in Automation Studio if the compatible browser and the driver version are not present.

Browser might be incompatible.

Warning : Compatible Microsoft Edge and Driver Version not present. This may potentially cause error. Refer Product User Manual to achieve compatibility.

OK

To achieve compatibility:

1. Verify the browser version and download the required web driver from the Microsoft website.
2. Navigate to **client-tools > AutomationStudio > bin > SeleniumDrivers** folder. Alternatively, navigate to **%localappdata% > EdgeVerve > AutomationStudio > bin > SeleniumDrivers** folder if you have downloaded Automation Studio from the Admin module. Place the latest downloaded web driver in the **SeleniumDrivers** folder.

Name	Date modified	Type	Size
2.34	6/10/2020 9:44 AM	File folder	
2.37	6/10/2020 9:44 AM	File folder	
2.38	6/10/2020 9:44 AM	File folder	
81.0.4044.69	6/10/2020 9:44 AM	File folder	
chromedriver.exe	7/6/2020 8:18 PM	Application	6,256 KB
DriverCheck.bat	7/6/2020 8:18 PM	Windows Batch File	1 KB
geckodriver.exe	7/6/2020 8:18 PM	Application	11,339 KB
IEDriverServer.exe	7/6/2020 8:18 PM	Application	2,907 KB
msedgedriver.exe	9/2/2020 6:18 PM	Application	10,556 KB

3. Navigate to **client-tools > AutomationStudio > bin > ApplicationAutomationMetadata.json** configuration file. Alternatively, navigate to **%localappdata% > EdgeVerve > AutomationStudio > bin > ApplicationAutomationMetadata.json** configuration file, if you have downloaded Automation Studio from the Admin module. Update the browser and driver entries specific to Microsoft Edge browser. Below is a sample screenshot of the entries:

```
{
  "ComponentName": "microsoftedge",
  "ComponentSpecifics": [
    {
      "ThirdPartyVersion": "84.0.522.49",
      "CompatibleComponentWithThirdParty": {
        "MinVersion": "84",
        "MaxVersion": "84"
      }
    }
  ],
  "SupportedVersions": [
    "84"
  ]
},
```

4. Locate the user specific root directory folder (OSDisk (C:) > Users), and then navigate to Documents > EdgeVerveSystems Limited > AssistEdge_CE > PA folder. Delete the SeleniumDrivers folder and the ApplicationAutomationMetaData.json file.
5. Restart Automation Studio.

The browser is now compatible.

Note:

- The browser and driver versions mentioned in the JSON file are not sacrosanct. The combination of application, browser version, and the driver version must be taken into consideration.
- This feature only gives a warning and does not affect the execution of automation process workflow.

7.5 SAP Thick Client Compatibility Check

By default, following binaries / libraries needs to be downloaded and placed at below mentioned location.

Libraries to be downloaded are:

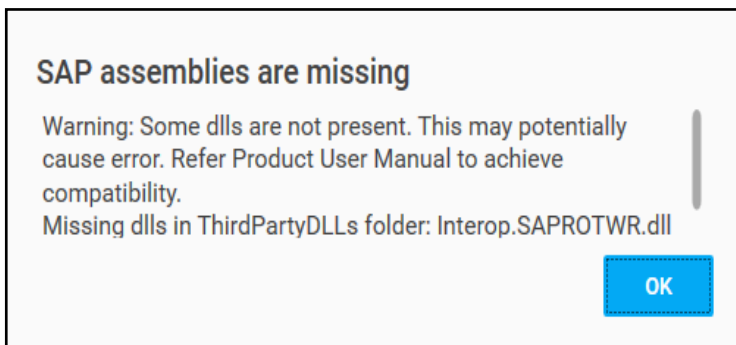
- Interop.SAPFEWSELib.dll
- Interop.SAPROTWR.dll

Locations to place the libraries:

- “..\AutomationStudio\ThirdPartyDLLs”
- “..\AutomationStudio\SAPEngine”

If these libraries are not present, then it will show alert message that required libraries not present.

Automation Studio provides an OOB (Out of box) compatibility alert when any of the dlls are missing.



In case of such alerts, follow the below steps:

1. Download and install the SAP GUI from SAP Site. After downloading and installing it, Interop.SAPFEWSELib.dll and Interop.SAPROTWR.dll could be located at C:\Program Files\SAP\FrontEnd\SapGui\SapDtsCOMFramework\.

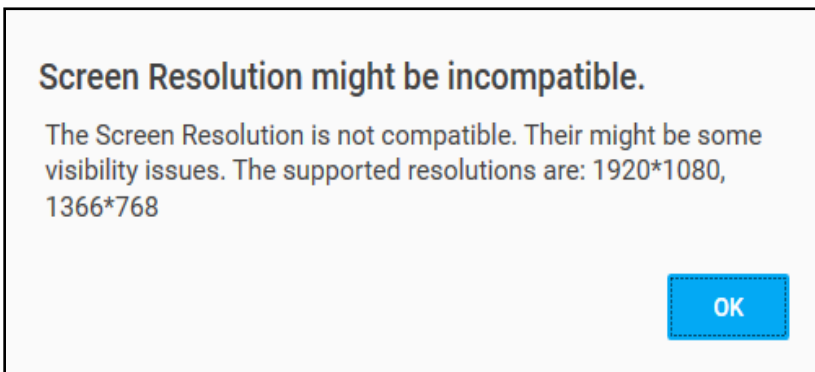
2. The third party libraries Interop.SAPFEWSELib.dll and Interop.SAPROTWR.dll need to be added to the ThirdPartyDLLs folder and SAPEngine folder respectively.
3. Automation studio needs to be restarted to take effect to changes done.
4. In case there is a requirement to place the SAP third party dlls in some other folder or a new third party dll is given by SAP.
5. Then folder name and dll name need to be updated in the ApplicationAutomationMetadata.json file. The snapshot of the json file entries is as shown below.

```
"SAPDetails": {
  "FolderNames": [ "ThirdPartyDLLs", "SAPEngine" ],
  "DllNames": [ "Interop.SAPROTWR.dll", "Interop.SAPFEWSELib.dll" ]
}
```

7.6 Screen Resolution Compatibility Check

The product supports two resolutions currently, 1920*1080 and 1366*768 so if the resolution is not in these ranges then some of the components of the studio might not be visible.

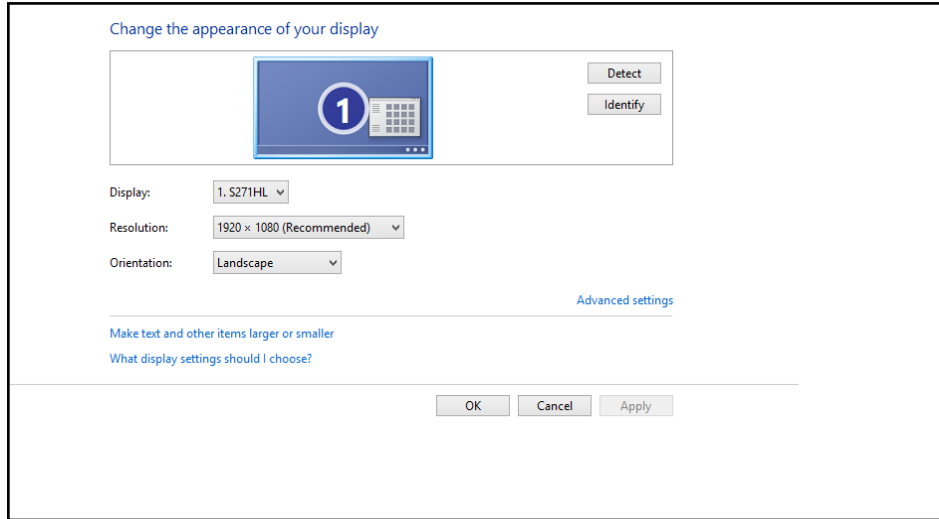
Automation Studio provides an OOB (Out of box) compatibility alert when screen resolution is incompatible.



1. To add a specific resolution to be compatible to the product, add that entry in the ApplicationAutomationMetadata.json file as shown below:

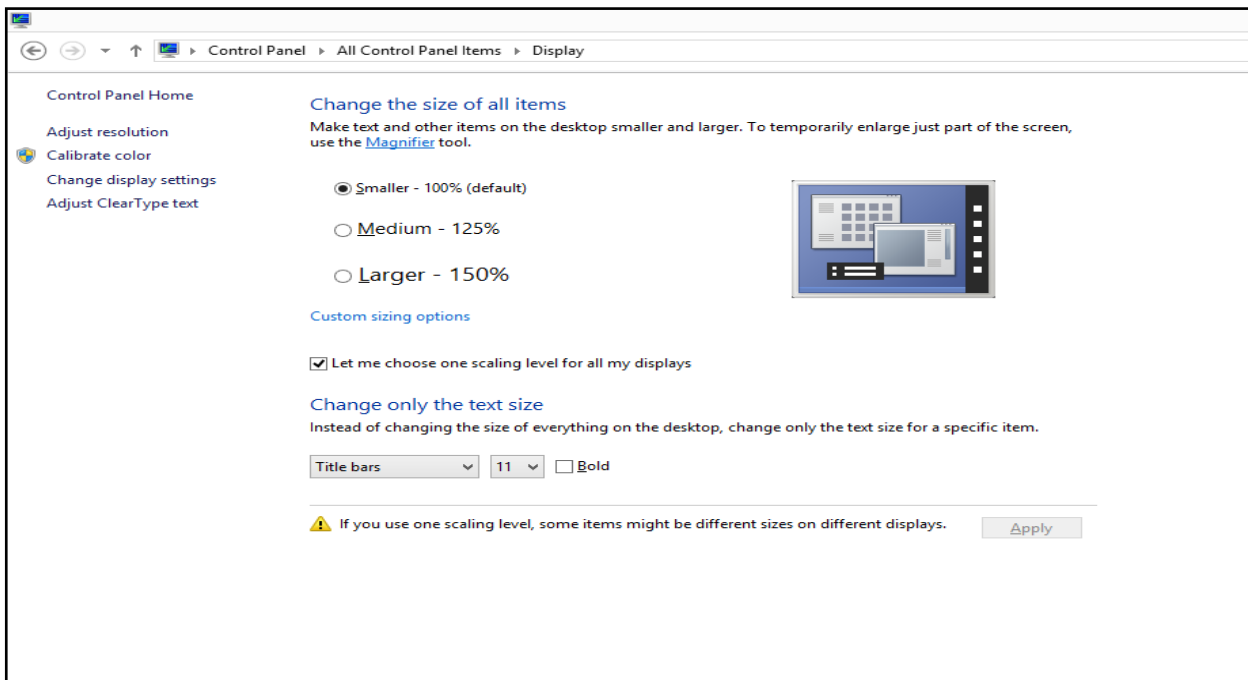
```
{
  "ComponentName": "Resolution",
  "SupportedVersions": [
    "1920*1080",
    "1366*768"
  ]
},
```

2. Change the screen resolution to prescribed range to experience the seamless workflow of the studio. The screen resolution can be changed from the personalization window.



7.7 Display Scaling Compatibility Check

Display scaling also referred to as dots per pixel is supported at 100% by the product so if it is not at 100% then some of the components of the studio might not be visible. As shown in figure below the scaling must be set at 100%.



Automation Studio provides an OOB (Out of box) compatibility alert when screen resolution is incompatible.

Display Scaling might be incompatible.

Warning: The System Display Scaling is not a 100%. There might be some visibility issues. Please set it at 100% for proper visibility.

OK

Change the screen scaling to prescribed range to experience the seamless working of the studio.

7.8 Microsoft Excel Compatibility Check

The Microsoft Excel 2013 and Microsoft Excel 2016 are supported by the product and if there is some other Excel detected on the system then alert will be shown to the user about the possibility of the incompatibility.

Automation Studio provides an OOB (Out of box) compatibility alert when excel is incompatible.

Excel might be incompatible.

Warning: Compatible Excel and Driver Version not present. This may potentially cause error. Refer Product User Manual to achieve compatibility.

OK

To add a specific, excel version to be compatible to the product, add that entry in the ApplicationAutomationMetaData.json file as shown below:

```
"SupportedVersions": [
  "Microsoft Excel 2007",
  "Microsoft Excel 2010",
  "Microsoft Excel 2013",
  "Microsoft Excel 2016"
]
```

- Incompatibility is not concrete

There is a possibility of the incompatibility as the product has not been tested with all versions of excel. So a warning is shown to alert the user. The product might also work with the excel installed so we cannot be concrete about the working of the excel outside the product boundary.

- In case there is an exception because of missing assembly Microsoft.Office.Interop.Excel.dll

This assembly is by default present in the GAC_MSIL folder.

If it is somehow missing from specified location, then user needs to manually download this assembly of version 15.0.0.0 and place it at the output directory of the studio: -

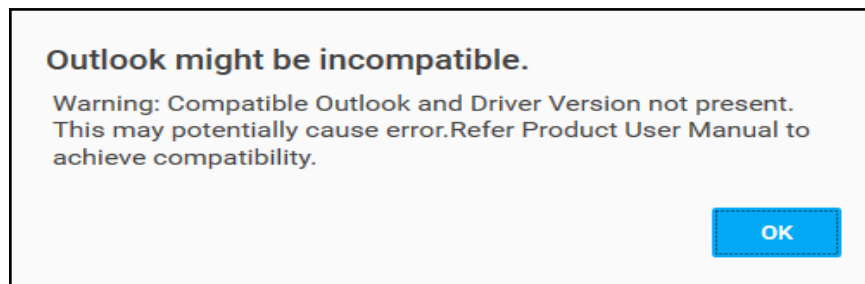
../client-tools/AutomationStudio/ (or, %localappdata% > **EdgeVerve > AutomationStudio**, if you download/access Automation Studio from the Admin module)

User can refer the FOSS document to find the links to download this assembly.

7.9 Microsoft Outlook Compatibility Check

The Microsoft Outlook 2013 and Microsoft Outlook 2016 are supported by the product and if there is some other Outlook detected on the system then alert will be shown to the user about the possibility of the incompatibility.

Automation Studio provides an OOB (Out of box) compatibility alert when outlook is incompatible.



To add a specific outlook version to be compatible to the product, add that entry in the ApplicationAutomationMetaData.json file as shown below:

```
"SupportedVersions": [
  "Microsoft Outlook 2013",
  "Microsoft Outlook 2016"
]
```

- Incompatibility is not concrete

There is a possibility of the incompatibility as the product has not been tested with all versions of outlook. So a warning is shown to alert the user. The product might also work with the outlook installed so we cannot be concrete about the working of the outlook outside the product boundary.

- In case there is an exception because of missing assembly Microsoft.Office.Interop.Outlook.dll

This assembly is by default present in the GAC_MSIL folder.

If it is somehow missing from specified location, then user needs to manually download this assembly of version 15.0.0.0 and place it at the output directory of the studio: -

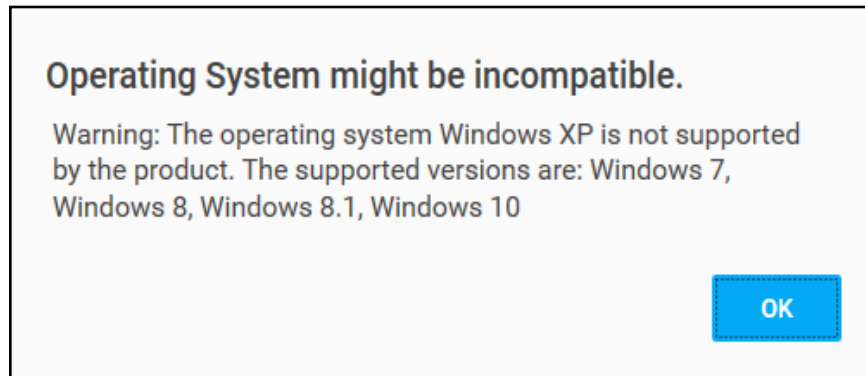
../client-tools/AutomationStudio/ (or, %localappdata% > **EdgeVerve > AutomationStudio**, if you download/access Automation Studio from the Admin module)

User can refer the FOSS document to find the links to download this assembly.

7.10 Operating System Compatibility Check

Windows 7, 8, 8.1, 10 are supported by the product and if there is some other Operating System detected on the system then alert will be shown to the user about the incompatibility.

Automation Studio provides an OOB (Out of box) compatibility alert when OS is incompatible.



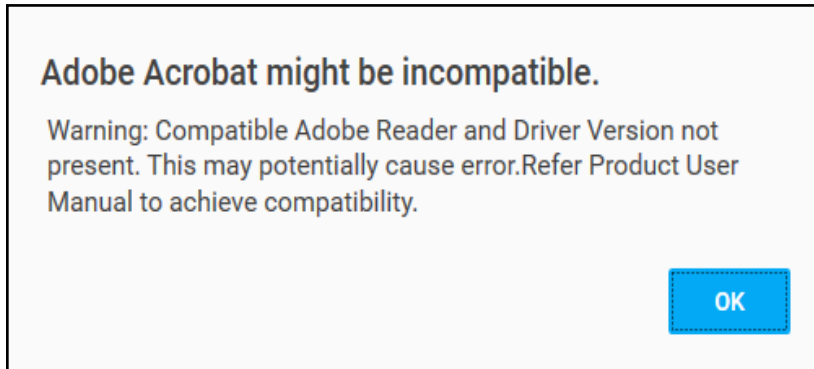
To add a specific Operating system version to be compatible to the product, add that entry in the ApplicationAutomationMetaData.json file as shown below:

```
"SupportedVersions": [  
  "Windows 7",  
  "Windows 8",  
  "Windows 8.1",  
  "Windows 10"  
]
```

7.11 Adobe Acrobat Reader Compatibility Check

In case we use the pdf versions outside the range of 1.1 to 1.7 we get alert in studio about the incompatibility. The product only supports pdf 1.1 (Acrobat 2.x) to pdf 1.7 (Acrobat DC). The compatibility checker for pdf only works for file splitter and file merger activities when the user specifies the pdf path.

Automation Studio provides an OOB (Out of box) compatibility alert when incompatible pdf is used.



In case of such alerts, follow the below steps:

1. Find out the pdfsharp.dll version compatible with the pdf that is used in automation.
2. The compatibility ranges supported by product can be visible from the ApplicationAutomationMetadata.json file.

Name	Date modified	Type	Size
pdfbox-app	1/9/2019 10:16 AM	Executable Jar File	8,080 KB
PdfSharp.dll	1/9/2019 10:16 AM	Application extens...	523 KB
Polly.dll	1/9/2019 10:17 AM	Application extens...	59 KB
Polly	1/9/2019 10:17 AM	XML Document	136 KB
PopupBlocker.dll	1/9/2019 10:30 AM	Application extens...	30 KB
ProcessExplorer.dll	1/9/2019 10:30 AM	Application extens...	58 KB
ProcessFeatureView.dll	1/9/2019 10:21 AM	Application extens...	16 KB
ProcessFeatureViewModel.dll	1/9/2019 10:30 AM	Application extens...	60 KB
ProccsLabelConfigurator.dll	1/9/2019 10:30 AM	Application extens...	42 KB
ProjectBuilder.dll	1/9/2019 10:30 AM	Application extens...	13 KB
PropertyWindow.dll	1/9/2019 10:30 AM	Application extens...	12 KB
Proton.Miscellaneous.ActionControlInter...	1/9/2019 10:30 AM	Application extens...	6 KB
Proton.Miscellaneous.Framework.dll	1/9/2019 10:30 AM	Application extens...	88 KB
Proton.Miscellaneous.XMLExtraction.dll	1/9/2019 10:30 AM	Application extens...	64 KB
Proton.Utilities.CodePluginInterfaces.dll	1/9/2019 10:30 AM	Application extens...	22 KB
Proton.Utilities.CommonUI.dll	1/9/2019 10:19 AM	Application extens...	22 KB
PROTON.Utilities.Interfaces.dll	1/11/2019 9:57 AM	Application extens...	69 KB
PROTON.Utilities.Interfaces.pdb	1/11/2019 9:57 AM	PDB File	96 KB
ReadMe	1/9/2019 10:16 AM	Text Document	1 KB
Renci.SshNet.dll	1/9/2019 10:16 AM	Application extens...	399 KB
Renci.SshNet	1/9/2019 10:16 AM	XML Document	939 KB

3. Close the studio and take backup of the pdfsharp assembly already present.
4. Download the compatible pdfSharp library from the web and replace it with the library already present in the AutomationStudio folder.

To add a specific Acrobat Reader version to be compatible to the product, add that entry in the ApplicationAutomationMetaData.json file as shown below:

```
{
  "ComponentName": "pdf",
  "ComponentSpecifics": [
    {
      "ThirdPartyVersion": "1.5",
      "CompatibleComponentWithThirdParty": {
        "MinVersion": "1.1 (Acrobat 2.x)",
        "MaxVersion": "1.7 (Acrobat DC)"
      }
    }
  ],
  "SupportedVersions": [
    "Acrobat Reader 2.x", "Acrobat Reader 3.x", "Acrobat Reader 4.x", "Acrobat Reader 5.x", "Acrobat Reader 6.x", "Acrobat Reader 7.x", "Acrobat Reader DC"
  ]
}
```

7.12 Java Compatibility Check

In case the java application uses java outside the range of Java 5 to Java 8 we get alert in studio about the incompatibility. The product only supports Java 5 to Java 8.

Automation Studio provides an OOB (Out of box) compatibility alert when incompatible java is used.

```
{
  "ComponentName": "Java",
  "SupportedVersions": [
    "Java 5", "Java 6", "Java 7", "Java 8"
  ]
}
```

Java might be incompatible.

Warning : Java run by application is not supported. This may potentially cause error. Update java to supported version for compatibility.

OK

Note:

- Java run by the application might be different from the default java installed on the system.
- The jar file used in the java application automatically runs java.exe. The java.exe run by the application is detected at runtime and its compatibility is checked with the product.

7.13 Health Report and Warning Console

Health report is an auto generated summary view of the data accumulated by Prerequisite Checker which consists of the following details:

- Evaluated Prerequisite’s name.
- Version installed on the system.
- Is the evaluated prerequisite supported by the product?
- The range of versions supported by the product.
- Name of the driver.
- Version of the driver detected.
- Is the driver supported by the installed version of the component?

This summary is auto created when the user uses any of the components evaluated by the Prerequisite Checker.

Health Report can be seen by selecting the Health Report option from the menu of Automation Studio.

Health Report						
Evaluated Prerequisite	Installed Version	Is Supported	Supported Versions	Driver Name	Driver Version	Is Driver Supported
Chrome	74.0.3729.131	No	Min version 61 and Max version 68	chromedriver.exe	Not Available	No
Display Scaling	100%	Yes	100%	Not Applicable	Not Applicable	Not Applicable
Screen Resolution	1920*1080	Yes	1920*1080, 1366*768	Not Applicable	Not Applicable	Not Applicable
Operating System	Windows 10 Enterprise	Yes	Windows 7, Windows 8, Windows 8.1, Windows 10	Not Applicable	Not Applicable	Not Applicable
Internet Explorer	7	No	Min version 10 and Max version 11	IEDriverServer.exe	3.12.0.0	No
Firefox	67.0	No	Min version 59 and Max version 66	geckodriver.exe	Not Available	No
SAP ThICK Client	Not Applicable	Yes	Not Applicable	Required dll missing	1.0.0.0	No
Excel	Microsoft Excel 365	No	Min version Microsoft Excel 2013 and Max version Mi	Microsoft.Office.Interop.Excel.dll	15.0.4569.1506	Yes
Outlook	Microsoft Outlook 365	No	Min version Microsoft Outlook 2013 and Max version	Microsoft.Office.Interop.Outlook.dll	15.0.4569.1507	Yes
JAVA	9.0.1810.13	No	Java 5, Java 6, Java 7, Java 8	Not Applicable	Not Applicable	Not Applicable

Close

8

Process Management

Process Management lets you design, implement, and manage the business goals by automating repetitive and time-consuming processes and to streamline operations involving multiple business and IT environments. You can create different types of processes using the multiple out-of-box adapters and activities, that is fast and easy to implement.

8.1 Automation Process Workflow

An automation process workflow in Automation Studio can be designed in the Flowchart designer of the Canvas area, based on your business need. It provides you a canvas to create process workflow of the task you want to automate. You can create a flowchart, add sequences, add activities, and manage exceptions.

An automation process workflow once created and published can be re-used that increases the productivity and accuracy of your work. It also helps you to reduce the human effort required for completing the task.

Additionally, any data input used in Automation Studio that is a Personally Identifiable Information (PII) or is confidential, can be protected and masked using the Data Protection feature.

8.2 Auto-Save Process Workflow

The Studio enables to auto-save the process at the configured interval. The auto-save is performed only when the process is edited that can be identified via the asterisk (*) symbol visible at the process name tab. At any point in time, you can manually save the process by clicking the SAVE button in the Menu bar.

If you manually save the process, the auto-save interval is refreshed. For example, if the auto-save duration is set for 10 mins and you manually save the process at the 5th minute, the auto-save interval resets to 0 from 5.

The auto-save functionality and the interval can be changed by following the instructions below:

8.2.1 Auto-Save Interval

1. Access AutomationStudio folder > Automation Studio.exe.config file.
2. Locate the AutoSaveInterval key, change its value as per your requirement, and save the file.

The auto-save interval is set in minutes and by default, the duration is configured for 10 mins. The minimum value for the AutoSaveInterval can be set to 2 and maximum value can be 480 minutes.

```
<!--Interval in minutes to AutoSave the process. Min value -> 2 minutes, Max value -> 480 minutes -->
<add key="AutoSaveInterval" value="5" />
```

Note: If the interval value is above 480 minutes, the auto-save option becomes ineffective while if it is below 2 minutes, the auto-save option sets 2 as the default value.

8.2.2 Enable or Disable Auto-Save Option

1. Access AutomationStudio folder > Automation Studio.exe.config file.
2. Locate the EnableAutoSave key and change its value as per the requirement, and save the file.
 - To enable, set the value to **true**.
 - To disable, set the value to **false**.

```
<!--To enable/disable AutoSave feature. Enable -> true, Disable -> false -->
<add key="EnableAutoSave" value="false" />
```

8.3 Types of Automation Processes

The automation processes in Automation Studio are categorized into:

- [Process](#)
- [Sign In Process](#)
- [Reset Process](#)

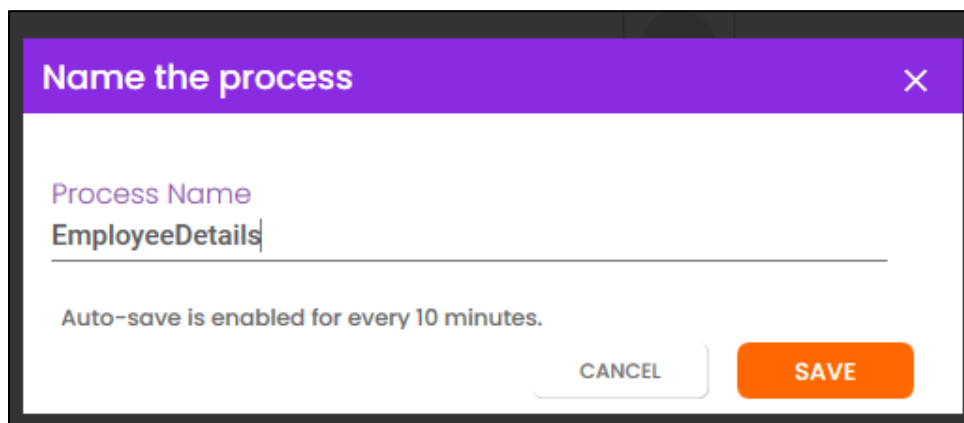
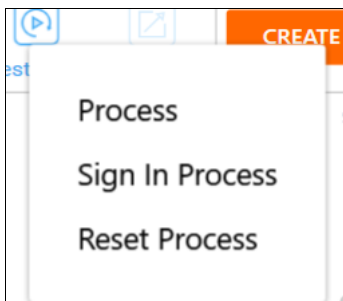
8.3.1 Process

A process in Automation Studio is the automation workflow created by the process designer as per the business need.

Take an example of printing Hello World on the console. Automation Studio offers WriteLine activity that can directly print the output in console itself. So, let's create a Hello World process using WriteLine activity.

To create a Process:

1. In the **Tool** bar, click **Create** and then click **Process**. The **Name the process** dialog box appears.



2. In the **Process Name** field, enter a name of your choice relevant to the business use case and then click **SAVE**.
3. From the **Canvas Tools** pane, drag the **WriteLine** activity and drop on to the **Flowchart** designer, below the **Start** element.



The process is created and saved even if there is any validation error, however it cannot be published.

If you want to enable-disable the auto-save option or change its duration, see the section [Auto-Save Process Workflow](#).

8.3.2 Sign In Process

A sign in process (also known as a log in process) is used to access an application upon authentication. In a typical scenario, a username and password are required to sign in to an application.

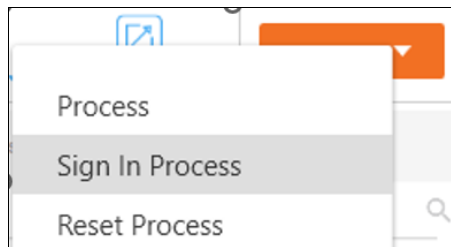
You can use a Sign In Process to automate the repetitive task of signing in to an application.

Prerequisite

Configure the application in Automation Studio that you want to automate for sign in.

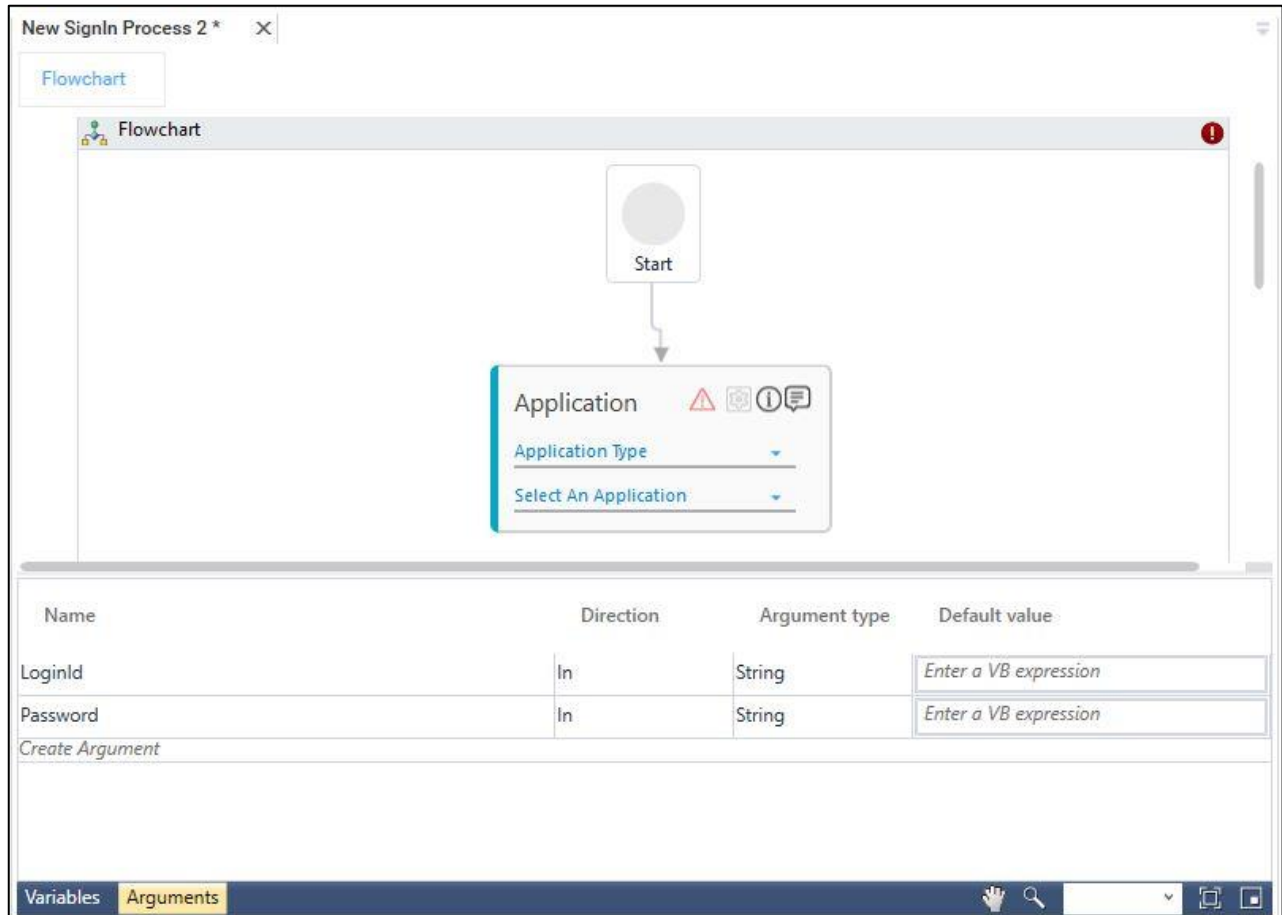
To create a signin process:


1. In the **Tool** bar, click **Create** and then click **Sign In Process**. The **Flowchart** designer opens.



2. In the **Studio** menu, from the **Canvas Tools** pane, drag **Application** activity and drop on to the **Flowchart** designer, below the **Start** element. The **LoginId** and **Password** attribute gets automatically created for a **Sign In Process**.
3. In the **Application Type** list, select the type of application you want to reset.
4. In the **Select an Application** list, select the configured application. Alternatively, you can add a new application at this level.

Note: An Application activity must be added, and the relevant application is selected; else, the system displays a validation error, and the process does not get saved



5. Click the  (**Settings**) icon to open the **Multimodal** interface. The application opens and the respective **Multimodal** interface appears.
6. Configure the fields required for signing in to the application in the same way you capture any other automation action.
7. Click **Save** to save the **Sign In Process**.

The SignIn Process is created with the name same as the selected application.

If the required application is selected as mentioned in step 4, any validation errors are ignored while saving the process, however the Sign In Process cannot be published until the errors are fixed. You can view the saved process in Homepage. To know how to use a SignIn Process, see Using SignIn Process section.

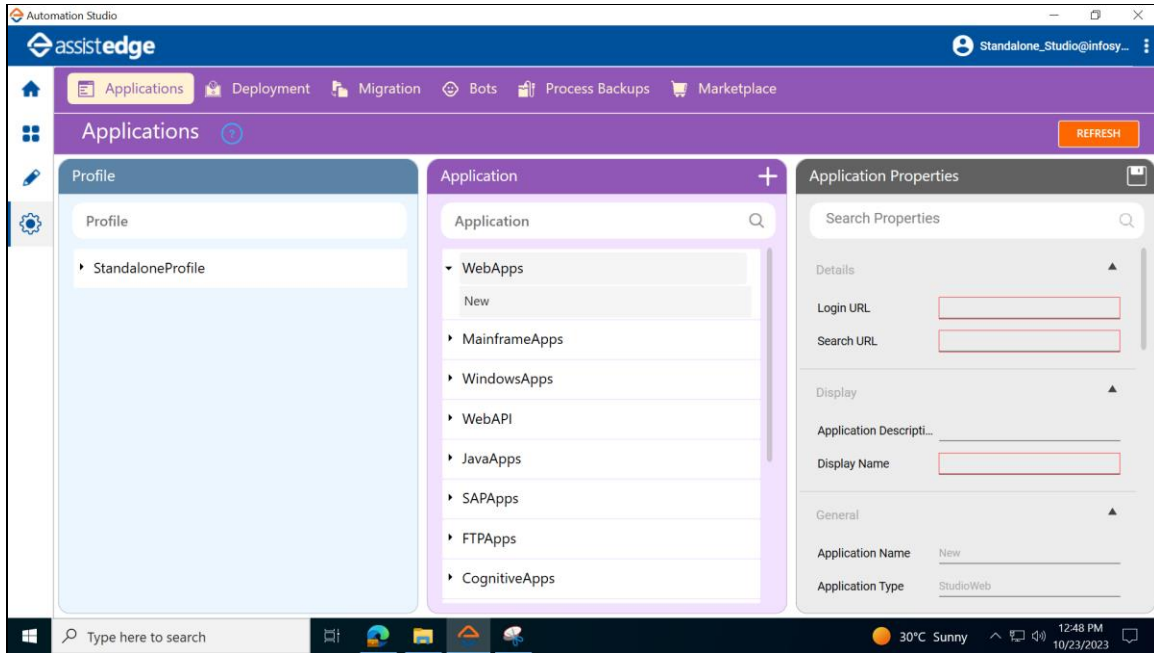
Note: If you want to enable-disable the auto-save option or change its duration see the section [Auto-Save Process Workflow](#).


Example of Creating a SignIn Process

Take an example of configuring a web application that requires signing in, this establishes a connection between Automation Studio and the application. Here we are configuring the LinkedIn account. In an organization, multiple HRs might require access to LinkedIn using admin credentials. Instead of logging into the profile of admin user individually, an automation process can be created to automate this task.

To create a sign in process:

1. In the **Admin** menu, add an application of **Application Type- WebApps**.

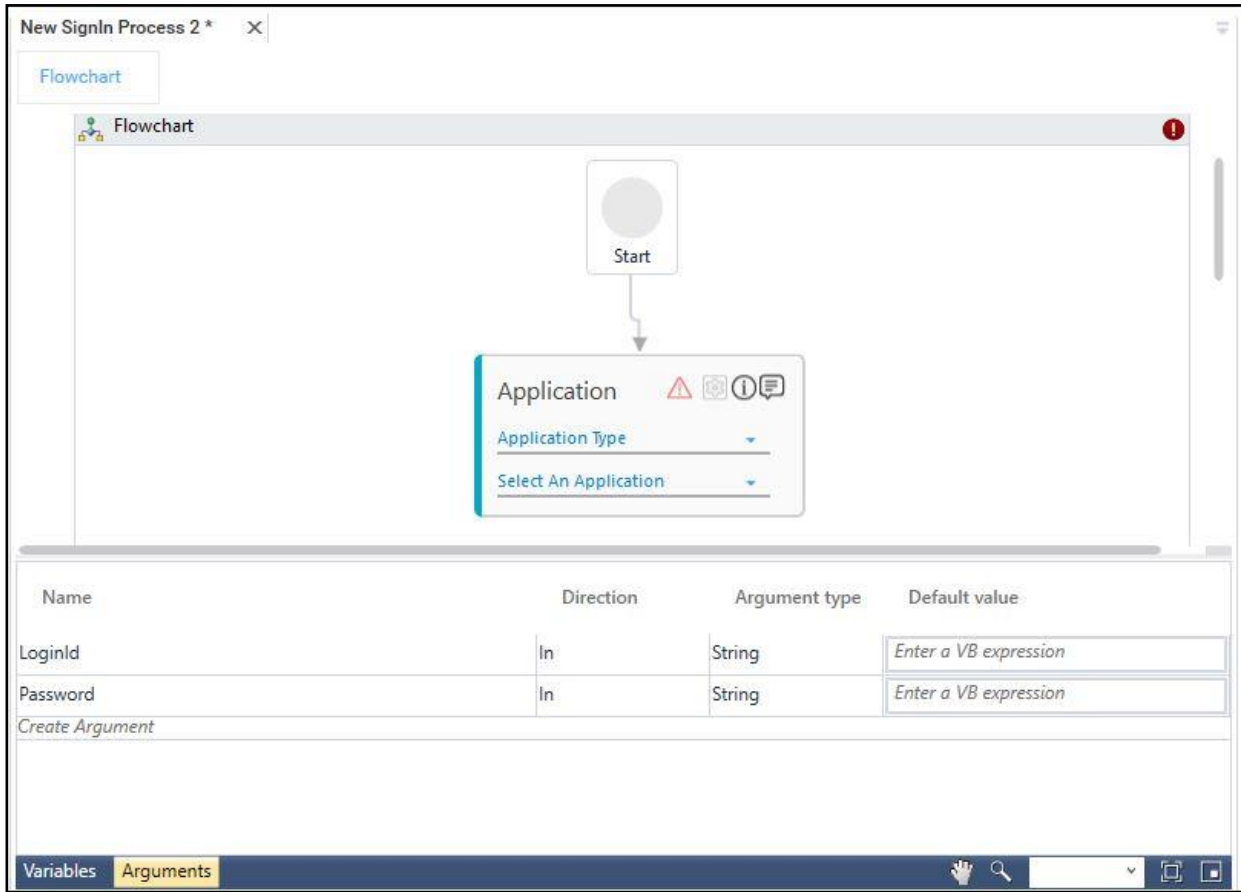


2. In the Application Properties panel:
 - In the **Login URL** field, enter the login URL of the LinkedIn page.
 - In the **Search URL** field, enter the same URL the appears post sign in to the LinkedIn page.
 - In the **Display Name**, enter a desired name of the LinkedIn application being configured.
3. Click the  (**Save Properties**) icon to save the application.


The LinkedIn application is configured.

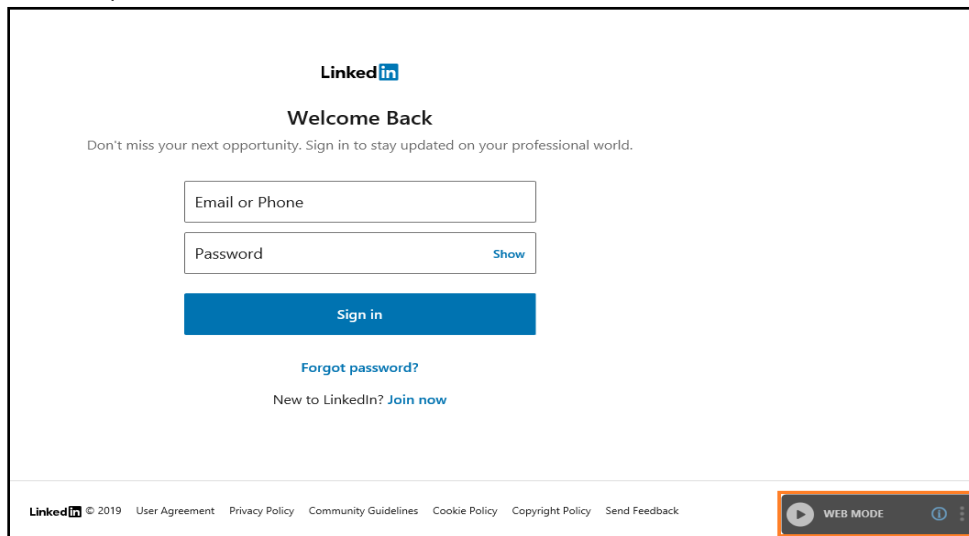
To create a sign in process:

1. In the **Tool** bar, click **Create** and then click **Sign In Process**. The **Flowchart** designer opens.
2. In the **Studio** menu, from the **Canvas Tools** pane, drag **Application** activity and drop on to the **Flowchart** designer, below the **Start** element.

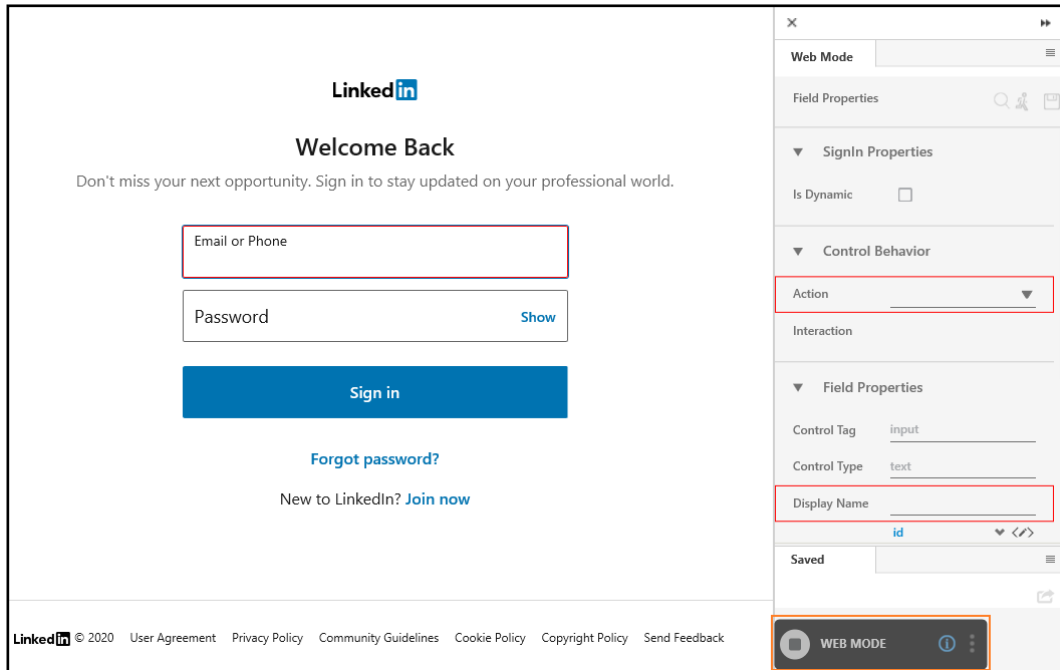


The LoginId and Password attribute gets automatically created for a Sign In Process.


3. In the Application Type list, select WebApps.
4. **In the Select an Application list, select LinkedIn.** Alternatively, you can add a new application at this level.
5. Click the  (**Settings**) icon to open the **Multimodal** Interface. The **LinkedIn Login, Sign in** page and the **WEB MODE** of the **Multimodal** interfaces opens.




6. Click the **Play** button of the **WEB MODE** to capture the user id field of the login page.
7. Hover over the **Email or Phone** text box. Click the field when it gets highlighted with red box.





The Field Properties panel of the Web Mode appears.

8. In the **Action** list, select **Set Attribute**.
9. In the **Input Source** list, select **LoginId**. By default, the **LoginId** attribute gets created for a SignIn process.
10. In the **Display Name** field, enter a desired name of the user Id field captured.
11. Click the  (**Save**) icon, to save the details of the user id field captured. It starts appearing in the **Saved** pane of the **Field Properties** panel at the bottom.
12. Click the **Play** button of the **WEB MODE** to capture the password field of the login page.
13. Hover over the **Password** text box. Click the field when it gets highlighted with red box.

The Field Properties panel of the Web Mode appears.

14. In the **Action** list, select **Set Attribute**.
15. In the **Input Source** list, select **Password**. By default, the **Password** attribute gets created for a SignIn process.
16. In the **Display Name** field, enter a user defined name of the password field captured.
17. Click the  (**Save**) icon, to save the details of the password field captured. It starts appearing in the Saved pane of the **Field Properties** panel in the bottom.
18. Click the **Play** button of the **WEB MODE** to capture click on the Sign in button to login to the sign in page.
19. Hover mouse over the **Sign in button** and then click once it gets highlighted with red box.

The Field Properties panel of the Web Mode appears.

20. In the **Action** list, select **Click**.
21. In the **Display Name** field, enter a user defined name of the Sign in button available on the web page.
22. Click the  (**Save**) icon, to save the details of user id field captured. It starts appearing in the Saved pane of the **Field Properties** panel.
23. Repeat step 19-step 20 and capture all the details to form an end-to end process.
24. Click the  (**Save configured fields and return to Studio**) icon to save the field details and return to the **Studio** menu.
25. Click **Save As** to save the process.

The Sign In Process is created and saved with the name of the application. You can view the saved process in Home menu.

Note: You cannot perform test run for a Sign In Process in Automation Studio.

Configuration of Additional Controls

This section details how additional controls on a login screen of an application are configured as a part of Sign-In process creation.

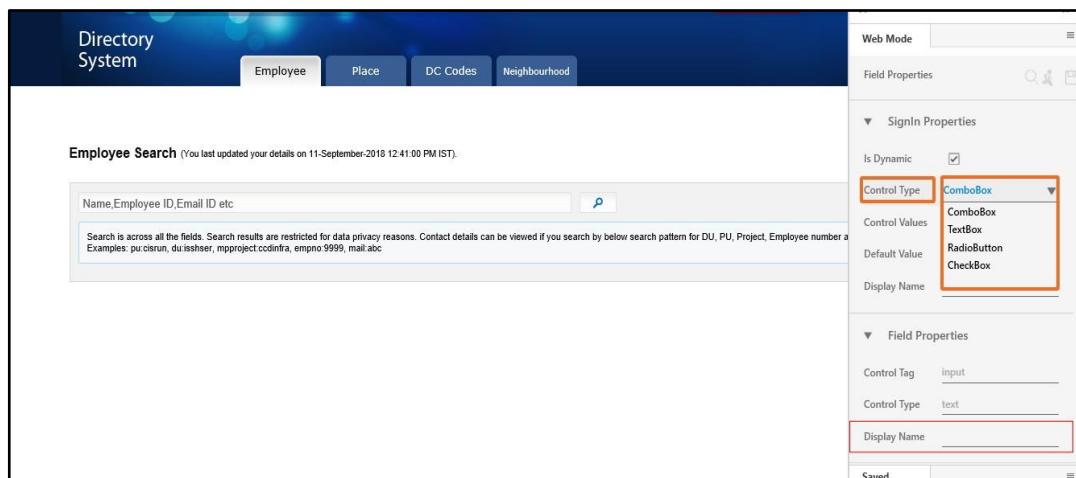
When a new sign-in process is created, two arguments: "LoginId" and "Password" are created by default, which are used to set values in the typical User Id and Password boxes on an application login screen.

Additional controls other than LoginId and Password are configured as dynamic controls in the sign in process.

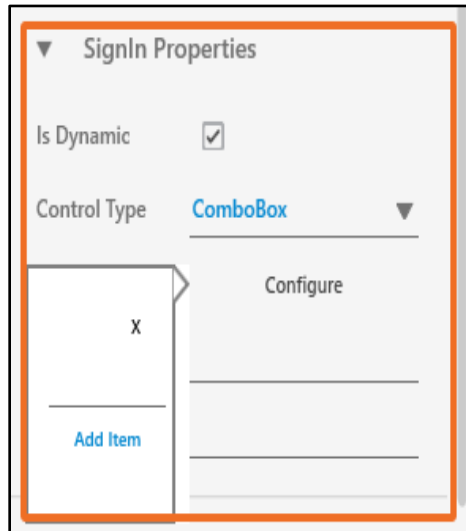
Configure the dynamic controls using the following steps:

1. In the application opened for field configuration in the Studio tab, click on **"Start Field Configuration"** and select the **"additional control"** to be configured.

The role drop-down is being configured in the below screenshot.



2. Check the **“Is Dynamic”** checkbox under the **“Field Properties”** to mark it as an additional control.
3. Choose the **“Control type”** to be configured for the selected controls. There are four types of controls.
 - a. Combo box
 - b. Textbox
 - c. Radio button
 - d. Checkbox
4. If ComboBox is selected, configure the values to be shown in the dynamic drop-down control on Credential Manager in CT or the Sign In Manager in AssistEdge Engage. The Display Name configured for the field is used as a label for the control.



5. If the textbox is selected, configure the default value to be shown in the textbox control.
6. For radio button and checkbox, the default value is true/false. Based on the value, the radio button and checkbox is checked/unchecked.
7. An argument is created for every configured dynamic control. Value of the argument is set in the configured control when the sign in process is executed.
 - a. Test the Signin process using the **“Setup Environment”** option.
 - b. Test Run is not enabled for the sign in processes.
8. Close the **“application”** tab and double click on the **“Application activity”** to view the field configuration.

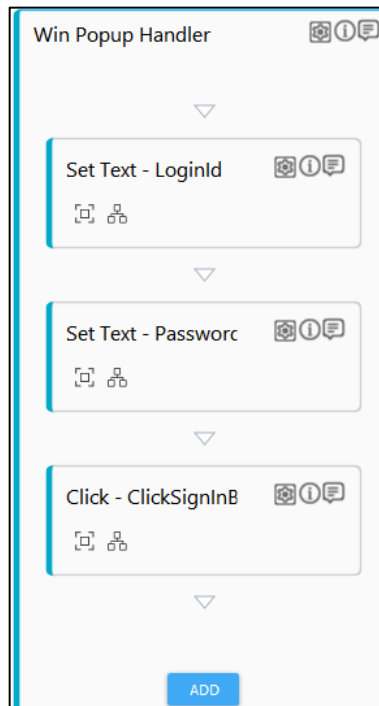
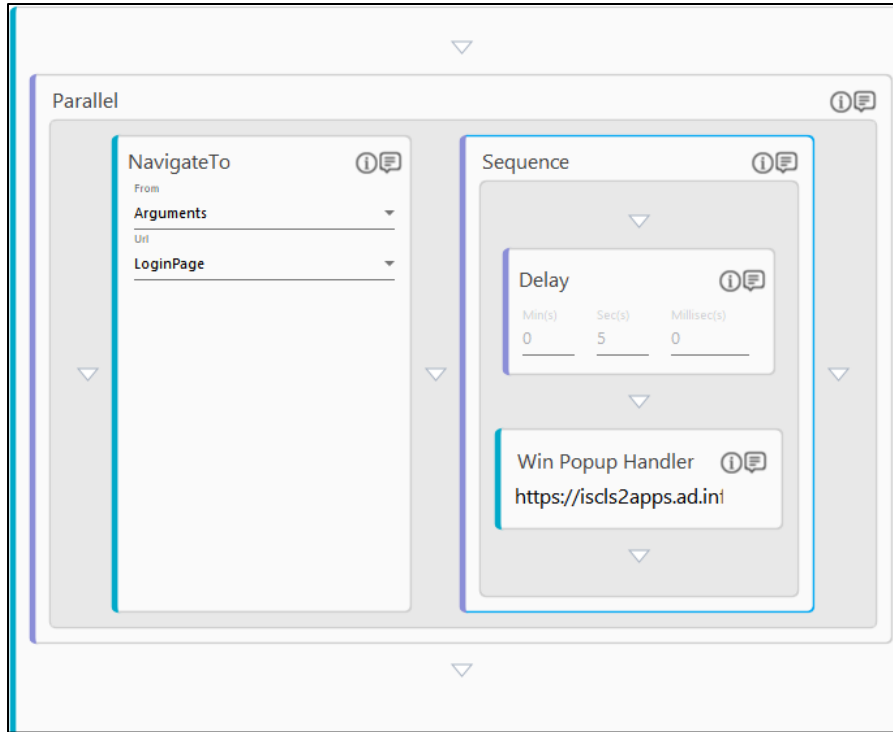
Handling Basic Authentication for Chrome and Firefox

1. Add application with different initial sign in url which doesn't have basic authentication enabled such as <https://www.bing.com>.
2. Create sign in process for the application.
3. Add a parallel activity in sign in process.
4. Add Navigate to activity inside parallel block as 1st block and navigate to actual sign in url which presents the basic

authentication dialog.

- 5. Add Sequence as 2nd block which should contain delay activity of several seconds and steps to interact with authentication dialog using windows scraping.

Following screenshot depicts this configuration:



What's Next

Once the process is created, you can setup the environment to set the user ID and password. See [Setup Environment](#) to know how to setup the environment for process execution in Automation Studio.

8.3.3 Reset Process

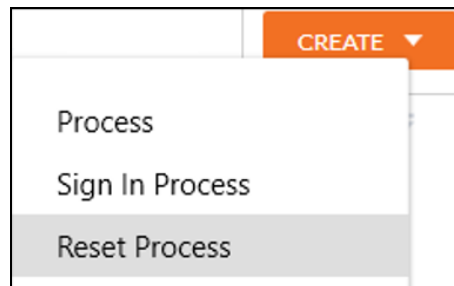
The Reset Process helps you to change the current state of the application involved in your process to the configured state.

Prerequisite

Configure the application in Automation Studio that you want to automate for reset.

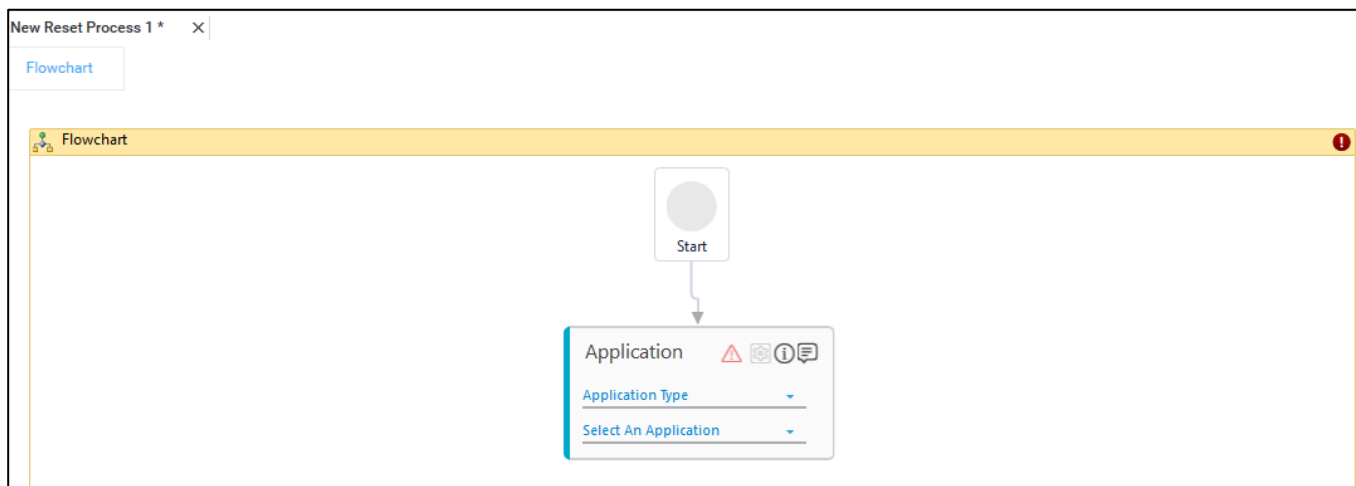
To create a reset process:


1. In the **Tool** bar, click **Create** and then click **Reset Process**. The **Flowchart** designer opens.



2. In the **Studio** menu, from the **Canvas Tools** pane, drag **Application** activity and drop on to the Flowchart designer, below the **Start** element.
3. In the **Application Type** list, select the type of application you want to reset.
4. In the **Select an Application** list, select the configured application. Alternatively, you can add a new application at this level.

Note: An Application activity must be added, and the relevant application is selected; else, the system displays a validation error and the process does not get saved.



5. Click the  (**Settings**) icon to open the **Multimodal** interface. The application opens and the respective **Multimodal** interface appears.
6. Configure the reset state of the application in the same way you capture any other automation action.
7. Click **Save As** to save the process.

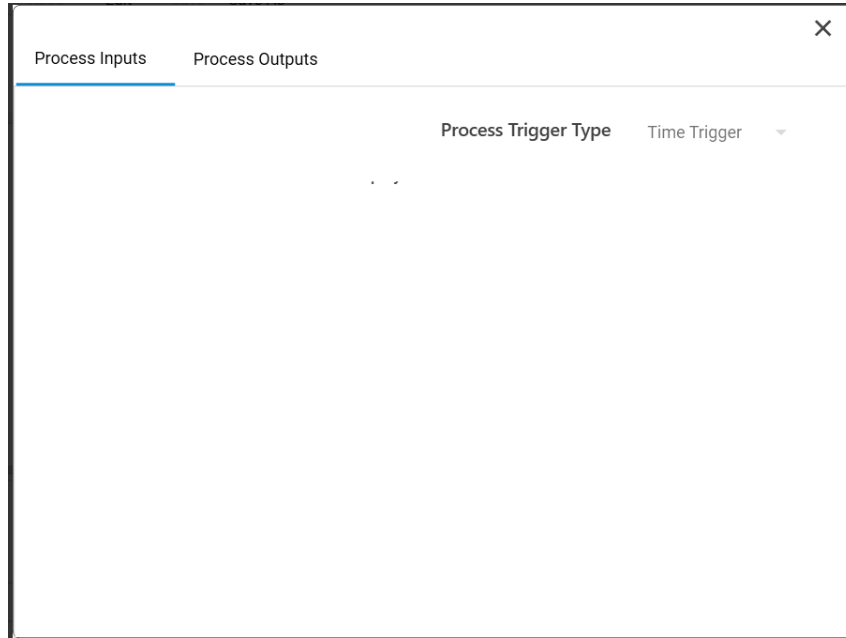
The Reset Process is created and saved with the name of the application.

If the required application is selected as mentioned in step 4, any validation errors are ignored while saving the process, however the Reset Process cannot be published until the errors are fixed. You can view the saved process in Homepage. To know how to use a Reset Process, see [Using Reset Process](#) section.

Note: If you want to enable-disable the auto-save option or change its duration see the section [Auto-Save Process Workflow](#).

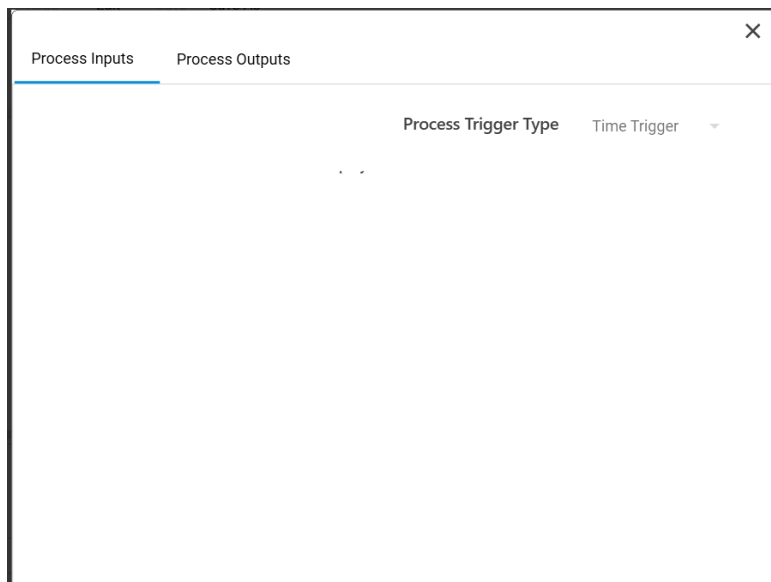
8.4 Process Input/Output

It is defined as the input parameter based on which the process execution is initiated. The values of the process input parameter can be provided by other process workflow or by the trigger created in EVA Bot while generating the automation request. Refer AE-EVA-bot Guide.pdf to know the details about triggers.



8.4.1 Using Process Input

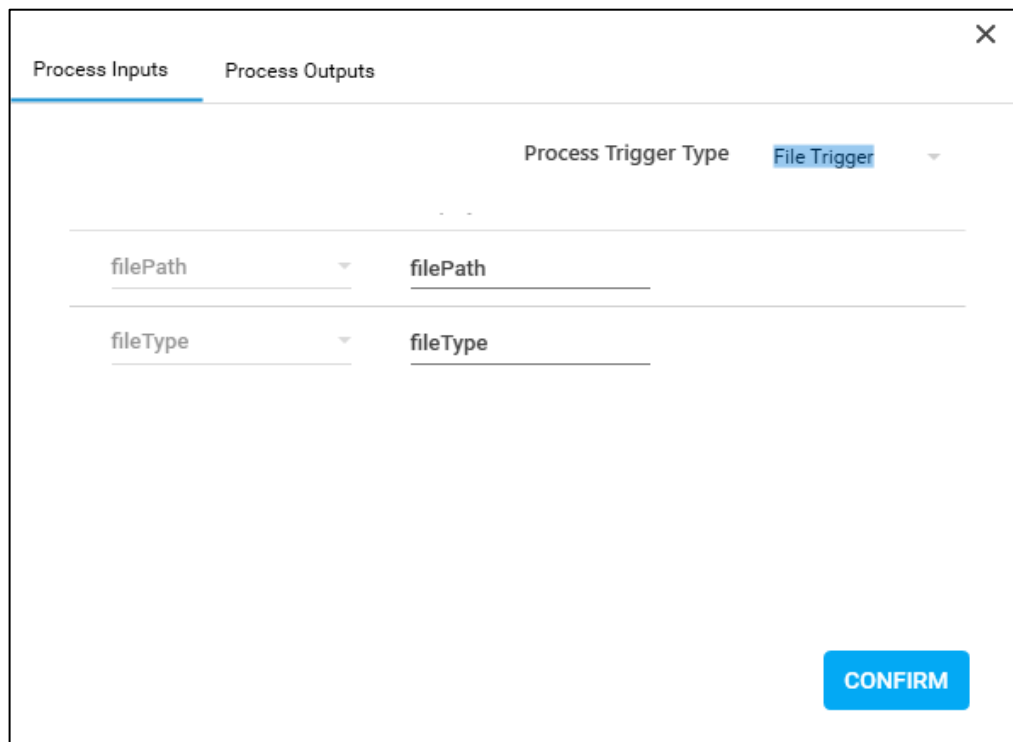
1. In the **Studio** menu, click **Process Inputs** in the **Menu** bar. The **Process Inputs** dialog box appears.



- In the **Process Trigger Type** list, select the type of trigger you want to set as process input.
- Click **Add** and then click **CONFIRM** to create the process input related to the selected **Process Trigger Type**. Click following link to see the details related to the types of trigger parameters that can be set as process input:
 - [File Trigger](#)
 - [Email Trigger](#)
 - [Time Trigger](#)

File Trigger

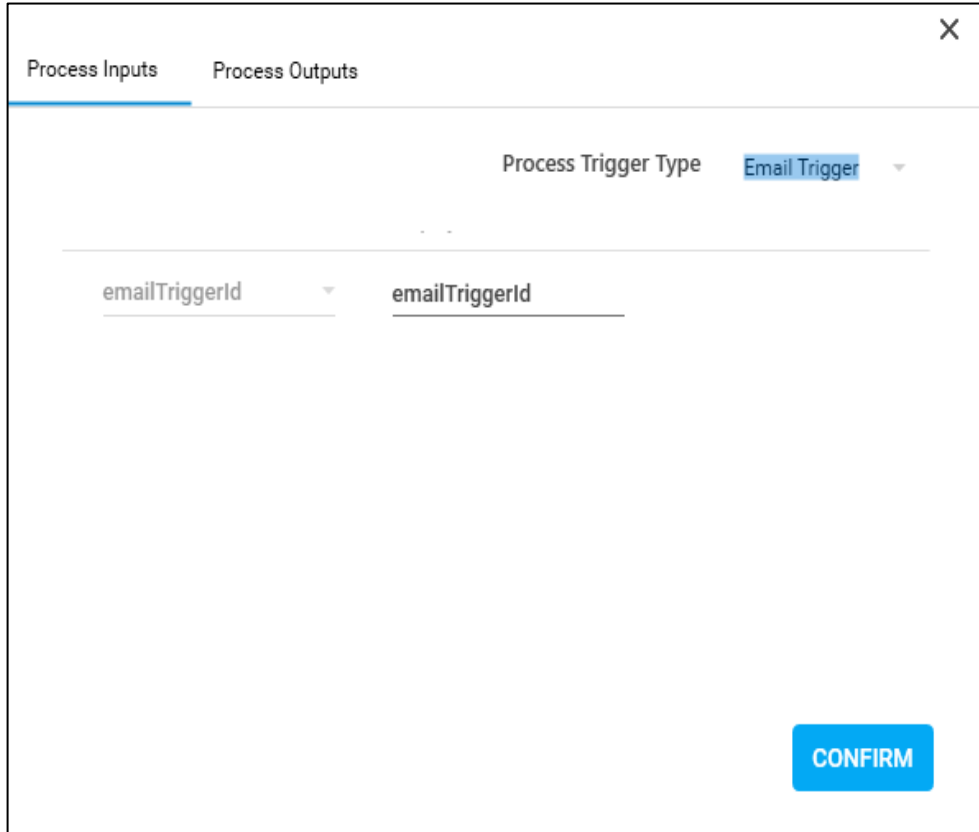
It creates two default process input parameters- filepath and filetype, that starts appearing in the Parameter bar of the selected process workflow. The values of these parameters are passed while creating a Trigger in the Control Tower.



The screenshot shows a configuration window titled "Process Inputs" with a close button (X) in the top right corner. The window is divided into two tabs: "Process Inputs" (selected) and "Process Outputs". Under the "Process Inputs" tab, there is a "Process Trigger Type" dropdown menu set to "File Trigger". Below this, there are two rows of input parameters, each consisting of a dropdown menu on the left and a text input field on the right. The first row has "filePath" in both the dropdown and the text field. The second row has "fileType" in both the dropdown and the text field. A blue "CONFIRM" button is located in the bottom right corner of the window.

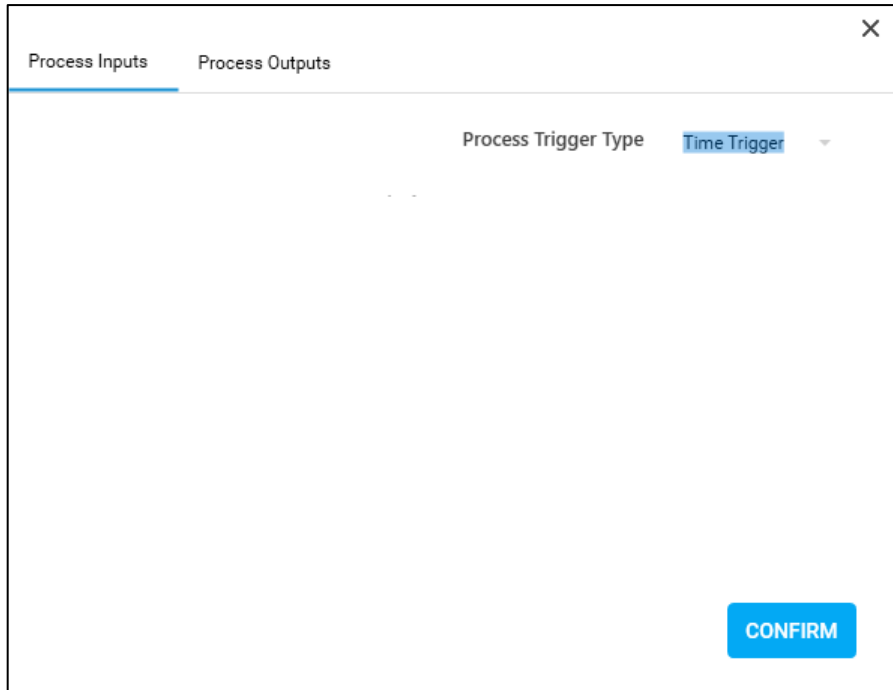
Email Trigger

It creates a default process input parameter- emailTriggerId, that starts appearing in the Parameter bar of the selected process workflow. The value of this parameter is passed while creating a Trigger in the Control Tower.



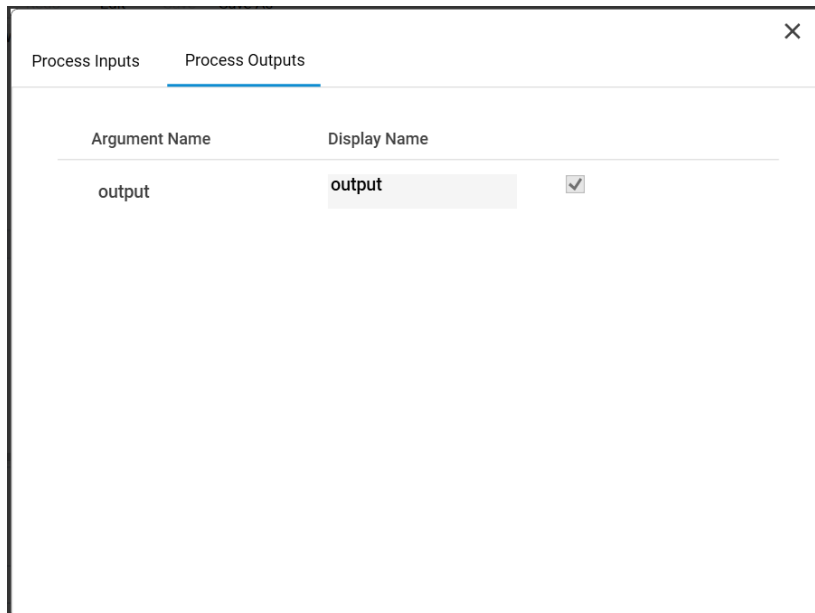
Time Tigger

Once you click CONFIRM, it adds an interval- based trigger that is created in the Control Tower.



8.4.2 Process Outputs

It is defined as the capability that allows you to view the parameters holding the output values of the process workflow such as the web application extracted values, data extracted from the Excel application, and so on.

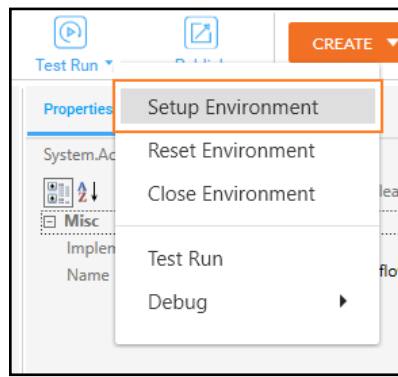


8.5 Setup Environment

Setting up the environment ensures that all the applications involved in the automation process are launched. Launching the applications prepares your applications for testing.

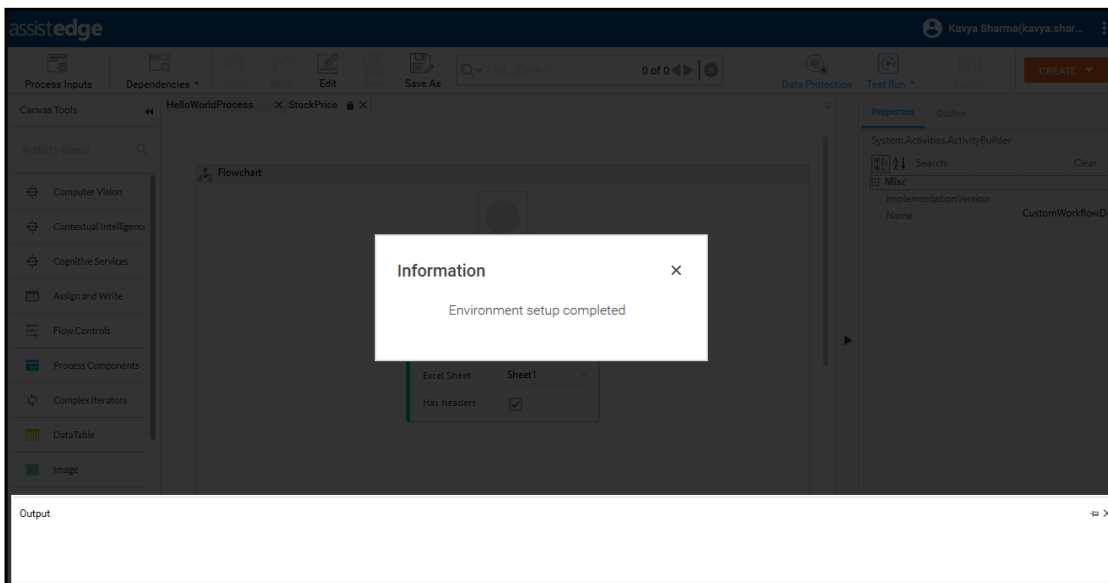
To setup an environment:

1. In the **Home** menu, open a saved process. The **Studio** menu opens along with the process available in the **Flowchart** designer.
2. In the **Tool** bar, click **Test Run**, and then click **Setup Environment**.



3. The system initiates setting up of environment launching the application/s involved in the process.

Note: Login credentials are required to complete the setup if the application configured requires sign-in.



A message for environment setup is displayed and the setup is done.

8.5.1 Reset Environment

This feature allows you to set the applications and the associated values (if any), involved in the automation process workflow to its initial state. For example, if the automation process includes searching information using the application; Reset Environment clears the search value, search results and resets the application to its previous state.

8.5.2 Close Environment

Closing an environment closes all the opened applications that were launched during setting up the environment.

What's Next

Once you have setup the environment, you can test run the process to verify your automation process steps and the desired output. See [Test Run Process](#) to know how to perform testing of the process.

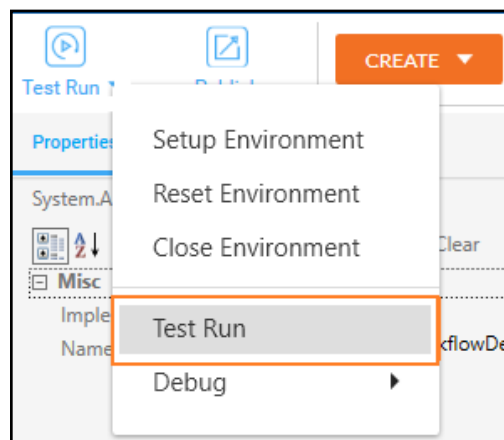
8.6 Test Run Process

Testing a process executes all the steps involved in the process workflow. The output of the process can be viewed to ensure that the desired output is received.

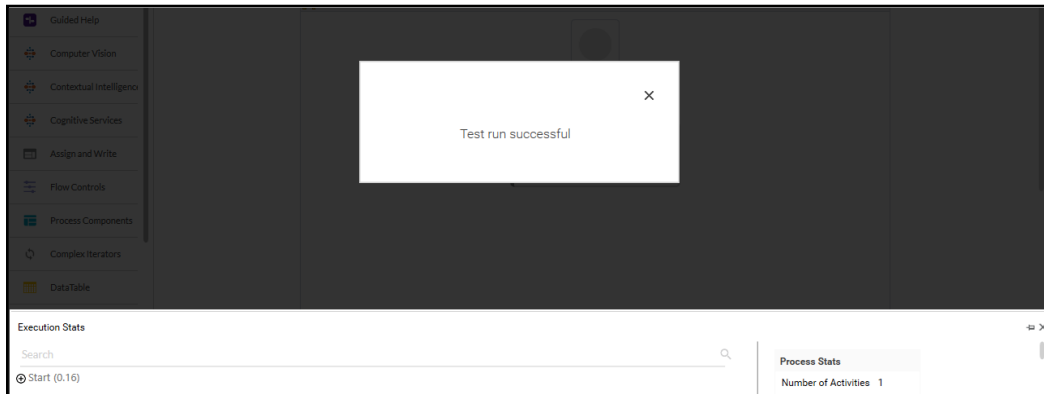
Test run can be performed after you have setup the environment. If you do not setup environment and initiate the test run, the system automatically sets up the environment once you click Test Run.

To test run a process:

1. In the **Home** menu, open the process you want to test. The process workflow opens in the **Flowchart** designer.
2. In the **Tool** bar, click the **Test Run** list and then click **Test Run**.



- The system initiates the test for the entire process workflow.

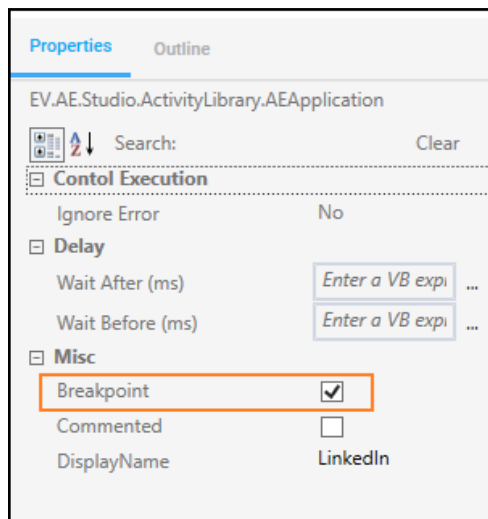


A message for successful test run is displayed. Additionally, the information related to the execution of the process is available in the Execution Stats panel. You can view [Execution Stats](#) in the Status Bar to know the details related to the process execution like CPU utilization, memory consumed.

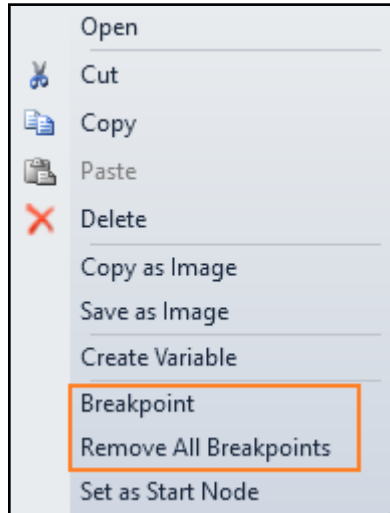
8.6.1 Debug

It allows you to debug an automation process workflow to identify and remove errors, if any. You can perform debugging by adding breakpoints to pause the process execution and then check, detect and correct the error.

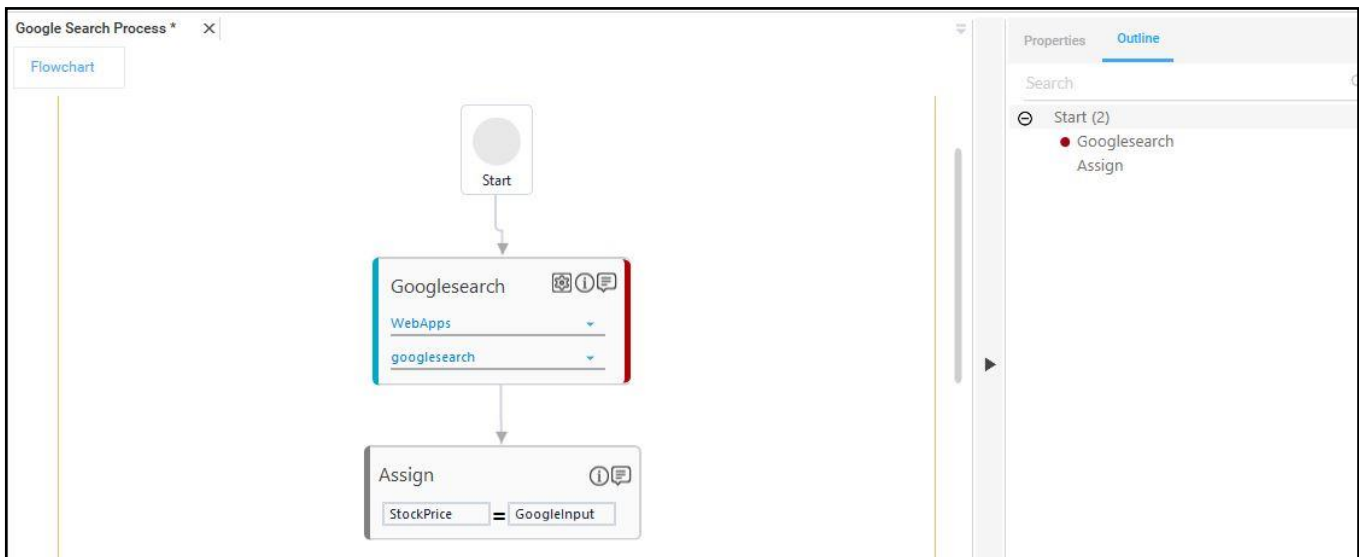
You can mark an activity as a breakpoint by selecting the Breakpoint check box in the Properties grid.



Alternatively, right click an activity and select Breakpoint or press F9 key to set a breakpoint. You can remove the breakpoint by clicking Breakpoint or Remove All Breakpoints to remove all the breakpoints at once.



The activity gets highlighted in red where breakpoint is set. Additionally, it is marked with a red dot in the Outline grid.



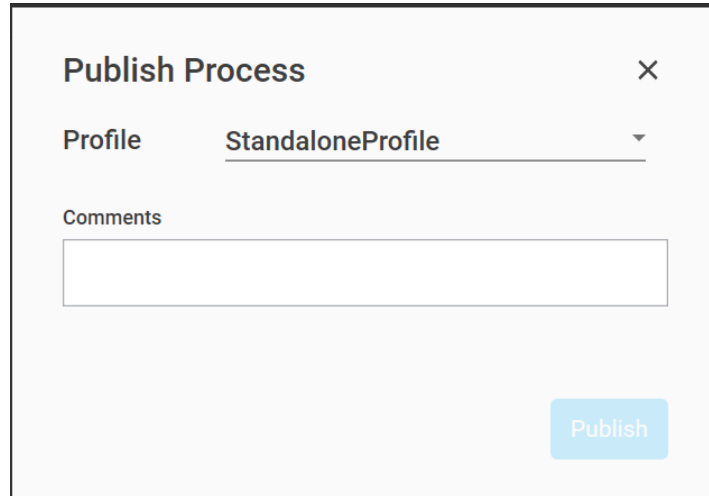
8.7 Publish Process

You must publish a process for reusability and for other users to view the process. A backup of the process is created in Automation Studio once it is published. Additionally, you can edit a published process as per your requirement.

Note: Only a saved process can be published.

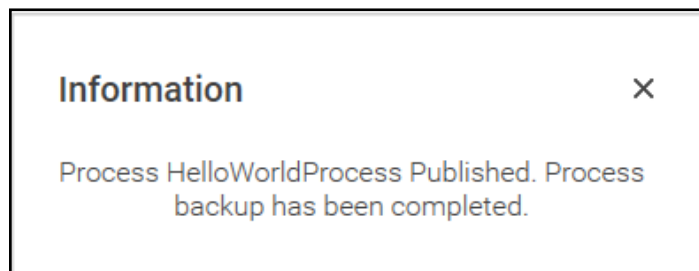
To publish a process:

1. In the **Home** menu, open the process you want to publish. The process opens in the **Flowchart** designer.
2. In the **Tool** bar, click **Publish**.



The **Public Process** dialog box appears.

3. In the **Profile** list, select the Stanalone_Profile.
4. In the Comments text box, enter any additional information about the process and then click Publish.



A message for successful publishing of the process is displayed.

Note: If you publish a process and edit it later, then another process version with the same name is created. This new process version is in the Saved state and locked for other users. If other users try to edit the Published process version for which a Saved state process version is created, then they are notified with the Process Locked message.

What's Next

After publishing the process, you can deploy the process on to the server so that they can be assigned to Robots for automation process execution. See [Deploy Process](#) to know more about deployment in Automation Studio.

8.8 Deploy Process

An automation process workflow must be deployed to send it to the server. A deployed process is assigned to a robot for execution.

You can deploy a process in automation Studio only after it is published. A process once published is available in the Deployment tab of the Admin menu for further processing. See [Deploying a Process](#) section to know how to deploy a process.

9 Canvas Tools

Canvas Tools panel contains many automation tools. This guide explains the functionalities of these tools.

- [Computer Vision](#)
- [Contextual Intelligence](#)
- [Cognitive Services](#)
- [Assign and Write](#)
- [Flow Controls](#)
- [Process Components](#)
- [Complex Iterators](#)
- [Data Table](#)
- [Image](#)
- [Files](#)
- [Word](#)
- [Exception Handling](#)
- [E-Mail](#)
- [SAP Thick Client](#)
- [SharePoint Activities](#)
- [PDF Template Creator](#)
- [Excel Recorder](#)
- [Process Recorder](#)
- [Oracle EBS Codeless Configuration](#)
- [PeopleSoft](#)
- [Siebel](#)
- [Automation of Siebel HI Application using Microbots](#)
- [Automation of Oracle E-Business Suite using Micro-bots](#)
- [Send Keys](#)

9.1 Computer Vision

9.1.1 NIA Vision

NIA Vision is an Intelligent capability (AI) which is used as a sub routine in the Process Automation. NIA Vision is a native Computer Vision capability which identifies the object or area of interest within a digital document (pdf) through machine learning (training).

NIA Vision is not an OCR or Advanced OCR capability. In fact, it compliments usage of any OCR or advanced OCR. NIA Vision when combined with OCR tools increases OCR efficiency and reduces the overall effort, cost.

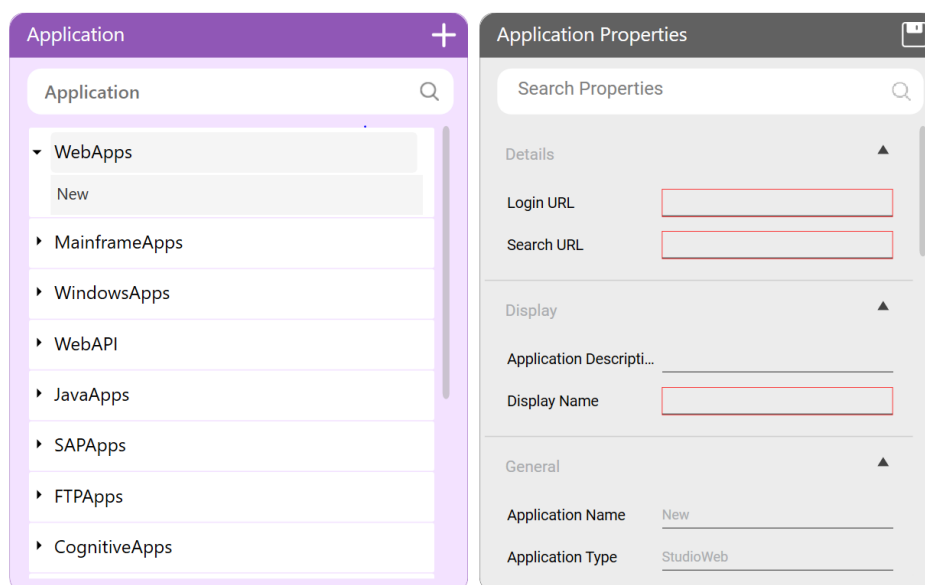
Adding of NIA Vision Application

The Studio Canvas Tools would have NIA Vision activity as the first category of tools. Before using the NIA Vision activity, you need to add the NIA Vision application via Settings (similar to addition of any new application type).

There is a new category of application called NIA Apps where NIA Vision application can be added. There are 2 key parameters needed for using NIA Vision in 18.0

- **NIA Vision Service URL:** It is an EdgeVerve hosted URL, API is called to consume the NIA services.
- **NIA Vision Subs key:** It is used for authenticated usage of NIA Vision API.

Application parameter requires URL while the subscription key is required for making a call viz – during setup environment and fetching the training models from NIA Vision.



Training

NIA Vision Activity need configuration of 3 things – the model name, the input file path (pdf) and the mapping of the output (which is a JSON).

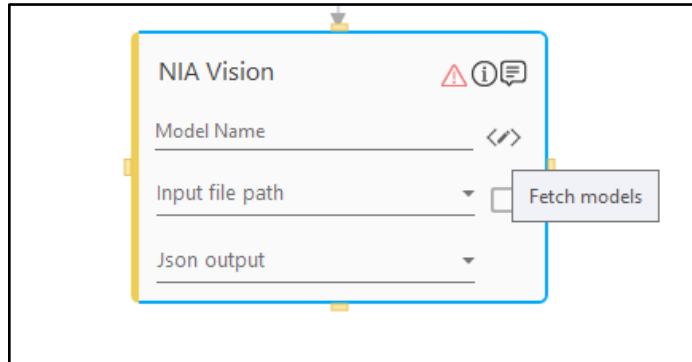
Let us take an example to move ahead with. You have to apply NIA Vision to different bank cheques. The input pdf file has one or more cheques and the process automation (use case) needs the following fields from each cheque – Date, Payee, Cheque No., Amount.

The NIA Services team is involved to perform training on sample documents to create a training model. This model can identify the above 4 fields in a cheque. Ideally, sample documents for training need 5-10 samples for each variation. Example: 5-10 samples for each cheque variation for the banks in consideration. Some cheque objects may have old structure, some may have new, some may be colored, some may be black and white, some with noise, some with different widths and so on. Whether these samples occur in 3-4 or a single document is immaterial.

NIA Workbench is used to upload the training documents, tag and create a suitable training model for the use case (with desired model accuracy) which suffices the business need.

The complete training process is part of NIA Vision services implementation.

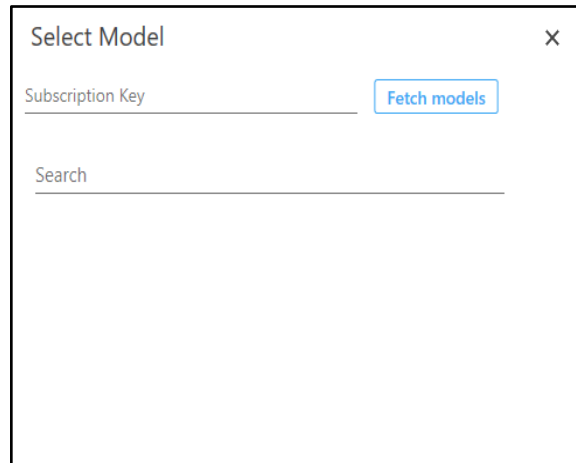
The output of the training is a training model with a user defined name. The same name (model name) will be needed in the Studio which would be applied on any new incoming pdf of cheques or a respective file.

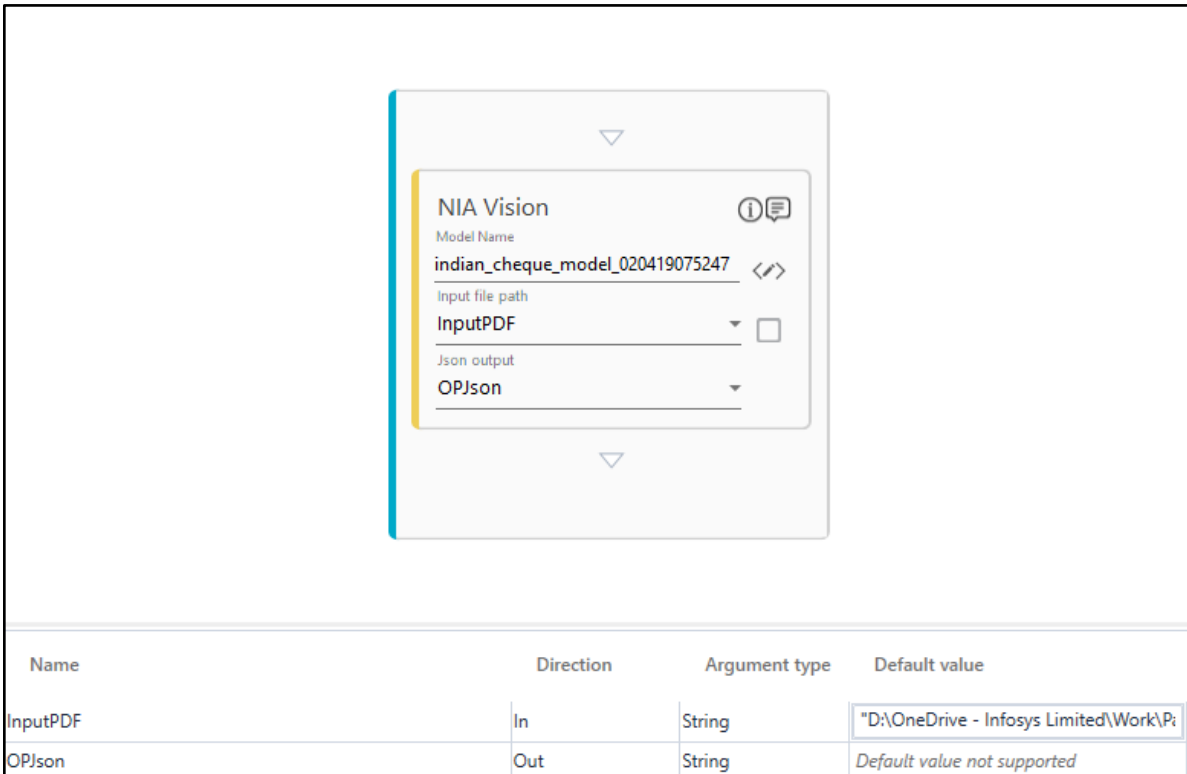


Model Selection and Configuration

Once you have the details of the training model name, you can click the fetch models button, enter the subscription key and fetch the models. A call is made to NIA Vision services and the available training models will be auto populated in a list below. You can choose the respective training model which needs to be applied for the input file.

The input file is a pdf file and the output parameter is a JSON in a string format. This argument will be an OUT type argument.



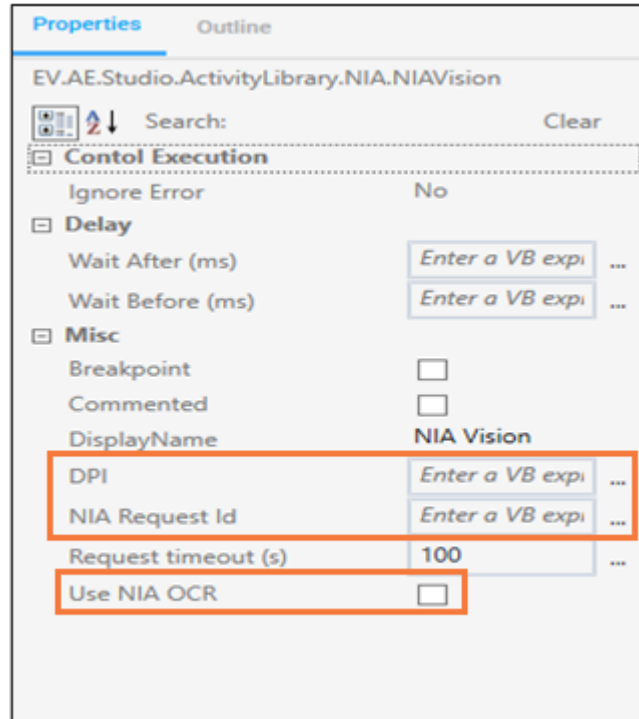


Name	Direction	Argument type	Default value
InputPDF	In	String	"D:\OneDrive - Infosys Limited\Work\Pa
OPJson	Out	String	Default value not supported

9.1.2 NIA Vision Activity Properties - NIA provided OCR

There are certain key parameters as part of the NIA Vision activity properties which can be configured for advanced usage.

- **DPI:** Using this property, you can throttle the max DPI for the file image processing.
- **NIA Request ID:** Every NIA Vision service call has a request ID. Using this property, you can track NIA Vision requests (if needed) by mapping it to an argument label in Studio.
- **Use NIA OCR:** NIA Vision has an ability to apply OCR (free Tesseract in 18.0) to the identified object (Payee for example in a cheque). For using this ability, you need to (ask the NIA services team) enable this option by creating a label for OCR (along with regex if needed) for each object. After that, when you click on use NIA OCR, you will be able to consume the OCR value of the object from NIA. In most cases, you may not use this OCR as you will rely on Advanced OCRs (MS, Google, Abbyy) where you would combine NIA + Advanced OCRs.



9.1.3 Output Consumption

The NIA Vision JSON output has the following key values:

1. Object Label – This is the identification by the NIA Vision for the object (Payee, Date, Cheque No. and Amount).
2. Object Bounding Box – Each identified object will have four co-ordinates to mark that object (Xmax, Ymax, Xmin, Ymin). These are pixel co-ordinates.
3. Object Confidence score – Ranging from 0 to 1, this is a normal distribution confidence. 0.89 = 89% confidence of the NIA Vision machine learning model that what it has identified is the respective object.
4. NIA OCR text value in the predefined label.

The JSON activity in Studio can be used to extract each of the key values from the NIA Vision output. For multiple iterations (most of the times), Advanced Loop is used to iterate over JSON (Loop within a loop). Thus you will be able to get the values for all identified objects in a key value pair format in respective predefined arguments.

PDF to Image for using any other OCR

In many use cases, you need to use NIA Vision for data classification and MS Advance OCR or any other OCR for data extraction. Because NIA Vision reduces the data extraction area and hence increases the efficiency of OCR. The PDF to image activity helps for the same. It understands the pdf and the objects within the pdf through the NIA Vision bounding box and produces images as output only for those bounding box objects. These object images can then be passed to any OCR for extraction.

The screenshot shows a configuration window titled "Pdf to image". It contains the following fields:

- Pdf file path
- Page number
- Output Image Type
- Specific Image Area
- X min co-ordinate
- Y min co-ordinate
- X max co-ordinate
- Y max co-ordinate
- Output image path

Each field has a dropdown arrow and a checkbox. At the top right, there are three icons: a warning triangle, an information 'i' icon, and a help speech bubble icon.

Exception Management with NIA Vision

When working with NIA Vision activity, the following techniques can be used for managing exceptions:

1. Using Confidence Score: Confidence Score can be used for rejecting an object if the score is too low, selecting amongst multiple objects of same type.
2. Try Catch and Regex validations.
3. Routing failure cases to other sub routine processes.

Note: Data classification and Data extraction without template is an approach where even if the Studio configuration has 100% accuracy, the approach should be to automate the process incrementally at runtime. There is a need to setup a separate process for validation of extracted data before decision making.

9.2 Contextual Intelligence

Contextual Intelligence is the tool that provides the ability of unified workforce to make key business decisions in context to business use cases of the organizations. The decision activity under this category can be used within the scope of contextual intelligence applications or it can be used independently as well. \

9.2.1 Decision Model

This activity allows you to utilize an existing machine learning (ML) model designed using Machine Learning Workbench that is deployed on the ML server such as RapidMiner server. It helps you to retrieve prediction on the new data provided.

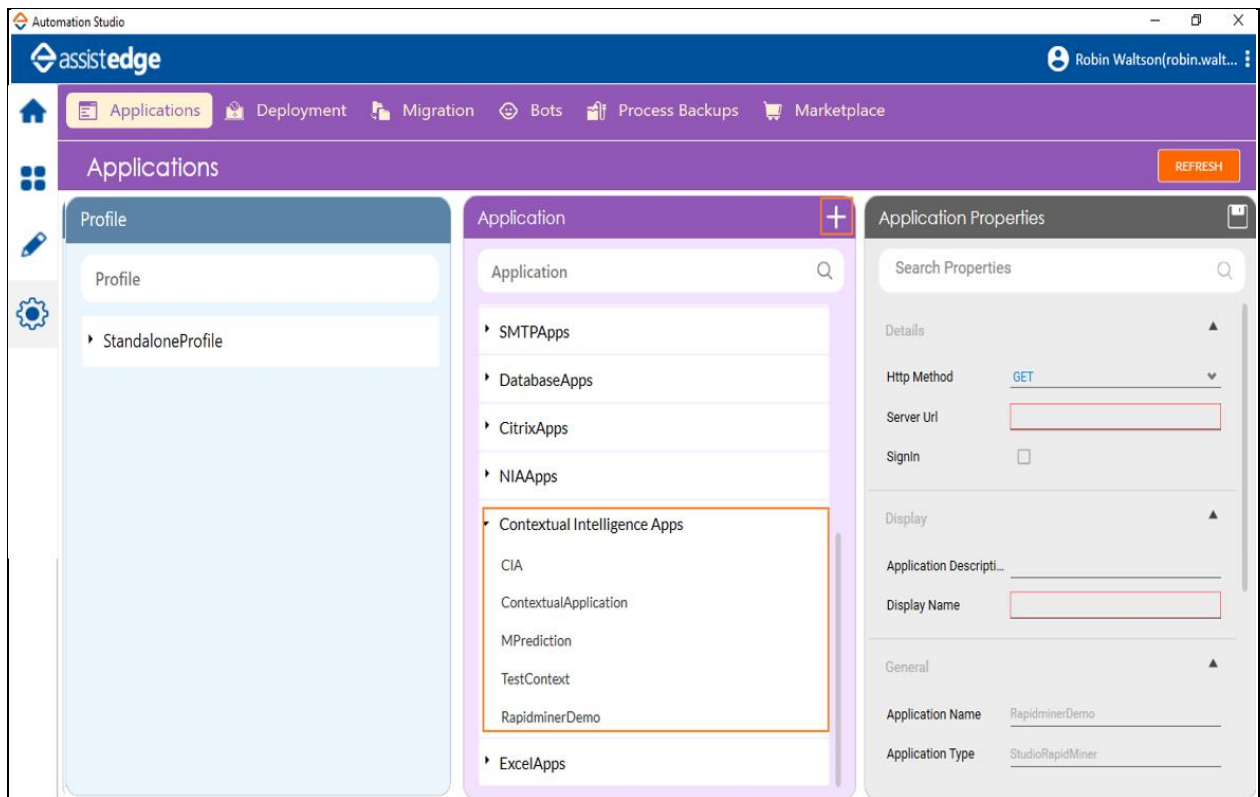
Configuring the ML server and the ML model in automation Studio deploys them as a web service creating the REST API URL. The supported http request methods that uses the web services corresponding to the ML model are:

- **GET** - predictions are made by passing input fields as the URL query parameters.
- **POST** - predictions are made by passing the JSON file.

Prerequisite

Configure the ML server in Automation Studio to utilize the web services of the available ML model. Below are the minimum required properties for configuring the intended ML server. Remaining fields are auto populated with the default values already configured in Automation Studio. If you want to change the default values, refer Contextual Intelligence Apps in Admin Capabilities section.

1. In the **Admin** menu, add an application of **Application Type- Contextual Intelligence Apps**. The mandatory fields are highlighted with red box.



2. In the Application Properties panel:

- In the **HTTP Method** list, select the request method. Available options are, **GET** and **POST**.
- In the **Server Url** field, enter the URL of the server where the ML models are deployed.
- Select the **Signin** checkbox if sign into the server is required.

- In the **Display Name**, enter a desired name of the server.

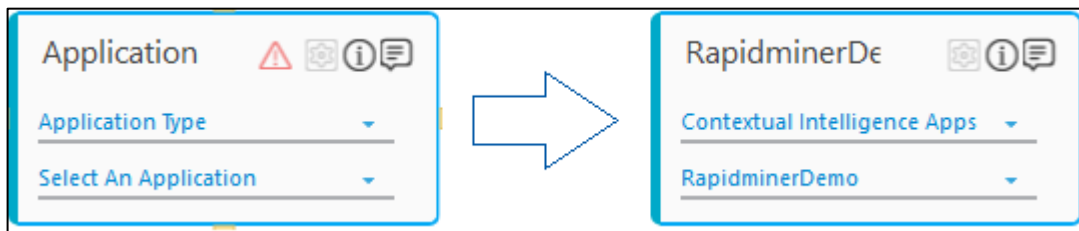
Details		▲
Http Method	GET	▼
Server Url	http://10.73.123.254:8080/	
SignIn	<input checked="" type="checkbox"/>	
Display		▲
Application Descripti...	_____	
Display Name	RapidminerDemo	

3. Click the  (**Save Properties**) icon to save the application details.

The ML server is configured in Automation Studio.

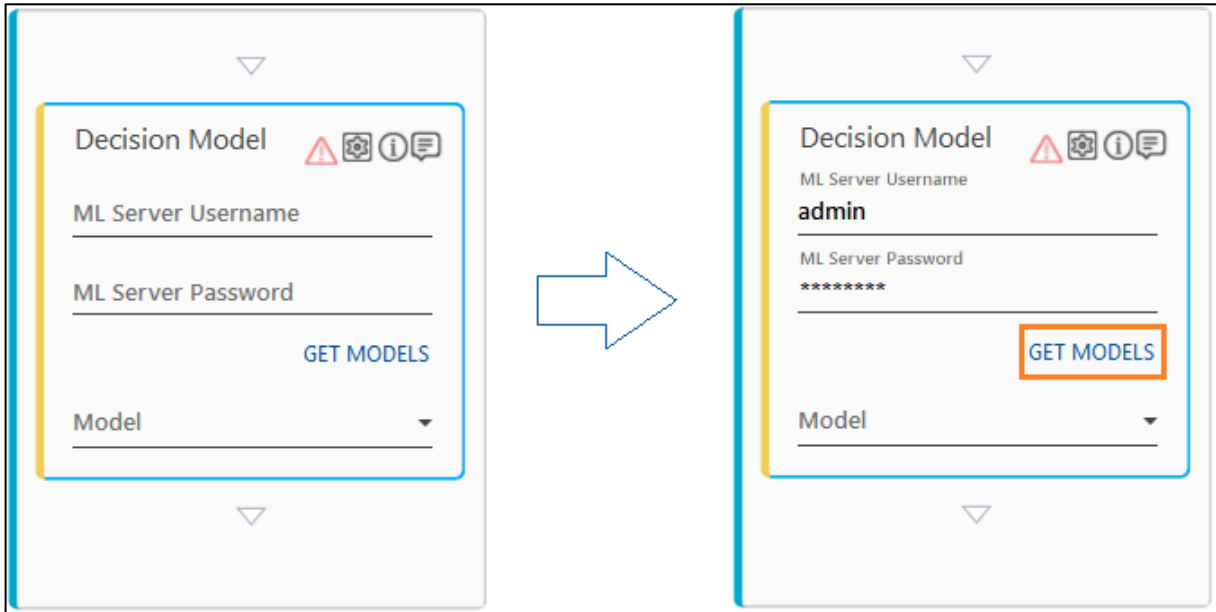
Using Decision Model Activity

1. Within the required automation process workflow, click **Process Components** in the **Canvas Tools** pane to expand the tool and view the associated activities.
2. Drag the **Application** activity and drop on to the **Flowchart designer** on the **Canvas**

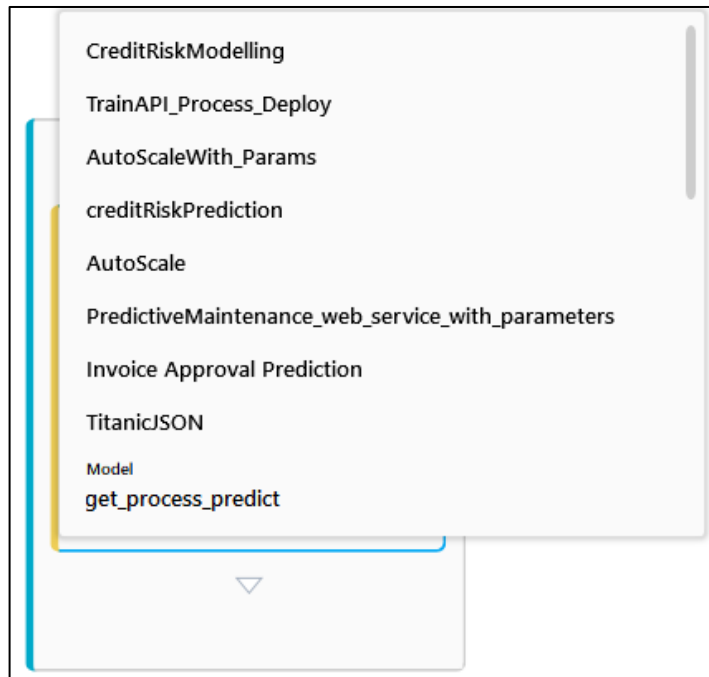


3. In the Application Type list, select Contextual Intelligence Apps.
4. In the **Select An Application** list, select the configured ML server.

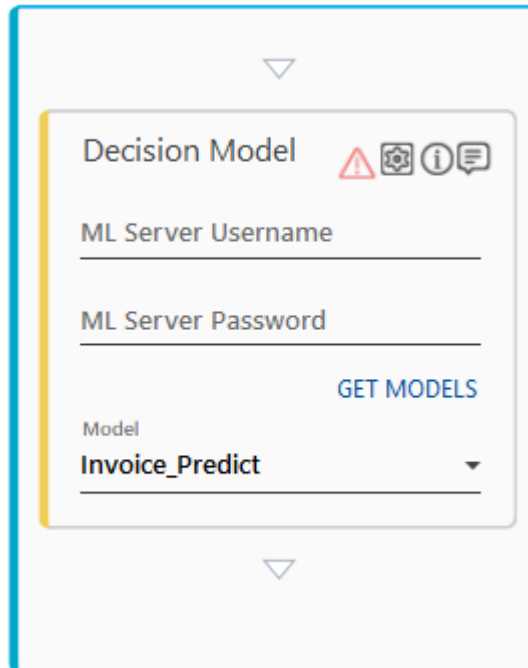
5. Double click the **Application** activity, drag the **Decision Model** activity and drop inside the **Application** activity.



6. In the **ML Server Username** field, enter the username of the configured ML server.
7. In the **ML Server Password** field, enter the password of the configured ML server.
8. Click **GET MODELS** to populate the available ML models in the Model list.



- In the **Model** list, select the required ML model as per your business need.



Decision Model


ML Server Username

ML Server Password

GET MODELS

Model

Invoice_Predict

- Click the  (**Model Input and Response**) icon to configure the ML model inputs and the output. The **ML Model Inputs and Response (GET)** or the **ML Model Inputs and Response (POST)** dialog box appears, depending on the selected request type.

The configuration of the ML model input and output varies as per the selected request method. Click any of the links below to know the step-by-step process related to the selected request type:

- GET
- POST

Decision Model Activity Properties

The properties of a Decision Model activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Application ID	It is internally created and managed by Automation Studio itself.
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as Decision Model. You can change the name as required.

Using Get Method - Decision Model

Configuring the ML server and the ML model as a web service creates the REST API URL that is displayed in the Model Input field. Send the inputs to the ML model by adding the input fields to the ML model REST API URL.

The screenshot shows the 'Model Inputs and Response (GET)' configuration window. The 'Model Input' field contains the URL: "http://10.73.123.254:8080/api/rest/process/Invoice_Predict". Below this is a 'Key Value' section with a table for 'Parameter Name' and 'Parameter Value'. A red box highlights a '+' icon in the top right corner of this table. Below the table is a 'Model Response' dropdown menu. At the bottom right, there are 'CANCEL' and 'SAVE' buttons.

Do one of the following:

- In the **Model Input** field, add the input fields directly to the REST API URL of the selected ML model web service.

OR

- Add the input fields one by one in the **Key Value** screen that automatically gets converted to the URL format.

1. Click the **+** (**Add**) icon to add input data for all the URL query parameters. By default, a single field entry is already available.
2. In the **Parameter Name** field, enter a display name of your choice.
3. In the **Parameter Value** list, select the relevant parameter holding the value of the input data. You must predefine the parameters and their respective values in the **Parameter** bar to use this option. Alternatively, select the check box available beside the **Parameter Value** field and enter an input value that you want to set as the default value.

4. Repeat steps i through step iii to add inputs for all URL query parameters.

Model Inputs and Response (GET)
Advanced Settings
✕

Model Input:

Key Value

Parameter Name	Parameter Value	+	✕
InvoiceAmt	InvoiceAmount	<input type="checkbox"/>	<input type="checkbox"/>
QltMismatch	QualityMismatch	<input type="checkbox"/>	<input type="checkbox"/>
VendCategory	VendorCategory	<input type="checkbox"/>	<input type="checkbox"/>
DeliveriesinLast3Months	Deliveries_inLastMonth	<input type="checkbox"/>	<input type="checkbox"/>
InvoiceApproval3Months	InvoiceApprovalRates_Last3Months	<input type="checkbox"/>	<input type="checkbox"/>
InvoiceApprovalRate	InvoiceApprovalRateOverall	<input type="checkbox"/>	<input type="checkbox"/>
DamagedGoodsAmt	400	<input checked="" type="checkbox"/>	<input type="checkbox"/>
PriceMismatch	PriceMismatch	<input type="checkbox"/>	<input type="checkbox"/>

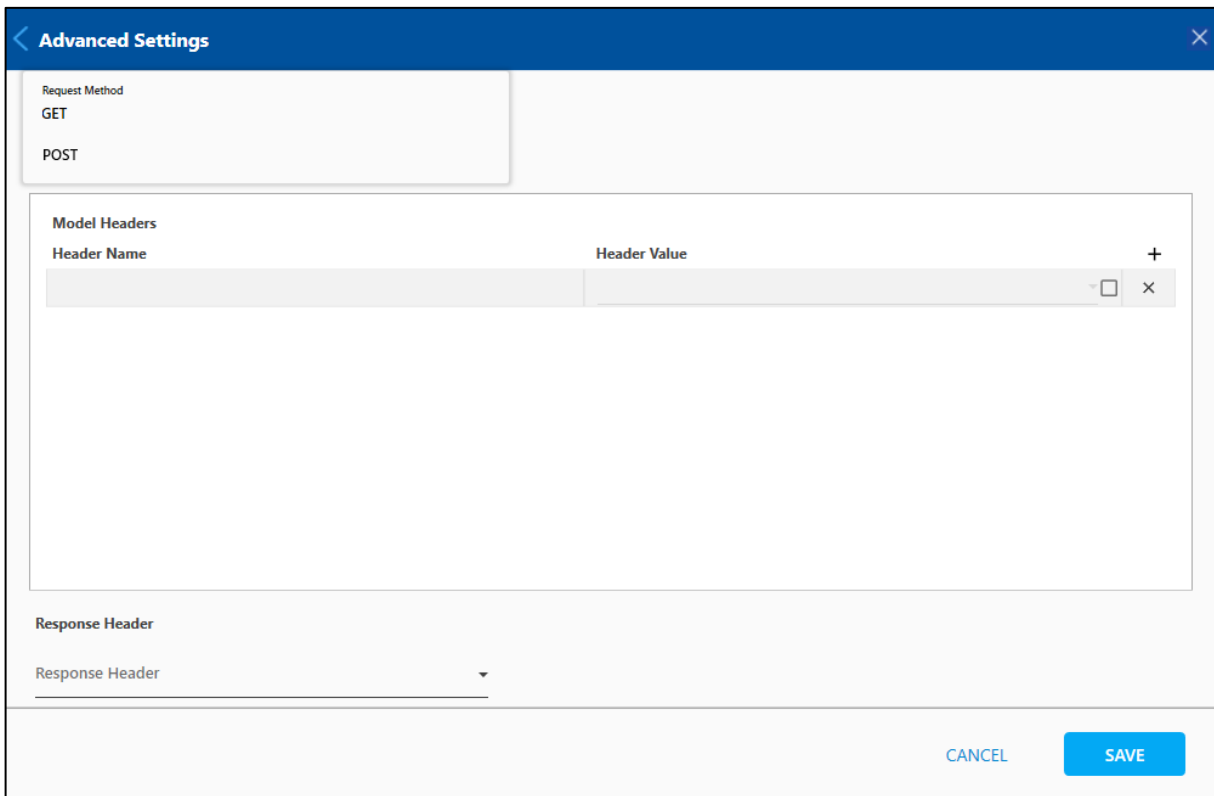
Model Response

Model Response

CANCEL
SAVE

- a. In the **Model Response** list, select the parameter to store the JSON response.
- b. You can use the **X** icon to delete the added input field.

- c. Click **Advanced Settings** to update the request method, request header and to capture the response header if required. The **Advanced Settings** dialog box appears.



- d. In the **Request Method** list, select **POST** if you want to change the request method.
- e. The **Accept** and **Content-Type** header details that are added in **Headers** section while configuring the ML server in the **Admin** menu of Automation Studio auto populates in the **Model Headers** screen, as shown in the below screen shot. This lets you inherit the information related to the ML model REST API request and response. By default, JSON format is supported if the headers are not defined.



- f. You can manually add headers by clicking the **+** (Add) icon as per your requirement.

Advanced Settings

Request Method
GET

Model Headers

Header Name	Header Value		
		<input type="checkbox"/>	X
Content-Type	application/json	<input checked="" type="checkbox"/>	X
Accept	application/json	<input checked="" type="checkbox"/>	X
Content-Length	54138	<input checked="" type="checkbox"/>	X

Response Header
Response Header
HeaderOutput

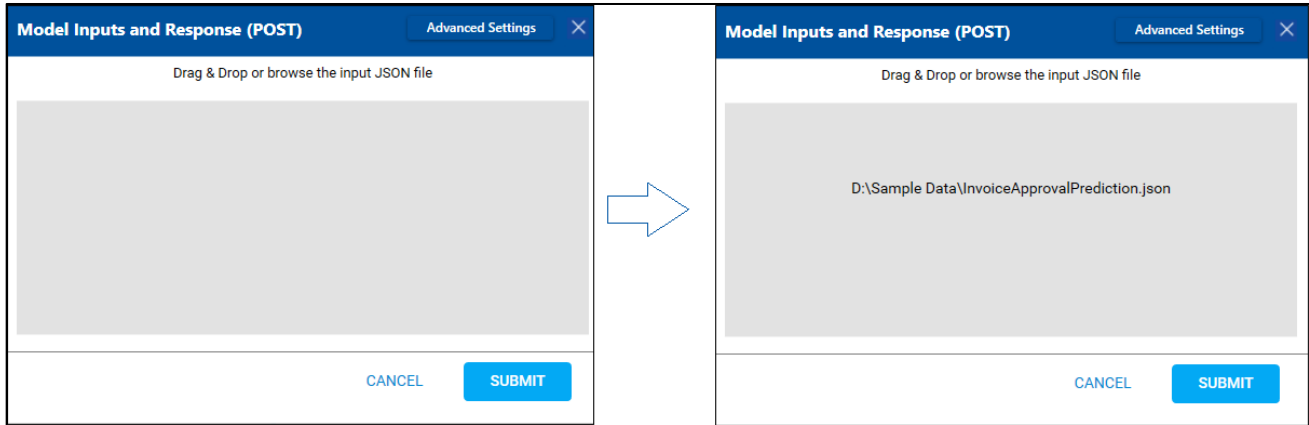
CANCEL SAVE

- g. In the **Header Name** field, enter the name of the header.
- h. In the **Header Value** field, select the relevant parameter holding the value of the specified header. You must predefine the parameter and the respective value in the **Parameter** bar to use this option. Alternatively, select the check box available beside the **Header Value** field and enter the header value that you want to set as the default value.
- i. Repeat step x to step xii to add multiple headers.
- j. You can use the **X** icon to delete the added headers.
- k. In the **Response Header** list, select the parameter to store the header response.
- l. Click **SAVE**.
- m. Close the Model Inputs and Response (GET) dialog box.

The Get method of retrieving the predictions is configured. The JSON activity can then be used to retrieve the required output fields from the ML model response JSON and configure the automation process flow accordingly.

Using Post Method - Decision Model

Send the inputs to the ML model by uploading a JSON file corresponding to the selected ML model.



1. Either drag and drop or browse the required JSON file.
2. Click **SUBMIT**. The **Simplified** view of the list of input fields to be sent to the selected ML model appears.

Model Inputs and Response (POST)
Advanced Settings ✕

InvoiceApprovalPrediction.json ✕

Standard
Simplified

Key Value

Parameter Name	Parameter Value	✓
Invoice Amount	7238	<input checked="" type="checkbox"/>
Vendor Category	Premium	<input checked="" type="checkbox"/>
Quantity Mismatch		<input type="checkbox"/>
Price Mismatch		<input type="checkbox"/>
Damaged Goods Amount	320	<input checked="" type="checkbox"/>
Deliveries in Last 3 Months	39	<input checked="" type="checkbox"/>
Invoice Approval Rate Overall	92	<input checked="" type="checkbox"/>
Invoice Approval Rate_Last3M	94	<input checked="" type="checkbox"/>

Model Response

Model Response

InvoiceApprovalDecision ▼

CANCEL
SAVE

- Click **Parameter Value** field of any of the input to modify the required value. By default, all the input fields are selected. You can clear the check box beside the required **Parameter Value** if you do not want to send the respective input field to the selected ML model. Alternatively, click **Standard** tab to view and modify the inputs using standard JSON editing method.

Model Inputs and Response (POST) Advanced Settings ✕

InvoiceApprovalPrediction.json ✕

Standard Standard Simplified

```
{
  "Invoice Amount": 7238, 
  "Vendor Category": Source File 
  "Quantity Mismatch": "No", 
  "Price Mismatch": "No", 
  "Damaged Goods Amount": 320, 
  "Deliveries in Last 3 Months": 39, 
  "Invoice Approval Rate Overall": 92, 
  "Invoice Approval RateLast3M": 94 
}
```

Model Response
Model Response
 InvoiceApprova/Decision

CANCEL SAVE

- In the **Model Response** list, select the parameter to store the JSON response.
- Click **Advanced Settings** to update the request method, request header and to capture the response header if required. The **Advanced Settings** dialog box appears. The **Input Format** displays the format of sending the input to the ML model. The **URL** field displays the ML model REST API URL.

Advanced Settings ✕

Request Method: POST Input Format: JSON

URL: "http://10.73.123.254:8080/api/rest/process/Invoice_Predict"

Model Headers

Header Name	Header Value	
		+ <input type="checkbox"/> ✕

Response Header
Response Header
 HeaderOutput

CANCEL SAVE

- In the **Request Method** list, select **GET** if you want to change the request method.
- The **Accept** and **Content-Type** header details that are added in **Headers** section while configuring the ML server in the **Admin** menu of Automation Studio auto populates in the **Model Headers** screen, as shown in the below screen shot. This

lets you inherit the information related to the ML model REST API request and response. By default, JSON format is supported if the headers are not defined.

Headers	
Accept	application/json
Content-Type	application/json

8. You can manually add headers by clicking the **+** (Add) icon as per your requirement.

Advanced Settings

Request Method: POST | Input Format: JSON

URL: "http://10.73.123.254:8080/api/rest/process/invoice_Predict"

Header Name	Header Value		
		<input type="checkbox"/>	X
Content-Type	application/json	<input checked="" type="checkbox"/>	X
Accept	application/json	<input checked="" type="checkbox"/>	X
Content-Length	54138	<input checked="" type="checkbox"/>	X

Response Header: HeaderOutput

CANCEL SAVE

- a. In the **Header Name** field, enter the name of the header.
 - b. In the **Header Value** field, select the relevant parameter holding the value of the specified header. You must predefine the parameter and the respective value in the **Parameter** bar to use this option. Alternatively, select the check box available beside the **Header Value** field and enter the header value that you want to set as the default value.
 - c. Repeat step x to step xii to add multiple headers.
 - d. You can use the **X** icon to delete the added headers.
9. In the **Response Header** list, select the parameter to store the header response.
10. Click **SAVE**.
11. Close the Model Inputs and Response (POST) dialog box.

The Post method of retrieving the predictions is configured. The JSON activity can then be used to retrieve the required output fields from the ML model response JSON and configure the automation process flow accordingly.

9.3 Cognitive Services

A powerful feature that lets you extract insight and subjective information from the content helps you with a better customer service and market approach. Ability to mine the data based upon negative or positive attitude of the customer, takes your business to a new height. At enterprise level you need to process images into a more meaningful and actionable data.

Automation Studio supports third party cognitive services, such as, Microsoft, Google, and leverages speech, vision and text to analysis sentiments and process images.

The output of the cognitive services is a JSON. You can use the JSON activity to get a value, such as, score, from the output. Before test run, you need to setup the environment where you can provide the subscription key in a secure manner.

Note: You must have a valid subscription key to run cognitive services. If you need a subscription key, contact your service provider or EdgeVerve marketplace team to obtain the required subscription key.

The activities in the cognitive services tool include:

- [Language](#)
- [Vision](#)
- [Speech](#)
- [Generic Cognitive Services](#)

In Automation Studio, there are few cognitive services or APIs which are already configured and ready to use at their respective activity level such as, language, vision, speech. If required cognitive service is unavailable in Automation Studio, then you can configure the required cognitive service/API in generic cognitive services activity. In this activity, you need to configure the service first and then consume the service for automation.

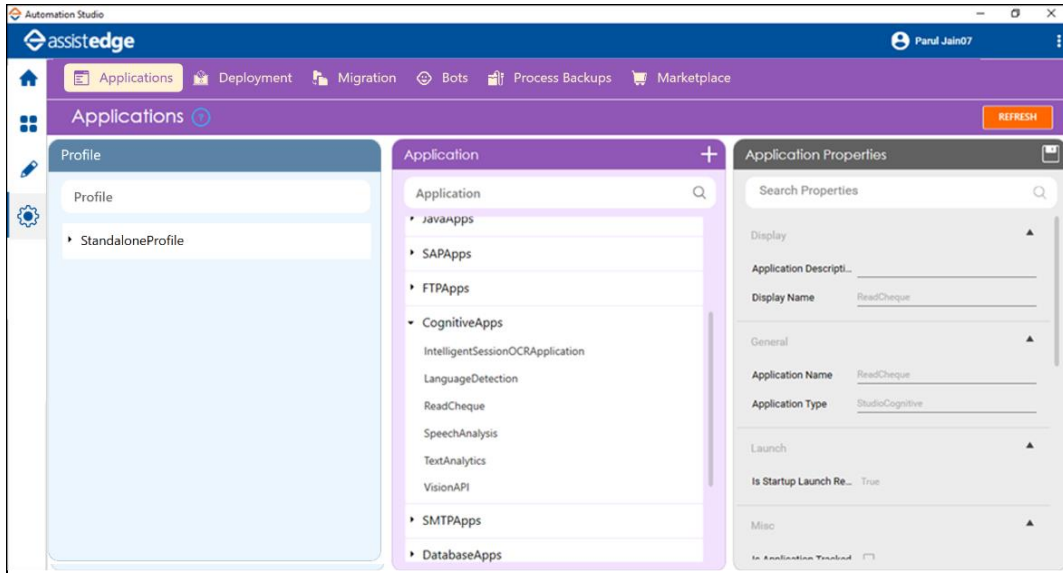
9.3.1 Prerequisites

To use cognitive services activities, you need to configure corresponding application in Automation Studio.

To add application:

Below is the minimum required properties for configuring cognitive application. If you want to define remaining properties, see CognitiveApps in the Admin Capabilities section. Refer the CognitiveApps section to know more about cognitive application properties.

1. In the **Admin** menu, add an application of Application Type- **CognitiveApps**.



2. In **Application Properties**, enter below details -
 - a. Enter Application Name
 - b. Select license type in **License Type** list.

If you have your own license, then select BYOL (Bring Your Own License) or If you have EdgeVerve provided license then select EVOL (EdgeVerve Owned License).

- c. Select login type in **Login Type** list.

If you want to login through subscription key, select Key and if you want to login through username and password, select Credentials.

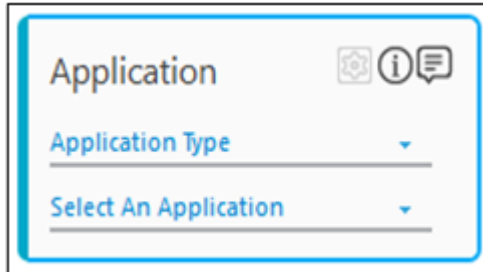
3. Click the  (**Save Properties**) icon to save the properties.

9.3.2 Language

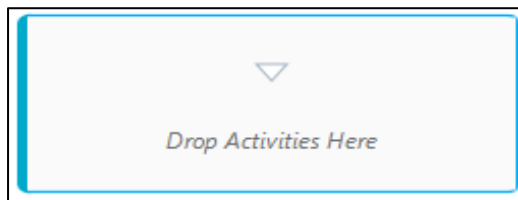
Language activity is used to perform AI based text analysis. It is a pre-trained machine learning model which helps you to perform activities without human intervention, such as, classify text, entity analysis, syntax analysis, sentiment analysis, language detection.

Using Language Activity

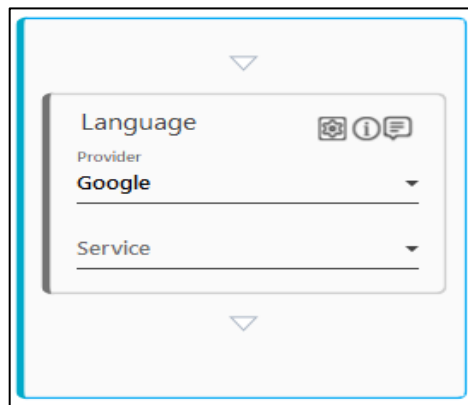
1. In the **Canvas Tools** pane, click **Process Components** to expand and view the associated activities.
2. Drag the **Application** activity and drop on to the **Flowchart** designer area on the **Canvas**.



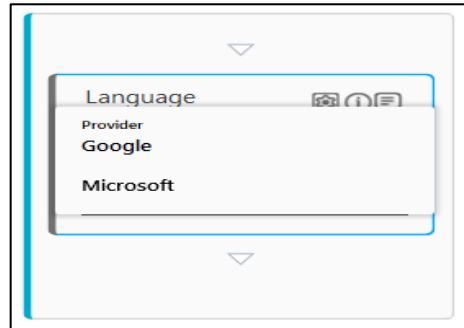
3. In the Application Type list, select CognitiveApps.
4. In the **Select an Application** list, select application.
5. Double-click the **Application** activity to add Language activity as cognitive activities works inside the application activity.



6. In the **Canvas Tools** pane, click **Cognitive Services** to expand the tool and view the associated activities.
7. Drag the **Language** activity and drop on to the **Flowchart** designer area on the **Canvas**.

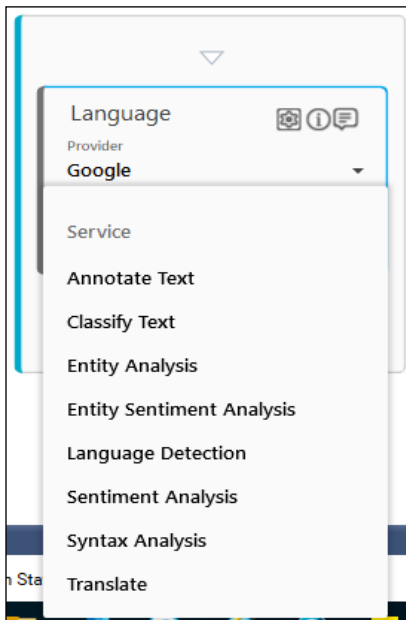


8. In the **Provider** list, select the provider name. By default, provider is set to **Google**.



Automation Studio supports two service providers for cognitive language API.

- Microsoft
 - Google
9. In the **Service** list, select the service which you want to use for cognitive language API.




Based on your selection of provider, service list reflects the supported APIs. If you have selected Google as your provider, following API services will reflect in the service list:

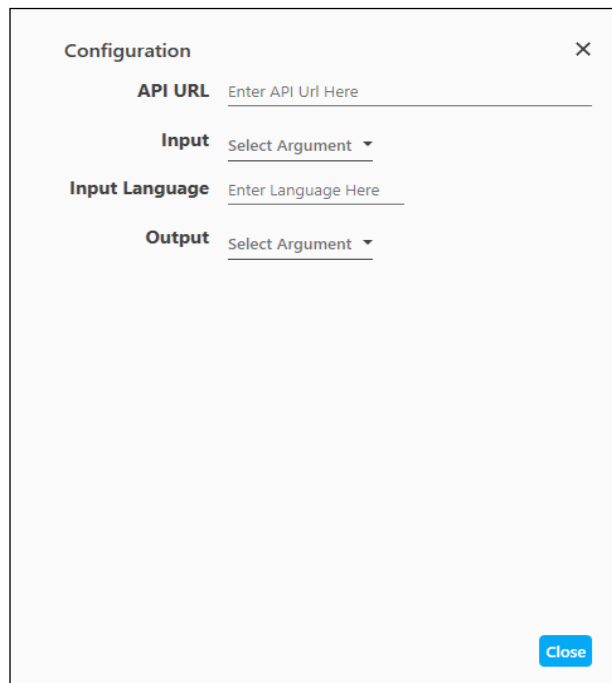
- **Annotate Text:** It is a machine learning capability in which machine learns from previous annotated text and read new input and provide the output with annotated text. In this API, you can get all syntax, sentiment, entity, and classification features in one call.
- **Classify Text:** This API is used to classify the text and assign it to the predefined classified categories.
- **Entity Analysis:** This API is used to find entities, such as, common nouns, proper nouns, in the text and return information about identified entities.

- **Entity Sentiment Analysis:** This API is used to find entities in the text, like entity analysis. Additionally, analyzes sentiments associated with the entity.
- **Language Detection:** This API is used to detect the language of the given text.
- **Sentiment Analysis:** This API is used to analyzes the sentiment of the text, such as, positive feedback, negative feedback.
- **Syntax Analysis:** This API is used to detect the syntax in the text by breaking text into a series of sentences and tokens.
- **Translate:** This API is used to translate the text into the desired language.

If you have selected Microsoft as your provider, following API services will reflect in the service list:

- **Entity Analysis:** This API is used to find entities, such as, common nouns, proper nouns, in the text and return information about identified entities.
- **Key Phrases:** This API is used to detect the key words in the text.
- **Language Detection:** This API is used to detect the language of the given text.
- **Sentiment Analysis:** This API is used to analyzes the sentiment of the text, such as, positive feedback, negative feedback.
- **Translate:** This API is used to translate the text into the desired language.

10. Click the  (**Settings**) icon. The **Configuration** window appears.



- a. **API URL:** Provide API URL based on the provider and service selected by you in the **Vision** activity.

- b. **Input:** This is an input for your API. You can provide file path by creating argument in the argument section and then select the created argument in the **Input** list.
- c. **Input Language:** This is an optional parameter; you can keep it blank for auto detect. You can define text in following format, such as, **en-US**. **en** depicts English and **US** depicts United States. As a whole API detects US English. Appears if **Service** selected is **Entity Analysis, Key Phrases, Sentiment Analysis, Annotate Text, Classify Text, Entity Sentiment Analysis,** and **Syntax Analysis**.
- d. **Target Language:** Specify the target language in which you want the translate the provided text. You can define text in following format, such as, **en-US**. **en** depicts English and **US** depicts United States. As a whole API detects US English. Appears if **Service** selected is **Translate**.
- e. **Output:** Provide the JSON file to store the output. This is an output of your API. You can provide the JSON file for output by creating argument in the **Argument** pane and then select the created argument in the **Output** list.

The above mentioned parameters may vary as per your selection.

11. Click **Close** to confirm your changes.

Language Properties

The properties of Language activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
ApplicationID	It is internally created and managed by Automation Studio itself.
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.

Property Name	Usage
	In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Language . You can change the name as required.
InputLanguage	Specify the text language in this field. The format to define the text language is en-US . Alternatively, you can enter the text language in the Input Language field of the Configuration window. The text language entered in the Properties grid reflects in the Configuration window and vice versa.
InputText	Specify the file path for the input. Alternatively, you can specify the file path by selecting the argument holding file path, in the Input list of the Configuration window. The file path specified in the Properties grid reflects in the Configuration window and vice versa.
ResultJson	Specify the JSON file path to store the JSON output. Alternatively, you can specify the JSON file path by selecting the argument holding file path, in the Output list of the Configuration window. The file path specified in the Properties grid reflects in the Configuration window and vice versa.
SelectedService	The name of the service selected in the language activity box. You can change the name as required.
Url	Specify the API URL in the field. Alternatively, you can specify the API URL by selecting the argument holding the API URL, in the API URL list of the Configuration window. The API URL specified in the Properties grid reflects in the Configuration window and vice versa.

See Also

You can visit the below mentioned links to know about the supported languages:

- Google Natural language API: <https://cloud.google.com/natural-language/docs/languages>
- Google Language Detection & Translation API: <https://cloud.google.com/translate/docs/languages>
- Microsoft Text Analytics API: <https://docs.microsoft.com/en-us/azure/cognitive-services/text-analytics/language-support>
- Microsoft Translation API: <https://docs.microsoft.com/en-in/azure/cognitive-services/translator/language-support>

For more information:

Google:

<https://cloud.google.com/natural-language/docs/reference/rest/v1/documents/analyzeEntities>

<https://cloud.google.com/natural-language/docs/reference/rest/v1/documents/analyzeEntitySentiment>

<https://cloud.google.com/natural-language/docs/reference/rest/v1/documents/analyzeSentiment>

<https://cloud.google.com/natural-language/docs/reference/rest/v1/documents/analyzeSyntax>

<https://cloud.google.com/natural-language/docs/reference/rest/v1/documents/annotateText>

<https://cloud.google.com/natural-language/docs/reference/rest/v1/documents/classifyText>

<https://cloud.google.com/translate/docs/reference/detect>

<https://cloud.google.com/translate/docs/reference/translate>

Microsoft:

<https://westus.dev.cognitive.microsoft.com/docs/services/TextAnalytics.V2.0/operations/5ac4251d5b4ccd1554da7634>

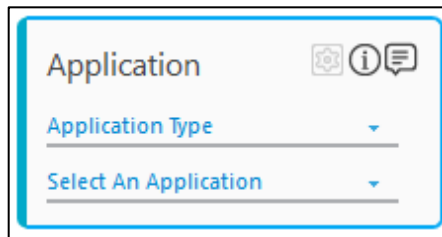
<https://docs.microsoft.com/en-us/azure/cognitive-services/Translator/reference/v3-0-reference>

9.3.3 Vision

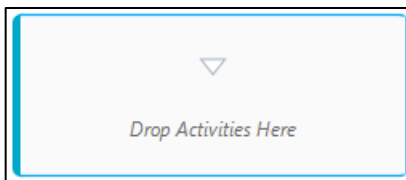
Vision activity is used to perform AI based image analysis. In this activity, you can process the images and retrieve information from images, such as, payee in the cheque, amount in the cheque, with the help of the cognitive vision API.

Using Vision Activity

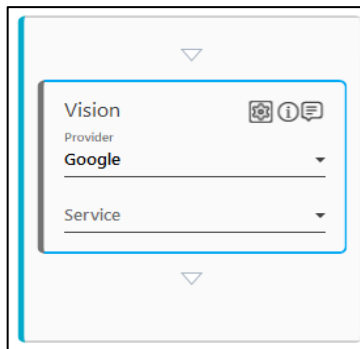
1. In the **Canvas Tools** pane, click **Process Components** to expand and view the associated activities.
2. Drag the **Application** activity and drop on to the **Flowchart** designer area on the **Canvas**.



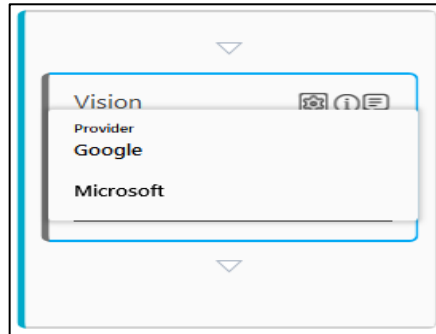
3. In the Application Type list, select CognitiveApps.
4. In the **Select An Application** list, select the application.
5. Double-click the **Application** activity to add Vision activity as cognitive activities works inside the application activity.



6. In the **Canvas Tools** pane, click **Cognitive Services** to expand the tool and view the associated activities.
7. Drag the **Vision** activity and drop on to the **Flowchart** designer area on the **Canvas**.

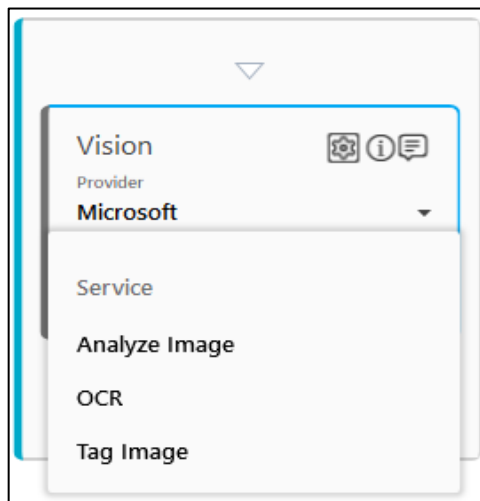


8. In the **Provider** list, select the Provider name. By default, Provider is set to **Google**.



Automation Studio supports two service providers for cognitive vision API.

- Microsoft
 - Google
9. In the **Service** list, select the service which you want to use for cognitive vision API.




Based on your selection of provider, service list reflects the supported APIs. If you have selected Google as your provider, following API services will reflect in the service list:

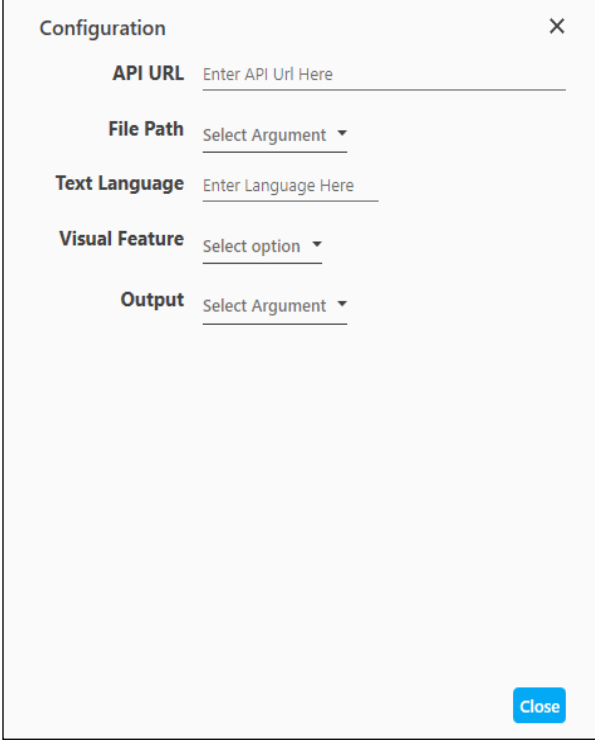
- **Analyze Image:** This API extracts a rich set of visual features based on the image content. You can analyze APIs, such as, categories detection, label detention, face detection, SafeSearch detection, by selecting an appropriate visual feature in the configuration window.

If you have selected Microsoft as your provider, following API services will reflect in the service list:

- **Analyze Image:** This API extracts a rich set of visual features based on the image content. You can analyze APIs, such as, categories detection, label detention, face detection, SafeSearch detection, by selecting an appropriate visual feature in the configuration window.
- **OCR:** Optical Character Recognition (OCR) detects text in an image and extracts the recognized characters into a machine usable character stream.

- **Tag Image:** The API generates a list of words, or tags, that are relevant to the content of the provided image.

10. Click the  (Settings) icon, a **Configuration** window appears.



- API URL:** Provide API URL based on the provider and service selected by you in the Vision activity.
- File Path:** Provide image path for analyzing the image. This is an input for your API. You can provide image path by creating argument in the **Arguments** pane, and then select the created argument in the **File Path** list.
- Text Language:** This is an optional parameter and applicable for OCR and text detection visual features. You can keep it blank for auto detect. You can define text in following format, such as, **en-US**. **en** depicts English and **US** depicts United States. As a whole, OCR detects US English.
- Output:** Provide the JSON file to store the output. This is an output of your API. You can provide the JSON file for output by creating argument in the **Argument** pane and then select the created argument in the **Output** list.
- Visual Feature:** Select the visual feature from the list, such as, categories, Adult, Color. This list contains the visual features API list. Appears if **Service** selected is **Analyze Image**.

The above mentioned parameters may vary as per your selection.

11. Click **Close** to confirm your changes.

Vision Properties

The properties of Vision activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
ApplicationID	It is internally created and managed by Automation Studio itself.
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Vision . You can change the name as required.
FilePath	Specify the file path for the input. Alternatively, you can specify the file path by selecting the argument holding file path, in the Input list of the Configuration window. The file path specified in the Properties grid reflects in the Configuration window and vice versa.
ResultJson	Specify the JSON file path to store the JSON output. Alternatively, you can specify the JSON file path by selecting the argument holding file path, in the Output list of the Configuration window. The file path specified in the Properties grid reflects in the Configuration window and vice versa.
SelectedService	The name of the service selected in the language activity box. You can change the name as required.
TextLanguage	Specify the text language in this field. The format to define the text language is en-US. Alternatively, you can enter the text language in the Text Language field of the Configuration window. The text language entered in the Properties grid reflects in the Configuration window and vice versa.

Property Name	Usage
Url	Specify the API URL in the field. Alternatively, you can specify the API URL by selecting the argument holding the API URL, in the API URL list of the Configuration window. The API URL specified in the Properties grid reflects in the Configuration window and vice versa.
VisualFeature	The name of the visual feature selected in the Visual Feature list of configuration window is displayed.

See Also

Refer the below links to learn more about the supported languages in vision activity:

Google: <https://cloud.google.com/vision/docs/reference/rest/v1/images/annotate>

Google Vision API: <https://cloud.google.com/vision/docs/languages>

Microsoft Analyze Image API:

<https://westus.dev.cognitive.microsoft.com/docs/services/56f91f2d778daf23d8ec6739/operations/56f91f2e778daf14a499e1fa>

Microsoft OCR API:

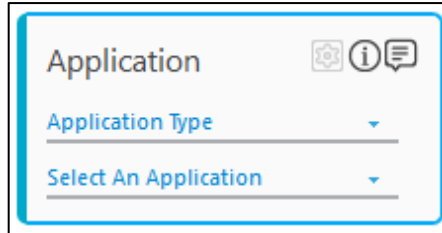
<https://westus.dev.cognitive.microsoft.com/docs/services/56f91f2d778daf23d8ec6739/operations/56f91f2e778daf14a499e1fc>

9.3.4 Speech

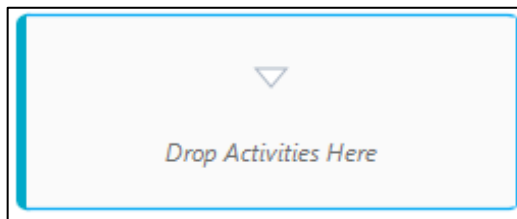
Speech activity is used to perform AI based audio analysis. In this activity, you can perform speech to text, text to speech and speaker verification.

Using Speech Activity

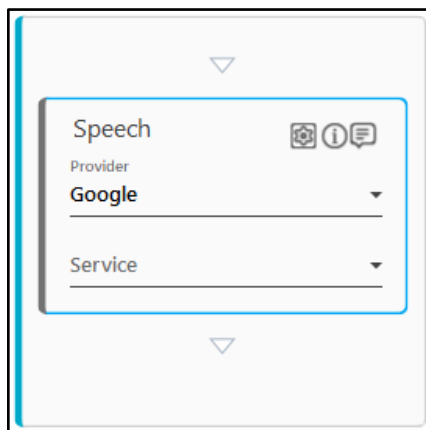
1. In the **Canvas Tools** pane, click **Process Components** to expand and view the associated activities.
2. Drag the **Application** activity and drop on to the **Flowchart** designer area on the **Canvas**.



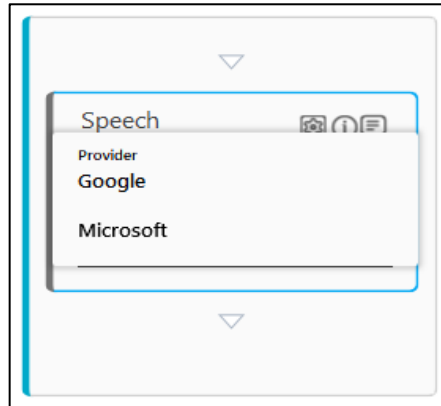
3. In the Application Type list, select CognitiveApps.
4. In the **Select An Application** list, select application.
5. Double-click the **Application** activity to add Speech activity as cognitive activities works inside the application activity.



6. In the **Canvas Tools** pane, click **Cognitive Services** to expand the tool and view the associated activities.
7. Drag the **Speech** activity and drop on to the **Flowchart** designer area on the **Canvas**.

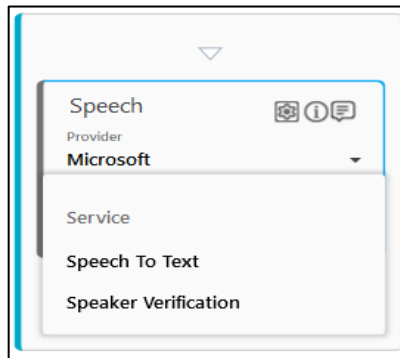


8. In the **Provider** list, select the provider name. By default, provider is set to **Google**.



Automation Studio supports two service providers for cognitive speech API.

- Microsoft
 - Google
9. In the **Service** list, select the service which you want to use for cognitive speech API.




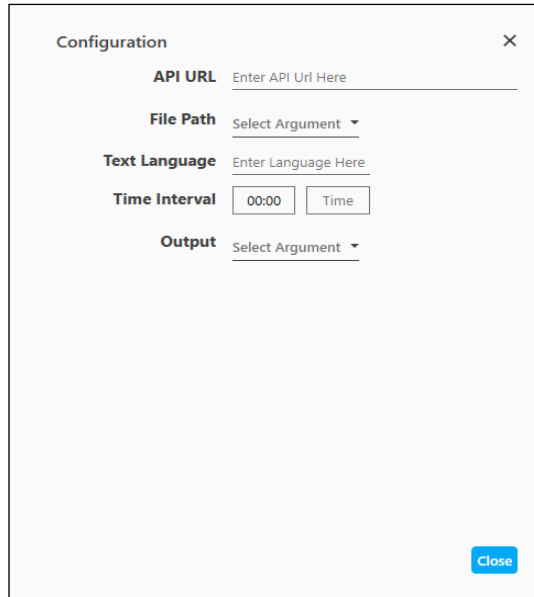
Based on your selection of provider, service list reflects the supported APIs. If you have selected Google as your provider, following API services will reflect in the service list:

- **Speech To Text:** This API is used to convert audio to text.
- **Text To Speech:** This API is used to convert text to audio.

If you have selected Microsoft as your provider, following API services will reflect in the service list:

- **Speech To Text:** This API is used to convert audio to text.
- **Speaker Verification:** This API is used to verify an audio input with pre-enrolled profile. It lets you match the voice with a pre-recorded one. Pre-enrolled profile is the pre-recorded voice of an enrolled speaker.

10. Click the  (**Settings**) icon, the **Configuration** window appears.



- a. **API URL:** Provide API URL based on the provider and service selected by you in the Vision activity.
- b. **File Path:** Provide file path for analyzing speech/text. This is an input for your API. You can provide file path by creating argument in the **Argument** pane, and then select the created argument in the **File Path** list.
- c. **Text Language:** This is an optional parameter and applicable for OCR and text detection visual features. You can keep it blank for auto detect. You can define text in following format, such as, **en-US**. **en** depicts English and **US** depicts United States. As a whole, OCR detects US English.
- d. **Output:** Provide the JSON file to store the output. This is an output of your API. You can provide the JSON file for output by creating argument in the **Argument** pane and then select the created argument in the **Output** list.
- e. **Time Interval:** This field is used to trim the long audio into short audio clip for analyzing. The format to define time is MM:SS (Minutes:Seconds). Appears if **Service** selected is **Speech to Text**, and **Speaker Verification**.
- f. **Verification Profile Id:** This is an optional field, applicable for **Speaker Verification** API. Provide pre-enrolled profile ID for **Speaker Verification** API. Appears if **Service** selected is **Speaker Verification**.
- g. **Audio Output Path:** Provide the audio output path to store the output which is in audio format. You can provide the file for output by creating argument in the **Argument** pane and then select the created argument in the **Output** list. The string format for output argument is defined in **D:\audio\sample.aw** format.

The above mentioned parameters may vary as per your selection.

11. Click **Close** to confirm your changes.

Speech Properties

The properties of Speech activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
ApplicationID	It is internally created and managed by Automation Studio itself.
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Speech . You can change the name as required.
EndTime	Specify the end time for the audio clip. The format to specify the end time is MM:SS.
FilePath	Specify the file path for the input. Provided file path is displayed in the configuration window Input parameter and vice versa.
InputText	Specify the file path for the input. Provided file path is displayed in the configuration window Input Text parameter and vice versa.
ResultAudioPath	Specify the audio path to store the audio output. Provided audio path is displayed in the configuration window Audio Output Path parameter and vice versa.
ResultJson	Specify the JSON file path to store the JSON output. Provided JSON file path is displayed in the configuration window Output parameter and vice versa.

Property Name	Usage
SelectedService	The name of the service selected in the Speech activity box. You can change the name as required.
StartTime	Specify the start time for the audio clip. The format to specify the start time is MM:SS.
TextLanguages	Specify the text language in this field. The format to define the text language is en-US. Alternatively, you can enter the text language in the Text Language field of the Configuration window. The text language entered in the Properties grid reflects in the Configuration window and vice versa. .
Url	Specify the API URL in the field. Alternatively, you can specify the API URL by selecting the argument holding the API URL, in the API URL list of the Configuration window. The API URL specified in the Properties grid reflects in the Configuration window and vice versa.
VerificationProfileId	Specify the pre-enrolled verification ID for speaker verification. The verification profile Id is displayed in the Verification Profile Id parameter of the Configuration window.

See Also

Languages supported:

Google Text To Speech API: <https://cloud.google.com/text-to-speech/docs/voices>

Google Speech To Text API: <https://cloud.google.com/speech-to-text/docs/languages>

Microsoft Speaker Verification API:

<https://westus.dev.cognitive.microsoft.com/docs/services/563309b6778daf02acc0a508/operations/563309b7778daf06340c9652>

Microsoft Speech To Text API:

<https://docs.microsoft.com/en-in/azure/cognitive-services/speech-service/language-support>

For more information:

Google: <https://cloud.google.com/text-to-speech/docs/reference/rest/v1beta1/text/synthesize>

<https://cloud.google.com/speech-to-text/docs/reference/rest/v1/speech/recognize>

<https://cloud.google.com/speech-to-text/docs/languages>

Microsoft:

<https://westus.dev.cognitive.microsoft.com/docs/services/563309b6778daf02acc0a508/operations/56406930e597ed20c8d8549c>

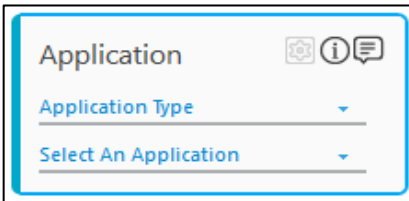
9.3.5 Generic Cognitive Service

Generic Cognitive Services activity is used to consume cognitive REST APIs of different providers. If there are any newly launched APIs in the market and it is not available in our product, then you can use this activity to consume the API.

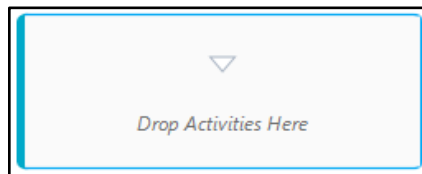
Note: To use Generic Cognitive Service, service API details are required.

Using Generic Cognitive Service

1. In the **Canvas Tools** pane, click **Process Components** to expand and view the associated activities.
2. Drag the **Application** activity and drop on to the **Flowchart** designer area on the **Canvas**.

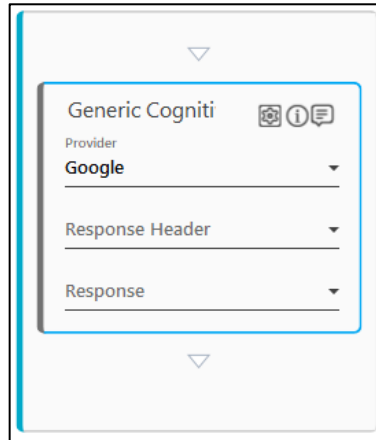


3. In the Application Type list, select CognitiveApps.
4. In the **Select An Application** list, select application.
5. Double-click the **Application** activity to add Vision activity as cognitive activities works inside the application activity.

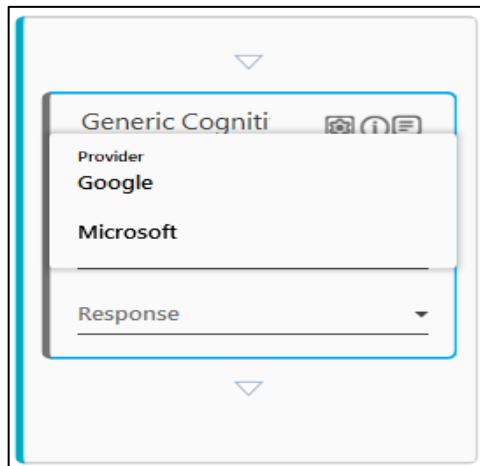


6. In the **Canvas Tools** pane, click **Cognitive Services** to expand the tool and view the associated activities.

7. Drag the **Generic Cognitive Service** activity and drop on to the **Flowchart** designer area on the **Canvas**.




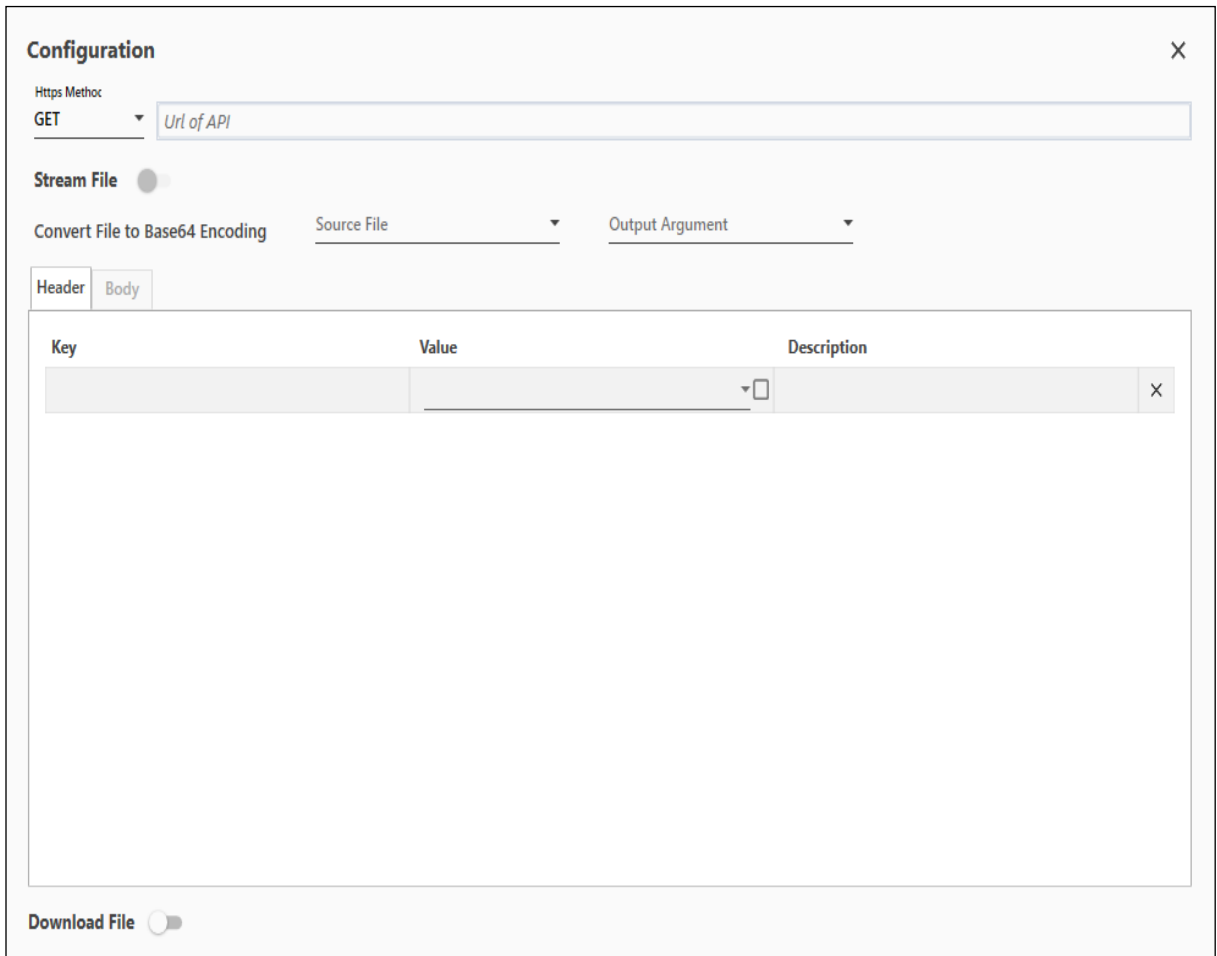
8. In the **Provider** list, select the Provider name. By default, the provider is set to **Google**.



Automation Studio supports two service providers for Generic Cognitive Service.

- Microsoft
 - Google
9. In the **Response Header** list, select the response header file for the API call. This file provides JSON for all the headers in the response.
10. In the **Response** list, select the response file for the API call. This file provides body of the response.

11. Click the  (**Settings**) icon, the **Configuration** window appears.



Configuration [X]

Https Method
GET [Url of API]

Stream File

Convert File to Base64 Encoding Source File [] Output Argument []

Header | Body

Key	Value	Description


Download File

To configure API service:

- a. **Https Method:** The request made by the API for retrieving or sending data to the server. By default, **Get** is selected.
 - i. **GET** - Retrieves data from the server.
 - ii. **POST** - Sends data to the server for creating new resource.
 - iii. **PUT** - Updates an existing resource.
 - iv. **DELETE** - Deletes data or resource from the server.
- b. **Url of API:** An API is accessed via URL where you can form the endpoint of the API URL by passing the resource as a query parameter or by simply stating the actual value of the endpoint. It is an expression textbox where a VB scripts can be entered such as, **baseurl+ "? language=en-US&format=detailed"** where **baseurl** is an argument concatenated with string literal.
- c. **Stream File:** Enables to stream the API response as a binary stream. This option is available only for **Https Methods- POST, PUT and DELETE**. If it is enabled, the **Body** tab of the **Configuration** dialog box, remains disabled. This is an optional field which is used with endpoints like Microsoft's Speech to Text.
 - i. **Stream Source File:** This field is available only when **Stream File** is enabled. Select the parameter that contains the

path of the streaming file. You must define the parameter of string type, in the **Parameter** bar to use this option.

- d. **Convert File to Base64 Encoding:** This field is available only when **Stream File** is disabled and is an optional field. It helps you to encode the file required by the endpoint, to the Base64 format and then add the file as JSON value in the request body.
 - i. **Source File:** Select the parameter that contains the path of the file that needs to be converted to the Base64 encoding. You must define the parameter of string type, in the **Parameter** bar to use this option.
 - ii. **Output Argument:** Select the parameter to store the Base64 encoded string of source file. You must define the parameter of string type, in the **Parameter** bar to use this option.
- e. **Header:** Signifies the information related to the API request and response.
 - i. **Key** - Enter the name of the header. Once the name is entered, a new line item is available.
 - ii. **Value** - In the **Value** list, select the parameter containing the value of the header. You must define a parameter in the **Parameter** bar to use this option. Alternatively, select the check box to enter the value of the header to set it as the default value.
 - iii. **Description** - Enter a small description of the header. This field is only for informative purpose and is not used in any form while consuming an endpoint.
- f. **Body:** Signifies the custom view that contains information that you want to send to the server. This field is disabled for **GET** Https method. This field is disabled if Stream Files is enabled for any of the Https methods.
 - In the **Body Type** list, select the type of response to be sent to the server. Available options are:
 - **JSON** - The request is sent in JSON format.
 - **Others** - The request is sent in any other format such as xml, ssml and others.
 - If **Body Type** selected is **JSON**.

1. In the **Raw Input JSON** field, enter the sample or the template JSON request.
2. Click the  (Click to convert JSON) icon to transform the sample JSON to formatted JSON. Additionally, the validation of the JSON is done once you transform the sample data.
3. In the **Transformed JSON** field, the transformed and validated JSON is available for parameterizing the request

body as shown in the sample Argument pane below.

Following is the sample of the Raw JSON formatted without parameters:

```
Raw
{
  "encodingType": "UTF8",
  "document": {
    "type": "PLAIN_TEXT",
    "language": "fr",
    "content": "je t'aime"
  }
}

Transformed JSON
{
  "encodingType": "UTF8", 
  "document": { 
    "type": "PLAINTEXT", 
    "language": "fr", 
    "content": "je t'aime" 
  }
}
```

Following is the sample Argument pane:

language_code	In	String	"fr"
text	In	String	"je t'aime"

Following is the sample of the Raw JSON formatted with parameters:

```
Raw
{
  "encodingType": "UTF8",
  "document": {
    "type": "PLAIN_TEXT",
    "language": "fr",
    "content": "je t'aime"
  }
}

Transformed JSON
{
  "encodingType": "UTF8", 
  "document": { 
    "type": "PLAINTEXT", 
    "language": language_code 
    "content": text 
  }
}
```

- If **Body Type** selected is **Others**, in the **Raw** field, enter the request body in any other format to send to the server.

Header	Body
	<p>Body Type</p> <p>Body Type Other ▼</p> <p>Raw</p>

- g. **Download File:** Enable the **Download File** option when response of the endpoint is a binary data file. The **Download Filepath** field appears.
- In the **Download Filepath** list, select the parameter containing the file path along with the file name to store the binary data file. You must define a parameter in the **Parameter** bar to use this option.

Generic Cognitive Service Properties

The properties of Generic Cognitive Service activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.

Property Name	Usage
Misc	
ApplicationID	It is internally created and managed by Automation Studio itself.
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Generic Cognitive Service . You can change the name as required.

9.4 Assign and Writeline

The **Assign and Writeline** tool allows you to configure assign and write activities within a process. The activities in the tool include:

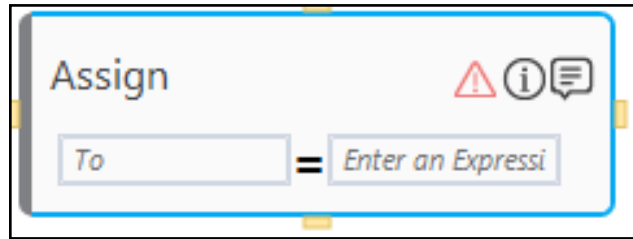
- [Assign](#)
- [WriteLine](#)
- [Write to Log file](#)
- [Input Dialog](#)
- [MessageBox](#)
- [Add Log Fields](#)
- [Remove Log Fields](#)

9.4.1 Assign

This activity assigns a value to a variable or an argument. You can use an **Assign** activity for logical operation such as a loop counter, to increment the value, hold a value, assign a value to other variable or assign values to an array.

Using Assign Activity

1. In the **Canvas Tools** pane, click **Assign and Write** to expand the tool and view the associated activities.
2. Drag the Assign activity and drop on to the **Flowchart designer on the Canvas**



3. In the **To** field, enter a name of parameter to which the value should be assigned. You can press **Ctrl+Space** key together from the keyboard to use IntelliSense to select the parameter. The name of the variable must be defined in the **Variable** or **Argument** panel in the Parameter bar or the system displays a compiler error.
4. In the **Value** field, enter the value to assign to the parameter. The value can be constant (default value), dynamic (a parameter) or a combination of both.

An **Assign** activity with a default display name is created. The warning sign disappears once you provide the required inputs.

Assign Properties

The properties of an **Assign** activity are listed in the following table and can be edited in the **Properties** grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes, the application ignores any error while executing the activity. If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No.
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.

Property Name	Usage
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Assign . You can change the name as required.
To	Name of the variable or the argument that the activity assigns its value to.
Value	A value that is assigned to the variable or argument.

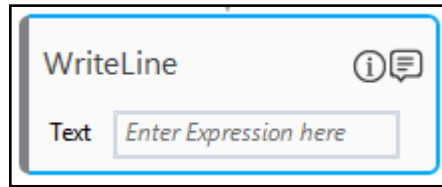
9.4.2 Writeline

This activity allows you to write text in the **Output** console. It prints one or more value on a single line with a new line character inserted at the end. This means, any subsequent output will be printed on a new line.

Note: It is a utility for testing and debugging the automation process in Automation Studio. It does not impact the execution using bots.

Using WriteLine Activity

1. In the **Canvas Tools** pane, click **Assign and Write** to expand the tool and view the associated activities.
2. Drag the WriteLine activity and drop on to the **Flowchart designer on the Canvas**



3. In the **Text** field, enter the text to be written in the **Output** panel. The text entered should be a string or a string variable.

A **WriteLine** activity with a default display name is created.

WriteLine Properties

The properties of **WriteLine** activity is listed in the following table and can be edited in the **Properties** grid on the right pane.

Property Name	Usage
Misc	
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as WriteLine . You can change the name as required.
Text	The text to be written in the Output panel. The text entered should be a string or a string variable. If the text entered is a string, it has to be placed within quotes.

	You can use the ToString method to convert the non-string parameter to a string that is suitable to display.
TextWriter	Displays text in an output window other than the Output panel.

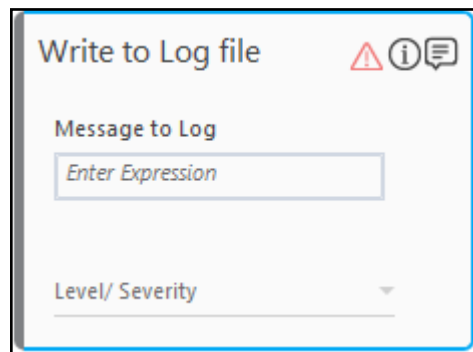
9.4.3 Write to Log files

This activity allows you append additional information in log files and categorize the appended data as debug message, error, warning or just an information. A log file stores events, messages and other communications logged during the execution.

Instead of updating the log file manually, you can use the **Write to Log file** activity to write the additional execution data automatically in to the log file. It ensures that unwanted data is not written in the log file. Additionally, you can avoid any human error while making changes into the log file.

Using Write to Log file Activity

1. In the **Canvas Tools** pane, click **Assign and Write** to expand the tool and view the associated activities.
1. Drag the Write to Log file activity and drop on to the **Flowchart designer on the Canvas**



2. In the **Message to Log** field, enter the message in double quotes to update in the application log file.
3. From the **Level/ Severity** drop-down, select the required level. The level or severity defines the category and purpose of the message written in the log file. Available options are:
 - Debug
 - Information
 - Warning
 - Error

A **Write to Log file** activity is created. The warning sign disappears once you provide the required inputs.

Write to Log file Properties

The properties of **Write to Log file** activity are listed in the following table and can be edited in the **Properties** grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Write to Log . You can change the name as required.

9.4.4 Add Log Fields


This activity allows you to extend existing records in your application log files by adding your own custom fields. The custom log fields and the associated value, are added as a statement in the log files. You can update user defined log events by adding arguments against custom log fields.

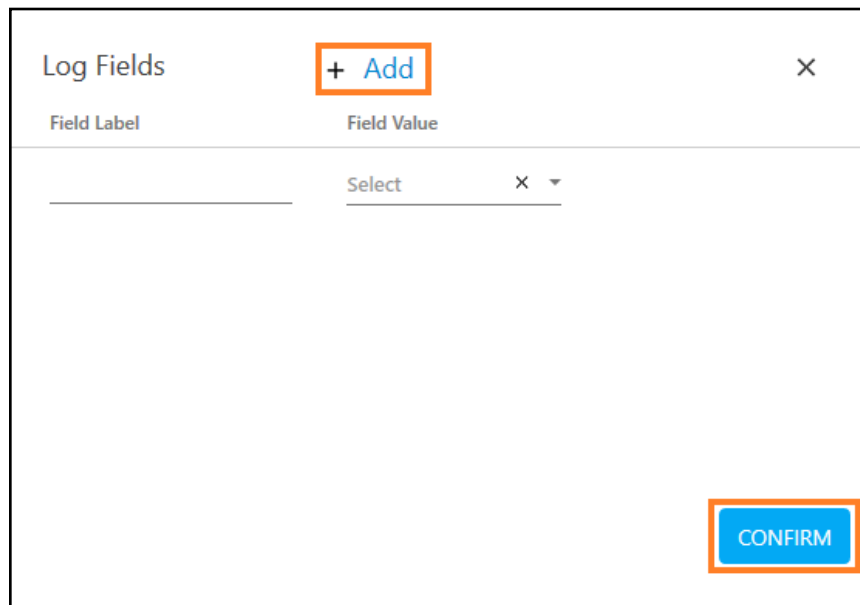
Note: All user defined log fields and their values remain in process scope until they are removed using the 'Remove Log Fields' activity.

Using Add Log Fields Activity

1. In the **Canvas Tools** pane, click **Assign and Write** to expand the tool and view the associated activities.
2. Drag the **Add Log Fields** activity and drop on to the **Flowchart designer on the Canvas**. The validation error symbol disappears when required inputs are provided in the correct format.



3. In the Parameter bar, define an argument that you want to add as the field value to the log file. When you add a field, the argument is selected as the field value.
4. Click the  (**Settings**) icon to configure the log fields and define the field values.



The **Log Fields** dialog box appears.

5. Click **Add** to add a new field.
 - a. In the **Field Label**, enter the desired label of the field.
 - b. In the **Field Value**, select an argument that you created in the **Parameter** bar.
6. Click **Confirm** to save the details entered.

The **Add Log Fields** activity is created. The activity keeps adding an argument to every subsequent log line in the application log file after each process execution.

Add Log Fields Properties

The properties of **Add Log Fields** activity are listed in the following table and can be edited in the **Property** grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity.</p> <p>If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Add Log Fields . You can change the name as required.

9.4.5 Remove Log Fields

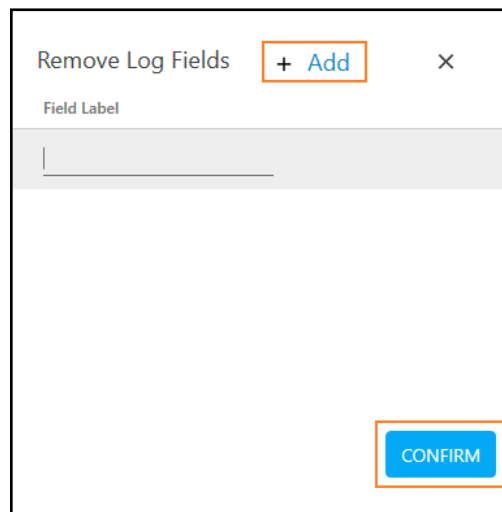
This activity allows you to remove custom fields that are added into the application log files.

Using Remove Log Fields Activity:

1. In the **Canvas Tools** pane, click **Assign and Write** to expand the tool and view the associated activities.
2. Drag Remove Log Fields activity and drop on to the **Flowchart designer on the Canvas**



3. Click the  (**Settings**) icon. The **Remove Log Fields** dialog box appears.



4. Click **Add**.
5. In the **Field** Label, enter the field name you want to remove from the log file.
6. Click **Confirm** to remove the field from the log file.

The **Remove Log Fields** activity is created.

Remove Log Fields Properties

The properties of **Remove Log Fields** activity are listed in the following table and can be edited in the **Property** grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Remove Log Fields . You can change the name as required.

9.5 Flow Controls

Flow Controls tool provides several business validation activities. You can use these activities to control the flow of the process. The activities in the tool includes:

- [If](#)
- [Parallel](#)
- [DoWhile](#)
- [Sequence](#)
- [While](#)

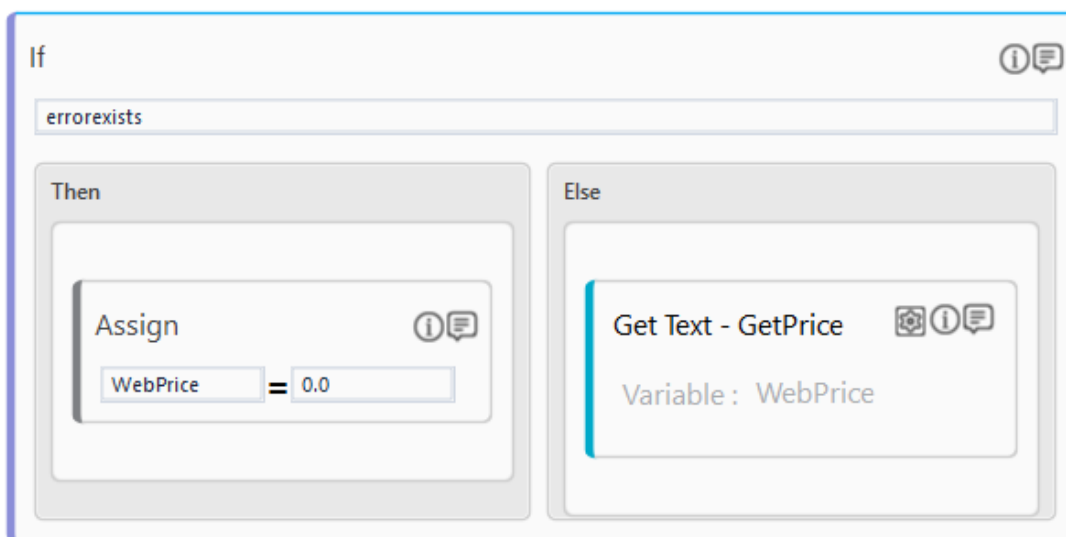
- [Decision Node](#)
- [Delay](#)

9.5.1 If

This activity is used to evaluate a condition and execute a process flow accordingly. It is a conditional activity where you can define a condition and based on the evaluation of the condition further actions are performed. If condition evaluates to true, then If actions are performed and if condition evaluates to false, Else actions are performed.

Using If Activity

1. In the **Canvas Tools** pane, click **Flow Controls** to expand the tool and view the associated activities.
2. Drag the **If** activity and drop on to the **Flowchart designer on the Canvas**



3. In the **Enter Expression here** field, enter the if condition which you want to execute.
4. In the **Then** and **Else** block, place the activity that must be performed based on the **If** condition.

An If activity with a default display name is created. The warning sign disappears once you provide the required inputs.

If Properties

The properties of an If activity is listed in the following table and can be edited in the **Properties** grid on the right pane.

Property Name	Usage
Misc	
Condition	This property is used to determine the flow of the process. You can define the condition in this section.
DisplayName	The name of the activity that is displayed in the flowchart. By default, the value is set as If . You can edit this value as per your need.

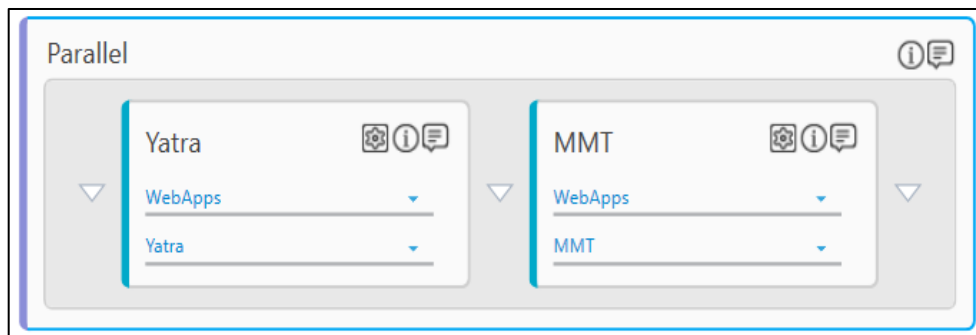
9.5.2 Parallel

This activity can process more than one workflow in parallel. You can perform one child activity with any number of workflows in a single process. For example, you want to check the cheapest flights for your travel. For running this process manually, you will login to 3 different applications and provide the travel details at three different places and then compare the results.

In Parallel activity, you can provide the travel details once and parallel activity runs this information on three different portals (for example, Yatra, MakeMyTrip and Cleartrip) for you and provide the cheapest flight details as per the defined process.

Using Parallel Activity

1. In the **Canvas Tools** pane, click **Flow Controls** to expand the tool and view the associated activities.
2. Drag the **Parallel** activity and drop on to the Flowchart designer on the Canvas



3. Add the activities that you want to run parallel in the **Parallel** activity.

The Parallel activity with a default display name is created. The warning sign disappears once you provide the required inputs.

Parallel Activity Properties

The properties of a **Parallel** activity are listed in the following table and can be edited in the **Properties** grid on the right pane.

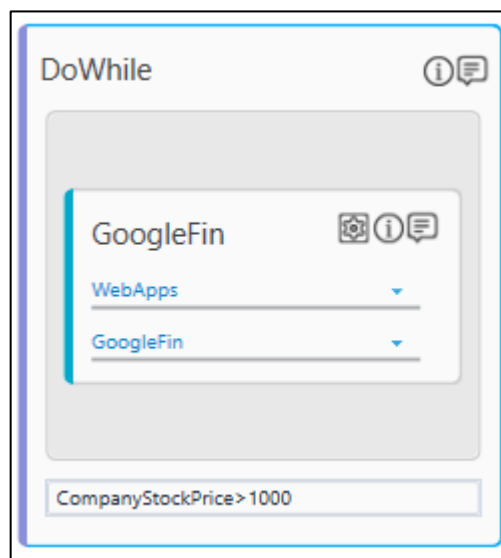
Property Name	Usage
Misc	
CompletionCondition	The condition to evaluate completion of the task of the child activity. If the CompletionCondition is true , the remaining parallel workflows are aborted. This is an optional field. By default, value is set as Null.
DisplayName	The name of the activity that is displayed in the flowchart. By default, the value is set as Parallel . You can edit this value as per your need.

9.5.3 DoWhile

This activity is used to execute an action in loop until the specified condition is met. In this activity, Boolean condition is defined after an execution of code. DoWhile activity works like an exit-condition loop i.e. block of code execution is terminated when the condition evaluates to false.

Using DoWhile Activity

1. In the **Canvas Tools** pane, click **Flow Controls** to expand the tool and view the associated activities.
2. Drag the DoWhile activity and drop on to the **Flowchart designer on the Canvas**



In the **DoWhile** block, place the activity that must be performed based on the do while condition.

3. In the **Enter Expression here** field, enter the do while condition you want to execute. For example,

CompanyStockPrice > 1000, that implies to run the loop until the stock price of the company is greater than 1000.

The DoWhile activity with a default display name is created. The warning sign disappears once you provide the required inputs.

DoWhile Properties

The properties of an **DoWhile** activity are listed in the following table and can be edited in the **Properties** grid on the right pane.

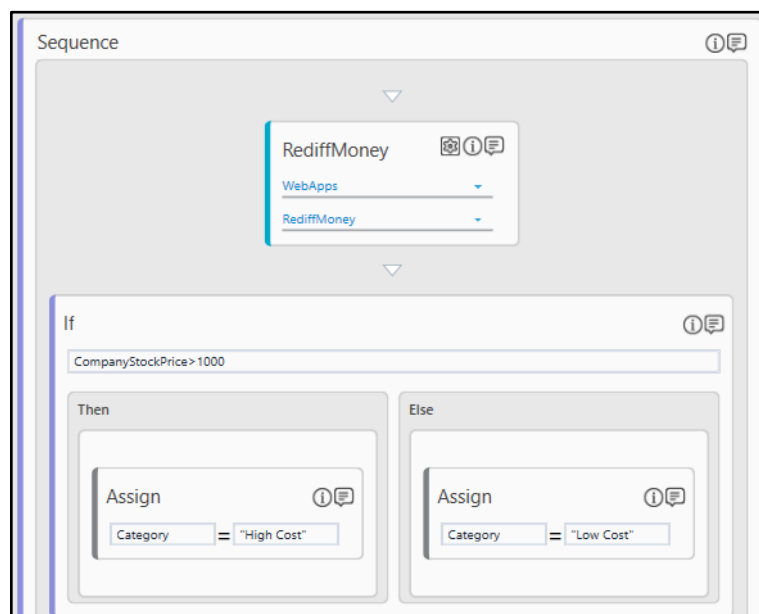
Property Name	Usage
Misc	
Condition	Define a Boolean condition in this field for a loop or block of code.
DisplayName	The name of the activity that is displayed in the flowchart. By default, the value is set as DoWhile . You can edit this value as per your need.

9.5.4 Sequence

This activity is used to execute the set of activities as per the defined steps.

Using DoWhile Activity

1. In the Canvas Tools pane, click Flow Controls to expand the tool and view the associated activities.
2. Drag the **Sequence** activity and drop on to the Flowchart designer on the Canvas



3. In the **Sequence** block, add the activities you want execute.

The Sequence activity with a default display name is created. The warning sign disappears once you provide the required inputs.

Sequence Properties

The properties of a Sequence activity are listed in the following table and can be edited in the Properties grid on the right pane.

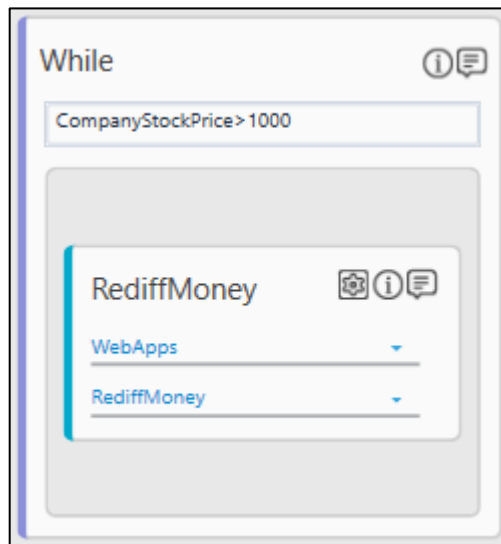
Property Name	Usage
Misc	
DisplayName	The name of the activity that is displayed in the flowchart. By default, the value is set as Sequence . You can edit this value as per your need.

9.5.5 While

This activity works like a repetitive If activity. If the defined Boolean condition evaluates to true then task is performed, else it will terminate the task. This activity is similar to DoWhile, but in DoWhile task is performed before the defined Boolean condition.

Using While Activity

1. In the Canvas Toolspane, click Flow Controls to expand the tool and view the associated activities.
2. Drag the **While** activity and drop on to the Flowchart designer on the Canvas



3. In the **Enter Expression here** field, enter the while condition which you want to execute.
4. In the **While** block, place the activity that must be performed based on the while condition.

The While activity with a default display name is created. The warning sign disappears once you provide the required inputs.

While Properties

The properties of a While activity are listed in the following table and can be edited in the Properties grid on the right pane.

Properties	Usage
Misc	
Condition	Define a Boolean condition in this field for a loop to perform the required task.
DisplayName	The name of the activity that is displayed in the flowchart. By default, the value is set as While . You can edit this value as per your need.

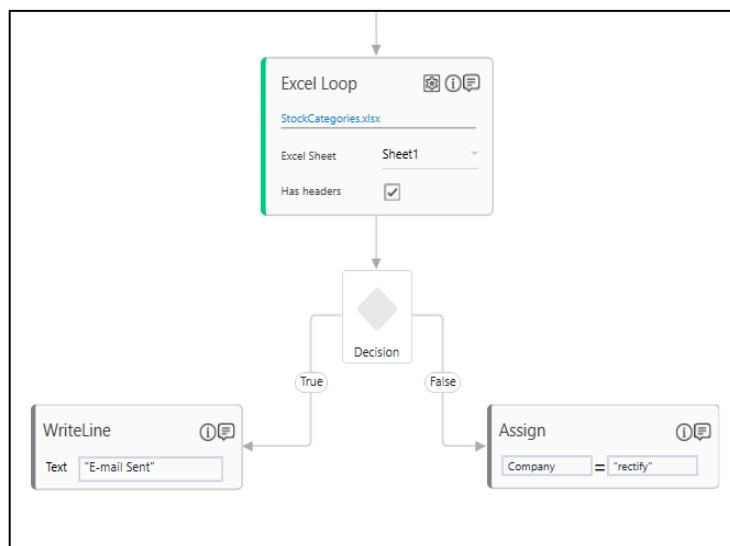
9.5.6 Decision Node

This activity is used to configure a business rule or validation by adding decisions to the flowcharts. It is a conditional activity where you can define a condition and based on the condition evaluation further actions are performed. If condition evaluates to true, then true actions are performed and if condition evaluates to false, then false actions are performed.

Note: The Decision Node activity can only be added at the Flowchart designer level and cannot be dragged and dropped within another activity.

Using Decision Node Activity

1. In the Canvas Toolspane, click Flow Controls to expand the tool and view the associated activities.
2. Drag the **Decision** activity and drop between other activities present in the process workflow to set up a validation.



The Decision activity with a default display name is created. The warning sign disappears once you provide the required inputs.

Decision Node Properties

The properties of a Decision activity are listed in the following table and can be edited in the Properties grid on the right pane.

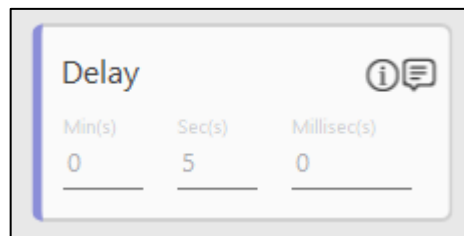
Property Name	Usage
Misc	
Condition	Define a Boolean condition in this field. This condition decides the path which flowchart can follow.
DisplayName	The name of the activity that is displayed in the flowchart. By default, the value is set as Decision . You can edit this value as per your need.
FalseLabel	Decision Node activity display two flows, True and False . By default, the value is set as False. You can edit this value as per the need.

9.5.7 Delay

This activity is used to add a time delay in a process. You can create a timer for any activity and process will be performed once the timer is expired. Delay in time can be defined in Milisec(s), Sec(s) or Min(S).

Using Delay Activity

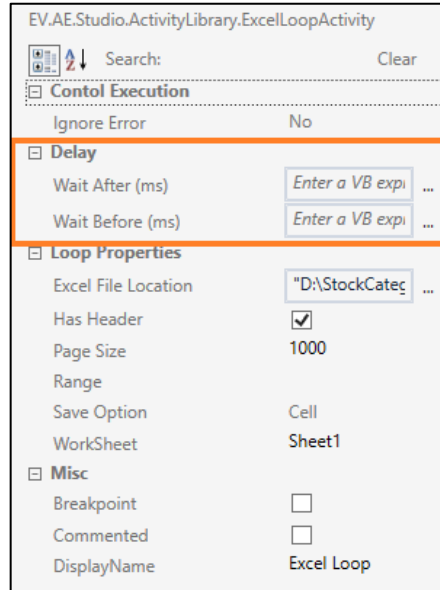
1. In the Canvas Toolspane, click Flow Controls to expand the tool and view the associated activities.
2. Drag the **Delay** activity and drop on to the Flowchart designer on the Canvas



3. In the **Min(s)**, **Sec(s)** and **Millisec(s)** fields enter the time you want to set as delay.

The Delay activity with a default display name is created.

Apart from the Delay activity, pre and post activity delays can also be defined in Properties pane of the activity.



- **Wait After (ms)** - Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
- **Wait Before (ms)** - Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.

Delay Properties

The properties of a Delay activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
Misc	
DisplayName	The name of the activity that is displayed in the flowchart. By default, the value is set as Delay . You can edit this value as per your need.

9.6 Process Components

The Process Components tool is used for automating tasks and processes related to an application. For example, retrieve available information from a target resource, perform actions in an application and so on. The activities in the Process Components tool includes:

9.6.1 Applications

With the help of this activity, applications like web, excel, mainframe, Windows, Java and others can be automated.

The Application activity performs automation using field configuration. Automation Studio supports Multimodal Interface to efficiently capture steps involved in the automation process workflow.

Following are the different types of application that can be automated using application activity and are explained below in detail:


- [WebApps](#)
- [ExcelApps](#)
- [MainframeApps](#)
- [WindowsApps](#)
- [JavaApps](#)
- [SAPApps](#)
- [DatabaseApps](#)
- [CognitiveApps](#)
- [FTPApps](#)
- [SMTPApps](#)
- [NIAApps](#)
- [WebAPIApps](#)
- [Contextual Intelligence Apps](#)

Multimodal Interface

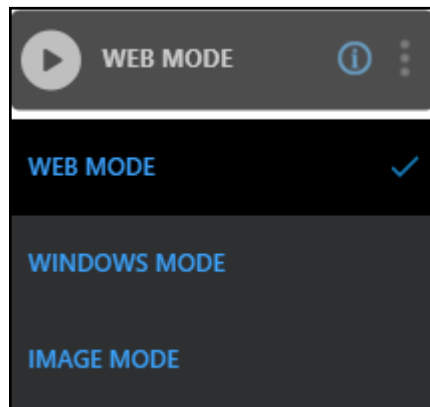
A Multimodal Interface provides multiple modes to interact with the application.

Multimodal Interface helps to automate all the steps involved in the automation process workflow through field configuration using web control, windows control or image based configuration, without docking the application. You can move the Multimodal Interface as per your convenience across the application window. Every step captured through Multimodal configuration is re-

traceable.

Launch Multimodal interface from the  (Settings) icon of the related activities available within Process Components tool. The field properties for each application type is explained in detail in the Application activity. Primary modes available in the Multimodal Interface are:



- [WEB MODE](#) - Captures properties related to the automation of web based application. By default, WEB MODE is selected.
- [WINDOWS MODE](#) - Captures properties related to the automation of Windows application.
- [IMAGE MODE](#) - Captures properties related to the automation of image based actions.




Other modes available as per the application being automated are:

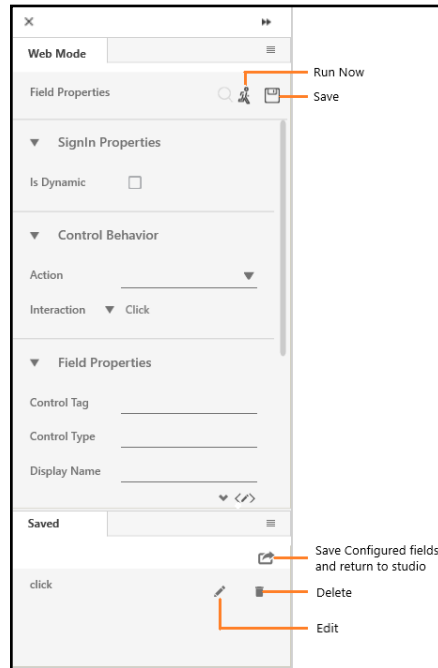
- [.NET OBJECT MODE](#) - Captures properties related to the automation of .Net based windows thick clients. This mode is enabled only when working with windows applications.
- [EXCEL MODE](#) - Captures properties related to the automation of an excel application. This mode is enabled only when working with the excel file.
- [MAINFRAME MODE](#) - Captures properties related to the automation of a mainframe application. This mode is enabled only when working with the mainframe application.
- [JAVA MODE](#) - Captures properties related to the automation of a Java application. This mode is enabled only when working with the Java application.




Multimodal Interface provides the ability to switch among the different modes to automate all the steps related to an application without leaving the application until all the steps are configured. It enables different options to interact with the control present on the application window that enhances user experience to achieve the desired output.

Use the  (Play) icon to capture the steps that you want to perform in the configured application. The application opens and comes into focus. The Play icon changes to the  (Recording) icon. Once you start capturing the steps to automate, the Field Properties pane of the respective modes open at the right side of the screen. Steps to capture the steps are discussed in detail at application level.

Note: While using Web Mode, click the  (Play) icon and hover over the area you want to capture. The area gets highlighted in




red box, See [Web Application](#) automation to know more.

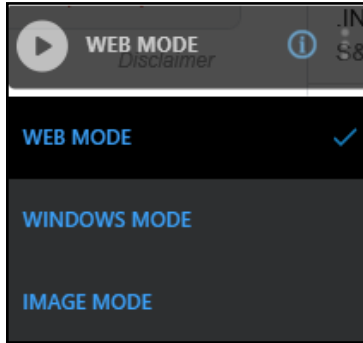


With the help of  (Run Now) icon, you can perform the action (retrace) to see if the intended configuration is done correctly. Click the  (Save) icon to save the configured fields keeping the respective mode open. While click the  (Save Configured fields and return to studio) icon to save the fields configured and return to the Studio menu. Closing the Interface manually might result in slowing down the machine. Studio menu might hang and experience a lag.

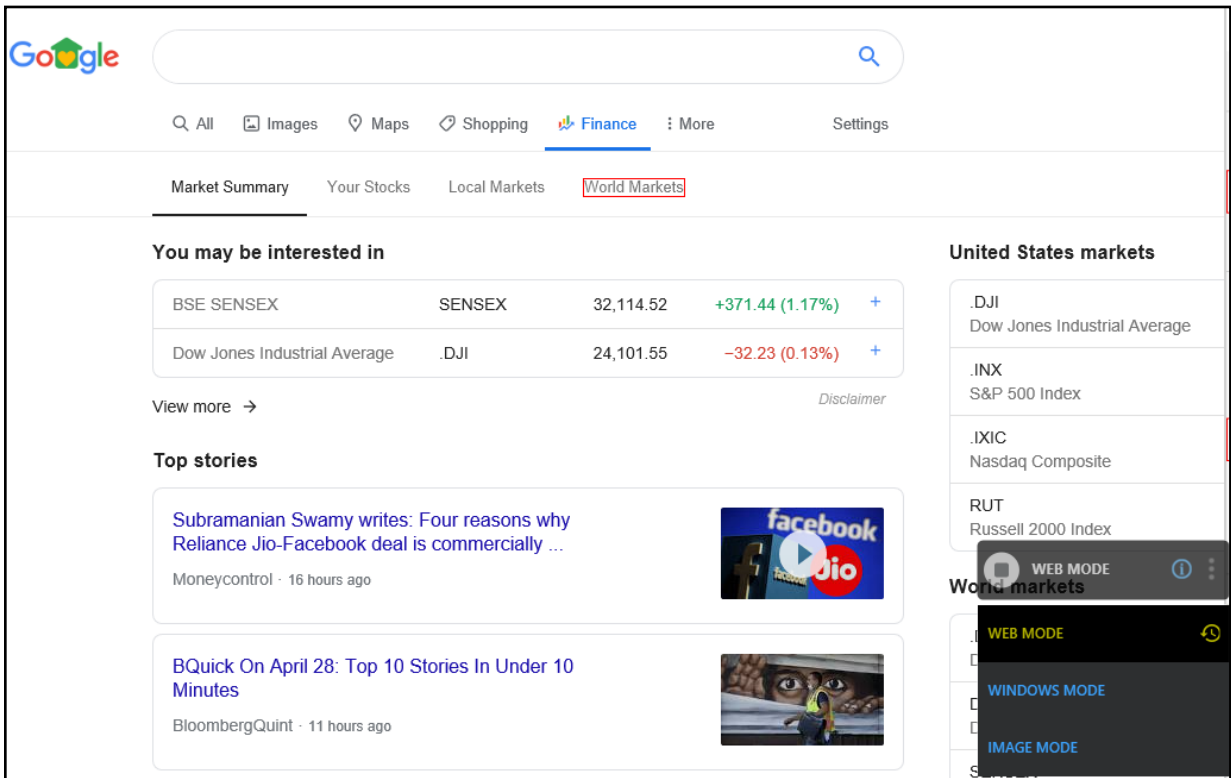
Using Web Mode

Automation of web application using the WEB MODE highlights the area that you can capture for configuration. Once the area is capture, perform the relevant action using the out of box interactions available. Scenarios where the WEB MODE does not work as expected, you can even switch between WINDOWS MODE and IMAGE MODE to configure the field. Following are the instructions to perform field configuration using the WEB MODE:

1. Use the  (**Settings**) icon of the required **Application** activity to launch Multimodal Interface and the configured application. By default, **WEB MODE** is set. See [Web Application](#) automation to know how to use the activity.
2. Click the  (Play) icon to the steps that you want to perform within the web application. The application opens and the Play icon changes to the  (Recording) icon.





3. Hover over the field that you want to capture. The field gets highlighted with red box.



- Click the highlighted area. The Field Properties panel of the Web Mode appears. The fields that are mandatory are highlighted with red box.

The screenshot shows the EdgeVerve Automation Studio interface. On the left, a Google Finance page is displayed with 'Top world markets' and 'United States markets' sections. On the right, the 'Field Properties' panel is open, showing 'Control Behavior' and 'Field Properties' sections. The 'Action' dropdown and 'Display Name' text input are highlighted with red boxes. The 'Field Properties' section includes fields for 'Control Tag', 'Control Type', 'Find By', and 'Tab Number'. The 'Find By' field contains the CSS selector '.qfx13b > nth-child(1) > a'. The 'Tab Number' field is set to 0. Below the 'Field Properties' panel, there is a 'Saved' section with a list of saved configurations: 'World markets WEB MODE', 'WEB MODE', 'WINDOWS MODE', and 'IMAGE MODE'.


- In the Action list, select the action that you want to perform on the captured area. Other mandatory fields change depending on your selection from the drop-down list
- In the **Display Name** field, enter a desired name of your choice.
- Enter details of all the mandatory fields and other relevant fields as per your requirement. Refer [Web Mode Control Behavior and Field Descriptions](#) table to know more about the available fields and their respective properties.
- Click the  (Save) icon to configure the actions and other related fields.
- Repeat step 2 through 6 to capture all the steps involved in the process. The fields configured and saved appears in the Saved pane of the Field Properties panel. You can edit or delete these steps in the Saved pane.
- Click the  (Save configured field and return to studio) icon to save the field that you configured and return to the Studio menu.

The fields related to Web MODE are configured.

Web Mode Control Behavior and Field Descriptions

The properties of Web Mode are listed in the following table:

Property Name	Usage
Control Behavior	
Action	<p>It is the action or the interaction that can be performed on a web based application depending on the type of UI element as per the requirement. Available options are:</p> <ul style="list-style-type: none"> ▪ Control Exists- Checks if the UI element exists at the specified area. It is useful when working with dynamics controls. ▪ Set Value- Sets the user provided value of the UI element attribute and stores it as the value of the parameter. You must define the parameter in the Parameter bar before you start capturing the element. ▪ Set Attribute- Sets the user provided attribute for the specified UI element and stores the provided value of the attribute into a parameter. You must define the parameter in the Parameter bar before you start capturing the element. ▪ Get Value- Extracts the current value of the selected UI element such as input, select, textbox. ▪ Get Attribute- Extracts the attribute's value of the selected UI element. ▪ Click- Allows to left click the selected UI element. ▪ Click NoWait- Allows to click the selected UI element and move to the next activity without waiting for the page to load. ▪ Double Click- Performs double click on the selected UI element. ▪ MouseOver- Hovers the mouse over the selected UI element. ▪ SendKeys- Captures input of the selected UI element from the keyboard. The input can be a combination of keys or text input from the keyboard. Use this option when a keystroke needs to be automated on a specific web element and not on the overall application. ▪ Wait Until Exists- Waits until the UI element is found. You can use this to avoid any delay in the page loading time. This interaction returns a Boolean value of true or false (on success or failure respectively) on finding the image. The Boolean value can be set to a variable and checked in a conditional statement to perform an appropriate action after finding the web element. ▪ Scroll By- Scrolls to particular section of the web page horizontally or vertically. ▪ Scroll Into View- Scrolls to bring the application area viewable on screen of the web browser. ▪ Shift Click- It appears only when you want to capture action on a button or a link. this interaction opens the URL associated with the selected link or the

	button, in a new window.
Interaction	The corresponding interaction set against the selected action.
Appears if Action selected is SendKeys.	
Keys	<p>The key sent as input from the keyboard. It can be a combination of keys or a text.</p> <p>To configure the keys:</p> <ol style="list-style-type: none"> Once Action selected is SendKeys, click the  (Configure) icon. The SendKey Configuration dialog box appears. <div data-bbox="613 611 1260 1041" data-label="Image"> </div> <ol style="list-style-type: none"> In the Select Function Keys (Max 3) list, select the keys from the keyboard to create the key combination. You can select up to maximum of three keys. In the Input Key(optional) field, enter an alphabetical or a numeric key to create the combination. This is an optional field. Click CONFIRM. <p>The configuration for SendKeys action is done.</p>
Appears if Action selected is Get Attribute and Set Attribute.	
Attribute	Sub attributes or the properties of the html attribute selected.
Appears if Action selected is Set Value, Set Attribute.	
Input Source	<p>The input value that must be provided. You must define a parameter in the Parameter bar to use this option.</p> <p>Select the checkbox beside Input Source, and enter the input value, if you want to set a default value.</p>
Appears if Action selected is Scroll By.	
XAxis	Vertical scroll on the web page. The value must be entered in pixels.
YAxis	Horizontal scroll on the web page. The value must be entered in pixels.
Appears if Action selected is Scroll Into View.	


<p>Scroll To</p>	<p>Determines the area till where the scroll must be done. Available options are:</p> <ul style="list-style-type: none"> ▪ Start ▪ Center ▪ End <div style="background-color: #007bff; color: white; padding: 5px; border: 1px solid black;"> <p>Note: Internet Explorer does not support Centre option, so configuring center for ScrollTo displays exception at runtime.</p> </div>
<p>Dock Window</p> <p>This section appears only when Action selected is Shift Click.</p>	
<p>Dock Action</p>	<p>When a link or button on the web page is opened in a new browser window, the Dock Action button enables the new window listed in the multimodal UI. You can select one of the windows in the multimodal UI to configure controls on the intended window.</p>
<p>Interaction Attributes</p> <p>This section appears only when Action selected is Shift Click.</p>	
<p>Tab Name</p>	<p>Name of the new window when it is docked in Engage application.</p>
<p>Target Url</p>	<p>URL of the new window that opens when Shift Click is selected in Action.</p>
<p>Use Mouse</p>	<p>Performs left click of the mouse on the link or the button to open in the new browser window.</p>
<p>Field Properties</p>	
<p>Control Tag</p>	<p>User specified identifier of the selected UI element.</p>
<p>Control Type</p>	<p>Control type of the UI element as defined by the developer.</p>
<p>Display Name</p>	<p>User specified display name of the action configured.</p>
<p>Find By</p>	<p>Locator of the UI element on a web application.</p> <p>Identifying the correct element forms one of the first step towards creating an automation workflow. This Isit changes depending on the properties of the UI elements defined. Commonly available options are:</p> <div data-bbox="776 1570 1091 1793" style="border: 1px solid black; padding: 5px; margin: 10px auto; width: fit-content;"> </div> <ul style="list-style-type: none"> ▪ Id - Identifier of the UI element. It is useful for locating elements that has a unique Id associated with it.

- **name** - Field name of the control. Each element on the web page has a name assigned to it which can be used to select the intended element.
- **css3selector** - Part of the CSS rule that uses selectors such as tags, class, pseudo-element, pseudo-class, combination of selectors and other wide range of selectors to identify the UI element.
- **css3xpath** - Depending on the html tagging, css3xpath can also be available in this list. It uses the Xpath expression to locate the UI element.
- **manual** - Provides manual way of identifying the UI element. You can use this option when the selected set of properties available for the UI element are not able to identify the required it at automation runtime.

You can configure backup selectors to avoid situations where the automation process is unable to find the primary selector. This is not a mandatory step.

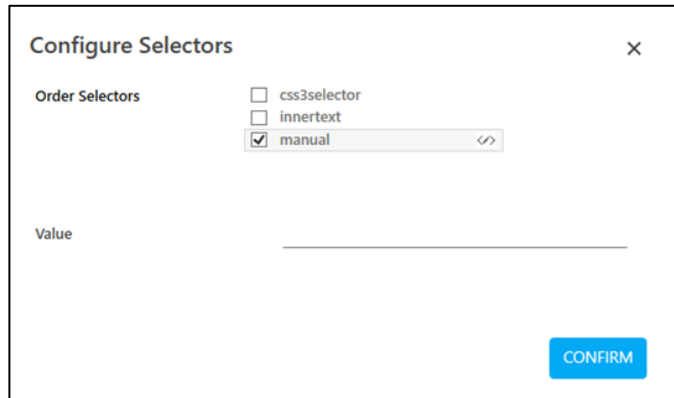
Follow the instructions to configure the backup selectors:

1. Click the  (**Configure**) icon. The **Configure Selectors** dialog page appears.




The screenshot shows a dialog box titled "Configure Selectors" with a close button (X) in the top right corner. Under the heading "Order Selectors", there are four checkboxes: "id" (checked), "name", "css3selector", and "manual". Below this is a "Value" label followed by an empty text input field. A blue "CONFIRM" button is located in the bottom right corner.

2. Select the additional selectors.
 - If the selector is **manual**:



The screenshot shows the same "Configure Selectors" dialog box. In this view, the "manual" checkbox is checked and highlighted with a grey background and a pencil icon. The other checkboxes, "css3selector" and "innertext", are unchecked. The "Value" input field and the "CONFIRM" button are also visible.

- a. Click the  (**pencil**) icon to configure the manual selector. The **Configure Selector** dialog box appears.

	<div data-bbox="581 237 1287 657" data-label="Image"> </div> <ol style="list-style-type: none"> b. In the Select Attribute list, select the attribute of the manual selector. As per the HTML structure and the properties of the UI element defined, the are displayed in this list. c. In the Value list, either select a parameter containing the value of the selected attribute or select the check box to set a default value. d. In the Condition list, select the condition of the selector. Available options are- Equals, Contains, Starts With and Ends With. e. The Generated Selector field displays the default value generated, if any. You can click Add to add this value in the Selector field. f. Click CONFIRM. <ol style="list-style-type: none"> 3. In the Value field, the captured selector is displayed. 4. You can drag the selector and drop it to change the sequence as per the priority of the selector you want to set. At the time of process execution, the selector is considered as per the sequence set, the top most having the highest priority while the the last one having the least priority for locating the UI element. 5. Click CONFIRM to save the configuration.
<p>Appears if Action selected is Control Exists, Get Attribute.</p>	
<p>Tab Number</p>	<p>The window tab on which you want to perform automation.</p>
<p>Appears if Action selected is Control Exists, Get Attribute.</p>	
<p>Variable Name</p>	<p>User defined name of the variable that stores the captured value. You can define the variable name here itself or predefine it in the Parameter bar. Refer Parameter section in the AE-RPA-Engage-AutomationStudio-UserManual.pdf if you want to know more about parameters and how to use it.</p>
<p>Appears if Action selected is Control Exists, Get Attribute</p>	
<p>Variable Type</p>	<p>Type of the variable defined. Available options are:</p> <ul style="list-style-type: none"> ▪ Text - A sequence of character, either as a constant or a variable. ▪ Numeric - An integer type that range from negative through positive.

	<ul style="list-style-type: none"> ▪ Decimal - An exact numeric value defined by its precision and scale. ▪ Boolean - A data type used for making decision. Can have only two values- true or false. ▪ Data Table - A data type that stores tabular data.
Misc	
FrameNo	Number of the frame of the web page where you want to perform the action. Automation Studio automatically detects the frame number based on the area selected during configuration.
Item Index	Index number of the selected item on the web page.
Item Offset	The distance between the target item and the anchor.


Using Windows Mode

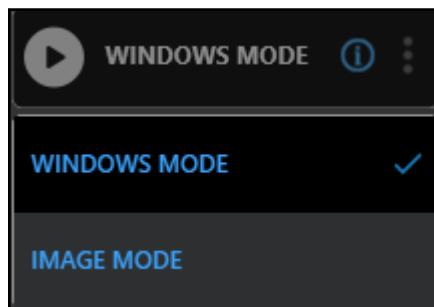
Unlike automation of web application, while configuring UI elements of the windows application, systems highlights the intended area and performs the subsequent action rather than just capturing the step of automation process workflow. For example, if you want to capture the X (Close) button of the windows application to automate closing of the application, the system closes the application when you click the area on the UI even before launching the Field Properties pane. To help avoid such scenarios, WINDOWS MODE configuration lets you configure the UI elements in two different ways:


- **Default Configuration**- Allows to configure the UI elements of the windows application by highlighting the area. It is useful when you only want to capture the area and there is no subsequent action taking place.
- **Advance Configuration**- Allows to configure the UI elements of the windows application in a hierarchical manner available within an element tree. You can select the intended UI element to open the **Field Properties** pane without performing the subsequent action.

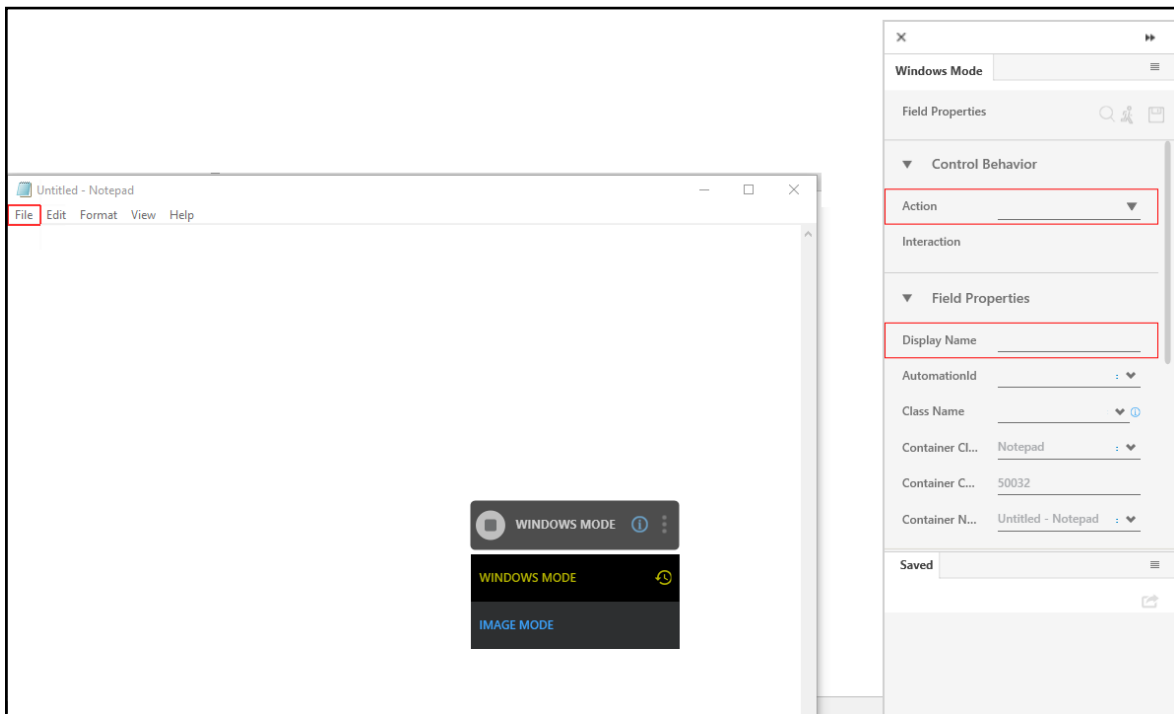
Scenarios where the WINDOWS MODE does not work as expected, you can even switch between WEB MODE and IMAGE MODE to configure the field.



Default Configuration

1. Use the  (**Settings**) icon of the required **Application** activity to launch Multimodal Interface. The configured application opens in the undocked mode. See [Windows Application](#) activity to know how to use the activity.





2. Click the  (Play) icon to capture the UI elements within the intended application.
3. Press the Ctrl key, hover over the windows application and simultaneously select the UI element to start capturing the automation steps that you want to perform in the windows application. The UI element that you can capture gets highlighted with a red box.
4. Click the highlighted area. The Field Properties panel of the Windows Mode of the Windows application appears. The fields that are mandatory are highlighted with red box.

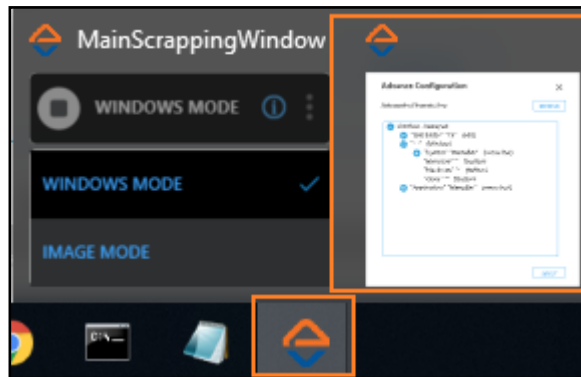


5. In the Action list, select the action that you want to perform on the captured area. Other mandatory fields changes depending on the selection from the list.
6. In the **Display Name** field, enter a desired name of your choice.
7. Enter details of all the mandatory fields and other relevant fields as per your requirement. Refer [Windows Mode Field Properties and Control Behavior](#) table to know more about the available fields and their respective properties.
8. Click the  (Save) icon to configure the fields.
9. Repeat step 2 through 7 to capture all the steps involved in the automation process workflow. The field that you configured and saved appears in the Saved pane of the Field Properties panel. You can edit or delete these steps in the Saved pane.
10. Click the  (Save configured field and return to studio) icon to save the field that you configured and return to the Studio menu.

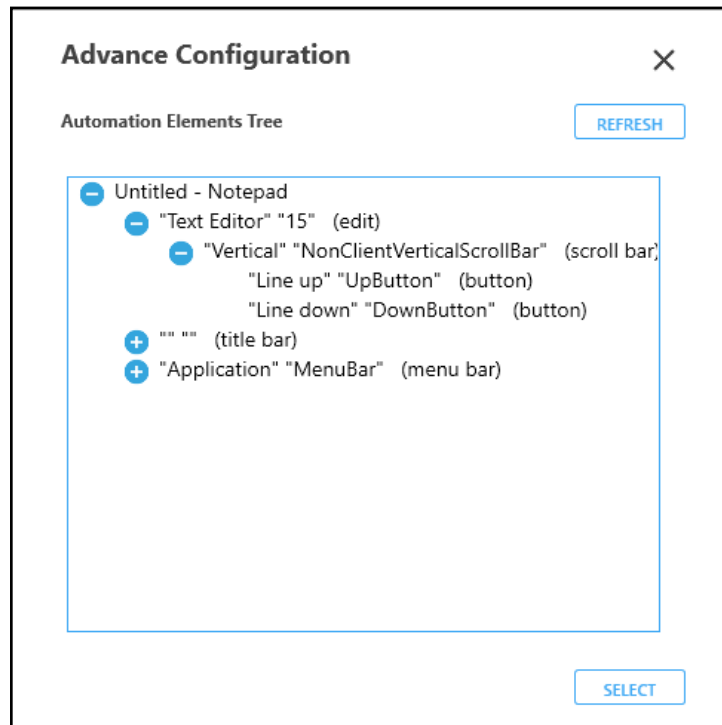
The fields related to WINDOWS MODE using default configuration are configured.

Advance Configuration

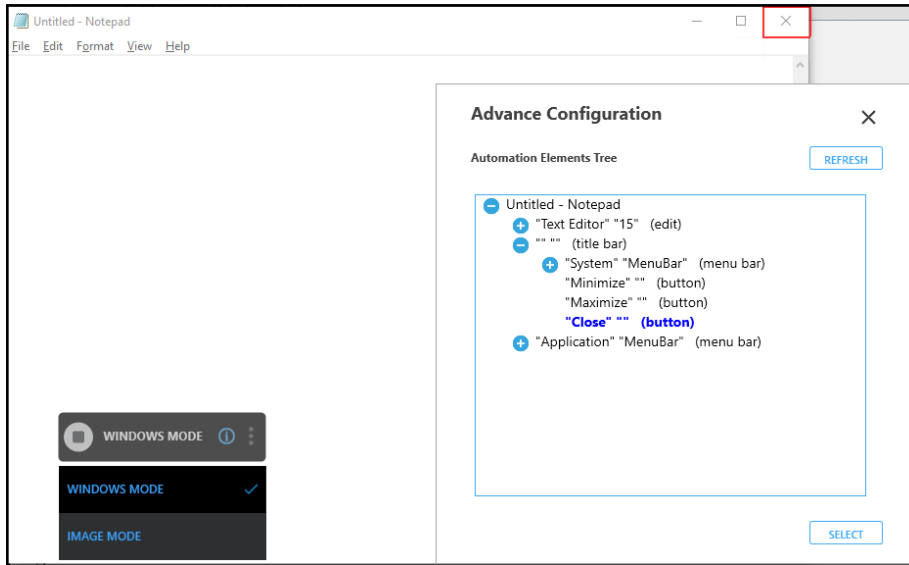
11. Use the  (**Settings**) icon of the required **Application** activity to launch Multimodal Interface. The configured application opens in the undocked mode. See [Windows Application](#) activity to know how to use the activity.
12. Click the  (Play) icon to capture the UI elements within the intended application. The **Advance Configuration** window appears in the **Task** bar of the system, available behind the **Multimodal Interface** window.



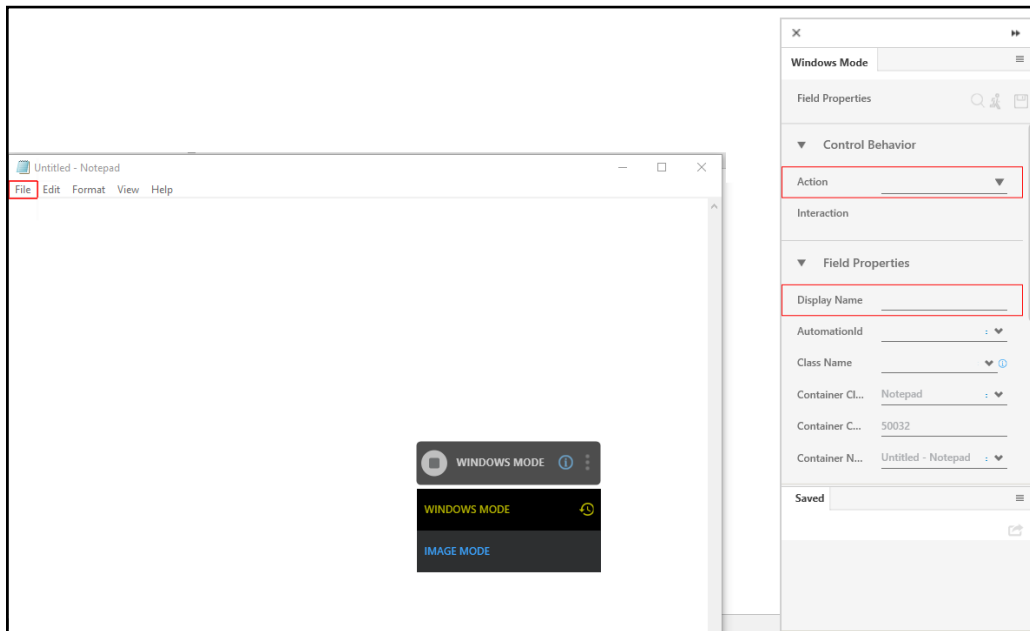
13. Click the **Advance Configuration** window to bring it in focus. It displays all the UI elements of the application under a tree structure.




14. Expand and select the element from the **Automation Element Tree**. The corresponding UI element on the windows applications gets highlighted in red. It helps to identify the required UI element for automation.




15. Click **SELECT**. The Field Properties panel of the Windows Mode of the Windows application appears. The fields that are mandatory are highlighted with red box.



16. In the Action list, select the action that you want to perform on the captured area. Other mandatory fields changes depending on the selection from the list. Enter details of all the mandatory fields and other relevant fields as per your requirement. Refer [Windows Mode Control Behavior and Field Descriptions](#) table to know more about the available fields and their respective properties.
17. Click the  (Save) icon to configure the fields.
18. Repeat step 2 through 7 to capture all the steps involved in the automation process workflow. The field that you

configured and saved appears in the Saved pane of the Field Properties panel. You can edit or delete these steps in the Saved pane.

- Click the  (Save configured field and return to studio) icon to save the field that you configured and return to the Studio menu.

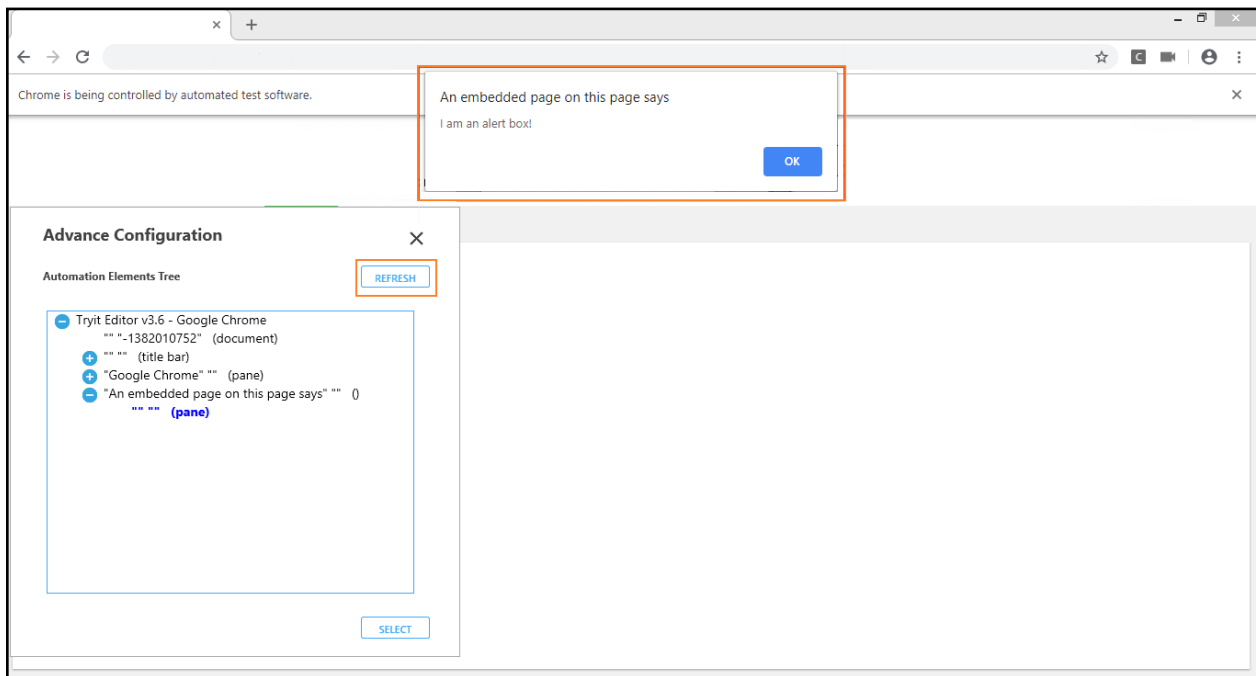
The fields related to WINDOWS MODE using advance configuration are configured.

Advance Configuration Special Cases

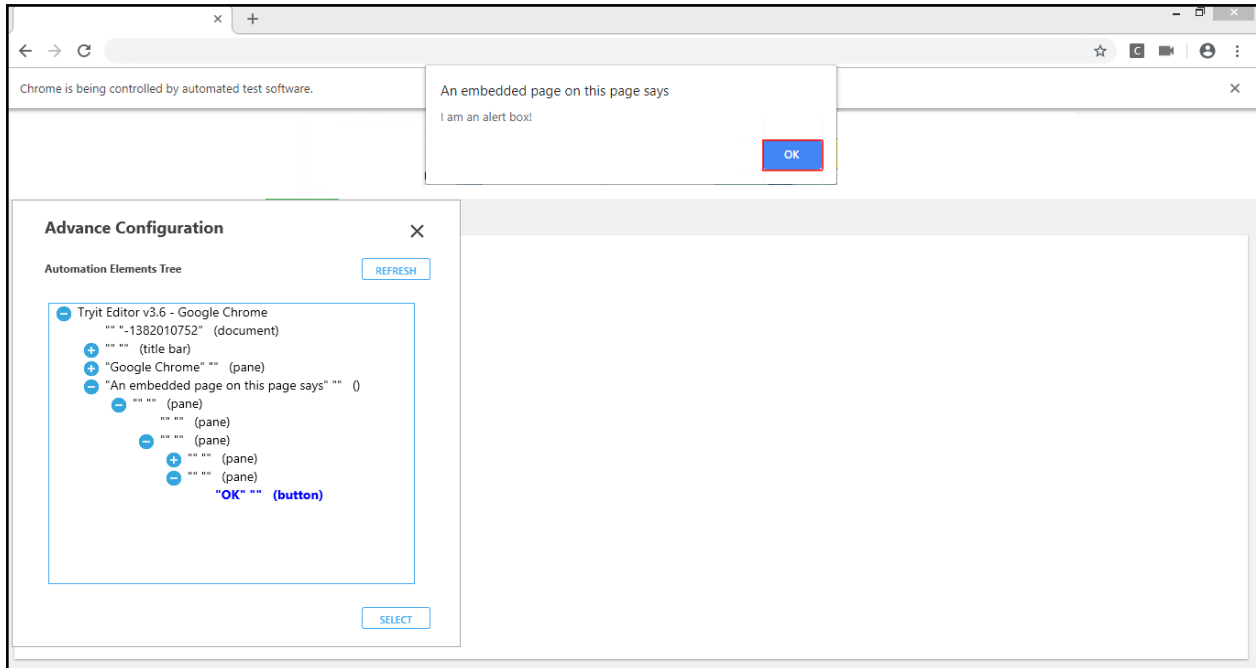
In some cases, it is observed that while configuring the JavaScript pop-up in Chrome browser, the Advance Configuration tree is not able to highlight some of the inner UI elements of the pop-up box.

To highlight the intended element, try highlighting the closest element from the target UI element, and then click REFRESH button on the Advance Configuration tree.

In the below sample screen shot, the OK button does not appear in the Advance Configuration tree. Try highlighting the pane itself, and then click REFRESH.



The Advance Configuration tree tries to find the target element that must start appearing after refreshing the configuration tree.



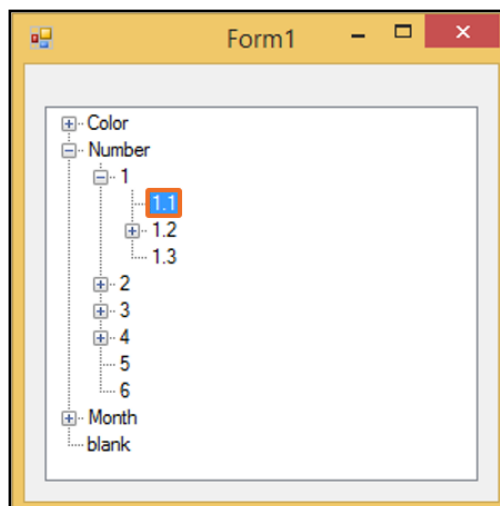
Windows Mode Control Behavior and Field Descriptions

The properties of Windows Mode are listed in the following table and can be edited in the Field Properties panel.

Property Name	Usage
Control Behavior	
Action	<p>The applicable action or interaction that can be performed on the selected UI element. The available options change as per the window application being automated:</p> <ul style="list-style-type: none"> ▪ Click- Allows to left click the selected UI element. ▪ Expand/Collapse- Allows to expand or collapse the selected UI element. ▪ Legacy Interaction- Allows to configure UI element of the Windows application which supports a legacy pattern. ▪ Get Text- Retrieves text from the selected UI element. ▪ Set Text- Sets user provided text input to the selected UI element and stores it into a parameter. You must define the parameter in the Parameter bar before you start capturing the UI element. ▪ Mouse Click- Allows mouse simulation. It requires the application to be in focus. You can choose to perform Left Click, Right Click or Double Click available from the drop-down list. ▪ Keyboard Input- Captures UI element input from the keyboard in the selected UI element. This interaction allows performing different keyboard input combinations with the help of Single Key, Double Keys, Triple Keys

and **TextEntry** options.

- **Set Focus**- brings the UI element in focus before performing any action on it.
- **If Control Exists**- Checks availability of the UI element on the windows application. It is used while working with dynamic controls.
- **Get Radio Button Status**- Retrieves status of the selected radio button status. It returns true or false.
- **Toggle Checkbox Action**- Toggles the current state of a checkbox.
- **Get Checkbox Status**- Returns the current selection status of a checkbox.
- **Perform ComboBox Select Value**- Selects the specified value from the drop-down list.
- **Get ComboBox Value**- Retrieves value of the selected combo box.
- **Perform ComboBox Set Text**- Sets a user-specified value on the selected ComboBox. All ComboBox controls do not support this interaction.
- **Scroll Horizontal**- Performs horizontal scroll based on the input provided on the percentage scale from 0 to 100.
- **Scroll Vertical**- Performs vertical scroll based on the input provided on the percentage scale from 0 to 100.
- **Select Tree Node**- Selects a node of a tree control using this interaction. Specify the input in a specific format. For example, for selecting node **1.1**, enter the input value as **Number/1/1.1**.



- **Expand Tree Node**- Expands a node of the tree UI element.
- **Select**- Selects a particular UI element.
- **Get DataGrid Value by Index**- Retrieves data grid value by its index. Specify the cell index from which the input is needed.

	<ul style="list-style-type: none"> ▪ Focus Data Grid Cell by Index- Brings a data grid cell in focus based on the input cell index. ▪ Set DataGrid Value by Index- Sets the value to a DataGrid cell based on its index. ▪ Get Data from Table- Retrieves complete data of a table control to a Data Table. If required, the data from the Data Table can then be saved in an excel/CSV file. Following are the steps involved to retrieve the data. <ul style="list-style-type: none"> a. Create an argument of type System.Data.DataTable, before extracting the UI element. b. Click WINDOWS MODE to extract the Data Grid Table control. c. Select Get Data from the Table. d. Set the Variable Name as the argument created in the first step. e. Save the configuration. <ul style="list-style-type: none"> Use Export Data Table activity to write data to excel. ▪ Set Focus- This interaction is used to bring any control to focus before performing an action on it.
Interaction	The corresponding interaction class name auto populated against the selected action.
Interaction	The interaction or the action type corresponding to the selected interaction.
Appears if Action selected is Legacy Interaction .	
Action Type	The interaction or the action type corresponding to the legacy interaction.
Key	Appears if Action selected is Keyboard Input . The single key set as input from the keyboard.
Key	Appears if Action selected is Keyboard Input and Interaction selected is DoubleKeys . The second key which is set as input from the keyboard along with the first key.
Key	Appears if Action selected is Keyboard Input and Interaction selected is TripleKeys . The third key which is set as input from the keyboard along with the first and the second key.
Text	Appears if Action selected is Keyboard Input and Interaction selected is TextEntry . Allows user input as a text.
Mouse Action	Appears if Action selected is Mouse Click. Allows user input using the mouse. Available options are: LeftClick - Performs a left click at the specified offset point. DoubleClick - Performs a double click at the specified offset point. RightClick - Performs a right click at the specified offset point.
Field Properties	
Display Name	User specified display name of the windows element selected.

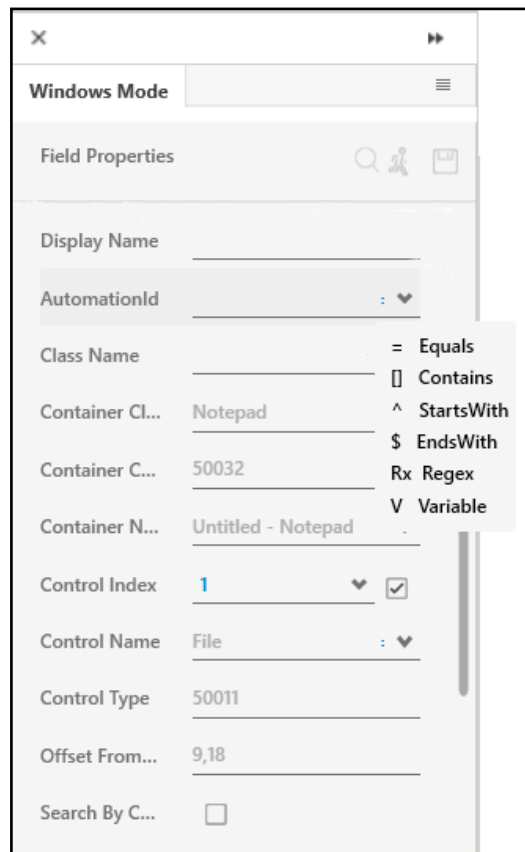
AutomationId	Unique identifier for the automation element in the automation tree. See Handling Dynamic Controls section to know more about the usage of this field.
Class Name	Class name of the UI element as defined by the UI element developer. See Handling Dynamic Controls section to know more about the usage of this field.
Container Class Name	Container class name of the UI element as defined by the developer. See Handling Dynamic Controls section to know more about the usage of this field.
Container Control Type	Control type of the container of the UI element as defined by the UI element developer.
Container Name	Container name of the UI element as defined by the developer. See Handling Dynamic Controls section to know more about the usage of this field.
Control Index	Index of the current UI element within the entire set of elements with the same values of automation Id, class name and UI element name as that of the current element.
Control Name	Control name of the UI element as defined by the UI element developer. See Handling Dynamic Controls section to know more about the usage of this field.
Control Type	Control type of the UI element as defined by the developer.
Appears if Action selected is Mouse Click .	
Offset From Control	It is the distance between the UI element and the anchor.
Appears if Action selected is Get Text .	
Variable Name	User defined name of the variable that stores the captured value.
Variable Type	Type of the variable defined. Available options are: <ul style="list-style-type: none"> ▪ Text - A sequence of character, either as a constant or a variable. ▪ Numeric - An integer type that range from negative through positive. ▪ Decimal - An exact numeric value defined by its precision and scale. ▪ Boolean - A data type used for making decision. Can have only two values- true or false. ▪ Data Table - A data type that stores tabular data.
Misc Properties	
Max wait for input idle	The maximum time interval in milliseconds for which the windows plugin waits on launch before it starts executing the configured interactions.
Retry Count	Number of times, the plugin retries to find a UI element, if not found.
Retry Interval (ms)	The time interval in milliseconds for which the plugin waits before it attempts for a retry.


Handling Dynamic Controls

Automation Studio provides different identification criteria and Parametrization of the value of the dynamic field properties of the UI elements. You can reconfigure some of the auto captured field properties of such UI elements to create a more effective way of identifying the indented UI element. For such controls, remove the dynamic part of the string and use an appropriate option for UI element identification. For example, if a Class Name field is recognized as AssistEdge_Studio_20180205083009, the later part of the string, which is a timestamp, is dynamic and can be removed. Retain only the static part and reconfigure the field properties using the suitable option.

Following are the different field properties that can be reconfigured against their respective values:

- AutomationId
- Class Name
- Container Class Name
- Container Name
- Control Name



Use the  (drop-down arrow) icon to reconfigure the identification criteria. Available options are:

- **Equals-** Identifies the UI element based on the exact match with the value of the selected field property.
- **Contains-** Identifies the UI element based on the string present in the value of the selected field property.

- **StartsWith**- Identifies the UI element based on the starting string of the value of the selected field property.
- **EndsWith**- Identifies the UI element based on the ending string of the value of the selected field property.
- **Regex**- Identifies the UI element with the string matching the value of the selected field property as per the identification pattern defined through the provided regular expression such as a.b, *txt and others.
- **Variable**- Identifies the UI element as per the parametrization of the value of the selected field property value. Parametrization allows to run the identification process over and over again using different values. With parameterization, windows controls with looping and assignment activities, are used.


.Net Object Mode

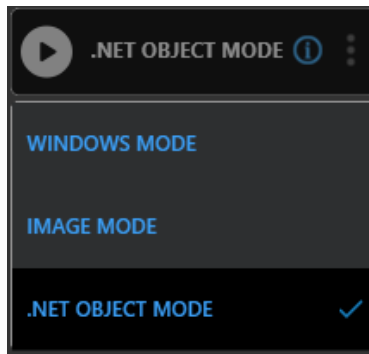
It helps you to automate complex UI controls of the .Net based Windows applications (WPF, WinForms). This mode is visible only when the configured application is of WindowsApps type.


Note: .NET Object Mode supports applications that are created using .NET technologies (WPF, WinForm) and target .Net Framework from 4.0 to 4.8. .Net Frameworks released after v4.8 are currently not supported.

This feature lets you configure the UI controls providing its object. You can write your own C# code for performing the respective action.

Following are the instructions to perform field configuration using the .NET OBJECT MODE:

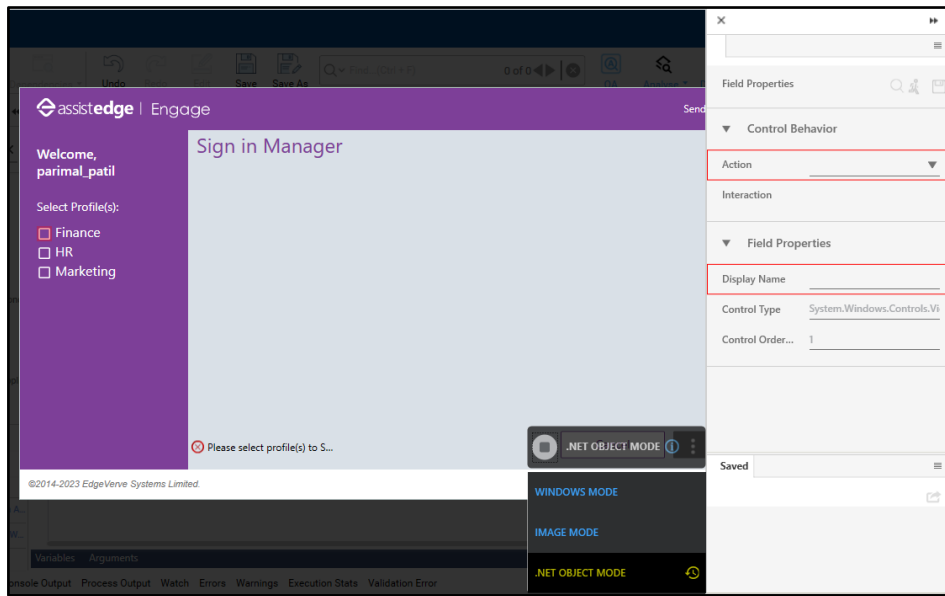
1. Use the  (**Settings**) icon of the required **Application** activity to launch Multimodal Interface. The configured application opens in an undocked mode. See [Windows Application](#) activity to know how to use the activity.



2. Click the  (Play) button to capture the UI elements within the intended application.
3. Press the Ctrl key, hover over the application and simultaneously select the UI element to start capturing the automation steps that you want to perform. The UI element that you can capture gets highlighted with a red box.


Note: If the UI element does not get highlighted there is a possibility that the added application is not compatible with the .Net Object Mode. In this scenario Error while detecting object in .net object mode, application might not be compatible error message is logged in the Automation Studio log file.

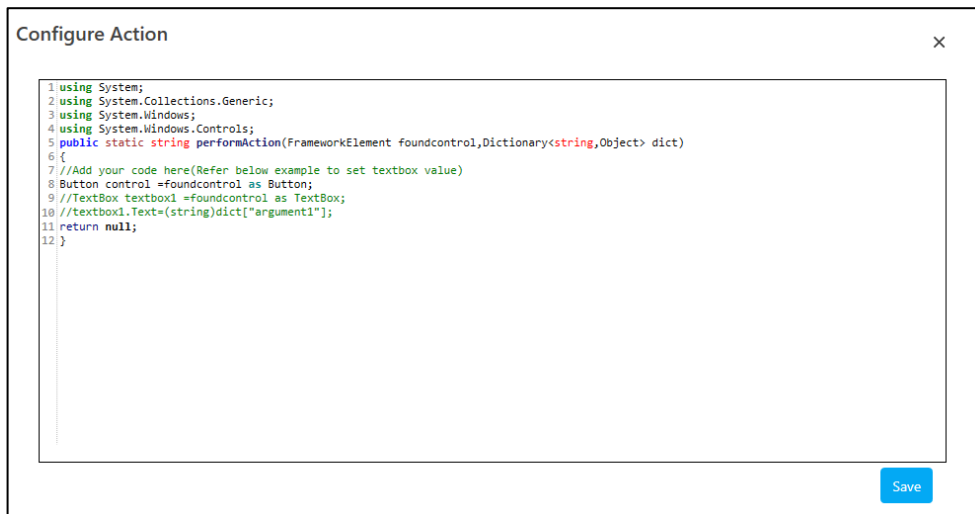
- Click the highlighted area. The Field Properties panel of the **.NET OBJECT MODE** appears. The fields that are mandatory are highlighted with red box.



- From the Action dropdown list, select the required action to be performed on the selected control. The list of actions available in the dropdown varies as per the control type selected for the automation.

If required action is not available in the dropdown or any custom functionality is required, then you can write custom c# code for the required action. To write the custom code:

- From the **Action** dropdown list, select **Code Editor** and click the  (horizontal ellipses) button to open the **Configure Action** screen. The **Configure Action** screen offers a piece of code built in C# to do specific tasks related to certain fields or perform operations associated with the extracted object of the application.





The control parameter in the Configure Action screen is the target UI control on which the desired action can be configured.

Additionally, the Configure Action screen lets you pass the parameter of the process workflow in the code as per the below format:

```
TextBox textbox =foundcontrol as TextBox;
```

```
textbox1.Text=(string)dict["argument1"];
```

Here argument1 is the parameter defined in the process workflow that holds the required value.

7. Click **Save** to save the entered code.
8. Click the  (Save) icon of the **Field Properties** pane to save the configured fields.
9. Repeat step 2 through 7 to capture all the steps involved in the automation process workflow. The field that you configured and saved appears in the Saved pane of the Field Properties panel. You can edit or delete these steps in the Saved pane.
10. Click the  (Save configured field and return to studio) icon to save the field that you configured and return to the Studio menu.

The fields related to .NET OBJECT MODE using default configuration are configured.


[.Net Object Mode Control Properties and Field Descriptions](#)

The properties of .Net Object Mode fields are listed in the following table:

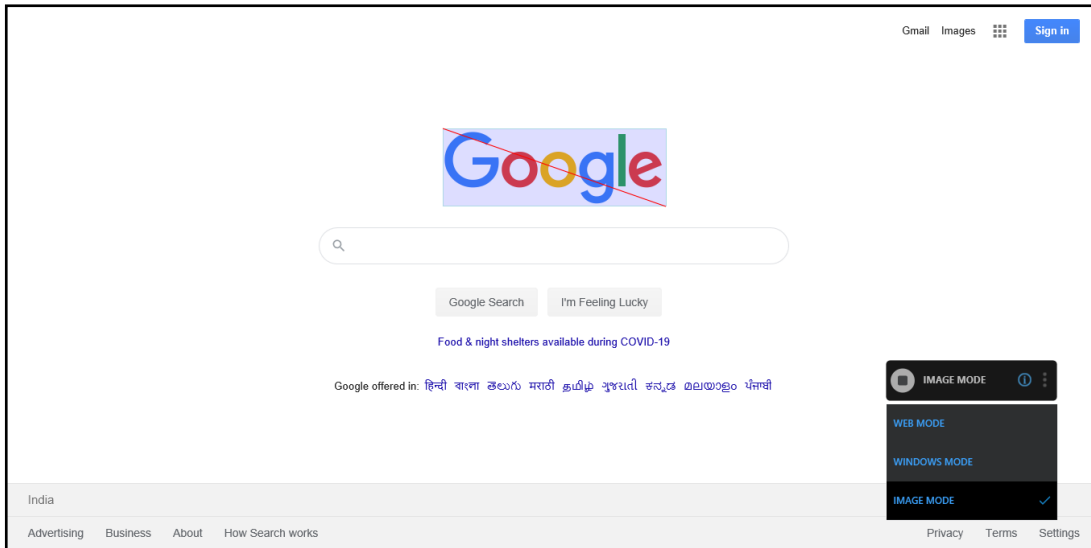
Property Name	Usage
Display Name	User specified display name of the .Net based windows element selected.
Control Type	Control type of the UI element as detected in the target application. This field is not editable.
Control Order	The order of the UI element.
Action	User specified code to perform the required action.
IsExtracted	Indicates if the action performed extracts any text.

Using Image Mode

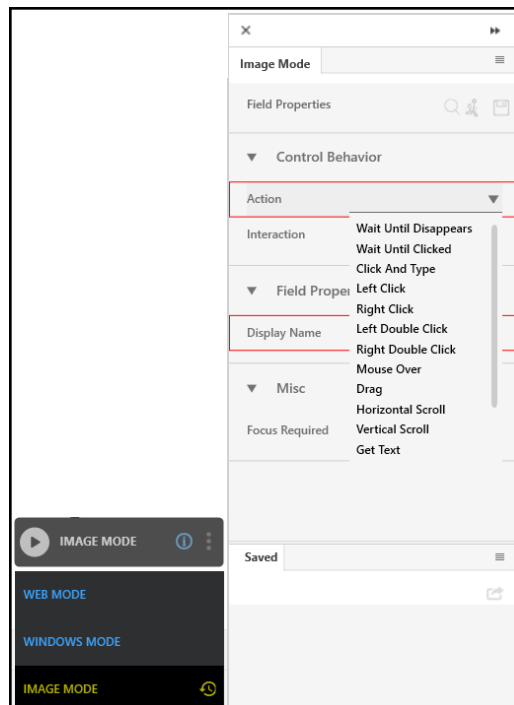
It is helpful when any of the control available on the web page or the window application does not get captured using the WEB MODE or the WINDOWS MODE. Following are the instructions to perform field configuration using the IMAGE MODE:

1. Use the  (**Settings**) icon of the required **Application** activity to launch Multimodal Interface and the configured application. See [Application](#) activity to know how to use the activity.
2. Select IMAGE MODE.

3. Click the  (Play) icon to capture the image within the intended application. The application opens.





4. Mark the image boundaries you want to capture in a rectangular area and then click the marked area. The cursor turns into a + (Plus) symbol and the diagonals of the rectangular area (from left to right diagonal) is marked with a red line, indicating the area that gets captured. The **Field Properties** pane of the **Image Mode** appears. The fields that are mandatory are highlighted with red box.



5. In the Action list, select the action that you want to perform on the captured area. Other mandatory fields change depending on your selection from the drop-down list.
6. In the **Display Name** field, enter a desired name of your choice.
7. Enter the other required details. Refer [Image Mode Control Behavior and Field Descriptions](#) table to know more about the

available fields and their respective properties.

8. Click the  (Save) icon to configure the actions and other related fields.
9. Repeat step 3 through 7 to capture all the steps involved in the process. The fields configured and saved appears in the Saved pane of the Field Properties panel. You can edit or delete these steps in the Saved pane.
10. Click the  (Save configured field and return to studio) icon to save the field that you configured and return to the Studio menu.

The fields related to IMAGE MODE are configured.

Double click the Application activity to configure the properties related to image automation.

Based on the type of action configured two different interactions gets triggered via IMAGE MODE - Control Image or OCR. Refer [Image Control](#) and [Text Extractor](#) activities to know details and steps related to capturing an image to perform certain action on it and extract texts from a text-based image respectively.


Image Mode Control Behavior and Field Descriptions

The properties of Image Mode are listed in the following table:

Property Name	Usage
Control Behavior	
Action	<p>Actions that can be performed on a web based application. Available options are:</p> <ul style="list-style-type: none"> ▪ Wait Until Exists- Waits till the image is found on the web page and is in focus. You can use this to avoid any delay that might occur for the image to appear. This interaction returns a Boolean value of true if the image is found and false if the image is not found. The Boolean value can be set to a variable and checked in a conditional statement to perform an appropriate action after finding the image. ▪ Wait Until Disappears- Waits until the image disappears from the web page. You can use this to avoid delay that might occur for the image to disappear. This interaction returns a Boolean value of true if the image is not found and false if it is found. The Boolean value can be set to a variable and checked in a conditional statement to perform an appropriate action after finding the image. ▪ Wait Until Clicked- Waits until the image is clicked. This interaction returns a Boolean value of true if the image is clicked and false if the image is not clicked. The Boolean value can be set to a variable and checked in a conditional statement to perform an appropriate action after finding the image. ▪ Click And Type- Allows to left click at the required area and provide the input

Property Name	Usage
	<p>value. Set a default value or pass the value using parameter defined in the Parameter bar.</p> <ul style="list-style-type: none"> ▪ Left Click- Performs a left click specified at the offset point. ▪ Right Click- Performs a right click at the specified offset point. ▪ Left Double Click- Performs a left double click at the specified offset point. ▪ Right Double Click- Performs a right double click at the specified offset point. ▪ Mouse Over- Moves mouse pointer over the area at the specified offset point. ▪ Drag- Moves the draggable image based upon the mentioned vertical and horizontal coordinates. ▪ Horizontal Scroll- Searches for the image towards either right or left direction based on the speed of the scroll as per the option selected in the Scroll Direction and Scroll Action fields respectively. ▪ Vertical Scroll- Searches for the image in the upward or downward direction based on the speed of the scroll as per the option selected in the Scroll Direction and Scroll Action fields respectively. ▪ Get Text- Extracts text from the selected image. ▪ Type Text- Allows entering the text where the cursor is present on the page. Set a default value or pass the value using parameter defined in the Parameter bar. ▪ Press Keys- Performs press action for the specified key. Apply modifiers like Control, Shift, Alt before the key press as specified in the fields Modifier One and Modifier Two. ▪ Key Down- Performs press of a single key specified on the keyboard. ▪ Key Up- Brings the pressed key back to normal position. It must be performed after the Key Down action to release the pressed key.
Interaction	The corresponding interaction set against the selected action.
Appears if Action selected is Wait Until Exists or Wait Until Disappears .	
Timeout (ms)	The wait time for the plugin until image is found or identified that it does not exists.
Poll Frequency (ms)	The frequency of checking the availability of the image and if the system is ready for the next action.
Appears if Action selected is Click And Type or Type Text .	
Input Source	The input value that must be entered. You must define a parameter in the Parameter bar to use this option.

Property Name	Usage
	Select the check box beside Input Source and enter the value, if you want to set a default value.
Appears if Action selected is Type Text .	
Delay In Characters (ms)	Triggers the delay when there is typing lag from the keyboard.
Appears if Action selected is Drag .	
Drag X	The vertical drag performed on the application window to drag the draggable image. The value must be entered in pixels.
Drag Y	The horizontal drag performed on the application window to drag the draggable image. The value must be entered in pixels.
Appears if Action selected is Horizontal Scroll or Vertical Scroll .	
Scroll Amount	The scroll speed (set in pixels) for each movement while searching for the image in the specified scrollable area.
Scroll Direction	The direction of scroll for searching the image. Available options for Horizontal Scroll are Left and Right and for Vertical Scrolls are Up and Down .
Appears if Action selected is Press Keys, Key Down or Key up .	
Key	The single key set as input from the keyboard.
Appears if Action selected is Press Keys .	
Modifier One	Select the first modifier to create the key combination. Available options are - Control, Shift, Alt and Win (Windows) keys of the keyboard.
Modifier Two	Select the second modifier to create the key combination. Available options are- Control, Shift, Alt and Win (Windows) keys of the keyboard.
Field Properties	
Variable Name	User defined name of the variable that stores the captured value. This field appears only for some of the selected actions.
Display Name	User specified display name of the action configured.
Variable Type	Type of the variable defined. This field appears only for some of the selected actions.
Misc	
Focus Required	Signifies if the Focus Window activity needs to be added before the interaction that gets created in Automation Studio from the multimodal interface. It is typically used if the configurations are

Property Name	Usage
	done on multiple windows and the intended window needs to be in focus for image automation to work. By default, this field is cleared.
OCR Properties	
This section appears only when Action selected is Get Text . This section is applicable only for text extracting functionality.	
Modes	<p>The mode to capture the image for text extraction. Available options are:</p> <ul style="list-style-type: none"> ▪ Fixed- When the image is captured in this mode, the coordinates of the selected area are stored to extract the text and not the image. ▪ Reference- When the image is captured in this mode, two different areas are captured. The first image captured is stored as a reference or a template, while the second image captured stores the coordinates from where the text is extracted. <p>By default, Fixed mode is set. You can change the mode as per the requirement.</p> <div style="background-color: #0070c0; color: white; padding: 5px;"> <p>Note: Once you change the mode, you need to capture the image again using the  (Recapture) icon to extract the text.</p> </div>
Engine	The OCR engine used for text extraction. By default, the inbuilt OCR engine is Tesseract . You can change the value in the Application activity. Refer Text Extractor activity to know more.

Web Applications

This activity allows you to automate the redundant and tedious task performed on a web application like log in to the website and validate information, monitoring a website to prevent downtime, web data extraction to consolidate information or update entries on online form and so on.

Web application using browsers, Google Chrome, Internet Explorer and Firefox can be automated using this activity. Compatible driver version must be present on the system where automation is performed to avoid potential errors.

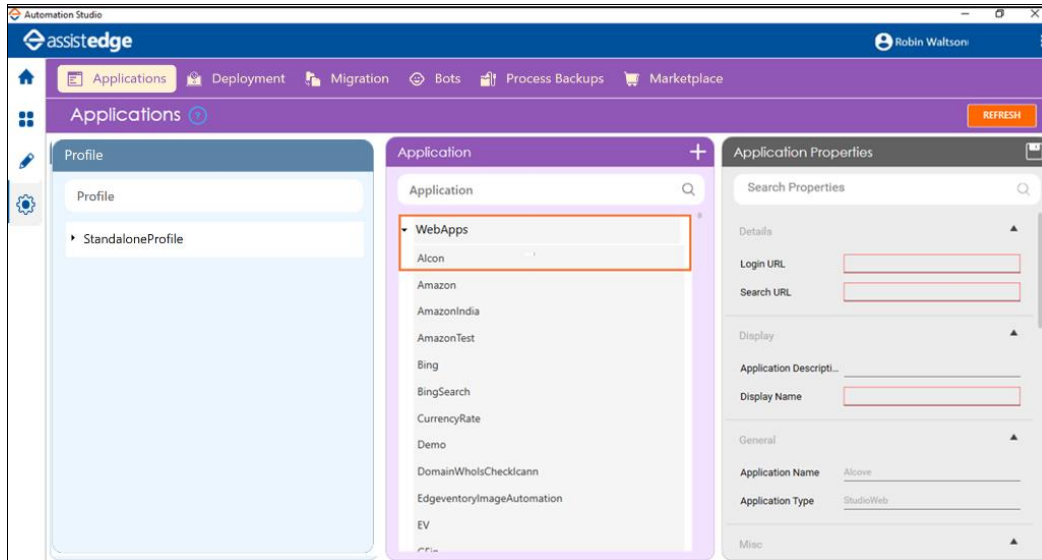
Prerequisite: Configuring Web Application


Note: Internet Explorer (IE) applications with document mode 5/7/8 are supported from the IE plugin. Selenium based plugin InternetExplorerSelenium uses the IE web driver to automate the web applications running in IE.

You must configure the web application in Automation Studio before you start configuring the steps of automation process workflow. This establishes the connection between the web application and Automation Studio to perform the automation.

Below is the minimum required properties for configuring the intended web application. If you want to define remaining properties, refer [WebApps](#) in the Admin Menu.

1. In the **Admin** menu, add an application of **Application Type- WebApps**.

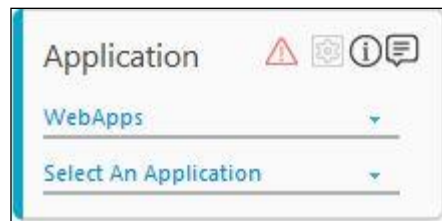


2. In the Application Properties panel:
 - In the **Login URL** field, enter the login URL of the web application you want to access.
 - In the **Search URL** field, enter the URL of the page where you want to perform the automation post login.
 - In the **Display Name**, enter a desired name of the web application.
3. Click the  (**Save Properties**) icon to save the application.

The WebApps application is configured.


Using Web Application Activity

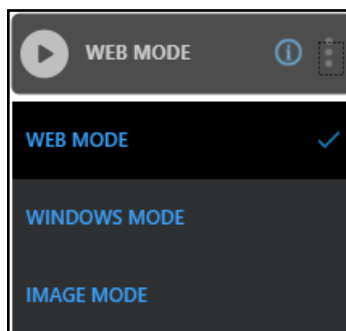
4. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
5. Drag the **Application** activity and drop on to the Flowchart designer on the Canvas





6. In the Application Type list, select **WebApps**. You must have at least one application added for it to appear in the list.
7. In the Select An Application list, select the application you want to perform automation on. Alternatively, you can add a new application at this point of time. To add web application:

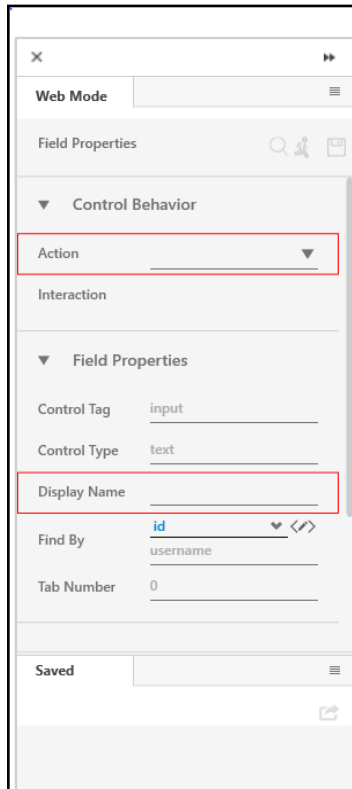
- a. In the Select An Application list, click **Add New Application**. The **Add New Application** dialog box appears.

- b. In the **Application Name** field, enter a desired name of the application.
- c. In the **LoginURL** field, enter the login URL of the web application you want to access.
- d. In the **SearchURL** field, enter the URL of the page that appears immediately post login.
- e. In the **PreferredBrowser** list, select the browser you prefer to launch the web application. By default, preference is set to **InternetExplorer**. Available options are:
- InternetExplorer
 - FireFox
 - Chrome
 - InternetExplorerSelenium
 - MicrosoftEdge
- f. Click **SAVE**. The web application is added.
8. Click the  (Settings) icon to launch the WEB MODE of the Multimodal interface.





9. Click the  (Play) icon to capture the steps that you want to perform within the web application. The application opens and the Play icon changes to the  (Recording) icon.

10. Hover over the field that you want to capture. The field gets highlighted with red box. Click the highlighted area.



The Field Properties panel of the Web Mode appears. The fields that are mandatory are highlighted with red box.

11. In the Action list, select the action that you want to perform on the captured area. Other mandatory fields change depending on your selection from the drop-down list.
12. In the **Display Name** field, enter a desired name of your choice.
13. Enter details of all the mandatory fields and other relevant fields as per your requirement. Refer [Field Properties](#) table to know more about the available fields and their respective properties.
14. Click the  (Save) icon to configure the fields.
15. Repeat step 6 through 10 to capture all the steps involved in the process. The configured field is saved and starts appearing in the Saved pane of the Field Properties panel. You can edit or delete these steps in the Saved pane, if required.
16. Click the  (Save configured field and return to studio) icon to save the field that you configured and return to the Studio menu.

The fields are configured and the Application activity for Application Type- WebApps is created.

Note: In case the field configuration does not work due to any security constraints in system, change the value of AutomationWay key to JAVA_OPTION in the AutomationStudio.exe.config file. You must update the same value in the Engage.exe.config and RoboSE.exe.config file as well.

Web Table Extraction


A web table in Automation Studio can be extracted based on the type of cells present in the table or the grid- heterogeneous cells (rows and columns with same html tagging) and non-heterogeneous cells (rows and columns with different html tagging).

- Extracting Nonhomogeneous cells- This method supports extraction of the entire table as well as specific rows or columns. It uses loop to extract data from the table. See the [example](#) to understand in detail.
- Extracting Homogeneous Cells- This method supports extraction of an entire table. It cannot extract specific row or column and does not use loop to extract data. See the [example](#) to understand in detail.

Web Mode Control Behavior and Field Descriptions

The properties of Web Mode are listed in the following table:

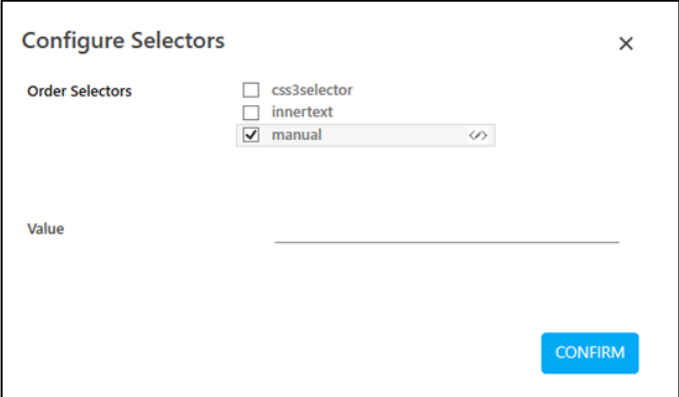
Property Name	Usage
Control Behavior	
Action	<p>It is the action or the interaction that can be performed on a web based application depending on the type of UI element as per the requirement. Available options are:</p> <ul style="list-style-type: none"> ▪ Control Exists- Checks if the UI element exists at the specified area. It is useful when working with dynamics controls. ▪ Set Value- Sets the user provided value of the UI element attribute and stores it as the value of the parameter. You must define the parameter in the Parameter bar before you start capturing the element. ▪ Set Attribute- Sets the user provided attribute for the specified UI element and stores the provided value of the attribute into a parameter. You must define the parameter in the Parameter bar before you start capturing the element. ▪ Get Value- Extracts the current value of the selected UI element such as input, select, textbox. ▪ Get Attribute- Extracts the attribute's value of the selected UI element. ▪ Click- Allows to left click the selected UI element. ▪ Click NoWait- Allows to click the selected UI element and move to the next activity without waiting for the page to load. ▪ Double Click- Performs double click on the selected UI element. ▪ MouseOver- Hovers the mouse over the selected UI element. ▪ SendKeys- Captures input of the selected UI element from the keyboard. The input can be a combination of keys or text input from the keyboard. Use this option when a keystroke needs to be automated on a specific web element and not on the overall application. ▪ Wait Until Exists- Waits until the UI element is found. You can use this to


Property Name	Usage
	<p>avoid any delay in the page loading time. This interaction returns a Boolean value of true or false (on success or failure respectively) on finding the image. The Boolean value can be set to a variable and checked in a conditional statement to perform an appropriate action after finding the web element.</p> <ul style="list-style-type: none"> ▪ Scroll By- Scrolls to particular section of the web page horizontally or vertically. ▪ Scroll Into View- Scrolls to bring the application area viewable on screen of the web browser. ▪ Shift Click- It appears only when you want to capture action on a button or a link. this interaction opens the URL associated with the selected link or the button, in a new window.
Interaction	The corresponding interaction set against the selected action.
Appears if Action selected is SendKeys.	
Keys	<p>The key sent as input from the keyboard. It can be a combination of keys or a text.</p> <p>To configure the keys:</p> <ol style="list-style-type: none"> 1. Once Action selected is SendKeys, click the  (Configure) icon. The SendKey 2. Configuration dialog box appears. <div data-bbox="625 1123 1274 1554" style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <p style="text-align: center;">SendKey Configuration ×</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>Select Function Keys(Max 3)</p> <p>Input Key (optional)</p> </div> <div style="width: 45%;"> <p style="text-align: center;">Key Combination</p> <ul style="list-style-type: none"> <input type="checkbox"/> Shift <input type="checkbox"/> Control <input type="checkbox"/> Alt <input type="checkbox"/> Backspace <input type="checkbox"/> Delete <input type="checkbox"/> ArrowUp <input type="checkbox"/> ArrowDown </div> </div> <div style="text-align: right; margin-top: 10px;"> <p style="background-color: #ADD8E6; padding: 5px 10px; border: 1px solid #ADD8E6;">CONFIRM</p> </div> </div> <ol style="list-style-type: none"> 3. In the Select Function Keys (Max 3) list, select the keys from the keyboard to create the key combination. You can select up to maximum of three keys. 4. In the Input Key(optional) filed, enter an alphabetical or a numeric key to create the combination. This is an optional field. 5. Click CONFIRM. <p>The configuration for SendKeys action is done.</p>
Appears if Action selected is Get Attribute and Set Attribute.	
Attribute	Sub attributes or the properties of the html attribute selected.

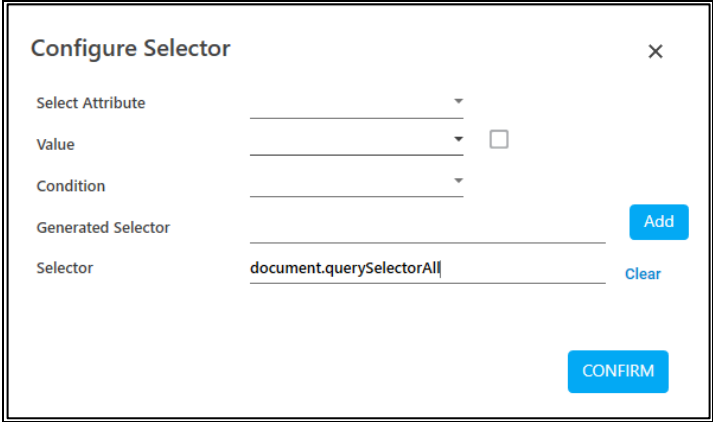
Property Name	Usage
Appears if Action selected is Set Value, Set Attribute.	
Input Source	The input value that must be provided. You must define a parameter in the Parameter bar to use this option. Select the checkbox beside Input Source , and enter the input value, if you want to set a default value.
Appears if Action selected is Scroll By.	
XAxis	Vertical scroll on the web page. The value must be entered in pixels.
YAxis	Horizontal scroll on the web page. The value must be entered in pixels.
Appears if Action selected is Scroll Into View.	
Scroll To	Determines the area till where the scroll must be done. Available options are: <ul style="list-style-type: none"> Start Center End <div style="background-color: #0070c0; color: white; padding: 5px; margin-top: 10px;"> Note: Internet Explorer does not support Centre option, so configuring center for ScrollTo displays exception at runtime. </div>
Dock Window	
This section appears only when Action selected is Shift Click.	
Dock Action	When a link or button on the web page is opened in a new browser window, the Dock Action button enables the new window listed in the multimodal UI. You can select one of the windows in the multimodal UI to configure controls on the intended window.
Interaction Attributes	
This section appears only when Action selected is Shift Click.	
Tab Name	Name of the new window when it is docked in Engage application.
Target Url	URL of the new window that opens when Shift Click is selected in Action .
Use Mouse	Performs left click of the mouse on the link or the button to open in the new browser window.
Field Properties	
Control Tag	User specified identifier of the selected UI element.
Control Type	Control type of the UI element as defined by the developer.
Display Name	User specified display name of the action configured.

Property Name	Usage
Find By	<p>Locator of the UI element on a web application.</p> <p>Identifying the correct element forms one of the first step towards creating an automation workflow. This list changes depending on the properties of the UI elements defined. Commonly available options are:</p> <div data-bbox="813 476 1078 663" style="border: 1px solid black; padding: 5px; margin: 10px auto; width: fit-content;"> </div> <ul style="list-style-type: none"> ▪ Id - Identifier of the UI element. It is useful for locating elements that has a unique Id associated with it. ▪ name - Field name of the control. Each element on the web page has a name assigned to it which can be used to select the intended element. ▪ css3selector - Part of the CSS rule that uses selectors such as tags, class, pseudo-element, pseudo-class, combination of selectors and other wide range of selectors to identify the UI element. ▪ css3xpath - Depending on the html tagging, css3xpath can also be available in this list. It uses the Xpath expression to locate the UI element. ▪ manual - Provides manual way of identifying the UI element. You can use this option when the selected set of properties available for the UI element are not able to identify the required it at automation runtime. <p>You can configure backup selectors to avoid situations where the automation process is unable to find the primary selector. This is not a mandatory step.</p> <p>Follow the instructions to configure the backup selectors:</p> <ol style="list-style-type: none"> 1. Click the (Configure) icon. The Configure Selectors dialog page appears. <div data-bbox="613 1455 1279 1845" style="border: 1px solid black; padding: 10px; margin: 10px auto; width: fit-content;"> </div> <ol style="list-style-type: none"> 2. Select the additional selectors. <ul style="list-style-type: none"> ▪ If the selector is manual:

Property Name	Usage
---------------	-------



- a. Click the  (pencil) icon to configure the manual selector. The **Configure Selector** dialog box appears.



- b. In the **Select Attribute** list, select the attribute of the manual selector. As per the HTML structure and the properties of the UI element defined, they are displayed in this list.
 - c. In the **Value** list, either select a parameter containing the value of the selected attribute or select the check box to set a default value.
 - d. In the **Condition** list, select the condition of the selector. Available options are- **Equals, Contains, Starts With** and **Ends With**.
 - e. The **Generated Selector** field displays the default value generated, if any. You can click **Add** to add this value in the **Selector** field.
 - f. Click **CONFIRM**.
3. In the **Value** field, the captured selector is displayed.
 4. You can drag the selector and drop it to change the sequence as per the priority of the selector you want to set. At the time of process execution, the selector is considered as per the sequence set, the top most having the highest priority while the last one having the least priority for locating the UI element.

Property Name	Usage
	5. Click CONFIRM to save the configuration.
Appears if Action selected is Control Exists, Get Attribute .	
Tab Number	The window tab on which you want to perform automation.
Appears if Action selected is Control Exists, Get Attribute .	
Variable Name	User defined name of the variable that stores the captured value. You can define the variable name here itself or predefine it in the Parameter bar
Appears if Action selected is Control Exists, Get Attribute	
Variable Type	Type of the variable defined. Available options are: <ul style="list-style-type: none"> ▪ Text - A sequence of character, either as a constant or a variable. ▪ Numeric - An integer type that range from negative through positive. ▪ Decimal - An exact numeric value defined by its precision and scale. ▪ Boolean - A data type used for making decision. Can have only two values- true or false. Data Table - A data type that stores tabular data.
Misc	
FrameNo	Number of the frame of the web page where you want to perform the action. Automation Studio automatically detects the frame number based on the area selected during configuration.
Item Index	Index number of the selected item on the web page.
Item Offset	The distance between the target item and the anchor.

Web Application Properties

The properties of a web application are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	

Property Name	Usage
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as the name of the selected application. You can change the name as required.

Example of Web Application Automation

Following are some of the examples to demonstrate web application automation.

Automating a Search Process in a Web Application


Configure a search process for information on a web application and updates the search result.

In this example, consider web page of Rediff Money where we automate the process of retrieving the real time stock price of a desired company.

Prerequisite:



1. In the **Admin** menu, add an application of **Application Type- WebApps**.
2. In the Application Properties panel:
 - Enter the **Login URL** as <https://money.rediff.com/index.html>
 - Enter the **Search URL** as <https://money.rediff.com/index.html>

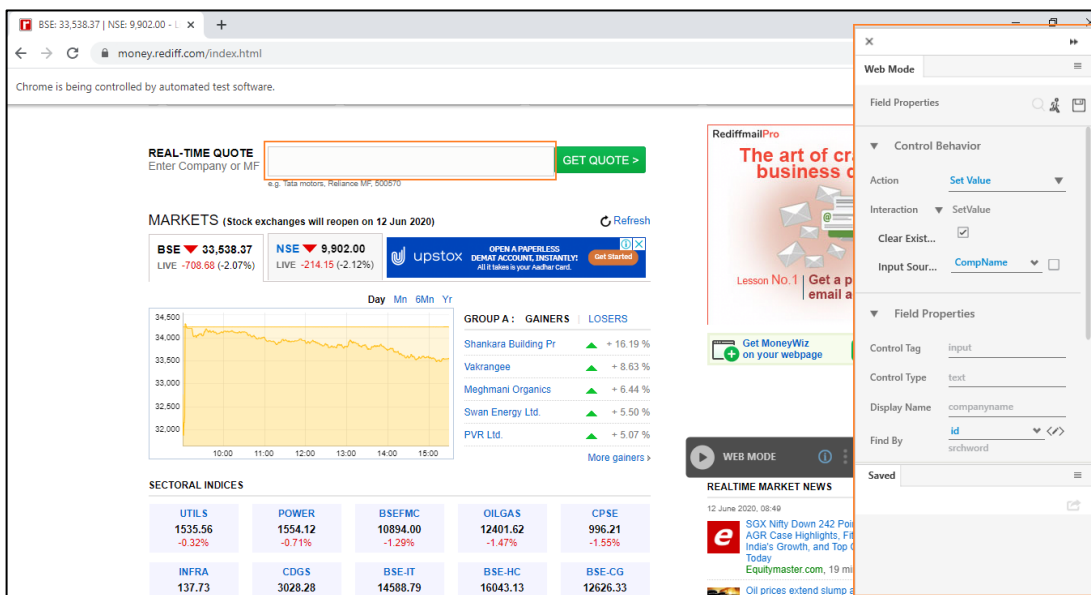
Note: Since this application does not require login, the login and the search URL remains same. However, it changes if they are different.



- Enter the Display **Name**.
3. Click the  (**Save Properties**) icon to save the application details.
 4. Enter other details as per your requirement.

The WebApps application is created.

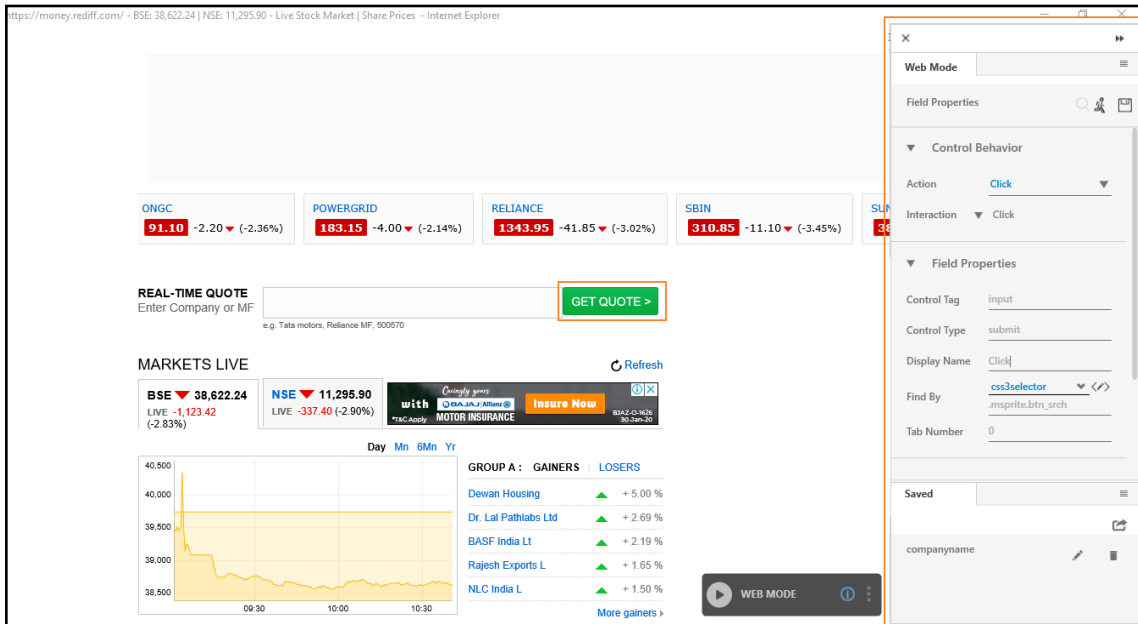
To automate the process of retrieving real time stock price of the company:

5. Create a new process.
6. Create string arguments, **CompName** and **StockPrice** to enter the name of the company and store the real time stock price of the same.
7. From the Canvas Tools panel, add Application activity to the Flowchart designer on the Canvas
8. In the **Application Type** list, select **WebApps**.
9. In the **Select an Application** list, select the available web application that you configured.
10. Click the  (**Settings**) icon to launch the WEB MODE of the Multimodal interface. The intended web page appears.
11. Click the  (Play) icon to capture the steps of entering the company name on the web page. Hover over the **Enter Company or MF** field. The field gets highlighted with red box.




12. Click the highlighted area. The Field Properties panel of the Web Mode appears. The fields that are mandatory are highlighted with red box.
13. In the Action list, select **Set Value**.
14. In the **Input Source** list, select the **CompName** argument.
15. Enter a **Display Name**, and then click the  (**Save**) icon to configure the details captured. The saved details start appearing in the **Saved** tab of **Field Properties** bar.
16. Click the  (Play) icon to capture the step of clicking the button to retrieve the real-time stock price of the entered company name.


17. Hover over the **GET QUOTE>** field. The field gets highlighted with red box.



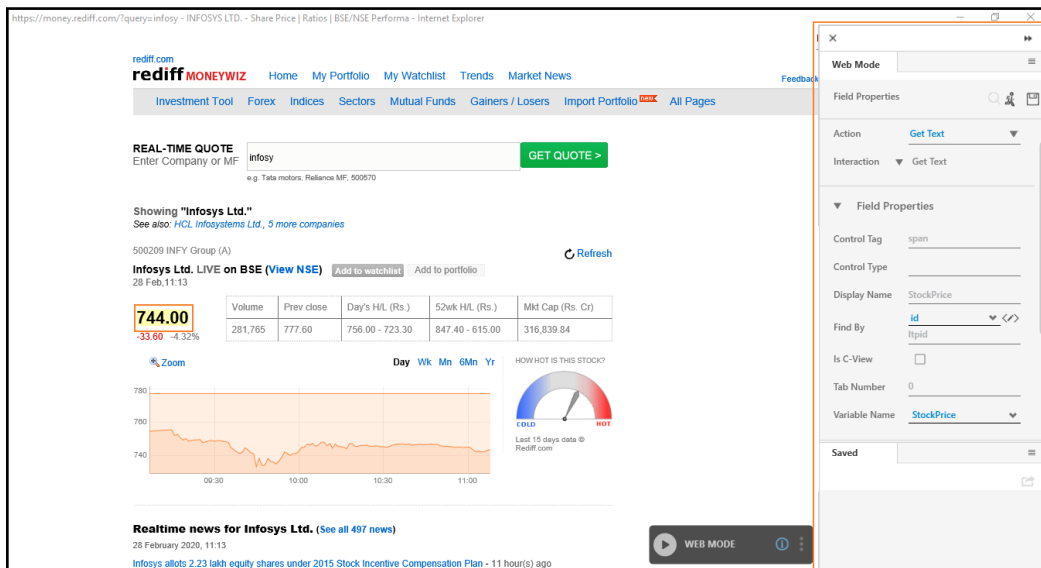
18. Click the highlighted area. The Field Properties panel of the Web Mode of the web application appears. The fields that are mandatory are highlighted with red box.




19. In the **Action** list, select **Click**.

20. Enter a **Display Name** and then click the  (Save) icon to configure the details captured. The saved details start appearing in the **Saved** tab of **Field Properties** bar.

21. Click the  (Play) icon to capture the step of getting the stock price displayed. Hover over the stock price displayed. The area gets highlighted with red box.

22. Click the highlighted area. The Field Properties panel of the Web Mode appears. The fields that are mandatory are highlighted with red box.



23. In the **Action** list, select **Get Text**. You can click the  (**Configure**) icon to configure the backup selectors for scenarios where primary selector does not work.
24. In the **Variable Name** list, select **StockPrice** and then click the  (**Save**) icon to configure the details captured. The saved details start appearing in the **Saved** tab of **Field Properties** bar.
25. Click the  (Save configured field and return to studio) icon to save the field that you configured and return to the Studio menu. The automation process workflow to configure the search process is created.
26. In the **Studio** menu, setup the environment and then perform the test run.

The process returns the real time stock price of the company entered.


Automating Web Table Extraction Using Loop

In this example, we are extracting the stock indices and the corresponding % change from the BSE INDICES table available on the Rediff Money web page.

Prerequisite:

1. In the **Admin** menu, add an application of **Application Type- WebApps**.
2. In the Application Properties panel:
 - Enter the **Login URL** as <https://money.rediff.com/index.html>
 - Enter the **Search URL** as <https://money.rediff.com/index.html>


Note: Since this application does not require login, the login and the search URL remains same. However, it changes if they are different.

- Enter the Display Name.
3. Click the  (**Save Properties**) icon to save the application details


The WebApps application is created.

To automate retrieving entire data from a table:

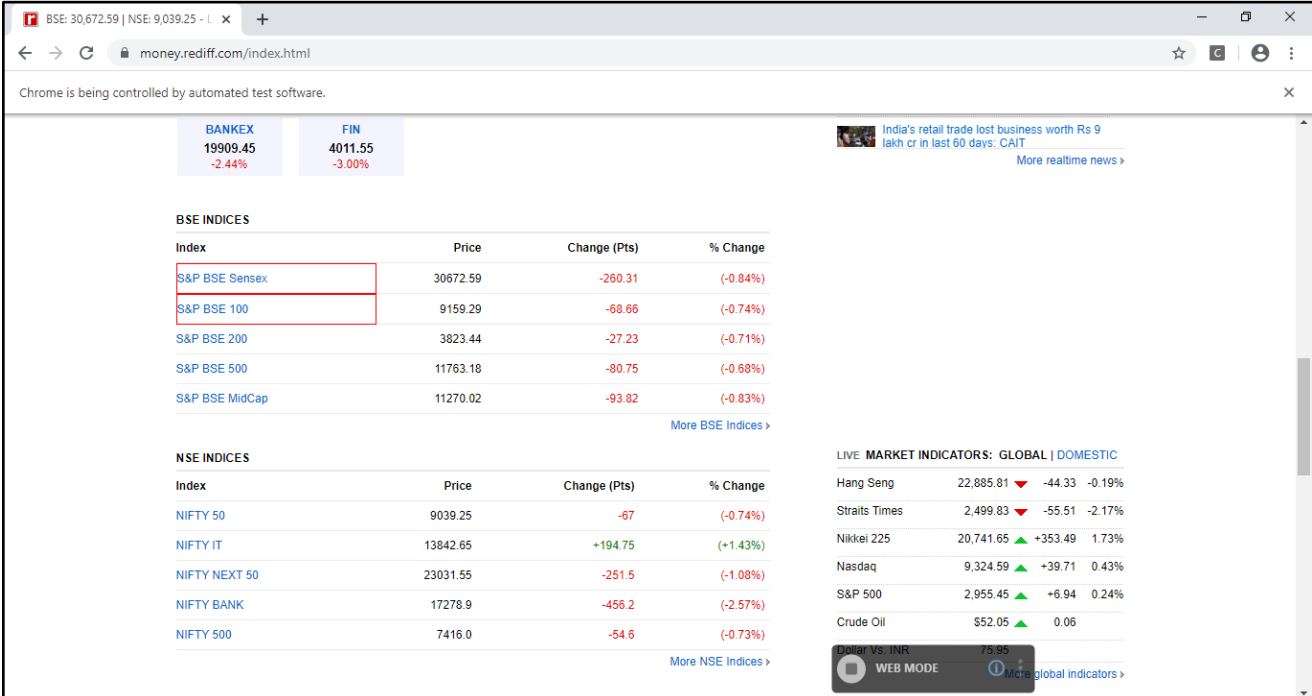
1. Create a new process.
2. Create the following **In** arguments:
 - a. **CompName** and **PercentageChange** of **String** type to store the company name and the corresponding percentage change extracted.
 - b. Create another argument, **RowExists** of **Boolean** type to check if the table row exists.
 - c. Create **RowIndex** of **Int32** type to specify the index of the row. It indicates the index of the row where the action must be performed. Set **RowIndex** as **0** to start data extraction from the first row of the table.
3. From the Canvas Tools panel, add Application activity to the Flowchart designer on the Canvas
4. In the **Application Type** list, select **WebApps**.

5. In the **Select an Application** list, select the available web application that you configured.
6. Click the  (**Settings**) icon to launch the WEB MODE of the Multimodal interface. The **Rediff Money** page opens. The steps involved in the extraction of data from a web table are:
 - [Indicate the system to check if intended row exists](#)
 - [Verify if intended row exists](#)
 - [Capture the first step of data extraction](#)
 - [Perform iteration over the first step of data extraction using While loop](#)

Indicate the System to Check If Intended Row Exists

- a. Locate the required table and then click the  (Play) icon.
- b. Press **Alt** key from the keyboard and select two consecutive rows within same column of the table. Click the highlighted area to indicate the column from where you want to check if the row exists for data extraction. The entire column gets highlighted and the **Field Properties** pane appears.

Note: The process of extracting data from columns is same as the process of extracting data from rows. You need to select two consecutive columns within same row of the table.



The screenshot displays a web browser window with the URL `money.rediff.com/index.html`. The page content includes market data for BANKEX (19909.45, -2.44%) and FIN (4011.55, -3.00%). Below this, there are two tables: 'BSE INDICES' and 'NSE INDICES'. The 'BSE INDICES' table has columns for Index, Price, Change (Pts), and % Change. The 'S&P BSE Sensex' row is highlighted with a red box. The 'NSE INDICES' table also has columns for Index, Price, Change (Pts), and % Change. To the right, there is a 'LIVE MARKET INDICATORS' section with a sub-section for 'GLOBAL | DOMESTIC'. This section lists various indices and their values, including Hang Seng, Straits Times, Nikkei 225, Nasdaq, S&P 500, and Crude Oil. At the bottom right, there is a 'WEB MODE' overlay with a play button and a 'More global indicators >' link.

Index	Price	Change (Pts)	% Change
S&P BSE Sensex	30672.59	-260.31	(-0.84%)
S&P BSE 100	9159.29	-68.66	(-0.74%)
S&P BSE 200	3823.44	-27.23	(-0.71%)
S&P BSE 500	11763.18	-80.75	(-0.68%)
S&P BSE MidCap	11270.02	-93.82	(-0.83%)

Index	Price	Change (Pts)	% Change
NIFTY 50	9039.25	-67	(-0.74%)
NIFTY IT	13842.65	+194.75	(+1.43%)
NIFTY NEXT 50	23031.55	-251.5	(-1.08%)
NIFTY BANK	17278.9	-456.2	(-2.57%)
NIFTY 500	7416.0	-54.6	(-0.73%)

Indicator	Value	Change	% Change
Hang Seng	22,885.81	-44.33	-0.19%
Straits Times	2,499.83	-55.51	-2.17%
Nikkei 225	20,741.65	+353.49	1.73%
Nasdaq	9,324.59	+39.71	0.43%
S&P 500	2,955.45	+6.94	0.24%
Crude Oil	\$52.05	0.06	

Verify if Intended Row Exists

- In the **Action** list, select **Control Exists**. This interaction checks if the intended row of the required table exists.

The screenshot shows a web browser window displaying a financial website with market indices. The 'Control Exists' configuration panel is open on the right side of the browser window.

Web Page Content:

BANKEKX: 19909.45 (-2.44%)
FIN: 4011.55 (-3.00%)

BSE INDICES


Index	Price	Change (Pts)	% Change
S&P BSE Sensex	30672.59	-260.31	(-0.84%)
S&P BSE 100	9159.29	-68.66	(-0.74%)
S&P BSE 200	3823.44	-27.23	(-0.71%)
S&P BSE 500	11763.18	-80.75	(-0.68%)
S&P BSE MidCap	11270.02	-93.82	(-0.83%)

NSE INDICES


Index	Price	Change (Pts)	% Change
NIFTY 50	9039.25	-67	(-0.74%)
NIFTY IT	13842.65	+194.75	(+1.43%)
NIFTY NEXT 50	23031.55	-251.5	(-1.08%)
NIFTY BANK	17278.9	-456.2	(-2.57%)
NIFTY 500	7416.0	-54.6	(-0.73%)

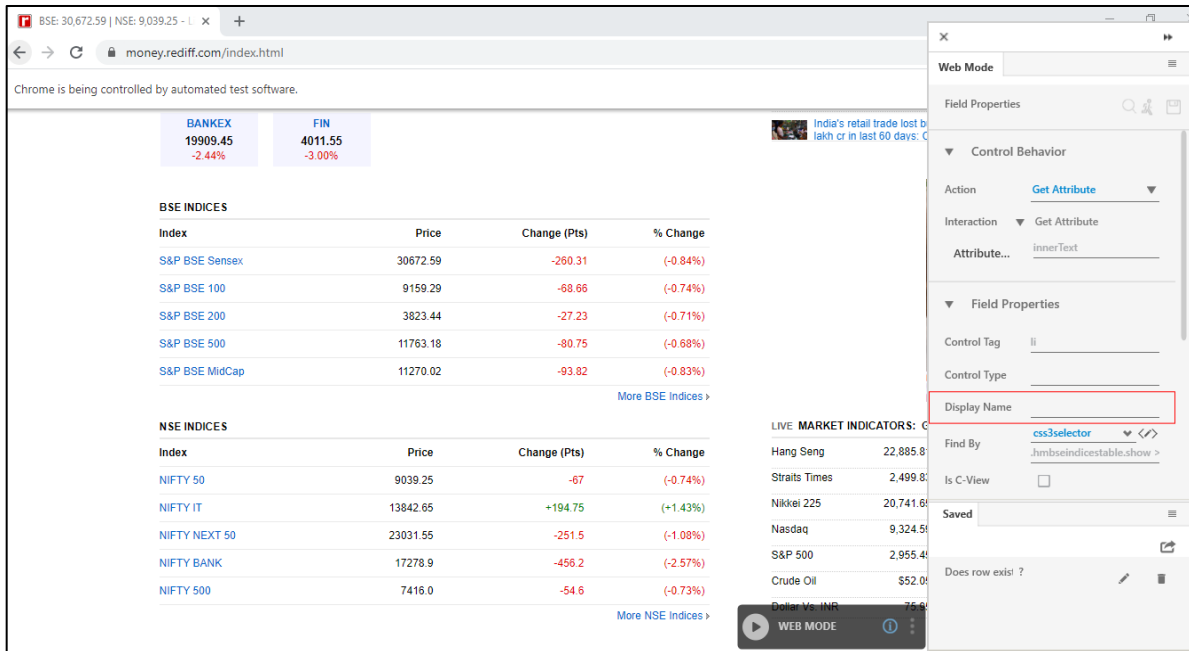
Configuration Panel (Control Exists):

- Action: Control Exists
- Interaction: ControlExistsInteraction
- Field Properties:
 - Control Tag: li
 - Control Type: [Empty]
 - Display Name: Does row exist in the table?
 - Find By: cssselector
 - Find By Value: .hmbseindicestable.show
 - Is C-View: [Unchecked]
 - Tab Number: 0
 - Variable Name: RowExists
- Saved: [Save Icon]

- In the **Display Name** field, enter a desired name.
- In the **Variable Name** list, select **RowExists** argument created in the **Parameter** bar.
- In the **Item Index** list, select **RowIndex** argument created in the **Parameter** bar. This provides the index of the element for extraction. Other fields are auto-populated based on the data selected.
- Click the  (**Save**) icon to configure the details captured. The saved details start appearing in the **Saved** tab of **Field Properties** bar.

Capture First Step of Data Extraction from Intended Row




- Click the  (Play) icon to automate the step of extraction the name of the company.
- Press **Alt** key from the keyboard and select two consecutive rows of the **Index** column. Click the highlighted area to indicate the column to extract data. The entire column gets highlighted and the **Field Properties** pane appears.




The screenshot shows a web browser window displaying stock market data. The browser address bar shows "money.rediff.com/index.html". The page content includes "BSE INDICES" and "NSE INDICES" tables. The "Field Properties" pane is open on the right, showing the "Action" set to "Get Attribute" and the "Display Name" field highlighted in red. The "Find By" field contains the CSS selector ".hmbseindicestable.show".

Index	Price	Change (Pts)	% Change
S&P BSE Sensex	30672.59	-260.31	(-0.84%)
S&P BSE 100	9159.29	-68.66	(-0.74%)
S&P BSE 200	3823.44	-27.23	(-0.71%)
S&P BSE 500	11763.18	-80.75	(-0.68%)
S&P BSE MidCap	11270.02	-93.82	(-0.83%)

Index	Price	Change (Pts)	% Change
NIFTY 50	9039.25	-67	(-0.74%)
NIFTY IT	13842.65	+194.75	(+1.43%)
NIFTY NEXT 50	23031.55	-251.5	(-1.08%)
NIFTY BANK	17278.9	-456.2	(-2.57%)
NIFTY 500	7416.0	-54.6	(-0.73%)

- In the **Action** list, select **Get Attribute** to fetch the name of the stock indices.
- In the **Display Name** field, enter a desired name.
- In the **Variable Name** list, select **CompName** argument created in the **Parameter** bar.
- In the **Item Index** list, select **RowIndex** argument created in the **Parameter** bar. This provides the index of the element for extraction. Other fields are auto-populated based on the data selected.
- Click the  (**Save**) icon to configure the details captured. The saved details, of extracting company name, starts appearing in the **Saved** tab of **Field Properties** bar.
- Click the  (Play) icon to capture data extraction of percentage change.
- Press **Alt** key from the keyboard and select two consecutive rows of the **% Change** column. Click the highlighted area to indicate the column to extract data. The entire column gets highlighted and the **Field Properties** pane appears.
- In the **Action** list, select **Get Attribute** to fetch the name of the company.
- In the **Display Name** field, enter a desired name.
- In the **Variable Name** list, select **PercentageChange** argument created in the **Parameter** bar.
- In the **Item Index** list, select **RowIndex** argument created in the **Parameter** bar. This provides the index of the element for extraction. Other fields are auto-populated based on the data selected.
- Click the  (**Save**) icon to configure the details captured. The saved details start appearing in the **Saved** tab of **Field**

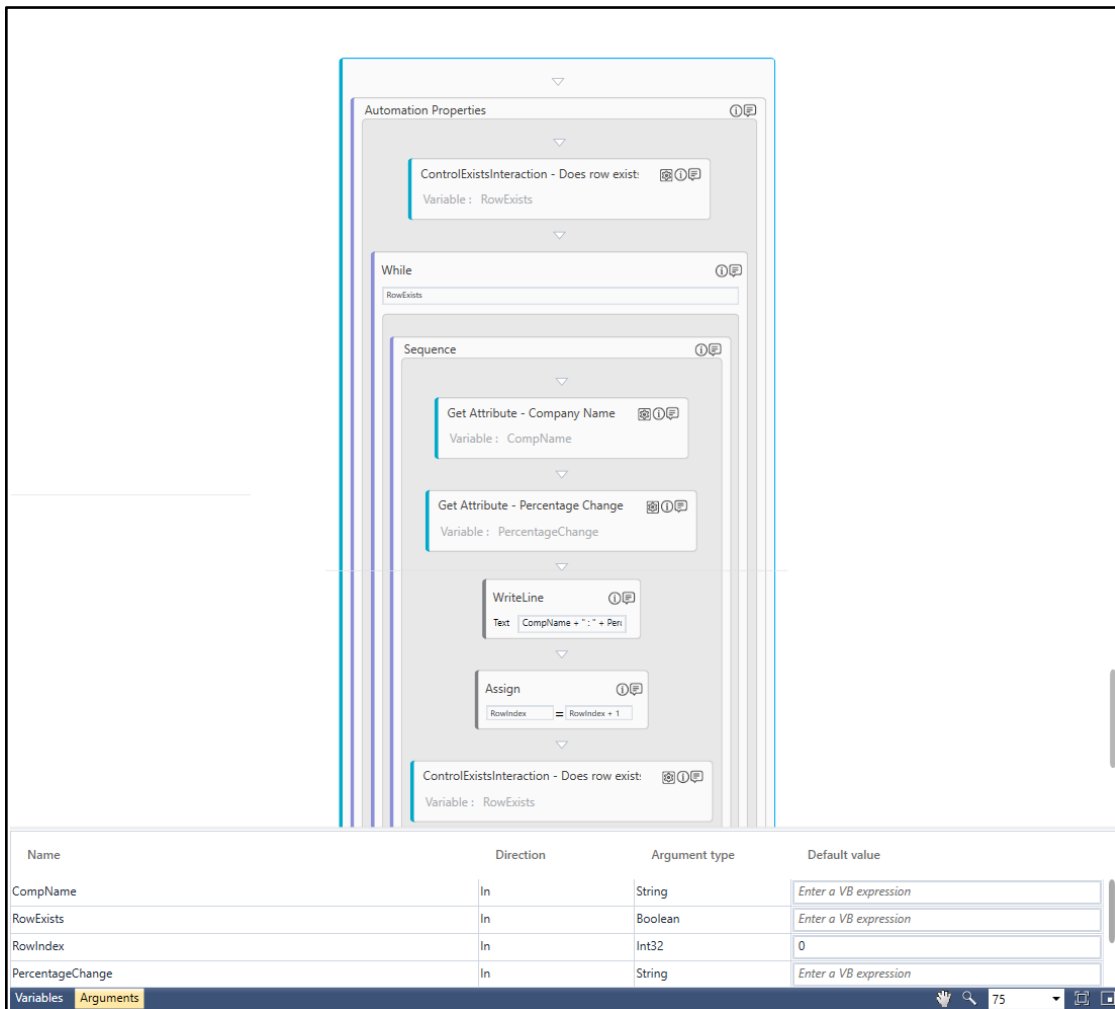
Properties bar.

- o. Click the  (Save configured field and return to Studio) icon to save the field that you configured and return to the Studio menu.

Perform Iteration Over First Step of Data Extraction Using While Loop

- a. Double click the **Application** activity and drop **While** activity below **ControlExistsInteraction** block.
- b. Move the captured interactions, [Capture First Step of Data Extraction](#) inside the **While** activity to create a loop over the steps captured.
- c. Set the while condition that until **RowExists** argument is true, extract the company name else break the loop. You can set a condition inside the **While** loop to break the loop if you want to extract data until a specified row.
- d. Add **Assign** activity and increment the value of **RowIndex** by **1** to let the loop advance to the next row index for data extraction.
- e. Copy and add the captured interaction, [Verify if Intended Row Exists](#) after the **Assign** activity to break the While loop after no row is found.

You can store the extracted data in an excel or use them further processing. To view the output in Automation Studio, let's add WriteLine activity. You can assign this process to a robot, if you want to execute this process outside Automation Studio.



7. Add a **WriteLine** activity below data extraction block and in the **Text** field, enter **CompName+ " : " +Percenatge** to print the company name and their corresponding percentage change.
8. Save the process.
9. Setup the environment and then perform test run. Following is the output displayed in the console:

```

Output

S&P BSE Sensex : (+1.94%)
S&P BSE 100 : (+1.92%)
S&P BSE 200 : (+1.83%)
S&P BSE 500 : (+1.82%)
S&P BSE MidCap : (+1.35%)
    
```

Note: If all the elements are not extracted even after successful execution of the process, use any of the available backup selectors to resolve this issue. See [Find By](#) field property for more details.


Automating Smart Google Web Toolkit (GWT) Based applications

In this example we are extracting data from a table that is present in a Smart GWT based application.

Prerequisite:



1. In the **Admin** menu, add an application of **Application Type- WebApps**.
2. In the Application Properties panel:
 - Enter the **Login URL** as <https://www.smartclient.com/smartclient/showcase/?id=adaptiveFilterFS>
 - Enter the **Search URL** as <https://www.smartclient.com/smartclient/showcase/?id=adaptiveFilterFS>

Note: Since this application does not require login, the login and the search URL remains same. However, it changes if they are different.

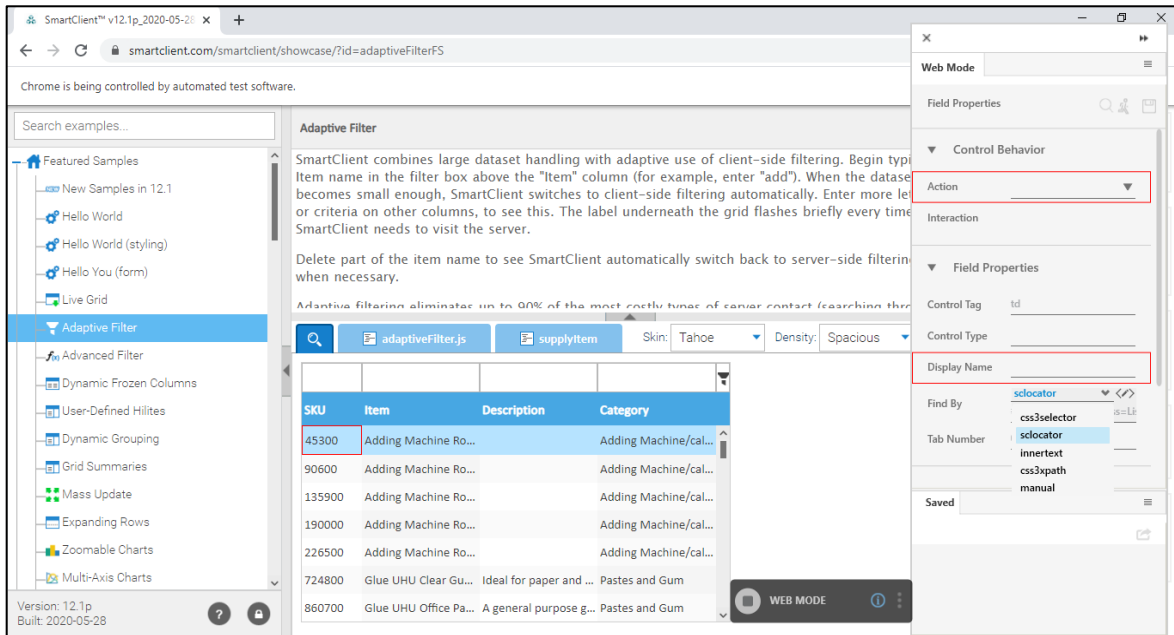
- Enter the **Display Name**.
3. Click the  (**Save Properties**) icon to save the application details.
 4. Enter other details as per your requirement.

The WebApps application is created.

To automate retrieving data from an entire table:

1. Create a new process.
2. From the Canvas Tools panel, add Application activity to the Flowchart designer on the Canvas
3. In the **Application Type** list, select **WebApps**.
4. In the **Select an Application** list, select the available Smart GWT application that you configured.
5. Click the  (**Settings**) icon to launch the WEB MODE of the Multimodal panel. The SmartClient page opens.
6. Locate the table and then click the  (Play) icon. To loop over the table and extract each row:
 - a. Extract the **SKU** column from the first row.

- b. In the **Find By** list, select **sclocator** and copy the generated code in a notepad file.



- c. Extract the **SKU** column from the second row.
- d. In the **Find By** list, select **sclocator** and copy the generated code below the first code in the same notepad file. Following is a sample of the generated values:

[Row - 1]

```
//testRoot[]/child[Class=ListGrid||index=0||length=2||classIndex=0||classLengt
h=1||roleIndex=0||roleLength=1||scRole=list]/body/row[itemID=1||itemName=Addin
g%20Machine%20Roll%2057x57mm%20Standard||SKU=45300||0]/col[fieldName=SKU||0]
```

[Row -2]

```
//testRoot[]/child[Class=ListGrid||index=0||length=2||classIndex=0||classLengt
h=1||roleIndex=0||roleLength=1||scRole=list]/body/row[itemID=2||itemName=Addin
g%20Machine%20Roll%2057x64mm%20Standard||SKU=90600||1]/col[fieldName=SKU||0]
```


The generated values of the sclocator contains row specific details for the row part, and rest of the details are the same for both the values as shown below:

[Row - 1]

```
row[itemID=1 || itemName=Adding%20Machine%20Roll%2057x57mm%20Standard || SKU=45300 |
| 0]
```

[Row -2]

```
row[itemID=2 || itemName=Adding%20Machine%20Roll%2057x64mm%20Standard || SKU=90600 |
| 1]
```

- e. Delete the specific details and add a placeholder for loop index to make it as a generic **sclocator** that is used for looping. In the above example, remove the specific information related to **itemName** and **SKU** fields. The **itemID** field and row identifier after the **SKU** field can be used for making placeholder for loop index.
- f. After deleting the specific information, use either of the following as a value of the **sclocator** in loop:

[Version - 1]- using itemID column

```
//testRoot[]/child[Class=ListGrid || index=0 || length=2 || classIndex=0 || classLength
=1 || roleIndex=0 || roleLength=1 || scRole=list]/body/row[itemID={0}]/col[fieldName=
SKU || 0]
```

[Version - 2]- using row identifier

```
//testRoot[]/child[Class=ListGrid || index=0 || length=2 || classIndex=0 || classLength
=1 || roleIndex=0 || roleLength=1 || scRole=list]/body/row[{0}]/col[fieldName=SKU | 0]
```

Note:

For Version 1, loop index must start from 1 or offset must be 1 Only the entire table or the grid can be extracted.

For Version 2, loop index must start from 0.

Automating Web Table Extraction Without Loop

In this example, we are extracting the BSE Sensex details from the BSE INDICES table available on the Rediff Money web page.

Note:


Only homogenous data cells (rows and columns with same html tagging) can be captured through this process.

Only the entire table or the grid can be extracted.

Prerequisite:

1. Goto **Admin** menu and in the **Application** tab, add a **WebApps** application in the Application
2. In the Application Properties panel:
 - Enter the **Login URL** as <https://money.rediff.com/index.html>
 - Enter the **Search URL** as <https://money.rediff.com/index.html>

Note: Since this application does not require login, the login and the search URL remains same. However it changes if they are different.


3. Enter name of the application in the Display **Name** field.
4. Click the  (**Save Properties**) icon to save the application details


The WebApps application is created.

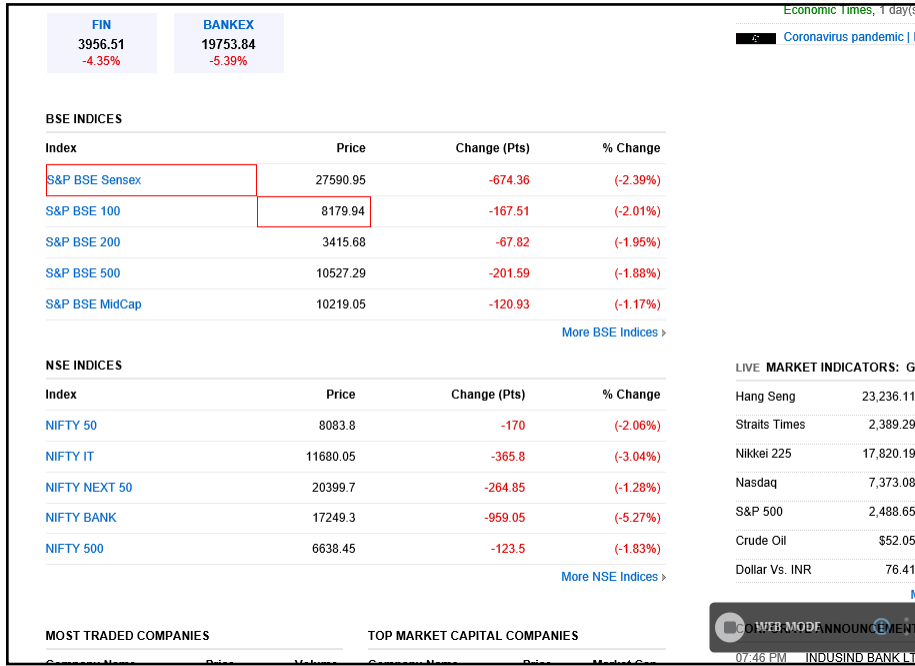
To automate retrieving details of top world market companies:

5. Create a new process.
6. Create an **Out** argument, **SensexTable** of type **DataTable** to store the datatable that gets created during the extraction. If you do not create an argument of **DataTable** type, it gets created at the time of field configuration and you can define the name.

Note: If the predefined argument of type Data Table is used, its structure is overwritten based on the web table extracted. Dummy headers are added to the Data Table namely Column1, Column2 and so on.

7. From the Canvas Tools panel, add Application activity to the Flowchart designer on the Canvas
8. In the **Application Type** list, select **WebApps**.
9. In the **Select an Application** list, select the available web application that you configured.
10. Click the  (**Settings**) icon to launch the WEB MODE of the Multimodal interface. The Rediff Money page appears.

- Click the  (Play) icon to capture the steps to automate the process of fetching the BSE Sensex details. Locate the required table, press shift and select the table cells diagonally (in this case header does not get selected as the html tagging is different for the header and rows/columns), and then click the highlighted area. The entire table gets highlighted and the **Field Properties** pane appears.



The screenshot shows a financial website with two main tables: BSE INDICES and NSE INDICES. The BSE INDICES table has the following data:

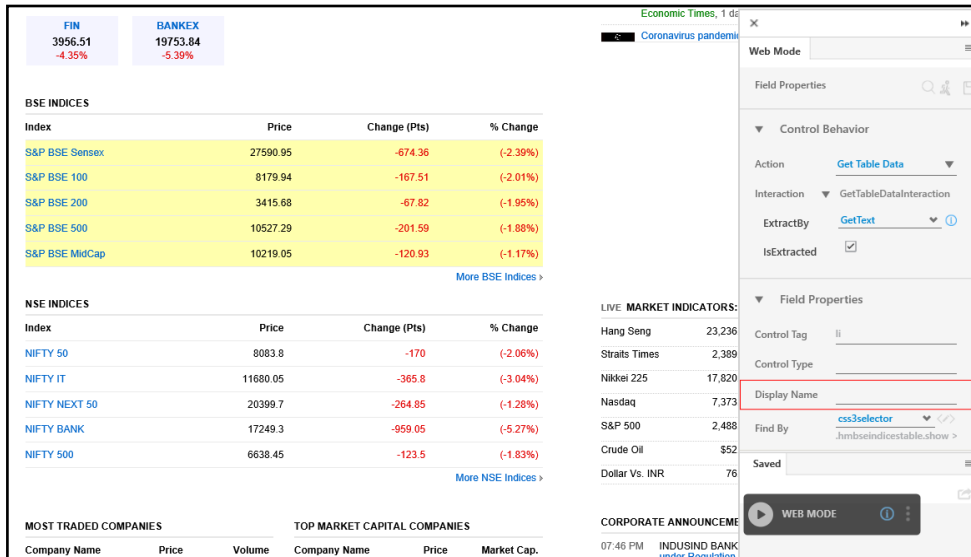
Index	Price	Change (Pts)	% Change
S&P BSE Sensex	27590.95	-674.36	(-2.39%)
S&P BSE 100	8179.94	-167.51	(-2.01%)
S&P BSE 200	3415.68	-67.82	(-1.95%)
S&P BSE 500	10527.29	-201.59	(-1.88%)
S&P BSE MidCap	10219.05	-120.93	(-1.17%)

The NSE INDICES table has the following data:

Index	Price	Change (Pts)	% Change
NIFTY 50	8083.8	-170	(-2.06%)
NIFTY IT	11680.05	-365.8	(-3.04%)
NIFTY NEXT 50	20399.7	-264.85	(-1.28%)
NIFTY BANK	17249.3	-959.05	(-5.27%)
NIFTY 500	6638.45	-123.5	(-1.83%)

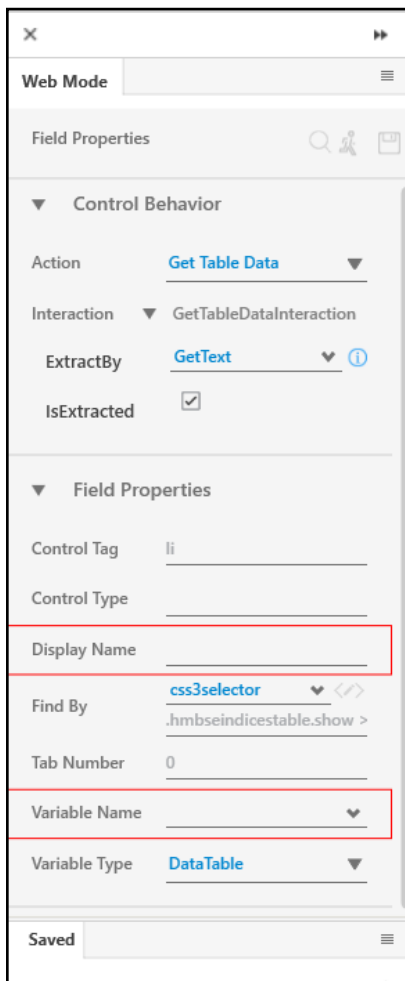
At the bottom of the page, there are sections for MOST TRADED COMPANIES and TOP MARKET CAPITAL COMPANIES. A 'WEB MODE' button is visible in the bottom right corner.

- In the **Action** list, **Get Table Data** is selected by default.



The screenshot shows the same financial website as above, but with the 'Field Properties' pane open on the right side. The 'Action' dropdown is set to 'Get Table Data'. The 'Interaction' is 'GetTableDataInteraction'. The 'ExtractBy' is 'GetText'. The 'IsExtracted' checkbox is checked. The 'Field Properties' section shows 'Control Tag' as 'li', 'Control Type' as 'Table', and 'Display Name' as 'BSE INDICES'. The 'Find By' is 'css3selector' with the value '.jmbseindicestable.show'. The 'Saved' button is visible at the bottom of the pane.



13. In the **ExtractBy** list, **GetText** is selected by default. You can change it to **InnerText** if the data table is empty. **Inner Text** option is used when you want to fetch data from the hidden rows of the table. This data is already retrieved when the page loads; however, the data is not visible until you click **More>>**.

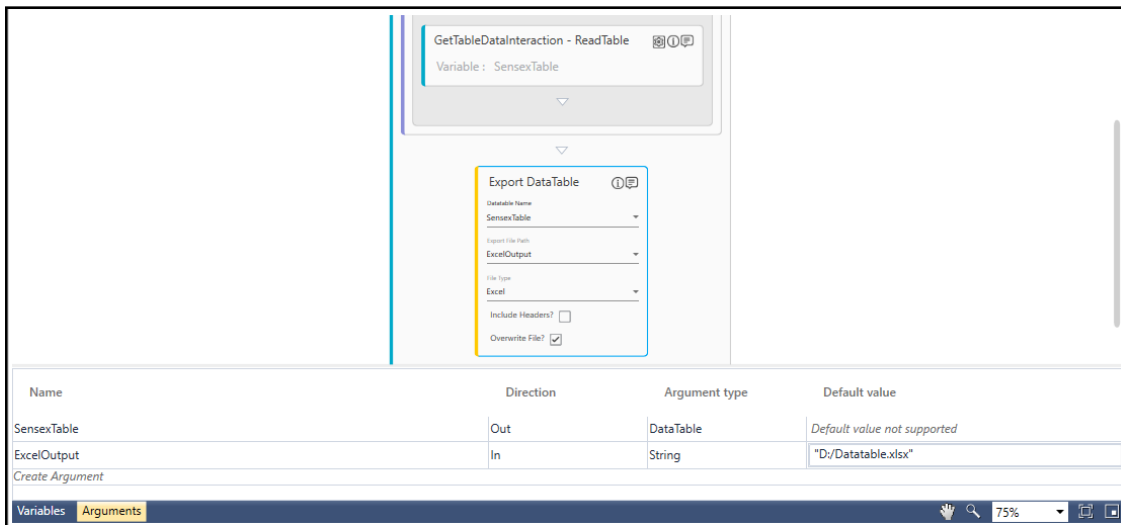


14. In the **Display Name** field, enter a desired name.
15. In the **Find By** list, **css3selector** is selected by default. Depending on the html tagging, **css3xpath** can also be available in this list.
16. In the **Variable Name** list, select the argument, **SensexTable**.

- Use  (**Run Now**) icon, to see if the intended configuration is done and the preview of the datatable is extracted.

Column1	Column2	Column3	Column4	Column5
S&P BSE Sensex	28535.78	29968.79	+1433.01	+5.02
S&P BSE 100	8343.22	8741.45	+398.23	+4.77
S&P BSE 200	3471.93	3635.60	+163.67	+4.71
S&P BSE 500	10666.93	11168.11	+501.18	+4.70
S&P BSE MidCap	10211.57	10588.67	+377.10	+3.69

- Click the  (**Save**) icon to configure the details captured. The saved details start appearing in the **Saved** tab of **Field Properties** bar.
- Click the  (Save configured field and return to studio) icon to save the field that you configured and return to the Studio menu.
- Create an **In** argument, **ExcelOutput** to read and view the datatable extracted.
- Double click the **Export Data Table** activity below the **Get Data** activity.



Name	Direction	Argument type	Default value
SensexTable	Out	DataTable	Default value not supported
ExcelOutput	In	String	"D:/Datatable.xlsx"

- In the **DataTable Name** list, select the **SensexTable** argument to provide the input to the **Export Data Table** activity.
- In the **Export File Path** list, select the **ExcelOutput** argument to export the datatable in the excel files saved in your system.
- In the **File Type** list, select **Excel**.
- In the **Studio** menu, setup the environment and then perform the test run.

The process returns the excel file containing the BSE Sensex detail.

Note: In case UI element of the header of the table header is different from other columns, header name does not get captured. this is because only homogeneous cells are captured using this method. The default header names assigned in the DataTable must be used.

Windows Application

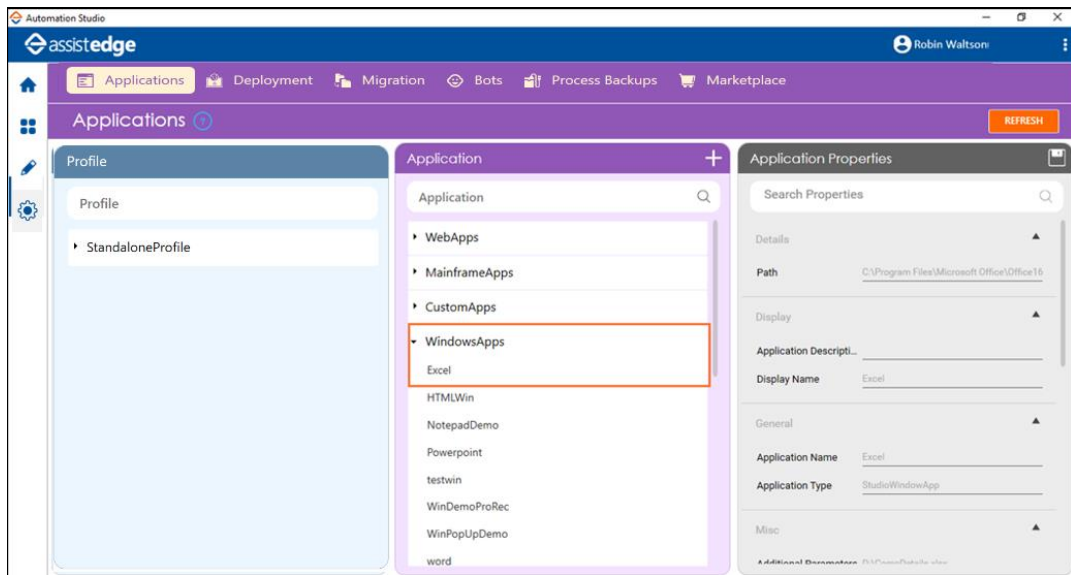
This activity allows you to automate Windows thick client (.exe files, the executable file for Microsoft Windows) such as Notepad, Command Prompt, .Net thick client applications and others in a codeless manner through out-of-box adapter, WindowsApps.

Prerequisite: Configuring Windows Application

You must configure the windows application in Automation Studio before you start configuring the steps of automation process workflow. This establishes the connection between the windows application and Automation Studio to perform the automation.

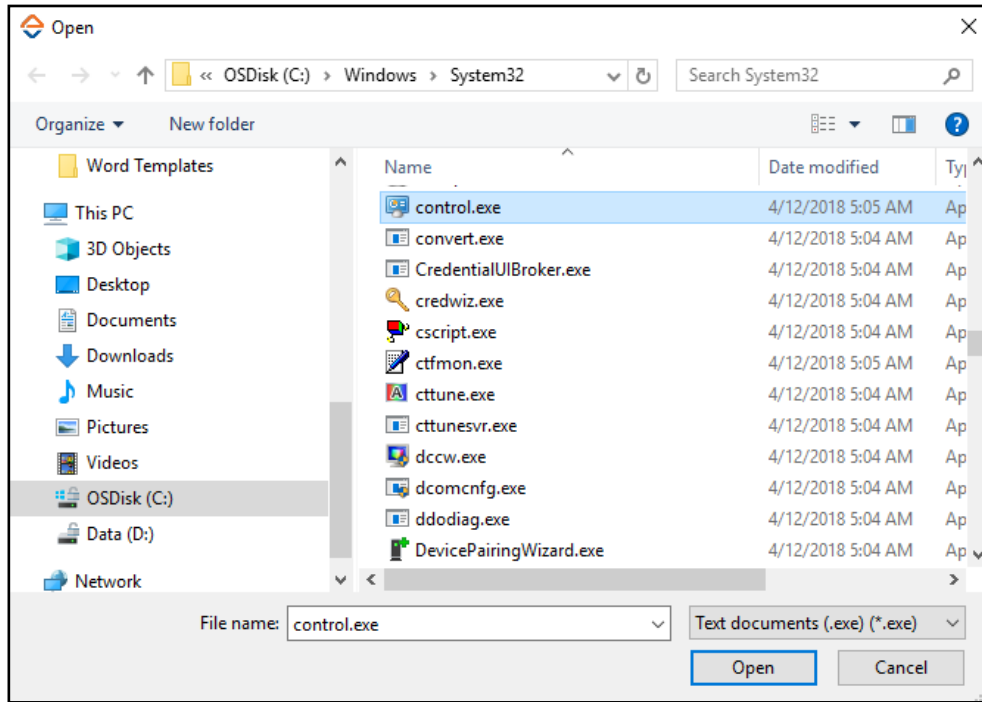
Below are the minimum required properties for configuring the intended windows application. If you want to define remaining properties, refer [WindowsApp](#) in the Admin Menu.

1. In the **Admin** menu, add an application of **Application Type- WindowsApps**. The **Add Application** dialog box appears.



2. In the **Application Name** field, enter a desired name of the EXE application you want to add. The name must not contain any special character or space.

3. Click **ADD** and browse for the windows application you want to configure. Click **Open**.



The WindowsApps application is configured.

Using Excel and Word

To use Excel and word features of MS Office following assembly file is required:

Component	Version
Microsoft.Office.Interop.Excel.dll	15.0.0
Microsoft.Office.Interop.Word.dll	15.0.0

To save assembly file:

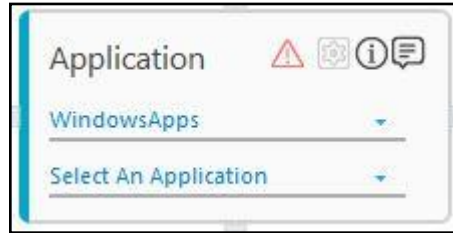
1. Download the assembly files for:
 - Excel- from [here](#) on your server.
 - Word- from [here](#) on your server.
2. Save the assembly file at client-tools > AutomationStudio > **bin** >. If you download/access Automation Studio from the Admin module, you must save the required DLLs at %localappdata% > EdgeVerve > AutomationStudio > bin.

The assembly files are saved.

Using Windows Application Activity

3. In the **Canvas Toolspane**, click **Process Components** to expand the tool and view the associated activities.

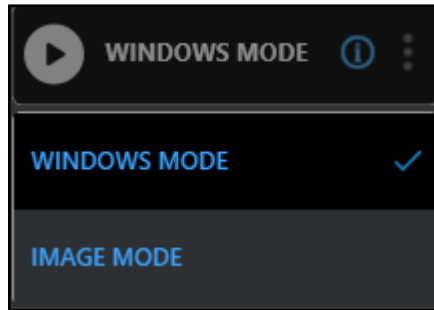
4. Drag the **Application** activity and drop on to the **Flowchart** designer on the **Canvas** area.





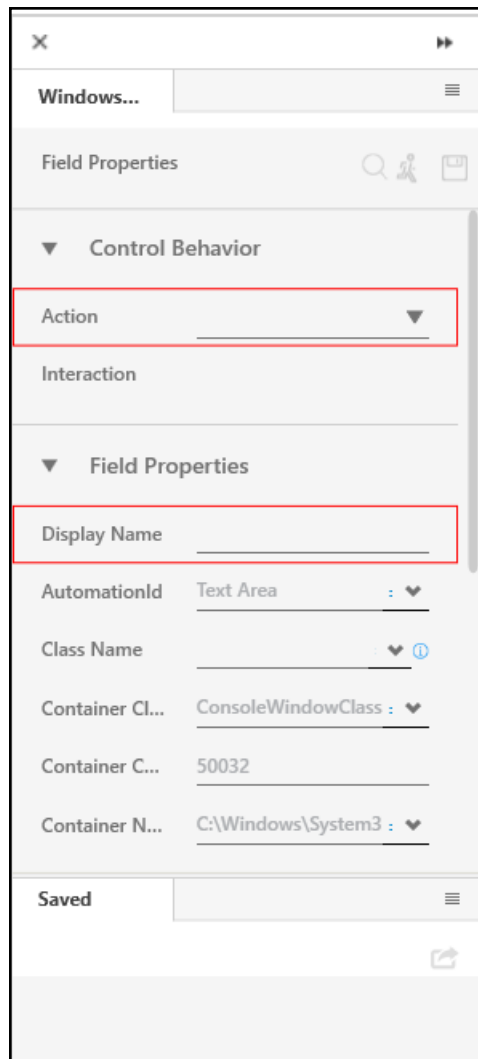
5. In the **Application Type** list, select **WindowsApps**. You must have at least one application added for it to appear in the list.
6. In the **Select An Application** list, select the configured excel file that you want to automate. Alternatively, you can add Windows application (EXE file) at this point of time. To add the windows application:
 - a. In the **Select An Application** list, click **Add New Application**. The **Add New Application** dialog box appears.



- b. In the **Application Name** field, enter a desired name of the application.
- c. In the **Path** field, enter the path of the EXE file of the windows application available on the system.
- d. In the **Additional Parameters** field, enter the path of the file (along with its name) that you want to open within the application. Specify the file name in double quotes, if there are spaces in the file name.
- e. Click **SAVE**. The windows application is added.

7. Click the  (Settings) icon to launch the WINDOWS MODE panel.



8. Click the  (**Play**) icon. The Windows application appears, and the Play icon changes to the  (Recording) icon.
9. Press the Ctrl key and hover over the windows application to start capturing the automation steps that you want to perform in the Windows application. The area that you can capture gets highlighted with a red box.
10. While pressing the **Ctrl** button, click the intended area. The Field Properties panel of the Windows Mode appears. The fields that are mandatory are highlighted with red box.



11. In the Action list, select the action that you want to perform on the captured area. Other mandatory fields changes depending on the selection from the list. Enter details of all the mandatory fields and other relevant fields as per your requirement. Refer [Windows Mode Field Properties and Control Behavior](#) table to know more about the available fields and their respective properties.
12. In the **Display Name** field, enter a desired name of your choice.
13. Click the  (Save) icon to configure the fields.
14. Repeat steps 6 through 10 to capture all the steps involved in the automation process workflow. The field that you configured and saved appears in the Saved pane of the Field Properties panel. You can edit or delete these steps in the Saved pane.
15. Click the  (Save configured field and return to studio) icon to save the field that you configured and return to the Studio menu.

The fields are configured and the Application activity for Application Type-WindowsApps is created.

Note: In certain scenarios the default field configuration may not work at the runtime, manually configure the field properties to identify the UI elements to perform automation. Refer [Handling Dynamic Controls and Manual Field Configuration](#) section to know the details.

Windows Mode Control Behavior and Field Descriptions

The properties of Windows Mode are listed in the following table and can be edited in the Field Properties panel.

Property Name	Usage
Control Behavior	
Action	<p>The applicable action or interaction that can be performed on the selected UI element. The available options change as per the window application being automated:</p> <ul style="list-style-type: none"> ▪ Click- Allows to left click the selected UI element. ▪ Expand/Collapse- Allows to expand or collapse the selected UI element. ▪ Legacy Interaction- Allows to configure UI element of the Windows application which supports a legacy pattern ▪ Get Text- Retrieves text from the selected UI element. ▪ Set Text- Sets user provided text input to the selected UI element and stores it into a parameter. You must define the parameter in the Parameter bar before you start capturing the UI element. ▪ Mouse Click- Allows mouse simulation. It requires the application to be in focus. You can choose to perform Left Click, Right Click or Double Click available from the drop-down list. ▪ Keyboard Input- Captures UI element input from the keyboard in the selected UI element. This interaction allows performing different keyboard input

Property Name	Usage
	<p>combinations with the help of Single Key, Double Keys, Triple Keys and TextEntry options.</p> <ul style="list-style-type: none"> ▪ Set Focus- Brings the UI element in focus before performing any action on it. ▪ If Control Exists- Checks availability of the UI element on the windows application. It is used while working with dynamic controls. ▪ Get Radio Button Status- Retrieves status of the selected radio button. It returns true or false. ▪ Toggle Checkbox Action- Toggles the current state of a checkbox. ▪ Get Checkbox Status- Returns the current selection status of a checkbox. ▪ Perform ComboBox Select Value- Selects the specified value from the drop-down list. ▪ Get ComboBox Value- Retrieves value of the selected combo box. ▪ Perform ComboBox Set Text- Sets a user-specified value on the selected ComboBox. All ComboBox controls do not support this interaction. ▪ Scroll Horizontal- Performs horizontal scroll based on the input provided on the percentage scale from 0 to 100. ▪ Scroll Vertical- Performs vertical scroll based on the input provided on the percentage scale from 0 to 100. ▪ Select Tree Node- Selects a node of a tree control using this interaction. Specify the input in a specific format. For example, for selecting node 1.1, enter the input value as Number/1/1.1. <div data-bbox="699 1289 1167 1755" style="border: 1px solid black; padding: 5px; margin: 10px 0;"> </div> <ul style="list-style-type: none"> ▪ Expand Tree Node- Expands a node of the tree UI element. ▪ Select- Selects a particular UI element. ▪ Get DataGrid Value by Index- Retrieves data grid value by its index. Specify the cell index from which the input is needed.

Property Name	Usage
	<ul style="list-style-type: none"> ▪ Focus Data Grid Cell by Index- Brings a data grid cell in focus based on the input cell index. ▪ Set DataGrid Value by Index- Sets the value to a DataGrid cell based on its index. ▪ Get Data from Table- Retrieves complete data of a table control to a DataTable. If required, the data from the DataTable can then be saved in an excel/CSV file. Following are the steps involved to retrieve the data. <ol style="list-style-type: none"> a. Create an argument of type System.Data.DataTable, before extracting the UI element. b. Click WINDOWS MODE to extract the Data Grid Table control. c. Select Get Data from the Table. d. Set the Variable Name as the argument created in the first step. e. Save the configuration. f. Use Export Data Table activity to write data to excel. <ul style="list-style-type: none"> ▪ Set Focus- This interaction is used to bring any control to focus before performing an action on it.
Interaction	The corresponding interaction class name auto populated against the selected action.
Interaction	The corresponding interaction set against the selected interaction.
Appears if Action selected is Legacy Interaction .	
Action Type	The interaction or the action type corresponding to the legacy interaction..
Key	Appears if Action selected is Keyboard Input . The single key set as input from the keyboard.
Key	Appears if Action selected is Keyboard Input and Interaction selected is DoubleKeys . The second key which is set as input from the keyboard along with the first key.
Key	Appears if Action selected is Keyboard Input and Interaction selected is TripleKeys . The third key which is set as input from the keyboard along with the first and the second key.
Text	Appears if Action selected is Keyboard Input and Interaction selected is TextEntry . Allows user input as a text.
Mouse Action	Appears if Action selected is Mouse Click. Allows user input using the mouse. Available options are: <ul style="list-style-type: none"> ▪ LeftClick- Performs a left click at the specified offset point. ▪ DoubleClick- Performs a double click at the specified offset point. ▪ RightClick- Performs a right click at the specified offset point.
Field Properties	
Display Name	User specified display name of the windows element selected.

Property Name	Usage
AutomationId	Unique identifier for the automation element in the automation tree. See Handling Dynamic Controls section to know more about the usage of this field.
Class Name	Class name of the UI element as defined by the UI element developer. See Handling Dynamic Controls section to know more about the usage of this field.
Container Class Name	Container class name of the UI element as defined by the developer. See Handling Dynamic Controls section to know more about the usage of this field.
Container Control Type	Control type of the container of the UI element as defined by the UI element developer.
Container Name	Container name of the UI element as defined by the developer. See Handling Dynamic Controls section to know more about the usage of this field.
Control Index	Index of the current UI element within the entire set of elements with the same values of automation Id, class name and UI element name as that of the current element.
Control Name	Control name of the UI element as defined by the UI element developer. See Handling Dynamic Controls section to know more about the usage of this field.
Control Type	Control type of the UI element as defined by the developer.
Appears if Action selected is Mouse Click .	
Offset From Control	It is the distance between the UI element and the anchor.
Is applicable only for AssistEdge Engage. Appears if Action selected is If Control Exist .	
Search By Control Order No.	Signifies if the UI element needs to be searched based on the UI element order number. Control order number is the index of the current element within the entire set of elements with the same value of UI element Id as that of the current element.
Appears if Action selected is Get Text .	
Variable Name	User defined name of the variable that stores the captured value.
Variable Type	Type of the variable defined. Available options are: <ul style="list-style-type: none"> ▪ Text - A sequence of character, either as a constant or a variable. ▪ Numeric - An integer type that range from negative through positive. ▪ Decimal - An exact numeric value defined by its precision and scale. ▪ Boolean - A data type used for making decision. Can have only two values- true or false. ▪ Data Table - A data type that stores tabular data.
Misc Properties	

Property Name	Usage
Max wait for input idle	The maximum time interval in milliseconds for which the windows plugin waits on launch before it starts executing the configured interactions.
Retry Count	Number of times, the plugin retries to find a UI element, if not found.
Retry Interval (ms)	The time interval in milliseconds for which the plugin waits before it attempts for a retry.

.NET OBJECT MODE Control Behavior and Field Descriptions

The properties of .NET Object Mode are listed in the following table and can be edited in the **Field Properties** panel.

Property Name	Usage
Control Behavior	
Action	<p>The applicable action or interaction that can be performed on the selected UI element. The available options change as per the window application being automated:</p> <ul style="list-style-type: none"> ▪ Click- Allows to left click the selected UI element. ▪ Get Text- Retrieves text from the selected UI element. ▪ Set Text- Sets user provided text input to the selected UI element and stores it into a parameter. You must define the parameter in the Parameter bar before you start capturing the UI element. ▪ If Control Exists- Checks availability of the UI element on the windows application. It is used while working with dynamic controls. ▪ Get Radio Button Status- Retrieves status of the selected radio button. It returns true or false. ▪ Select Radio Button – Selects the required radio button. ▪ Toggle Checkbox Action- Toggles the current state of a checkbox. ▪ Get Checkbox Status- Returns the current selection status of a checkbox. ▪ Perform ComboBox Select Value- Selects the specified value from the drop-down list. • Get ComboBox Value- Retrieves value of the selected combo box • Select Tree Node- Selects a node of a tree control using this interaction

Property Name	Usage
	<div data-bbox="516 268 984 730" style="border: 1px solid gray; padding: 5px; margin-bottom: 10px;"> </div> <ul style="list-style-type: none"> ▪ Get DataGrid Value by Index- Retrieves data grid value by its index. Specify the cell index from which the input is needed. ▪ Focus Data Grid Cell by Index- Brings a data grid cell in focus based on the input cell index. ▪ Set DataGrid Value by Index- Sets the value to a DataGrid cell based on its index. ▪ Get Data from Table- Retrieves complete data of a table control to a DataTable. If required, the data from the DataTable can then be saved in an excel/CSV file. Following are the steps involved to retrieve the data. <ol style="list-style-type: none"> a. Create an argument of type System.Data.DataTable, before extracting the UI element. b. Click .NET OBJECT MODE to extract the Data Grid Table control. c. Select Get Data from the Table. d. Set the Variable Name as the argument created in the first step. e. Save the configuration. f. Use Export Data Table activity to write data to excel. <ul style="list-style-type: none"> ▪ Perform ListBox Select Value – Selects required value in a List box ▪ Get ListBox Value - Retrieves the value selected in the List box ▪ Select ListViewItem - Selects a particular value in the List view or List box ▪ Select TabItem - Selects the required Tab
Field Properties	
Display Name	User specified display name of the windows element selected.
Control Order	Index of the current UI element within the entire set of elements with the same values of automation Id, class name and UI element name as that of the current element.

Property Name	Usage
Control Type	Control type of the UI element as defined by the developer.

Windows Application Properties

The properties of a windows application are listed in the following table and can be edited in the **Property** grid on the right pane.

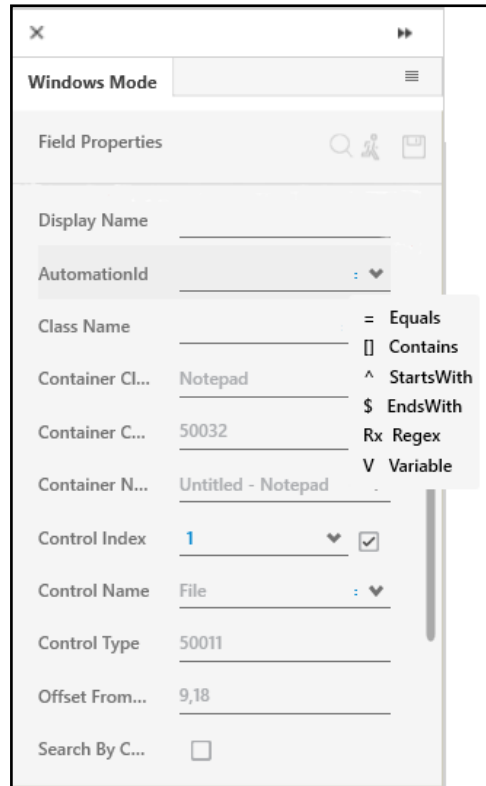
Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as the name of the selected application. You can change the name as required.


Handling Dynamic Controls and Manual Field Configuration

Automation Studio provides different identification criteria and Parametrization of the value of the dynamic field properties of the UI elements. You can reconfigure some of the auto captured field properties of such UI elements to create a more effective way of identifying the indented UI element. For such controls, remove the dynamic part of the string and use an appropriate option for UI element identification. For example, if a Class Name field is recognized as AssistEdge_Studio_20180205083009, the later part of the string, that is a timestamp, is dynamic and can be removed. Retain only the static part and reconfigure the field properties using the suitable option.

Following are the different field properties that can be reconfigured against their respective values:

- AutomationId
- Class Name
- Container Class Name
- Container Name
- Control Name



Use the  (drop-down arrow) icon to reconfigure the identification criteria. Available options are:

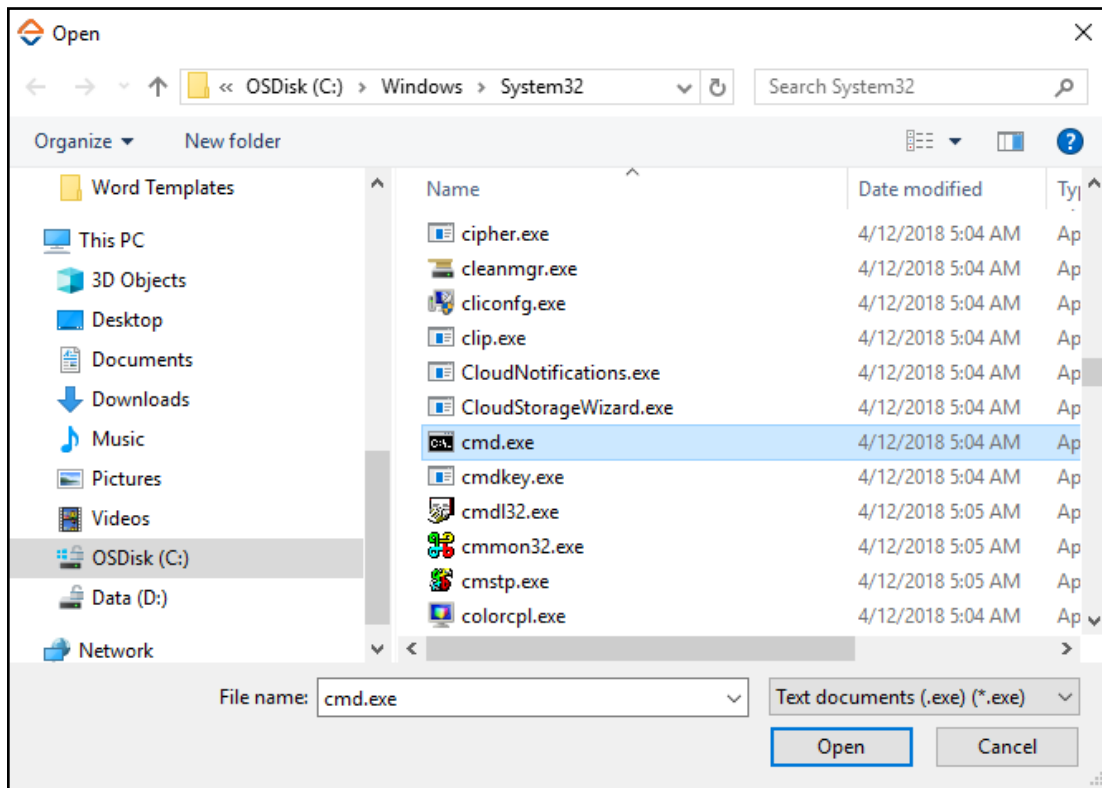
- **Equals**- Identifies the UI element based on the exact match with the value of the selected field property.
- **Contains**- Identifies the UI element based on the string present in the value of the selected field property.
- **StartsWith**- Identifies the UI element based on the starting string of the value of the selected field property.
- **EndsWith**- Identifies the UI element based on the ending string of the value of the selected field property.
- **Regex**- Identifies the UI element with the string matching the value of the selected field property as per the identification pattern defined through the provided regular expression such as a.b, *txt and others.
- **Variable**- Identifies the UI element as per the parametrization of the value of the selected field property. Parametrization allows to run the identification process over and over again using different values. With parameterization, windows controls with looping and assignment activities, are used.

Step-by-Step Guide to Automate a Windows Application

Let's create a process to know the MAC address of the system using the Command prompt (CMD).

Prerequisite:

1. In the **Admin** menu, add an application of **Application Type- WindowsApps**.
2. In the **Application Name** field, enter a desired name of the CMD application you want to add. The name must not contain any special character or space.
3. Click **ADD** and browse for the CMD application. Click **Open**.

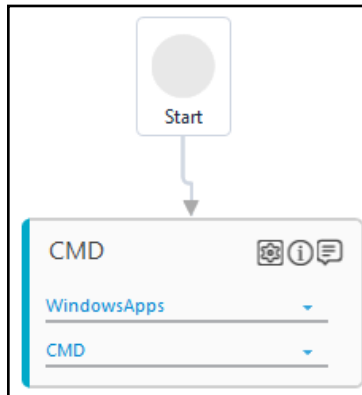




The CMD application is configured.

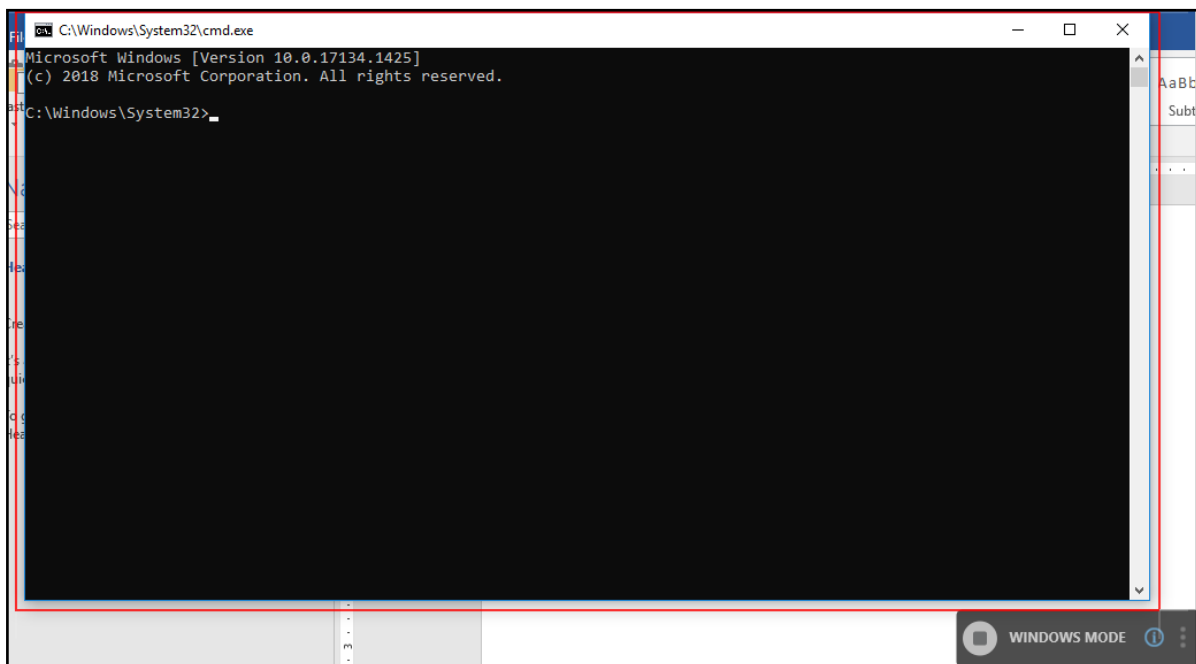
To automate the task of finding the MAC address of the system:

4. Create a new process.
5. From the Canvas Tools panel, add Application activity to the Flowchart designer on the Canvas
6. In the Application Type list, select WindowsApps.

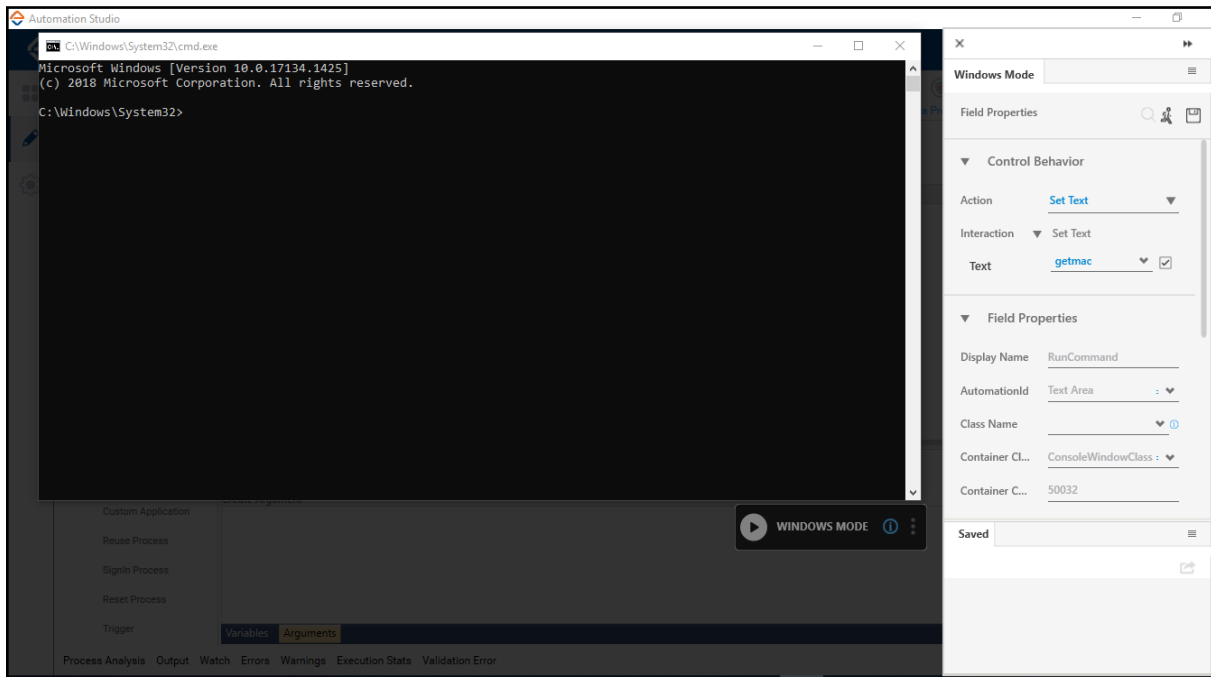
7. In the **Select an Application** list, select the configured CMD application.







8. Click the  (**Settings**) icon to launch the **WINDOWS MODE** of the Multimodal interface. The CMD application appears.
9. Click the  (Play) icon and then press the **Ctrl** button from the keyboard to highlight the area in red box that you want to capture.



- While pressing the **Ctrl** button, click the CMD interface to enter the command. The Field Properties panel of the Windows Mode of the Windows application appears. The mandatory fields are highlighted with red box.



- In the **Action** list, select **Set Text**. The **Text** field appears.
- In the **Text** field, enter **getmac** and select the check box beside the field to set **getmac** as the default value.
- Click the  (Save) icon to configure the fields.
- Click the  (Play) icon again and then press the **Ctrl** button from the keyboard to highlight the CMD interface where you want to capture the **Enter** interaction of the keyboard after **getmac** is entered.
- In the **Action** list, select **Keyboard Input**. The **Interaction** and **Key** fields appear.
- In the **Interaction** list, set **SingleKey** as only single key from the keyboard must be pressed.
- In the **Key** list, select **Return** to press the **Enter** key of the keyboard.
- Click the  (Save) icon to configure the fields.
- Click the  (Save configured field and return to studio) icon to save the field that you configured and return to the Studio menu.
- In the **Studio** menu, setup the environment and then perform the test run.

The process returns the Mac address of the system in the CMD prompt.

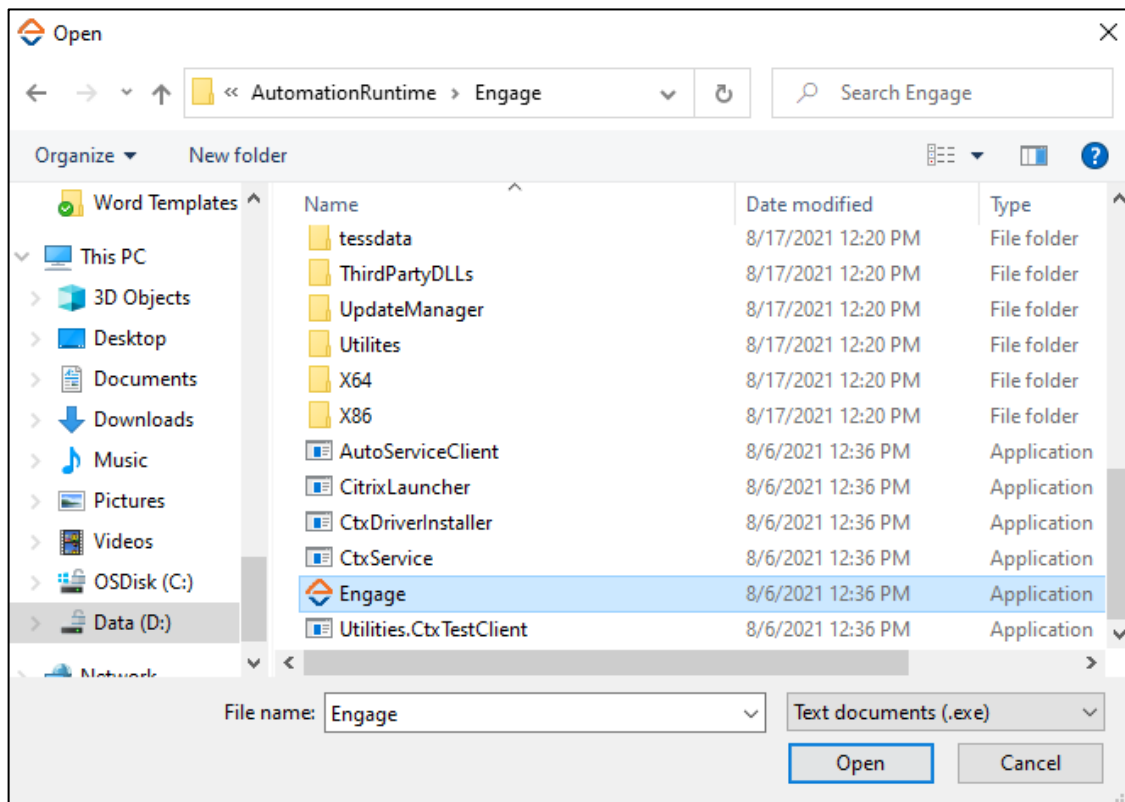
Step-by-Step Guide to Automate .Net Based Windows Application

Let's create a process to open the Engage application and extract the username from the sign-in page.

Engage is a .Net based WPF application of AssistEdge and we have used it as a test application in this example.

Prerequisite:

1. In the **Admin** menu, add the **Engage** application in the **WindowsApps** category.
2. In the **Application Name** field, enter a desired name for the **Engage** application. The name must not contain any special character or spaces.
3. Click **ADD** and browse for the **Engage.exe** file. Click **Open**.

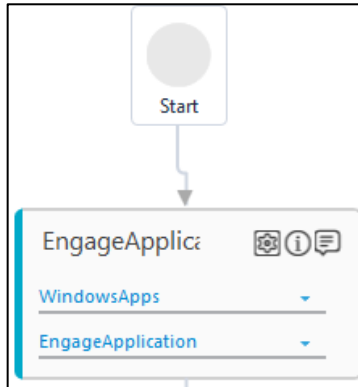




The Engage application is configured in Automation Studio.

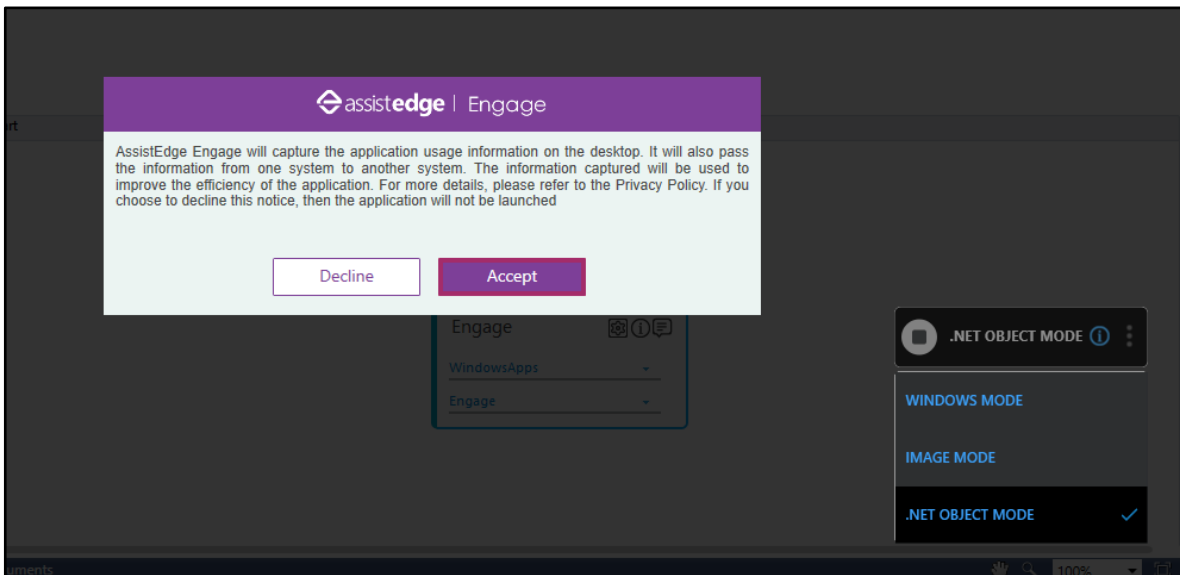
To automate extracting username from the sign-in page of the Engage application:

1. Create a new process.
2. Create an **In** argument in the **Parameter** bar, **userName** to store the extracted text.
3. From the Canvas Tools panel, add Application activity to the Flowchart designer on the Canvas
4. In the Application Type list, select **WindowsApps**.

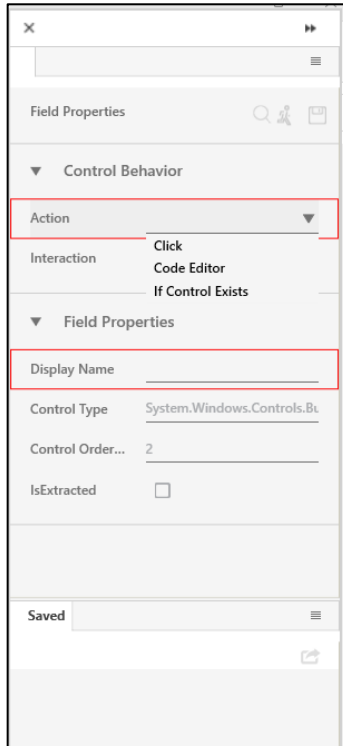
- In the **Select an Application** list, select the configured Engage application.




- Click the  (**Settings**) icon to launch the **Multimode Interface** and select the **.NET OBJECT MODE**. The Engage application appears.
- Click the  (Play) button and then press the **Ctrl** button from the keyboard to highlight the area in red box that you want to capture. The Field Properties panel appears and the mandatory fields are highlighted with red box.



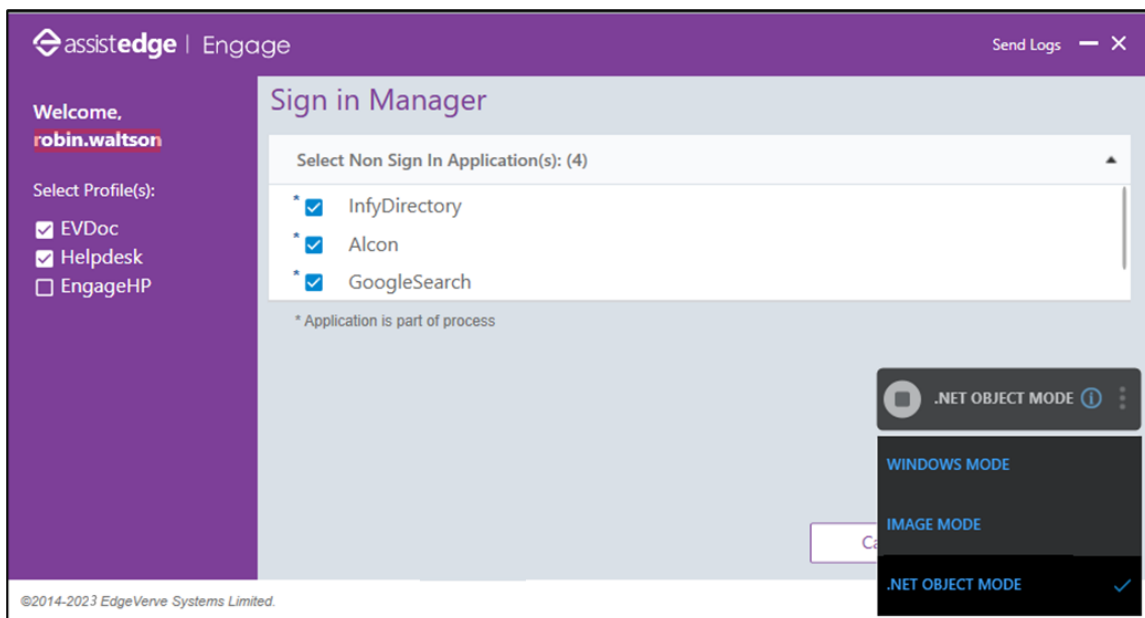
8. In the **Action** list enter the code to click the **ACCEPT** button.



9. Click the  (Save) icon to configure the fields.

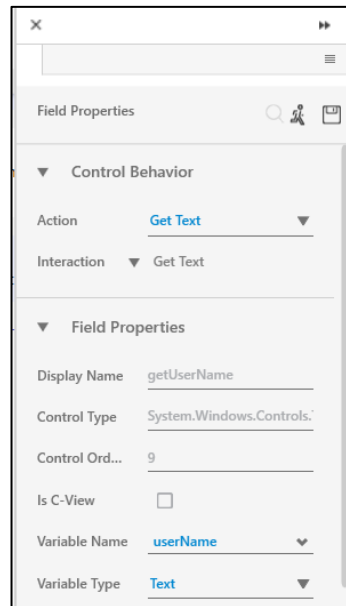
10. Click the  (Play) button.


11. Press the **Ctrl** button from the keyboard , hover the username area to highlight it in red box and then click. The Field Properties panel appears, and the mandatory fields are highlighted with red box.

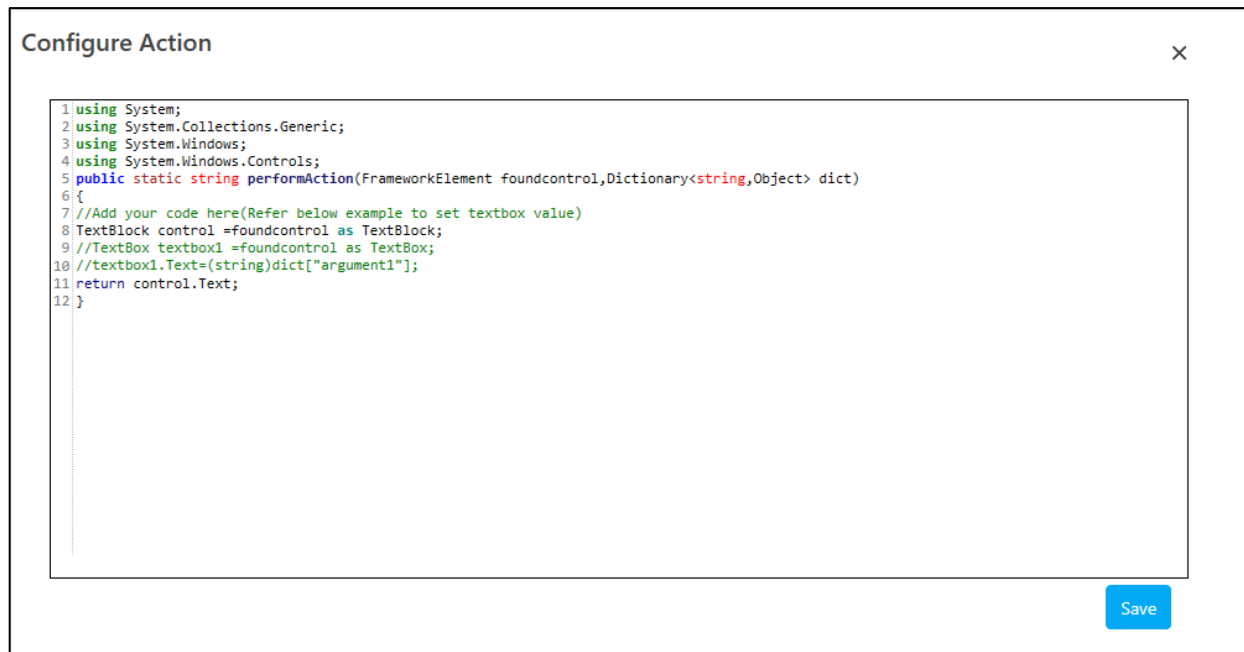



12. In the **Action** list, select the action **Get Text** to be performed on the selected control.


13. In the **Field Properties** pane, enter the **Display Name** of your choice.
14. In the **Variable Name** list, select **userName** parameter created above.



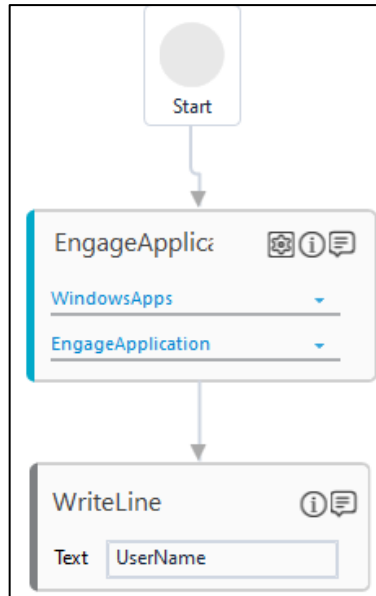
15. If you to use the code editor to write custom C# code, select option of **Code Editor** in the **Action** field click the  (horizontal ellipsis) button. The **Configure Action** screen opens. Configure the required code to return the text.



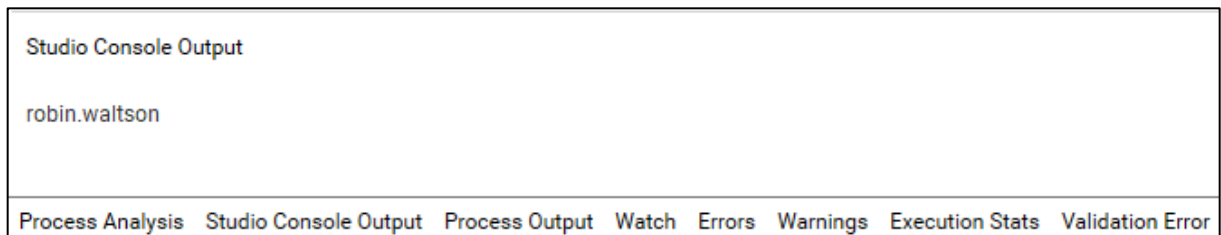
16. Click the  (Save) icon to configure the fields.

17. Click the  (Save configured field and return to studio) icon to save the field that you configured and return to the Studio menu.

To view the output in Automation Studio, let's add WriteLine activity. You can assign this process to a robot if you want to execute this process outside Automation Studio.



18. Add a **WriteLine** activity below the **Application** activity and in the **Text** field, enter `userName` (or select) parameter to print the extracted username.
19. Save the process.
20. Setup the environment and then perform test run. Following is the output displayed in the **Studio Console Output**:



Java Applications

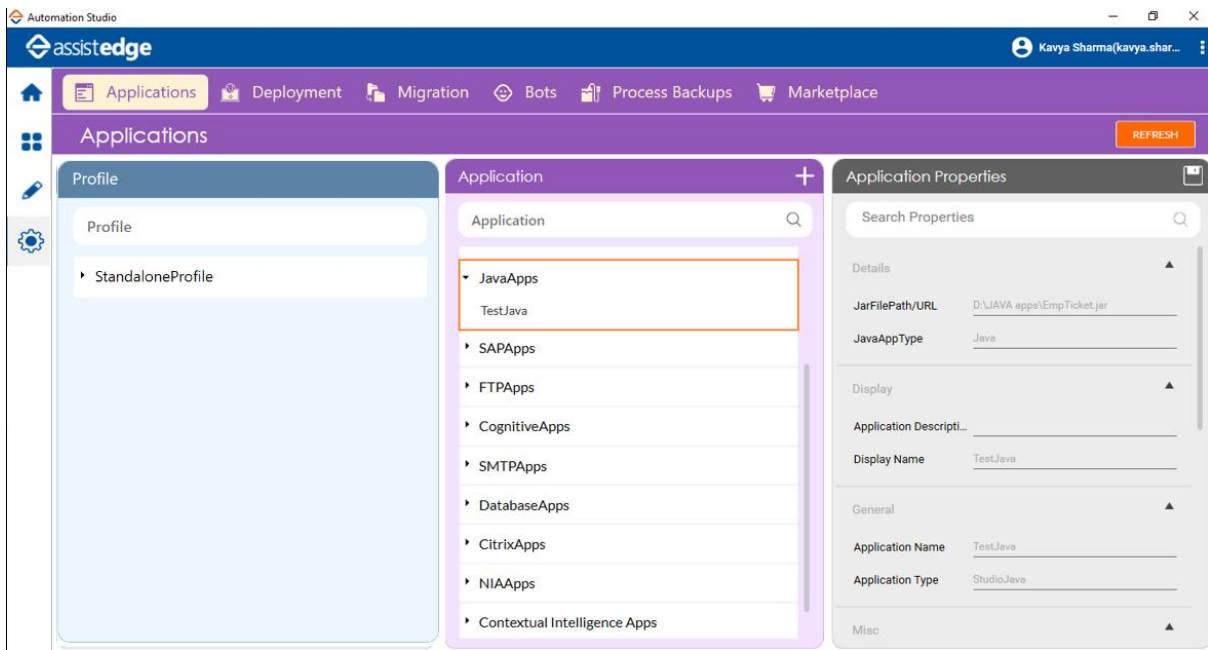
This activity allows you to automate the actions performed on the Java application such as testing Java applications or performing operations on the UI.


The supported files types are - JAR, JNLP (Java Network Launch Protocol) and Java Applet.

Prerequisites

- Install Java (JDK, v) on the system where the automation process workflow needs to be created and where the automation would run.
- Configure the required Java application in Automation Studio where you want to perform the automation. Below are the minimum required properties for configuring the intended Java application. If you want to define remaining properties, refer JavaApps section. To configure Java application with minimum required properties:

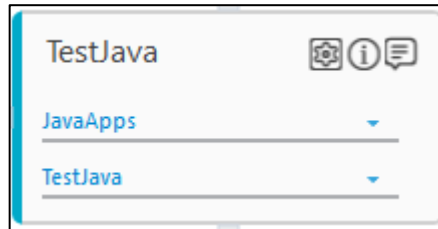
1. In the **Admin** menu, add an application of **Application Type - JavaApps**.



2. Browse and select the required .jar file.
3. In the Application Properties panel:
 - Enter the URL of the JNLP file/applet in the **JarFilePath/URL** field.
 - Enter the time out duration of the application launch in the **Launch Time Out (in secs)** field, if the **JavaAppType** is **JNLP Applications** or **Applet**.
4. In the **Environment Type** list, select the type of environment of the added Java application.
5. Click the  (**Save Properties**) icon to save the application. The Java application is configured.

Using Java Application Activity

1. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
2. Drag the **Application** activity and drop on to the Flowchart designer on the **Canvas** area. The validation error symbol disappears when required inputs are provided in the correct format.



3. From the Application Type list, select **JavaApps**. You must have at least one application added for it to appear in the list.
4. In the Select An Application list, select the configured Java application that you want to automate. Alternatively, you can add a new Java application at this point of time. To add a Java application:
 - a. In the Select An Application list, click Add New Application. The Add New Application dialog box appears.

Add New Application ✕

Application Type
JavaApps


ApplicationName

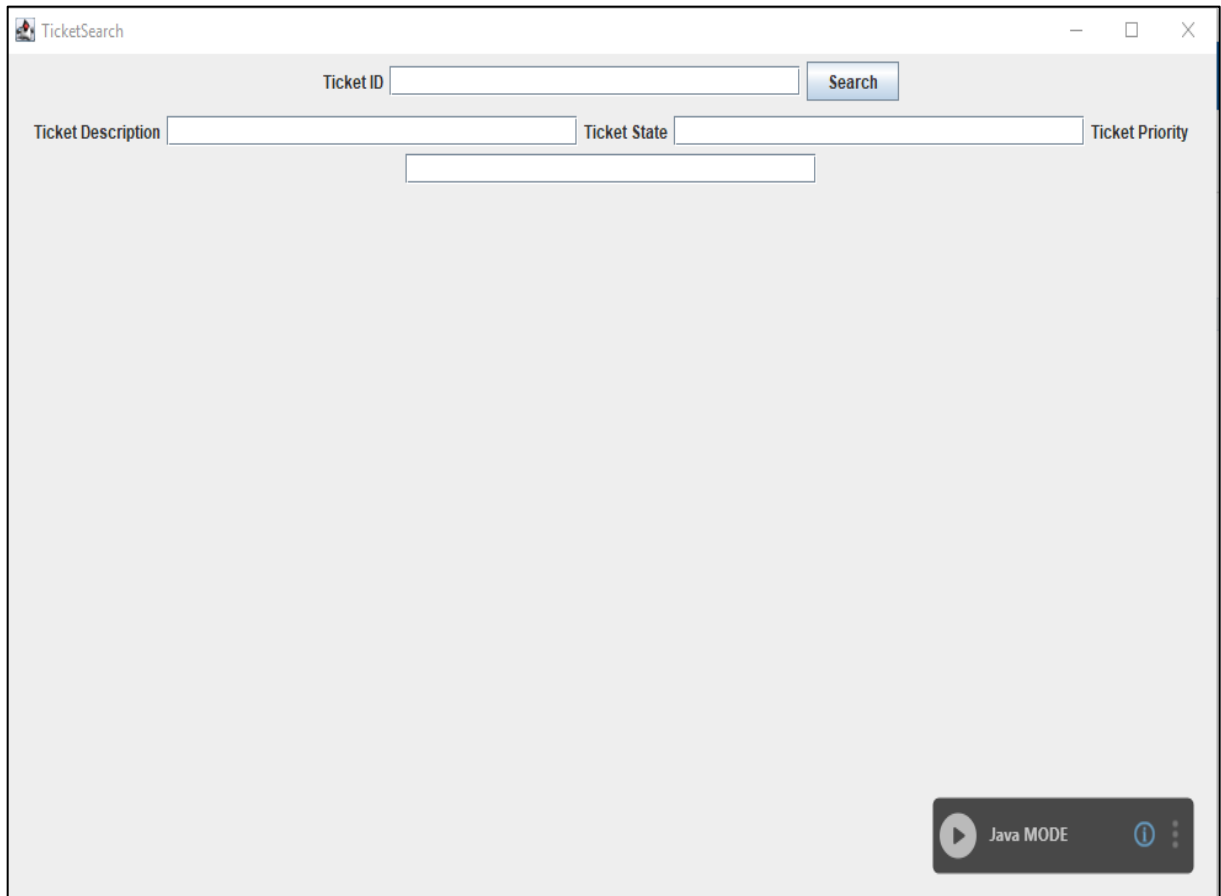
JarFilepath



EnvironmentType
EnvironmentType

SAVE

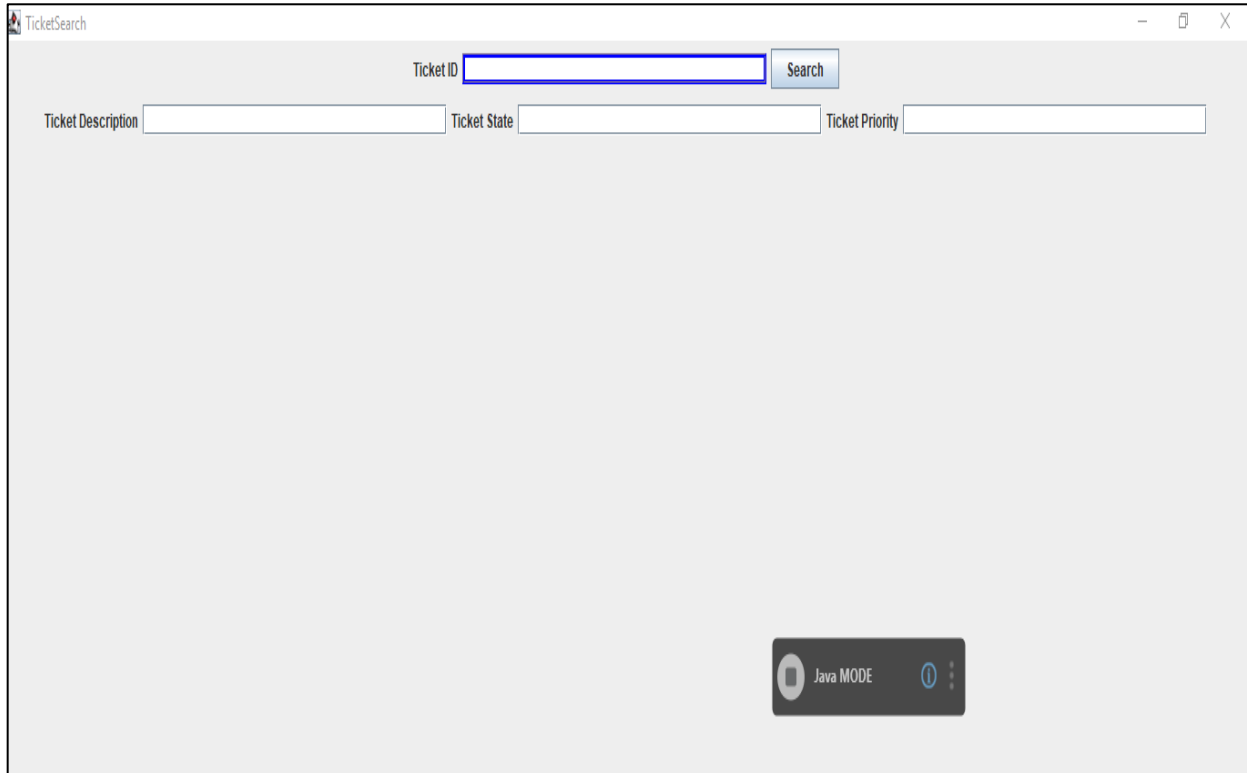
5. In the Application Type field, select the required application type of the Java application you want to configure in Automation Studio. Available options are:
 - JavaApps
 - JavaApps - JNLP
 - JavaApps - Applet
6. In the **ApplicationName** field, enter a desired name of the Java application.
7. If **JavaApps** is selected, enter the file path along with the file name and extension of the required JAR file in the **JarFilePath** field.
8. If **JavaApps - JNLP** is selected, enter the URL of the JNLP file in the **URL** field.
 - a. Enter the time out duration of the application launch of the JNLP file in the **Launch Time Out(in secs)** field.
 - b. Enter the window title of the JNLP window in the **Launch Window Caption** field.
9. If **JavaApps - Applet** is selected, enter the URL of the Java applet in the **URL** field.
 - Enter the time out duration of the application launch of the Java applet in the **Launch Time Out(in secs)** field.
10. In the **Environment Type** list, select the type of environment of the Java application. Available options are:
 - i. WINDOW_CLASS_BASED
 - ii. EVENT_BASED
11. Click **SAVE**. The Java application is added.

- Click the  (**Settings**.) icon to launch the **Java** MODE interface. The configured Java application appears along with **Java** **MODE** interface.

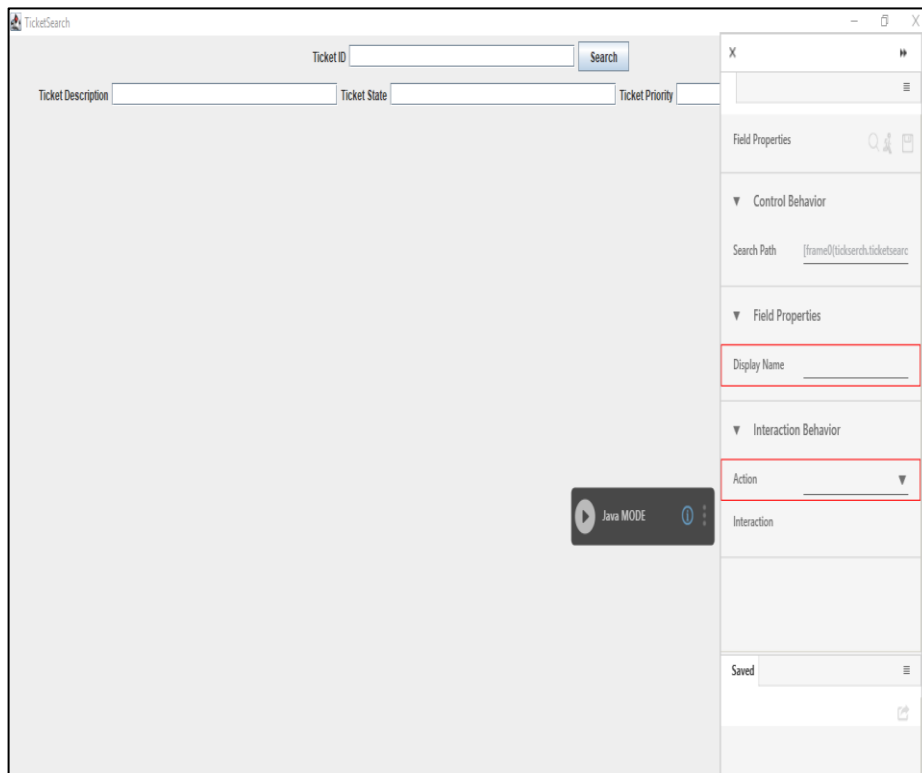




- Click the  (Play) button to capture the steps that you want to perform within the Java application. The Play button changes to the  (Recording) button.

14. Hover over the field that you want to capture. The field gets highlighted with the blue box.



15. Click the highlighted area. The Field Properties panel of the **Java MODE** appears. The fields that are mandatory are highlighted with red box.



16. In the Action list, select the action that you want to perform on the captured area. Other mandatory fields change depending on your selection from the drop down list. Enter the required details. Refer Field Properties table to know more about the available fields and their respective properties.
17. Click the  (Save) icon to configure the fields.
18. Repeat steps 13 through 17 to capture all the steps involved in the process. The configured field is saved and starts appearing in the Saved pane of the Field Properties panel. You can edit or delete these steps in the Saved pane, if required. Click the  (Save configured field and return to studio) icon to save the field that you configured and return to the Studio menu.

The fields are configured and the Application activity for Application Type- **JavaApps** is created.

Java Mode Control Behavior and Field Descriptions

The properties of the JAVA MODE are listed in the following table:

Property Name	Usage
Control Behavior	
Search Path	
Interaction Behavior	
Action	<p>It is the action or the interaction that can be performed on a Java application depending on the type of UI element as per the requirement. Available options are:</p> <p>Set Text - Sets the user provided value of the UI element attribute and stores it as the value of the parameter. You must define the parameter in the Parameter bar before you start capturing the element.</p> <p>Dock Window -</p> <p>Click - Allows to left click the selected UI element.</p> <p>Set Key Event Text - Sets the user provided value of the keystroke event that occurred in the selected UI element attribute and stores it as the value of the parameter selected in the Text field. You must define the parameter in the Parameter bar before you start capturing the element. Alternatively, select the Is Default Value checkbox to set a default value of the keystroke event.</p> <p>Get Text - Extracts the current value of the selected UI element such as, value of an input field , text box or the selected area.</p> <p>Get Table Cell Value - Extracts the cell value from a table available in the Java application by indicating the row and column of the required cell.</p>

	<p>Get Table Data - Extracts the table data in sting format with column and row details.</p> <p>Get Table Selected Rows - Extracts the value from a table selected row available in the Java application.</p> <p>Perform Table Click - Clicks the area on the table as per the specified row and column number.</p> <p>Perform Table DoubleClick - Double clicks the area on the table as per the specified row and column number.</p> <p>Perform Table Row Selection -</p> <p>Get Tree Data - Extracts the data from the specified tree data structure in the XML format.</p> <p>Get Tree Selection Paths - Extracts path of the selected node in the tree data structure.</p> <p>Perform Tree Click - Clicks the tree data structure as per the selection path of a node.</p> <p>Perform Tree DoubleClick - Double clicks the tree data structure as per the selection path of a node.</p> <p>Perform Popup Click On JTree - Clicks the pop menu item.</p> <p>Perform TabbedPane Select - Clicks the pane as per the specified pane title of the Java application.</p> <p>Perform ComboBox Select Value - Selects the value for a combo box as per the specified name.</p> <p>Perform ComboBox Select ValueAt - Selects the value for a combo box as per the specified index.</p> <p>Perform Menu Item Click - Click the menu item as per the specified main menu and the sub menu.</p> <p>Get RadioButton Status - Specifies if the radio button is checked or not.</p> <p>Get RadioButton Text - Extracts the text of the radio button.</p> <p>Select ListItem - Selects the list item as per the specified indices.</p> <p>Is Exists - Checks if the UI element exists at the specified area. It returns a boolean value and is useful when working with dynamics controls.</p> <p>Is Enabled - Checks if the UI element is enabled. It returns a boolean value.</p> <p>Is Visible - Checks if the UI element is visible. It returns a boolean value.</p> <p>Is Focused - Checks if the UI element is in focus. It returns a boolean value.</p> <p>Request Focus - Sets the UI element into focus.</p> <p>Insert JText Comp Text - Sets the user proved string value into the JText component as per the specified offset.</p> <p>Button Enter - Performs the return action of the keybaord on the specified UI element</p>
--	--

	<p>GridRowRobotEnter - Performs the return action of the keyboard.</p> <p>CheckBox Click - Clicks the specified checkbox.</p> <p>Perform Click at Offset - Clicks at the specified X and Y coordinates.</p> <p>Set Table Cell Value - Sets value in a field as per specified row and column number.</p> <p>Select Radio Button - Selects the desired radio-button element from the given group.</p> <p>Click Checkbox Action - Checks or un-checks the desired check-box element.</p> <p>PopUp Click - Click the UI element which opens a pop-up menu.</p> <p>Enter Click</p>
Interaction	The corresponding interaction set against the selected action.
Appears if Action selected is Set Text, Set Key event Text, Insert JTextComp Text, Set Table Cell Value, Select Radio Button	
Text	Allows user input as a text. You can select the parameter holding the required value defined in the Parameter bar. Alternatively, select the Is Default Value check box and specify the required text in the Text field to set the value as the default value.
Appears if the Action selected is Insert JTextComp Text.	
Offset	The distance between the target item and the anchor.
Appears if Action selected is Get Table Cell Value, Set Table Cell Value.	
Column Number	The column number that contains the required cell from the table.
Row Number	The row number that contains the required cell from the table.
Dock Window	This section appears only when Action selected is Dock.
Dock Action	When a link or button on the web page is opened in a new browser window, the Dock Action button enables the new window listed in the multimodal UI. You can select one of the windows in the multimodal UI to configure controls on the intended window.
Appears if Action selected is Click, Perform ComboBox Select Value or Perform ComboBox Select ValueAt, CheckBox Click	
BlockingCall	Indicates that the current thread is blocked until the operation is executed. If it is not handled it can the application or the system itself.
Value	Allows user input of the value of the combo box set. You can select the parameter holding the required value defined in the Parameter bar. Alternatively, select the Is Default Value check box and specify the required value in the Value field to set the value as the default value.
Appears if Action selected is Perform Menu Item Click.	

Main Menu	The main menu where the required item needs to be clicked.
Sub Menu	The sub menu where the required item needs to be clicked.
Appears if the Action selected is Select ListItem .	
ItemIndices	Index number of the selected item of the list.
Appears if the Action selected is Perform Click At Offset.	
X Offset	Vertical offset on the Java application. The value must be entered in pixels.
Y Offset	Horizontal offset on the Java application. The value must be entered in pixels.
Appears if the Action selected is Click CheckBox Action.	
Field Properties	
Display Name	User specified display name of the action configured.
Appears if Action selected is Get Text, Get Table Cell Value, Get RadioButton Status. Get RadioButton Text, Get CheckBox Status. Get CheckBox Text, Is Exists, Is Enabled, Is Visible, Is Focused,	
Variable Name	User defined name of the variable that stores the captured value. You can define the variable name here itself or predefine it in the Parameter bar.
Variable Type	Type of the variable defined. Available options are: Text - A sequence of character, either as a constant or a variable. Numeric - An integer type that range from negative through positive. Decimal - An exact numeric value defined by its precision and scale. Boolean - A data type used for making decision. Can have only two values- true or false. Data Table - A data type that stores tabular data.

Java Application Properties

The properties of a web application are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes, the application ignores any error while executing the activity. If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No.
Delay	

Property Name	Usage
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as the name of the selected application. You can change the name as required.

Database Applications

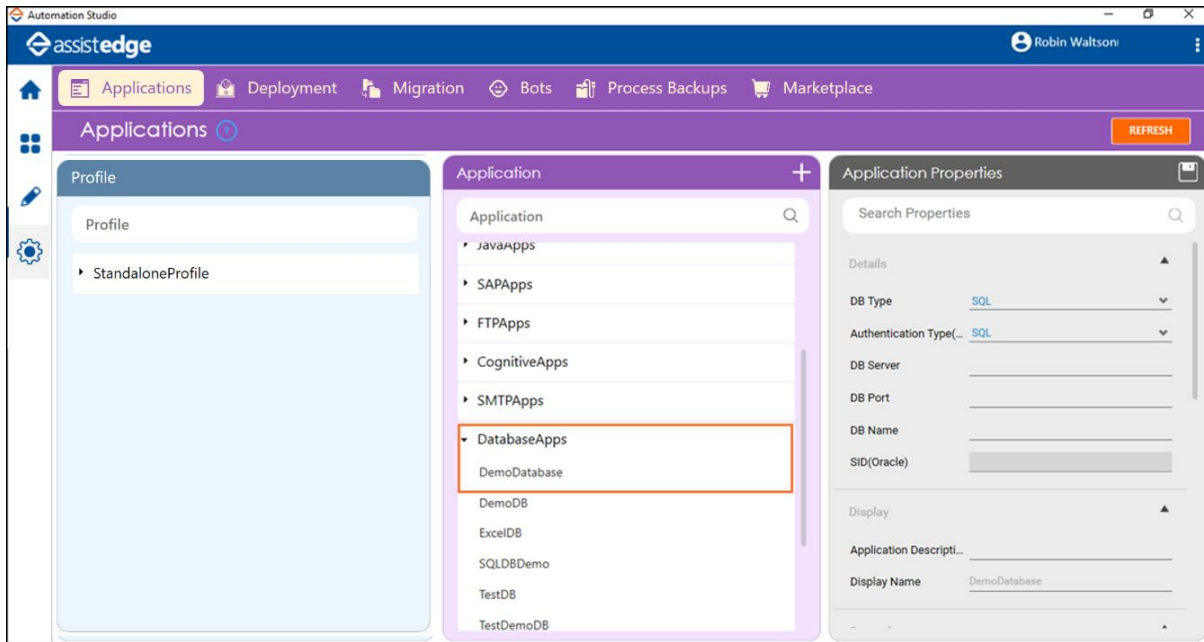
This activity allows you to automate the task you perform in a database such as retrieving data, updating data, automating database administration task and so on.


Prerequisite: Configuring Database Application

You must configure the database application in Automation Studio before you start configuring the steps of automation process workflow. This establishes the connection between the database application and Automation Studio to perform the automation.

Below are the minimum required properties for configuring the intended database application. If you want to define remaining properties, refer [DataApps](#) in the Admin menu.

1. In the **Admin** menu, add an application of **Application Type- DataApps**.

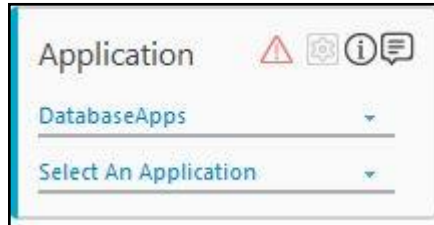


2. In the Application Properties panel:
 - In the **DB Type** list, select the type of database to configure. Based on the type of database selected provide the other relevant details.
3. Click the  (**Save Properties**) icon to save the application.

The DatabaseApp application is configured.

Using Database Application Activity

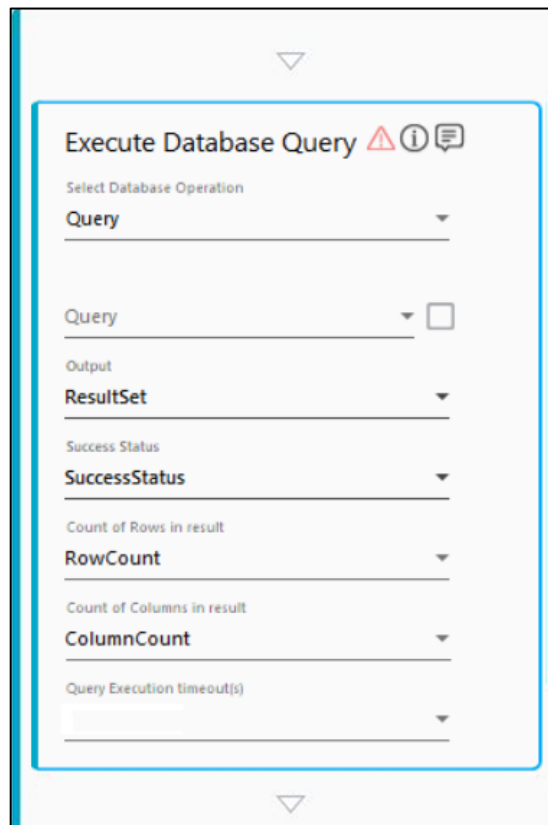
1. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
2. Drag the **Application** activity and drop on to the Flowchart designer on the Canvas



3. In the Application Type list, select **DatabaseApps**. You must have at least one application added for it to appear in the list.
4. In the Select An Application list, select the database application you want to perform automation on. Alternatively, you can add a new excel application at this point of time. To add the database application:
 - a. In the Select An Application list, click Add New Application. The Add New Application dialog box appears.

- b. In the **Application Name** field, enter a desired name of the application.
- c. In the **DbType** list, select the type of database you want to add. Supported types are:
 - SQL
 - Oracle
 - MySQL
 - Excel
- d. If **DbType** selected is **SQL**, select the **Authentication Type** based on your settings. Enter the **Server** and **DataBase Name** related to your SQL server.

- e. If **DbType** selected is **Oracle** or **MySQL**, enter the **Server**, **Port** and **DataBase Name** related to the database type selected.
 - f. If **DbType** selected is **Excel**, enter the path of the excel application saved on the system.
 - g. Click **SAVE**. The database application gets added.
5. Double click the **Application** activity, drag the **Execute DB Query** activity and drop inside the Application activity.



6. In the **Select Database Operation** list, select the type of database operation you want to perform. This drop-down appears only if the database application configured is of type **SQL**, **MYSQL** or **Oracle**. Available options are:
- **StoredProcedure**- a group of one or more line of statements or query for storing, manipulating or retrieving data from a database.
 - **Query**- a single line of statement or query for storing, manipulating or retrieving data from a database.
7. In the **Query** list, select the parameter to which you have assigned the query.
8. By default:
- a. In the **Output** field, **ResultSet** argument gets created that stores the output of the entered query.
 - b. In the **Success** Status field, **SuccessStatus** argument gets created that returns the status of the entered query.
 - c. In the **Count of Rows in result** field, **RowCount** argument gets created that stores the total number of rows present in the output of the entered query.
 - d. In the **Count of Columns in result** field, **ColumnCount** argument gets created that stores the total number of columns present in the output of the query you entered.

9. In the **Query Execution timeout(s)** list, select the parameter to the query execution timeout value in seconds. If this field is left blank, 30 secs is set as the default value of the query timeout. This field is available only for the SQL database application.

The fields are configured and the Application activity for Application Type-DatabaseApps is created.

Database Application Properties

The properties of a database application are listed in the following table and can be edited in the Property grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes, the application ignores any error while executing the activity. If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No.
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as the name of the selected application. You can change the name as required.

Excel Applications

This activity allows you to automate the redundant and tedious task performed on an excel application such as reading and writing data, data extraction and migration, comparing data in columns or rows, executing analysis reports, integrating with other applications and databases and so on.

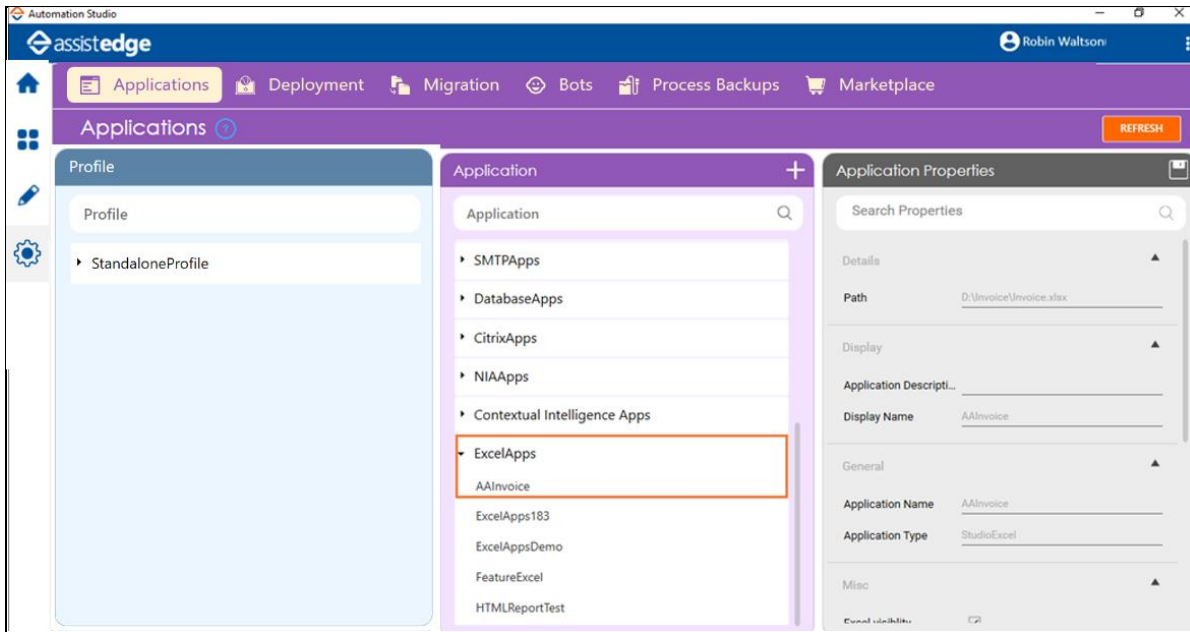
You can automate the cells and worksheet of the excel file using the EXCEL MODE of the Multimodal Interface. Additionally,

WINDOWS MODE and IMAGE MODE are also available which is useful to automate the ribbon and menu controls of the excel file.

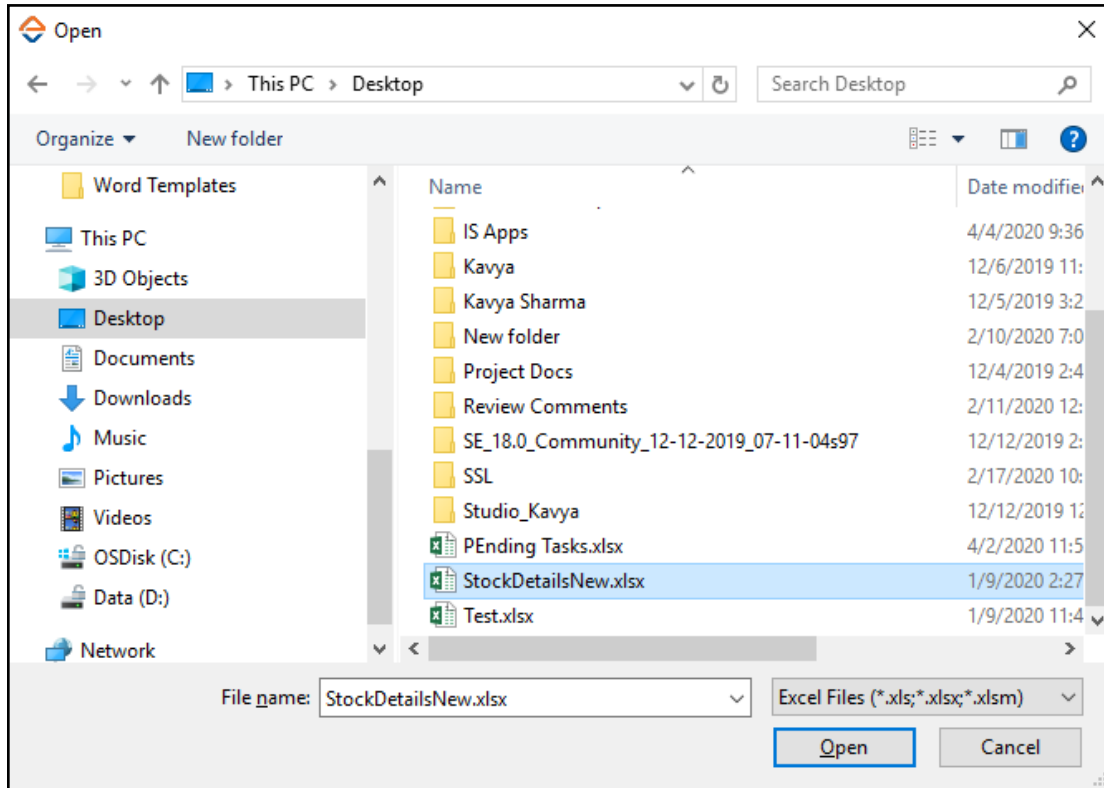
Prerequisite: Configuring Excel Application

Configure the excel application in Automation Studio where you want to perform the automation. Below are the minimum required properties for configuring the intended excel application. If you want to define remaining properties, refer [ExcelApps](#) in the Admin Menu.

1. In the **Admin** menu, add an application of **Application Type- ExcelApps**. The Add **Application** dialog box opens.



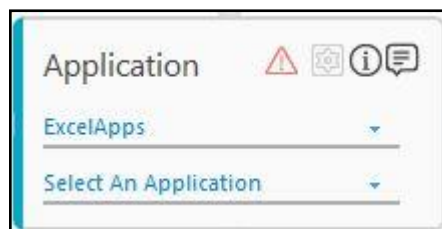
2. In the **Application Name** field, enter a desired name of excel application you want to add. The name must not contain any special character or space.
3. Click **ADD**. Navigate and select the excel file you want to configure. Click **Open**.



The ExcelApps application is configured.

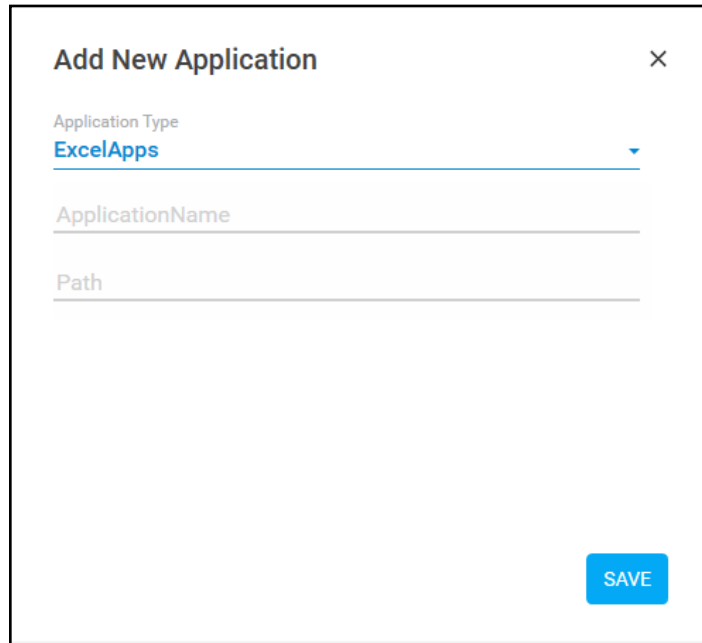
Using Excel Application Activity


1. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
2. Drag the **Application** activity and drop on to the Flowchart designer on the Canvas

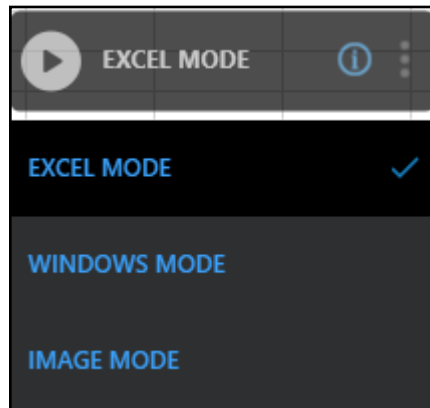




3. From the Application Type list, select **ExcelApps**. You must have at least one application added for it to appear in the list.
4. In the Select An Application list, select the configured excel file that you want to automate. Alternatively, you can add a new excel application at this point of time. To add an excel file:

- a. In the Select An Application list, click **Add New Application**. The **Add New Application** dialog box appears.

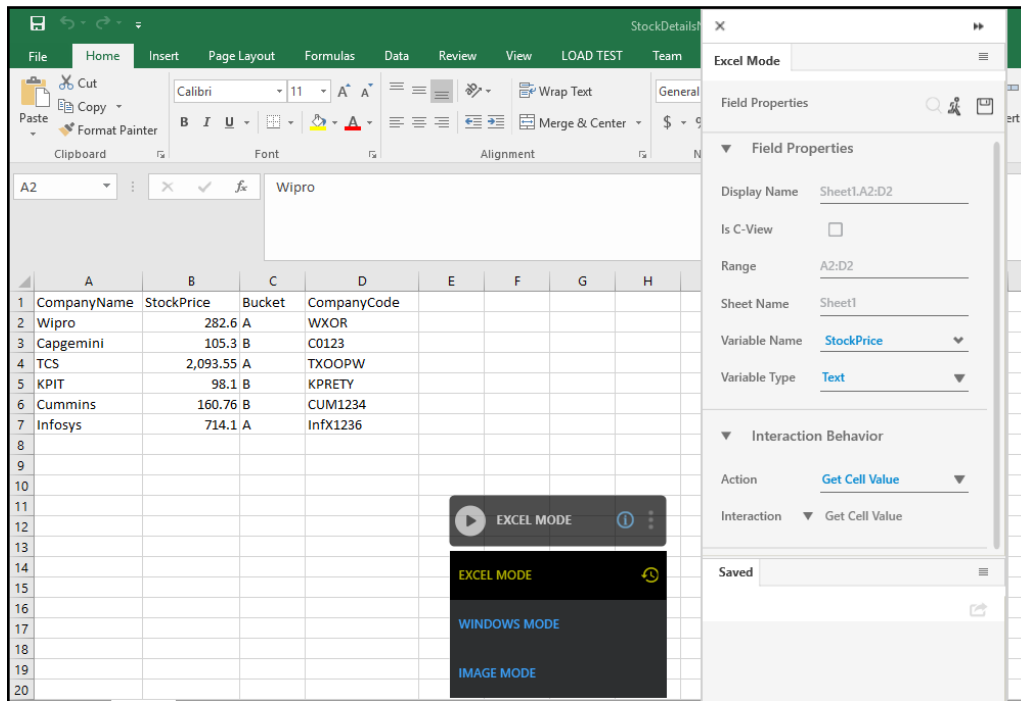



- b. In the **Application Name** field, enter a desired name of the application.
- c. In the **Path** field, enter the path of the excel file (along with the file name) available on the system.
- d. Click **SAVE**. The excel file is added.
5. Click the  (Settings) icon to launch the EXCEL MODE panel.




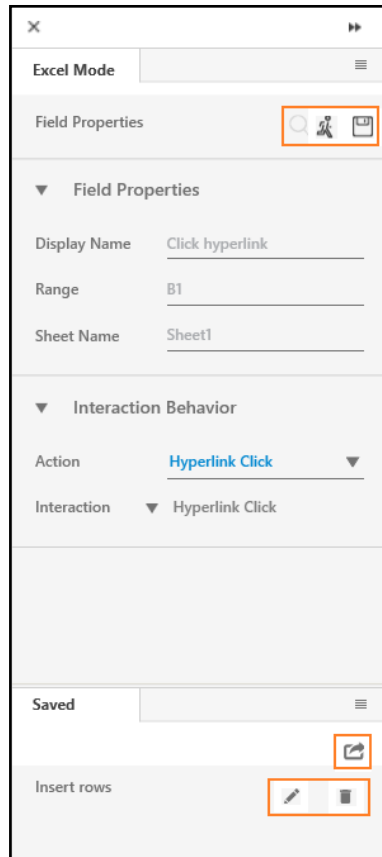
6. Click the  (Play) icon, to start capturing the automation steps that you want to perform within the excel file. The excel file appears and the Play icon changes to the  (Recording) icon. You can switch among **EXCEL MODE**, **WINDOWS MODE** and **IMAGE MODE** based on the requirement.

- Click the area that you want to capture as part of the automation process workflow. The Field Properties panel of the Excel Mode appears. The fields that are mandatory are highlighted with red box.



- In the Action list, select the action that you want to perform on the captured area. Other mandatory fields changes depending on your selection from the drop-down list.
- Enter details of all the mandatory fields and other relevant fields as per your requirement.
- Click the  (Save) icon to configure the fields.

11. Repeat steps 6 through 10 to capture all the steps involved in the automation process workflow. With the help of  (Run **Now**) icon, you can perform the action related to the configured field to see if the intended configuration is done.



The configured field appears in the Saved pane of the Field Properties panel. You can edit or delete these steps in the Saved pane.

12. Click the  (Save configured field and return to studio) icon to save the field that you configured and return to the Studio menu.

The fields are configured and the Application activity for Application Type- ExcelApps is created.

Excel Mode Field Properties

The properties of Excel Mode are listed in the following table and can be edited in the Field Properties panel.

Property Name	Usage
Field Properties	
Display Name	User specified display name of the excel element selected.
Range	A single cell or the collection of cells. It can be collection of two or more cells and not necessarily to be adjacent to each other.
Sheet Name	The name of the excel sheet to automate.
Appears if Action selected is Get Cell Value and Search .	
Variable Name	User defined name of the variable that stores the captured value.
Variable Type	Type of the variable defined. Available options are: <ul style="list-style-type: none"> ▪ Text - A sequence of character, either as a constant or a variable. ▪ Numeric - An integer type that range from negative through positive. ▪ Decimal - An exact numeric value defined by its precision and scale. ▪ Boolean - A data type used for making decision. Can have only two values- true or false. ▪ DataTable - A data type that stores tabular data.
Interaction Behavior	
Action	Actions that can be performed on the excel application. Available options are: <ul style="list-style-type: none"> ▪ Copy- Copies the selected area from the excel to the clipboard. ▪ Run Macro- Runs the required macro in the selected area. ▪ Get Cell Value- Extracts the value of the selected cell. ▪ Hyperlink Click- clicks the hyperlink available in the excel. The cell should have the hyperlink already present ▪ Paste- Pastes the copied content or the content available in the clipboard to the selected area. ▪ Search- Searches for the required content in the selected area. ▪ Set Text- Sets text on the specified area and stores it into a variable. You must define the variable in Variable panel of the Parameter bar before you start capturing the element. ▪ Insert Column- Inserts column at the specified location in the excel. ▪ Insert Row- Inserts row at the specified location in the excel.

Property Name	Usage
	<ul style="list-style-type: none"> ▪ Remove Duplicate- Removes the duplicates from the selected range. ▪ Rename Worksheet- Renames the defined worksheet. ▪ Merge Cells- Merges the selected range cells in an Excel sheet. In case there is data available in selected cells, then content of the top left cell appears in the merged cell. The content of the remaining cells is deleted at the time of merging. ▪ Get Cell Formula- Retrieves the formula from the specified cell. ▪ Delete Worksheet- Deletes the specified sheet from Excel file. ▪ Add Worksheet- Creates a new worksheet in Excel file. ▪ Refresh Pivot- Refreshes the pivot table in Excel file. ▪ Sort Data- Sorts the specified column data in ascending/descending order. ▪ Autofill Data- Auto populates the data in the cell. ▪ Copy Worksheet- Copies the specified sheet in Excel file. ▪ Delete Operations- Deletes cells, row, or column at the specified location in Excel file. ▪ Add DataTable- Adds data table in an Excel worksheet or add data from the data table at a specified location in Excel file. ▪ Apply Filter- Applies filter in the specified column in Excel file.
Interaction	The corresponding interaction set against the selected action.
Appears if Action selected is Run Macro .	
Macro Argument	The arguments which you need to pass to the macro to execute.
Macro Name	<p>The name of the macro that you want to run.</p> <p>You must define a parameter in the Parameter bar if you want to pass the macro as an argument or as a variable. Select the check box beside the Macro Name field and then enter the name of the macro if you want to pass the macro as a default macro.</p>
Appears if Action selected is Paste .	
Paste	<p>The options to paste content in the excel sheet. Available options are:</p> <ul style="list-style-type: none"> ▪ Paste - Pastes the entire content. ▪ Paste Formulas - Pastes only the formulas without any formatting of the copied content. ▪ Paste Formulas and Number Formats - Pastes the formulas and number formatting of the copied content. ▪ Paste Values and Number Formats - Pastes values and formatting of the

Property Name	Usage
	copied content.
Appears if Action selected as Search .	
Search Type	The type of search you want to perform in excel sheet. Available options are: <ul style="list-style-type: none"> ▪ ColumnLookup - Searches for the required content in the specified column range. ▪ RowLookup - Searches for the required content in the specified row range.
Search Range	The criteria to select the range to search for the required content. Available options are: <ul style="list-style-type: none"> ▪ ConfiguredRange - Searches for the required content based on the cells defined in the Range field. ▪ UsedRange - Searches for the required content based on the row or column name specified in the Search Location field.
Search Value	The content for which the corresponding value must be searched based on the Search Range field.
Search Location	The row or column name where the search must be done.
Fetch Result From	The row or the column name corresponding to the Search Location field from where the search value must be retrieved.
Appears if Action selected is Set Text .	
Column Separator	The separator to insert the values in the column like ' (coma)', ';' (semicolon)' and others.
Data to insert	The value that you want to set in the selected cell or range of cells. The values are inserted based on the defined Column Separator or Row Separator . You must define a parameter in the Parameter bar if you want to pass the values as an argument or as a variable. Select the check box beside the Data to insert field and then enter default values with a defined separator.
Row Separator	The separator to insert the values in the rows like ' (coma)', ';' (semicolon)' and others.
Appears if Action selected is Insert Column .	
Column	The default column name where the new column would be inserted.
Appears if Action selected is Insert Column .	
Row number	The row number where the new row would be inserted.
Appears if Action selected is Remove Duplicate .	
Column	Define the index of the required column. For multiple columns, separate the indexes with delimiters. By default, the delimiter is defined as comma (,). <div style="background-color: #0070c0; color: white; padding: 5px;"> <p>Note: The indexing of the column is dependent on the Range selected in Field Properties. If the range is selected as C:E and you want to remove duplicates from column D, then the</p> </div>

Property Name	Usage
	<p>indexing for column D is 2.</p> <p>You must define a parameter in the Parameter bar if you want to pass the column index as an argument or as a variable. Select the check box beside the Column field and then enter pivot column index if you want to pass the column as default.</p>
Has Headers	<p>Select the required option from the list. The available options are:</p> <ul style="list-style-type: none"> ▪ Guess- Select this option if headers availability is not confirmed in Excel sheet. ▪ No- Select this option if headers are not available in Excel sheet. ▪ Yes- Select this option if headers are available in Excel sheet.
Index Delimiters	<p>Delimiters are used to separate multiple indexes defined in the Column field. By default, the delimiter is defined as comma (,). You can change a predefined delimiter if required.</p>
Appears if Action selected is Rename Worksheet .	
Rename Sheet as	<p>Specify the new name for the worksheet.</p> <p>You must define a parameter in the Parameter bar if you want to pass the name as an argument or as a variable. Select the check box beside the Rename Sheet as field, and then enter the name of the sheet if you want to pass the sheet name as default.</p>
Appears if Action selected is Refresh Pivot .	
Refresh All	<p>Select this option to refresh all the available pivot tables in Excel file.</p>
PivotTable Name	<p>Specify the pivot table name.</p> <p>You must define a parameter in the Parameter bar if you want to pass the pivot table name as an argument or as a variable. Select the check box beside the PivotTable Name field, and then enter pivot table name if you want to pass the pivot table name as default.</p>
Appears if Action selected is Sort Data .	
Has Header	<p>Select the required option from the list. By default, it is set as Guess. The available options are:</p> <ul style="list-style-type: none"> ▪ Guess- Select this option if headers availability is not confirmed in Excel sheet. ▪ No- Select this option if headers are not available in Excel sheet. ▪ Yes- Select this option if headers are available in Excel sheet.
Sort Order	<p>Select the sorting order for a specified column from the list. By default, it is set as Ascending. The available options are:</p> <p>Ascending</p> <p>Descending</p>
Sort Worksheet	<p>Select this option if you want to move the entire row with the specified column. This option retains the data intact in specified Excel sheet.</p>

Property Name	Usage
Sorting Column	<p>Define the column name for sorting.</p> <p>You must define a parameter in the Parameter bar if you want to pass the sorting column as an argument or as a variable. Select the check box beside the Sorting Column field and then enter the column name if you want to pass the column as default.</p>
Appears if Action selected is Autofill Data .	
Autofill Type	<p>Select the auto-fill type from the list. By default, it is set as Default. The available options are:</p> <ul style="list-style-type: none"> ▪ Default- Maintains the default auto-fill functionality. ▪ Copy Value & Formats- Copies the value with formatting. ▪ Fill Series- Auto fills the defined series in the selected cell. ▪ Fill Values- Auto fills the values in the cell. If the formula is defined, then auto-fills the value as the defined formula. ▪ Fill Formats- Copies the format and auto-fills the copied format in the selected cell. ▪ Fill Days- Auto-fills the days/date in the selected cell. ▪ Fill Months- Auto-fills the months in the selected cell. ▪ Fill Weekdays- Auto-fills the weekdays in the selected cell. ▪ Fill Year- Auto-fills the year in the selected cell.
Destination Range	<p>Define the cells start point and end point.</p> <p>You must define a parameter in the Parameter bar if you want to pass the destination range as an argument or as a variable. Select the check box beside the Destination Range field and then enter the required range if you want to pass the destination range as default.</p>

Appears if Action selected is Copy Worksheet.	
Copied SheetName	<p>Define a name for the copied sheet or use as default. If named as default, then the functionality will work as per Microsoft Excel.</p> <p>You must define a parameter in the Parameter bar if you want to pass the sheet name as an argument or as a variable. Select the check box beside the Copied SheetName field and then enter the name of the sheet if you want to pass the sheet name as default</p>
Target Workbook	<p>Define the path of the targeted Excel file if you want to copy the sheet in a different workbook.</p> <p>You must define a parameter in the Parameter bar if you want to pass the targeted workbook as an argument or as a variable. Select the check box beside the Target Workbook field and then enter the path of the targeted Excel file if you want to pass the targeted workbook as default.</p>
Appears if Action selected is Delete Operations .	

Delete Row	Deletes entire row at the specified location in Excel sheet.
Delete Column	Deletes entire column at the specified location in Excel sheet.
Delete Range	<p>Deletes cells at the specified location in Excel sheet.</p> <p>Select the behavior which is executed after deleting the selected range. There are two types of behavior available in the Shift Action list.</p> <ul style="list-style-type: none"> ▪ Shift cells up- Shifts all the available cells up after deleting the selected range. ▪ Shift cells left- Shifts all the available cells towards the left after deleting the selected range.
Appears if Action selected is Add DataTable .	
DataTable	<p>Define the data table name.</p> <p>You must define a parameter in the Parameter bar if you want to pass the data table name as an argument or as a variable. Select the check box beside the DataTable field, and then enter the data table name if you want to pass the data table as default.</p>
Include Headers	Select this option if headers need to be copied in the specified Excel sheet.
Appears if Action selected is Apply Filter .	
Data Type	<p>Select the data type from the list. By default, it is set as Number. The available options are:</p> <ul style="list-style-type: none"> ▪ String- A sequence of characters, either as a constant or a variable. ▪ Number- An integer type that range from negative through positive.
Number Filter	<p>Select the filter from the list for the number data type. By default, it is set as Equal. The available options are:</p> <ul style="list-style-type: none"> ▪ Equal- Retains the cell based on the exact match. ▪ Does Not Equal- Retains the cell if value does not match. ▪ Greater Than- Retains the cell if the value is greater than the defined filter value. ▪ Greater Than Or Equal- Retains the cell if the value is greater than or equal to the defined filter value. ▪ Less Than- Retains the cell if the value is less than the defined filter value. ▪ Less Than Or Equal- Retains the cell if the value is less than or equal to the defined filter value. ▪ Between- Retains the cell if the value lies within the defined range. ▪ Top10- Retains the 10 highest value cells in the specified column. ▪ Above Average- Calculates the column average and retains the above-average cells.

	<ul style="list-style-type: none"> ▪ Below Average- Calculates the column average and retains the below-average cells.
String Filter	<p>Select the filter from the list for string data type. The available options are:</p> <ul style="list-style-type: none"> ▪ Equal- Retains the cell based on the exact match. ▪ Does Not Equal- Retains the cell if value does not match. ▪ Begins With- Retains the cell if starting string matches to the defined filter value. ▪ Ends With- Retains the cell if ending string matches to the defined filter value. ▪ Contains- Retains the cell if defined filter value is present in the string. ▪ Does Not Contains- Retains the cell if defined filter value is not present in the string.
Column Number	<p>Specify the required column number to apply filters in that particular column. The column number starts from 0.</p> <p>You must define a parameter in the Parameter bar if you want to pass the value as an argument or as a variable. Select the check box beside the Column Number field and then enter the column number if you want to pass the column number as default.</p>
Filter Value	<p>Define a value for the filter.</p> <p>You must define a parameter in the Parameter bar if you want to pass the filter value as an argument or as a variable. Select the check box beside the Filter value field, and then enter the filter value if you want to pass the filter value as default.</p>

Excel Application Properties

The properties of an excel application are listed in the following table and can be edited in the Property grid on the right pane.

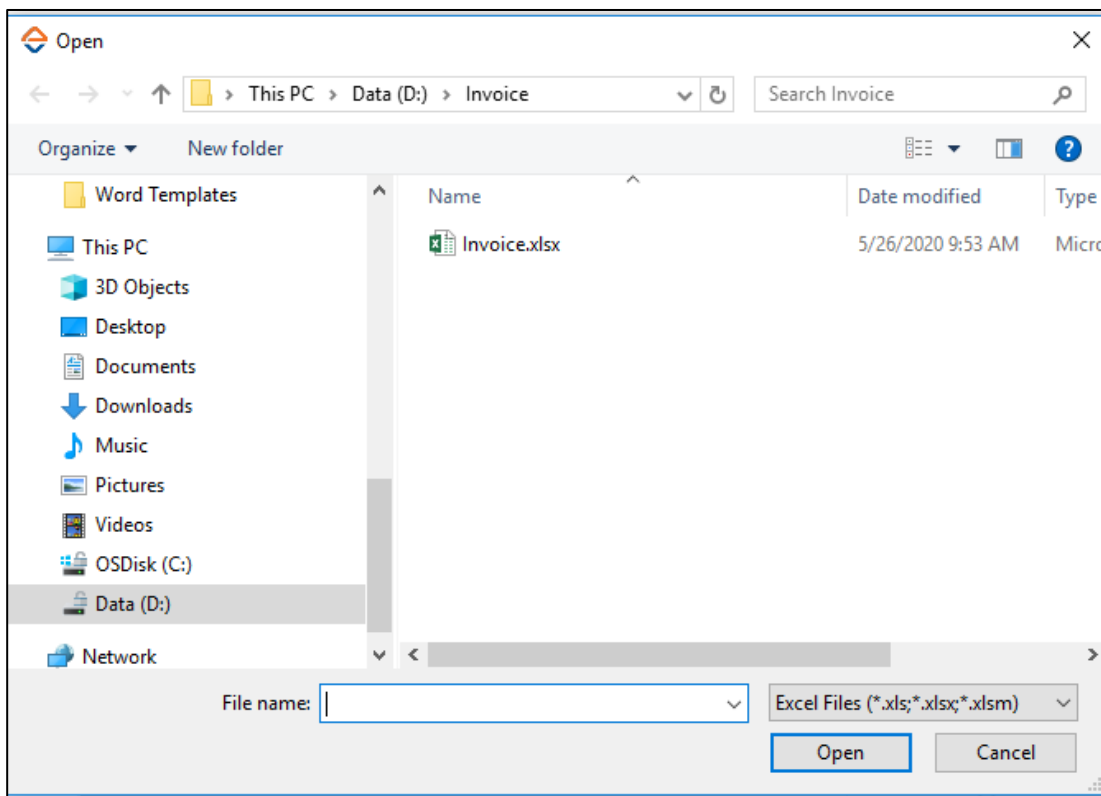
Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes, the application ignores any error while executing the activity. If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No.
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as the name of the selected application. You can change the name as required.

Example of Excel Application

Let's create a process to extract the total amount of the invoice saved in an excel application.

Prerequisite:

1. In the **Admin** menu, add an application of **Application Type- ExcelApps**.
2. In the **Application Name** field, enter a desired name of the excel application you want to add. The name must not contain any special character or space.
3. Click **ADD**. Navigate and select the excel file. Click **Open**.

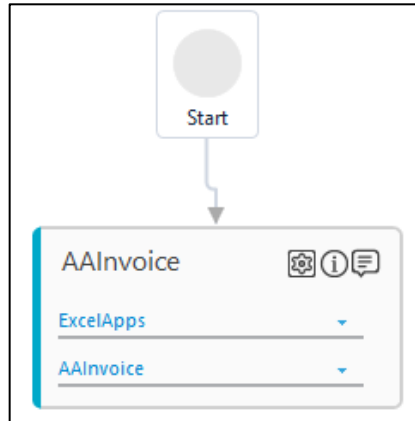


A message for successful addition of the excel application is displayed.

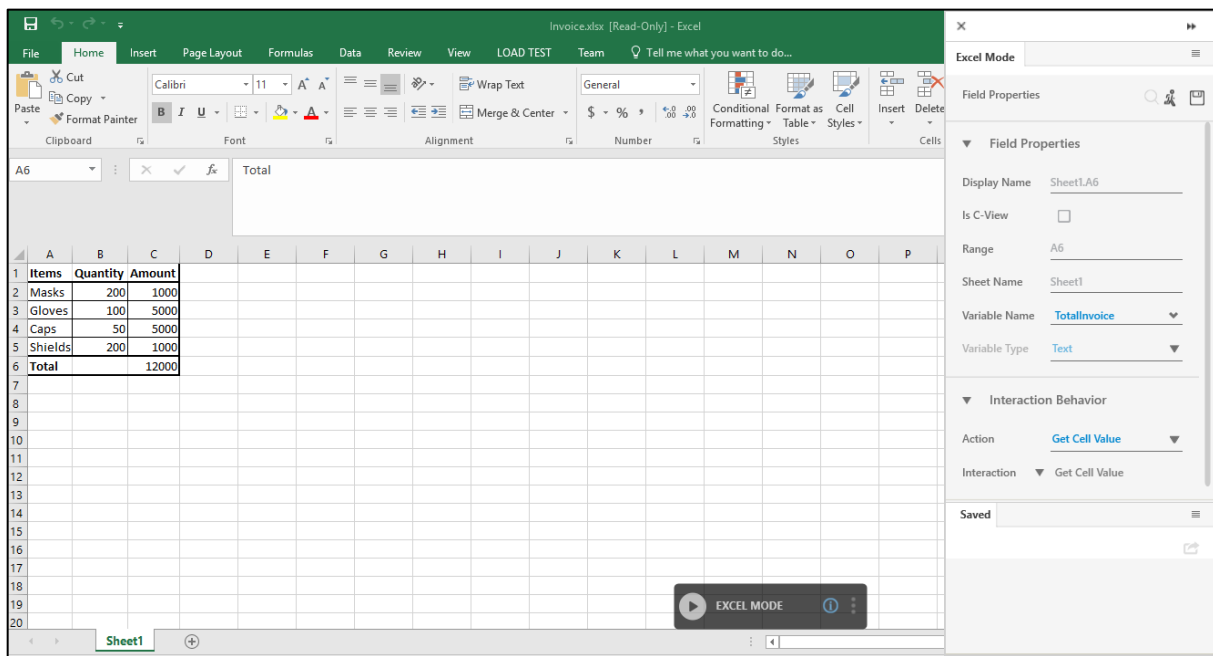
To automate the task of extracting cell values:




1. Create a new process.
2. Create an **In** arguments, **TotalInvoice** of **String** type and **Amount** of **Int32** type. These parameters are used to store the extracted cell values.
3. From the Canvas Tools panel, add Application activity to the Flowchart designer on the Canvas
4. In the Application Type list, select ExcelApps.



- In the **Select an Application** list, select the configured excel file.



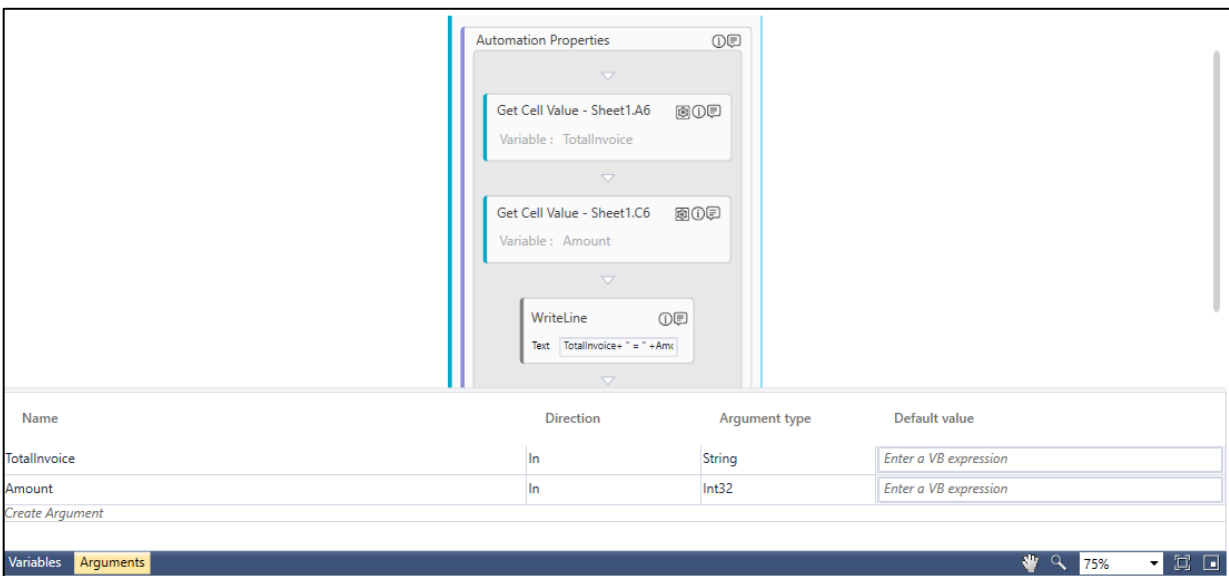
- Click the  icon to launch the **EXCEL MODE** of the Multimodal panel. The excel file appears.



- Click the  (Play) icon, and then click the A6 cell to extract the required cell value. The Field Properties panel of the Excel Mode appears. The fields that are mandatory are highlighted with red box.
- In the **Action** list, select **Get Cell Value**.
- In **Variable Name** list, select **TotalInvoice** argument created in the **Parameter** bar. Other fields get auto populated based on the cell details.
- Click the  (**Save**) icon to configure the details captured. The saved details start appearing in the **Saved** tab of **Field Properties** bar.
- Click the  (Play) icon, and then click the C6 cell to extract the required cell value. The Field Properties panel of the Excel Mode appears. The fields that are mandatory are highlighted with red box.
- In the **Action** list, select **Get Cell Value**.

13. In **Variable Name** list, select the **Amount** argument created in the **Parameter** bar.
14. In the **Variable** Type list, select **Numeric**.
15. Click the  (**Save**) icon to configure the details captured. The saved details start appearing in the **Saved** tab of **Field Properties** bar.
16. Click the  (Save configured field and return to studio) icon to save the configured field and return to the Studio menu. The automation process workflow to configure the search process is created.

To view the output in Automation Studio, let's add WriteLine activity. You can assign this process to a robot, if you want to execute this process outside Automation Studio.



The screenshot shows the Automation Properties panel with three activities: 'Get Cell Value - Sheet1.A6' (Variable: TotalInvoice), 'Get Cell Value - Sheet1.C6' (Variable: Amount), and 'WriteLine' (Text: TotalInvoice+ " = " +Amc). Below the panel is a table with the following data:

Name	Direction	Argument type	Default value
TotalInvoice	In	String	Enter a VB expression
Amount	In	Int32	Enter a VB expression
Create Argument			

At the bottom, there are tabs for 'Variables' and 'Arguments', and a status bar showing '75%' zoom.

17. Add a **WriteLine** activity below the **Get Cell Value** block.
18. Save the process.
19. Setup the environment and then perform test run. Following is the output displayed in the console:

Output

Total=12000

Mainframe Application

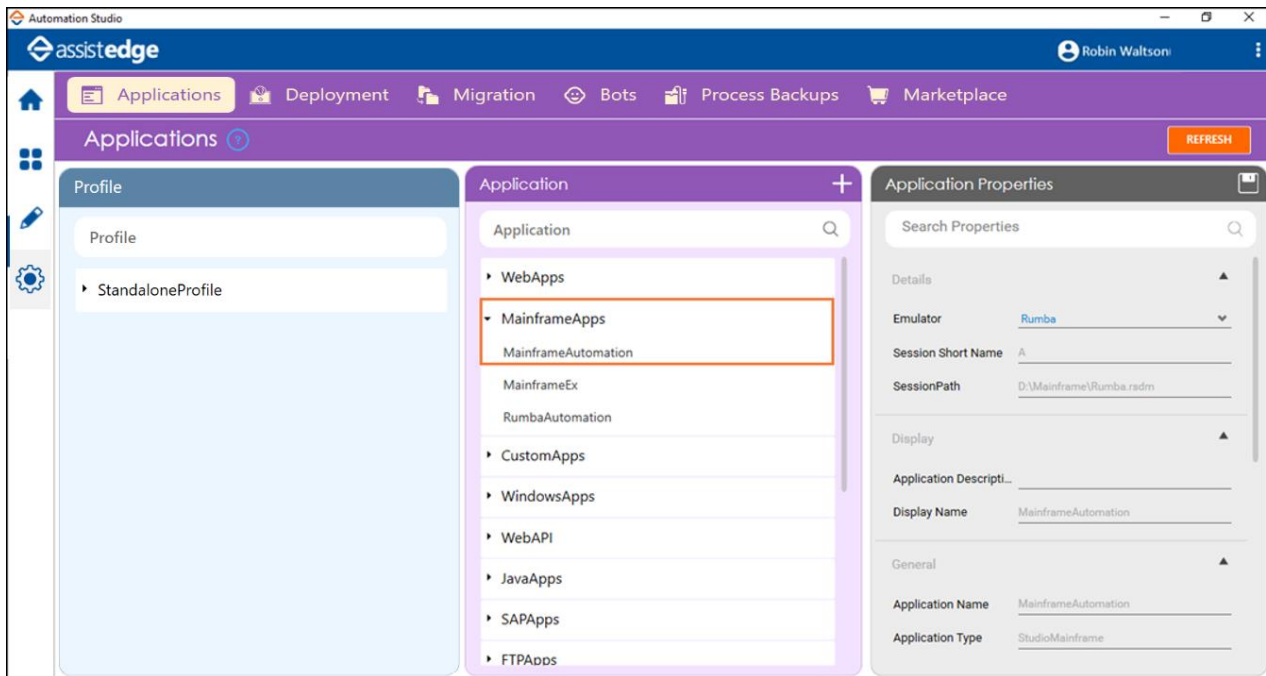
This activity allows you to automate the actions performed on the mainframe application such as processing bulk data, financial transactions, inventory control and other critical services utilized mainly by large organizations.


In Automation Studio, field configuration of the mainframe application is done with the help of the coordinates of the selected field. The Multimodal Interface uses the coordinated of the row and the column to identify the field.

Prerequisite: Configuring Mainframe Application

Configure the mainframe application in Automation Studio where you want to perform the automation. Below are the minimum required properties for configuring the intended mainframe application. If you want to define remaining properties, refer [MainframeApps](#) in the Admin Menu.

1. In the **Admin** menu, add an application of **Application Type- MainframeApps**.



2. In the Application Properties panel:
 - Select the required emulator in the **Emulator** list.
 - Enter the short name of the session as set in the selected emulator in the **Session Short Name** field. This field appears only if **Emulator** selected is **Rumba** or **EHLAPI**.
 - Enter the path of the session file in the **SessionPath** field, that contains all the necessary details required to connect to the mainframe terminal.
3. Click the  (**Save Properties**) icon to save the application.

The mainframe application is configured.

Performing Mainframe Automation using Passport Emulator

Passport emulator enables Windows users to securely connect to applications running on IBM Mainframe systems. To use this feature, the assembly files required are:

Component	Version
AxInterop.PASSTERMLib.dll	Latest version
Interop.PASSHIOLib.dll	Latest version
Interop.PASSTERMLib.dll	Latest version

To copy assembly files:

1. Locate the above mentioned files that is available during the Passport Emulator installation. For more details about the stated emulator, refer [Passport Emulator](#).
2. Copy the respective assembly files to the locations mentioned below:
 - client-tools > AutomationStudio > **bin** > Plugins > Passport. If you download/access Automation Studio from the Admin module, you must save the required DLLs at %localappdata% > EdgeVerve > AutomationStudio > bin > Plugins > Passport.
 - client-tools > AutomationRuntime > AEAttendedRPA > Plugins > Passport
 - client-tools > AutomationRuntime > Engage > Plugins > Passport
 - client-tools > AutomationRuntime > Robot > RobotAgent > RobotSE > Plugins > Passport

The assembly files are copied.

Performing Mainframe Automation using Mainframe Automation Reflection

Automation of Mainframe applications can be done using Mainframe Automation Reflection. To use this feature following assembly files are required:

Component	Version
Attachmate.Reflection.dll	16.1
Attachmate.Reflection.Emulation.dll	16.1
Attachmate.Reflection.Emulation.IbmHosts.dll	16.1
Attachmate.Reflection.Emulation.OpenSystems.dll	16.1
Attachmate.Reflection.Framework.dll	16.1

To copy assembly file:

1. Locate the above mentioned files that is available during the Reflection Emulator installation. For more details about the stated emulator, refer [Reflection Emulator](#).
2. Copy all the above mentioned assembly files at the locations mentioned below:

- client-tools > AutomationRuntime > AEAttendedRPA > ThirdPartyDLLS
- client-tools > AutomationRuntime > Engage > ThirdPartyDLLS
- client-tools > AutomationRuntime > Robot > RobotAgent > RobotSE > ThirdPartyDLLS
- client-tools > AutomationStudio > **bin** > ThirdPartyDLLS. If you download/access Automation Studio from the Admin module, you must save the required DLLs at %localappdata% > EdgeVerve > AutomationStudio > bin ThirdPARTyDLLS.

The assembly files are copied.

Performing Mainframe Automation using Attachmate Extra Emulator

Automation of Mainframe applications can be done using Attachmate Extra. Mainframe Automation assembly that comes with the licensed version of Attachmate Extra is required to perform the automation of mainframe application. To use this feature following assembly file is required:

Component	Version
Interop.Extra.dll	9.3.1

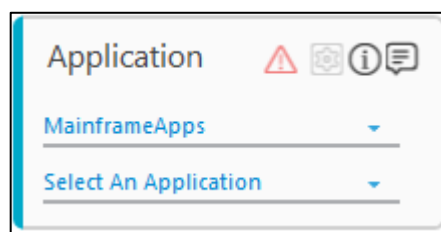
To copy assembly file:

1. Locate the above mentioned file that is available during the Attachmate Extra client installation. For more details about the stated emulator, refer [Attachmate Extra Emulator](#).
2. Copy the assembly file to the locations mentioned below:
 - client-tools > AutomationRuntime > AEAttendedRPA
 - client-tools > AutomationRuntime > Engage
 - client-tools > AutomationRuntime > Robot > RobotAgent > RobotSE

The assembly file is copied.


Using Mainframe Application Activity

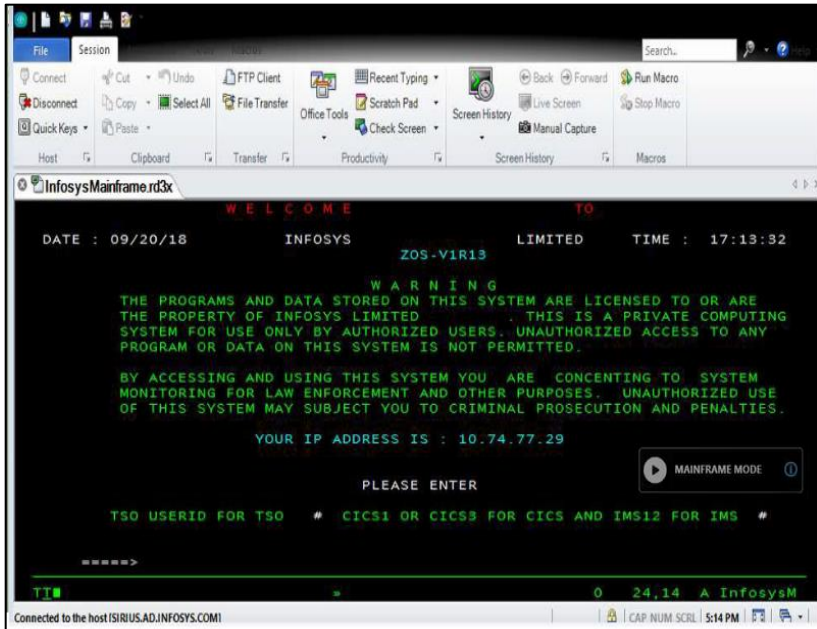
1. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
2. Drag the **Application** activity and drop on to the Flowchart designer on the **Canvas** area. The validation error symbol disappears when required inputs are provided in the correct format.




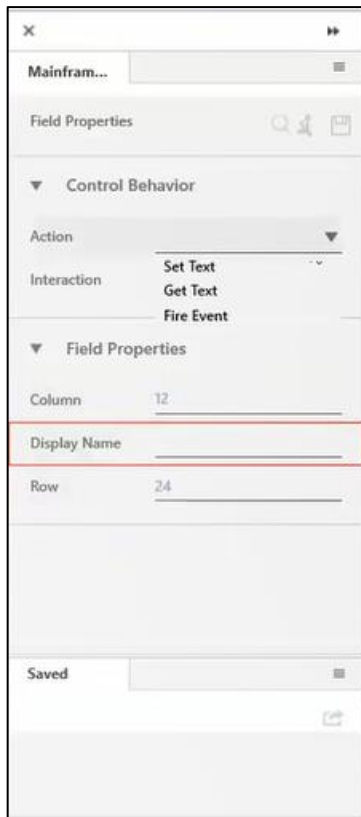
3. From the Application Type list, select **Mainframeapps**. You must have at least one application added for it to appear in the list.
4. In the Select An Application list, select the configured mainframe application that you want to automate. Alternatively, you can add a new mainframe application at this point of time. To add a mainframe application:
 - a. In the Select An Application list, click **Add New Application**. The **Add New Application** dialog box appears.



- b. In the **Application Name** field, enter a desired name of the mainframe application.
- c. In the **Emulator** list, select the emulator you want to configure. The available options are:
 - AttachmateExtra
 - Passport
 - ReflectionUnixVMS
 - AttachmateReflectionIBM
 - ReflectionOpenSystem
 - Rumba
 - EHLLAPI
- d. In the **Session Path** field, enter the path of the session file that contains all the necessary details required to connect to the mainframe terminal.
- e. In the **SessionShortName** field, enter the short name of the session, as set in the emulator. This field appears only if **Emulator** selected is **Rumba** or **EHLLAPI**.
- f. Click **SAVE**. The web application is added.

- Click the  (**Settings**) icon to launch the **MAINFRAME** MODE interface. The configured emulator of the mainframe application appears along with **MAINFRAME MODE** interface.



- Place the cursor at the required location and click the  (Play) icon to start capturing the automation steps that you want to perform within the mainframe application. The Field Properties panel of the **Mainframe Field Properties** appears. The fields that are mandatory are highlighted with red box.



7. In **Action** list, select the action that you want to perform on the captured area. Other mandatory fields change depending on your selection from the drop-down list. Enter the required details. Refer [Field Properties](#) table to know more about the available fields and their respective properties.
8. Click the  (Save) icon to configure the fields.
9. Repeat steps 6 through 8 to capture all the steps involved in the process. The configured field is saved and starts appearing in the Saved pane of the Field Properties panel. You can edit or delete these steps in the Saved pane, if required.
10. Click the  (Save configured field and return to studio) icon to save the field that you configured and return to the Studio menu.

The fields configuration of the mainframe application is done.

Mainframe automation using IBM standard EHLAPI

Prerequisite - Emulator must implement the EHLAPI standard.

- To automate Mainframe terminals using EHLAPI, select Emulator as **EHLAPI** while adding mainframe applications.
- **Session Path** should contain the path of session file, which contains all the necessary details required to connect to the mainframe terminal.
- Provide the **Session Short** Name as configured in the emulator.

Configure Session Short Name in various emulators:

- **Rumba** – Open Rumba emulator and navigate to **Menu**→**Options**→**API**
- **Reflection** – Open reflection workspace and navigate to File->Settings->Document Settings->Configure Advanced Connection Settings.
- **Extra X-treme** – Open Extra X-treme and navigate to **Option**->**Global Preferences** under tab **Advanced**.
- **Bluezone** – Open Bluezone and navigate to **Session**->**Configure**->**API**

Recommendations

- Use Reset process instead of default reset.

Mainframe Mode Control Behavior and Field Descriptions

The properties of Mainframe Mode are listed in the following table:

Property Name	Usage
Control Behavior	
Action	It is the action or the interaction that can be performed on a mainframe application. Available options are:

Property Name	Usage
	<ul style="list-style-type: none"> ▪ Set Text - Sets a value at the selected position. ▪ Get Text - Retrieves the value from the selected position. ▪ Fire Event - Executes or triggers an action at the selected position.
Interaction	The corresponding interaction set against the selected action.
Appears if Action selected is Fire Event.	
Event Name	<p>The trigger that executes an action on the mainframe application. Available options are:</p> <ul style="list-style-type: none"> ▪ Enter - Performs the return action of the keyboard. ▪ Clear - Deletes the entire content. ▪ Delete - Performs Delete key action of the keyboard. ▪ Erase EOF - Deletes the entire content from cursor till the end of the field. ▪ Cursor Left - Moves the cursor to left. ▪ Left Tab - Performs Tab key action of the keyboard to move towards left. ▪ Right Tab - Performs Tab key action of the keyboard to move towards right. ▪ Cursor Up - Performs up arrow key action of the keyboard. ▪ Cursor Down - Performs down arrow key action of the keyboard. ▪ Home - Performs Home key action of the keyboard. ▪ Function Keys (F1- F12) - Performs selected function key action of the keyboard. ▪ End - Performs End key action of the keyboard.
Field Properties	
Column	The column coordinate of the selected field on the mainframe application.
Display Name	User specified display name of the action configured.
Row	The row coordinate of the selected field on the mainframe application.

9.6.2 Using Reuse Process

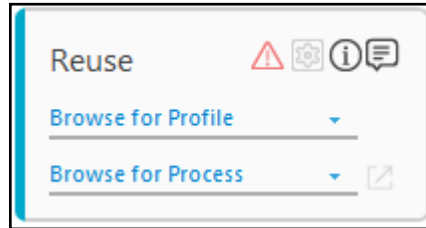
This activity allows you to reuse an existing automation process workflow or activity in a new process workflow.

Reusing an existing process helps you save time and effort required for recreating a new process.

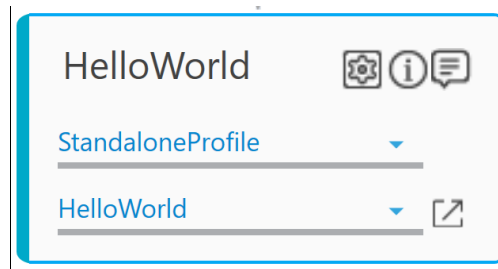
Note: Only a published process can be reused.


Using Reuse Process Activity

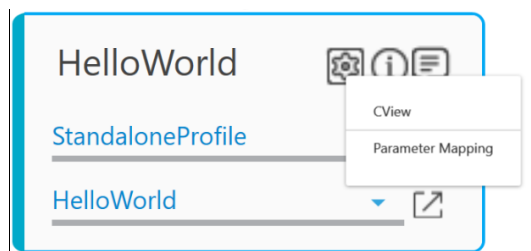
1. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
2. Drag the **Reuse Process** activity and drop on to the Flowchart designer on the Canvas




3. In the **Browse for Profile** list select the profile you want to reuse.

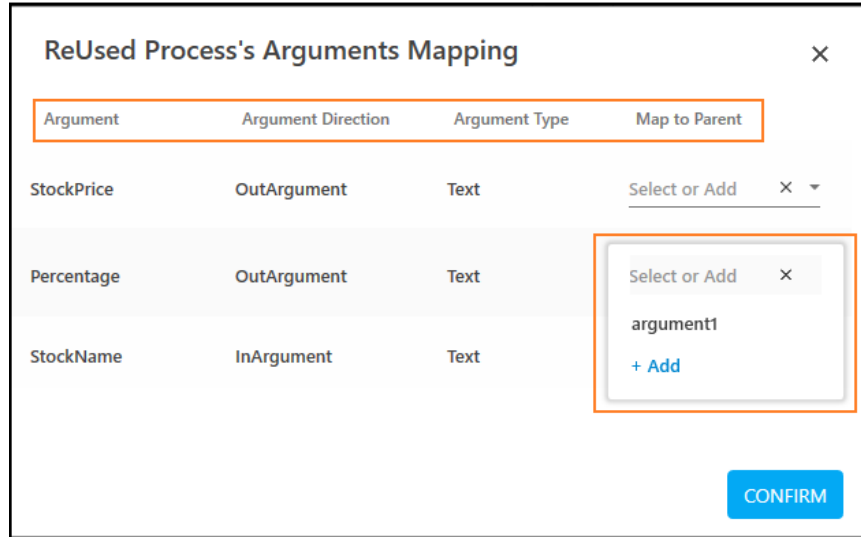


4. In the **Browse for Process** list, select the process you want to reuse in the current workflow. The process list displayed depends on the profile you have selected above. You can click the  (**Open Process**) icon open the selected process in the Flowchart designer are in the **Canvas**.



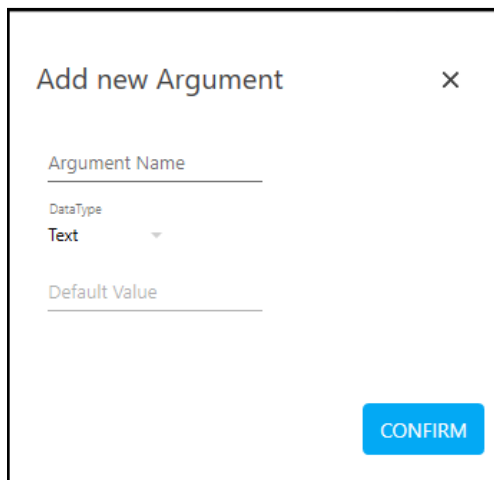
5. Select the filed configurations and then click **CONFIRM**.

6. Click the  (**Settings**) icon once again, and then click **Parameter Mapping** to align the parameters of the reused process with the parameters of the process where it is being used (also called as Parent process). The **ReUsed Process's Arguments Mapping** dialog box appears. See the sample mapping in the following screen shot:



The Argument, Argument Direction and Argument Type lists the details related to the parameters of the reused process.

7. In the **Map to Parent** list, select the parameter that you want to assign to the parent process parameter. Repeat this step for each of the parameter you want to map. If there is no parameter available in the parent process matching the direction or type of the current parameter, you can add a parameter. Follow the mentioned to steps to add the parameter:
 - a. Click **Select or Add** to select an existing parameter from the available list or create a new parameter. The **Add new Argument** dialog box appears.



- b. In the **Argument Name** field, enter a desired name.
- c. The **DataType** is set as the data type of the corresponding reused process parameter.
- d. In the **Default Value** field is enabled only when the **Argument Direction** of the corresponding reused process parameter is **In**. For **Out** or **InOut** direction, providing default value is not supported. Once the field is enabled **for In** parameter, enter a default value of the argument added, if required.

- e. Click **CONFIRM** to save the mapping.

The Reuse Process activity with the name of the process selected, is created.

Reuse Process Properties

The properties of a Reuse Process activity are listed in the following table and can be edited in the Property grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as the name of the selected process. You can change the name as required.

9.6.3 Using SignIn Process

This activity allows you to sign-in to the application for which a Sign In Process is already deployed.

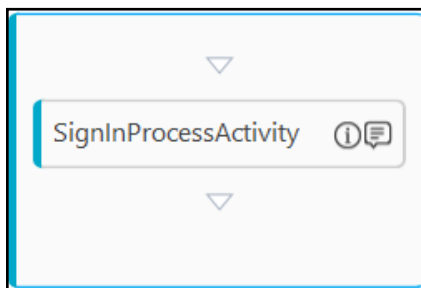
In some applications, sign out after a certain time interval may occur due to reasons like security, leaving the application idle for a longer duration, you can sign-in back to the application instance anywhere in the automation process workflow using this activity.

Note:

- This activity must be used inside the application for which the **Sign-In Process** is created.
- The **Sign-In Process** for the intended application must be available and deployed or the application does not get signed-in by this activity.

Using SignIn Process Activity

1. Create a process automation workflow and add an **Application** activity.
2. Double click the Application activity, drag the SignIn Process activity and drop inside the Application activity.



The SignIn Process activity with a default display name is created.

SignIn Process Properties

The properties of a SignIn Process activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity. If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as SignInProcessActivity. You can change the name as required.

9.6.4 Using Reset Process

This activity allows you reset the application for which a Reset Process is already deployed. Reset an application instance anywhere in the automation process workflow using this activity.

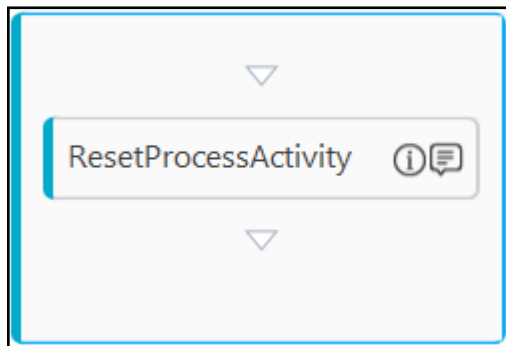
Resetting an application sets the application to the state where it was before running the automation. For example, if the automation process consists of a search process, reset clears the search value, search result and sets it back to the initial state.

Note:

- This activity must be used inside the application for which the **Reset Process** is created.
- The **Reset Process** for the intended application must be available and deployed or the application does not get reset by this activity.

Using Reuse Process Activity

1. Create a process automation workflow and add an **Application** activity. Select the required **Application Type** and **Application**. Reset process must already be created and deployed for the selected application.
2. Double click the Application activity, drag the Reset Process activity and drop inside the Application activity.



The Reset Process activity with a default display name is created.

Reset Process Properties

The properties of a Reset Process activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as ResetProcessActivity . You can change the name as required.

9.6.5 Process Kill

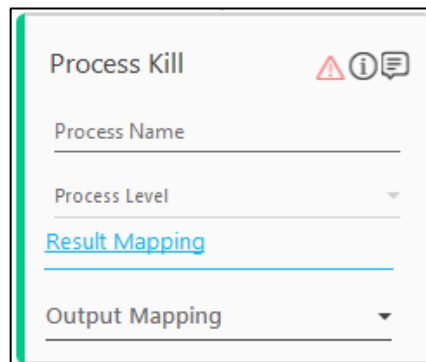
This activity allows you to terminate a specific process on the machine. Before process execution starts, there can be some applications running which may impact the process automation. You can kill those application to avoid any impact on the process automation. After process execution, there can be some residues of the application remaining which may cause high CPU and memory utilizations on the server. You can kill these applications to increase the performance and efficiency.

Process kill activity is used to kill any type of application running on the server. It can be used inside and outside of the application activity. It can kill application launched for the client-tools, user, system and for the applications not responding. Whereas, close Application activity is used to kill automation application. This is used inside Application activity. And it kills the application launched by client-tools. For example- Studio, robots, engage and so on. For more information, see Close Activity.

For example- When Java application is used in an automation process which in turn was launching many other associated applications. After process execution, java application was getting killed, but other applications were still running causing high CPU and memory utilizations and also impacting process automation. Using Process Kill activity, you can kill all the associated applications and resolve the related issues.

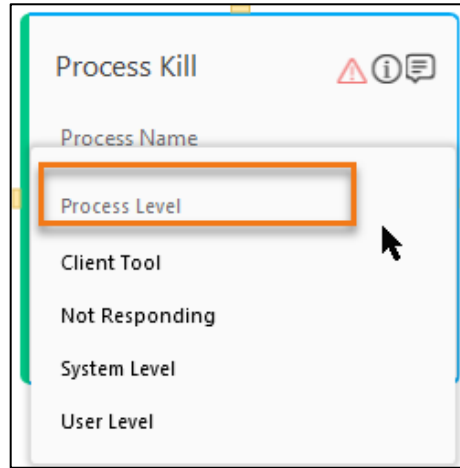
Using Process Kill

1. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
2. Drag the **Process Kill** activity and drop on to the Flowchart designer on the Canvas



3. In the **Process Name** field, specify the name of process which is required to be killed from the available application in the task manager details window. You can specify either the name of application or the name of application with its extension. For example: chrome or chrome.exe.

4. In the **Process Level** field, select the name of process level from the available list.



- **Client Tool:** Select this option to kill the processes that are launched with client-tools such as Automation Studio, Engage, bots and so on.
 - **Not Responding:** Select this option to kill the processes that are only in not responding state
 - **System Level:** Select this option to kill the processes on the system level irrespective of the logged user.
 - **User Level:** Select this option to kill the processes that are launched for a specific user.
5. Click the **Result Mapping** link, the Output Mapping list appears. Select the preferred parameter from the available parameter to map the status of the operation performed.
- a. Both Boolean and String type are supported.
 - b. You must pre-define the parameter to make it available for the selection.
6. If the process kill is successful, it returns with status as **true** or if the operations fail it returns with status as **false**. It can be used as an input for the next action that you want to perform depending upon the success or failure of the operation.
- a. In case, Result mapping is true then process is killed.
 - b. In case, Result mapping appears false, then process must have not found or not killed.

Process Kill Properties

The properties of a Process Kill activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity. If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as Process Kill . You can change the name as required.
Process Name	The process name of process which is required to be killed from the available application

9.6.6 Application Host

This activity helps to automate UI elements of a Windows/JAVA based host application or a pop-up window of the host application that may not get captured or identified using other methods like the [Multimodal Interface](#).

Automation Studio supports automation of the Java-based Oracle Forms in a codeless manner with the help of out of box adapters. For an Oracle EBS based application, only Application Host activity is available that must be used to configure the related automation process workflow.

Note:

This activity must be used inside an Application activity or the system displays an error. This lets you establish a connection between Automation Studio and the configured application whose UI elements you want to automate.

Before you start capturing automation steps using this activity, make sure that the intended application is open that must be launched using Automation Studio. Perform Setup Environment to launch the application.

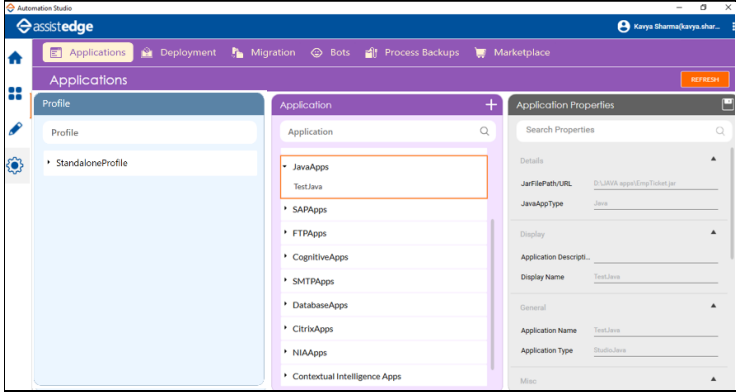

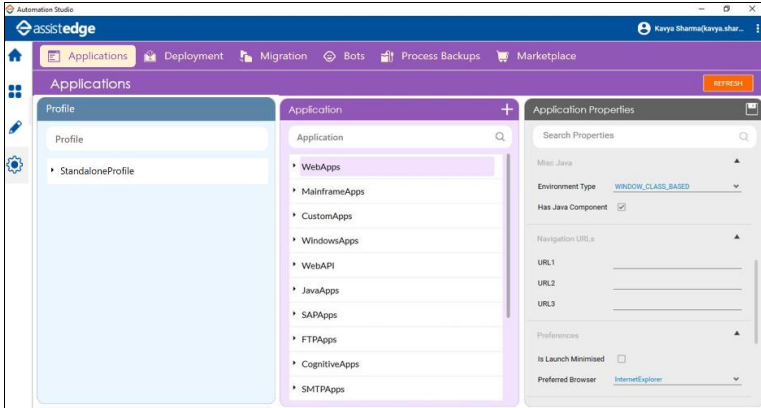
This activity lets you attach a plug in of the selected application type to the configured and launched host application to identify the UI elements.

Prerequisite: Configuring Application

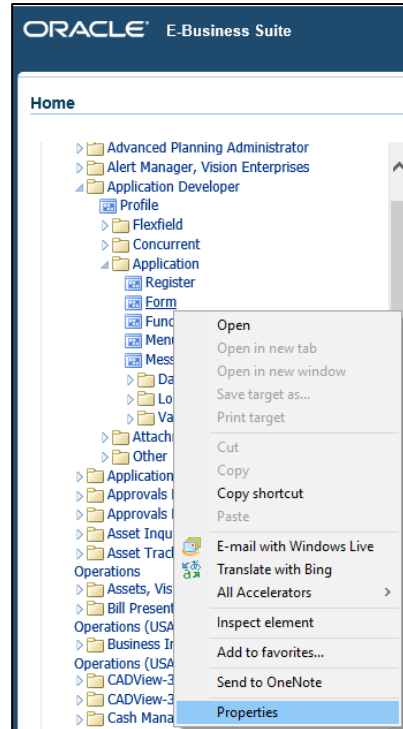
You must configure the intended host application in Automation Studio before you start configuring the steps of automation process workflow. This establishes the connection between the required application and Automation Studio to perform the automation.

Below are the minimum required properties for configuring the intended host application. If you want to define remaining properties, refer [WindowsApps](#) for windows applications, [WebApps](#) for Oracle EBS application or any other web application and [JavaApps](#) for Windows and Java applications respectively.

Requirement	Description
Java Application	<p>Install Java (JDK, v) on the system where the automation process workflow is designed and on the system where the automation would run.</p> <p>Configure Java application:</p> <ol style="list-style-type: none"> 1. In the Admin menu, add an application of Application Type - JavaApps. 2. Browse and select the required .jar file. 3. In the Application Properties panel: <ul style="list-style-type: none"> • Enter the URL of the JNLP file/applet in the JarFilePath/URL field. • Enter the time out duration of the application launch in the Launch Time Out (in secs) field, if the JavaAppType is JNLP Applications or Applet.

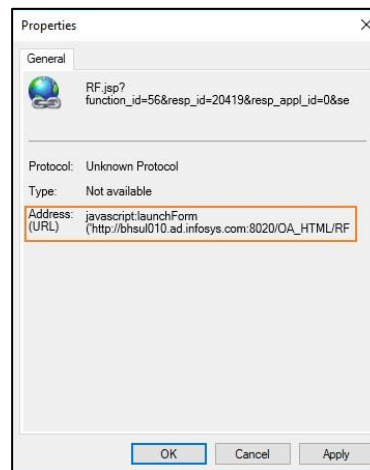
Requirement	Description
	<p>4. In the Environment Type list, select the type of environment of the added Java application.</p>  <p>5. Click the  (Save Properties) icon to save the application.</p> <p>The Java application is configured.</p>
<p>Oracle EBS Application/Other Web Application</p>	<p>Configure Oracle EBS application/ Other Web application:</p> <p>1. In the Admin menu, add an application of Application Type- WebApps. The mandatory fields are highlighted with red box.</p>  <p>2. In the Application Properties panel:</p> <ol style="list-style-type: none"> In the Login URL field, enter the URL of the Oracle Form you want to access. URL of a specific Oracle Form can be retrieved from the properties of that form. To retrieve the Oracle form URL: <ol style="list-style-type: none"> Navigate to the required Oracle Form.

Requirement	Description
-------------	-------------


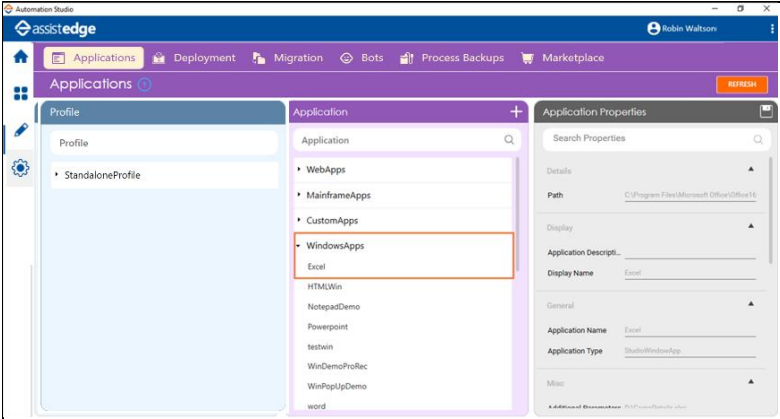


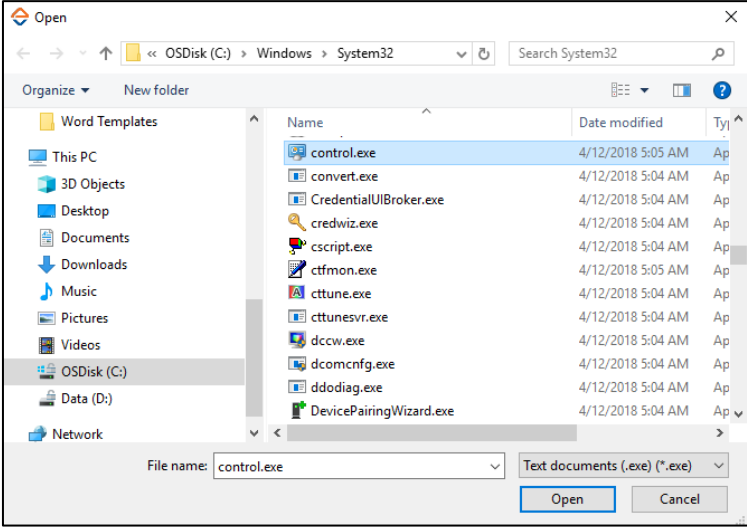
ii. Right click **Form** and then click **Properties**. The **Properties** dialog box appears.

Note the complete Address(URL) of the Form that must be entered as the Login URL in the Application Properties pane of the Admin menu. You may need to scroll down to retrieve the complete Address(URL) field.



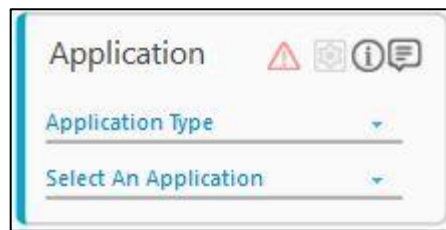
1. In the **Search URL** field, enter the same URL of the Oracle form you want to access.
2. In the **Display Name**, enter a desired name of the web application.

Requirement	Description
	<p>3. In the Misc Java section, select the Has Java Components check box.</p> <p>4. In the Misc Java section, select the type of environment in the Environment Type list.</p> <div data-bbox="670 453 1321 674" style="border: 1px solid gray; padding: 5px; margin: 10px 0;"> <p>Misc Java ▲</p> <p>Environment Type <u>WINDOW_CLASS_BASED</u> ▼</p> <p>Has Java Component <input checked="" type="checkbox"/></p> </div> <p>3. Click the  (Save Properties) icon to save the application.</p> <p>The Oracle EBS application is configured.</p> <div data-bbox="479 850 1534 1207" style="background-color: #007bff; color: white; padding: 10px; border: 1px solid #007bff;"> <p>Note: To start capturing the automation steps, you must login to the required Oracle EBS application.</p> <p>You can automate the process of signing to an Oracle EBS application using the Mutimodal Interface. See the Web Mode section to know how to configure the steps. Alternatively, create a Sign-In Process for the intended Oracle EBS application and use the SignIn Process activity to automate sign in steps. This helps to reuse the sign in steps across other process workflow as well. See Sign In Process and Using SignIn Process sections to know how to automate signing in to an application</p> </div>
<p>Windows application</p>	<p>Configure Windows application:</p> <ol style="list-style-type: none"> In the Admin menu, add an application of Application Type- WindowsApps. The Add Application dialog box opens. <div data-bbox="492 1461 1268 1877" style="border: 1px solid gray; padding: 5px; margin: 10px 0;">  </div> <ol style="list-style-type: none"> In the Application Name field, enter a desired name of the EXE application you want to

Requirement	Description
	<p>add. The name must not contain any special character or space.</p> <p>3. Click ADD and browse for the windows application you want to configure. Click Open.</p>  <p>The windows application is configured.</p>

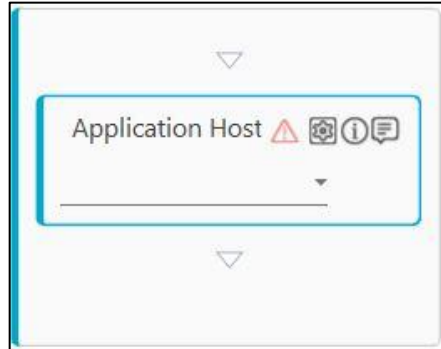
Using Application Host Activity

1. In an automation process workflow, click **Process Components** in the **Canvas Tools** pane to expand the tool and view the associated activities.
2. Drag the **Application** activity and drop on to the **Flowchart designer** on the **Canvas**. The validation error symbol disappears when required inputs are provided.

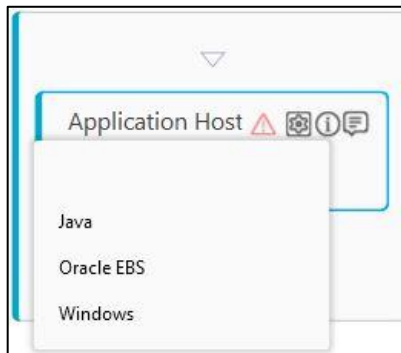



3. In the **Application Type** list, select the type of configured host application you want to automate. For example, select **WebApps** for Siebel or Oracle EBS applications, **WindowsApps** for applications like Notepad or CMD and **JavaApps** for Java applications. You must have at least one application added for it to appear in the list.
4. In the **Select An Application** list, select the configured host application. Alternatively, you can add the required host application at this point of time. To add the application, click any of the links below:
 - a. [Java Applications](#) - Lets you configure the Java applications.
 - b. [Oracle EBS Applications](#) - Lets you configure the Oracle EBS and other web applications like Siebel.

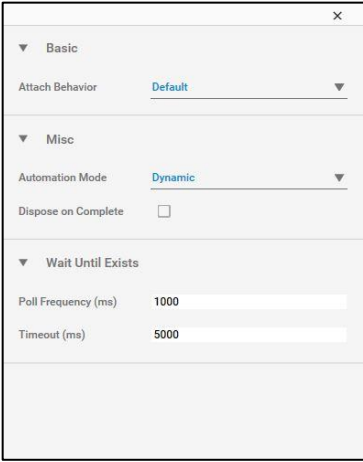
- c. [Windows Applications](#) - Lets you configure the Windows applications.
5. Once the application is configured and the login steps are captured (if required), in the **Menu** bar, click **Test Run** drop down, and then click **Setup Environment** to setup and launch the configured application. It helps to launch the application using automation Studio. Before you start capturing automation steps using this activity, make sure that the intended application is open using the Automation Studio.
6. Double click the **Application** activity, drag the **Application Host** and drop inside the **Application** activity. The validation error symbol disappears when required inputs are provided.



7. From the drop down list, select the type you host application you want to automate. Available options are - **Java**, **Oracle EBS** and **Windows**. A typical web application can be automated in the **Windows** category.

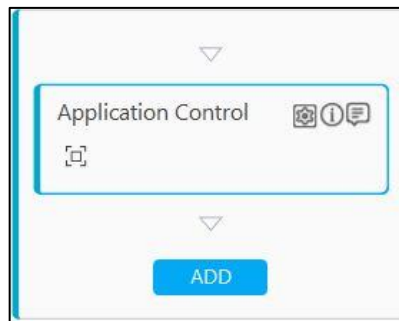


8. Click the  (**Settings**) icon of the **Application Host** activity to attach the plug in based on the type of selected host application. A dialog box appears where you can define configurations related to the attached plug in.



9. In the **Attach Behavior** list of the **Basics** category, select the identification criteria to identify the required host application window to which the plug in must be attached. Available options are:
- **Default** -
 - **Equals** - Selects the window based on the exact match with the window name.
 - **Contains** - Selects the window based on the text present in the window name.
 - **StartsWith** - Selects the window based on the starting text of the window name.
 - **EndsWith** - Selects the window based on the text with which the window name ends.
10. In the **Window Title** field, enter the window name of the host application to which the plugin must be attached. This field is available only if the **Attach Behavior** selected is **Equals**, **Contains**, **StartsWith** or **EndsWith**. For example, if the name of the Oracle Form is **Oracle Applications – AUTOSC**, select the Attach Behavior as **StartsWith** and the enter the Window Title as **Oracle**.
11. In the **Application Mode** list, select Dynamic or Static depending upon the type of UI element to be extracted while performing the automation. This field is available only if the host application type is **Oracle EBS**.
12. Select the **Dispose on Complete** check box in the **Misc** category if you want to close the application window once the automation is completed.
13. In the **Wait Until Exists** category you can configure the time for the plug in to wait until the UI element is found. You can use this to avoid any delay in the page loading time:
- In the **Poll Frequency (ms)** field, enter the frequency of checking the availability of the UI element. The time must be entered in milliseconds.
 - In the **Timeout (ms)** field, enter the wait time for the plug in until the UI element is found or identified that it does not exist. The time must be entered in milliseconds.
14. Close the dialog box to save the plug in configurations.
15. Double click to open the **Application Host** activity block.

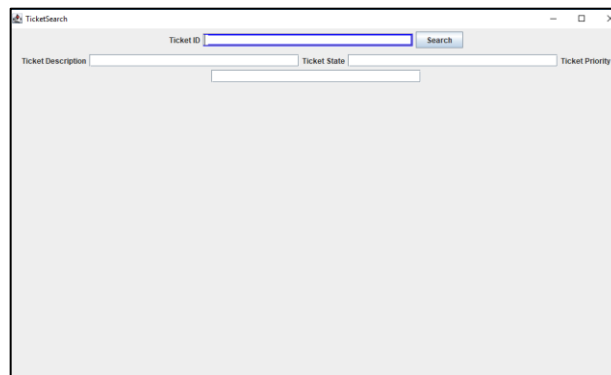
16. Click **ADD** to add the **Application Control** activity. Alternatively, you can drag the Application Control activity from the **Canvas Tools** pane, and drop inside the **Application Host** activity.



17. Click the  (**Field Configuration**) icon. Manually bring the application window in focus. Click any of the links below for step by step guide to automate the UI elements of each of the host application type:
 - [Java](#) - Lets you automate UI elements of a Java application
 - [Oracle EBS](#) - Lets you automate UI elements of a Java based Oracle EBS application
 - [Windows](#) - Lets you automate Windows and Web applications


Java

1. Hover over the field that you want to capture. The field gets highlighted with the blue box.



- Click the highlighted area. A dialog box appears where you can configure the details related to the captured UI element. The search path of the component and the window name are auto filled. The fields that are mandatory are highlighted with the red box.

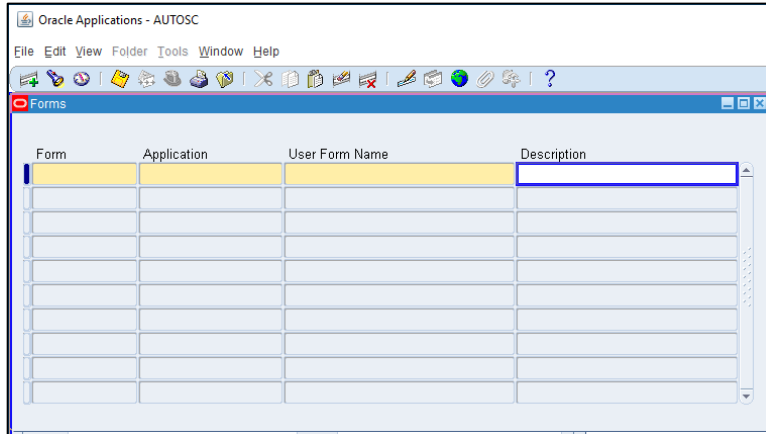
- In the **Action** list, select the action that you want to perform on the captured area. Other fields change depending on the selection from the list. Enter details of all the mandatory fields and other relevant fields as per your requirement. Refer [Windows_Application_Field_Descriptions](#) table to know more about the available fields and their respective properties.
- Close the dialog box. The **Save Confirmation** message box appears.
- Click **YES** to save the entered details. Click **NO** to exit the message box without saving the entered details and to re-configure the required details.

- You need to add the **Application Control** activity for every UI element you want to capture. Repeat steps 16 and 17 to capture all the UI elements involved in the automation process workflow. You can edit the configuration by clicking the  (**Settings**) icon of the Application Control activity.

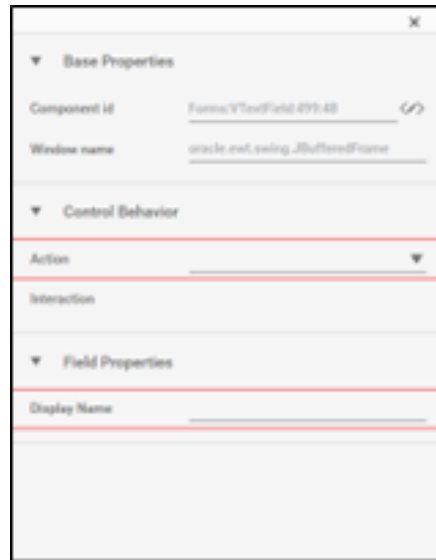
The Java application host is configured.

Oracle EBS

1. Hover over the application window to start capturing the automation steps that you want to perform in the Oracle EBS application. The area that you can capture gets highlighted with a blue box.

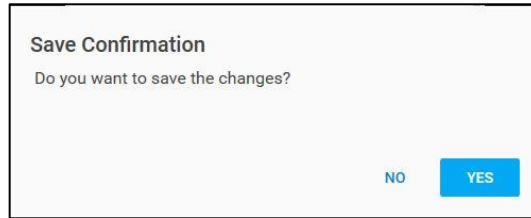



2. Click the intended area from where you want to extract the Id of the UI element. A dialog box appears where you can configure the details related to the captured UI element. The Id of the component and the window name are auto filled. The fields that are mandatory are highlighted with red box.



3. In the **Action** list, select the action that you want to perform on the captured area. Other fields change depending on the selection from the list. Enter details of all the mandatory fields and other relevant fields as per your requirement. Refer [Oracle EBS Application Field Descriptions](#) table to know more about the available fields and their respective properties.
4. Close the dialog box. The **Save Confirmation** message box appears.

- Click **YES** to save the entered details. Click **NO** to exit the message box without saving the entered details.



- You need to add the **Application Control** activity for every UI element you want to capture. Repeat steps 16 and 18 to capture all the UI elements involved in the automation process workflow. You can edit the configuration by clicking the  (**Settings**) icon of the **Application Control** activity.
- The Oracle EBS application is automated.

Note: You must Close Environment and then Reset Environment every time you want to [Test Run](#) a process workflow, else, error is received or the test run fails.

Oracle EBS Application Field Descriptions

Property Name	Usage
Base Properties	
Component id	<p>The Id of the selected UI element that gets auto filled on selection of the UI element. You can replace the static part of the component Id with an argument to make a dynamic part of the component Id.</p> <p>To create dynamic component Id:</p> <p>Click the (Configure) icon. The Dynamic ComponentId Creator dialog box opens.</p> <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> </div>
Window name	The name of the window of the selected UI element.
Control Behavior	
Action	It is the action or the interaction that can be performed on the Oracle EBS application

Property Name	Usage
	<p>depending on the type of UI element as per the requirement. Available options are:</p> <ul style="list-style-type: none"> ▪ Activate Window - Allows to activate a form which is currently not in focus and perform required action. ▪ Click - Allows to left click the selected UI element. ▪ Click Toolbar Item By Index - Clicks the corresponding toolbar item selected as per its index in the Oracle EBS form. ▪ Click Toolbar Item By Value - Clicks the corresponding toolbar item selected as per its value in the Oracle EBS form. ▪ Close Window - Closes the selected Oracle EBS Form. ▪ Decrement scroll bar - Decrements the scroll bar as per the entered index. ▪ Get Text- Retrieves text from the selected UI element. ▪ Get Statusbar Text - Retrieves the text of the selected status bar. ▪ Get Radiobutton Text - Retrieves the text of the selected radio button. ▪ Get Choicebox Text - Retrieves the text or index of the selected choice box. ▪ Get selected Choicebox index - Retrieves the index of the selected choice box. ▪ Get checkbox status- Retrieves the status of the selected check-box. ▪ Get checkbox text - Retrieves the text of the selected check-box. ▪ Get CellData - Extracts data of the selected cell. ▪ Get Visible Row count - Retrieves the count of visible rows in the grid. ▪ Get visible column count - Retrieves the count of visible column s in the grid. ▪ Get Row count - Retrieves the total number of rows available in the grid. ▪ Get Column count - Retrieves the total number of columns available in the grid. ▪ Get Scroll bar value - Retrieves the current position of the scroll bar of the table. ▪ Get selected Tab Text - Retrieves the text of the selected tab. ▪ Increment scroll bar - Increments the scroll bar as per the entered index.

Property Name	Usage
	<ul style="list-style-type: none"> ▪ Is Focused - Retrieves the current state of the form. If the selected Form is in focus, True is returned else False is returned. ▪ Is Exists - Retrieves the current state of the form. If the selected Form exists, True is returned else False is returned. ▪ Is visible - Retrieves the current state of the form. If the selected Form is visible, True is returned else False is returned. ▪ Is enabled - Retrieves the current state of the form. If the selected Form is enabled, True is returned else False is returned. ▪ Minimize Window - Minimizes the selected Oracle EBS Form. ▪ Maximize Window - Maximizes the selected Oracle EBS Form. ▪ Set Text - Sets user provided text input to the selected UI element of the Oracle EBS Form. ▪ Select Navigator Item by Index - Selects the specified Function available in the Navigator window of an Oracle EBS Form as per the specified index. ▪ Select Tab By Value - Selects the required tab as per the entered tab name. It helps in switching of tabs. ▪ Select grid row - Selects and clicks the corresponding row of the grid. ▪ Select Listview Item By Index - Selects an item from a list-view (drop-down list) as per the specified index. ▪ Select Choicebox By Value - Selects a value from the corresponding choice box. ▪ Select RadioButton By Text - Sets the corresponding radio button in the radio button group. ▪ Set Checkbox Status - Selects the required check-box. ▪ Set Focus - Brings the Oracle EBS Form in focus before performing any action on it. ▪ Set Scroll bar value - Sets the scroll bar at the desired position.
Interaction	The corresponding interaction set against the selected action.
Appears for Action that require input index for execution.	
Index	The index of the UI element where you want to perform the selected Action on the Oracle EBS Form such as 3, 5 and so on.

Property Name	Usage
Appears for Action that require input value for execution.	
Value	The value of the UI element where you want to perform the selected Action on the Oracle EBS Form such as the tab name, tool bar item name and so on.
Appears only if Action selected is Select grid row.	
Row Number	The row number that must be selected.
Appears only if Action selected is Select Listview item by index.	
Item Index	The index of the item that must be selected from the list-view (drop down list).
Appears only if Action selected is Set Text, Select Choicebox By Value, and Select RadioButton By Text	
Text	The user input to be specified. You must define a parameter in the Parameter bar to use this option. Select the Is Default Value check-box, and enter the input value, if you want to set a default value.
Appears only if Action selected is Set CheckBox Status.	
IsSelected	Signifies the status of the check box. You must define a parameter in the Parameter bar to use this option. Select the Is Default Value check-box, and enter the status manually, if you want to set a default value.
Field Properties	
Display Name	User specified display name of the windows element selected.
Appears for Action that return a value such as actions related to Get and Is interactions.	
Variable Name	User defined name of the variable that stores the captured or returned value. You can define the variable name here itself or predefine it in the Parameter bar.

Property Name	Usage
Variable Type	Type of the variable defined. Available options are: <ul style="list-style-type: none"> ▪ Text ▪ Numeric ▪ Decimal ▪ Boolean ▪ Data Table

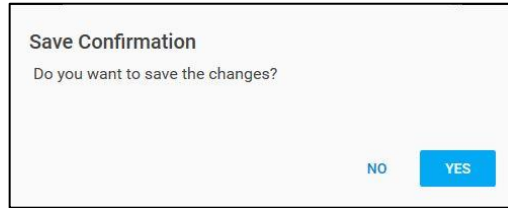
Windows


1. Press the **Ctrl** key and hover over the windows/web application to start capturing the automation steps that you want to perform in the Windows application. The area that you can capture gets highlighted with a red box.
2. While pressing the **Ctrl** button click the intended area. A dialog box appears where you can configure the details related to the captured UI element. The fields that are mandatory are highlighted with red box.

The screenshot shows a dialog box with the following fields and values:

- Control Behavior**
 - Action: [Dropdown menu]
 - Interaction: [Text field]
- Field Properties**
 - Display Name: [Text field]
 - AutomationId: [Text field]
 - Class Name: [Text field]
 - Container Class Name: IEFram... [Text field]
 - Container Control Ty...: 50032 [Text field]
 - Container Name: Bing - Internet Explorer [Text field]
 - Control Index: 1 [Text field]
 - Control Name: Languages: [Text field]

3. In the Action list, select the action that you want to perform on the captured area. Other fields change depending on the selection from the list. Enter details of all the mandatory fields and other relevant fields as per your requirement. Refer [Windows Application Field Descriptions](#) table to know more about the available fields and their respective properties.
4. Close the dialog box. The Save Confirmation message box appears. Click YES to save the entered details. Click NO to exit the message box without saving the entered details.



- You need to add the **Application Control** activity for every UI element you want to capture. Repeat steps 16 and 17 to capture all the UI elements involved in the automation process workflow. You can edit the configuration by clicking the  (**Settings**) icon of the Application Control activity.

The Windows application host is configured.

Windows Application Field Descriptions

Property Name	Usage
Control Behavior	
Action	<p>The applicable action or interaction that can be performed on the selected UI element. The available options change as per the type of host application being automated:</p> <ul style="list-style-type: none"> Click- Allows to left click the selected UI element. Expand/Collapse- Allows to expand or collapse the selected UI element. Legacy Interaction- Allows to configure UI element of the Windows application which supports a legacy pattern. Get Text- Retrieves text from the selected UI element. Set Text- Sets user provided text input to the selected UI element and stores it into a parameter. You must define the parameter in the Parameter bar before you start capturing the UI element. Mouse Click- Allows mouse simulation. It requires the application to be in focus. You can choose to perform Left Click, Right Click or Double Click available from the drop down list. Keyboard Input-Captures UI element input from the keyboard in the selected UI element. This interaction allows performing different keyboard input combinations with the help of Single Key, Double Keys, Triple Keys and TextEntry options. Set Focus- brings the UI element in focus before performing any action on it. If Control Exists- Checks availability of the UI element on the windows application. It is used while working with dynamic controls. Get Radio Button Status- Retrieves status of the selected radio button status. It returns true or false. Toggle Checkbox Action- Toggles the current state of a checkbox. Get Checkbox Status- Returns the current selection status of a checkbox. Perform ComboBox Select Value- Selects the specified value from the drop-down list.

Property Name	Usage
	<p>Get ComboBox Value- Retrieves value of the selected combo box.</p> <p>Perform ComboBox Set Text- Sets a user-specified value on the selected ComboBox. All ComboBox controls do not support this interaction.</p> <p>Scroll Horizontal- Performs horizontal scroll based on the input provided on the percentage scale from 0 to 100.</p> <p>Scroll Vertical- Performs vertical scroll based on the input provided on the percentage scale from 0 to 100.</p> <p>Select Tree Node- Selects a node of a tree control using this interaction. Specify the input in a specific format. For example. for selecting node 1.1, enter the input value as Number/1/1.1.</p> <p>Expand Tree Node- Expands a node of the tree UI element.</p> <p>Select- Selects a particular UI element.</p> <p>Get DataGrid Value by Index- Retrieves data grid value by its index. Specify the cell index from which the input is needed.</p> <p>Focus Data Grid Cell by Index- Brings a data grid cell in focus based on the input cell index.</p> <p>Set DataGrid Value by Index- Sets the value to a DataGrid cell based on its index.</p> <p>Get Data from Table- Retrieves complete data of a table control to a DataTable. If required, the data from the DataTable can then be saved in an excel/CSV file. Following are the steps involved to retrieve the data.</p> <p>Create an argument of type System.Data.DataTable, before extracting the UI element.</p> <p>Click WINDOWS MODE to extract the Data Grid Table control.</p> <p>Select Get Data from the Table.</p> <p>Set the Variable Name as the argument created in the first step.</p> <p>Save the configuration.</p> <p>Use Export Data Table activity to write data to excel.</p> <p>Set Focus- This interaction is used to bring any control to focus before performing an action on it.</p>
Interaction	<p>The corresponding interaction class name auto populated against the selected action.</p> <p>The interaction or the action type corresponding to the selected interaction.</p>
Appears if Action selected is Legacy Interaction	
Action Type	The interaction or the action type corresponding to the legacy interaction.
Key	Appears if Action selected is Keyboard Input. The single key set as input from the keyboard.
Key	Appears if Action selected is Keyboard Input and Interaction selected is DoubleKeys. The second key which is set as input from the keyboard along with the first key.

Property Name	Usage
Key	Appears if Action selected is Keyboard Input and Interaction selected is TripleKeys. The third key which is set as input from the keyboard along with the first and the second key
Text	Appears if Action selected is Keyboard Input and Interaction selected is TextEntry. Allows user input as a text.
Mouse Action	Appears if Action selected is Mouse Click. Allows user input using the mouse. Available options are: LeftClick- Performs a left click at the specified offset point. DoubleClick- Performs a double click at the specified offset point. RightClick- Performs a right click at the specified offset point.
Field Properties	
Display Name	User specified display name of the windows element selected.
AutomationId	Unique identifier for the automation element in the automation tree. See Handling Dynamic Controls section to know more about the usage of this field.
Class Name	Class name of the UI element as defined by the UI element developer. See Handling Dynamic Controls section to know more about the usage of this field.
Container Class Name	Container class name of the UI element as defined by the developer. See Handling Dynamic Controls section to know more about the usage of this field.
Container Control Type	Control type of the container of the UI element as defined by the UI element developer.
Container Name	Container name of the UI element as defined by the developer. See Handling Dynamic Controls section to know more about the usage of this field.
Control Index	Index of the current UI element within the entire set of elements with the same values of automation Id, class name and UI element name as that of the current element.
Control Name	Control name of the UI element as defined by the UI element developer. See Handling Dynamic Controls section to know more about the usage of this field.
Control Type	Control type of the UI element as defined by the developer.
Appears if Action selected is Mouse Click.	
Offset From Control	It is the distance between the UI element and the anchor.
Search By Control Order No.	Signifies if the UI element needs to be searched based on the UI element order number. Control order number is the index of the current element within the entire set of elements with the same value of UI element Id as that of the current element.
Appears if Action selected is Get Text.	
Variable Name	User defined name of the variable that stores the captured value.

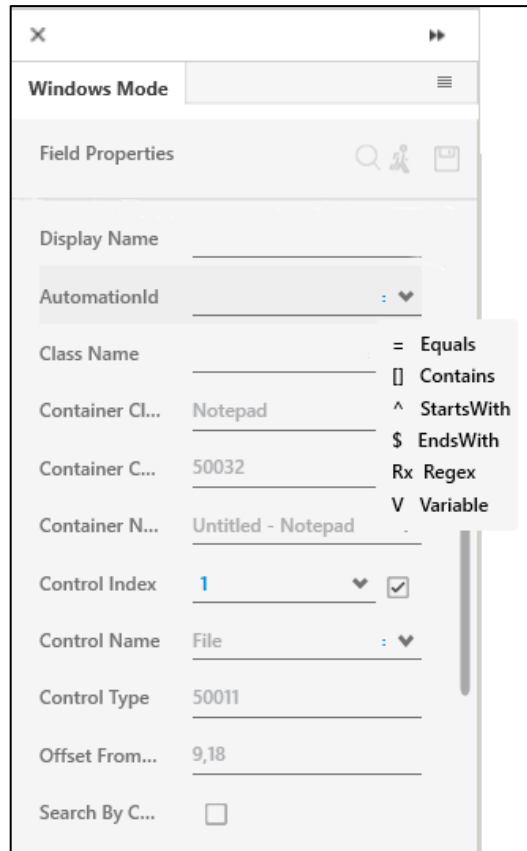
Property Name	Usage
Variable Type	Type of the variable defined. Available options are: <ul style="list-style-type: none"> Text Numeric Decimal Boolean Data Table
Misc Properties	
Max wait for input idle	The maximum time interval in milliseconds for which the windows plugin waits on launch before it starts executing the configured interactions.
Retry Count	Number of times, the plugin retries to find a UI element, if not found.
Retry Interval (ms)	The time interval in milliseconds for which the plugin waits before it attempts for a retry.


Handling Dynamic Controls

Automation Studio provides different identification criteria and Parametrization of the value of the dynamic field properties of the UI elements. You can reconfigure some of the auto captured field properties of such UI elements to create a more effective way of identifying the indented UI element. For such controls, remove the dynamic part of the string and use an appropriate option for UI element identification. For example, if a Class Name field is recognized as AssistEdge_Studio_20180205083009, the later part of the string, which is a timestamp, is dynamic and can be removed. Retain only the static part and reconfigure the field properties using the suitable option.

Following are the different field properties that can be reconfigured against their respective values:

- AutomationId
- Class Name
- Container Class Name
- Container Name
- Control Name



Use the  (drop down arrow) icon to reconfigure the identification criteria. Available options are:

- **Equals**- Identifies the UI element based on the exact match with the value of the selected field property.
- **Contains**- Identifies the UI element based on the string present in the value of the selected field property.
- **StartsWith**- Identifies the UI element based on the starting string of the value of the selected field property.
- **EndsWith**- Identifies the UI element based on the ending string of the value of the selected field property.
- **Regex**- Identifies the UI element with the string matching the value of the selected field property as per the identification pattern defined through the provided regular expression such as a.b, *txt and others.
- **Variable**- Identifies the UI element as per the parametrization of the value of the selected field property value. Parametrization allows to run the identification process over and over again using different values. With

parameterization, windows controls with looping and assignment activities, are used.

Application Host Properties

The properties of the Application Host activity are listed in the following table and can be edited in the Property grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity.</p> <p>If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
AppType	The plugin that gets attached to the host application depending on the type of host application.
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as the name of the selected application. You can change the name as required.

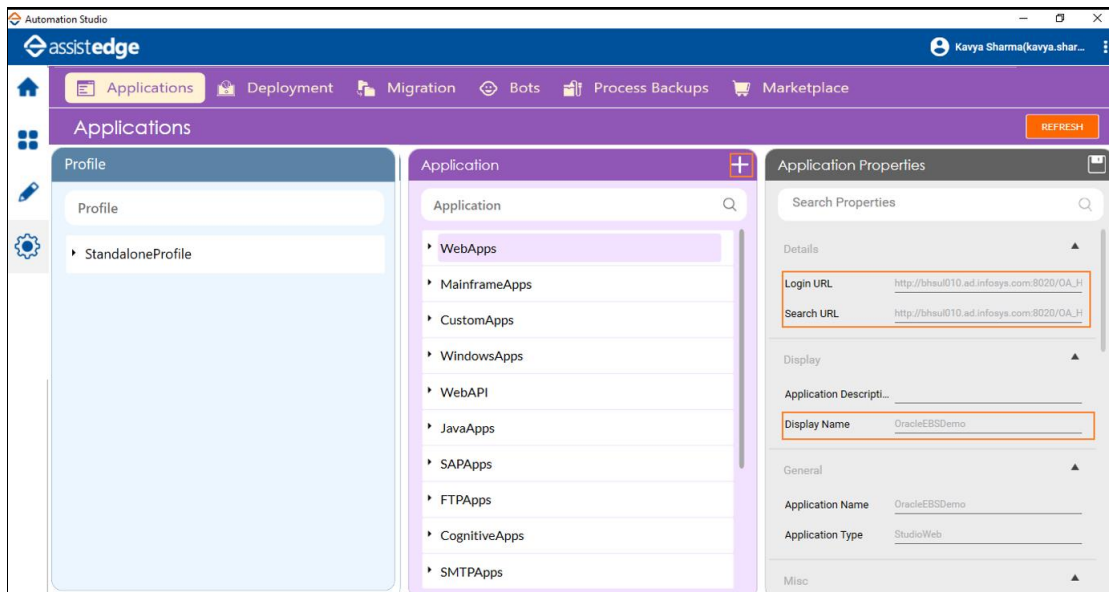
Step-by-Step Guide to Use Oracle EBS Application to Retrieve Employee Information

Let's create an example to open the Human Resource, Vision Enterprises Form of Oracle EBS application and search an employee with the last name damin in the Find People window. Select an employee named Damini Gaur from the view-list displayed and then retrieve the employee number from the People window.

Next, we open the Navigator window and navigate to the Human > View > Work Assignment function. Open the Work Assignment window, enter the retrieved employee number, and then automatically fill the full name of the employee. As the output of the automated steps, view all the actions happening on the Oracle EBS Form automatically. At the completion of the workflow the Assignment History of Damini Gaur gets displayed and the Organization name is retrieved.

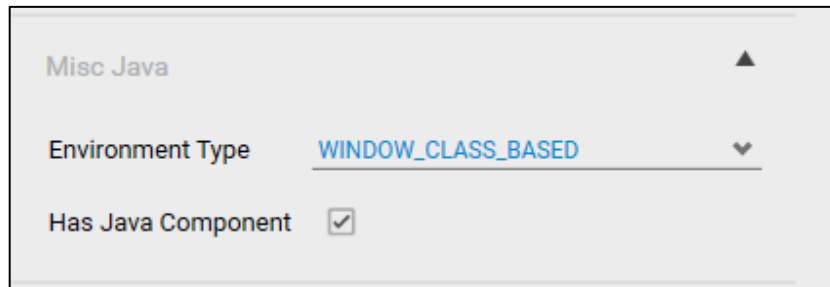
Note: You must perform Close Environment and then Reset Environment every time you want to Test Run a process workflow, else, error is received or the test run fails.


1. In the **Admin** menu, add a **WebApps** application.
2. In the Application Properties panel:
 - Enter the **Login URL** of the Oracle EBS **Form, Human Resource, Vision Enterprises**. To retrieve the Form URL:
 - Navigate to Human Resource, Vision Enterprises Form.
 - Right click, and then click **Properties**
 - Enter the **Search URL** of the Oracle EBS Form. It must be same as the Login URL.
 - Enter a **Display Name, OracleEBSDemo**. You can enter any name of your choice.

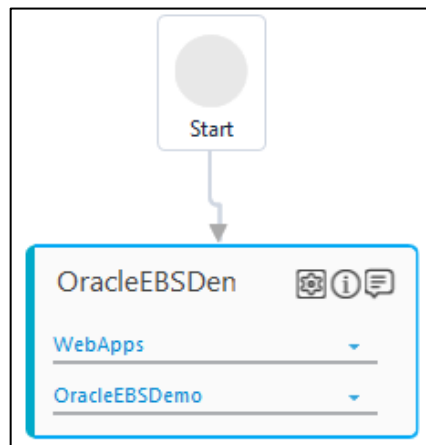



- In the **Misc** Java section, select the **Has Java Components** check box.


- In the **Misc Java** section, select the type of environment in the **Environment Type** list.

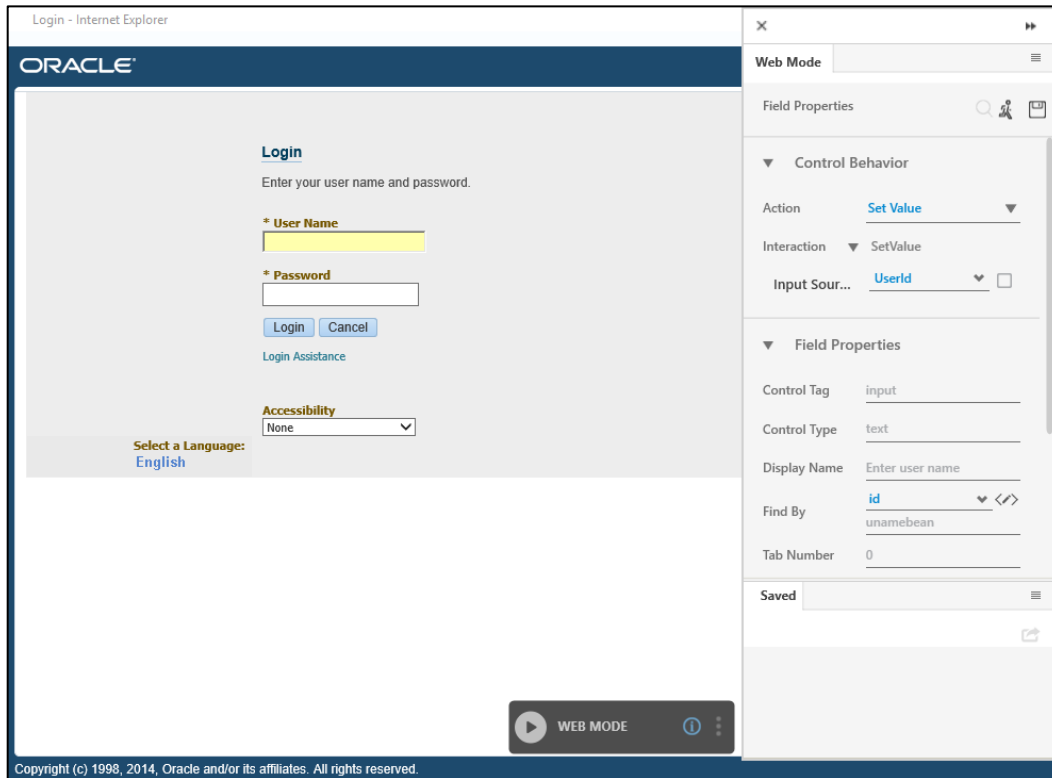








3. Click the  (**Save Properties**) icon to save the application. The Oracle EBS application is configured.
4. Create a new **Process**.
5. In the **Canvas Tools** pane click **Process Components** to view the associated activities.
6. Drag the **Application** activity and drop on to the **Flowchart designer** on the **Canvas**. The validation error symbol disappears when required inputs are provided.
7. In the **Application Type** list, select **WebApps**.
8. In the **Select An Application** list, select the configured host application, **OracleEBSDemo**.



9. Click the  (**Settings**) icon to launch the **WEB MODE** to capture the steps of logging to the Oracle EBS application. The configured Oracle EBS application appears.

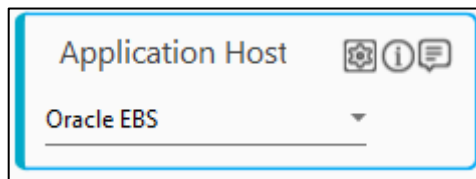
- Click the  (**Play**) icon and then click the **User ID** field. The **Field Properties** panel of the **Web Mode** of the web application appears. The fields that are mandatory are highlighted with red box.




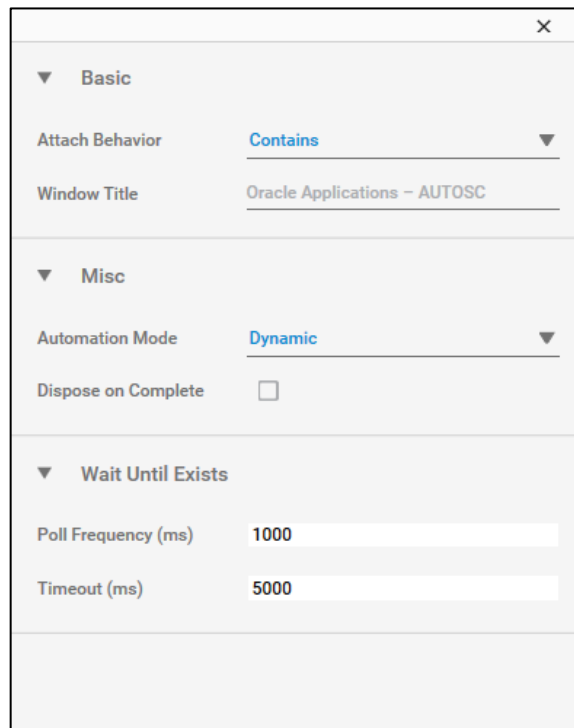
- In the **Action list**, select **Set Value**.
- In the **Input Source**, select **Userld**.
- Enter a **Display Name** and then click the  (**Save**) icon to configure the details captured. The saved details start appearing in the **Saved** tab of **Field Properties** bar.
- Click the  (**Play**) icon and then click the **Password** field. The **Field Properties** panel of the **Web Mode** of the web application appears. The fields that are mandatory are highlighted with red box.
- In the **Action list**, select **Set Attribute**.
- In the Input Source, select Password.
- Enter a **Display Name** and then click the  (**Save**) icon to configure the details captured. The saved details start appearing in the **Saved** tab of **Field Properties** bar.
- Click the  (**Play**) icon and then click the **Login** button. The Field Properties panel of the **Web Mode** of the web application appears. The fields that are mandatory are highlighted with red box.
- In the **Action list**, select **Click**.
- Enter a **Display Name** and then click the  (**Save**) icon to configure the details captured. The saved details start appearing in the **Saved** tab of **Field Properties** bar.
- Click the  (**Save configured field and return to studio**) icon to save the field that you configured and return to the Studio

menu. The login steps are captured.

22. In the **Menu** bar of the **Studio** menu, click **Test Run** drop down, and then click **Setup Environment** to setup and launch the configured application using the Automation Studio.
23. Double click the **Application** activity and add the **Application Host** activity inside the **Application** activity.
24. In the **Application Host** activity, select **Oracle EBS** from the drop down list.



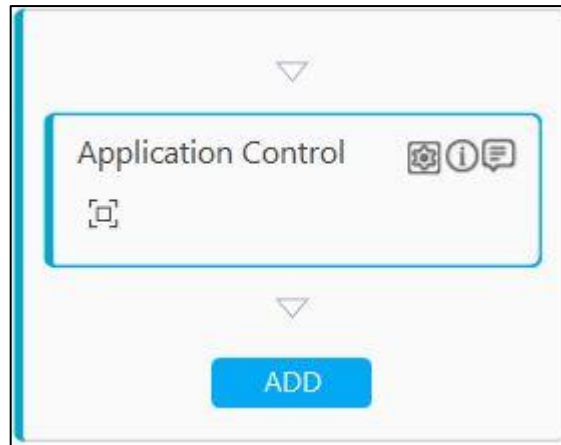
25. Click the  (**Settings**) icon of the Application Host activity to attach the plug in based on the type of selected host application. A dialog box appears where you can define configurations related to the attached plug in.




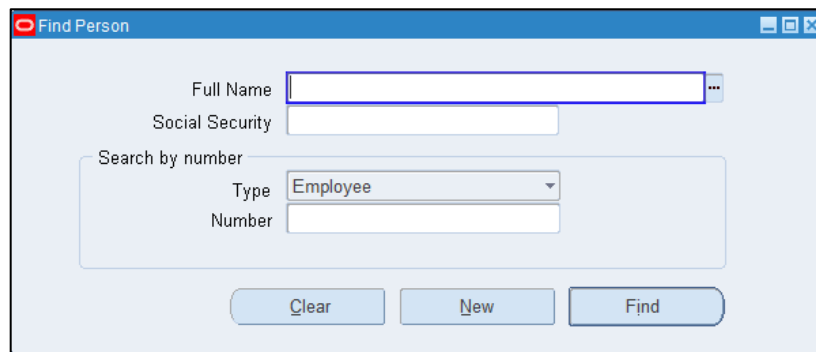
26. In the **Attach Behavior** list of the **Basics** category, select **Contains**.
27. In the **Window Title** field, enter **Oracle Applications – AUTOSC**. Other default settings remain same. Close the dialog box to save the plug in configurations.
28. Double click to open the **Application Host** activity block.
29. We have demonstrated **Set Text, Get Text, Click, Select Listview item by index and Activate Window** operations, that can be performed on the configured Oracle EBS form. You can configure other operations in the similar way depending on your business requirement.

Set Text

1. Click ADD to add the Application Control activity.



2. Click the  (Field Configuration) icon of the of the Application Control activity, and then bring the Oracle EBS application window in focus.
3. Hover over the Full Name field . The area that you can capture gets highlighted with a blue box.



- Click the highlighted area to extract the Id of the UI element. A dialog box appears where you can configure the details related to the captured UI element. The Id of the component and the window name are auto filled. The fields that are mandatory are highlighted with red box.


Base Properties	
Component id	Find Person:VTextField:200:24
Window name	oracle.ewt.swing.JBufferedFrame

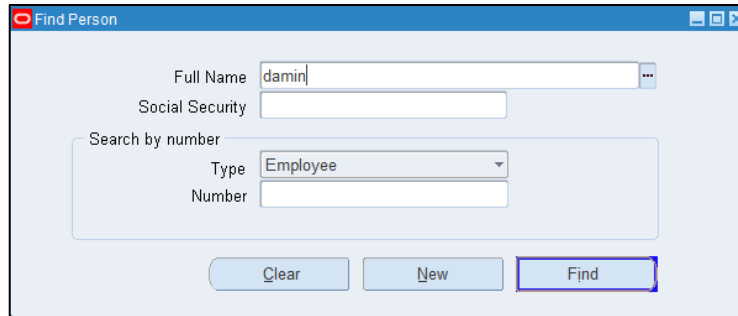
Control Behavior	
Action	Set Text
Interaction	Set Text
Text	damin <input checked="" type="checkbox"/>

Field Properties	
Display Name	Enter name

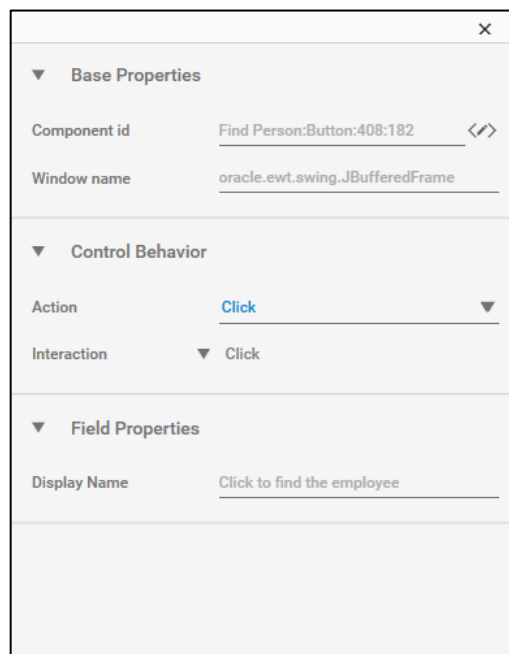
- In the Action list, select Set Text.
- In the Text field, enter damin and select the Is Default Value. You can enter any employee name that you want to search.
- In the Display Name, enter Name. You can enter any name of your choice to be displayed.
- Close the dialog box. The Save Confirmation message box appears.
- Click YES to save the entered details.
- In the Properties pane of the added Application Control activity, set the Wait After time as 3000 as the next screen takes time to load.
- Follow below steps to configure the subsequent steps of the automation process workflow example.

Click

1. Click **ADD** to add the **Application Control** activity.
2. Click the  (**Field Configuration**) icon of the of the **Application Control** activity, and then bring the Oracle EBS application window in focus.




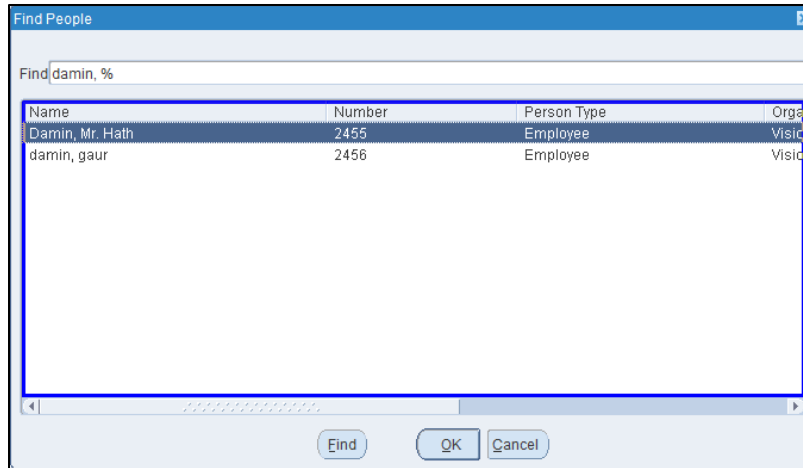
3. Hover over the **Find** field, and then click the button. A dialog box appears.



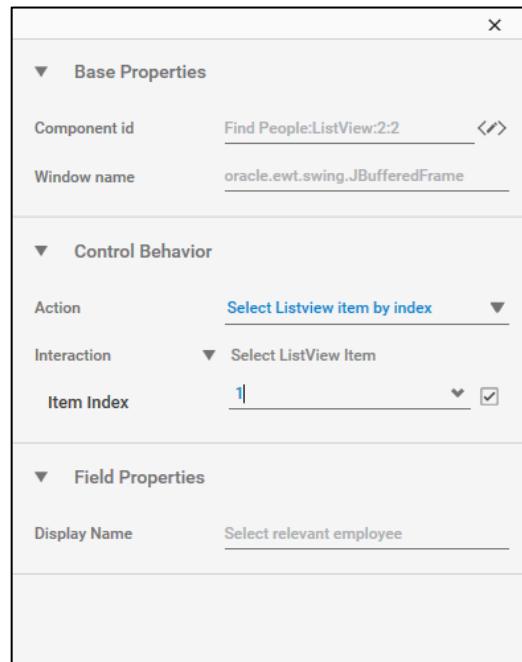
4. In the **Action** list, select **Click**.
5. In the **Display Name**, enter **Click to find the employee**. You can enter any name of your choice to be displayed.
6. Close the dialog box. The **Save Confirmation** message box appears.
7. Click **YES** to save the entered details.
8. In the **Properties** pane of the added **Application Control** activity, set the **Wait After** time as **3000** as the next screen takes time to load.
9. Follow below steps to configure the subsequent steps of the automation process workflow example.

Select Listview item by index


1. Click **ADD** to add the Application Control activity.
2. Click the  (**Field Configuration**) icon of the of the **Application Control** activity, and then bring the Oracle EBS application window in focus.

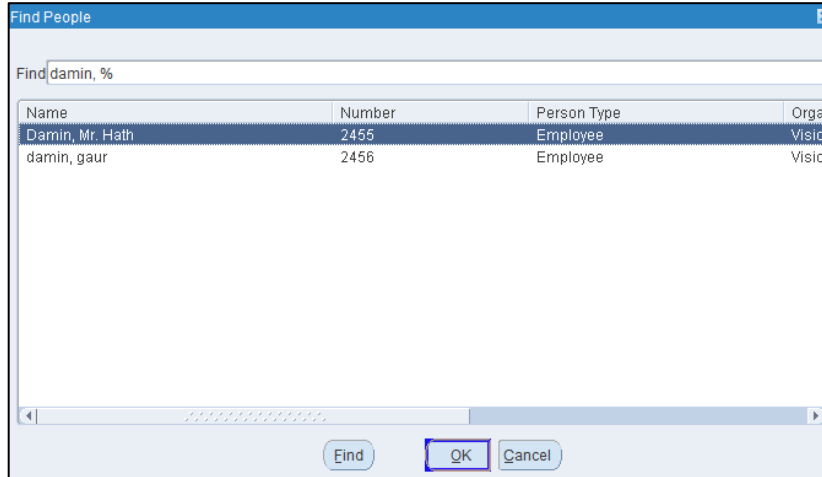


3. Hover over the list view, and the click the highlighted area. A dialog box appears.

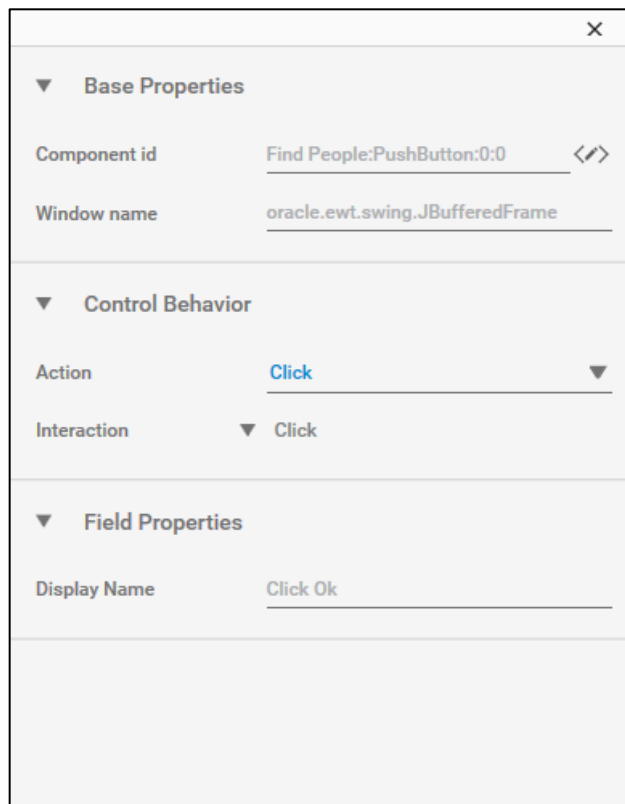


4. In the Action list, select Select Listview item by index.
5. In the **Index** field, enter **1** to select the second data available in the list.
6. In the **Display Name**, enter **Select relevant employee**. You can enter any name of your choice to be displayed.
7. Close the dialog box. The **Save Confirmation** message box appears.
8. Click **YES** to save the entered details.

9. Click **ADD** to add the Application Control activity.
10. Click the  (**Field Configuration**) icon of the of the **Application Control** activity, and then bring the Oracle EBS application window in focus.




11. Hover over the **OK** field, and the click the button. A dialog box appears.

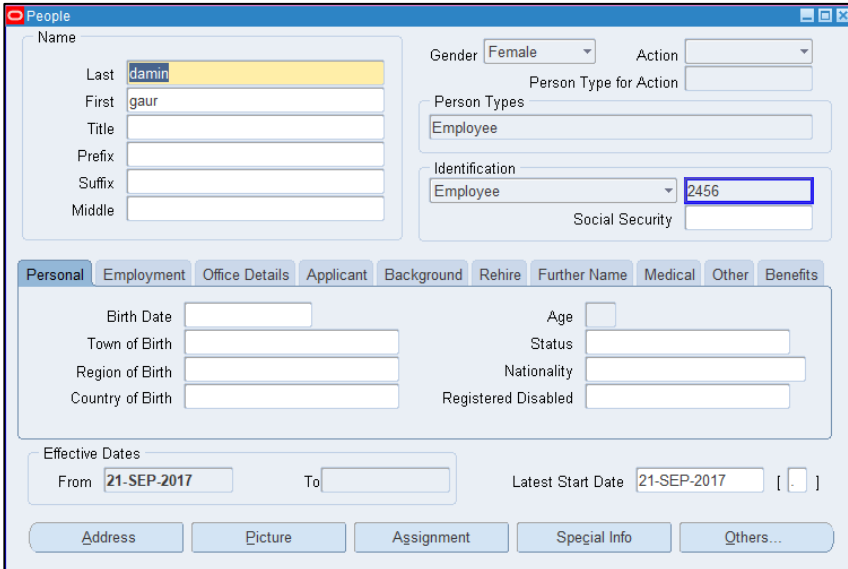


12. In the **Action** list, select **Click**.
13. In the **Display Name**, enter **Click Ok**. You can enter any name of your choice to be displayed.

14. Close the dialog box. The Save **Confirmation** message box appears.
15. Click **YES** to save the entered details.
16. In the **Properties** pane of the added **Application Control** activity, set the **Wait After** time as **3000** as the next screen takes time to load.
17. Follow below steps to configure the subsequent steps of the automation process workflow example.

Get Text

1. Click **ADD** to add the **Application Control** activity.
2. Click the  (**Field Configuration**) icon of the of the **Application Control** activity, and then bring the Oracle EBS application window in focus.



The screenshot shows the 'People' form in Oracle EBS. The form is divided into several sections:

- Name:** Last name is 'damin', First name is 'gaur'. Other fields for Title, Prefix, Suffix, and Middle are empty.
- Gender:** Female
- Action:** (Dropdown menu)
- Person Type for Action:** (Dropdown menu)
- Person Types:** Employee
- Identification:** Employee (dropdown), 2456 (text field)
- Social Security:** (Text field)

Below these fields are tabs for: Personal, Employment, Office Details, Applicant, Background, Rehire, Further Name, Medical, Other, Benefits.

The **Personal** tab is active, showing:

- Birth Date: (Text field)
- Age: (Text field)
- Town of Birth: (Text field)
- Status: (Text field)
- Region of Birth: (Text field)
- Nationality: (Text field)
- Country of Birth: (Text field)
- Registered Disabled: (Text field)

Effective Dates:

- From: 21-SEP-2017
- To: (Text field)
- Latest Start Date: 21-SEP-2017


At the bottom, there are buttons for: Address, Picture, Assignment, Special Info, Others...

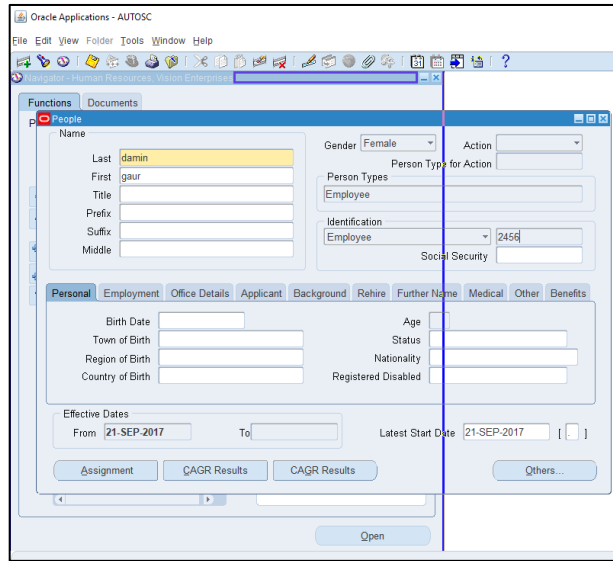
3. Hover over the **Identification field**, and then click the field containing the employee Id. A dialog box appears.

Base Properties	
Component id	People:VTextField:600:133
Window name	oracle.ewt.swing.JBufferedFrame
Control Behavior	
Action	Get Text
Interaction	Get Text
Field Properties	
Display Name	Employeeid
Is C-View	<input type="checkbox"/>
Variable Name	Employeeid
Variable Type	Text

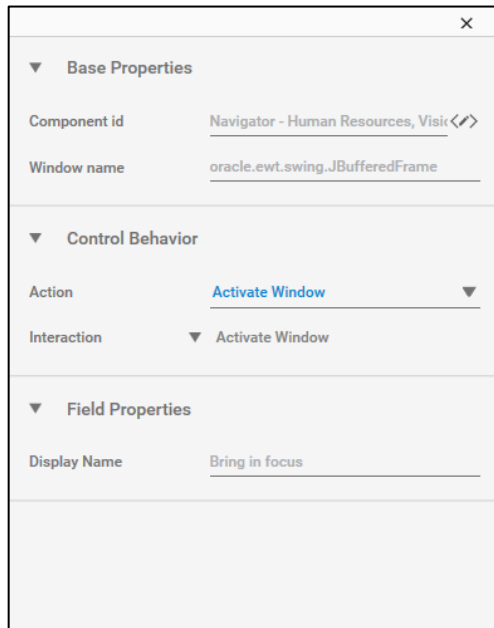
4. In the **Action** list, select **Get Text**.
5. In the **Display Name** field, enter **Employeeid**. You can enter any name of your choice to be displayed.
6. In the **Variable Name** field, enter **Employeeid**. The variable gets created and the extracted data gets stored in this variable. You can define a relevant parameter in the Parameter bar and select the same from the **Variable Name** list.
7. Close the dialog box. The **Save Confirmation** message box appears.
8. Click **YES** to save the entered details.
9. In the **Properties** pane of the added **Application Control** activity, set the **Wait After** time as **3000** as the next screen takes time to load.
10. Follow below steps to configure the subsequent steps of the automation process workflow example.

Activate Window

1. Click **ADD** to add the Application Control activity.
2. Click the  (**Field Configuration**) icon of the of the **Application Control** activity, and then bring the Oracle EBS application window in focus.




3. Hover over the **Navigator**, and the click the window to bring it in focus. A dialog box appears.

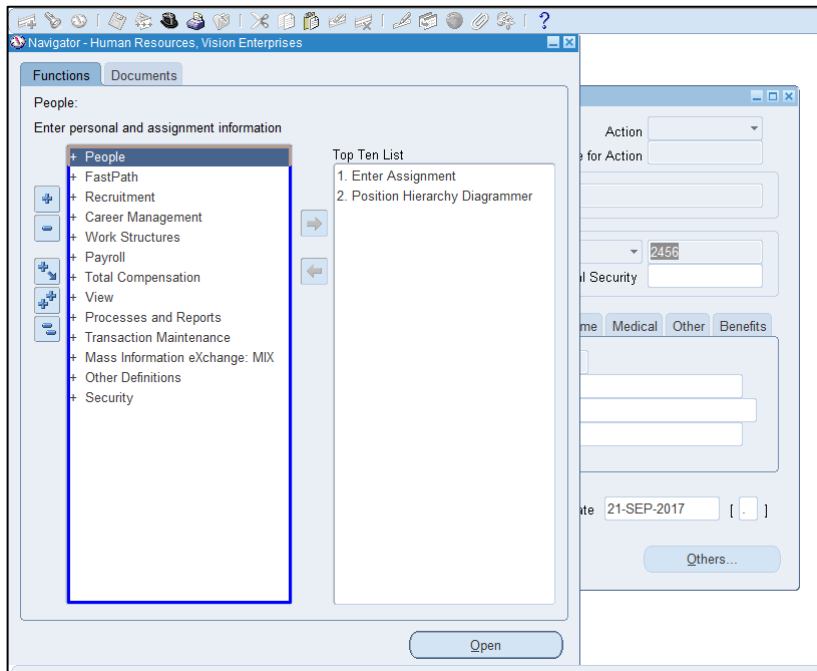


4. In the **Action** list, select **Activate window**.
5. In the **Display Name** field, enter **Bring in focus**. You can enter any name of your choice to be displayed.
6. Close the dialog box. The **Save Confirmation** message box appears.
7. Click **YES** to save the entered details.

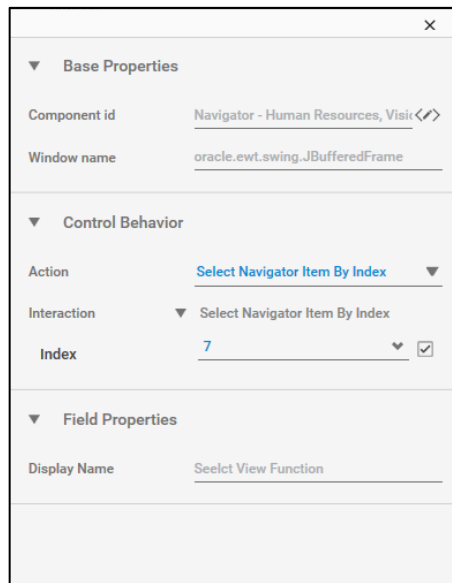
8. In the **Properties** pane of the added **Application Control** activity, set the **Wait After** time as **3000** as the next screen takes time to load.
9. Follow below steps to configure the subsequent steps of the automation process workflow example.


Select Navigate Item By Index

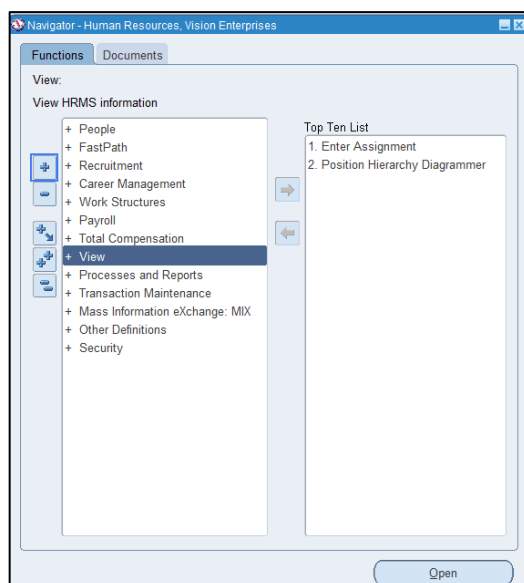
1. Click **ADD** to add the **Application Control** activity.
2. Click the  (**Field Configuration**) icon of the of the Application Control activity , and then bring the Oracle EBS application window in focus.



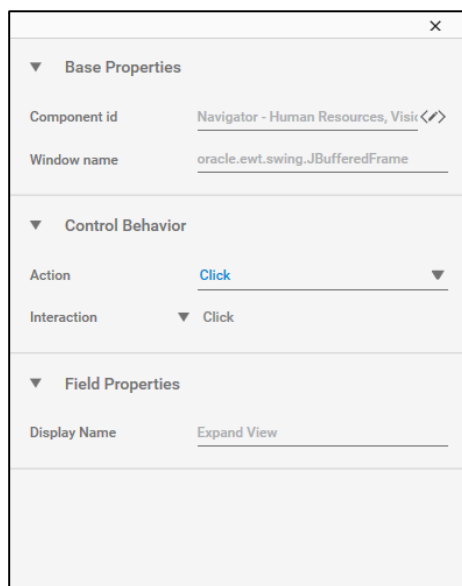
3. Hover over the **Navigator** window and click the pane where **View** function is available to extract the Id of that pane. A dialog box appears.



4. In the Action list, select Select Navigator Item By Index.
5. In the **Index** field, enter **7**. The index of the **View** function is 7.
6. In the **Display Name**, enter **Select Payroll**. You can enter any name of your choice.
7. Close the dialog box. The **Save Confirmation** message box appears.
8. Click **YES** to save the entered details.
9. Click **ADD** to add the **Application Control** activity.
10. Click the  (**Field Configuration**) icon of the of the Application Control activity, and then bring the Oracle EBS application window in focus.




11. Hover over the **Expand** button and then click the highlighted area. The dialog box appears.

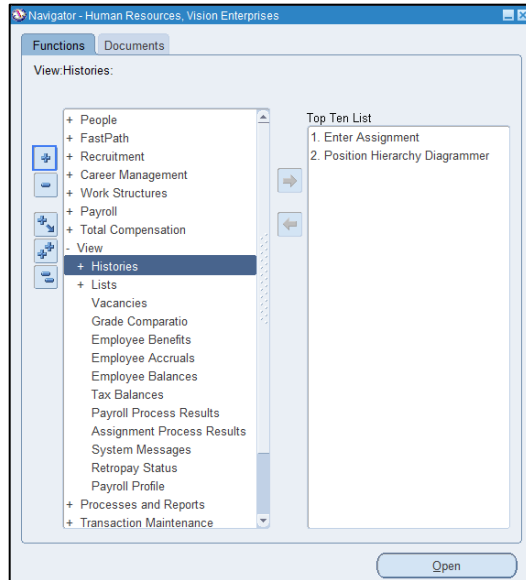


The dialog box is titled with a close button (X) in the top right corner. It is divided into three sections:

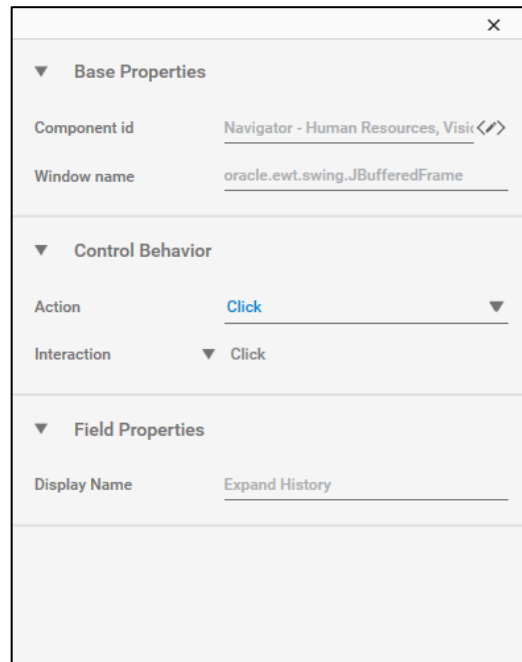
- Base Properties:**
 - Component id: Navigator - Human Resources, Visit <>
 - Window name: oracle.ewt.swing.JBufferedFrame
- Control Behavior:**
 - Action: Click (with a dropdown arrow)
 - Interaction: Click (with a dropdown arrow)
- Field Properties:**
 - Display Name: Expand View

12. In the **Action** list, select Click.
13. In the **Display Name**, enter **Expand View**. You can enter any name of your choice to be displayed.
14. Close the dialog box. The **Save Confirmation** message box appears.
15. Click **YES** to save the entered details. Since we have to select and expand the first item, we would directly configure the action to expand the History function. If you need to select any other item, configure **Select Navigator Item By** Index and specify the index number accordingly.
16. In the **Properties** pane of the added **Application Control** activity, set the **Wait After** time as **3000** as the next screen takes time to load.
17. Click **ADD** to add the **Application Control** activity.


- Click the  (**Field Configuration**) icon of the of the Application Control activity, and then bring the Oracle EBS application window in focus.

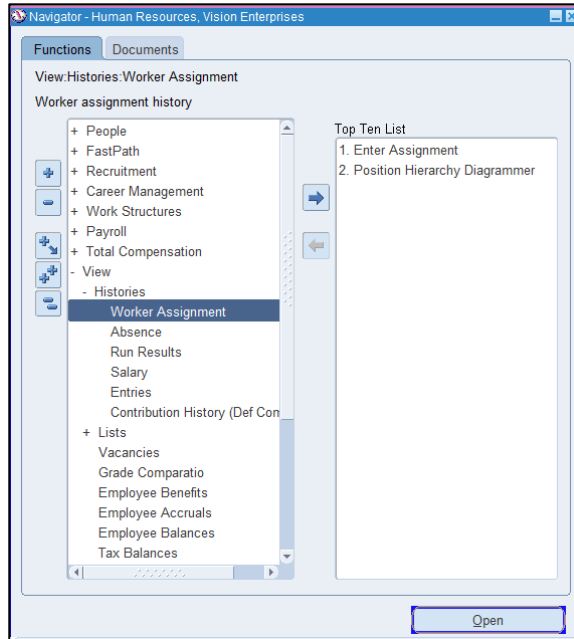


- Hover over the **Expand** button and then click the highlighted area. The dialog box appears.

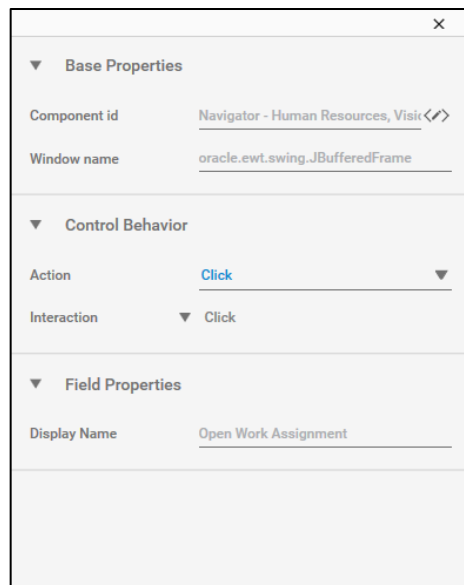


- In the **Action** list, select **Click**.
- In the **Display Name**, enter **Expand History**. You can enter any name of your choice to be displayed.
- Close the dialog box. The **Save Confirmation** message box appears.
- Click **YES** to save the entered details. Since we have to select and open the first item, we would directly configure the action to open the **Work Assignment** window. If you need to select any other item, configure **Select Navigator Item By Index** and specify the index number accordingly.


- 24. In the **Properties** pane of the added **Application Control** activity, set the **Wait After** time as **3000** as the next screen takes time to load.
- 25. Click **ADD** to add the **Application Control** activity.
- 26. Click the  (**Field Configuration**) icon of the of the Application Control activity, and then bring the Oracle EBS application window in focus.

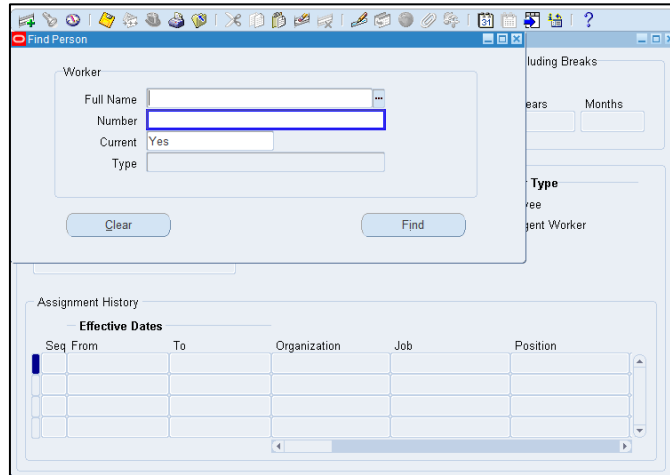


- 27. Hover over the **Open** button and then click the highlighted area. The dialog box appears.
- 28. In the **Action** list, select **Click**.
- 29. In the **Display Name**, enter **Click Open**. You can enter any name of your choice to be displayed.

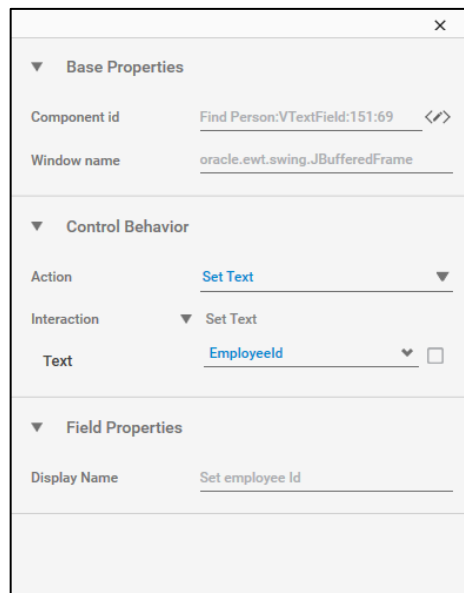


- 30. Close the dialog box. The **Save Confirmation** message box appears.


- 31. In the **Properties** pane of the added **Application Control** activity, set the **Wait After** time as **3000** as the next screen takes time to load.
- 32. Click **ADD** to add the **Application Control** activity.
- 33. Click the  (**Field Configuration**) icon of the of the Application Control activity, and then bring the Oracle EBS application window in focus.

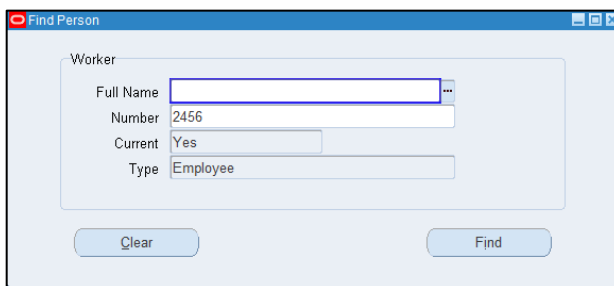


- 34. Hover over the **Number** field and then click the highlighted area. The dialog box appears.

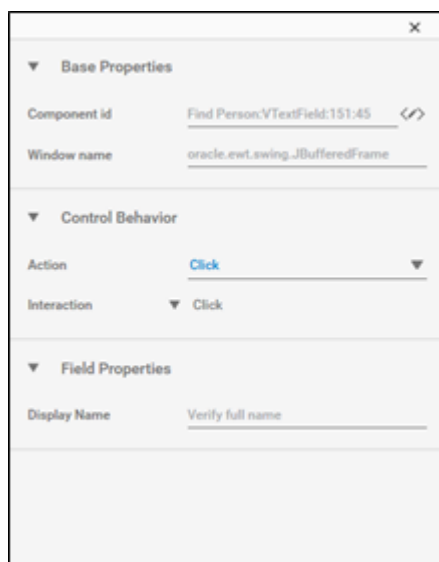


- 35. In the **Action** list, select **Set Text**.
- 36. In the **Display Name**, enter **Set employee Id**. You can enter any name of your choice to be displayed.
- 37. Close the dialog box. The **Save Confirmation** message box appears.
- 38. In the **Properties** pane of the added **Application Control** activity, set the **Wait After** time as **3000** as the next screen takes time to load.
- 39. Click **ADD** to add the **Application Control** activity.


- 4.0. Click the  (**Field Configuration**) icon of the of the Application Control activity, and then bring the Oracle EBS application window in focus.

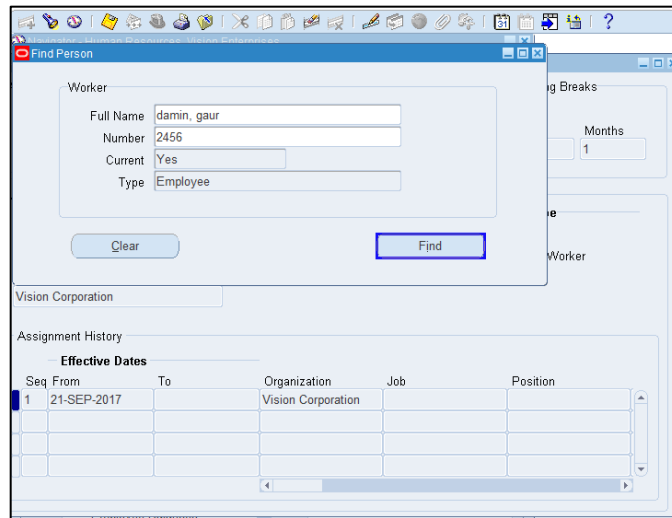


- 4.1. Hover over the **Full Name field** and then click the highlighted area. The dialog box appears.

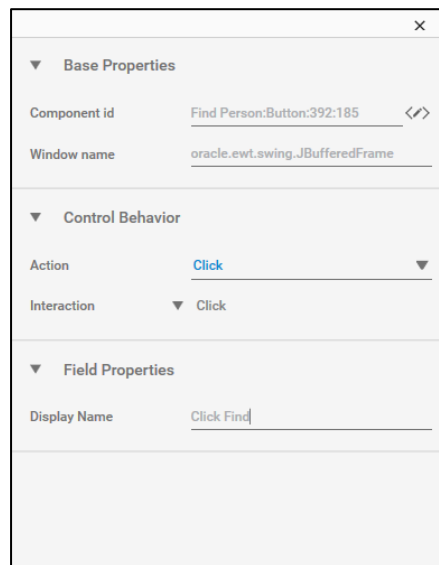


- 4.2. In the **Action** list, select **Click**.
- 4.3. In the **Display Name**, enter **Click Verify full name**. You can enter any name of your choice to be displayed.
- 4.4. Close the dialog box. The **Save Confirmation** message box appears.
- 4.5. In the **Properties** pane of the added **Application Control** activity, set the **Wait After** time as **3000** as the next screen takes time to load.
- 4.6. Click **ADD** to add the **Application Control** activity.


47. Click the  (**Field Configuration**) icon of the of the Application Control activity, and then bring the Oracle EBS application window in focus.

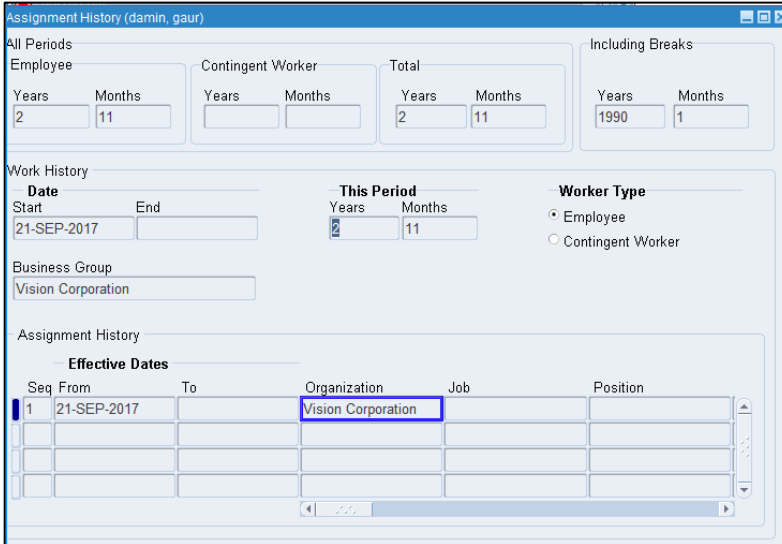


48. Hover over the **Find** button and then click the highlighted area. The dialog box appears.



49. In the **Action** list, select **Click**.
50. In the **Display Name**, enter **Click Find**. You can enter any name of your choice to be displayed.
51. Close the dialog box. The **Save Confirmation** message box appears.
52. In the **Properties** pane of the added **Application Control** activity, set the **Wait After** time as **3000** as the next screen takes time to load.
53. Click **ADD** to add the **Application Control** activity.

54. Click the  (**Field Configuration**) icon of the of the Application Control activity, and then bring the Oracle EBS application window in focus.

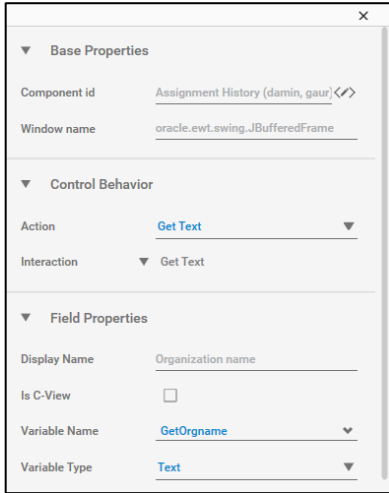


The screenshot shows the 'Assignment History' window with the following sections:

- All Periods:** Employee (Years: 2, Months: 11), Contingent Worker (Years: , Months:), Total (Years: 2, Months: 11), Including Breaks (Years: 1990, Months: 1).
- Work History:** Date (Start: 21-SEP-2017, End:), This Period (Years: 2, Months: 11), Worker Type (Employee, Contingent Worker).
- Business Group:** Vision Corporation.
- Assignment History Table:**

Seq	From	To	Organization	Job	Position
1	21-SEP-2017		Vision Corporation		

55. Hover over the **Organization** field and then click the highlighted area. The dialog box appears.



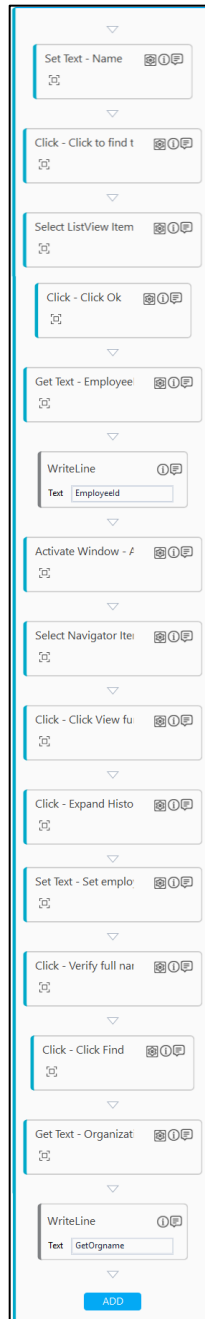
The dialog box shows the following properties:

- Base Properties:** Component id: Assignment History (damin, gaur), Window name: oracle.ewt.swing.JBufferedFrame.
- Control Behavior:** Action: Get Text, Interaction: Get Text.
- Field Properties:** Display Name: Organization name, Is C-View: , Variable Name: GetOrgname, Variable Type: Text.

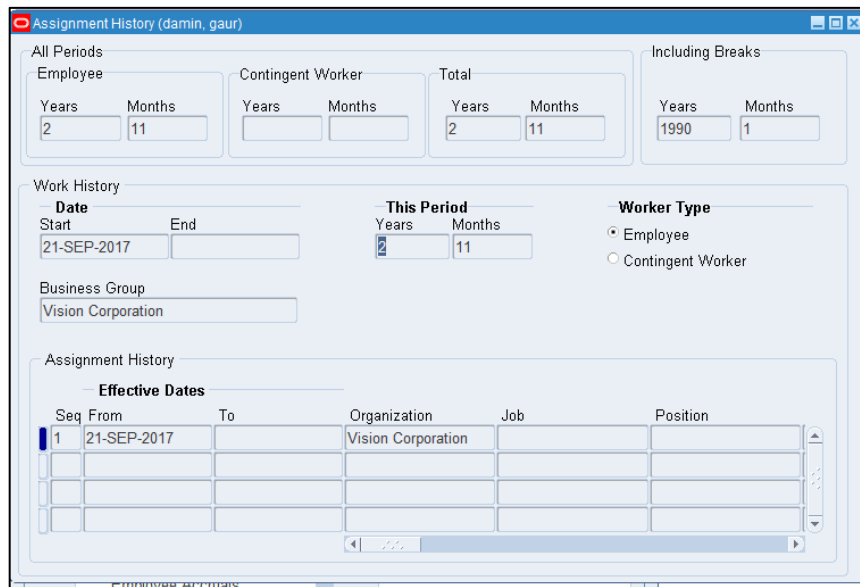
56. In the **Action** list, select **Get Text**.
57. In the **Display Name**, enter **Click Find**. You can enter any name of your choice to display to be displayed.
58. In the **Variable Name** field, enter **GetOrgname**. The variable gets created and the extracted data gets stored in this variable. You can define a relevant parameter in the Parameter bar and select the same from the **Variable Name** list.
59. Close the dialog box. The **Save Confirmation** message box appears.
60. Click **YES** to save the entered details.
61. Close the dialog box. The **Save Confirmation** message box appears.
62. In the **Properties** pane of the added **Application Control** activity, set the **Wait After** time as **3000** as the next screen takes time to load.

To view the organization name in Automation Studio, let's add WriteLine activity. You can assign this process to a robot, if you want to execute this process outside Automation Studio.

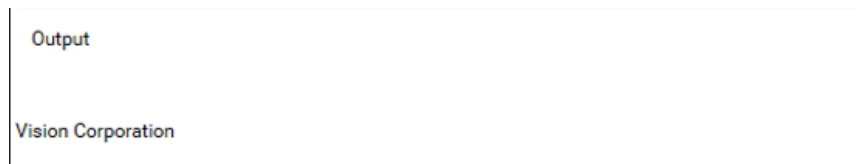
- 63. Add a **WriteLine** activity, and in the **Text** field, enter **GetOrgname** to print the organization name that gets saved in this variable.
- 64. Save the process.
- 65. Reset and then set up the environment.
- 66. Perform the test run. Below is the screen shot of how the process workflow looks:



67. You can view all the actions taking place on the Oracle EBS Form automatically. On the completion of the execution of the process workflow, the **Assignment History of Damini Gaur** gets displayed.

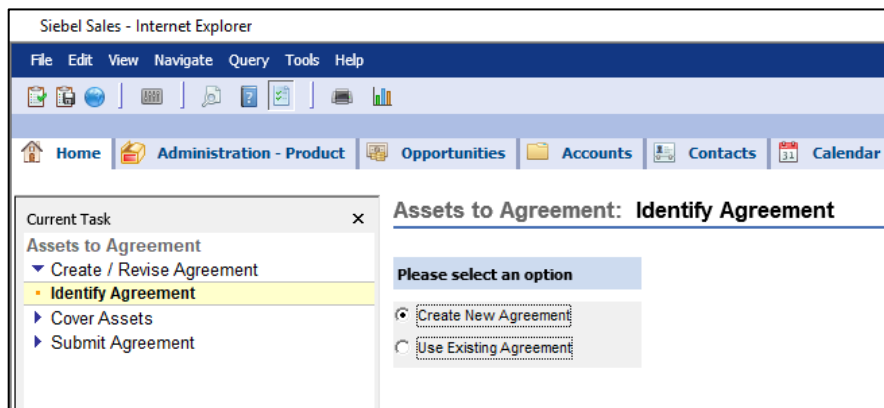


68. The **Output** console of Automation Studio displays the organization name retrieved.




Step-By-Step Guide to Use Application Host - Windows to Select the Agreement on the Siebel CRM Web Application

Let's see an example of selecting the Use Existing Agreement radio button option available on the Assets to Agreement: Identify Agreement task of the Siebel CRM web application. By default, the radio button, Create New Agreement is selected, as shown below:



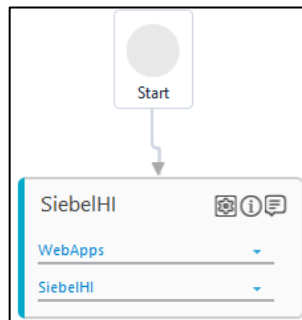
Additionally, this example demonstrates the scenario of using the Application Host activity to retrieve the required field name manually of the Siebel CRM web application, where few field names, mostly radio buttons are not retrieved using the manual process.

Prerequisite:

1. In the **Admin** menu, add an application of **Application Type- WebApps**.
2. In the Application Properties panel:
 - a. Enter the **Login URL** of the Siebel CRM application.
 - b. Enter the **Search URL** of the Siebel CRM application.
 - c. Enter the **Display Name**. In this example, **SiebelHI** is entered.
3. Click the  (**Save Properties**) icon to save the application details.
4. The Siebel CRM application is configured.

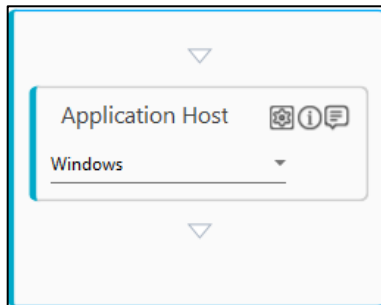
To select the required radio button option:


1. Create a **SignIn Process** for the configured Siebel CRM application. **Publish** and Deploy the **Sign In process**.
2. From the Canvas Tools panel, add Application activity to the Flowchart designer on the Canvas
3. In the **Application Type** list, select **WebApps**.
4. In the **Select an Application** list, select the **SiebelHI** option.

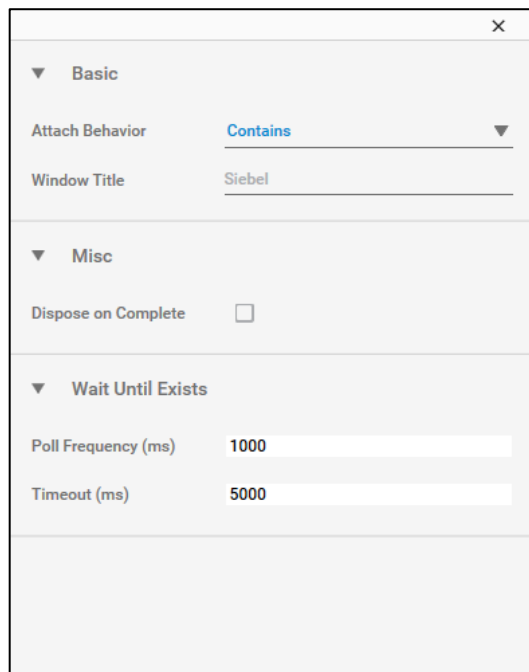


5. In the **Menu bar** of the **Studio** menu, click **Test Run** drop down, and then click **Setup Environment** to setup and launch the configured application using the Automation Studio.
6. Double click the **Application** activity and add the **Application Host** activity inside the **Application** activity.

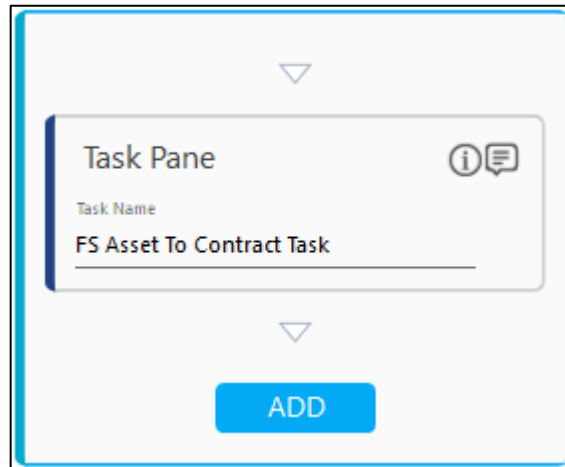
- In the **Application Host** activity, select **Windows** from the drop down list.



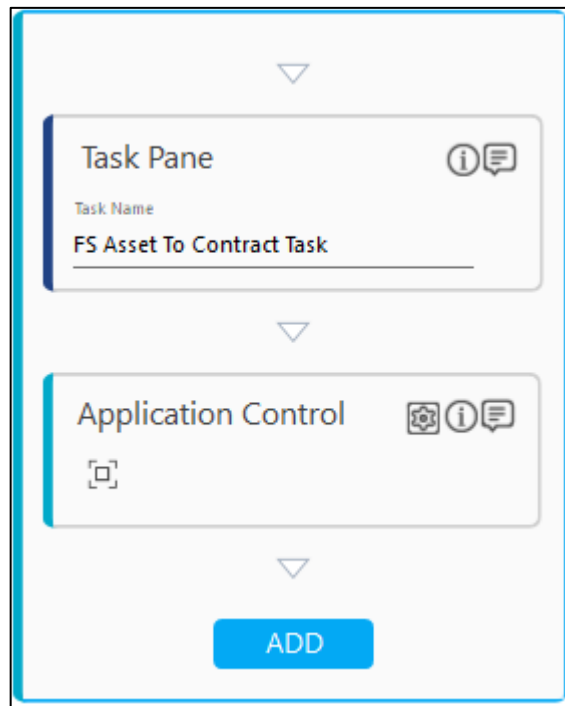
- Click the  (**Settings**) icon of the **Application Host** activity to attach the plug in based on the type of selected host application. A dialog box appears where you can define configurations related to the attached plug in.




- In the **Attach Behavior** list of the **Basics** category, select **Contains**.
- In the **Window Title** field, enter **Siebel**. Other default settings remain same. Close the dialog box to save
- the plug in configurations.
- Double click to open the **Application Host** activity block.
- Double click the **Application** activity and add the **Task Pane** activity inside the **Application** activity.
- Add the **Open Task** activity and drop inside the **Application** activity.
- In the **Task Name** field, enter the **FS Asset To Contract Task** task name.



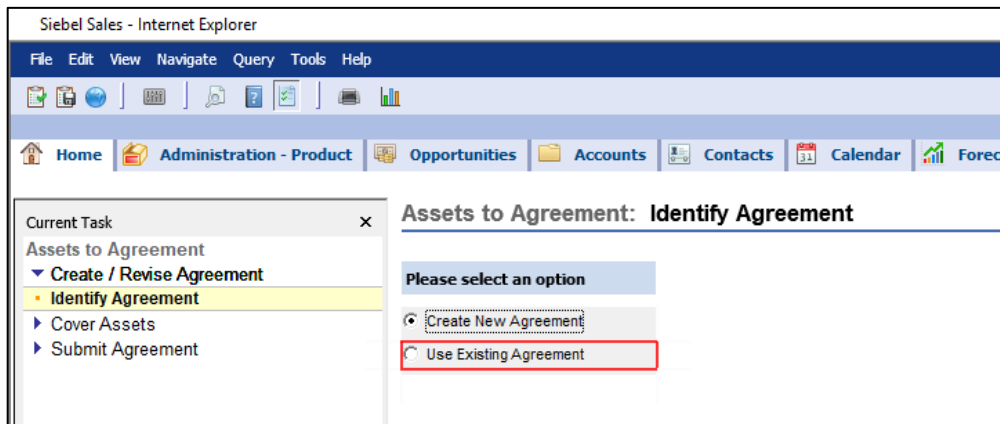
16. Click ADD to add the Application Control activity.



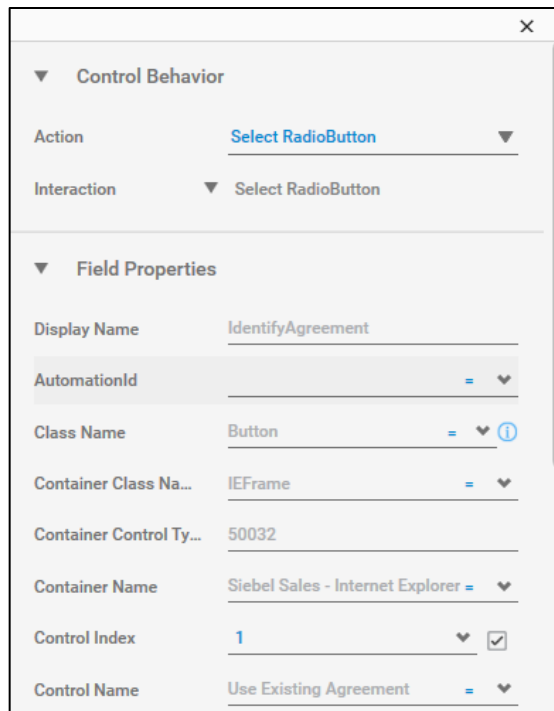
17. Click the  (Field Configuration) icon of the of the **Application Control** activity. The configured Siebel application comes into focus.

18. Press the **Ctrl** key from the keyboard and hover over the **Use Existing Agreement** radio button that needs to be selected.

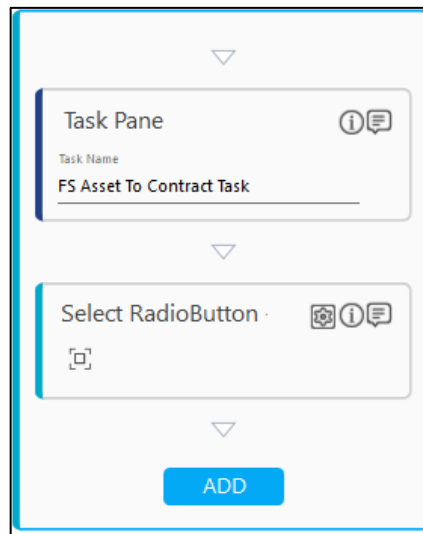
19. While pressing the **Ctrl** button click the highlighted area. A dialog box appears where you can configure the details related to the captured UI element. The fields that are mandatory are highlighted with red box.



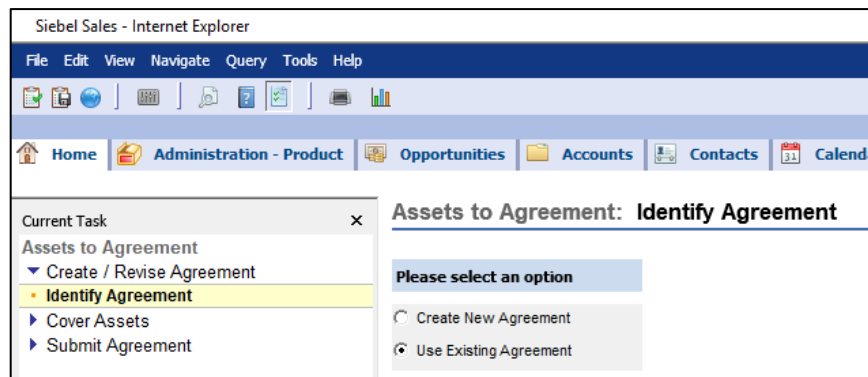
20. In the **Action** list, select the **Select RadioButton** option.
21. In the **Display Name** field, enter a name of your choice. In this example we have entered **IdentifyAgreement**.



22. Close the dialog box. The **Save Confirmation** message box appears. Click **YES** to save the entered details. Below is the sample process:



23. To view the output in Automation Studio, reset and then set up the environment. Perform the test run. You can assign this process to a robot, if you want to execute this process outside Automation Studio.
24. Observe that the Siebel HI application gets automatically logged in after the sign in details are entered and the **Asset to Agreement: Identify Agreement** task page opens. The **Use Existing Agreement** radio button is selected, as shown below:



9.6.7 Microbot

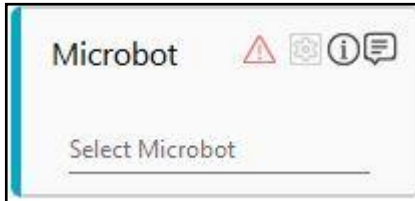
It allows you to create an automation process workflow using microbots that are already available in Automation Studio. A microbot is a piece of code built to do specific tasks that cannot be performed using predefined activities of Automation Studio.

Prerequisite

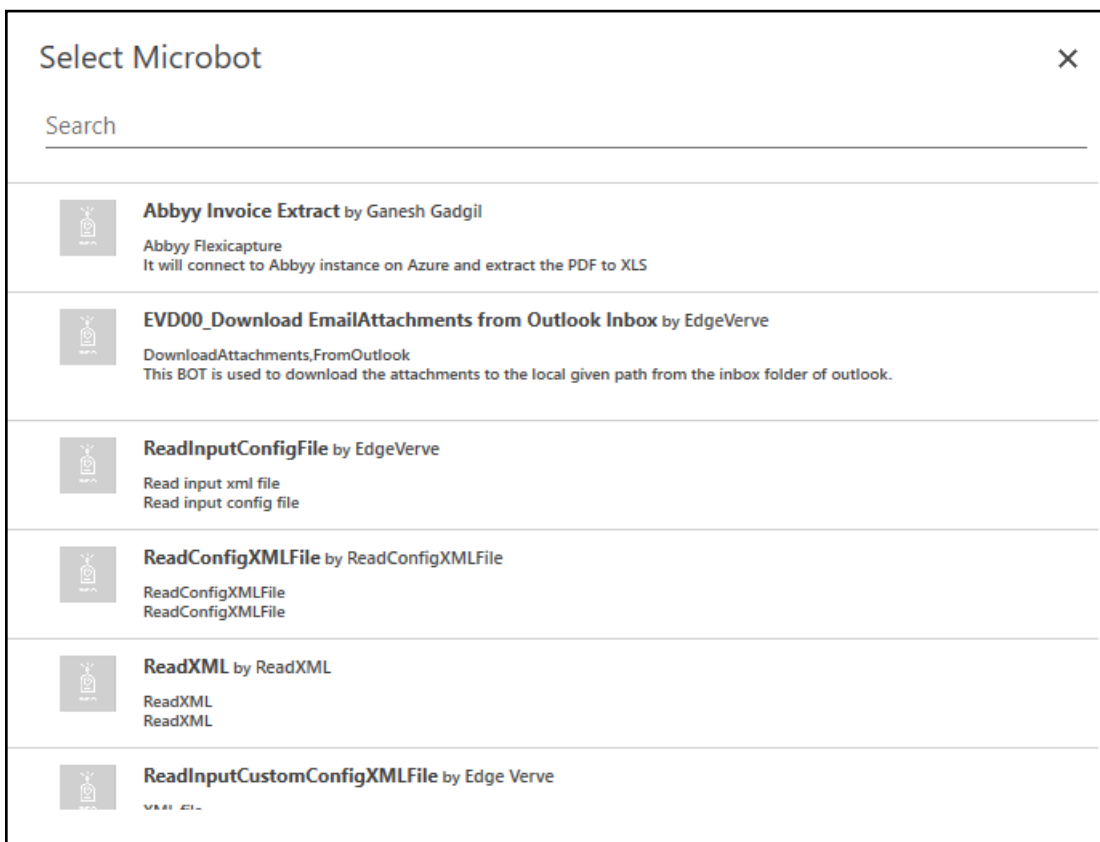
- [Create a microbot in Admin menu](#)

Using Microbot Activity


1. In the Canvas Toolpane, click Process **Components** to expand the tool and view the associated activities.
2. Drag the **Microbot** activity and drop on to the Flowchart designer on the Canvas

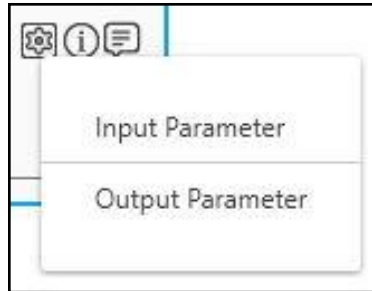


3. Click **Select** Microbot field. The Select Microbot dialog box appears.

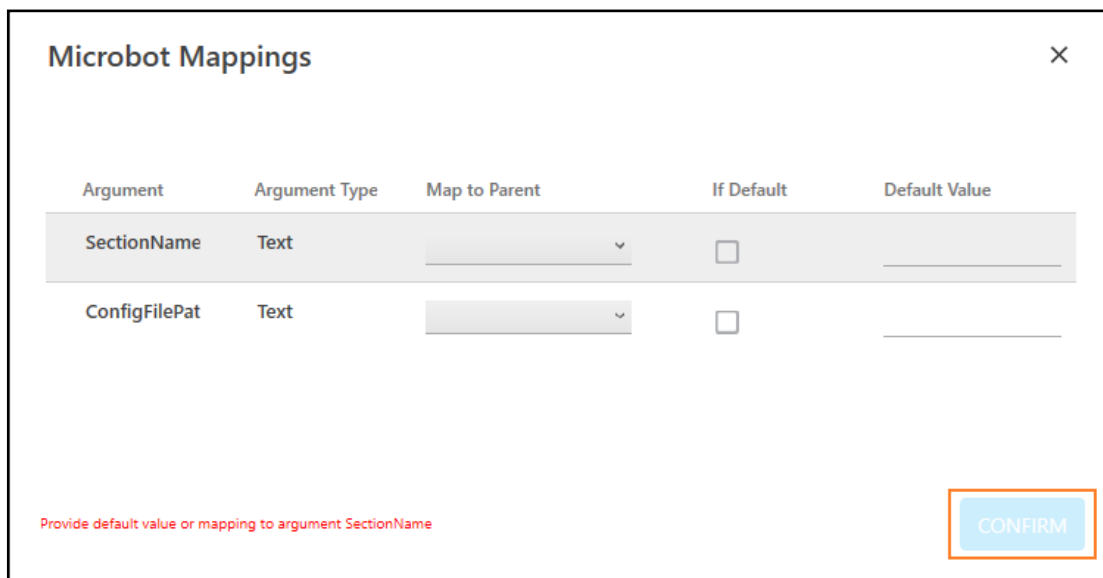


Select the required microbot from the available list. The display name of the Microbot activity changes to the name of the microbot selected.

4. Click the  (**Settings**) icon, to set the input and output parameters. The input parameter is used to map the values with the corresponding input parameter defined in the DLL file.



- Click **Input Parameter**. The **Microbot Mappings** dialog box appears.


 A screenshot of the 'Microbot Mappings' dialog box. The dialog has a title bar with 'Microbot Mappings' and a close 'X' button. Below the title bar is a table with the following columns: 'Argument', 'Argument Type', 'Map to Parent', 'If Default', and 'Default Value'. There are two rows of data:

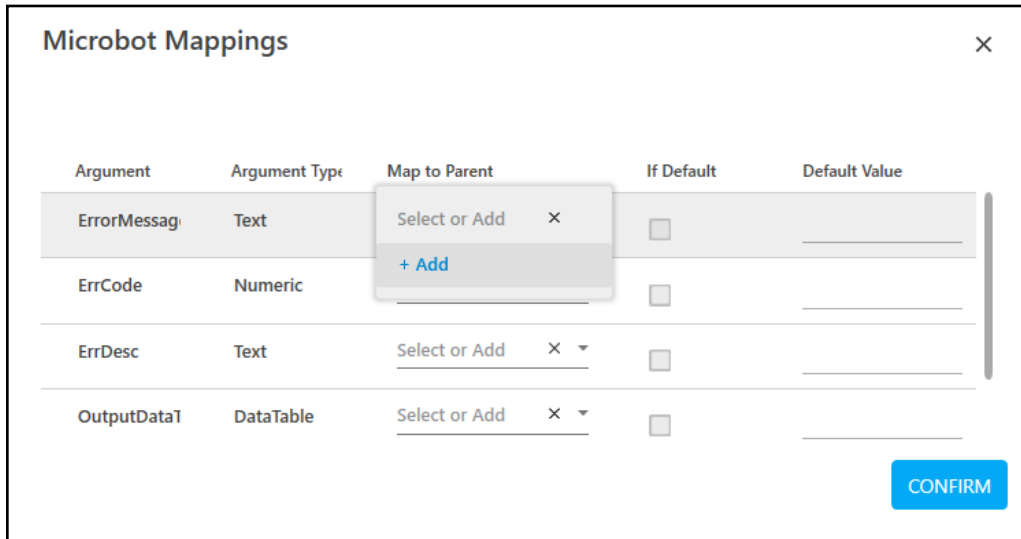
Argument	Argument Type	Map to Parent	If Default	Default Value
SectionName	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
ConfigFilePat	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>

 At the bottom right of the dialog, there is a blue 'CONFIRM' button. A red text message at the bottom left reads: 'Provide default value or mapping to argument SectionName'.

The **Argument** and the **Argument Type** columns display the input parameter and the type of the parameter respectively. The value of these parameters can either be defined now or in the microbot itself.

- In the **Map to Parent** drop-down list, select the parameter you want to map. You must define the parameter in the Parameter bar with input values for mapping. Alternatively, in the **Default Value** field, provide the input value and check **If Default** checkbox to set the value as default input value. Refer **Parameter** section in the **AE-RPA-Engage-AutomationStudio-UserManual.pdf** if you want to know more about parameters and how to use it.
- Repeat step a for each input parameter.
- Click **CONFIRM**. The input parameters are set

- Click **Output Parameter**. The **Microbot Mappings** dialog box appears.



The Argument column and the Argument Type column displays the output parameters and the type of parameters defined in the microbot.

- In the **Map to Parent** drop-down list, select the argument you want to map with the corresponding output parameter defined in the microbot. To use this option, the parameter must be pre-defined in the **Parameter** bar with the input values for mapping. You can also add a parameter at this point of time. Alternatively, in the **Default Value** field, provide the input value and check **If Default** checkbox to set the value as default input value.
- Repeat step a for each input parameter.
- Click **CONFIRM**. The output parameters are set

The Microbot activity is created.

Microbot Activity Properties

The properties of a Microbot activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.

Property Name	Usage
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as Microbot . You can change the name as required.

9.6.8 Code Editor

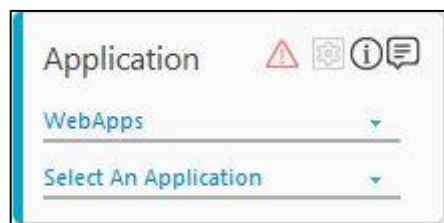
A code editor is a piece of code built in Visual Basic to do specific tasks related to certain fields or perform operations associated with the extracted fields of an application that could not be automated using out-of-box activities available in Automation Studio.

You can use an existing code editor DLL or create a new code editor using an in-built script editor available with the Code Editor activity.

Note: This activity must be used inside an Application activity or the system displays an error. This lets you establish a connection between Automation Studio and the configured application.

Using Code Editor Activity

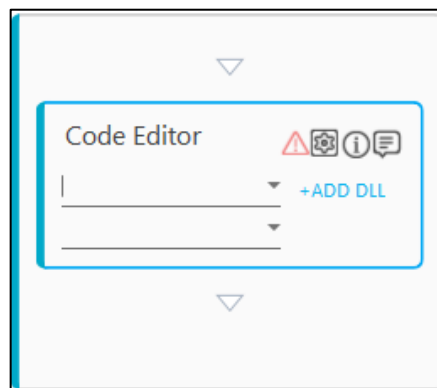
1. Within an automation process workflow, click **Process Components** in the **Canvas Tools** pane to expand the tool and view the associated activities.
2. Drag the **Application** activity and drop on to the Flowchart designer on the Canvas



3. In the **Application Type** list, select **WebApps**. You must have at least one application added for it to appear in the list.
4. In the **Select An Application** list, select the application where you want to perform automation. Alternatively, you can add a new web application at this point of time. To add web application:

- a. In the **Select An Application** list, click **Add New** Application. The **Add New Application** dialog box appears.

- b. In the **Application Name** field, enter a desired name of the web application.
- c. In the **LoginURL** field, enter the login URL of the web application you want to access.
- d. In the **SearchURL** field, enter the URL of the page that appears immediately post login.
- e. In the **PreferredBrowser** list, select the browser you prefer to launch the web application. By default, preference is set to **InternetExplorer**. Available options are:
- InternetExplorer
 - FireFox
 - Chrome
 - InternetExplorerSelenium
 - MicrosoftEdge
- f. Click **SAVE**. The web application is added.
5. Capture steps that you want to automate. If you want to perform specific tasks related to certain fields or perform operations associated with the extracted fields of the application that could not be automated using the out-of-box activity then follow the below steps to for configuring Code Editor activity in the automation process workflow.
6. Double click the **Application** activity, drag the **Code Editor** activity and drop inside the **Application** activity.



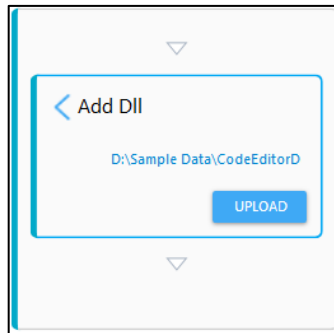
7. You can do one of the following:
 - Add_an_Existing_Code_Editor_DLL
 - Create_Code_Editor_DLL_in_Automation_Studio


Add an Existing Code Editor DLL

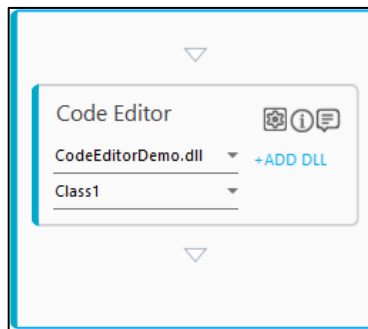
Prerequisite

Create the custom code in Visual Studio and add the ICodeEditor interface of Automation Studio to your project. Refer AE-CE-Customization Guide.pdf to know the steps to create a code editor DLL file.

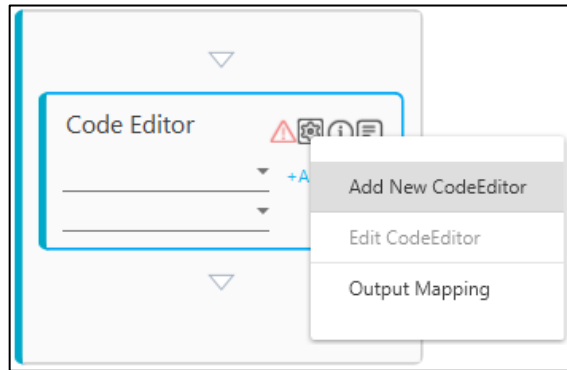
1. Click **+ADD DLL** and then click File Location to browse and select an existing code editor DLL file.
2. Click **UPLOAD**. The **Upload Successful** message is displayed.



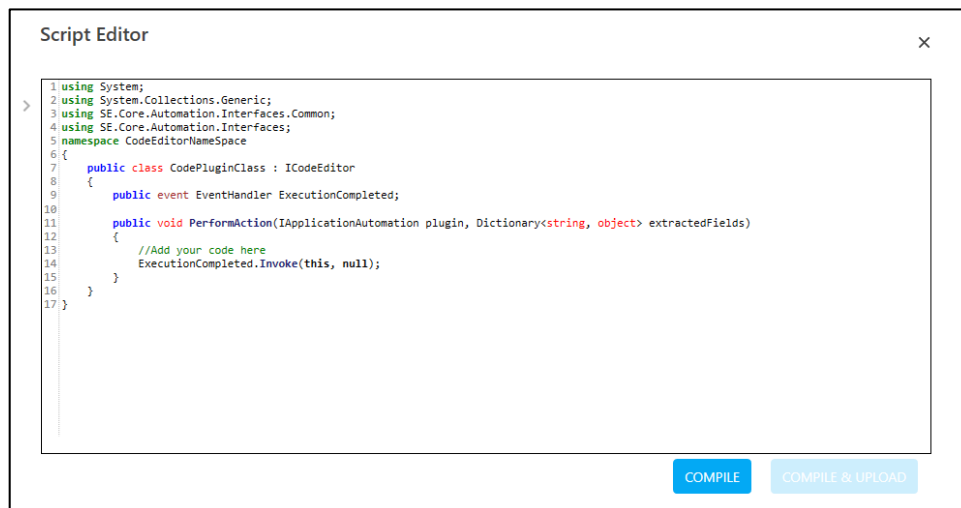
3. Click  to return to the activity block. The uploaded DLL is selected in the list. The list of all the available classes is displayed in the list below the selected DLL.
4. Browse and select the required class as per your requirement.



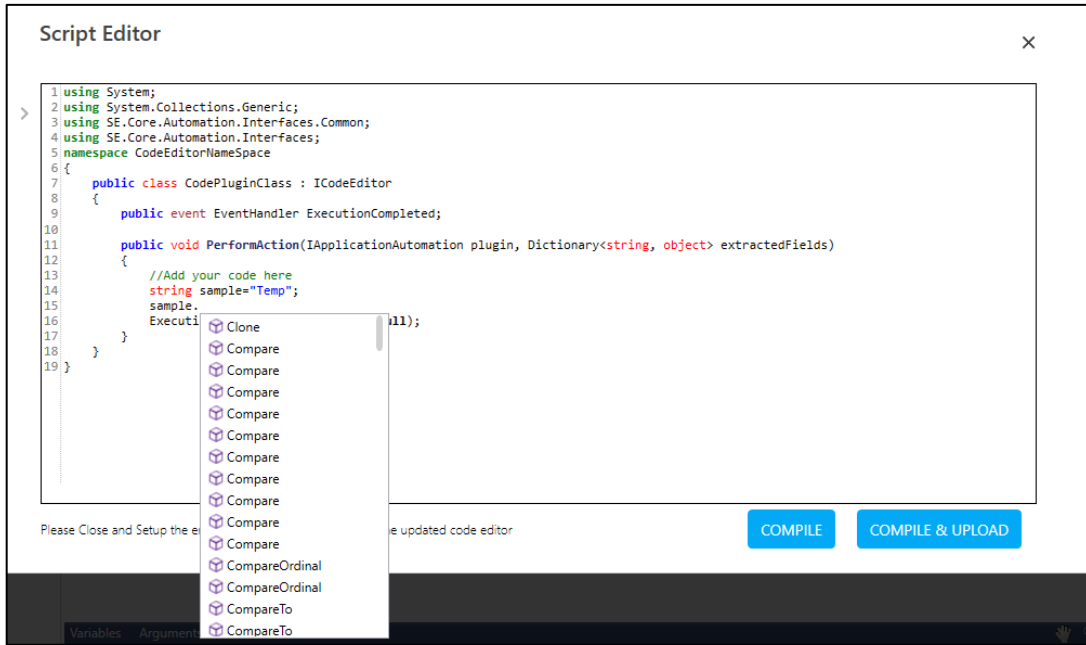
Create Code Editor DLL in Automation Studio




1. Click the  (Settings) icon and then click **Add New CodeEditor**. The **Script Editor** dialog box appears with the default template and the required interface already added.




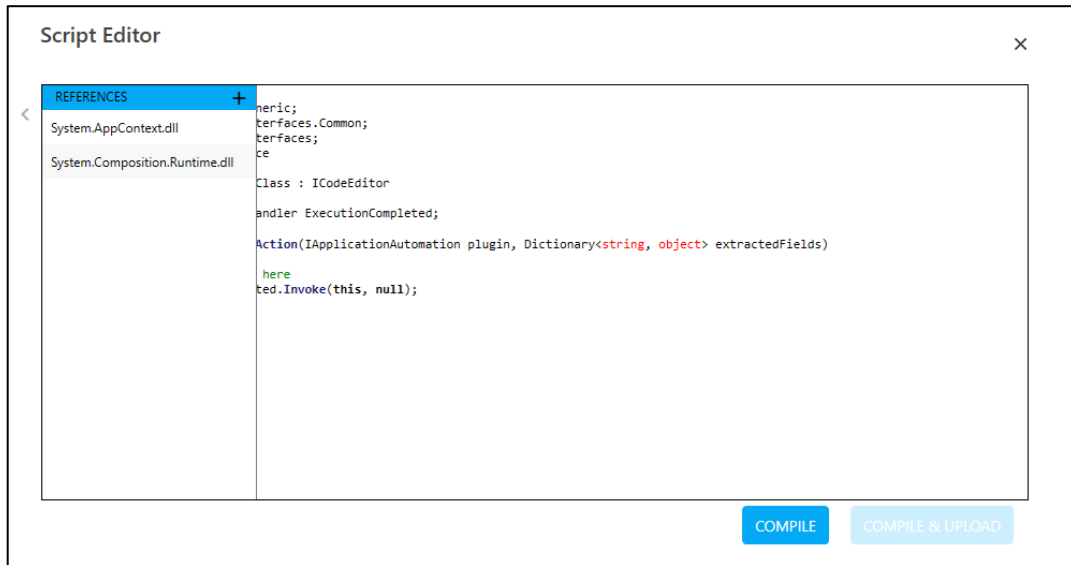
- Write the required code under the **PerformAction** method. The in-built script editor provides expression building aid using the IntelliSense feature of Automation Studio.



- After the code editor operations are completed, raise the **ExecutionCompleted** event.
- Click the  icon to open the **REFERENCES** pane. This lets you add references of third-party DLLs to the code written.




- Click the  icon to browse and add the required external DLL references. The **Reference added successfully** message is displayed and the reference starts appearing in the **REFERENCES** pane. Repeat this step to add multiple references to the code.




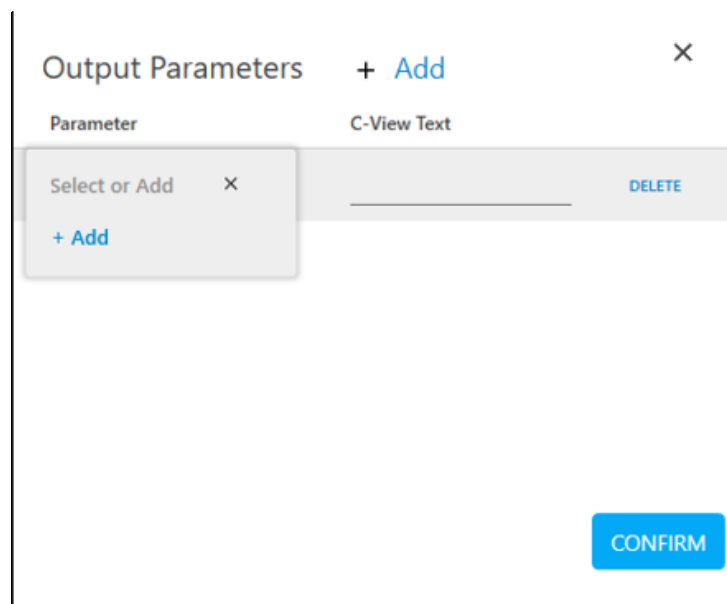
- Click **COMPILE** to check for any compilation error in the code. Errors found are highlighted; else, the **No Compilation Errors** message is displayed. Close the displayed message.



- Click **COMPILE & UPLOAD** to upload and save the code editor DLL in the AssistEdge database. The Save Code Editor dialog box appears.



8. In the **CODEEDITOR NAME** field, enter a desired name of the code editor, and then click **SAVE**. The code editor is successfully uploaded and saved in the AssistEdge database.
9. Click the  (**Settings**) icon, and then click **Output Mapping**. The **Output Parameters** dialog box appears. This lets you store the output of the code editor DLL file (if any) in a parameter of the automation process workflow.

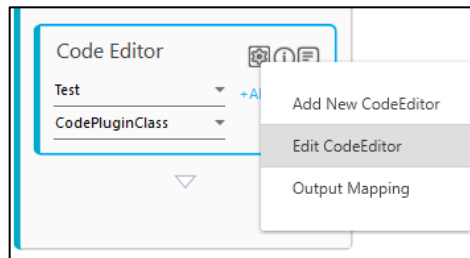



10. Click **Add** to add a new row for mapping the required parameter to store the output.
11. In the **Parameter** drop down list, select the parameter you want to map with the corresponding output defined in the code editor DLL. To use this option, the parameter must be pre-defined in the Parameter bar. Alternatively, you can also add a parameter at this point of time. Follow the mentioned to steps to add the parameter:

- a. Click **Select or Add** and then click **Add**. The **Add new Argument** dialog box appears.

- b. In the **Argument Name** field, enter a desired name.
 - c. In the **DataType** list select the type of parameter you want to create. Available options are – **Text, Numeric, Decimal, Boolean** and **DateTime**. As this parameter stores the output value, providing default value is not enabled.
 - d. Click **DELETE** if you do not want to save the provided details or skip to step v.
 - e. Click **CONFIRM** to save the mapping.
12. Repeat step 9 for multiple output received.
 13. The code editor for the required operation is configured.

Editing the Code Editor Created in Automation Studio



1. Select the required code editor created in Automation Studio.
2. Click the  (**Settings**) icon, and then click Edit **CodeEditor** to edit the uploaded code editor. The **Script Editor** dialog box appears with the selected code editor. Make the changes as per your requirement. You can find and replace text in the **Script Editor** dialog box. To find and replace the text:
 - a. Press the **Ctrl+F** keys from the keyboard to open the **Find** box.



The screenshot shows the Script Editor window with a search bar at the top right containing the text 'test'. The code in the editor is as follows:

```

1 using System;
2 using System.Collections.Generic;
3 using SE.Core.Automation.Interfaces.Common;
4 using SE.Core.Automation.Interfaces;
5 namespace CodeEditorNameSpace
6 {
7     public class CodePluginClass : ICodeEditor
8     {
9         public event EventHandler ExecutionCompleted;
10
11         public void PerformAction(IApplicationAutomation plugin, Dictionary<string, object> extractedFields)
12         {
13             //Add your code here
14             string sample = "test";
15             ExecutionCompleted.Invoke(this, null);
16         }
17     }
18 }

```

At the bottom of the window, there are two buttons: 'COMPILE' and 'COMPILE & UPLOAD'.

- b. In the **Find** box, enter the text that you want to find and then press **Enter** key. the required text gets highlighted.
- c. Press the **Ctrl+H** keys from the keyboard to open the **Replace** box.
- d. In the **Replace** box, enter the text to replace the existing text.



The screenshot shows the Script Editor window with the search bar containing 'test' and a Replace dialog box open below it. The dialog box has 'ReplaceTest' entered in the 'Replace with' field. The code in the editor is the same as in the previous screenshot:

```

1 using System;
2 using System.Collections.Generic;
3 using SE.Core.Automation.Interfaces.Common;
4 using SE.Core.Automation.Interfaces;
5 namespace CodeEditorNameSpace
6 {
7     public class CodePluginClass : ICodeEditor
8     {
9         public event EventHandler ExecutionCompleted;
10
11         public void PerformAction(IApplicationAutomation plugin, Dictionary<string, object> extractedFields)
12         {
13             //Add your code here
14             string sample = "test";
15             ExecutionCompleted.Invoke(this, null);
16         }
17     }
18 }

```

At the bottom of the window, there are two buttons: 'COMPILE' and 'COMPILE & UPLOAD'.

- e. Click **Replace** to replace the text.
 - f. You can press **Ctrl+Z** and **Ctrl+Y** keys respectively to undo and redo any changes if required.
3. Once you have made changes in the code, compile and re-upload the code for the changes to get saved and reflected in the code editor.
 4. The code editor DLL is edited and saved in the AssistEdge database.

Code Editor Activity Properties


The properties of a Code Editor activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity.</p> <p>If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as Code Editor. You can change the name as required.
DIIDisplayName	The name of the selected code editor DLL.

Step-By-Step Guide to Use Code Editor to Change the File Extension

Let's see an example of changing the file extension of a .txt file to .jpeg file using the Code Editor activity and print a message for successful change of the file extension.

To create the automation process:

1. Create a new process.
2. In the **Parameter** bar:
 - Create an **In** argument, **inputargname** and define the file path of the .txt file.
 - Create an **Out** argument, **outputargname** to store the display message.
3. From the **Canvas Tools** panel, add the **Application** activity and in the **Application Type** list, select **WindowsApps**.
4. In the **Select An Application** list, add a new windows application at this point of time. To add windows application:
 - a. In the **Select An Application** list, click Add New Application. The Add New Application dialog box appears.
 - b. In the **Application Name** field, enter a desired name of the windows application.
 - c. In the **Path** field, enter the notepad EXE file available on the system.
 - d. In the **Additional Parameters** field, enter the path of the file (along with its name) that you want to open within the application.
 - e. Click **SAVE**. The windows application is added.
5. Double click the **Application** activity, drag the **Code Editor** activity and drop inside the **Application** activity.
 - a. Click the  (**Settings**) icon and then click **Add New CodeEditor**. The **Script Editor** dialog box appears with the default template and the required interface already added.

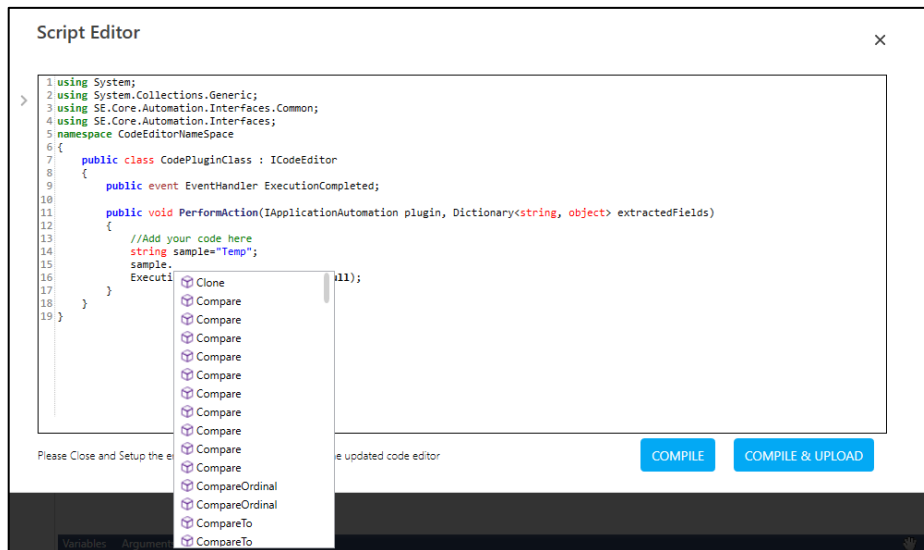


```

1 using System;
2 using System.Collections.Generic;
3 using SE.Core.Automation.Interfaces.Common;
4 using SE.Core.Automation.Interfaces;
5 namespace CodeEditorNameSpace
6 {
7     public class CodePluginClass : ICodeEditor
8     {
9         public event EventHandler ExecutionCompleted;
10
11         public void PerformAction(IApplicationAutomation plugin, Dictionary<string, object> extractedFields)
12         {
13             //Add your code here
14             ExecutionCompleted.Invoke(this, null);
15         }
16     }
17 }
  
```

COMPILE COMPILE & UPLOAD

- b. Write the required code under the **PerformAction** method. The in-built script editor provides expression building aid using the IntelliSense feature of Automation Studio.




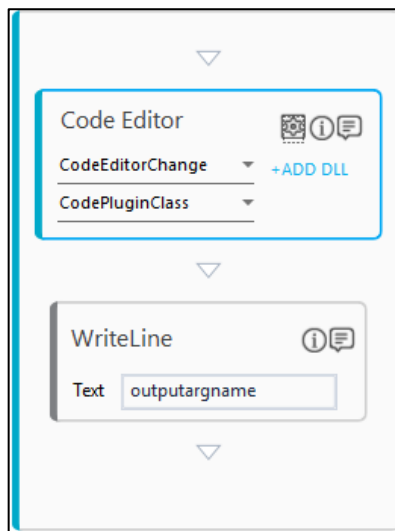
- c. After the code editor operations are completed, raise the **ExecutionCompleted** event.
- d. Click **COMPILE** to check for any compilation error in the code. Errors found are highlighted; else, the **No Compilation Errors** message is displayed. Close the displayed message.



- e. Click **COMPILE & UPLOAD** to upload and save the code editor DLL in the AssistEdge database. The Save Code Editor dialog box appears.



- f. In the **CODEEDITOR NAME** field, enter a desired name of the code editor, and then click **SAVE**. The code editor is successfully uploaded and saved in the AssistEdge database.
6. Click the  (**Settings**) icon, and then click **Output Mapping**. The **Output Parameters** dialog box appears. This lets you store the output of the code editor DLL file (if any) in a parameter of the automation process workflow.
 - a. Click **Add** to add a new row for mapping the required parameter to store the output.
 - b. In the **Parameter** drop down list, select the **outputargname** parameter you want to map with the corresponding output defined in the code editor DLL.
 - c. Click **CONFIRM** to save the mapping.
7. To view the output in Automation Studio, let's add **WriteLine** activity. You can publish, deploy and assign this process to a robot, if you want to execute the process outside Automation Studio.
8. Add **Writeline** activity and in the **Text** field, enter the **outputargname** argument.
9. Save the process. Below is the sample automation process workflow:



10. Setup the environment and perform test run.
11. The Output console displays the specified message:

Output

File extention changed successfully

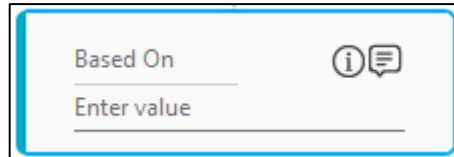
Studio Console Output Watch Errors Warnings Execution Stats Validation Error

9.6.9 Validation

This activity is used in sign-in processes to check for sign in success.

It compares the value extracted from the application screen after successful sign-in with the static value provided in the activity.

In case of web apps, it can compare the URL as well.



9.6.10 Popup Handler


This activity allows you to automate the actions that you want to perform on a popup dialog window in front of the web or an excel application. Examples of a pop up window are authentication pop up, confirmation alert and so no.

This activity must be used inside an Application activity or the system displays an error.

Using Popup Handler Activity

1. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
2. Drag the **Popup Handler** activity and drop to the Flowchart designer on the Canvas



3. Click the  (**Settings**) icon, to configure the controls of the pop up. The **Popup Configurator** dialog box appears. The configuration of control differs based upon the type of application and the type of browser used to open the application. Following are the steps to configure the control based on the type of application and browser:

- If the **Popup Handler** activity is used for an IE based web application or an Excel application:

Control Type	Control Id	Field To Map
TextBox		
Button		

- Click **Add** to add the control.
- In the **Popup Window Caption**, enter the name of the pop up window you want to handle.
- In the **Control Type** list, select the type of control to be automated. Currently, text box and button type are supported. A text box needs user input while a button needs while a button needs an action. For example, while creating Sign-in process, if popup prompt for credentials appear, you can use the text box option of the Popup Handler activity to provide the inputs.
- In the **Control Id** field, enter the Id of the control to automate. The Id attribute is an identifier by which the selected control is accessed in the code, for example, ffnf.
- In the **Field To Map** list, select the available argument in the process workflow to map the value set in the text box. You must define the parameter in the Parameter bar to make it available for selection. This option is disabled if the **Control Type** selected is **Button**.

- If the **Popup Handler** activity is used for a Chrome or a Firefox based web application:

The screenshot shows the 'Popup Configurator' interface. At the top, there is a title bar with '+ Add' and 'X' buttons. Below the title bar is a text input field for 'Popup Window Caption'. Underneath is a table with three columns: 'Control Type', 'Control Id', and 'Field To Map'. The 'Control Type' dropdown is currently set to 'Button', and a context menu is open over it, showing 'Accept' and 'Cancel' options. To the right of the table, there is a 'DELETE' button. At the bottom right of the interface, there is a blue 'CONFIRM' button.

- In the **Popup Window Caption**, enter the name of the popup window you want to handle. This field is not relevant when the application opens in a Chrome or a Firefox browser but the caption must be set as a dummy value.
- In the **Control Type** list, select the type of control to be automated. Currently, text box and button type are supported. A text box needs user input while a button needs while a button needs an action. For example, while creating Sign-in process, if popup prompt for credentials appear, you can use the text box option of the **Popup Handler** activity to provide the inputs.
- In the **Control Id** list, select the action to perform on the popup handler. Available options are **Accept** and **Cancel**. This option is disabled if the **Control Type** selected is **Textbox**.
- In the **Field To Map** list, select the available argument in the process workflow to map the value set in the text box. You must define the parameter in the Parameter bar to make it available for selection. This option is not available if **Control Type** selected is **Button**. The Popup Handler activity is created.

Popup Handler Properties

The properties of a Focus Window activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as Popup Handler . You can change the name as required.
Window Title	Name of the pop up window. Is same as Popup Window Caption .

9.6.11 API Activity

This activity allows you to perform SAP automation using BAPI (Business Application Programming Interface). It also invokes REST APIs.

9.6.12 SOAP API Activity

Soap Activity can be used to make SOAP web services over http(s) protocol hosted at a given endpoint.

Application Type: API type application has to be created to use the soap service activity.

Add Application [Close]

Application Type: WebAPI

Application Name: SoapSample

[ADD]

Create WebAPI type of application.

Search Properties [Search]

Details [Expand]

Login URL: http://www.ecs.syr.edu/faculty/faw...

SignIn:

Type: **SOAP**

Display [Expand]

Application Descripti...: SoapSample

Display Name: SampleSoap

General [Expand]

Select the type as SOAP.

In the place of Login URL, any of the following are accepted:

1. URL of the place where the WSDL is hosted
2. UNC path of the place where the WSDL file is located

It is always suggested to have network path for the WSDL file, if file system is used.

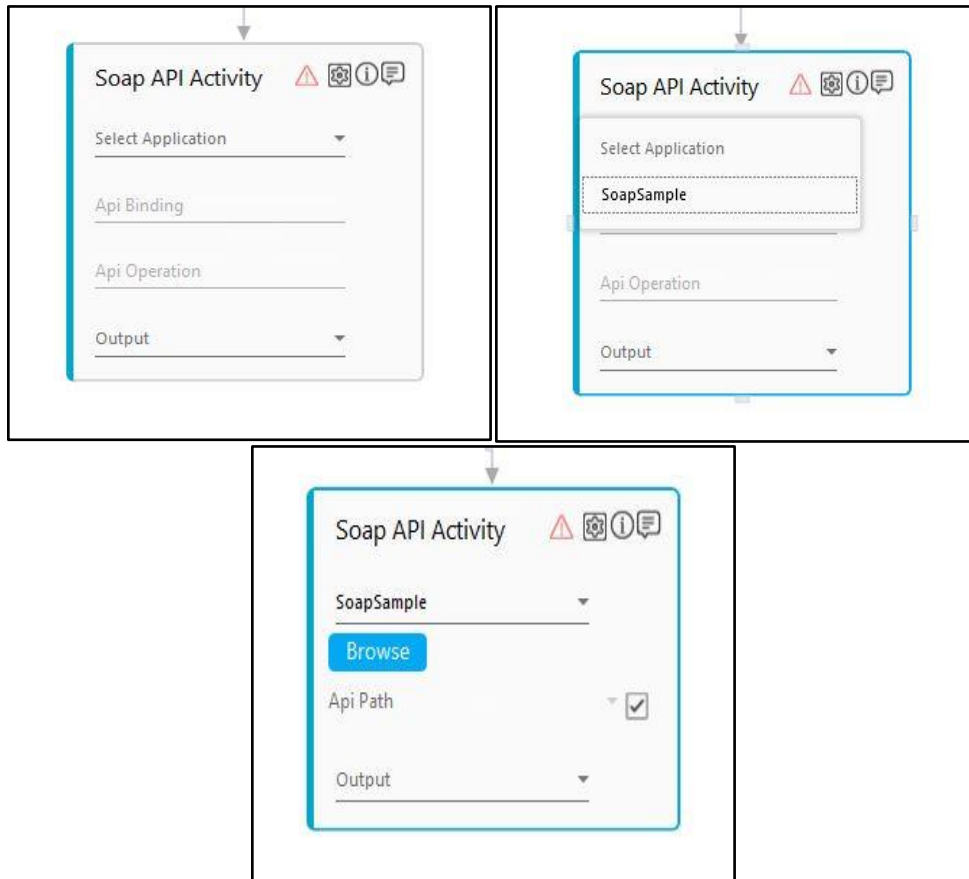
URL: <http://example.com/calcWebService/Calc.asmx?wsdl>

FILE: C:\\Users\\varun_alla\\Desktop\\soap.WSDL

<\\10.12.12.12.\\WSDLS\\soap.WSDL> (Recommended, for multiple locations)

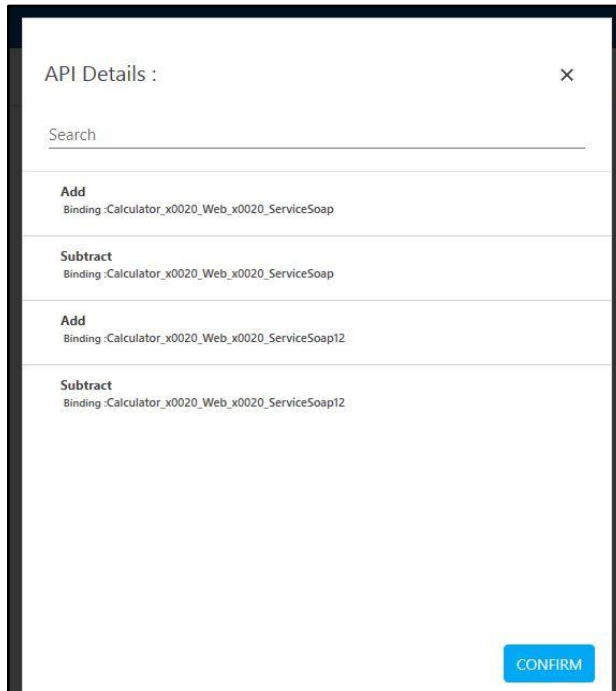
- **Step 1:** Configure App and endpoint

When the activity is dragged and dropped onto the canvas the below image appears.



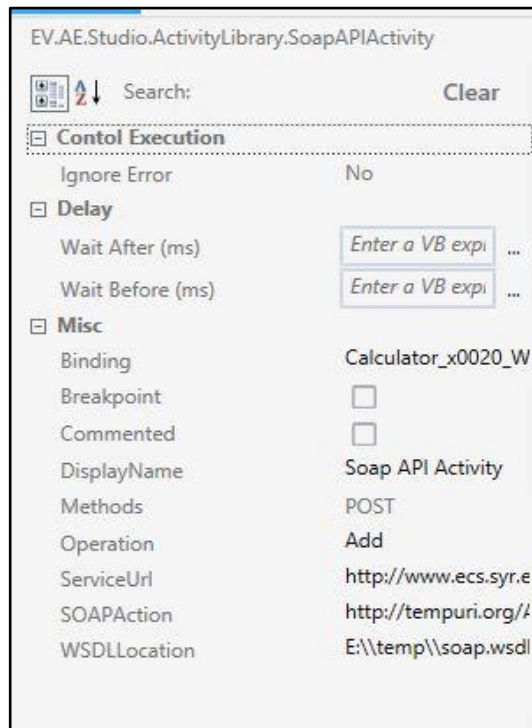
Select the Application in the Select SOAP API drop-down.

API SELECTION:

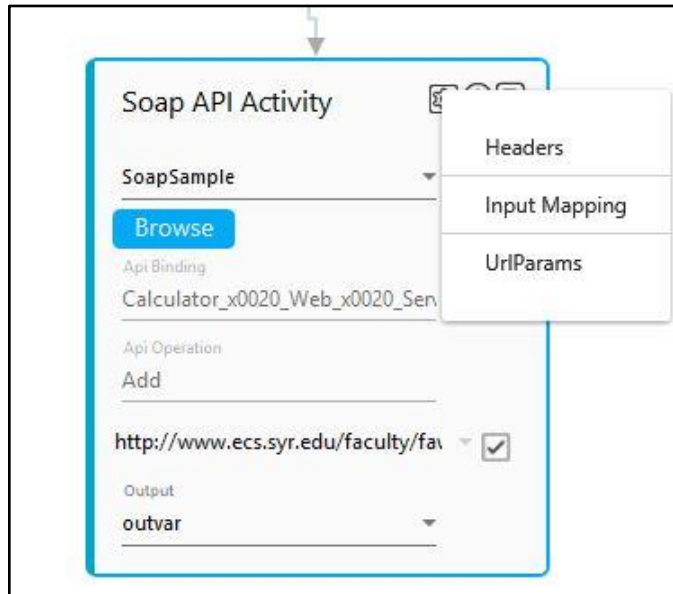


API Selection can be done by clicking on the browse button, then selecting the API.

This populates all the fields on the properties pane, shown in below image.

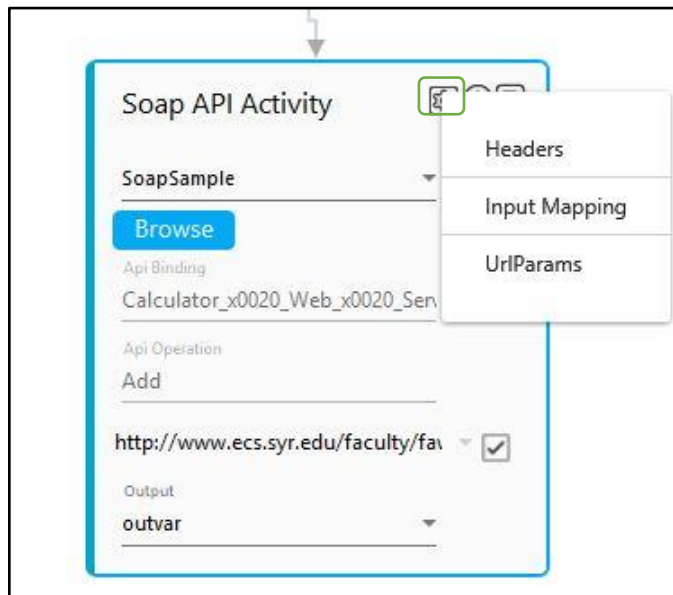


These values can be overridden by the user.

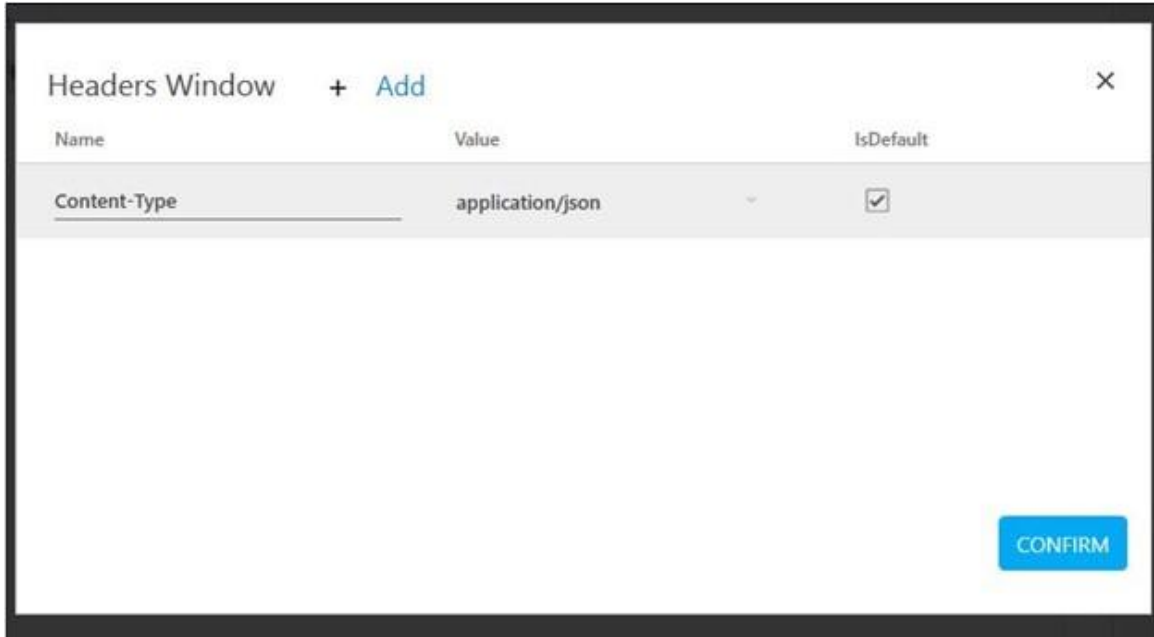


- **Step 2:** Configure Headers and URL parameters for Http Request

If you click on the gear symbol to the right, the below image appears.



Click on the Headers option to see the below window:



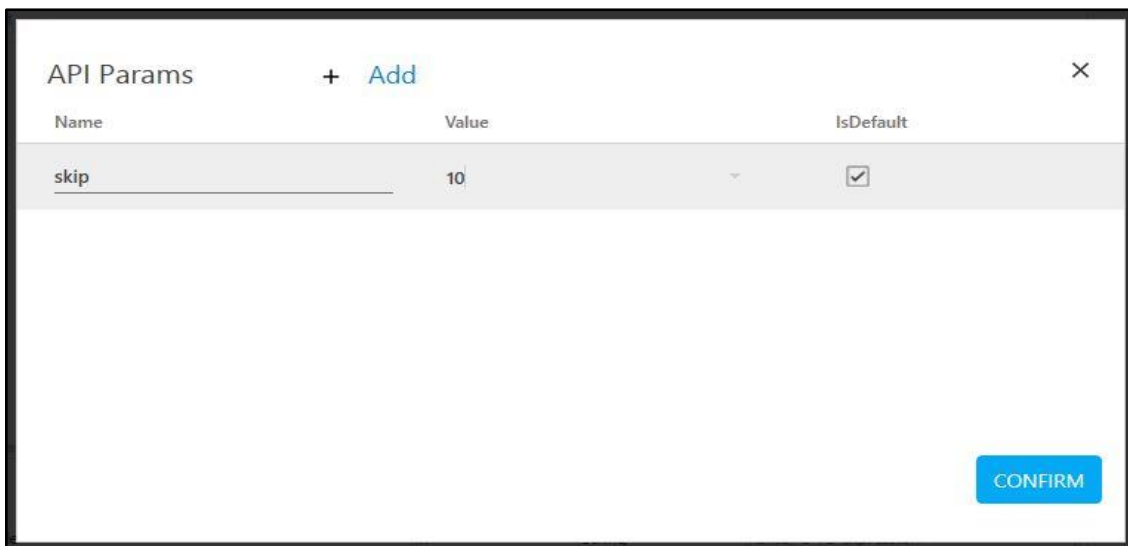
Name	Value	IsDefault
Content-Type	application/json	<input checked="" type="checkbox"/>

This window can be used to provide headers for the request.

IsDefault: If Ticked, the value is used as a string, if not, then value is used as input argument that will be resolved at run time. Just like the API PATH.

Click on the UriParams option to see the below window:

This window can be used to add query parameters to the request URL.



Name	Value	IsDefault
skip	10	<input checked="" type="checkbox"/>

Name and value represent query parameter that will be appended at the end of the URL.

Above configuration from image resolves to <http://example.com/calcWebService/Calc.aspx?skip=10>

IsDefault: If ticked, the value is used as a string, if not, then the value is used as input argument that will be resolved at run time. Just like the API PATH.

- Step 3: Configure Request Body:

SOAP body can be configured as below:



API input screen can take the XML that has to be provided for the soap request.



If there is a requirement to inject variables into the payload, it can be done as shown above.

This activity can be used along with the XML Activity to build and serve complex requests.

9.6.13 Rest API Activity

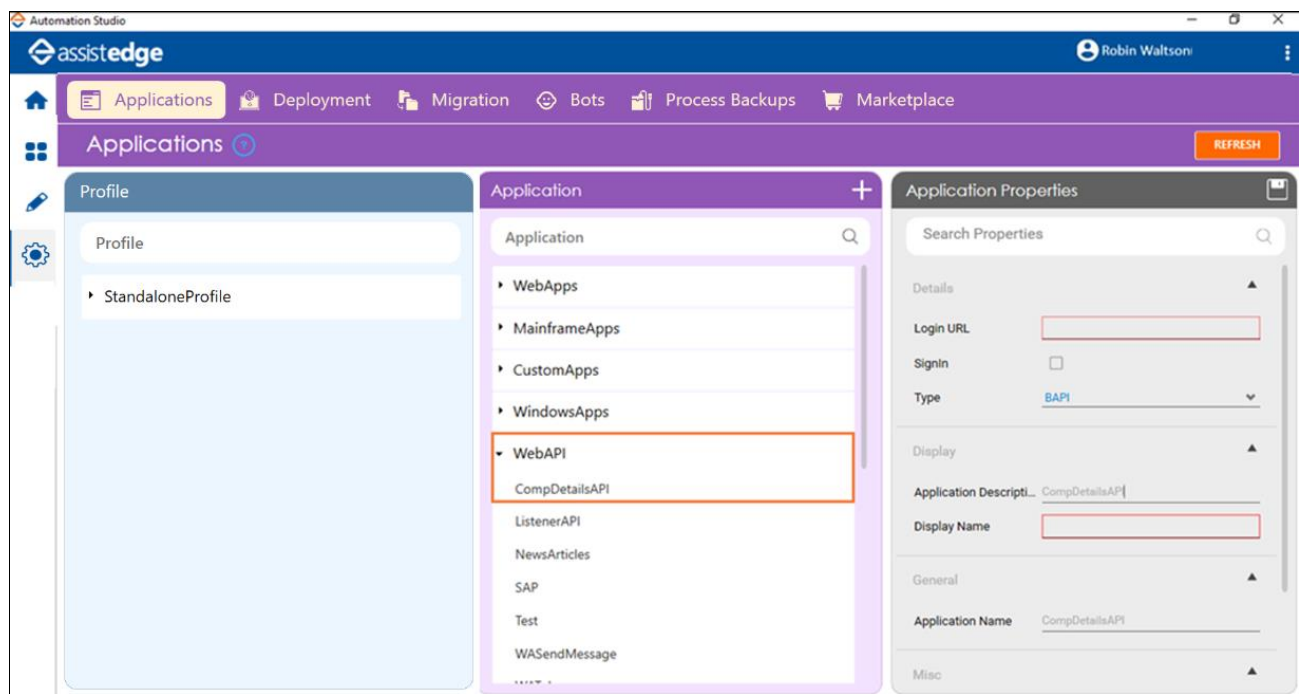
This activity allows you to utilize services of a REST API. You can define the actions performed by an API using the http methods - Get, Post, Put and Delete.”.

Using Automation Studio configure the information for the request made through an API such as the input path, URL parameters, http method, list of headers and body.

Prerequisite

Configure the REST API web service in Automation Studio where you want to perform the automation. Below are the minimum required properties for configuring the intended API. Remaining fields are auto populated with the default values already configured in Automation Studio. If you want to change the default values, refer [WebAPI](#) in Admin Capabilities.

1. In the **Admin** menu, add an application of **Application Type- WebAPI**.



2. In the Application Properties panel:
 - In the **Login URL** field, enter the login URL of the REST API you want to access.

Note: Do not provide the input path or the query parameter to the URL of the WebAPI application added to Automation Studio.

- In the **Type** list, select REST. By default, the value set is BAPI.
- In the **Display Name**, enter a desired name of the REST API application.

3. Click the  (**Save Properties**) icon to save the application details.

The WebAPI is configured.

Using REST API Activity

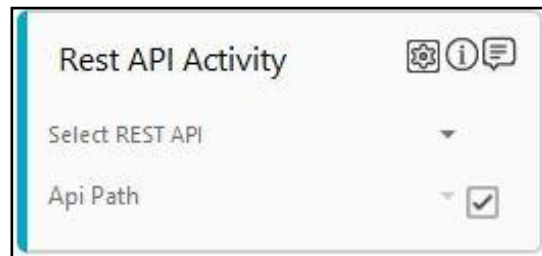
You can use the REST API Activity by forming the complete API URL that includes configuring the endpoint, header and body as per the request type selected.

Configure Input Path and URL Parameters

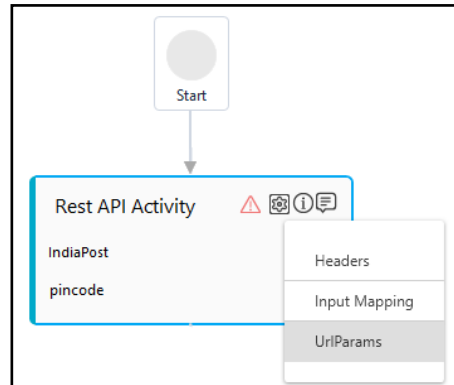
You can pass the required resource as:

- **Path Parameters:** the actual value that returns the information requested. For example, in the REST API request URL, `http://app-sample.com/v2/test/:parametervalue`, `/test` forms the input path and `parametervalue` is the actual value of the variable.
- **Query Parameter:** the key- value pair that must be passed as a parameter to return the request. For example, in the REST API request URL, `http://app-sample.com/v2/test?parameter=value`, `/test` forms the input path and `parameter=value` forms the query parameter.

1. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
2. Drag the **Rest API Activity** and drop on to the Flowchart designer on the Canvas



3. In the **Select Rest API** list, select the available WebAPI application of type **REST**.
4. In the Api Path, do one of the following:
 - If path parameters form the part of the API URL:
 - a. Select the Manual API Name check box and then enter the input path of the API URL along with the required value of the variable to set as default value that is passed as a string. Alternatively, clear the Manual API Name checkbox, and then select the available parameter holding the value of the input path and the required variable. You must define the parameter in the Parameter bar to use this option. If query parameter forms the part of the API URL:
 - b. Select the Manual API Name check box, and then enter the input path of the API URL to set as default value that is passed as a string. Alternatively, clear the Manual API Name checkbox, and then in the API Path list, select the parameter holding the value of the input path of the API URL. You must define the parameter in the [Parameter](#) bar to use this option.



- c. Click the  (**Settings**) icon, and then click **UriParams**. The **API Params** dialog box appears.

Name	Value	IsDefault
		<input type="checkbox"/>

- d. Click **Add** to provide the Url parameters.
- e. In the **Name** field, enter the name of input parameter that you want to map with the key-value pair of the query parameter. It is passed as an argument.
- f. In the **Value** field, select the parameter to map with the key. You must define the parameter and its corresponding value in the **Parameter** bar to use this option. Alternatively, select the **IsDefault** checkbox to set the value entered as the default value.
5. In the **Properties** pane, select the type of action performed by the API in the **Method** list. Depending upon the method selected, configure Header and Request Body:

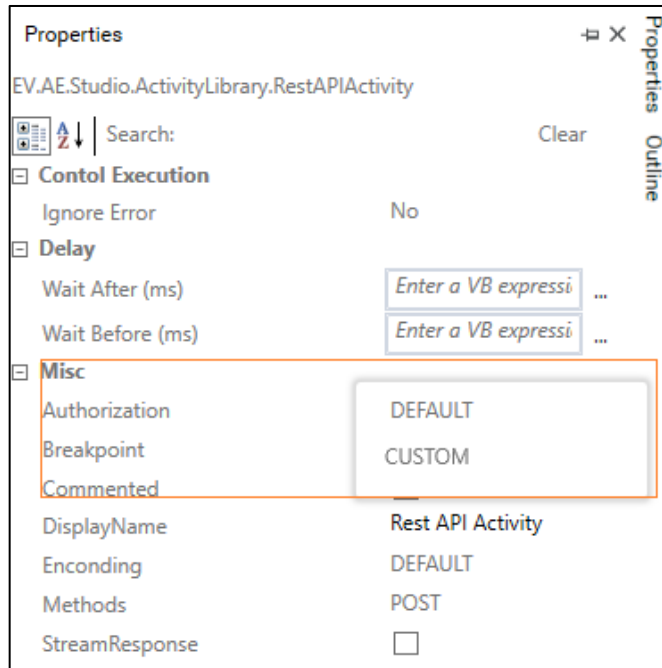
Configure Header

It allows you to authenticate the client for the resource being requested. It also provides information related to the API request and response. Headers are property-values pair such as Content-Type, Accept and others.

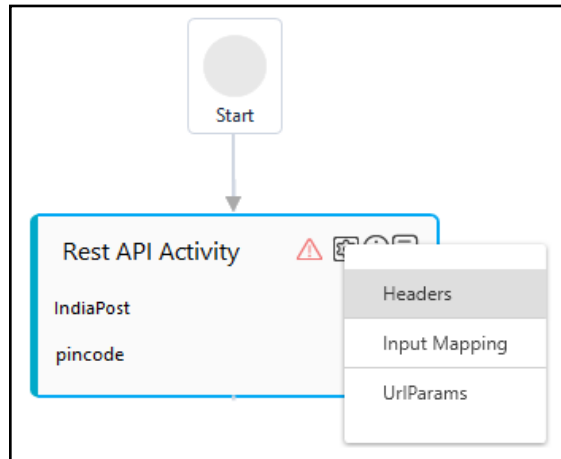
If the REST API authorization is default sign-in to the API, you can define it in the Admin menu of Automation Studio by selecting

the Signin checkbox. Refer WebAPI for more details.

The default authorization configures an authorization header that cannot be overridden. You can override the default behavior by defining the custom authorization such as passing the required token by selecting CUSTOM in the Authorization field available in the Properties pane of the selected REST API activity.



To define the request/response and authorization header:



1. Click the  (**Settings**) icon, and then click **Headers**.

The 'Headers Window' dialog box is shown. It has a title bar with 'Headers Window', '+ Add', and a close button 'X'. Below the title bar is a table with three columns: 'Name', 'Value', and 'IsDefault'. The 'IsDefault' column has a checked checkbox. There is an 'Add' button in the top right and a 'CONFIRM' button in the bottom right.

Name	Value	IsDefault
		<input checked="" type="checkbox"/>

The Header Window appears.

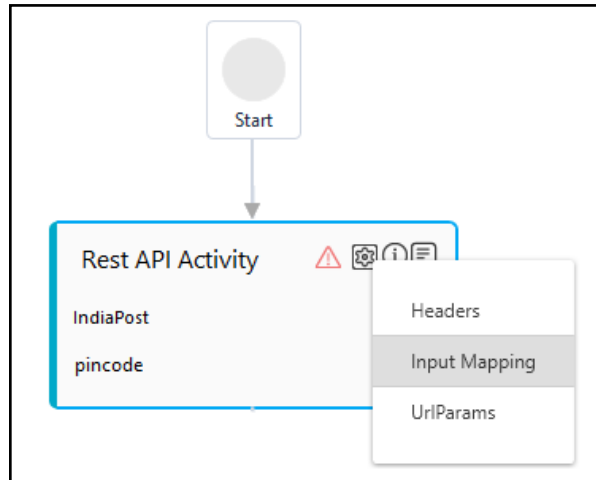
2. Click **Add** to provide the details related to the request/response and authentication headers.
3. In the **Name** field, enter a name of the property of the header.
 - a. In the **Value** field, select the parameter to map with the property. You must define the parameter in the Parameter bar to use this option. Alternatively, select the **IsDefault** checkbox to set the value entered as the default value.
4. Click **CONFIRM**.


The header is configured.

Configure Request Body

Contains information that you want to send to the server. The body can be in the JSON, or any multi-lingual string format. It is used with methods that update or insert a resource such as PUT, POST, and DELETE requests.

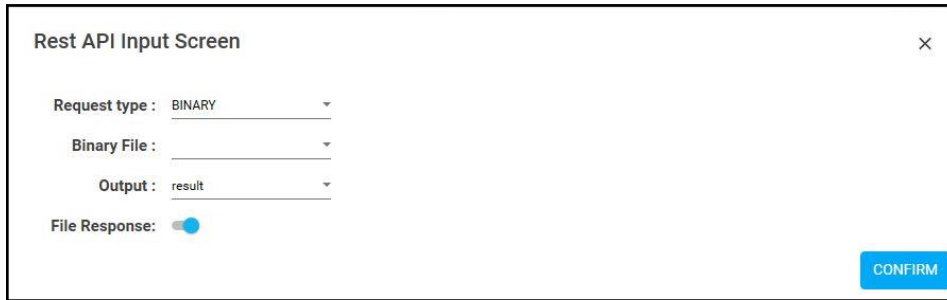
This section defines the body content of request type- PUT, POST and DELETE.



1. Click the  (**Settings**) icon, and then click **Input Mapping**.

The Rest API Input Screen appears.

2. In the **Request type** list, select the request format. Available options are, **RAW** and **BINARY**. If selected **Request type** is:
 - **RAW**- request is of JSON or html format
 - a. In the **Input** text box, enter string, html or json object.
 - b. In the **Output** list, select the parameter to map the return value of the API. You must define the parameter in the Parameter bar use this option. If the return value is a file, select the **File Response** button. The output parameter must contain the location with the filename where file needs to be downloaded.
 - **BINARY**- request is of Binary file upload type, The **Rest API Input Screen** appears.



- c. In the **Binary File** list, select the parameter that contain the location of the binary file to be uploaded. You must define the parameter in the Parameter bar use this option.
- d. In the **Output** list, select the parameter to map the return value of the API. You must define an out parameter in the Parameter bar to use this option.
- e. If the return value is a file, select the **File Response** button. The output parameter must contain the location with the filename where file needs to be downloaded.

Note: The file upload is not multipart/form data. It is binary file upload only.

- f. Click **CONFIRM**. The Request Body is configured.

The automation process workflow for the REST API is created.

REST API Activity Properties

The properties of a REST API Activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Authorization	Signifies the type of API authentication configred. Available options are: <ul style="list-style-type: none"> ▪ DEFAULT - Indicates the configured default authorization of the REST API is

	<p>used for authentication.</p> <p>If the Signin checkbox is selected in the Admin menu at the time of configuring the REST API in Automation Studio, the basic authentication (username-password) is used even if the authorization header is present. While, if the Signin check box is not selected in the Admin menu and authorization header is required for the authentication, an error is received.</p> <ul style="list-style-type: none"> ▪ CUSTOM – Indicates the custom authorization is used for the authentication. If the Signin checkbox is selected in the Admin menu at the time of configuring the REST API in Automation Studio, the basic authentication (username-password) is taken into consideration and it cannot be overridden. If authorization header is present, you can override the values. While, if the Signin checkbox is not selected in the Admin menu and the authorization header is present, it is left unmodified.
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	<p>Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.</p>
DisplayName	<p>The display name of the activity in the flowchart designer. By default, the name is set as REST API Activity. You can change the name as required.</p>
Encoding	<p>Specifies the type of character/content encoding used by the request body. Available options are - DEFAULT, UTF8 and ASCII. The DEFAULT option supports UTF8 encoding.</p>
Methods	<p>The request made by the API for retrieving or sending data to the server. Available options are:</p> <ul style="list-style-type: none"> ▪ GET- retrieves data from the server. ▪ POST-sends data to the server for creating new resource. ▪ PUT- updates an existing resource. ▪ DELETE- deletes data or resource from the server. ▪ OPTIONS- requests information about the communication options available for the target resource.
StreamResponse	<p>Signifies if the output of the web request is a file.</p>


Step-By-Step Guide to Use Rest API to Retrieve Details of a Global Company

Let us consider the API of Finnhub.io to access the company details of a global company like Apple or Microsoft.

API URL: <https://finnhub.io/api/v1/stock/profile?symbol=<symbol of the company>&token=<API authentication key>>, where

the base URL is `https://finnhub.io/api/v1/stock/`, input path is `profile` and query parameter is `symbol=<symbol of the company>&token=<API authentication key>`

Prerequisite:

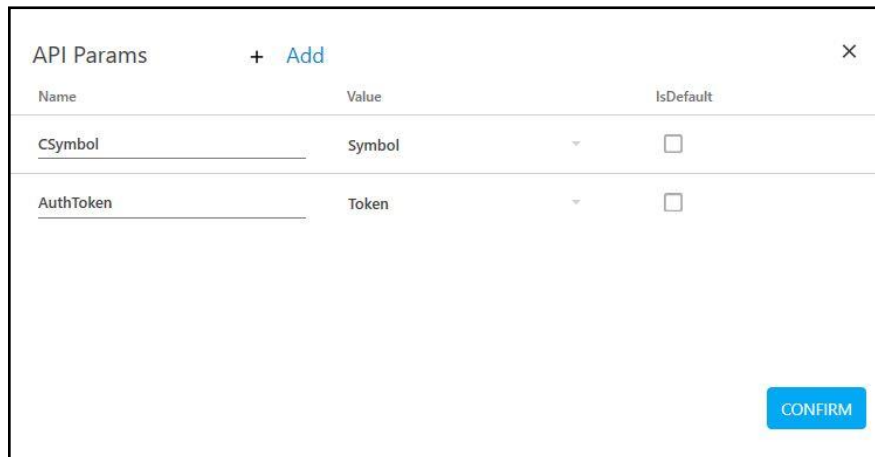
1. In the **Admin** menu, add a **WebAPI** application.
2. In the Application Properties panel:
 - Enter the Login URL as `https://finnhub.io/api/v1/stock/`
 - Select the **Type** as **REST**.
 - Enter the Application Description, Display Name, and the Application Name.
3. Click the  (**Save Properties**) icon to save the application details

The REST WebAPI application is created.

To automate the process of retrieving company details:

1. Create a new process.
2. In the Parameter bar
 - Create **In** arguments and define their respective values:
 - **Profile**, to store the input path-profile
 - **Symbol**, to store and pass the company name- AAPL as the input
 - **Token**, to store and pass the authentication token as the input.
 - Create an **Out** argument, `CompDetails` to store the API output.
3. From the **Canvas Tools** panel, add Rest API Activity to the Flowchart designer on the Canvas
4. In the **Properties** pane of the **REST API Activity** (available on the right hand side), set **Methods** as **Get**.
5. In the **Select Rest API** list, select the available Rest API.
6. In the **Api Path** list, select the **Profile** argument created above.


7. Click the  (**Settings**) icon, and then click **UriParams**.



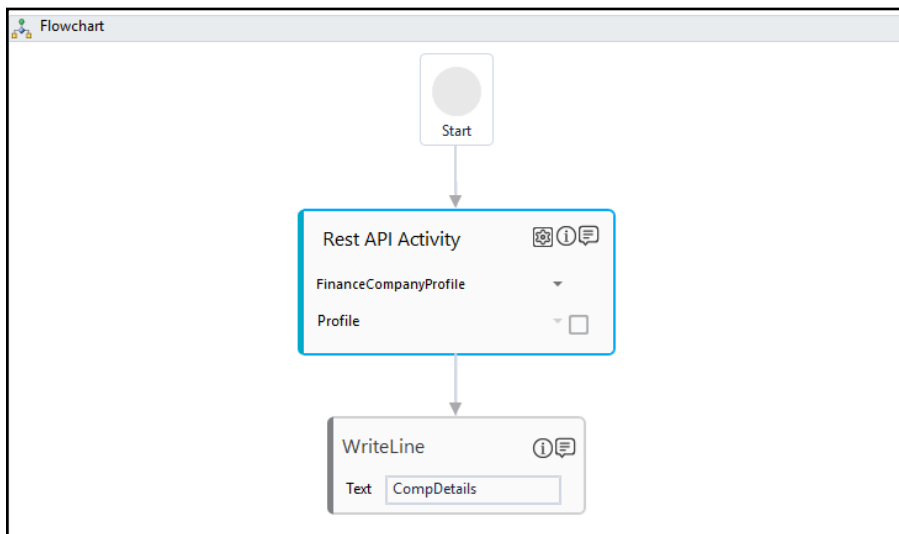
Name	Value	IsDefault
CSymbol	Symbol	<input type="checkbox"/>
AuthToken	Token	<input type="checkbox"/>

CONFIRM

The API Params dialog box appears.

8. Click **Add** and then enter a name of the company symbol parameter.
9. In the **Value** list, select **Symbol**. Argument created above.
10. Click **Add** and then enter a name of the authentication token.
11. In the **Value** list, select **Token**.
12. Click **Confirm**.
13. In the **Parameter** bar, provide a value to the **Symbol** and **Token** arguments respectively. This sets the values as default value of the query parameter.
14. Click the  (**Settings**) icon, and then click **Input Mapping**.
15. In the **Output** list, select the **CompDetails** argument to store the details of the company retrieved by the API. You can use **CompDetails** to pass the data to another activity for further processing.
16. To view the output of this example, add the Writeline activity that prints the information on the console.

17. Add **WriteLine** activity and in the **Text** field, write **CompDetails**.



18. Setup the environment and then perform test run.
19. The **Output** console displays the XML response received from the API. You can use the XML activity to view the details as per your requirement.

9.6.14 JSON Activity

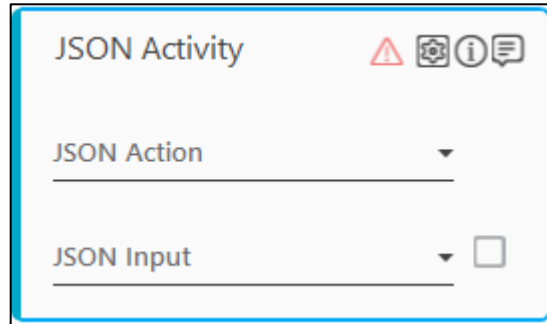
This activity allows you to modify, form request, or retrieve data of the JSON response received from a REST API or other activities in the Automation Studio.

Additionally, you can parse the existing JSON string by manually passing the JSON string using the inbuilt JSON editor.

It helps you to extract a relevant part of the response as per your requirement. You can even update the JSON object and the JSON array by using this activity. Refer <https://www.json.org/json-en.html> to understand the concept of JSON format in detail.

Using JSON Activity

1. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
2. Drag the **JSON Activity** and drop to the Flowchart designer on the Canvas



3. The instructions related to JSON activity differs depending upon the supported JSON request method selected in the JSON Action list. Click the appropriate link to know the details:
 - [Setter](#)- Modifies or adds a new property into a JSON Object as per the location provided.
 - [Getter](#)- Retrieves data from the JSON data file.
 - [Push](#)- Adds new object into the JSON array.


Note: You must Close Environment and then Reset Environment every time you want to Test Run a process workflow, else, error is received or the test run fails.

Setter

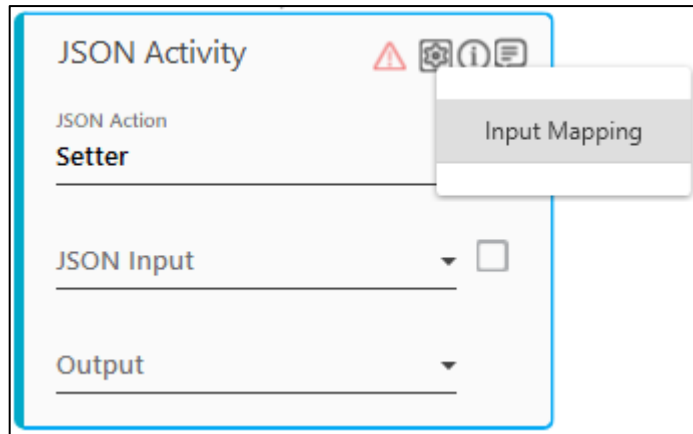
You can pass the JSON input for processing the request in two different ways. Click any of the following link to know the details:

- [Pass data as JSON file](#) - the data is passed as a JSON file for the update to be done.
- [Pass data Manually](#) - the data is passed manually in the editor available in Automation Studio for the update to be done.

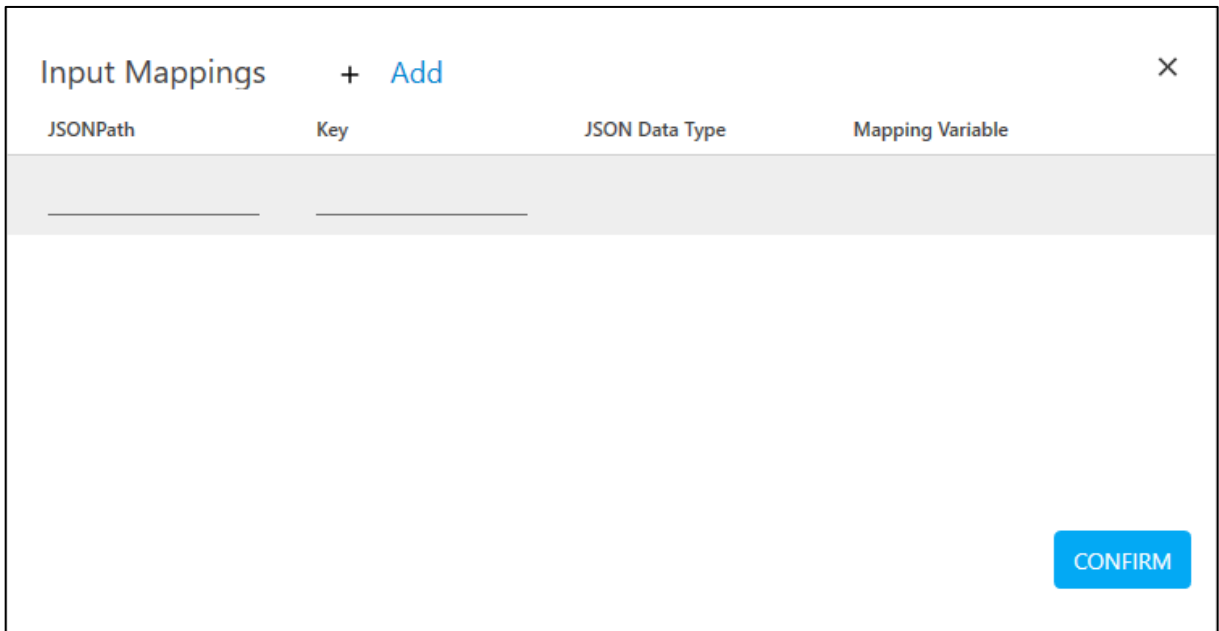
Passing Data as JSON File

1. In the **JSON Action** list, select **Setter**.
2. In the **JSON Input** list, select the available parameter holding the JSON response received from the REST API activity or other activity for processing.
3. Click the  (**Settings**) icon, and then click **Input Mapping**. The **Input Mapping** dialog box appears where you can align the

key-value pair with a parameter to set in the JSON input.




4. Click **Add**. The fields and row appear where you can enter the relevant data.

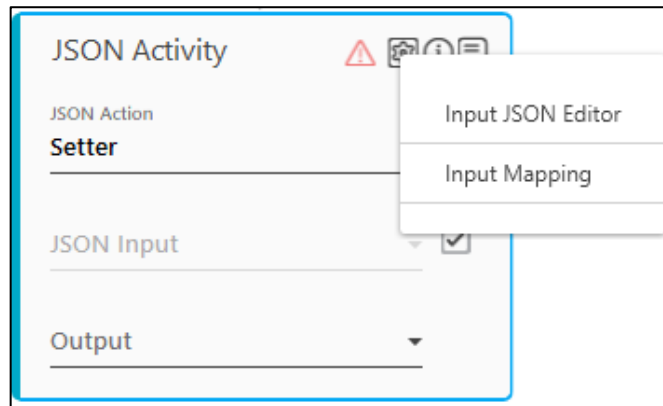


5. In the **JSONPath** field, enter the JSON path expression where you want to set the key-value pair. Refer <https://goessner.net/articles/JsonPath/> to know the JSON path expression in detail.
6. In the **Key** field, enter the key that you want to insert in the JSON data file.
7. In the **JSON Data Type** list, select the data type of the value. Available options are:
 - **Number** - a double- precision floating-point format of an integer, fraction and exponent type such as digits 1-9, 0, positive integer, negative integer, .3, .6, e, e+ and so on.

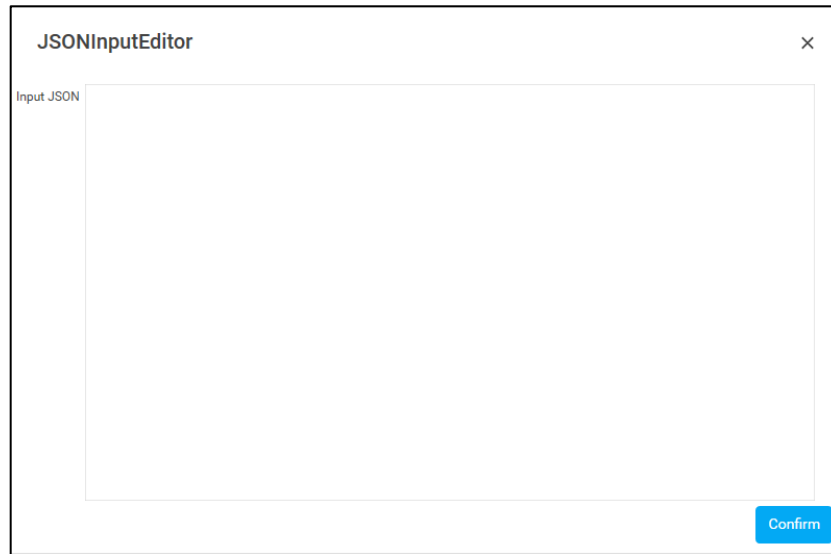
- **String** - a sequence of characters or a string literal.
 - **Boolean** - a datatype used for making decision. Can have only two values- true or false.
 - **JSONArray** - an ordered sequence of values.
 - **JSONObject** - an ordered collection of key-value pair.
 - **Null** - an empty value.
8. In the **Mapping Variable** list, select the parameter holding the value you want to align with the key. You must define the parameter in the **Parameter** bar to use this option.
 9. Click **CONFIRM** to save the input mapping configuration. You are directed back to the JSON activity in the Canvas.
 10. In the **Output** list, select the available parameter to store the output. You must define the parameter in the **Argument** pane of **Parameter** bar to use this option.


Passing Data Manually

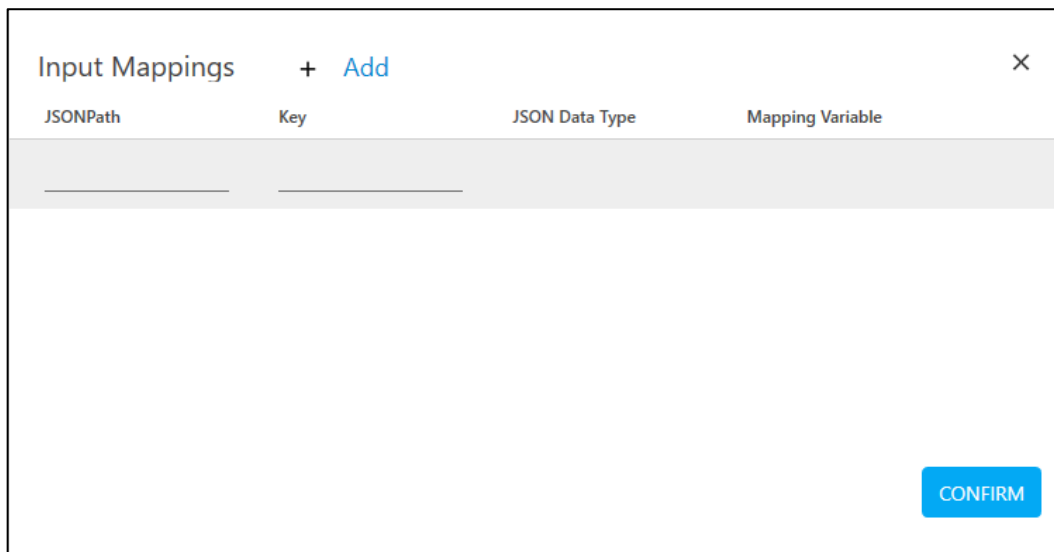
1. In the **JSON** Action list, select **Setter**.
2. Select the **Manual Input Entry** check box to provide the JSON data manually.
3. Click the  (**Settings**) icon, and then click **Input JSON Editor**. The **JSONInputEditor** dialog box appears where you can enter the JSON data manually.



4. Enter the data in JSON format and then click **Confirm**.



5. Click the  (**Settings**) icon, and then click **Input Mapping** to map the key-value pair with an parameter to set in the JSON input. The **Input Mapping** dialog box appears.



6. Click **Add** to provide details related to the mapping of input parameters. This lets you add the key of the key-value pair of the JSON input.
7. In the **JSONPath** field, enter the JSON path expression where you want to set the key-value data. Refer <https://goessner.net/articles/JsonPath/> to know the JSON path expression in detail.
8. In the **Key** field, enter the key that you want to insert in the JSON data.
9. In the **JSON Data Type** list, select the data type of the value. Available options are:
 - **Number** - a double- precision floating-point format of an integer, fraction and exponent type such as digits 1-9, 0, positive integer, negative integer, .3, .6, e, e+ and so on.
 - **String** - a sequence of characters or a string literal.

- **Boolean** - a datatype used for making decision. Can have only two values- true or false.
 - **JSONArray** - an ordered sequence of values.
 - **JSONObject** - an ordered collection of key-value pair.
 - **Null** - an empty value.
10. In the **Mapping Variable** list, select the parameter holding the value you want to map with the key.
 11. Click **CONFIRM** to save the input mapping configuration. You are directed back to the JSON activity in the Canvas.
 12. In the **Output** list, select the available parameter to store the output. You must define the parameter in the **Parameter** bar to use this option.

The key-value data is set in the JSON input.

Getter

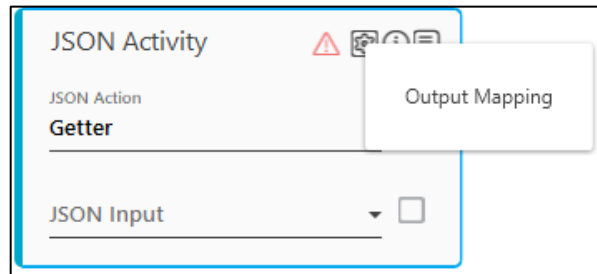
You can pass the JSON input for processing the request in two different ways. Click any of the link to know the details:

- [Passing data as JSON file](#) - the data is passed as a JSON file is passed for the update to be done.
- [Passing data Manually](#) - the data is passed manually in the editor available in Automation Studio for the update to be done.

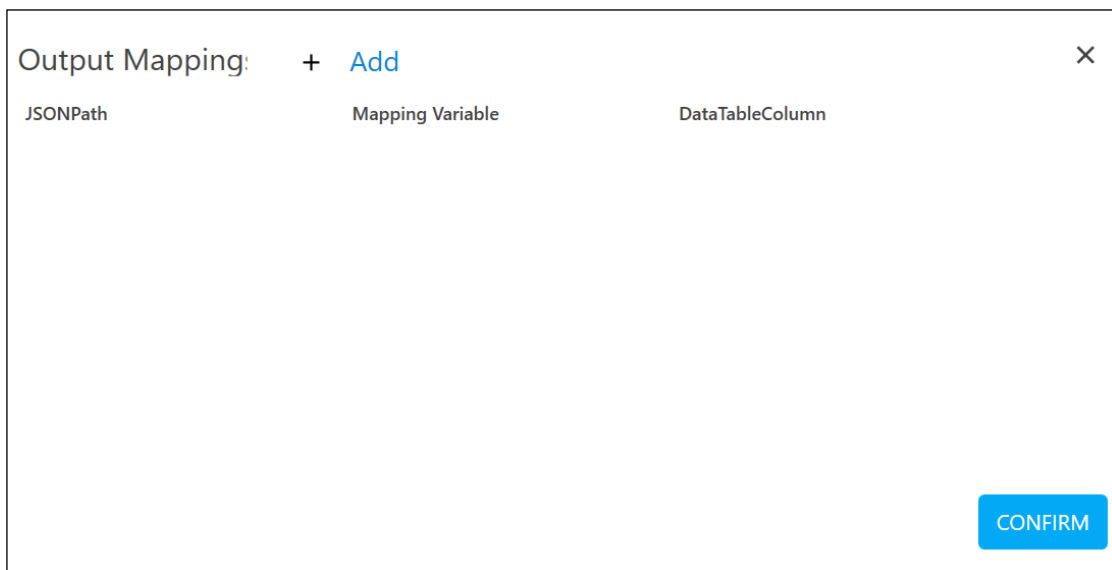
Passing Data as JSON File

1. In the **JSON Action** list, select **Getter**.
2. In the **JSON Input** list, select the available parameter to pass the JSON input file. The file gets stored in the selected parameter. You must define the parameter in the Parameter bar to use this option.

- Click the  (**Settings**) icon, and then click **Output Mapping** to assign the fetched data to an parameter for further processing. The **Output Mapping** dialog box appears.



- Click **Add** to provide details related to the mapping of output parameters. This lets you provide the path to retrieve the data.
- In the **JSONPath** field, enter the JSON path expression from where you want to retrieve the data. Refer <https://goessner.net/articles/JsonPath/> to know the JSON path expression in detail.




- In the **Mapping Variable** list, select the parameter to map with the retrieved data. You must define the parameter in the Parameter bar to use this option.
- In the **DataTableColumn** list, select the required column. You must define the parameter in the Parameter bar to use this option.

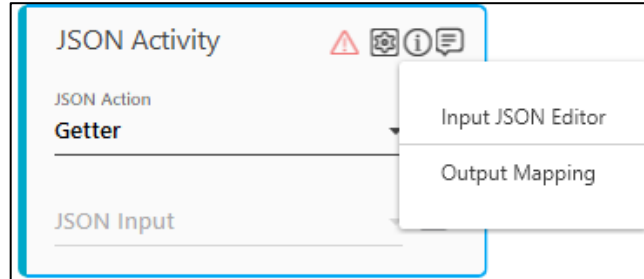
Note: The **DataTableColumn** list is populated, only if, the value selected under **Mapping Variable** is an argument of type **Datatable**.

- Click **CONFIRM** to save the output mapping configuration.

Passing Data Manually


- In the **JSON Action** list, select **Getter**.

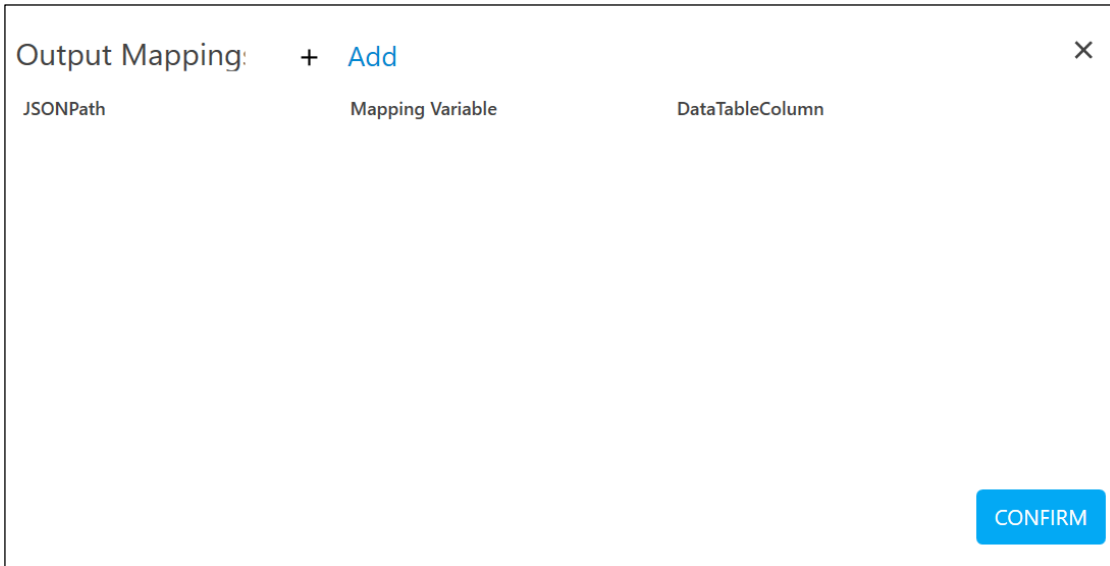
2. Select the **Manual Input Entry** check box to provide the input manually.
3. Click the  (**Settings**) icon, and then click **Input JSON Editor** to enter the JSON data manually. The **JSONInputEditor** dialog box appears.



4. Enter the input data in JSON format and then click **Confirm**. You are directed back to the JSON activity in the **Canvas**.



- Click the  (**Settings**) icon, and then click **Output Mapping** to map the JSON path and the data fetched with a parameter. The **Output Mapping** dialog box appears.



- Click **Add** to provide details related to the mapping of the output parameters.
- In the **JSONPath** field, enter the JSON path expression from where you want to retrieve data. Refer <https://goessner.net/articles/JsonPath/> to know the JSON path expression in detail.
- In the **Mapping Variable** list, select the parameter to map with data retrieved.
- In the **DataTableColumn** list, select the required column. You must define the parameter in the Parameter bar to use this option.

Note: The DataTableColumn list is populated, only if, the value selected under Mapping Variable is an argument of type Datatable.

- Click **CONFIRM** to save the output mapping configuration.

The data is retrieved from the provided JSON input.


Note: Only single value can be fetched using Getter action. If multiple tokens are returned, system displays an error.

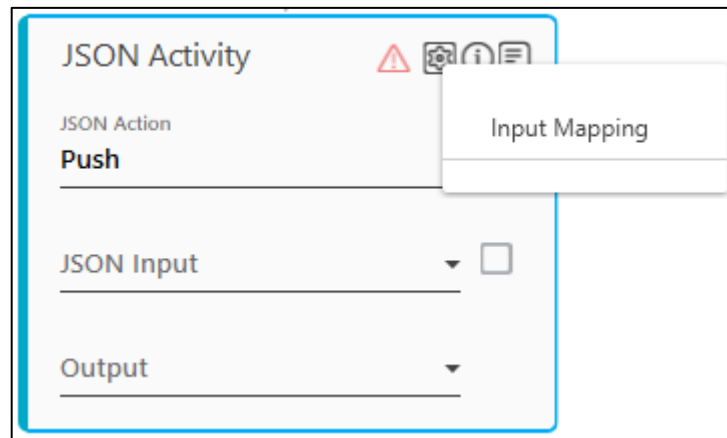
Push

You can pass the JSON input for processing the request in two different ways. Click any of the link to know the details:

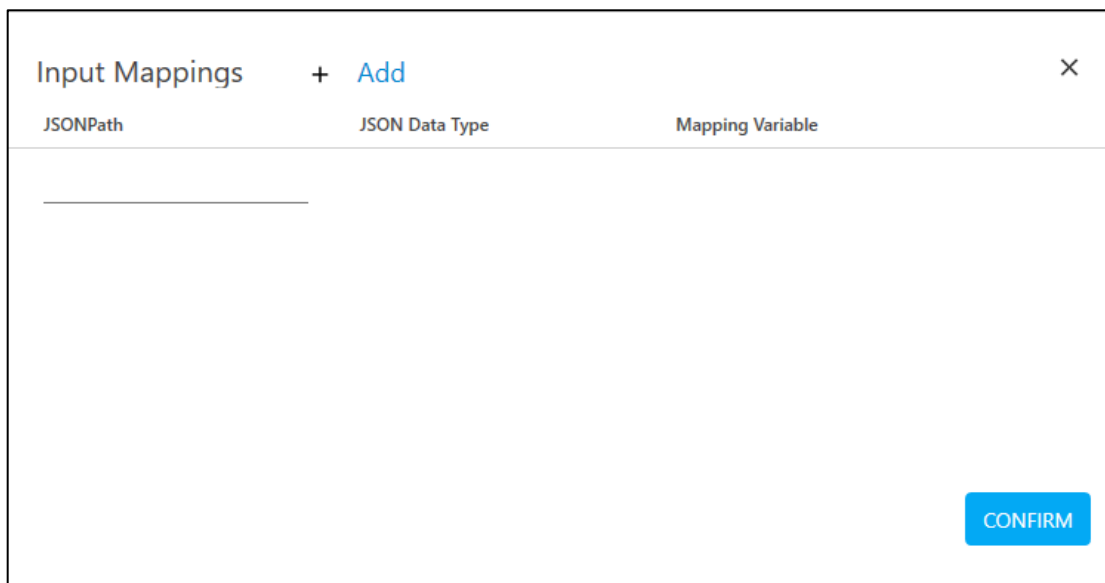
- [Passing data as JSON file](#) - the data is passed as a JSON file is passed for the update to be done.
- [Passing data manually](#) - the data is passed manually in the editor available in Automation Studio for the update to be done.

Passing Data as JSON File

1. In the **JSON Action** list, select **Push**.
2. In the **JSON Input** list, select the available parameter to pass the JSON input file. The file gets stored in the selected parameter. You must define the parameter in the **Parameter** bar to use this option.
3. Click the  (**Settings**) icon, and then click **Input Mapping**. The **Input Mapping** dialog box appears where you can map the value you want to add at the end of the JSON array.




4. Click **Add**. The fields row appears where you can to enter the relevant data.

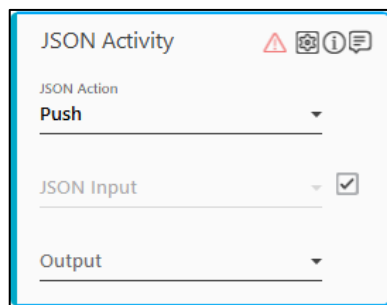


5. In the **JSONPath** field, enter the JSON path expression (that is path of the JSON array) where you want to set the value. Refer <https://goessner.net/articles/JsonPath/> to know the JSON path expression in detail.


- In the **JSON Data Type** list, select the data type of the value. Available options are:
 - Number** - a double-precision floating-point format of an integer, fraction and exponent type such as digits 1-9, 0, positive integer, negative integer, .3, .6, e, e+ and so on.
 - String** - a sequence of characters or a string literal.
 - Boolean** - a datatype used for making decision. Can have only two values- true or false.
 - JSONArray** - an ordered sequence of values.
 - JSONObject** - an ordered collection of key-value pair.
 - Null** - an empty value.
- In the **Mapping Variable** list, select the parameter to map with value you want to add. You must define the parameter in the **Parameter** bar use this option.
- Click **CONFIRM** to save the input mapping configuration. You are directed back to the JSON activity in the **Canvas**.
- In the **Output** list, select the available parameter to store the output. You must define the parameter in the **Parameter** bar to use this option.

Passing Data Manually

- In the **JSON Action** list, select **Push**.
- Select the **Manual Input Entry** check box to provide the JSON input as default value.
- Click the  (**Settings**) icon, and then click **Input JSON Editor** to enter the JSON input manually. The **JSONInputEditor** dialog box appears.



4. Enter the data in JSON format and then click **Confirm**. You are directed back to the JSON activity in the **Canvas**.

5. Click the  (**Settings**) icon, and then click **Input Mapping** to map the value you want to add at the end of the JSON array. The **Input Mapping** dialog box appears.

JSONPath	JSON Data Type	Mapping Variable

6. Click **Add** to provide details related to the mapping of input parameters.
7. the **JSONPath** field, enter the JSON path expression (that is location of the JSON array) where you want to add the value. Refer <https://goessner.net/articles/JsonPath/> to know the JSON path expression in detail.
8. In the **JSON Data Type** list, select the data type of the value. Available options are:

- **Number** - a double- precision floating-point format of an integer, fraction and exponent type such as digits 1-9, 0, positive integer, negative integer, .3, .6, e, e+ and so on.
 - **String** - a sequence of characters or a string literal.
 - **Boolean** - a datatype used for making decision. Can have only two values- true or false.
 - **JSONArray** - an ordered sequence of values.
 - **JSONObject** - an ordered collection of key-value pair.
 - **Null** - an empty value.
9. In the **Mapping Variable** list, select the parameter whose value you want to map. You must define the parameter in the **Parameter** bar use this option.
 10. Click **CONFIRM** to save the input mapping configurations.
 11. In the **Output** list, select the available parameter to store the output. You must define the parameter in the **Parameter** bar to use this option.

The data is added at the end of the JSON array.

JSON Activity Properties

The properties of a JSON Activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity.</p> <p>If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as JSON Activity . You can change the name as required.

Example of JSON Activity

Example 1: To retrieve the data from the JSON input:

Let's consider an example of passing the JSON input manually and retrieving key-value pair from a specific path.

We are passing a JSON input that contains a store having details of books and bicycles. We would retrieve details of the book available at the third book array of the JSON and its author name.


1. Create a new process.
2. From the Canvas Tools panel, add the JSON Activity to the Flowchart designer on the Canvas
3. In the **Parameter** bar, create **In** arguments, **outputstring** and **authorname** of type **String**. These arguments are used to store the book details and name of the author respectively.

The screenshot shows the Automation Studio interface. At the top, a flowchart titled "Flowchart" contains a "Start" node connected to a "JSON Activity" node. The "JSON Activity" node is highlighted with a blue border. Below the flowchart, the "Arguments" tab is active, displaying a table with the following configuration:


Name	Direction	Argument type	Default value
outputstring	In	String	Enter a VB expression
authorname	In	String	Enter a VB expression
Create Argument			

At the bottom of the interface, there are tabs for "Variables" and "Arguments", a search icon, a zoom level of "100%", and a refresh icon.

4. In the **JSON Action** list, select **Getter** to perform the action of retrieving data.
5. Select the **Manual Input Entry** check box to provide the input manually.

- Click the  (**Settings**) icon, and then click **Input JSON Editor** to enter the JSON data manually. The **JSONInputEditor** dialog box appears.



- Enter the details of the books and the bicycles in JSON format and then click **Confirm**. You are directed back to the JSON activity in the **Canvas**.
- Click the  (**Settings**) icon, and then click **Output Mapping** to align the JSON path and the data fetched with an argument. The **Output Mapping** dialog box appears.
- Click **Add** to provide details related to the mapping of the output parameters.
- In the **JSONPath** field, enter the JSON path expression, **\$.book [2]** to retrieve the details of the third book available in the JSON array.
- In the **Mapping Variable** list, select **outputstring** defined in the **Parameter** bar. The mapped argument stores the data retrieved from the stated JSON path.
- Click **Add**.
- In the **JSONPath** field, enter the JSON path expression, **\$.book[2].author** to retrieve the author name of the retrieved book.

14. In the **Mapping Variable** list, select **authorname** defined in the **Parameter** bar. The mapped argument stores the data retrieved from the stated JSON path.

JSONPath	Mapping Variable
<u>\$..book[2]</u>	outputstring
<u>\$..book[2].author</u>	authorname

CONFIRM

15. Click **CONFIRM** to save the output mapping configuration.

To view the output in Automation Studio, let's add WriteLine activity for both the retrieved set of data. You can assign this process to a robot, if you want to execute this process outside Automation Studio.

16. Add a **WriteLine** activity below the **JSON Activity** and in the **Text** field, enter **outputstring** to print the book details retrieved.
17. Add another **WriteLine** activity below the first **WriteLine activity** and in the **Text** field, enter **authorname** to print the name of the author of the retrieved book.
18. Save the process.
19. Setup the environment and then perform test run.

```

Output
{
  "category": "fiction",
  "author": "Herman Melville",
  "title": "Moby Dick",
  "isbn": "0-553-21311-3",
  "price": 8.99
}
Herman Melville


```

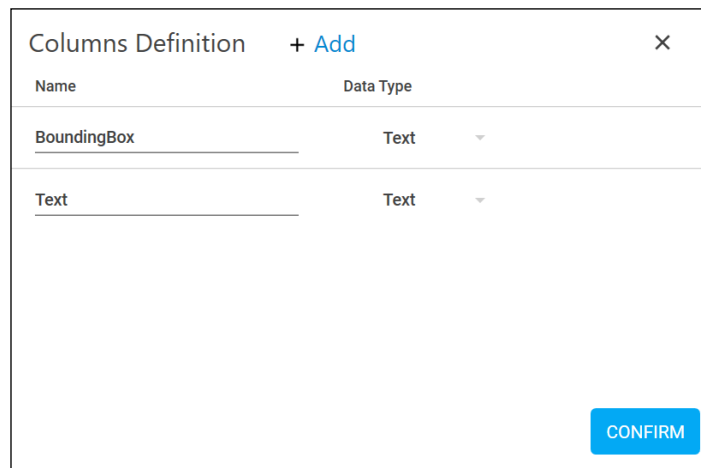
The console displays the book details and the name of the author.

Example 2: To retrieve the data from the JSON input in a tabular format

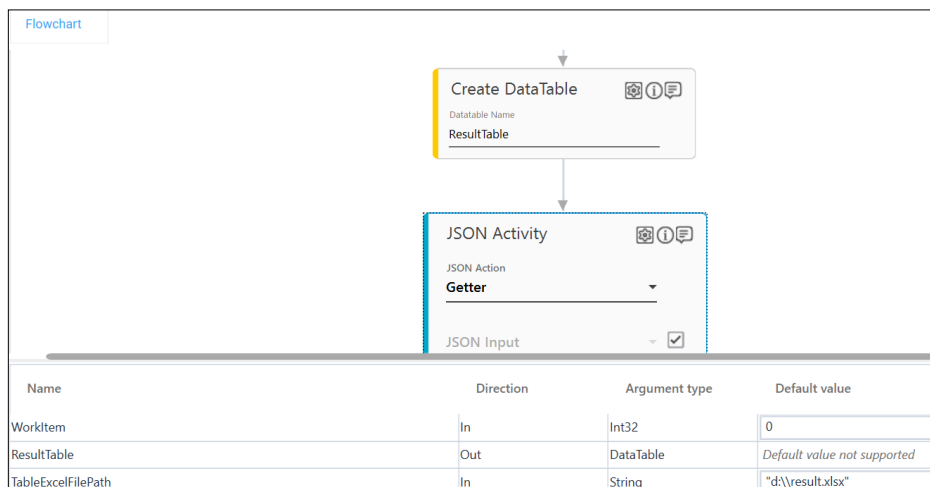
Let's consider an example of passing the JSON input manually and retrieving nested objects from a specific path.

We are passing a JSON as an input that contains details of regions, in form of object array. We would retrieve details of the bounding box and text in a tabular format.


1. Create a new process.
2. From the Canvas Tools panel, add the Create DataTable activity to the Flowchart designer on the Canvas
3. Create a data table named **ResultTable**. An **Out** argument of type **DataTable** gets automatically created for **ResultTable**.
4. Click the  (**Settings**) icon and add two columns **BoundingBox** and **Text**.

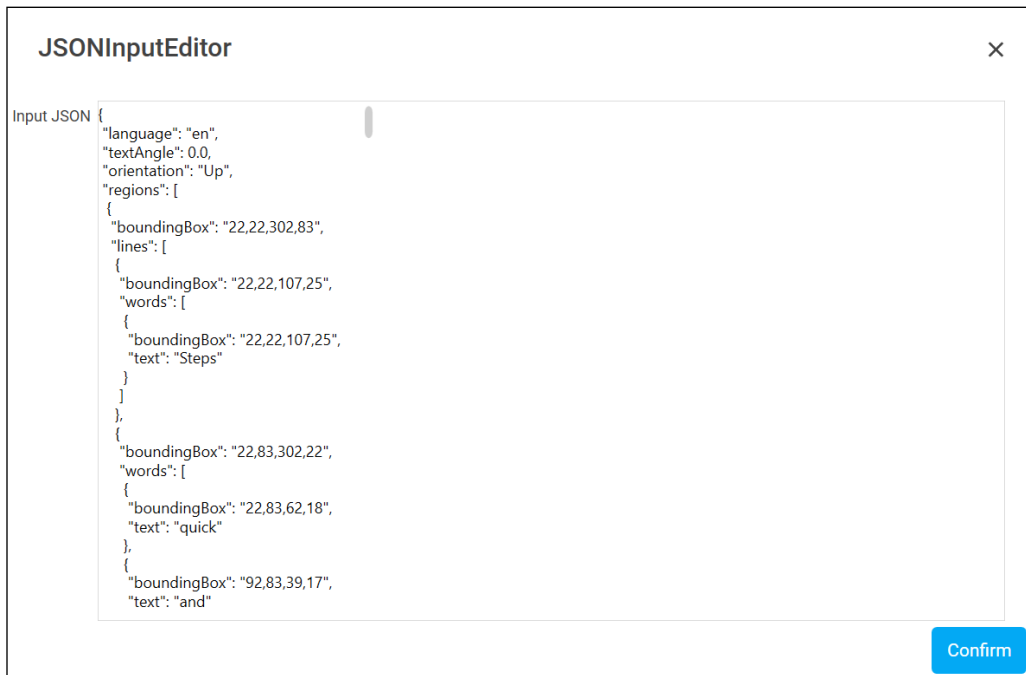



5. From the **Canvas Tools** panel, add the **JSON Activity** to the **Flowchart** designer on the **Canvas** and connect it to the **Create DataTable** activity.

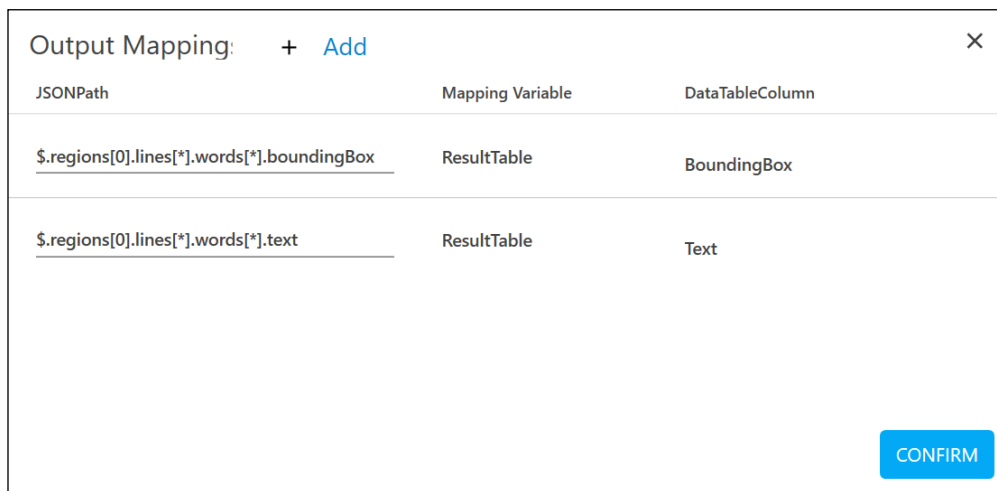


6. From the **JSON Action** list, select **Getter**.

7. Select the **JSON Input** checkbox to get the **Input JSON Editor** option in settings.
8. Click the  (**Settings**) icon and select **Input JSON Editor**.
9. Provide the JSON input in the editor.



10. Click the  (**Settings**) icon, and then click **Output Mapping** to align the JSON path and the data fetched with an argument. The **Output Mapping** dialog box appears.
11. Click **Add** to provide details related to the mapping of the output parameters.



12. In the **JSONPath** field, enter the JSON path expression, **\$.regions[0].lines[*].words[*].boundingBox** to retrieve the details for **BoundingBox**.

Note: Use wild card (*) to retrieve multiple objects from the JSON.

13. In the **Mapping Variable** list, select **ResultTable** defined in the **Parameter** bar.
14. In the **DataTableColumn** list, select **BoundingBox** column. The mapped column stores the data retrieved from stated **JSON** path.

Note: The DataTableColumn list is populated, only if, the value selected under Mapping Variable is an argument of type Datatable.

15. Click **Add**.
16. In the **JSONPath** field, enter the JSON path expression, **\$.regions[0].lines[*].words[*].text** to retrieve the details for **Text**.
17. In the **Mapping Variable** list, select **ResultTable** defined in the **Parameter** bar.
18. In the **DataTableColumn** list, select **Text** column.
19. Click **CONFIRM** to save the output mapping configuration.
20. To verify the datatable is successfully populated, create an **Export DataTable** activity in and enter the required fields to export data table in an **Excel**. You can assign this process to a robot, if you want to execute this process outside Automation Studio.

Export DataTable (i) (m)

Datatable Name
ResultTable

Export File Path
TableExcelFilePath

File Type
Excel

Include Headers?

Overwrite File?

21. Save the process.
22. Setup the environment and then perform test run.

	A	B
1	BoundingBox	Text
2	22,22,107,25	Steps
3	22,83,62,18	quick
4	92,83,39,17	and
5	140,88,50,17	easy
6	197,84,21,16	to
7	227,83,44,17	deal
8	280,83,44,17	with

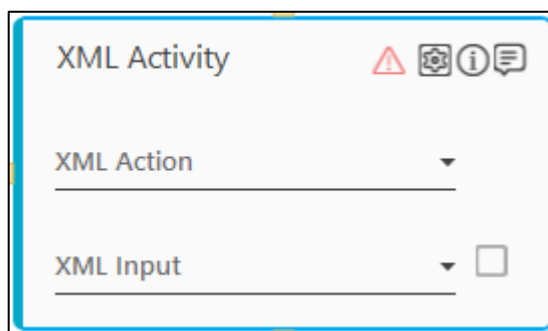
The exported excel displays the retrieved data.

9.6.15 XML Activity

This activity lets you manage and modify the XML response, form a request, or retrieve data from the existing XML file. You can automate the common XML tasks such as merging, transforming, validating and signing XML documents. It helps to read, edit, create or delete the XML nodes as well as extract XML fragments (collection of nodes).

Using XML Activity

1. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
2. Drag the **XML Activity** and drop to the Flowchart designer on the Canvas




3. The instructions related to XML activity differs depending upon the supported request method selected in the **XML Action** list. Click the appropriate link to know the details:
 - [Setter](#)- Modifies or adds data into the XML node
 - [Getter](#)- Retrieves data from the XML input.
 - [Push](#)- Adds XML node or XML fragments into the provided XML input

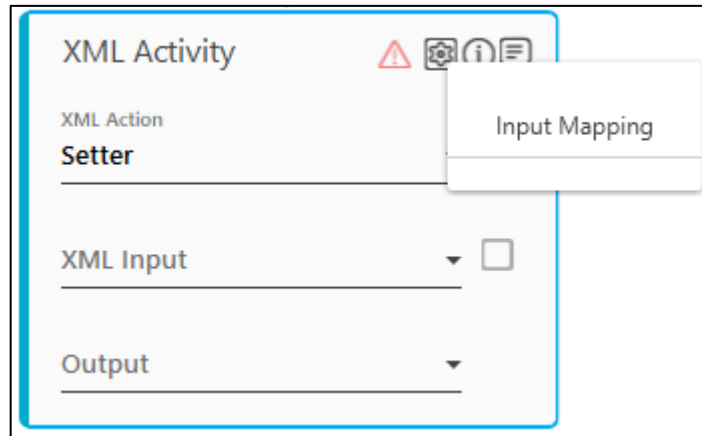
Setter

You can pass the XML input for processing the request in two different ways. Click any of the following link to know the details:

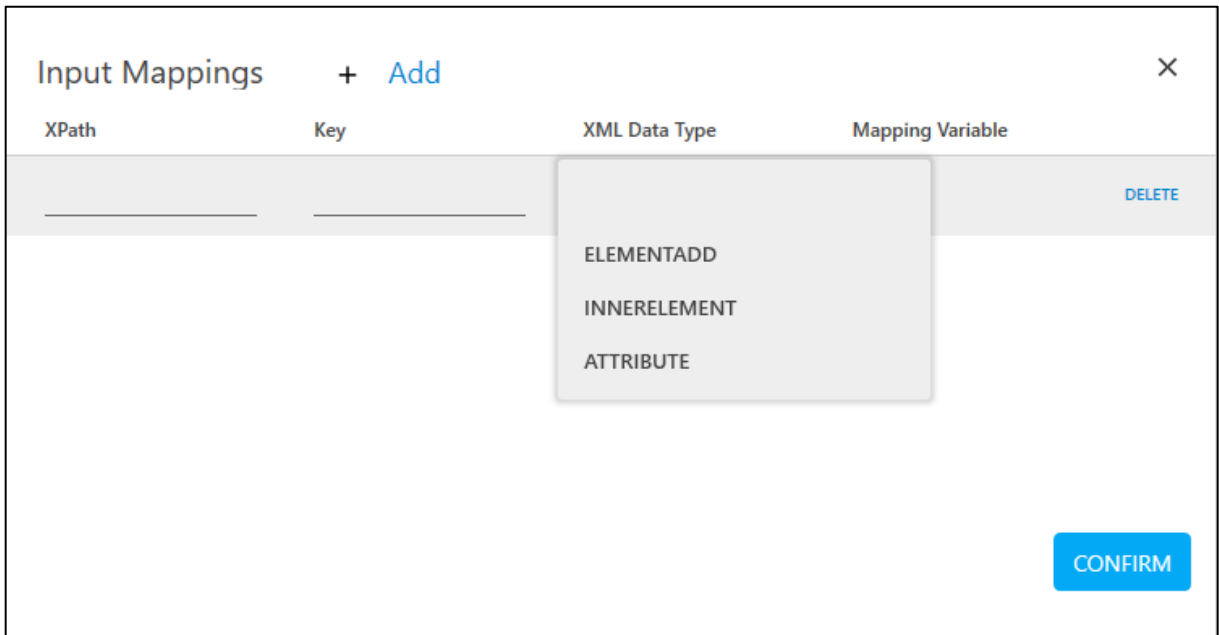
- [Pass data as file](#) - the data is passed as a XML file for the updates to be done.
- [Pass data manually](#) - the data is passed manually in the editor available in Automation Studio for the updates to be done.

Passing Data as XML File

1. In the **XML Action** list, select **Setter**.
2. In the **XML Input** list, select the available parameter to pass the XML input file for processing. The input file gets stored in the selected parameter. You must define the parameter in the Parameter bar to use this option.
3. Click the  (**Settings**) icon, and then click **Input Mapping**. The **Input Mapping** dialog box appears where you can align the key-value pair with a parameter to set in the XML file input.




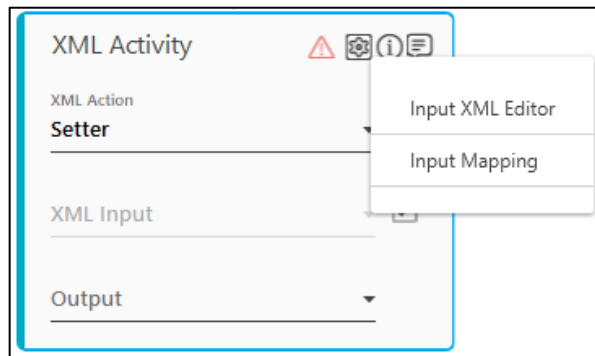
4. Click **Add**. The fields and row appear where you can enter the relevant data.




5. In the **XPath** field, enter the XML path expression for selecting the nodes from the XML data file where you want to set the key:value pair.
6. In the **Key** field, enter the key that you want to insert in the XML data file.
7. In the **XML Data Type** list, select the data type of the value. Available options are:
 - **ELEMENTADD** - creates a new XML tag with the intended value provided for the key at the required xpath.
 - **INNERELEMENT**- adds the value inside the mapping parameter at the given xpath. For this data type, key is not considered.
 - **ATTRIBUTE**- adds the provided key as attribute and its value at the given xpath.
8. In the **Mapping Variable** list, select the parameter whose value you want to map with the key. You must define the parameter in the **Parameter** bar to use this option.
9. Click **CONFIRM** to save the input mapping configuration. You are directed back to the XML activity in the Canvas.
10. In the **Output** list, select the available parameter to store the output of this action. You must define the parameter in the **Parameter** bar to use this option.

Passing Data Manually


1. In the **XML Action** list, select **Setter**.
2. Select the **Manual Input Entry** check box to provide the XML data manually.
3. Click the  (**Settings**) icon, and then click **Input XML Editor**. The **XML Input Editor** dialog box appears where you can enter the XML file data manually.

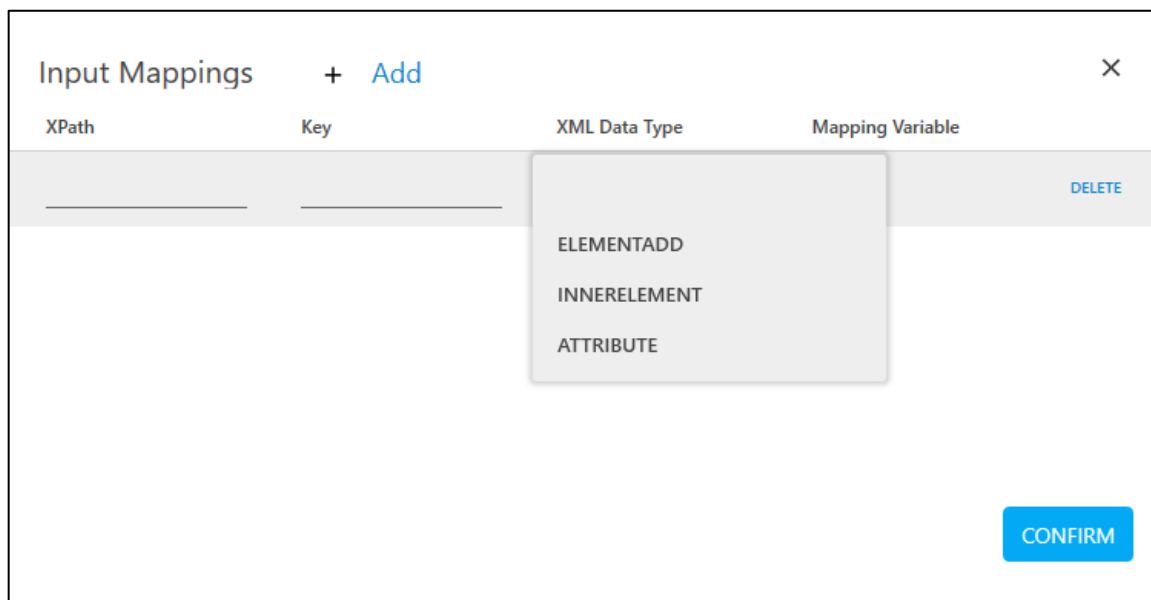


4. Enter the data in XML format and then click **Confirm**.



The XML Input Editor dialog box features a title bar with the text "XML Input Editor" and a close button (X). Below the title bar is a large, empty text area labeled "Input XML". At the bottom right corner of the dialog, there is a blue button labeled "Confirm".

5. Click the  (**Settings**) icon, and then click **Input Mapping** to map the key-value pair with a parameter to set in the XML input. The **Input Mapping** dialog box appears.



The Input Mappings dialog box has a title bar with "Input Mappings" and a "+ Add" button, along with a close button (X). The main area contains a table with the following headers: "XPath", "Key", "XML Data Type", and "Mapping Variable". A dropdown menu is open over the "XML Data Type" column, showing the options "ELEMENTADD", "INNERELEMENT", and "ATTRIBUTE". A "DELETE" button is located on the right side of the table. At the bottom right, there is a blue "CONFIRM" button.

XPath	Key	XML Data Type	Mapping Variable
_____	_____	ELEMENTADD INNERELEMENT ATTRIBUTE	_____

6. Click **Add** to provide details related to the mapping of input parameters. This lets you add the key of the key-value pair of the XML input.

7. In the **XPath** field, enter the XML path expression for selecting the nodes from the XML data file where you want to set the key:value pair.
8. In the **Key** field, enter the key that you want to insert in the XML data.
9. In the **XML Data Type** list, select the data type of the value. Available options are:
 - **ELEMENTADD** - creates a new XML tag with the intended value provided for the key at the required xpath.
 - **INNERELEMENT**- adds the value inside the mapping parameter at the given xpath. For this data type, key is not considered.
 - **ATTRIBUTE**- adds the provided key as attribute and its value at the given xpath.
10. In the **Mapping Variable** list, select the parameter whose value you want to map with the key.
11. Click **CONFIRM** to save the input mapping configuration. You are directed back to the XML activity in the Canvas.
12. In the **Output** list, select the available parameter to store the output of this action. You must define the parameter in the **Parameter** bar to use this option.


The key:value data is set in the XML input.

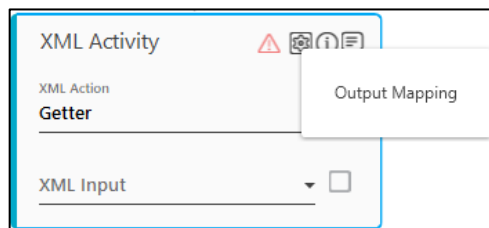
Getter

You can pass the XML input for processing the request in two different ways. Click any of the link to know the details:

- [Passing data as XML file](#) - the data is passed as a XML file is passed for the update to be done.
- [Passing data manually](#) - the data is passed manually in the editor available in Automation Studio for the update to be done.

Passing Data as XML File

1. In the **XML Action** list, select **Getter**.
2. In the **XML Input** list, select the available parameter to pass the XML input file. The file gets stored in the selected parameter. You must define the parameter in the Parameter bar to use this option.
3. Click the  (**Settings**) icon, and then click **Output Mapping** to assign the fetched data to a parameter for further processing. The **Output Mapping** dialog box appears.



4. Click **Add** to provide details related to the mapping of output parameters. This lets you provide the path to retrieve the


data.

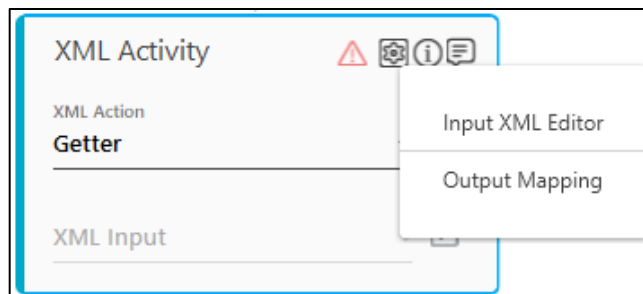
- In the **XPath** field, enter the XML path expression for selecting the nodes from the XML data file from where you want to retrieve the data.



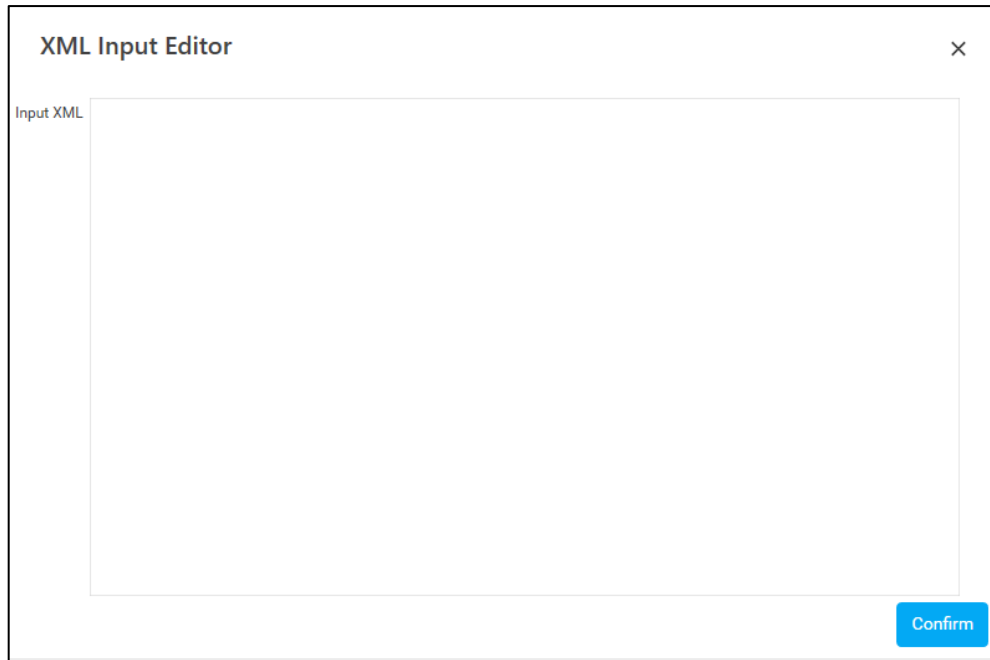
- In the **Mapping Variable** list, select the parameter to map with the retrieved data. You must define the parameter in the Parameter bar to use this option.
- Click **CONFIRM** to save the output mapping configuration.

Passing Data Manually


- In the **XML Action** list, select **Getter**.
- Select the **Manual Input Entry** check box to provide the input manually.
- Click the  (**Settings**) icon, and then click **Input XML Editor** to enter the XML file data manually. The **XML Input Editor** dialog box appears.

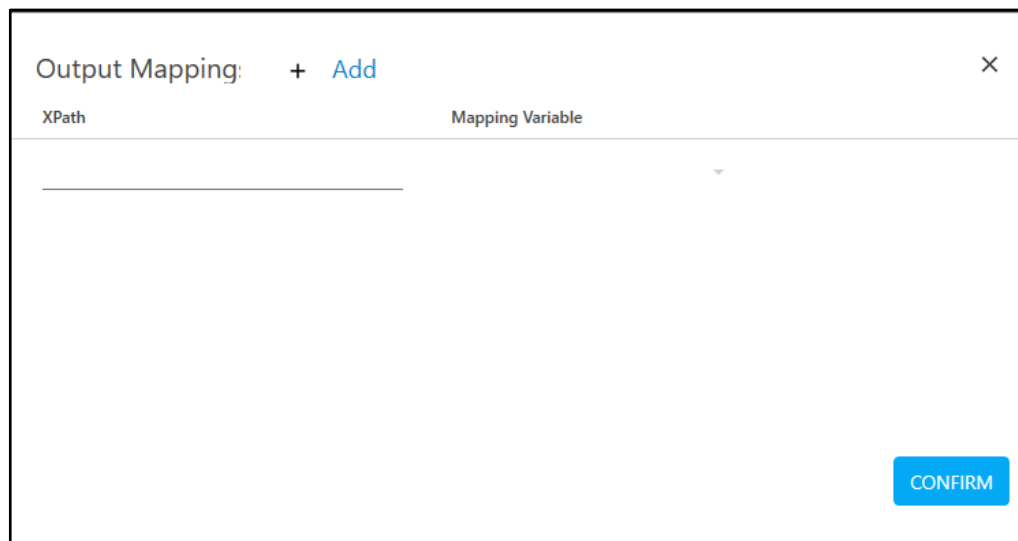


4. Enter the input data in XML format, and then click **Confirm**. You are directed back to the XML activity in the Canvas.



The screenshot shows a dialog box titled "XML Input Editor" with a close button (X) in the top right corner. Inside the dialog, there is a large text area labeled "Input XML" which is currently empty. At the bottom right of the dialog, there is a blue button labeled "Confirm".

5. Click the  (**Settings**) icon, and then click **Output Mapping** to map the XML path and the data fetched with a parameter. The **Output Mapping** dialog box appears.



The screenshot shows a dialog box titled "Output Mapping:" with a close button (X) in the top right corner. Below the title, there is a blue "+ Add" button. The dialog contains a table with two columns: "XPath" and "Mapping Variable". The "XPath" column has a text input field with a horizontal line. The "Mapping Variable" column has a dropdown menu with a downward arrow. At the bottom right of the dialog, there is a blue button labeled "CONFIRM".

6. Click **Add** to provide details related to the mapping of the output parameters.
7. In the XPath field, enter the XML path expression for selecting the nodes from the XML data file from where you want to retrieve data.

8. In the Mapping Variable list, select the parameter to map with data retrieved. You must define the parameter in the Parameter bar to use this option.
9. Click **CONFIRM** to save the output mapping configuration.


The data is retrieved from the provided XML input.

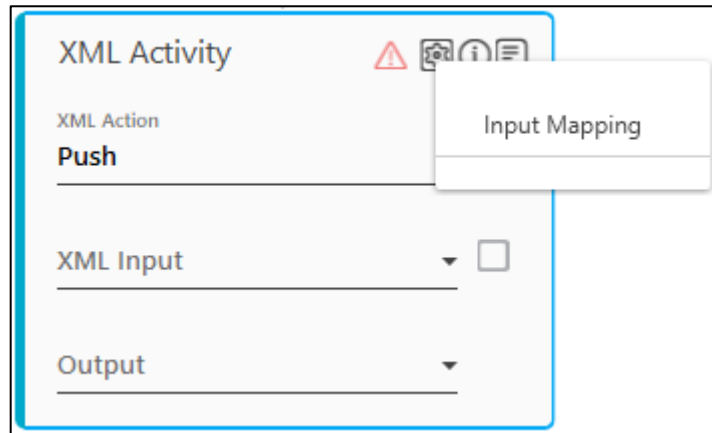
Push

You can pass the XML input for processing the request in two different ways. Click any of the link to know the details:

- [Passing data as XML file](#) - the data is passed as a XML file is passed for the update to be done.
- [Passing data manually](#) - the data is passed manually in the editor available in Automation Studio for the update to be done.

Passing Data as XML File


1. In the **XML Action** list, select **Push**.
2. In the **XML Input** list, select the available parameter to pass the XML input file. The file gets stored in the selected parameter. You must define the parameter in the **Parameter** bar to use this option.
3. Click the  (**Settings**) icon, and then click **Input Mapping**. The **Input Mapping** dialog box appears where you can map the value you want to modify at the stated path.



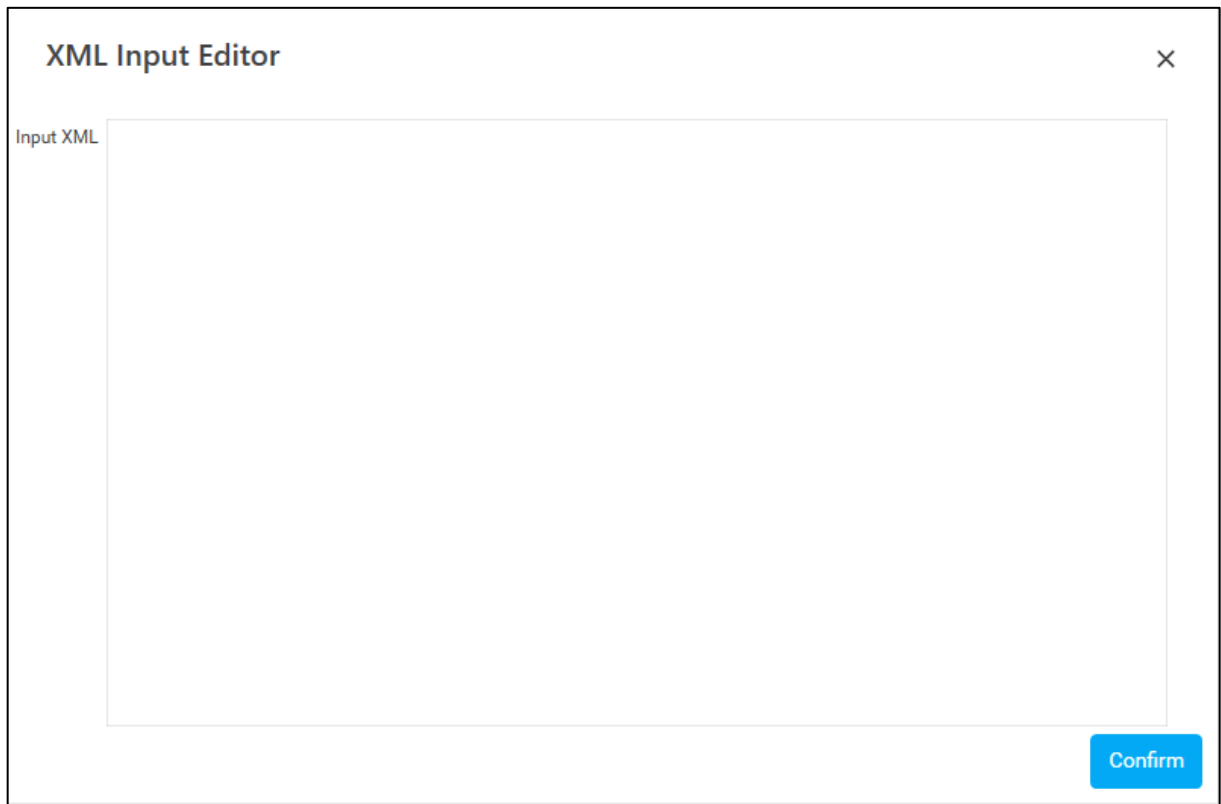
- Click **Add**. The fields row appears where you can to enter the relevant data.


- In the **XPath** field, enter the XML path expression where you want to modify the data.
- In the **Mapping Variable** list, select the parameter to align the value you want to modify at the stated path of the XML input. You must define the parameter in the Parameter bar to use this option.
- Click **CONFIRM** to save the input mapping configuration. You are directed back to the XML activity in the **Canvas**.
- In the **Output** list, select the available parameter to store the output. You must define parameter in the **Parameter** bar to use this option.

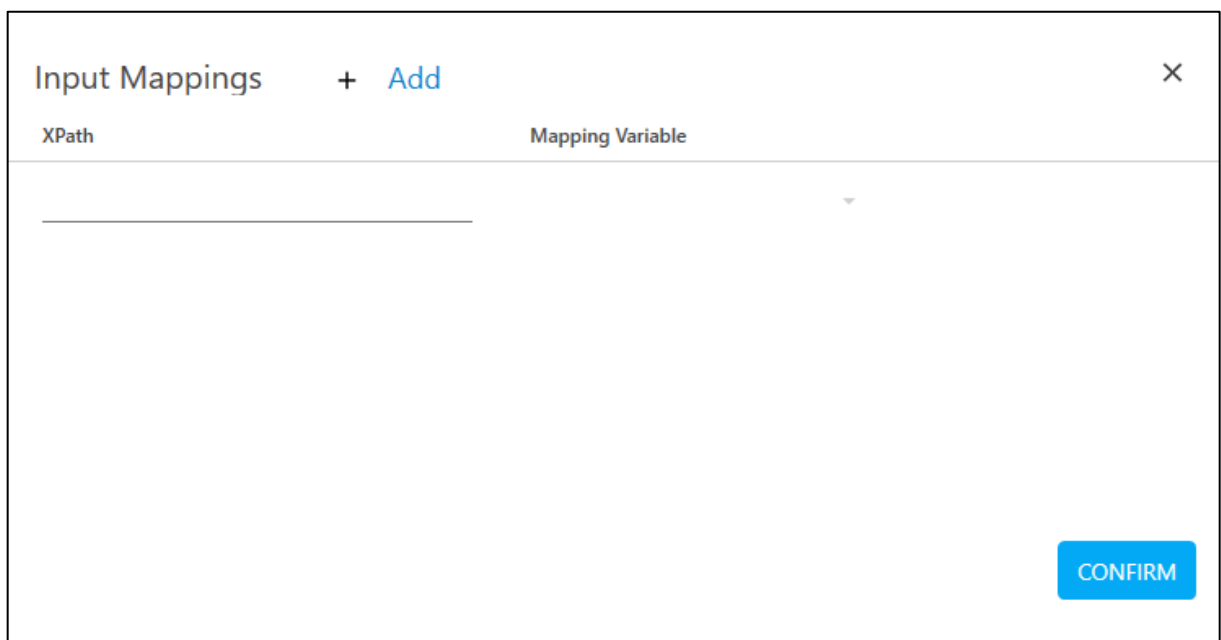
Passing Data Manually

- In the **XML Action** list, select **Push**.
- Select the **Manual Input Entry** check box to provide the XML input as default value.
- Click the  (**Settings**) icon, and then click **Input XML Editor** to enter the XML input manually. The **Input XML Editor** dialog box appears.

4. Enter the data in XML format and then click **Confirm**. You are directed back to the XML activity in the **Canvas**.



5. Click the  (**Settings**) icon, and then click **Input Mapping** to align the value you want to modify in the stated path of XML input. The **Input Mapping** dialog box appears.



6. Click **Add** to provide details related to the mapping of input parameters.
7. In the **XPath** field, enter the XML path expression where you want to modify the value.
8. In the **Mapping Variable** list, select the parameter whose value you want to map. You must define the parameter in the Parameter bar to use this option.
9. Click **CONFIRM** to save the input mapping configurations. You are directed back to the XML activity in the **Canvas**.
10. In the **Output** list, select the available parameter to store the output. You must define the parameter in the **Parameter** bar to use this option.

The data is added at the stated xpath of the XML input.

XML Activity Properties

The properties of a XML Activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
ait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as XML Activity. You can change the name as required.

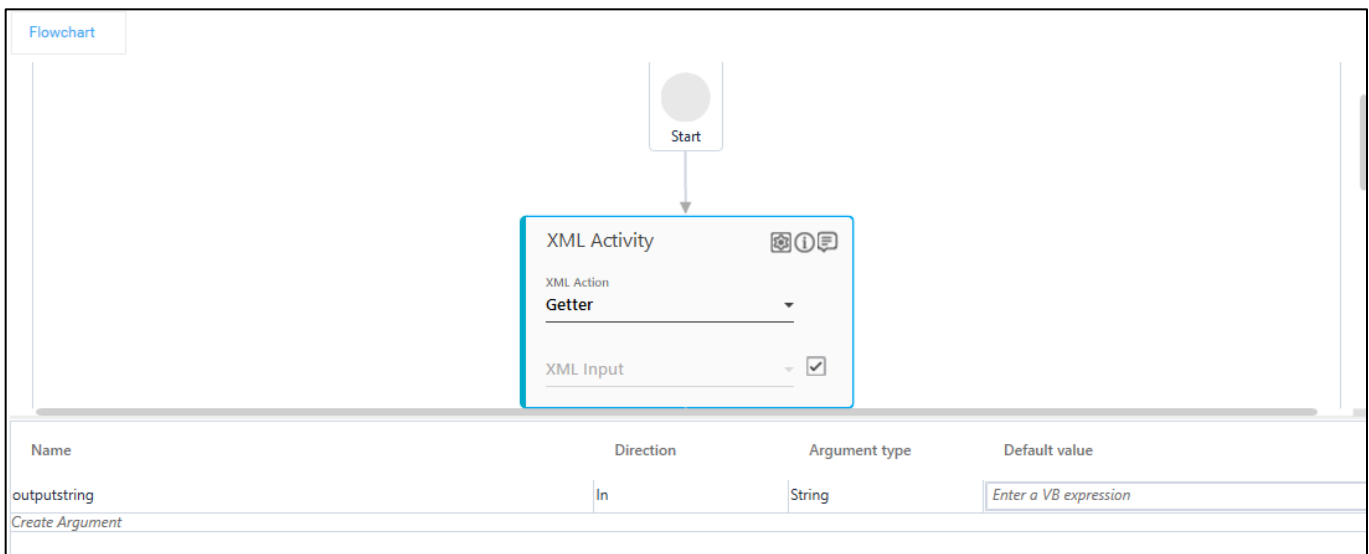
Example of XML Activity

Let's consider an example of passing the XML input manually and retrieving the XML node from the specified path.

We are passing a XML input that contains a store having details of books. We would retrieve details of the book element available at node four in the XML fragment.


To retrieve the data from the XML input:

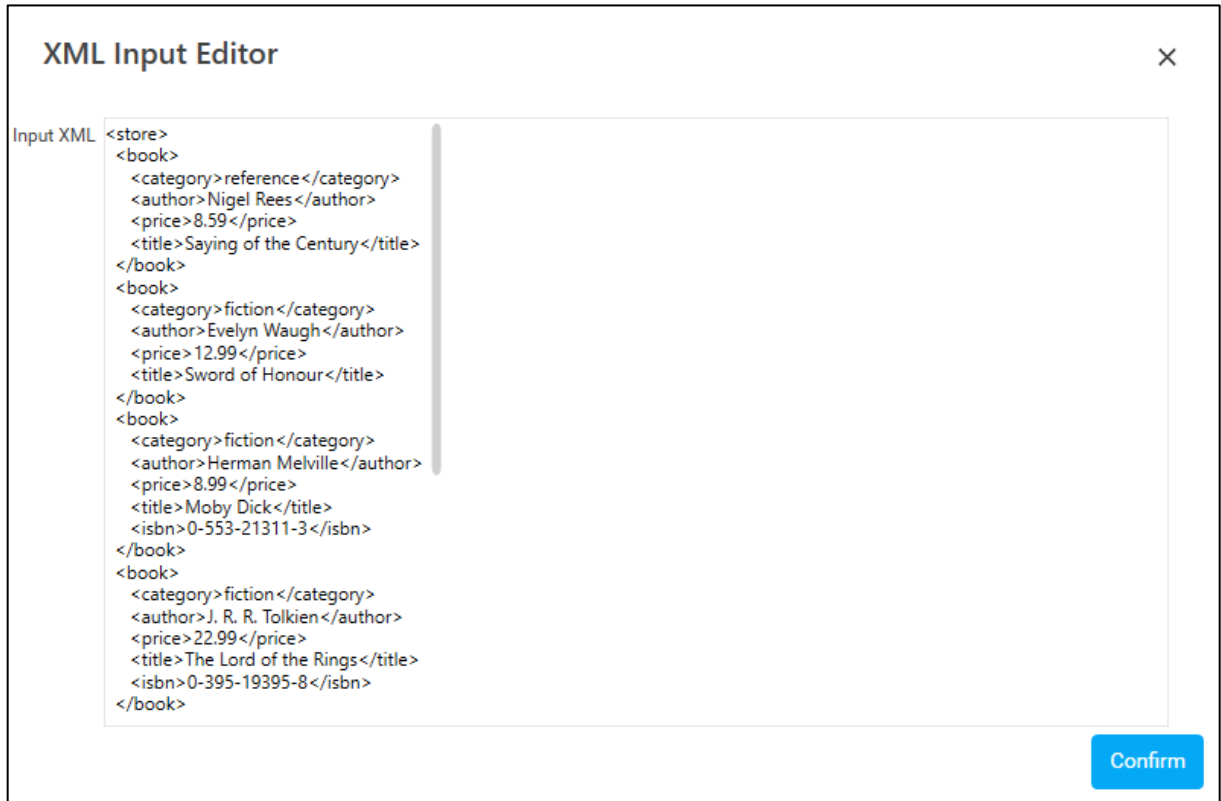
1. Create a new process.
2. From the Canvas Tools panel, add the XML Activity to the Flowchart designer on the Canvas
3. In the **Parameter** bar, create an **In** argument, **outputstring** of type **String**. This argument is used to store the book details.




Name	Direction	Argument type	Default value
outputstring	In	String	Enter a VB expression
<i>Create Argument</i>			

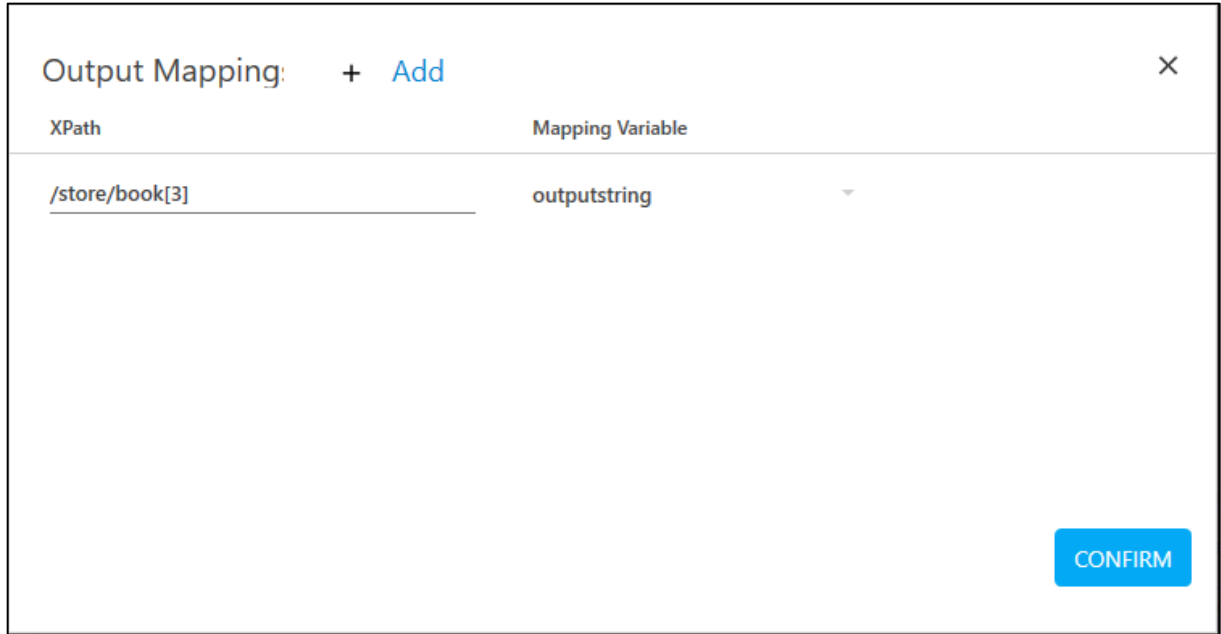
4. In the **XML Action** list, select **Getter** to perform the action of retrieving data.
5. Select the **Manual Input Entry** check box to provide the input manually.

6. Click the  (**Settings**) icon, and then click **Input XML Editor** to enter the XML data manually. The **XML Input Editor** dialog box appears.



7. Enter the details of the books in XML format and then click **Confirm**. You are directed back to the XML activity in the **Canvas**.
8. Click the  (**Settings**) icon, and then click **Output Mapping** to align the XML path and the data fetched with an argument. The **Output Mapping** dialog box appears.
9. Click **Add** to provide details related to the mapping of the output parameters.
10. In the **XPath** field, enter the XML path expression, **/store/book [3]** to retrieve the details of the book element available in the XML fragment.

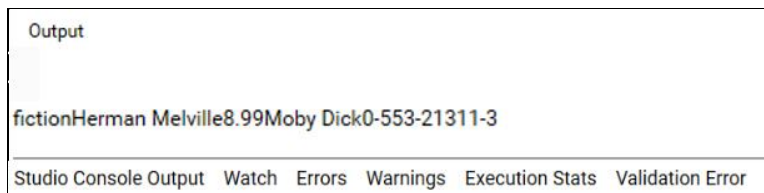
- In the **Mapping Variable** list, select **outputstring** defined in the **Parameter** bar. The mapped argument stores the data retrieved from the stated XML path.



- Click **CONFIRM** to save the output mapping configuration.

To view the output in Automation Studio, let's add WriteLine activity for both the retrieved set of data. You can assign this process to a robot, if you want to execute this process outside Automation Studio.

- Add a **WriteLine** activity below the **XML Activity** and in the **Text** field, enter **outputstring** to print the book details retrieved.
- Save the process.
- Setup the environment and then perform test run.



The console displays the book details.

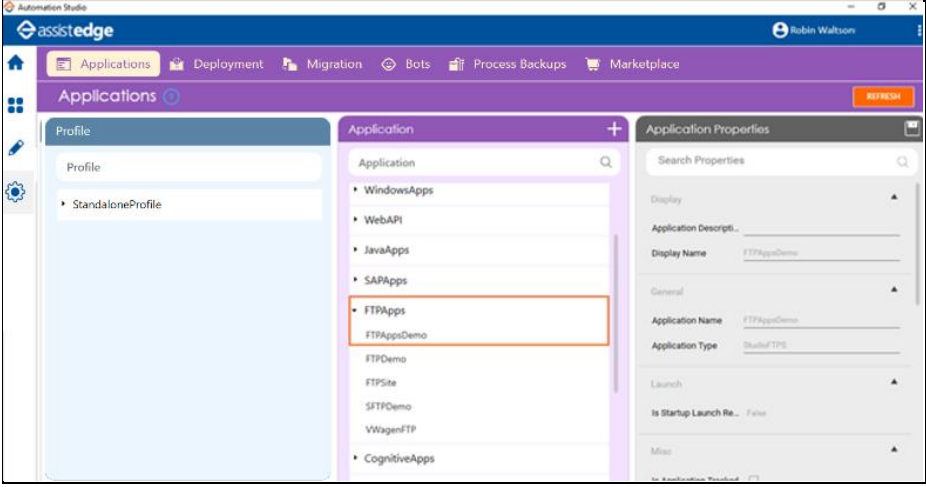

9.6.16 FTP Activity

This activity allows you to automate the actions that can be performed on FTP or SFTP site like upload, download, rename or

delete the required file.

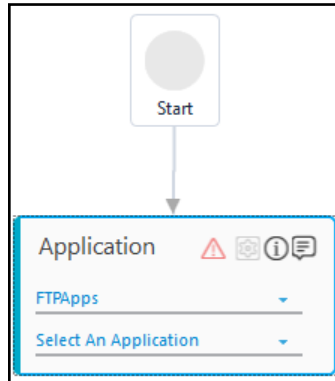
Note: This activity must be used inside an Application activity or the system displays an error. This lets you establish a connection between Automation Studio and the configured FTP or SFTP server.

Prerequisites

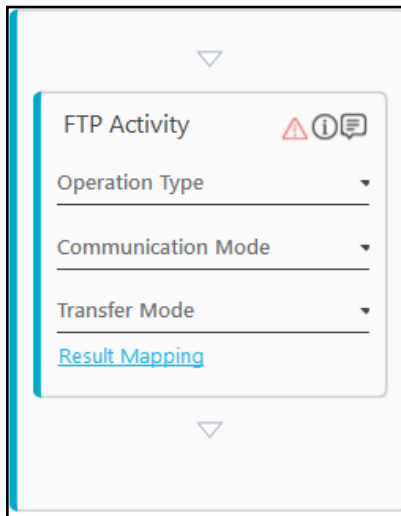
Requirement	Description
FTP or SFTP Server Details	Server details of the FTP or SFTP such as IP address of the server, port number must be available.
Access to FTP or SFTP server	You must have access and login credentials of the FTP/SFTP server where you want to perform the automation.
Configure FTP or SFTP server in Automation Studio	<p>Before you start creating the process, configure the FTP or SFTP server in Automation Studio where you want to perform the automation. Below are the minimum required properties for configuring the intended server. Remaining fields are auto populated with the default values already configured in Automation Studio. If you want to change the default values, refer FTPApps in Admin Capabilities.</p> <p>4. In the Admin menu, add an application of Application Type- FTPApps.</p>  <p>5. In the Application Properties panel:</p> <ul style="list-style-type: none"> In the Display Name field, enter a desired name of the FTP or SFTP server you want to add. In the Server IP field, enter the IP address of the FTP or SFTP sever. In the Server Port field, enter the port number of the FTP or SFTP server. In the Transfer Type list, select whether you want to configure FTP or SFTP server. <p>6. Click the  (Save Properties) icon to save the application details</p> <p>The FTP or SFTP sever is configured.</p>

Using FTP Activity

1. Make sure the [prerequisites](#) for using the FTP Activity are met.
2. Create a new process.
3. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
4. Drag the Application activity and drop on to the Flowchart designer on the Canvas



5. Double click the Application activity, drag the FTP Activity and drop inside the Application activity.



The FTP Activity supports following operations:

- **Push**- uploads the file in the FTP or SFTP site.
- **Pull**- downloads the file from the FTP or SFTP site.
- **List Directory**- lists all the files and folders available at the FTP or SFTP site location in an excel file.
- **Delete**- deletes the files from the FTP or SFTP site.
- **Rename**- renames the files present in the FTP or SFTP site
- **Create Directory** – creates folders in the FTP or SFTP site.

- **Remove Directory** – removes folders from the FTP or SFTP site.
- **File Exists** – verifies whether a file exists in the FTP or SFTP site.
- **Directory Exists** – verifies whether a folder exists in the FTP or SFTP site

Push

1. In the **Operation Type** list, select **Push**.

The screenshot shows a configuration window for 'FTP Activity'. It has a title bar with a dropdown arrow. Below the title, there are three icons: a warning triangle, an information 'i' icon, and a speech bubble icon. The main content area contains several dropdown menus: 'Operation Type' (set to 'Push'), 'Source File Path', 'Destination Folder Path', 'Communication Mode' (set to 'Active'), and 'Transfer Mode' (set to 'Binary'). At the bottom of the main area is a blue link labeled 'Result Mapping'. The window has a light gray background and a white border.

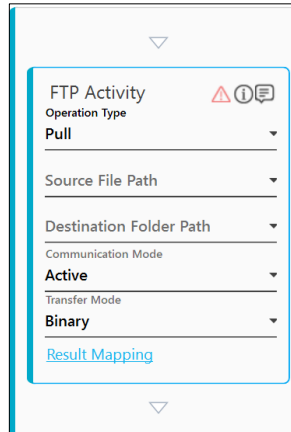
2. In the **Source File Path** list, select the parameter containing the path of the file to be uploaded (along with the file name, for example- "D:\Data\PDFOutput.pdf"). You must pre-define the parameter with the file path to make it appear for the selection.
3. In the **Destination Folder Path** list, select the parameter containing the path of the FTP or SFTP site location where you want to upload the file, for example- "\Output"). You must pre-define the parameter with the folder path to make it appear for the selection.
4. In the **Communication Mode** list, select **Active** or **Passive** depending on how the data connection is established between the client and the server.
5. In the **Transfer Mode** list, select **Binary** or **ASCII (Text)** depending on the file type to be transferred. You cannot transfer images file type using **ASCII (Text)** mode.
6. Click **Result Mapping** link, the **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the upload is successful, it returns true and if the operations fails it returns false. It can be used as an input for the next action that you want to perform depending upon the success or failure of the operation.

Note: Once you setup the environment, you need to provide the FTP or SFTP login credentials.

The file gets uploaded in the FTP folder once you run the process. You can test run the process post setting up the environment to verify if the file is uploaded to the required FTP or SFTP site location.

Pull

1. In the **Operation Type** list, select **Pull**.



2. In the **Source File Path** list, select the parameter containing the path of the file to be downloaded from the FTP or the SFTP site location for example- "\Output\PDFOutput.pdf" (if document is present at the root location itself, mention only the file name like "\PDFOutput.pdf"). You must define the parameter in the [Parameter](#) bar with the file path to let it appear for the selection.
3. In the **Destination Folder Path** list, select the parameter containing the path of the folder where you want to save the downloaded file, for example- "D:\Data\"). You must define the parameter in the Parameter bar with the folder path to let it appear for the selection.
4. In the **Communication Mode** list, select **Active** or **Passive** depending on how the data connection is established between the client and the server.
5. In the **Transfer Mode** list, select **Binary** or **ASCII (Text)** depending on the file type to be transferred. You cannot transfer images file type using **ASCII (Text)** mode.
6. Click **Result Mapping** link, **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must define the parameter in the Parameter bar to let it appear for the selection. If the download is successful, it returns true and if the operations fails it returns false. It can be used as an input for the next action that you want to perform depending upon the success or failure of the operation.

Note: Once you setup the environment, you need to provide the FTP or SFTP login credentials.

The file gets downloaded from the FTP folder once you run the process. You can test run the process post setting up the environment to verify if the file is downloaded from the required FTP or SFTP site location.

List Directory

The list of files and folders is saved inside the excel file that gets created while performing this operation. If an excel file is already present at the location with same name, this operations overrides the file.

Note:

The excel file extension supported is only .xlsx.

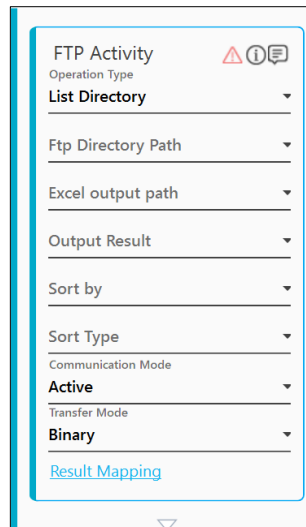
Use Advanced Loop activity to further filter out the required list of files post this operation as per your requirement. This iterates the process to run through the list of directory files to filter the required list of files, based on the filter expression or condition provided.

In the Advanced Loop activity, only DataTable and Excel File is recommended to use. Refer Advanced Loop activity for more details.

You can use an If activity along with FTP Activity to perform required FTP action such as download, upload, rename the required files from the filtered list of directory files. See the example to know how it is used. This iterates the process of required FTP action to run through the filtered list of files and makes the files available based on the FTP action performed.

You can use a wildcard to filter a list of similar files. See the example to know how it is used.

1. In the Operation Type list, select List Directory.



2. In the **Ftp Directory Path** list, select the parameter containing the FTP or SFTP directory path from where you want to list the files., for example- "\\FTP\". You must pre-define the parameter with the directory path to make it appear for the selection.
3. In the **Excel output path** list, select the parameter containing the path of the excel file (along with the file name, for example- "D:\DataTransfer\ExcelOutput.xlsx"). You must define the parameter in the [Parameter](#) bar to let it appear for the selection.
4. In the **Output Result** list, select the parameter where the output of the sorted file list is saved.
5. In the **Sort by** list, select the criteria based on which you want to sort the files. Available options are- **Name**, **DateModified**, **Size** and **Type**.

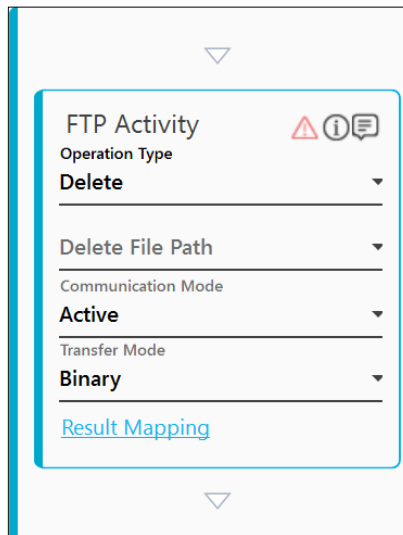
6. In the **Sort Type** list, select the order by which you to apply the sorting criteria. Available options are- **ASC** and **DESC**. **ASC** signifies ascending order while **DESC** signifies descending order.
7. In the **Communication Mode** list, select **Active** or **Passive** depending on how the data connection is established between the client and the server.
8. In the **Transfer Mode** list, select **Binary** or **ASCII (Text)** depending on the content of the file to be transferred. You cannot transfer images file type using **ASCII (Text)** mode.
9. Click **Result Mapping** link, **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must define the parameter in the Parameter bar to let it appear for the selection. If the operation is successful, it returns true and if the operations fails it returns false. It can be used as an input for the next action that you want to perform depending upon the success or failure of the operation.

Note: Once you setup the environment, you need to provide the FTP or SFTP login credentials.

Once you run the process, the list of files is available in the mentioned excel file. You can test run the process post setting up the environment to verify if the list of files and folders is available.

Delete

1. In the **Operation Type** list, select **Delete**.



2. In the **Delete File Path** list, select the parameter containing the path of the file to be deleted from the FTP or SFTP site location (along with the file name, for example- "\Output\PDFOutput.pdf" (if document is present at the root location itself, mention only the file name like "\PDFOutput.pdf"). You must define the parameter in the Parameter bar to let it appear for the selection.
3. In the **Communication Mode** list, select **Active** or **Passive** depending on how the data connection is established between the client and the server.
4. In the **Transfer Mode** list, select Binary or ASCII (Text) depending on the file type to be transferred. You cannot transfer images file type using ASCII (Text) mode.

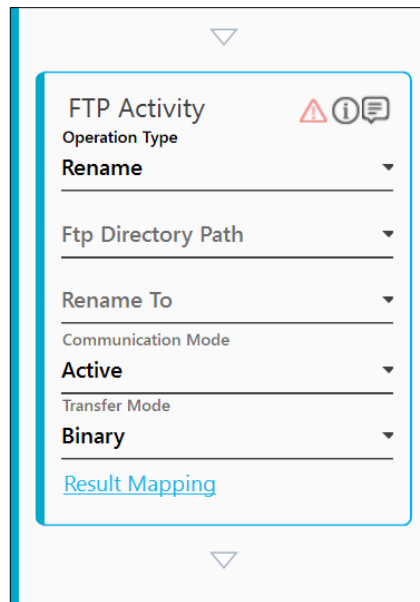
- Click **Result Mapping** link, **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must define the parameter in the Parameter bar to let it appear for the selection. If the rename is successful, it returns true and if the operations fails it returns false. It can be used as an input for the next action that you want to perform depending upon the success or failure of the operation.

Note: Once you setup the environment, you need to provide the FTP/SFTP login credentials.

The file gets deleted from the FTP folder once you run the process. You can test run the process post setting up the environment to verify if the file is deleted from the specified location.

Rename

- In the **Operation Type** list, select **Rename**.

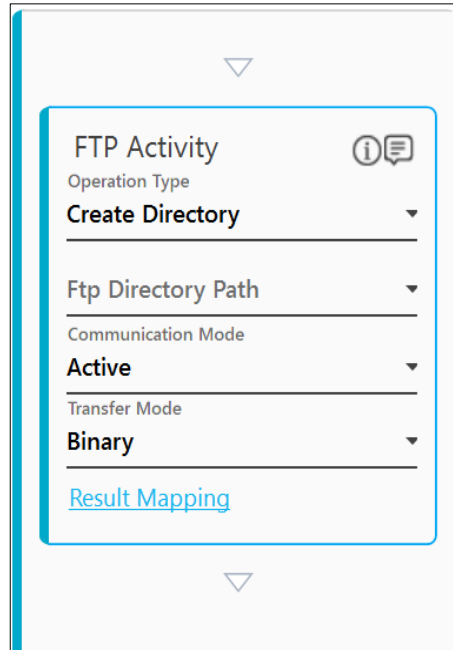


- In the **Ftp Directory Path** list, select the parameter containing the path of the file to be renamed in the FTP folder(along with the file name, for example- " \Output\PDFOutput.pdf"). You must pre-define the parameter with the file path to make it appear for the selection.
- In the **Rename to** list, select the parameter containing the name with which you want to rename the file. You must pre-define the parameter with the new name to make it appear for the selection.
- In the **Communication Mode** list, select **Active** or **Passive** depending on how the data connection is established between the client and the server.
- In the **Transfer Mode** list, select **Binary** or **ASCII (Text)** depending on the file type to be transferred. You cannot transfer images file type using **ASCII (Text)** mode.
- Click **Result Mapping** link, **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must define the parameter in the Parameter bar to let it appear for the selection. If the delete is successful, it returns true and if the operations fails it returns false. It can be used as an input for the next action that you want to perform depending upon the success or failure of the operation. **Note:** Once you setup the environment, you need to provide the FTP or SFTP login credentials.

The file gets renamed in the FTP folder once you run the process. You can test run the process post setting up the environment to verify if the file is renamed at the specified location.

Create Directory

1. In the Operation Type list, select Create Directory.



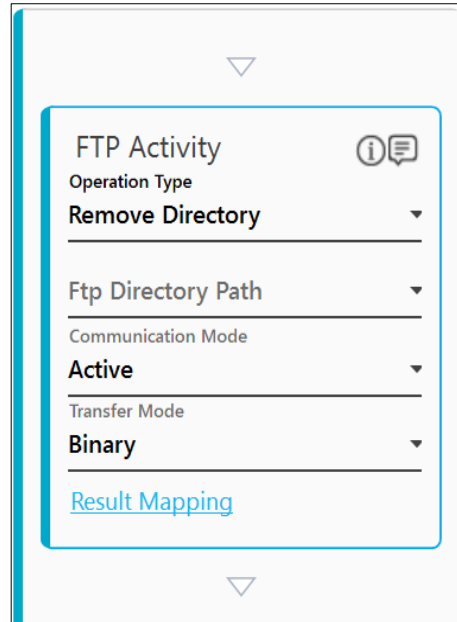
2. In the Ftp Directory Path list, select the parameter containing the FTP directory path, along with the name you want to provide the directory, for example- "/tempfolder/demo1". If you want to create the directory at the root location itself, mention only the directory name like "/demo1". You must pre-define the parameter with the FTP directory path to make it appear for the selection.
3. In the Communication Mode list, select Active or Passive depending on how the data connection is established between the client and the server.
4. In the Transfer Mode list, select Binary or ASCII (Text) depending on the file type to be transferred. You cannot transfer images file type using ASCII (Text) mode.
5. Click Result Mapping link, Output Mapping list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must define the parameter in the Parameter bar to let it appear for the selection. If the creation of the folder is successful, it returns true and if the operations fails it returns false. It can be used as an input for the next action that you want to perform depending upon the success or failure of the operation.

Note: Once you setup the environment, you need to provide the FTP or SFTP login credentials.

The directory gets created in the FTP folder once you run the process. You can test run the process post setting up the environment to verify if the directory gets created at FTP or SFTP site location.

Remove Directory

1. In the Operation Type list, select Remove Directory.



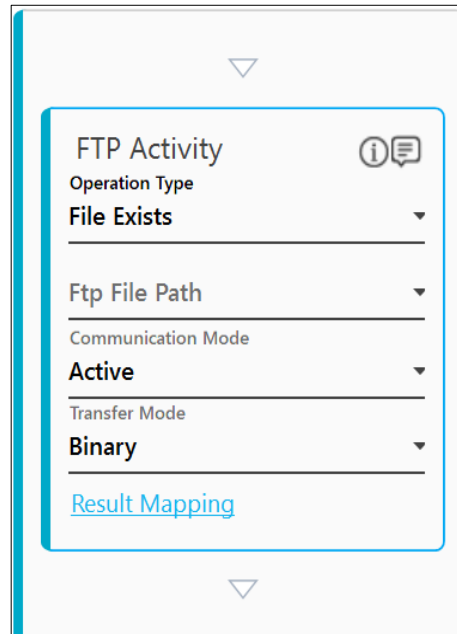
2. In the Ftp Directory Path list, select the parameter containing the path of the FTP directory to be removed (for example- "/tempfolder/demo1"). If the directory is present at the root location itself, mention only the directory name like "/demo1". You must pre-define the parameter with the FTP directory path to make it appear for the selection.
3. In the Communication Mode list, select Active or Passive depending on how the data connection is established between the client and the server.
4. In the Transfer Mode list, select Binary or ASCII (Text) depending on the file type to be transferred. You cannot transfer images file type using ASCII (Text) mode.
5. Click Result Mapping link, Output Mapping list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must define the parameter in the Parameter bar to let it appear for the selection. If the delete is successful, it returns true and if the operations fails it returns false. It can be used as an input for the next action that you want to perform depending upon the success or failure of the operation.

Note: Once you setup the environment, you need to provide the FTP or SFTP login credentials.

The directory gets removed from the FTP site location once you run the process. You can test run the process post setting up the environment to verify if the directory gets removed from the FTP or SFTP site location.

File Exists

1. In the Operation Type list, select File Exists.

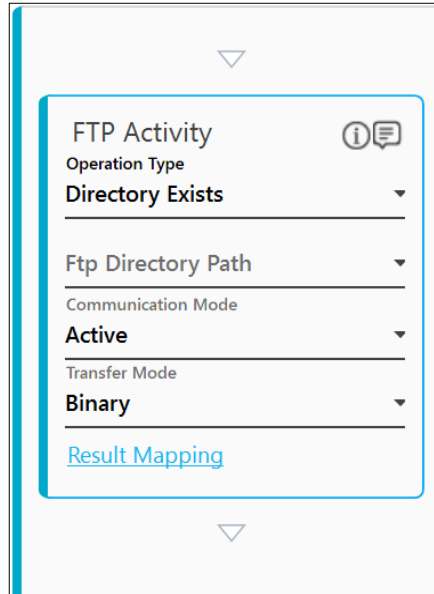


2. In the Ftp File Path list, select the parameter containing the path of the file to be verified, for example- "/tempfolder/Crypto.pdf". You must pre-define the parameter with the file path to make it appear for the selection.
3. In the Communication Mode list, select Active or Passive depending on how the data connection is established between the client and the server.
4. In the Transfer Mode list, select Binary or ASCII (Text) depending on the file type to be transferred. You cannot transfer images file type using ASCII (Text) mode.
5. Click Result Mapping link, Output Mapping list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must define the parameter in the Parameter bar to let it appear for the selection. If the file exists, it returns true and if the operations fails it returns false. It can be used as an input for the next action that you want to perform depending upon the success or failure of the operation.

Note: Once you setup the environment, you need to provide the FTP or SFTP login credentials.

Directory Exists

1. In the Operation Type list, select Directory Exists.



2. In the Ftp Directory Path list, select the parameter containing the path of the directory to be verified, for example- "/tempfolder/". You must pre-define the parameter with the directory path to make it appear for the selection.
3. In the Communication Mode list, select Active or Passive depending on how the data connection is established between the client and the server.
4. In the Transfer Mode list, select Binary or ASCII (Text) depending on the file type to be transferred. You cannot transfer images file type using ASCII (Text) mode.
5. Click Result Mapping link, Output Mapping list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must define the parameter in the Parameter bar to let it appear for the selection. If the directory exists, it returns true and if the operations fails it returns false. It can be used as an input for the next action that you want to perform depending upon the success or failure of the operation.

Note: Once you setup the environment, you need to provide the FTP or SFTP login credentials.

FTP Activity Properties

The properties of a FTP Activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity.

Property Name	Usage
	<p>If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as JSON Activity . You can change the name as required.
IsSftp	Signifies if SFTP server is used for the automation.


Example of FTP Activity

This example shows how to download multiple files from the directory of a FTP server and save them at a preferred location on your system.

The List Directory FTP action lists all the files available at the intended directory. Post this action, we are going to use a loop activity to filter the directory list based on our requirement and then the Pull FTP action in another loop to download the required files.

Prerequisite:

1. In the **Admin** menu, add a **FTPApps** application.
2. In the Application Properties panel:
 - a. In the **Display Name** field, enter a desired name of the FTP server you want to add.
 - b. In the **Server IP** field, enter the IP address of the FTP server.
 - c. In the **Server Port** field, enter the port number of the FTP server.
 - d. In the **Transfer Type** list, select whether you want to configure FTP server.

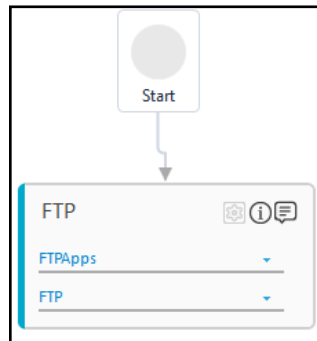
3. Click the  (**Save Properties**) icon to save the application details

The FTP server is configured.

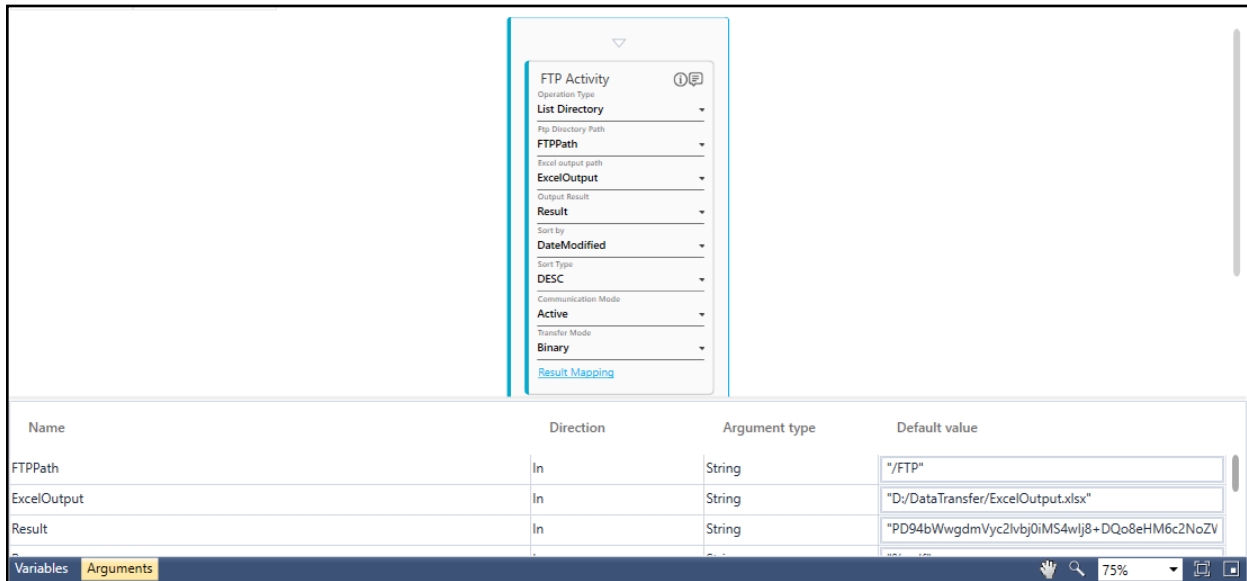
To automate the process of downloading required files and save them at a particular location:

Create List of Directory Files

1. Create a new process.
2. In the Parameter bar, create an in argument, **FTPPath**, **ExcelOutput**, and **Result** for the first FTP Activity that fetches the list of directory files.
 - a. In the **FTPPath** argument, set the FTP directory location from where you want to fetch the list of directory.
 - b. In the **ExcelOutput** argument, set the location of an empty excel file where you want the save the list of directory files.
 - c. In the **Result** argument, the output result of the FTP operations is saved. The results are stored during the runtime only, it is not available once the process execution completes.
3. From the Canvas Tools panel, add an Application activity to the Flowchart designer on the Canvas
4. In the Application Type list, select **FTPApps**.
5. In the Select an Application list, select the added FTP application.



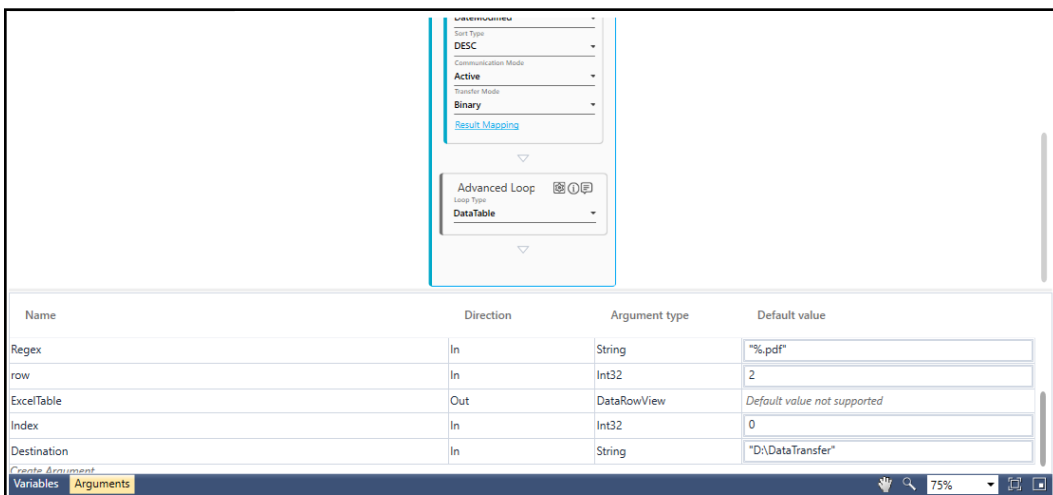
- Double click the Application activity, drag the FTP Activity and drop inside the Application activity.




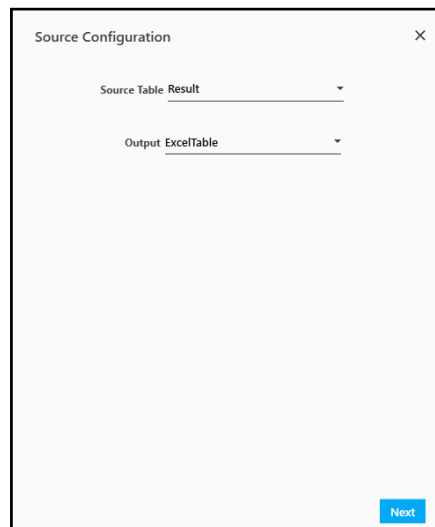
- In the Operation Type list, select List Directory.
- In the **Ftp Directory Path** list, select the argument- **FTPPath**.
- In the **Excel Output Path** list, select the argument- **ExcelOutput**.
- In the **Result** list, select the argument- **Result**.
- In the **Sort by** list, select the preferred criteria of sorting the list. You can sort the files based upon their name, date of modification, file size and file type.
- In the **Sort Type** list, select the preferred type of sorting. You can sort the file in the ascending or descending order.
- In the **Communication Mode** list, select **Active** or **Passive** depending on how the data connection is established between the client and the server.
- In the **Transfer Mode** list, select **Binary** or **ASCII (Text)** depending on the content of the file to be transferred. You cannot transfer images file type using **ASCII (Text)** mode.

Filter the List of Directory Files Based On Required Criteria

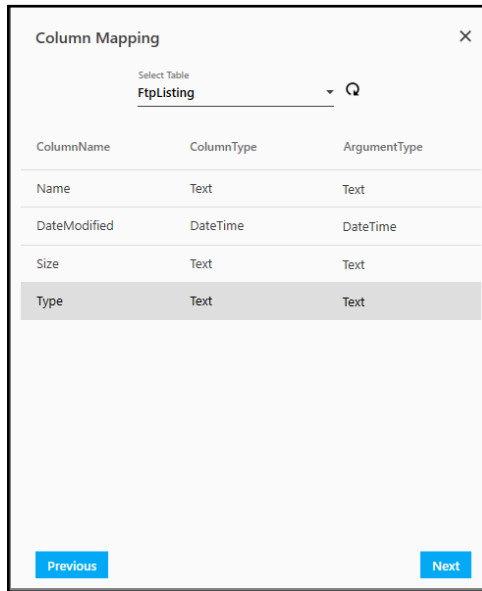
1. Add an **Advanced Loop** activity below the **FTP Activity**. This helps us to filter the list of directory files by running through each file name in a loop until the entire list is checked. We need to pick the intended file name and save it in a separate file. Refer **Advanced Loop** activity for more details.
2. Create in arguments, **Regex** an out argument, **ExcelTable** of **Argument Type- System.Data.DataRowView**. These arguments are used to provide input into the **Advance Loop** Activity, and access the data returned in the list of filtered files in the datatable or excel file configured.
 - a. In the **Regex** argument, set the wildcard based on which you want to filter the files.
 - b. In the **ExcelTable** argument, the list of filtered files is saved.



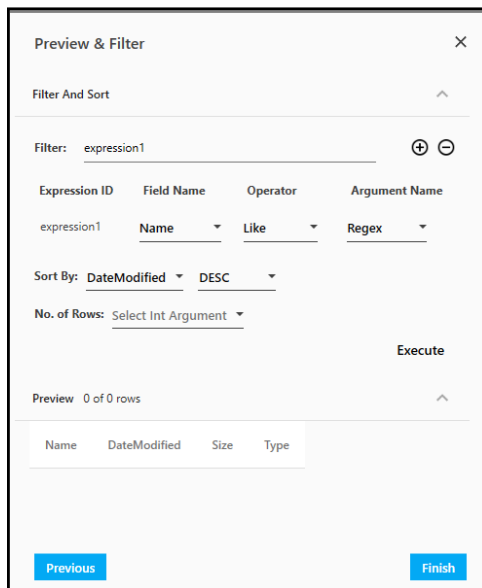
3. In the **Loop Type** list, select the preferred loop type. For FTP related processes, it is recommended to choose between **Excel File** and **DataTable** to avoid creating complexities and errors.
4. Click the  (**Settings**) icon to configure the loop. The Source Configuration dialog box appears.



5. In the **Source Table** list, select the argument, **Result** as the list of directory files from where you want you filter required files is saved in this argument.
6. In the **Output** list, select the argument, **ExcelTable** to store the filtered list.
7. Click **Next**. The **Column Mapping** details appear. The details such as the column name, column type and argument type, related to the table that stores the filtered file names are displayed.



8. Click **Next**. The **Preview and Filter** details appear.



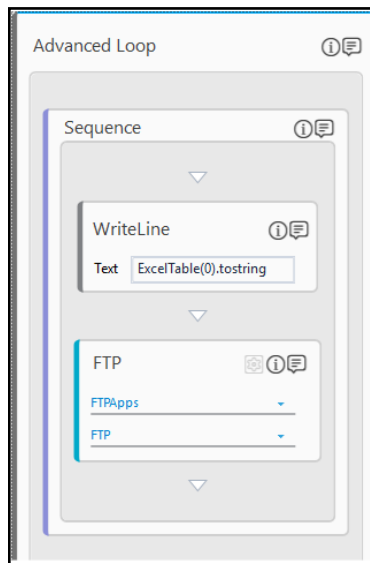
9. Expand **Filter And Sort** and click the **+** (**Add filter expression**) icon to add the filter expression.
 - a. In the **Field Name** list, select the field name where you want to apply the filter.
 - b. In the **Operator list**, select the filter operator. Available operators are, **Like**, **Equals** and **Not Equals**.

- c. In the **Argument Name** list, select the wild card argument, **Regex**.
 - d. In the **Sort By** list, select the sorting criteria if you want to sort the filtered list based on the file name and the order.
 - e. In the **No. of Rows** list, you can select an argument containing the row number that indicates the row or rows till which the result should be fetched from the **Output** table generated as part of **List Directory** operation.
 - f. Click **Execute** to apply the filter and sort criteria. A preview of the filtered list can be seen under the **Preview** pane.
 - g. Click **Finish**. Once the filter expression is set, the loop validates the filter expression for each record present in the source table containing list of directory files and saves the data in the **Output** table selected.
10. To perform the FTP action on the available list of filtered list of directory files, we need to use another loop that runs through the entire list and downloads the required files.

Iterate FTP action on Filtered List to Download Required Set of Files

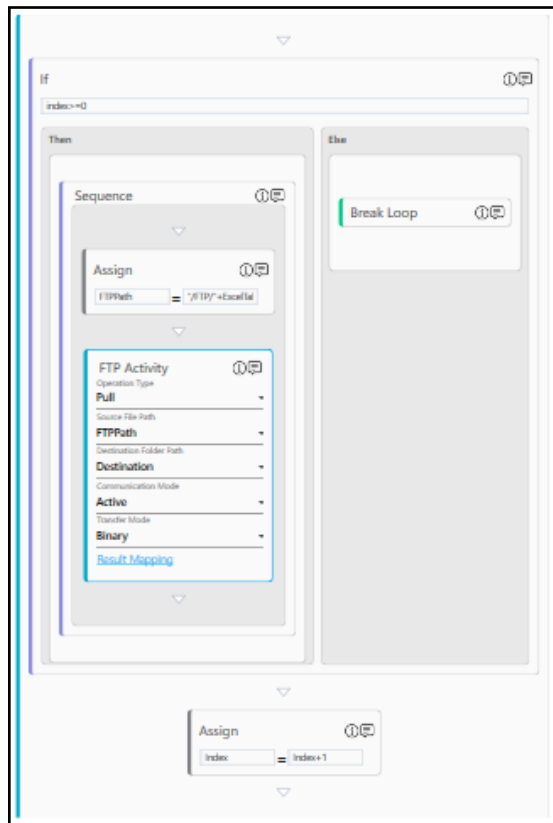
To test run the process in Automation Studio and see the process output you can add a WriteLine activity in the flow.

1. Double click the **Advanced Loop** activity, and drop the **WriteLine** activity to convert the data received in the Output table of above loop, into a string. This string stores the result of the Advanced Loop operation that is the filtered file name and perform the FTP action every time the required file name is encountered in this loop.
2. As an **FTP Activity** needs to be inside an **Application** Activity, drag an **Application** activity and drop below the **WriteLine** activity.



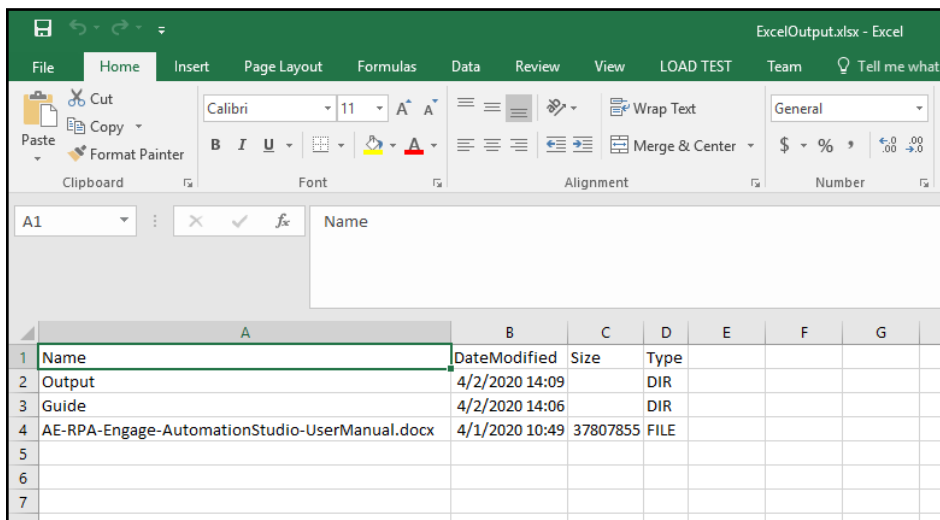
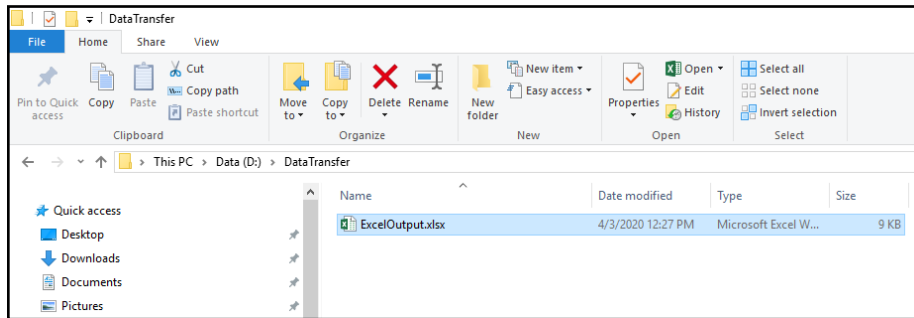
3. Drag the **Application** activity and drop below the **WriteLine** activity.
4. Set the **Application Type** as **FTPApps** and select the FTP application in the **Select An Application** list.
5. Double click the **Application** activity and drop the **If** activity and drop inside it to iterate the FTP operations of downloading the required files.

6. Create in arguments, **Index** and **Destination**. These arguments are used to set the loop through the entire filtered list and save the files in the required location.



7. Set the If condition as Index >= 0 and increment it by 1 to run through the entire filtered list of directory files saved in the **Output** table.
8. If True:
 - a. Drop **Assign** activity. Assign the value of the FTP directory and the filename filtered from the previous loop such as, **"\"+ExcelTable.toString** to the **FTPPath** argument.
 - b. Drop the FTP Activity and in the Operation Type set the type as Pull.
 - c. In the **Source File Path** list, select the argument, **FTPPath**.
 - d. In the **Destination Folder Path** list, select the argument, **Destination**.
 - e. Select Communication Mode and Transfer Mode.

10. If False, exit the loop.



The required files are downloaded, based on the filter criteria provided, at destination folder.

9.6.17 Execute DB Query

This activity allows you to automate the task of updating or retrieving data from a database.

Using Database Query Activity

- If Oracle is configured as the database, it can be used for Execute Database Query Activity. To perform this activity following assembly files are required:

Component	Version
Oracle.ManagedDataAccess.dll	3.21.61
Oracle.ManagedDataAccess.EntityFramework.dll	6.21.61

- If MySQL is configured as the database, it can be used for Execute Database Query Activity. To perform this activity following assembly files are required.

Component	Version	Remarks
MySQL.Data.dll	8.0.31	Download and install MySQL connector from the below URL https://dev.mysql.com/downloads/connector/net/ . Copy MySQL.Data.dll to Studio and Robot folders.

- If Excel is configured as the database, it can be used for Execute Database Query Activity. To perform this activity following assembly files are required.

Component	Version	Remarks
AccessDatabaseEngine. Exe	2007	Download and install from the URL https://www.microsoft.com/en-in/download/details.aspx?id=4438

The Execute DB Query activity can be used only inside an Application activity where Application Type is set to DatabaseApps. The step-by-step process to use the Execute DB Query activity is explained in [Database Application Activity](#).

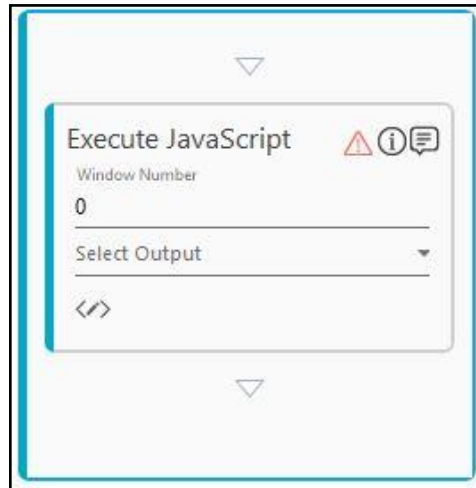
9.6.18 Execute Java Script

It activity allows you to perform automation of controls on a web page where a Java script is present. It can be helpful in scenarios where any of the controls on a web page do not get automated using the web application automation method.

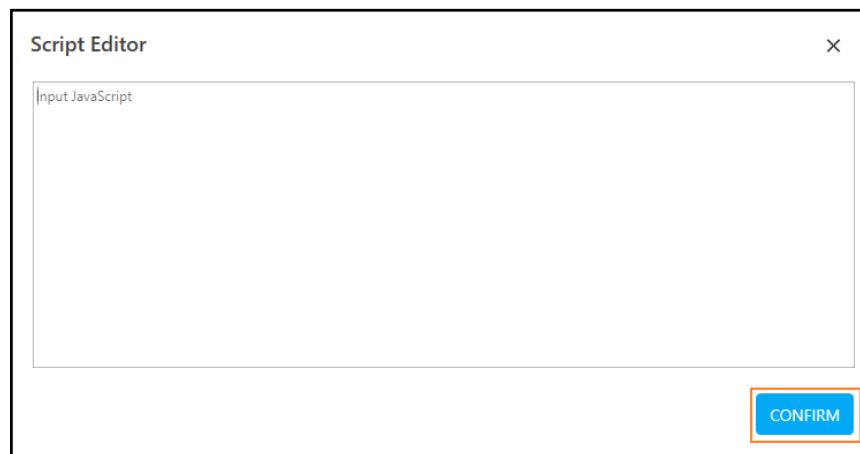
The Execute Java Script activity can be used only inside an Application activity where Application Type is set to WebApps.

Using Execute Java Script

1. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
2. Drag the **Application** activity and drop on to the Flowchart designer on the Canvas
3. In the Application Type list, select the type of application you want to perform automation on. The application type that has at least one application added to it appears in the list. See Admin Capabilities to know how to add application before using an activity.
4. In the Select An Application list, select the web application you want to perform automation on. You can add a web application that is already configure in Automation Studio or add a new application at this level.
5. Double click to open the Application activity, drag the **Execute JavaScript** activity and drop inside the Application activity.



6. In the **Window Number** field, enter the index of the window where you want to execute the Java script.
7. In the **Select Output** list, select the parameter to assign the return value. You must define the parameter in the Parameter bar use this option.
8. Click the **</>** (**Edit Script**) icon to open the **Script Editor**.



9. In the **Script Writer** dialog box, enter the Java script you want to run on the web page.
10. Click **CONFIRM**.

The Execute JavaScript is created.

Execute JavaScript Properties

The properties of Execute JavaScript activity are listed in the following table and can be edited in the Property grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as Application . You can change the name as required.
JavaScript	The Java script that you want to run on the web page.
WindowNumber	The index of the window where you want to execute the Java script. The index number starts with 0.

9.6.19 Key Events

It helps you to automate keystroke in an application which can boost productivity and increase efficiency of the daily tasks performed. A keystroke is a single key or a combination of key presses on a keyboard to perform a certain action.

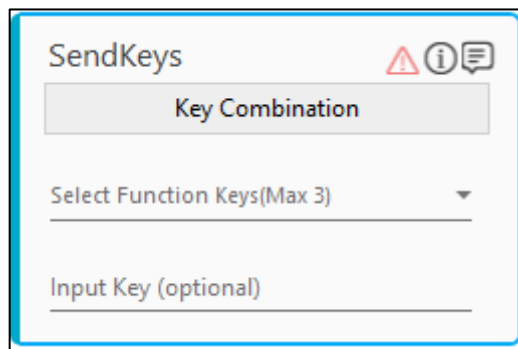
This activity works only in a focused mode. Bring the intended application into focus using the Focus Window activity and then

use this activity to configure the keystroke to perform the intended action. See Focus Window activity section to know how to bring an application in focus.

CAUTION: This activity does not work with any security related pop ups and IE browser of version 9 or above.

Using Focus Window Activity

1. In the **Canvas Toolspane**, click Process Components to expand the tool and view the associated activities.
2. Drag the Key Events activity and drop on to the Flowchart designer on the Canvas The validation error symbol disappears when required inputs are provided.

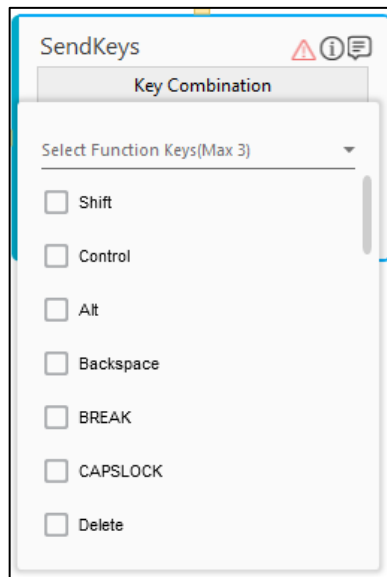


3. Configure the function Key, keystroke count and the input keys to form the combination of the keystroke for automation. You can configure a maximum of three keys while forming the keystroke combination. For a single function key, you can set the number the keystroke counts you want to perform along with one key input. However, you can set one key input if you want to configure a combination of two function keys, and no key input can be configured if you want to form a combination of three function keys. Click the following link to know the related steps in detail:
 - [Select Function Keys](#)
 - [Keystroke Count](#)
 - [Input Key](#)
 - [Keystroke Count](#)

Select Function Keys

The function keys are the keys available on the standard QWERTY keyboard.

In the Select Function Keys(Max 3) list, select the required function key. You can select a maximum of three keys. Available options are:



- Shift
- Ctrl
- Alt
- Backspace
- BREAK
- CAPSLOCK
- Delete
- ArrowUp
- ArrowDown
- ArrowRight
- ArrowLeft
- End
- Enter
- Space
- Escape
- Home
- Insert
- NUM LOCK
- PageDown

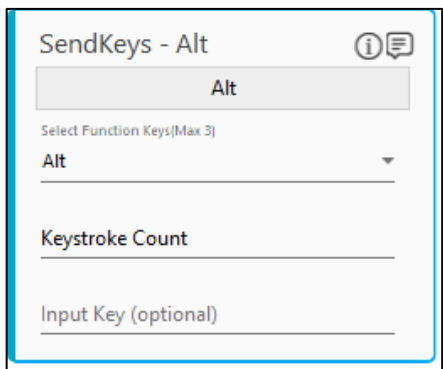
- PAGE UP
- Print Screen
- Tab
- Function Keys (F1-F12)

Keystroke Count

This field is available only when a single function key is selected that needs to be pressed multiple times to perform a certain action.

Note: Keystroke Count is currently not supported in the IE browser.

In the Keystroke Count field, enter the number counts the selected function key must be pressed. For example, if the Tab key needs to be pressed three times, then specify the number 3.



Input Key

It sets an input alphanumeric or special character key as a keystroke. Use this option either in combination with a function Key or an independent keystroke. This field is available only when either three function Keys are selected or when the Keystroke Count option is used.

In the Input Key (optional) field, enter any alphanumeric or special character to configure the keystroke. For example, to automate pressing the combination Ctrl + P, in the Select Function Key list, select Control and the in the Input Key field P key.

The Key Events activity is created.

Key Events Properties

The properties of a Key Events activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.

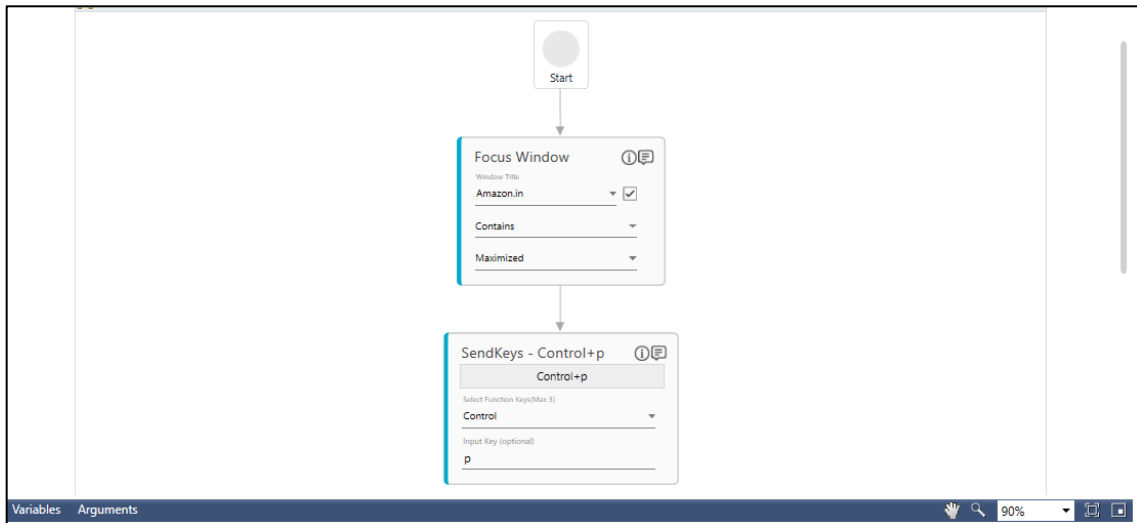
Property Name	Usage
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as per the configured keystroke. You can change the name as required.

[Step-By-Step Guide to Use Key Events to Automate Opening the Print Preview of E-commerce Webpage](#)

Let's create an example to automate opening the print preview of the Amazon web page, using the Ctrl+P keystroke, where you have short listed the top rated PS4 games.

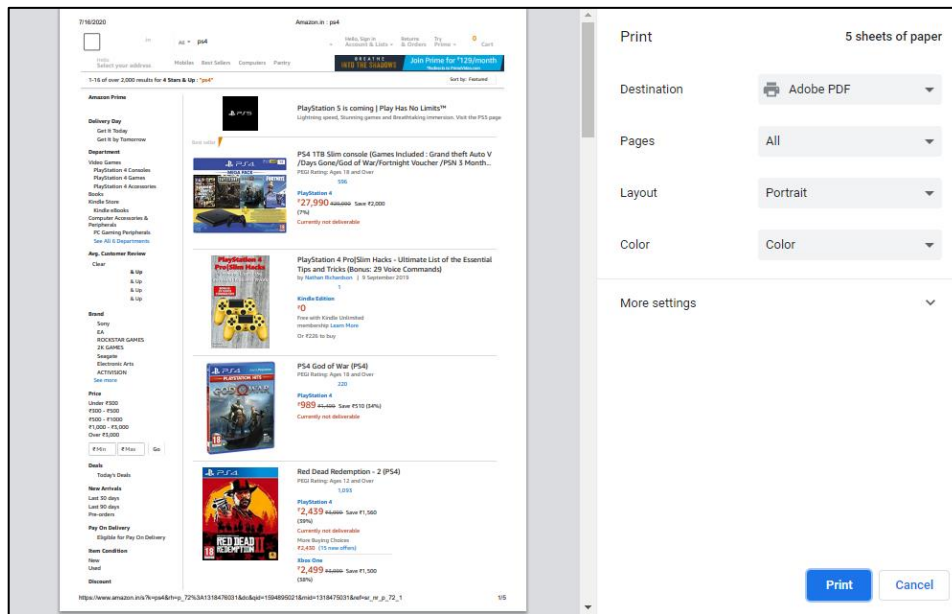
To automate printing using Ctrl+P keystroke:

1. Create a new process.
2. From the **Canvas Tools** panel, add the **Focus Window** activity to the **Flowchart designer** on the **Canvas**
3. Select the **Is Default Value** check box, and then enter **Amazon.in** in the **Window Title** field.
4. In the **Equals** list, select **Contains** to set the search criteria of the Amazon window that needs to be in focus.
5. From the **Canvas Tools** panel, add the **Key Events** activity after the **Focus Window** activity.



6. In the Select Function Keys(Max 3) list, select Control.
7. In the **Input Key (optional)** field, enter **p**.

8. To view the output in Automation Studio, set up the environment and then click **Test Run** to run the process. You can assign this process to a robot, if you want to execute this process outside Automation Studio.



9. The print preview screen appears with different setting options.

9.6.20 Focus Window

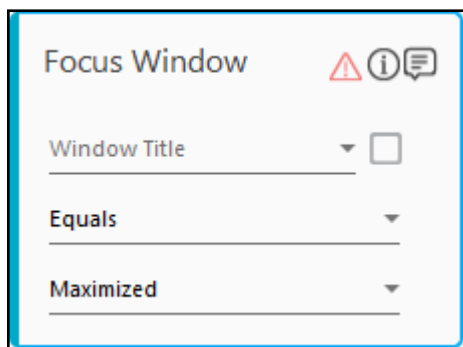
This activity allows you to set focus to an already open application window bringing the window to forefront. You must use this activity before any other activity that requires the window to be in focus to perform the configured action.

Following activities require focus to start capturing automation tasks:

- Print Screen
- Key Events
- Image Control
- Text Editor

Using Focus Window Activity

1. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
2. Drag the **Focus Window** activity and drop on to the Flowchart designer on the Canvas



3. In the **Window Title** list, select a parameter that holds the title you want to display. You must define the parameter in the Parameter bar use this option. Alternatively, select the **Is Default Value** check box, and then enter the title of the window that you want to bring in focus in the **Window Title** field.
4. In the **Equals** list, select the criteria to search for the required window in focus. You can select the intended window in focus based on certain criteria:
 - **StartsWith**- selects the window based on the starting text of the window name
 - **Equals**- selects the window based on the exact match with the window name
 - **EndsWith**- selects the window based on the text with which the window name ends
 - **Contains**- selects the window based on the text present in the window name
 - **Regex**- selects the window with the name matching the search pattern defined through the provided regular expression such as a.b, *txt and others.
5. In the **Maximized** list, select the size of the window when in focus. Available options are:
 - **Maximized**- displays the window with its maximum size
 - **Normal**- displays the window with the standard size of a window

The Focus Window activity with a default display name is created bring the window in focus.

Focus Window Properties

The properties of a Focus Window activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity.</p> <p>If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as Focus Window. You can change the name as required.

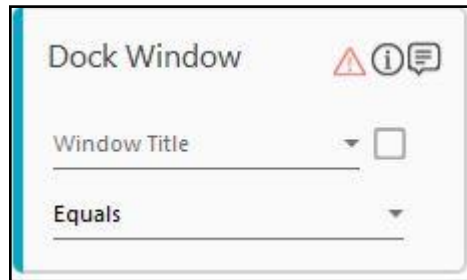
9.6.21 Dock Window

This activity is applicable only to AssistEdge Engage. It allows to dock the application window if it is undocked.

This activity must be used inside an Application activity or the system displays an error.

Using Dock Window

1. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
2. Drag the **Dock Window** activity and drop inside the **Application** activity.



3. In the **Window Title** list, select a parameter that holds the title you want to display. You must define the parameter in the Parameter bar use this option. Alternatively, in the **Window Title** field, enter a title of the window and select the **Is Default Value** check box to set it as the default value of the window title.
4. In the **Equals** drop-down list, select search criteria for the window in focus. You can select the intended window in focus based on certain criteria:
 - **StartsWith**- selects the window based on the starting text of the window name
 - **Equals**- selects the window based on the exact match with the window name
 - **EndsWith**- selects the window based on the text with which the window name ends
 - **Contains**- selects the window based on the text present in the window name
 - **Regex**- selects the window with the name matching the search pattern defined through the provided regular expression such as a.b, *txt and others.

The Dock Window activity with a default display name is created.

Dock Window Property

The properties of a Dock Window activity are listed in the following table and can be edited in the Property grid on the right panel.

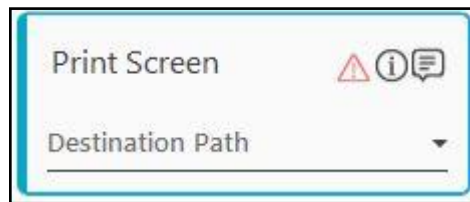
Property Name	Usage
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity. If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as Dock Window . You can change the name as required.
TitleValue	The title of the window to dock.

9.6.22 Print Screen

This activity allows you to take screen shot of the application in focus during execution of the process. See [Focus Window](#) activity to know how to set an application window in focus.

Using Print Screen Activity

1. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
2. Drag the **Print Screen** activity and drop on to the Flowchart designer on the Canvas



3. In the **Destination Path** list, select a parameter that holds the path (include file name with extension) to save the file. You must define the parameter in the Parameter bar use this option. The Print Screen activity with a default display name is created.

Print Screen Properties

The properties of a Print Screen activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.

Property Name	Usage
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as Print Screen . You can change the name as required.

9.6.23 Win Popup Handler

This is an advanced pop up handler that allows you to automate the actions you want to perform on a popup dialog window that appears in front of the web and excel application. Most of the pop ups except the ones related to security can be automated using this activity.

Note:

This activity must be used inside an Application activity or the system displays an error.

Capture all the steps that must take place before the window pop up appears along with the step that triggers the window popup using the [Multimodal Interface](#).

An instance of the application with the window popup must remain open in the configured browser to capture the automation step of handling the window popup.

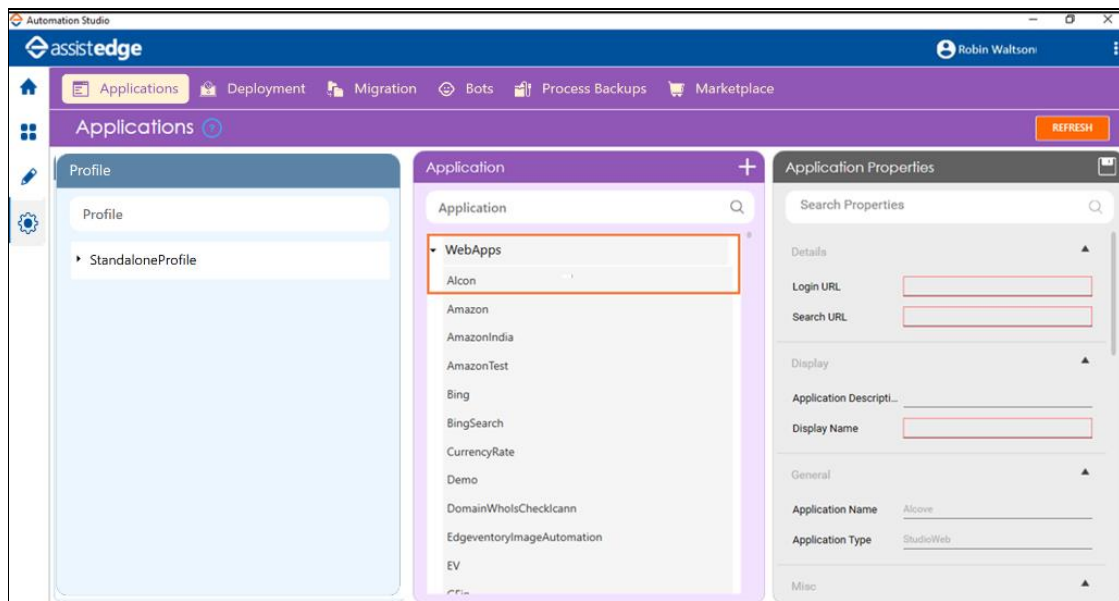
Prerequisite: Configure Web/Excel Application


You must configure the required application in Automation Studio before you start configuring the steps of automation process workflow. This establishes the connection between the intended application and Automation Studio to perform the automation.

Below are the minimum required properties for configuring a web or the excel application. If you want to define remaining properties, refer [WebApps](#) and [ExcelApps](#) for web and excel applications respectively.

To configure web application:

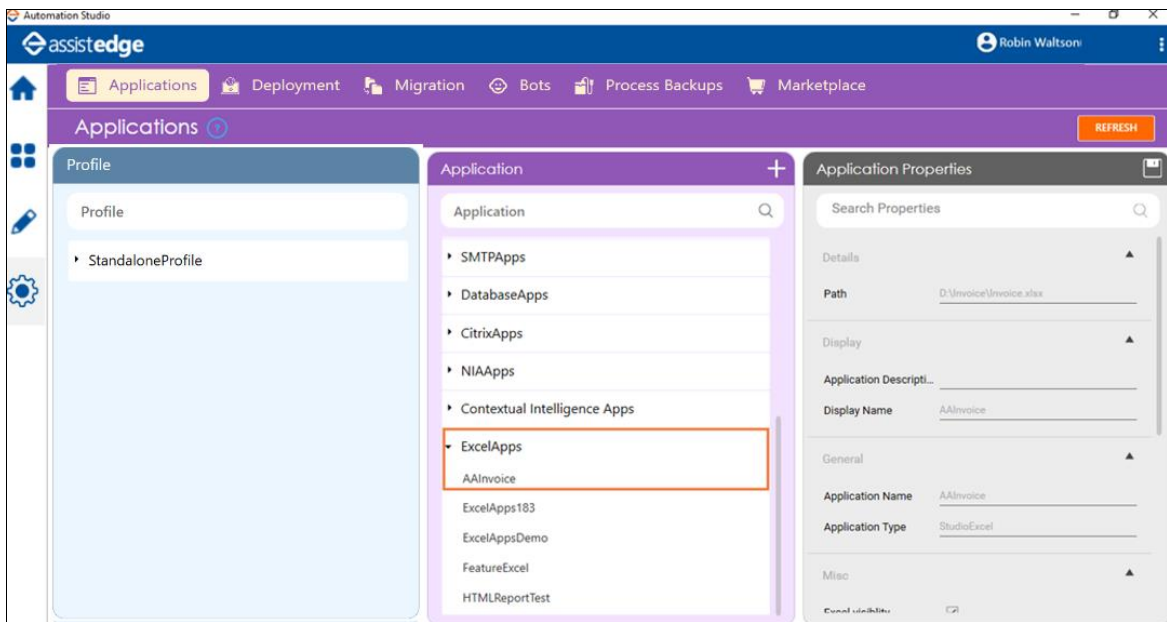
1. In the **Admin** menu, add an application of **Application Type- WebApps**. The mandatory fields are highlighted with red box.



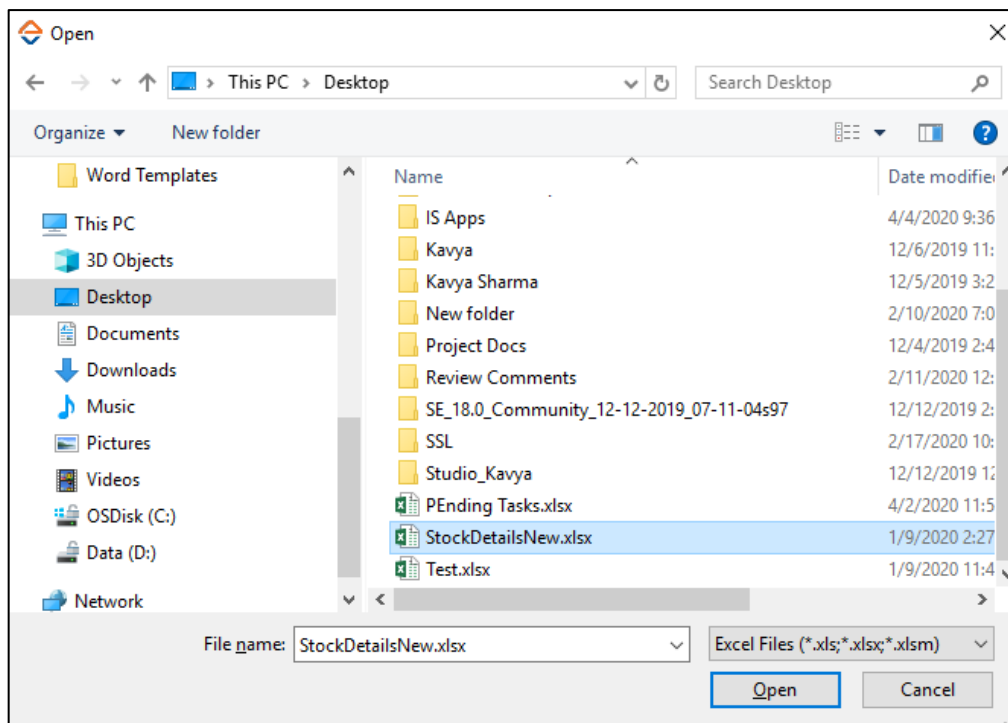
2. In the Application Properties panel:
 - In the **Login URL** field, enter the login URL of the web application you want to access.
 - In the **Search URL** field, enter the URL of the page where you want to perform the automation post login.
 - In the **Display Name**, enter a desired name of the web application.
3. Click the  (**Save Properties**) icon to save the application.
4. The **WebApps** application is configured.

To configure excel application:

1. In the **Admin** menu, add an application of **Application Type- ExcelApps**. The **Add Application** dialog box opens.



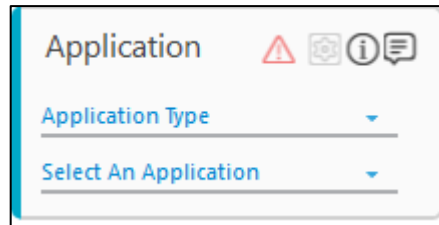
2. In the **Application Name** field, enter a desired name of excel application you want to add. The name must not contain any special character or space.
3. Click **ADD** and browse for the excel file you want to configure. Click **Open**.



The excel application is configured.

Using Win Popup Handler Activity

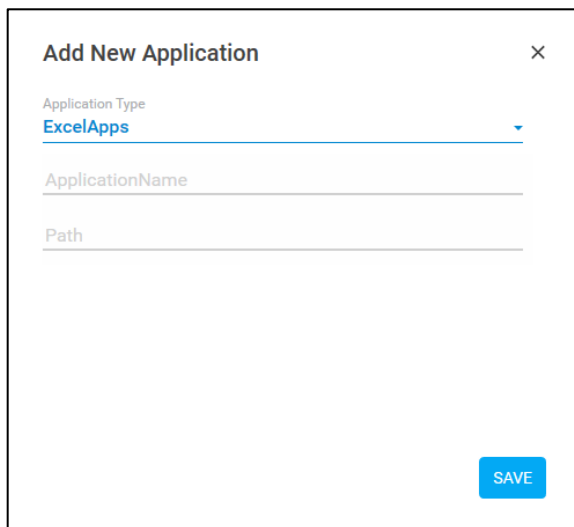
1. In the **Canvas Tools** pane, click **Process Components** to expand the tool and view the associated activities.
2. Drag the **Application** activity and drop on to the **Flowchart designer** on the **Canvas**. The validation error symbol disappears when required inputs are provided.



3. In the **Application Type** list, select **WebApps** for a web application and **ExcelApps** for an excel application.
4. In the **Select An Application** list, select the configured application that you want to automate. Alternatively, you can add a new web or an excel application at this point of time.
5. To add web application:
 - a. In the **Select An Application** list, click Add **New Application**. The **Add New Application** dialog box appears.

- b. In the **Application Name** field, enter a desired name of the web application.
 - c. In the **LoginURL** field, enter the login URL of the web application you want to access.
 - d. In the **SearchURL** field, enter the URL of the page that appears immediately post login.
 - e. In the **PreferredBrowser** list, select the browser you prefer to launch the web application. By default, preference is set to **InternetExplorer**. Available options are:
 - InternetExplorer
 - FireFox
 - Chrome
 - InternetExplorerSelenium
 - f. Click **SAVE**. The web application is added.
6. To add excel application:

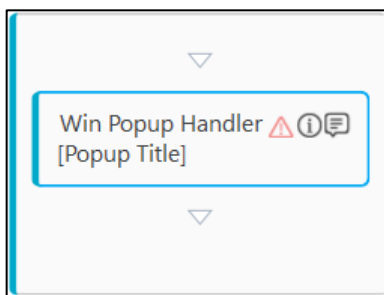
- a. In the Select An Application list, click Add New Application. The Add New Application dialog box appears.



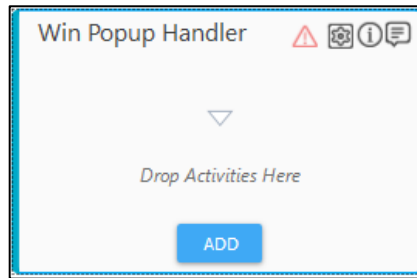
- b. In the **Application Name** field, enter a desired name of the application.
 - c. In the **Path** field, enter the path of the excel file (along with the file name) available on the system.
 - d. Click **SAVE**. The excel file is added.
7. First capture all the steps of the automation process workflow using the [Multimodal Interface](#) until the window popup appears and then add the **Win Popup Handler** activity.


Note: An instance of the application with the window popup must remain open in the configured browser to capture the automation step of handling the window popup.

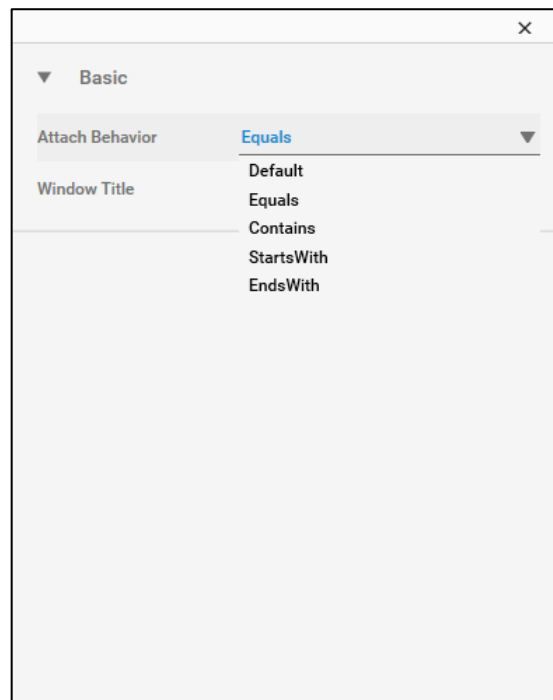
- 8. Double click the **Application** activity, drag the **Win Popup Handler** and drop inside the **Application** activity. The validation error symbol disappears when required inputs are provided.



9. Double click the Win Popup Handler activity.



10. Click the  (Settings) icon. The configuration dialog box of the **Win Popup Handler** activity appears.



11. In the **Attachment Behavior** list, select the identification criteria of the popup window.
12. In the **Window** Title field, enter the title of the popup window.
13. Click the **X** icon to save and close the configuration dialog box.


14. Click **ADD** to add the **Application Control** activity. Alternatively, you can drag the **Application Control** activity from the **Canvas Tools** pane, and drop inside the **Application Host** activity. The **Application Control** activity is used to handle various fields of the popup window. It lets you configuration the UI elements in two different ways. Make sure that an instance of the application with the window popup is open in the configured browser for the **Application Control** activity to dock the required application window. Click any of the links below to know the details:

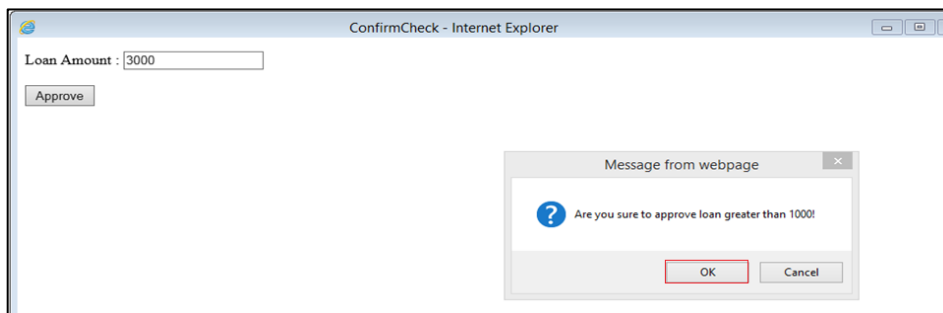


- [Field Configuration](#)
- [Advance Configuration](#)

Field Configuration

Allows to configure the UI elements of the window popup by highlighting the area. It is useful when you only want to capture the area and there is no subsequent action taking place.

1. In the **Application Control** activity, click the  (**Field Configuration**) icon. The required undocked application along with the window popup comes into focus.
2. Press the **Ctrl** key, hover over the window popup and simultaneously select the UI element on the window popup as per your requirement. The UI element that you can capture gets highlighted with a red box.




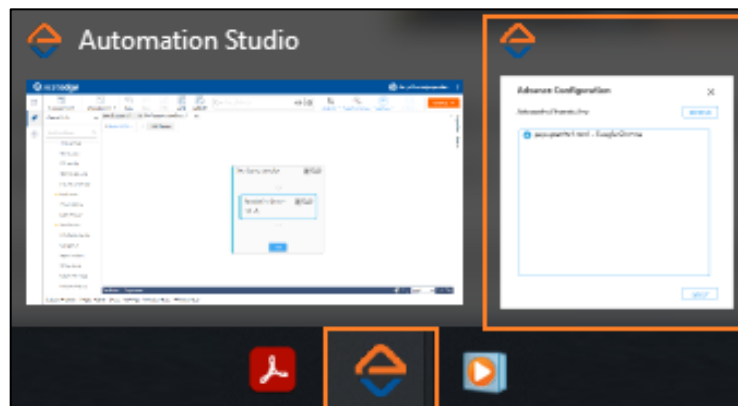
- Click the highlighted area. The configuration dialog box appears. The fields that are mandatory are highlighted with red box.

- In the **Action** list, select the action that you want to perform on the captured area. Other mandatory fields changes depending on the selection from the list. Enter details of all the mandatory fields and other relevant fields as per your requirement. Refer [Field Properties](#) table to know more about the available fields and their respective properties.
- Click the **X** icon to save and close the configuration dialog box.
- The window popup configuration using the **Field Configuration** option is done.

Advance Configuration

Allows to configure the UI elements of the window popup in a hierarchical manner available within an element tree. You can select the intended UI element to open the Field Properties pane without performing the subsequent action. It is also helpful the Field Configuration is not able to select the required UI element.

- In the **Application Control** activity, click the  (**Advanced Field Configuration**) icon. The required undocked application along with the window popup gets into focus. The **Advance Configuration** window box opens in the **Task** bar of the system, available behind the automation Studio window.



- Click **SELECT**. The configuration dialog box appears. The fields that are mandatory are highlighted with red box.

Control Behavior	
Action	[Red Box]
Interaction	
Field Properties	
Display Name	[Red Box]
AutomationId	= [Dropdown]
Class Name	= [Dropdown] ⓘ
Container Class Na...	Chrome_WidgetWin_1 = [Dropdown]
Container Control Ty...	50032
Container Name	This page says = [Dropdown]
Control Index	1 [Dropdown] [Checked]
Control Name	= [Dropdown]

- In the **Action** list, select the action that you want to perform on the captured area. Other mandatory fields changes depending on the selection from the list. Enter details of all the mandatory fields and other relevant fields as per your requirement. Refer [Field Properties](#) table to know more about the available fields and their respective properties.
- Click the **X** icon to save and close the configuration dialog box.
- The window popup configuration using the Advanced Field Configuration option is done.

Win Popup Handler Properties

The properties of a Win Popup Handler activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity. If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as Win Popup Handler . You can change the name as required.
TitleValue	The title of the window popup. Is same as Window Title field .

Step-By-Step Guide to Use Win Popup Handler to Handle the Popup Window While Joining Cisco WebEx Meeting

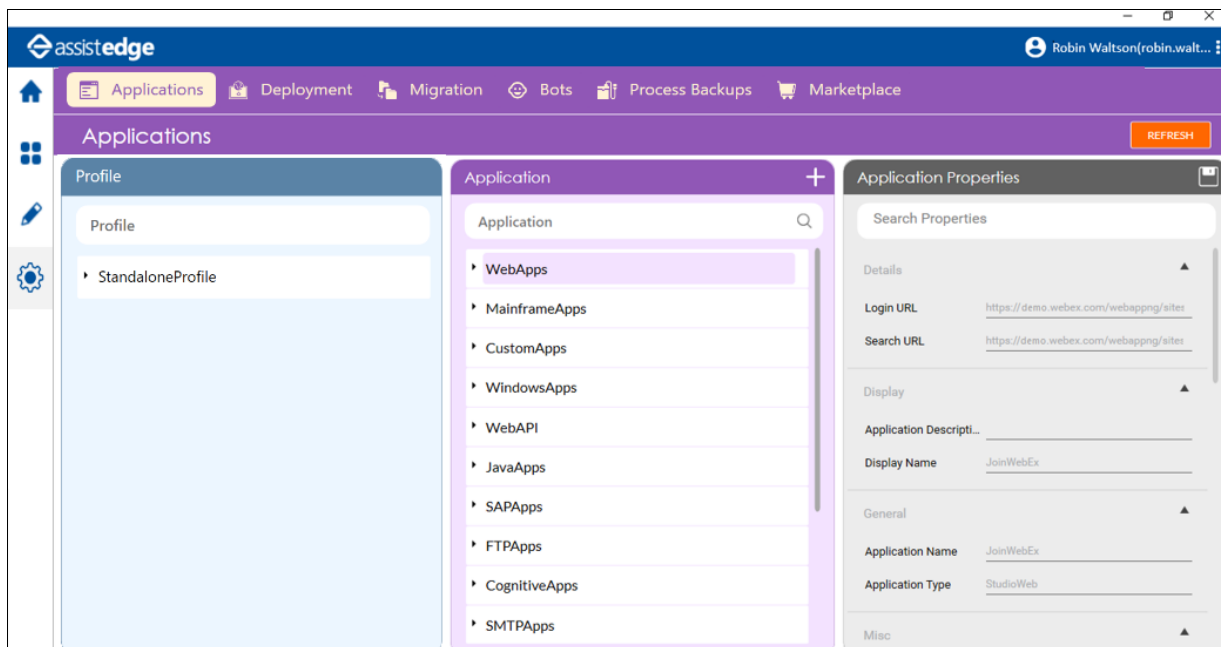
Let's create an example of handling the popup window that appears while joining the Cisco Webex meeting using the Webex URL.


Prerequisite:

1. In the **Admin** menu, add an application of **Application Type- WebApps**.
2. In the Application Properties panel:
 - Enter your enterprise Webex dashboard URL in the **Login URL** field.
 - Enter the same dashboard URL in the **Search URL** field.

Note: Since this application does not require login, hence the login and the search URL remains same. However, it changes if they are different post login.

- Enter a **Display Name** of your choice.

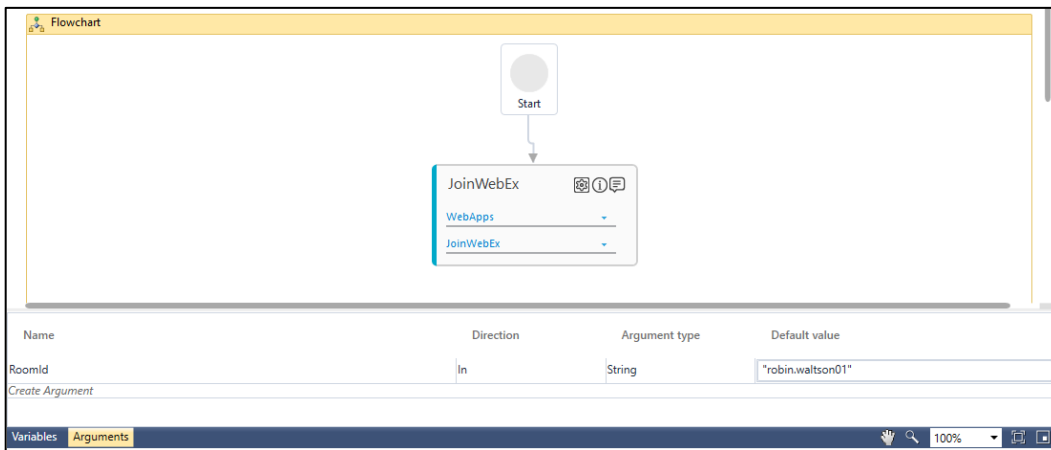


3. Click the  (**Save Properties**) icon to save the application details.
4. Enter other details as per your requirement.
5. The Webex application is configured.

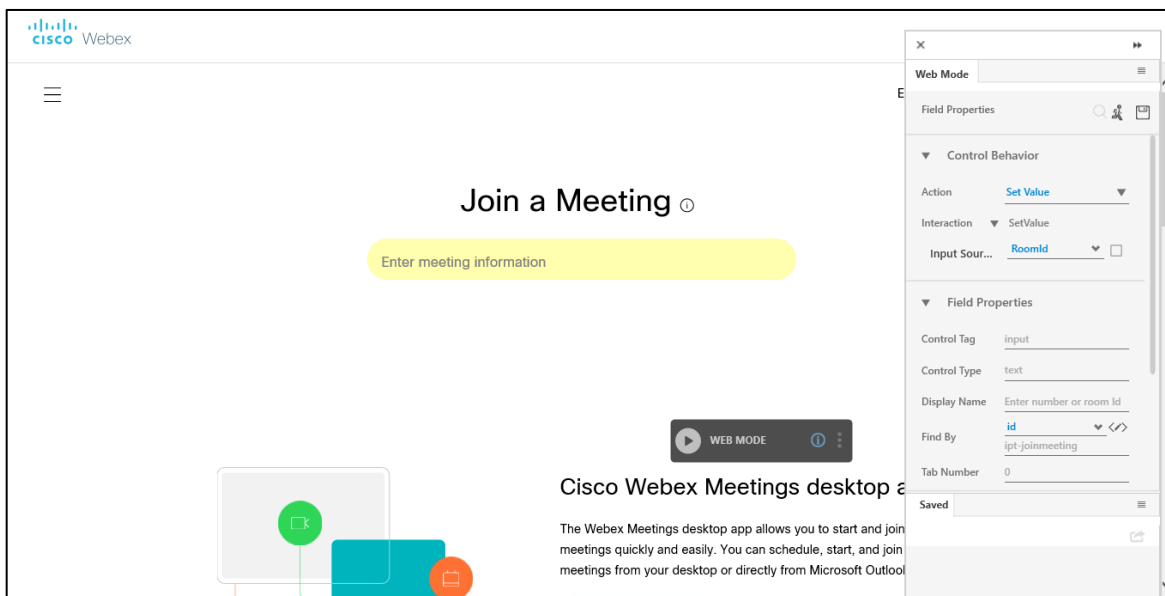
To handle the Cisco Webex popup window:

1. In the **Studio** menu, create a new process.
2. From the Canvas Tools panel, add the Application activity to the Flowchart designer on the Canvas

- In the Parameter bar, create an In arguments, **RoomId** and assign the required room id or the number required to join the Webex meeting.

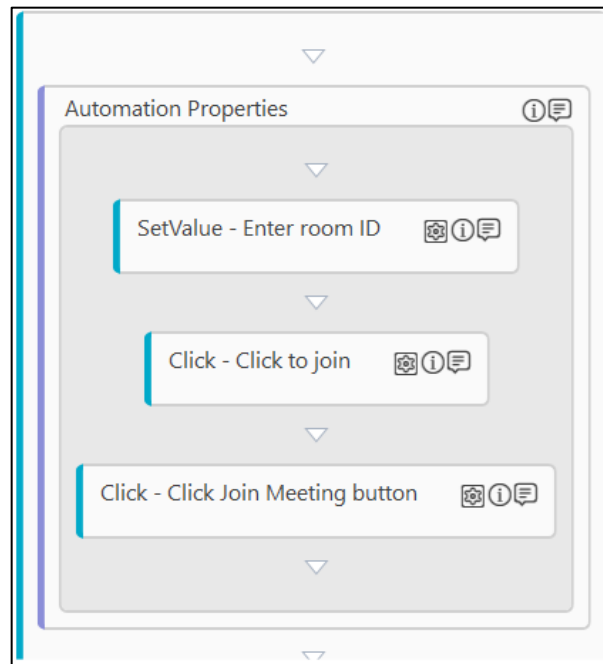
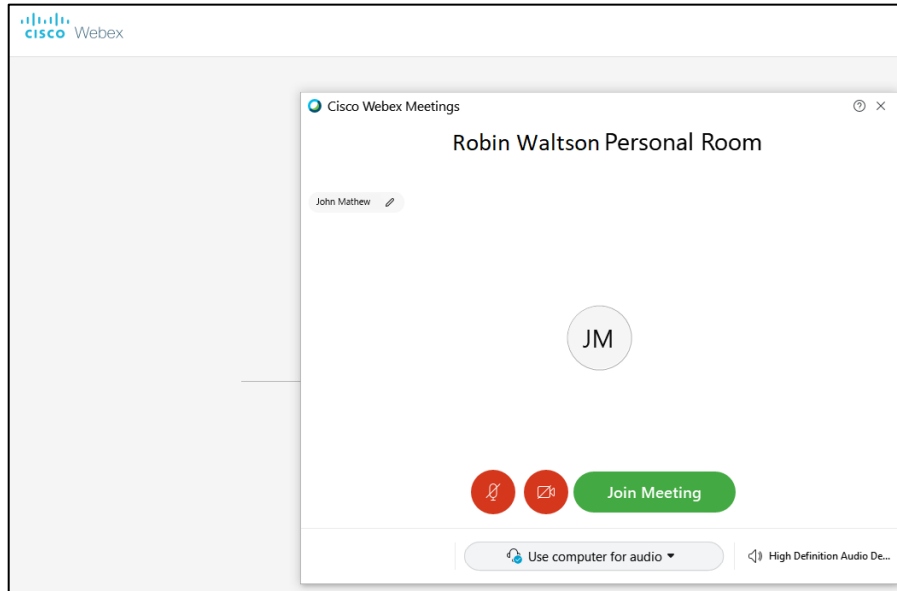


- In the Application Type list, select WebApps.
- In the **Select an Application** list, select the available web application that you configured.
- Click the (**Settings**) icon to launch the **WEB MODE** of the **Multimodal** interface. The configured Webex application page appears.
- Click the (**Play**) icon to capture the step of entering the meeting information such as the room Id or the required number to join the meeting. Hover over the **Enter meeting information** field and then click the highlighted area. The **Field Properties** panel of the **Web Mode** appears. The fields that are mandatory are highlighted with red box.



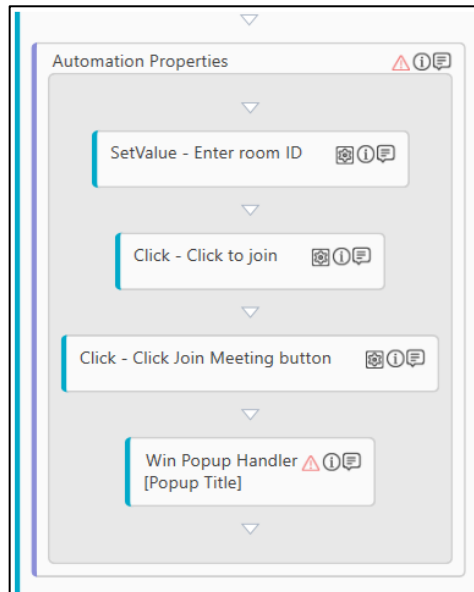
- In the **Action** list, select **Set Value**.
- In the **Input Source** list, select the **RoomId** argument created above.
- Enter a **Display Name**, and then click the (**Save**) icon to configure the details captured. The saved details start appearing in the **Saved** tab of **Field Properties** bar.

11. Click the **Play** button to capture the action of clicking the Join button that appears twice. The popup window appears after the second instance of the **Join** button is clicked.

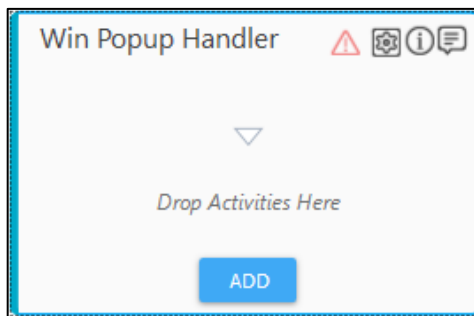


12. Open an instance of the Webex application with the popup window in the configured browser to capture the automation step of handling the popup window.

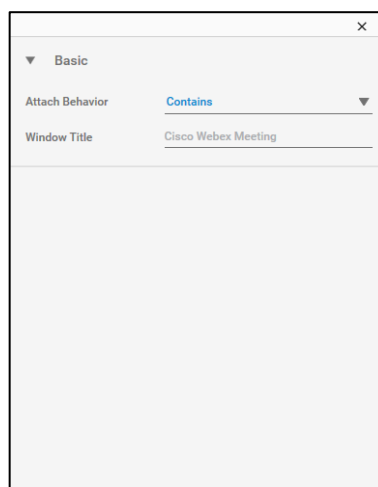
13. Add the **Win Popup Handler** activity below the second click interaction.



14. Double click the **Win Popup Handler** activity.

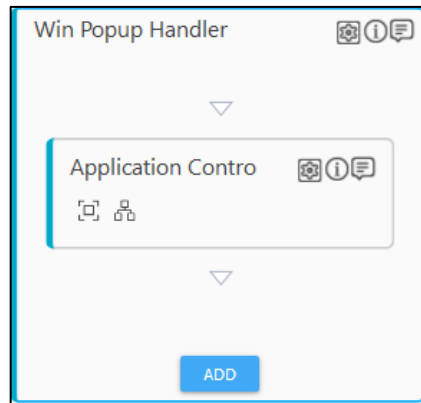



15. Click the  (**Settings**) icon. The configuration dialog box of the **Win Popup Handler** activity appears.

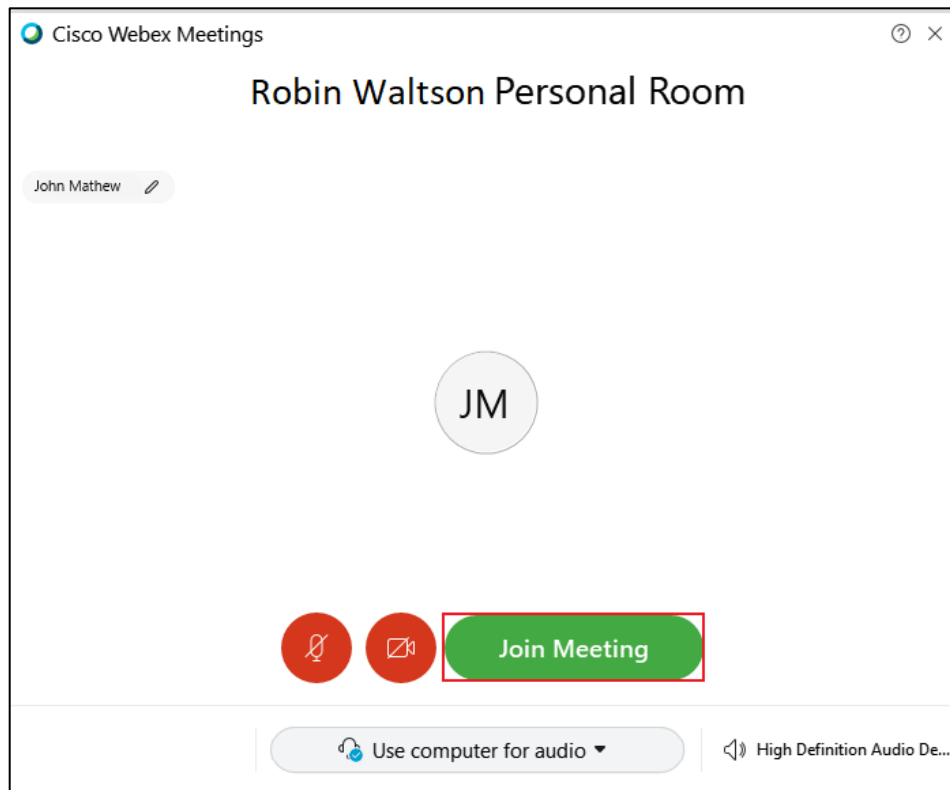


16. In the Attachment Behavior list, select Contains.

17. In the **Window Title** field, enter the title of the popup window, **Cisco Webex Meeting**. Close the configuration dialog box.
18. Click **ADD** to add the **Application Control** activity.



19. Click the  (**Field Configuration**) icon. The Webex popup window that you have kept open comes into focus.
20. Press the **Ctrl** key, hover over the window popup and simultaneously select the **Join Meeting** button.



21. Click the highlighted area. The configuration dialog box appears. The fields that are mandatory are highlighted with red box.

22. In the **Action** list, select **Click**.
23. In the **Display Name** enter a name of your choice and then close the configuration dialog box.
24. Save the process.
25. To view the output in Automation Studio, setup the environment and then perform test run. You can assign this process to a robot, if you want to execute this process outside Automation Studio.
26. Observe that the Webex application opens and the popup window getting clicked automatically to join the meeting.

9.6.24 Navigate To

It helps to navigate from existing URL of a web application to a different URL in the process workflow.

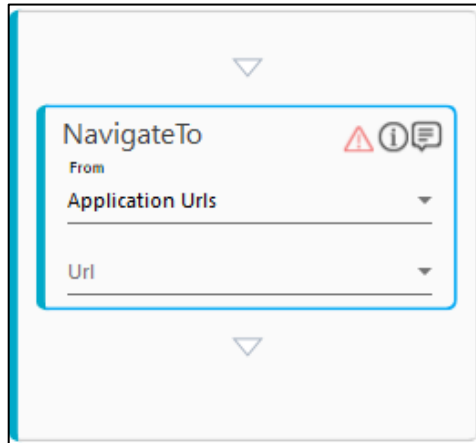
Note: This activity must be used inside an Application activity to establish a connection between Automation Studio and the the URLs; else the system displays an error.

Using Navigate To Activity

1. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
2. Drag the **Application** activity and drop on to the Flowchart designer on the Canvas
3. In the Application Type, select Webapps.
4. In the Select an Application list, select the indented application. The validation error symbol disappears when required inputs are provided.

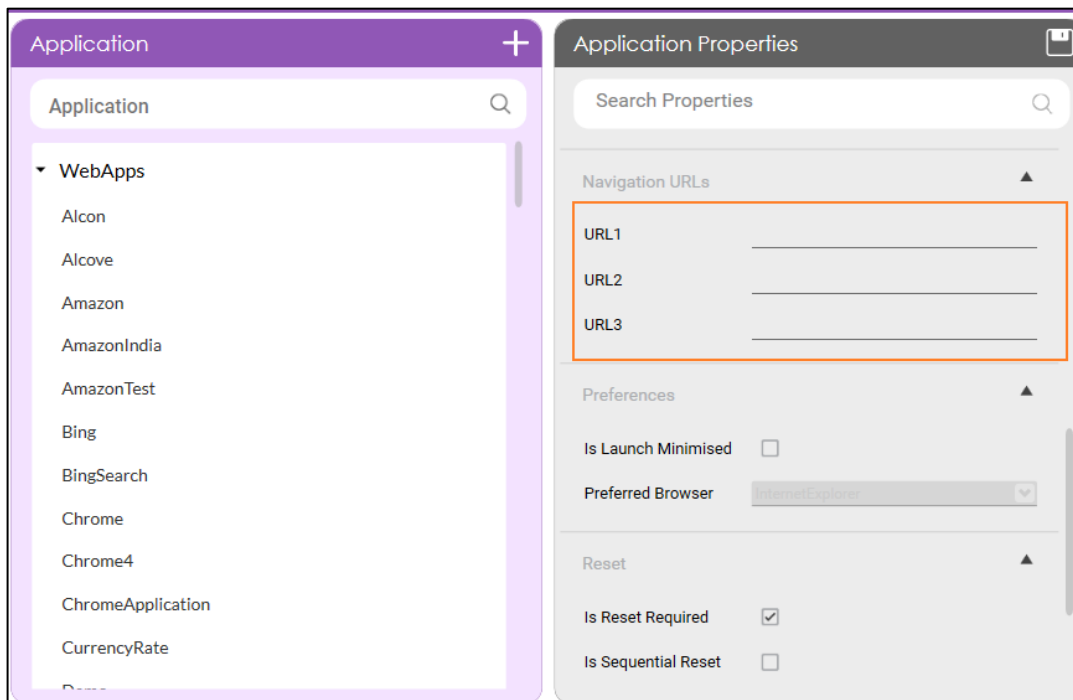
5. Double click the Application activity, drag the Navigate To activity and drop inside the Application activity. The URLs are configured either as part of the application properties or as a parameter in the process workflow. Click any of the following link to see the different ways of configuring the navigation URLs:


- Configure URL as application properties
- Configure URL as parameter

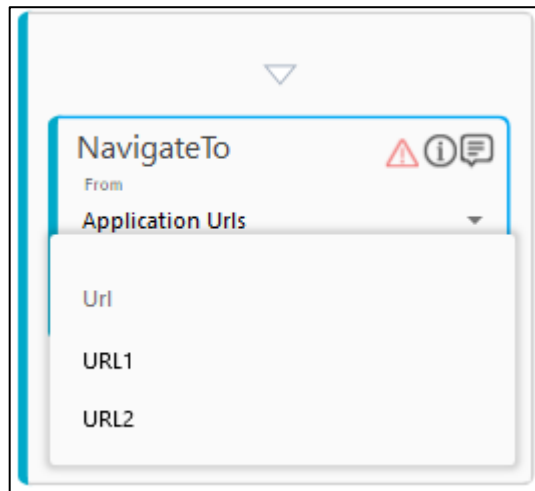


Configuring URL as Application Properties

1. In the **Admin** menu, expand the **WebApps** list in the **Applications** tab.
2. select the required application. The **Application Properties** pane of the configured application opens.

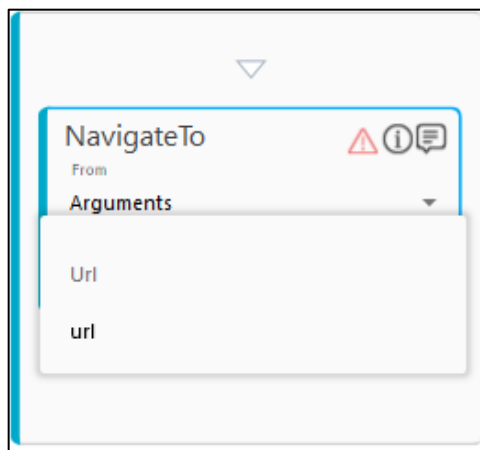


3. In the **URL1**, **URL2** and **URL 3** fields, enter the different URLs you want to navigate to in the process workflow. You enter URL in any one field or multiple fields as per your requirement.
4. Click the  (**Save Properties**) icon to save the application. The URLs are configured and starts appearing in the **Url** list of **Navigate To** activity.
5. Goto the **Studio** menu on the **Flowchart** designer area and then select **Application Urls** in the **From** list of the **Navigate To** activity.
6. In the **Url** list, select the configured URL.



Configure URL as Parameter

1. In the **From** list, select **Arguments**.
2. In the **Url** list, select the parameter holding the URL link you want to navigate to in the process workflow. You must define the parameter in the **Parameter** bar, along with the required URL, to make it appear for the selection.



The navigation to the desired URL is created.

Navigate To Activity Properties

The properties of the Navigate To activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as Navigate To . You can change the name as required.

9.6.25 Regex Validator

This activity allows you to validate the data that forms the input parameter in a process. It validates the required format of the data and helps in minimizing the errors, if any, during execution of the process.

The value stored corresponding to each variable is validated based on the type of regex validator configured. If the validation fails, configure the process to take the necessary actions.

By default, the Regex Validator activity supports e-mail and phone number (mobile numbers of India) validation. You can add more validations as per your business need by editing the RegexValidator.xml file. Follow the below steps to configure the validations other than default validations:

To configure a regex validation:

1. In the, client-tools > Automation Studio > configuration (or, %localappdata% > EdgeVerve > AutomationStudio > configuration, If you download/accessed Automation Studio from the Admin module) folder, locate RegexValidator.xml file.
2. Open **RegexValidaoatr.xml** file. Add **RegexDetails** tag and define a regex pattern. Add another tag, **RegexName** and define a desired name for the pattern.

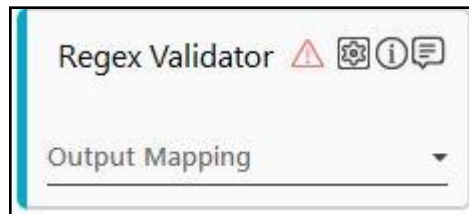
```
<?xml version="1.0"?>
- <ArrayOfRegexValidatorTypeModel xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
- <RegexValidatorTypeModel>
- <RegexDetails>[a-z0-9!#$%&'*+/,=?^_`{|}~]+(?:\.[a-z0-9!#$%&'*+/,=?^_`{|}~]+)*@(?:[a-z0-9](?:[a-z0-9]|)?)\.[a-z0-9](?:[a-z0-9]|)?)?>
- </RegexDetails>
- <RegexName>EmailID</RegexName>
- </RegexValidatorTypeModel>
- <RegexValidatorTypeModel>
- <RegexDetails>^((\+?)((0[ -]?)?(91)?)\d{12}|\d{10})$|(\d{5}([- ]*)\d{6}$))</RegexDetails>
- <RegexName>PhoneNumber</RegexName>
- </RegexValidatorTypeModel>
</ArrayOfRegexValidatorTypeModel>
```

3. Save the .xml file.

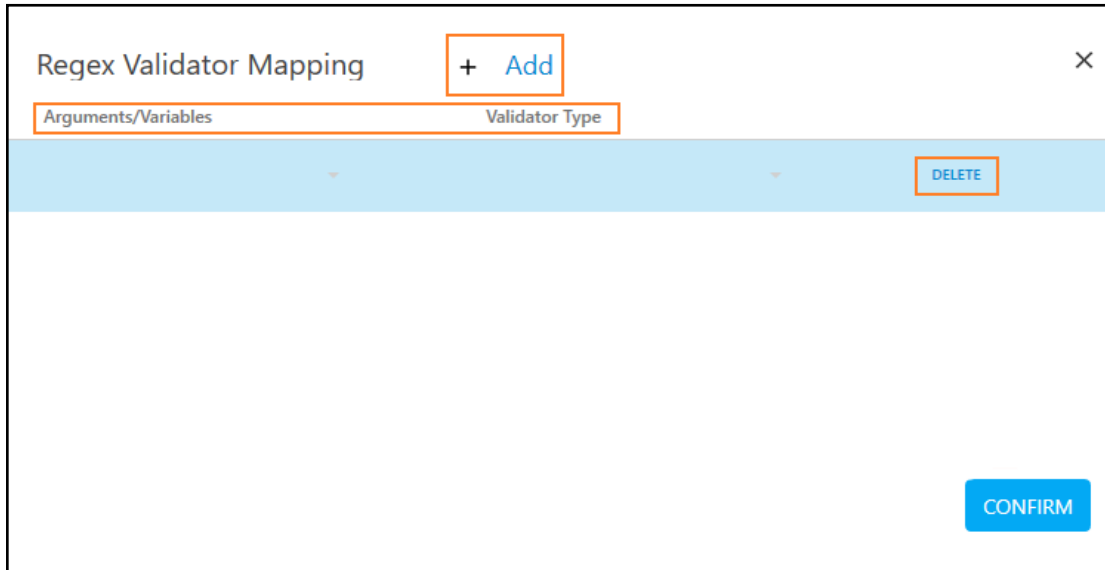
The new regex validation is configured.

Using Regex Validator Activity

1. In the Canvas Toolspane, click Process Components to expand the tool and view the associated activities.
2. Drag the **Regex Validator** activity and drop inside the **Application** activity.



- Click the  (**Settings**) icon. The **Regex Validator** Mapping dialog box appears.



- Click **Add** to add a row to select the parameters for mapping.
- In the **Arguments/Variables** drop-down list, select the parameter that you want to align with the required regex validator type for validation. You must define the parameter in the Parameter bar use this option.
- From the **Validator Type** drop-down list, select the validation you want to configure against the selected parameter. By default, **EmailID** and **PhoneNumber** is available.
- Click **CONFIRM**.
- You can click **DELETE** if you want to delete the mapping.
- In the **Output Mapping** list, select a parameter to store the status of the **Regex Validator** activity operation. You can use the validation result to take appropriate action to perform the subsequent action. To use this option, a parameter of type **Boolean** must be pre-defined in the **Parameter** bar.

Note: If you want to modify an existing pattern, open the RegexValidator.xml file and edit the required RegexDetails tag. Save the changes and reopen the activity for the changes to get reflected.

The Regex Validator activity with a default name is configured to perform the validation.

Regex Validator Properties

The properties of a Regex Validator activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as Regex Validator . You can change the name as required.
RegexOutputParam	The parameter configured to store the status of the Regex Validator activity.

9.6.26 Python Script Automation

AssistEdge RPA Studio supports running Python scripts as part of process automation. This section covers adding python script activity and steps to be followed in doing automation.

Prerequisite

You need to install Python (v3.6) on the client machines where the Engage application is running and also where Python scripts are developed. Configure the path of .exe file of Python as an Environment Variable- PYTHON_PATH.

Perform below steps on both the client machines where the Engage application is running and also where the Python scripts are developed:

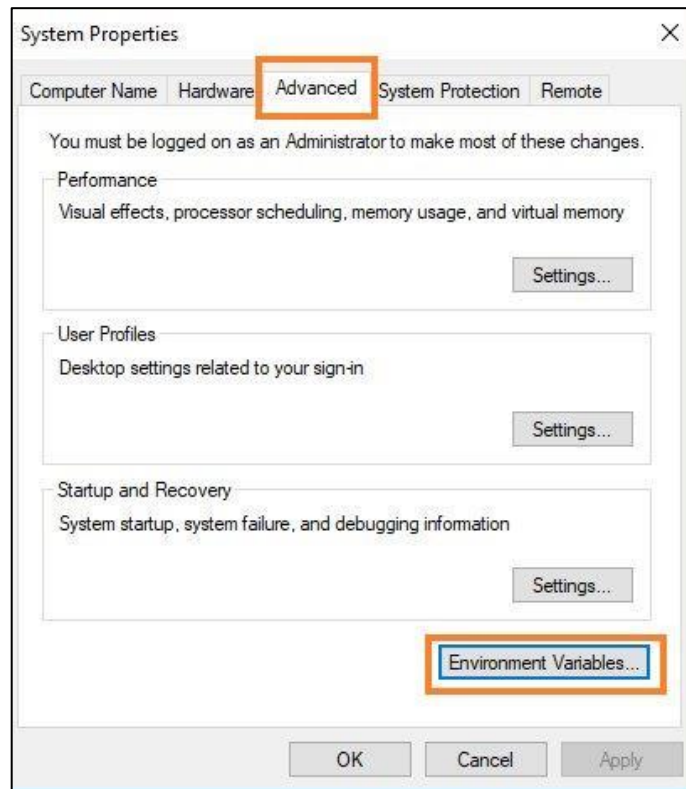
To configure path of .exe file of Python as Environment Variable:

1. In the **Start** menu, enter Edit the server environment variables.

Note:

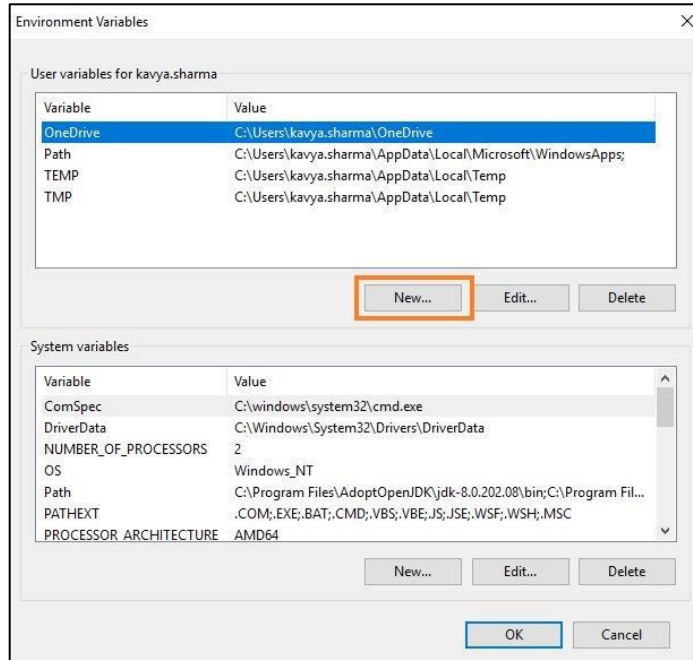
- You need Administrator rights to edit a server environment variable.
- If you do not have Administrator rights, use environments variable for your account.

2. Double click and open the displayed app.

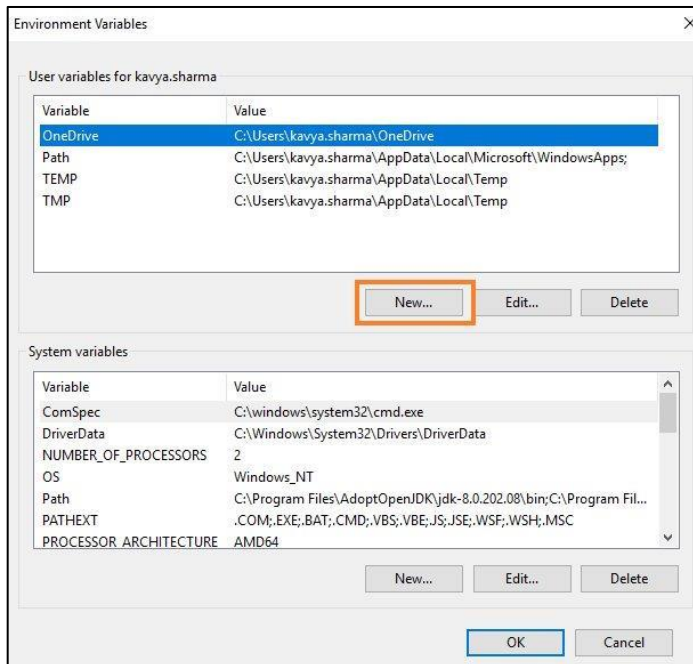


The Server Properties dialog box opens.

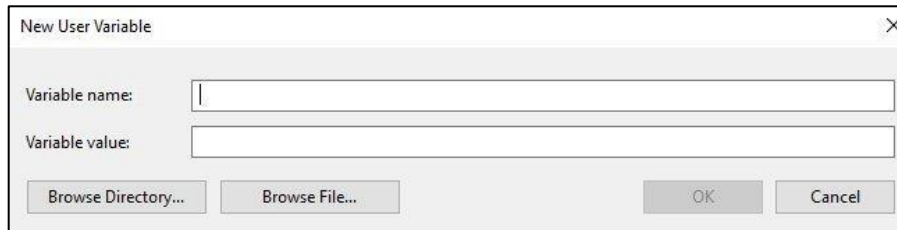
3. Click **Advanced** tab, and then click **Environment Variables...**



4. Click **OK**. The **Environment Variables** dialog box appears.



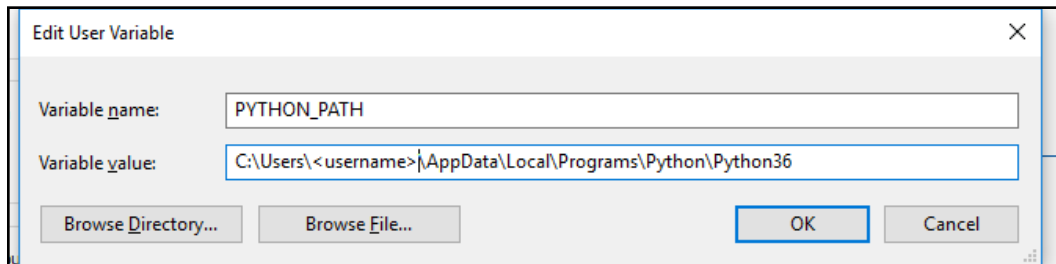
- Under **Server variables** section, click **New** to create new user variable.



The dialog box titled "New User Variable" contains two text input fields: "Variable name:" and "Variable value:". Below these fields are four buttons: "Browse Directory...", "Browse File...", "OK", and "Cancel".

The New User Variable dialog box opens.

- In the **Variable name** text box, enter `PYTHON_PATH`.
- In the **Variable values** text box, provide the path of .exe file of Python installed on the server. For example:

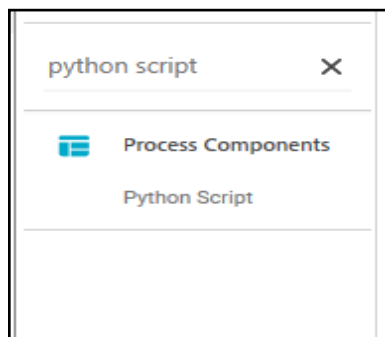


The dialog box titled "Edit User Variable" shows the "Variable name:" field containing "PYTHON_PATH" and the "Variable value:" field containing "C:\Users\<username>\AppData\Local\Programs\Python\Python36". The "OK" button is highlighted with a blue border.

The path of .exe file of Python is configured as Environment Variable.

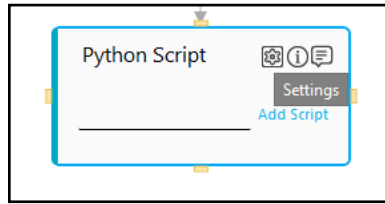
Automate using Python Script Activity

- Add** – drag and drop the python script activity to a process either by
 - Searching for it in **Canvas Tools** by typing the activity name i.e. Python Script.
 - Selecting the activity from under Process components.

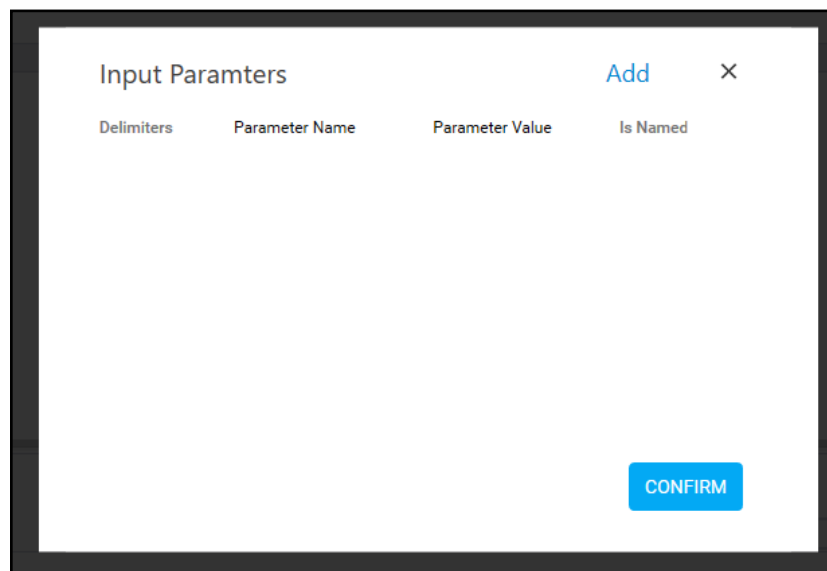


- To run a Python script – we need to pass the path of the script, inputs if any and store output if needed.

3. Passing the path where the script is present – this can be done by clicking on the Add Script link and selecting the python script. We cannot use an argument which has path details and pass it as input to the Add Script textbox or type the path details. It has to be done only via Add Script link.



4. Adding inputs if any –
 - a. First declare inputs to be passed as In Arguments in argument pane and assign values to it or if the values are populated by other activities then mark it as In/Out Arguments. Please read the help file which is part of activity to know the type of arguments supported
 - b. Pass the arguments declared in argument pane to the activity by clicking settings button



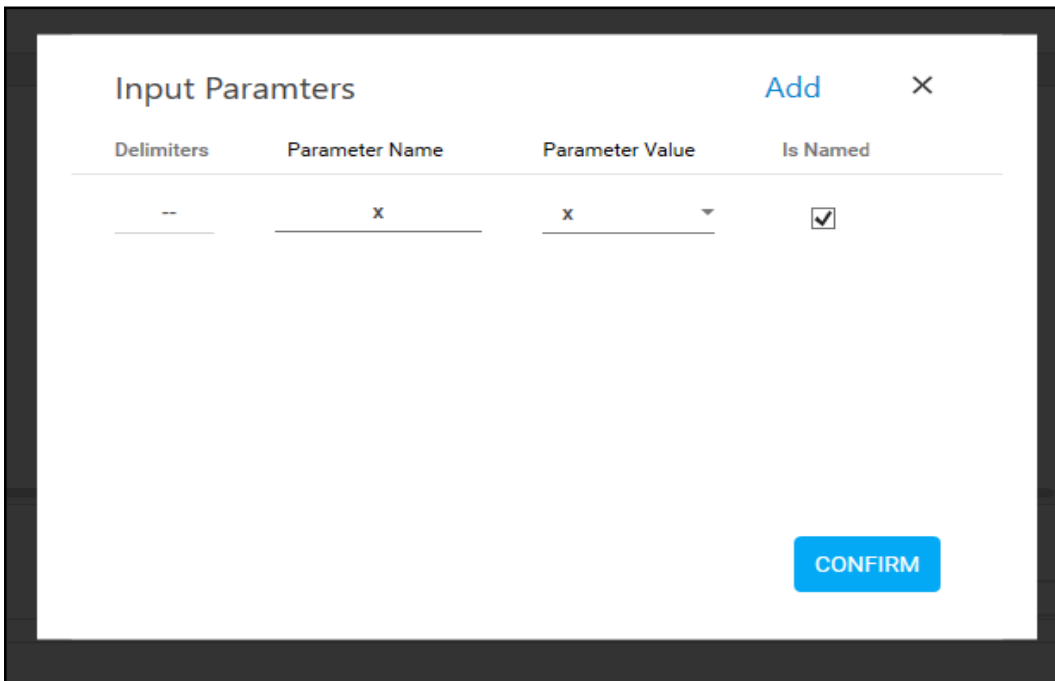
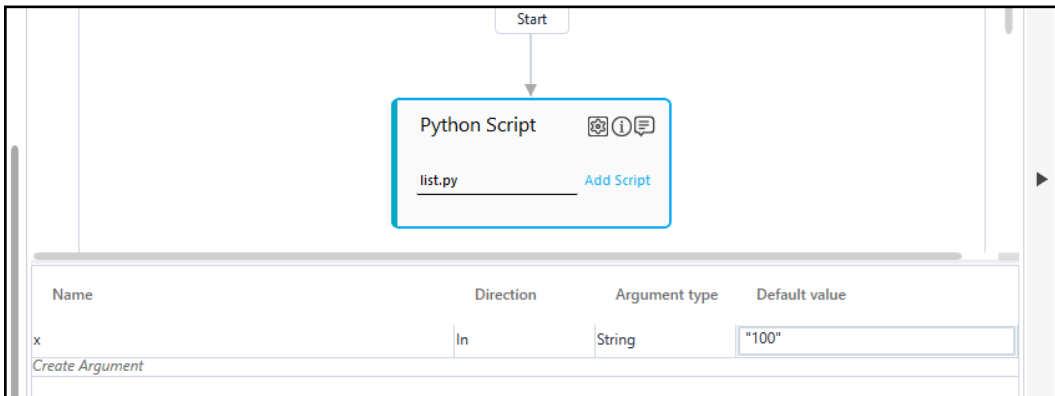
- c. Click on Add

Delimiters	Parameter Name	Parameter Value	Is Named
--		Select Arg ▾	<input checked="" type="checkbox"/>

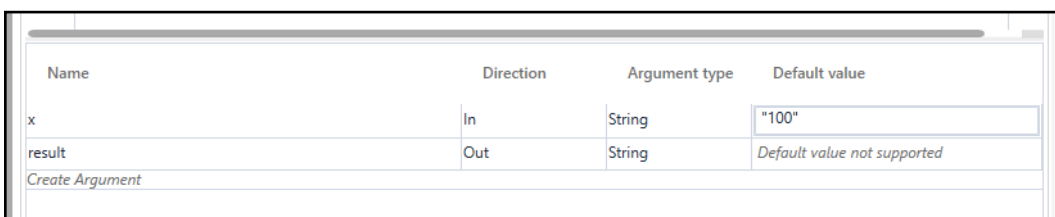
CONFIRM

- d. Key in the parameter name expected by the script.
- e. Select argument whose value is to be passed from the Parameter value drop-down. This list all the argument (InOut Arguments only) names declared in the argument pane. The argument value will be bound to it when script is run.
- f. Delimiters are needed if the script run requires named parameters. So, accordingly select the delimiter.

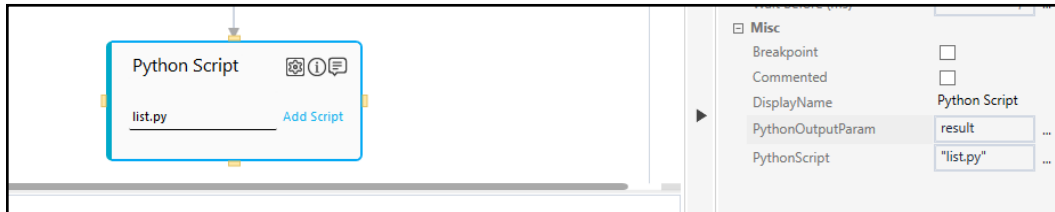
- g. Select **Is Named** if the parameters required are to be named else uncheck it, in case delimiters can be ignored. For example, if the script name is list.py and expects an input named parameter as "x" the following screen shots show how it is declared



- h. Click confirm once everything is declared.
- 5. Adding outputs, if any – If the script returns any output and we need to store it in an argument for any further use, like pass it to another activity, follow the following steps –
 - a. First declare an out argument in the argument pane like the result declared here in the screen shot



- b. Assign the out argument to the activity by setting PythonOutputParam to the out argument declared in the argument pane as shown in the below screen shot



- c. When the activity is run the output from the activity will now be set to result argument



Saving and Uploading Python Script Activity

1. Saving workflow/process also upload the python script to the database.
2. Script is zipped when uploaded to the database.
3. When we replace the script in the saved activity, like browse and select a different script, the same row will be updated in the database. A new row will not be created.
4. When the process is loaded from the database the script is also downloaded.

Dependency Management

1. The activity will assume that all the dependencies of the main executing script will be in the same folder as the main script. For example – if the user browses to *D:\PythonScripts* and selects a python script *test.py* and saves the process, the activity zips the entire D:\PythonScripts and uploads all the contents in it. The assumption here is that the script test.py has all its dependencies present in the folder. So, for correct execution of the script from any machine, place all the dependencies in the same folder as the main executing script.
2. If the script is placed in root folder itself, for e.g. in D drive then the activity is not able to take into account of all the dependencies. In this case, it will upload the main script only.

Advanced

1. When a python script is selected for execution by the activity, the folder where the script is present is zipped and is placed in UserInstanceDirectory path configured in Studio app.config. For example, if UserInstanceDirectory is, C:\Users\\AppData\Local\EdgeVerve\AutomationStudio, then the scripts will be found in, C:\Users\\AppData\Local\EdgeVerve\AutomationStudio\ProtonFiles\PyScripts folder. Every script folder zipped will have a guid for name e.g. 092722c8-5321-4b06-bd9e-f72c83793225.zip This guid is unique for an activity. The same zipped folder content will be replaced every time when we browse and select a different script for the same activity.
2. For some reason the zipped files are deleted in the folder mentioned in above point, then the script execution will fail throwing file(<guide>.zip) not found error.

The zipped folders are created every time when the workflow having python activity is downloaded.

9.6.27 Disable Retry

You can use the Disable Retry activity in the load generator process workflow to disable the retry configuration of a trigger. A trigger is created in Control Tower while generating the automation request or through the Trigger activity in the load generator process.

It lets you avoid retry of an action which does not remain valid if certain steps in the process workflow have been executed in the transaction.

Using Disable Retry Activity

1. In the Canvas Tools pane, click **Process Components** to expand the tool and view the associated activities.
2. Drag the **Disable Retry** activity and drop on to the Flowchart designer on the Canvas

3. In the **Step** field, enter the desired name of the action to be disabled.
4. In the **Step Details** field, enter a description of the action to be disabled.

The Disable Retry activity with default name is created. The Step Details text is viewed in Control Tower and the Step text is viewed in the logs that provides contextual information for monitoring as to why the action was disabled. Both the fields are optional and information only fields.

Disable Retry Properties

The properties of Disable Retry activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Disable Retry . You can change the name as required.

9.6.28 Launch and SignIn Application

It allows you to launch an application wherever required in the automation process workflow. Once an application is closed, you can use this activity to launch the application. Additionally, if sign-in is configured for the intended application, this activity automatically signs in to it.

Place this activity at the point where you want to launch the application before starting with the automation in that application.

Note:

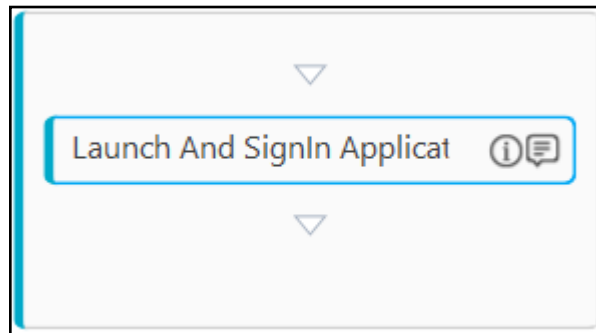
This activity must be used inside an Application activity or the system displays an error.

It does not relaunch and application if the application is already open.

In AE Engage, this activity creates a new tab with the launched application to dock.

Using Launch and SignIn Application Activity

1. Create a process automation workflow and add an **Application** activity.
2. Double click the Application activity, drag the Launch and SignIn Application activity and drop inside the Application activity.



The Launch and SignIn Application activity process with a default display name is created.

Launch and SignIn Application Properties

The properties of a Launch and SignIn Application activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.

Property Name	Usage
	By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as Launch and SignIn Application . You can change the name as required.

9.6.29 Close Application

It allows you to close the application wherever required in the automation process workflow. In Automation Studio, the application involved remain in the open state even after process execution, closing such application or the one that is not required in the further processing, saves the memory of the system increasing performance and efficiency.

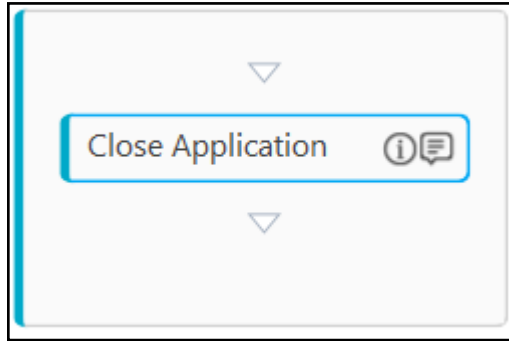
Place the activity at the point where you want to close the intended application after the pervious steps of automation are completed.

Note:

- This activity must be used inside an **Application** activity or the system displays an error.
- In AE Engage, this activity closes the application and the associated tab.

Using Close Application Activity

1. Create a process automation workflow and add an **Application** activity.
2. Double click the Application activity, drag the Close Application activity and drop inside the Application activity.



The Close Application activity process with a default display name is created.

Close Application Properties

The properties of a Close Application activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Description
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity.</p> <p>If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer. By default, the name is set as Close Application . You can change the name as required.

9.7 Complex Iterators

Complex Iterators automate the process to traverse through the complex list of data stored in different formats such as Excel, data table and others. It lets you process each element from the list with the help of a loop counter that controls the iterations of a loop.

Benefits

Following are the benefits of using the Complex Iterators - Advanced Loop:

- Easily manage complex processes which includes repetitive tasks to extract data or update any records.
- One-click configuration for different file formats.
- Preview, filter, and sort the processed data based on your business requirement.

9.7.1 Advanced Loop

It helps you to automate the task of looping over files such as Excel, CSV/Text and data table. It is a powerful automation tool that enables you to loop through numerous records saved in the file in just a few steps. Looping of a source file is typically required to execute a condition or a statement upto a desired number of times. This helps in executing actions like fetching data and updating records.

Prerequisite

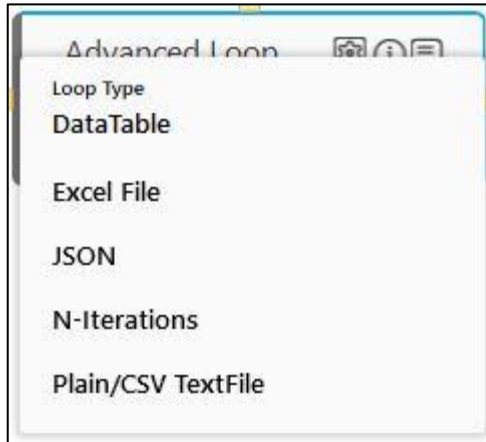
Microsoft Access Database Engine (V 2010 and above) must be installed on the system. It is a free software and can be downloaded from the Microsoft website.

Using Advanced Loop

1. In the Canvas Tools pane, click **Complex Iterators** to expand the tool and view the associated activities.
2. Drag the **Advanced Loop** activity and drop on to the Flowchart designer on the Canvas



3. In the **Loop Type** list, select the type of source to loop over. Available options are - **DataTable**, **Excel File**, **JSON**, **N-Iterations**, and **Plain/CSV TextFile**.




You can perform the following set ups to successfully configure the looping operation:

- [Source Configuration](#)
- [Column Mapping](#)
- [Preview and Filter](#)

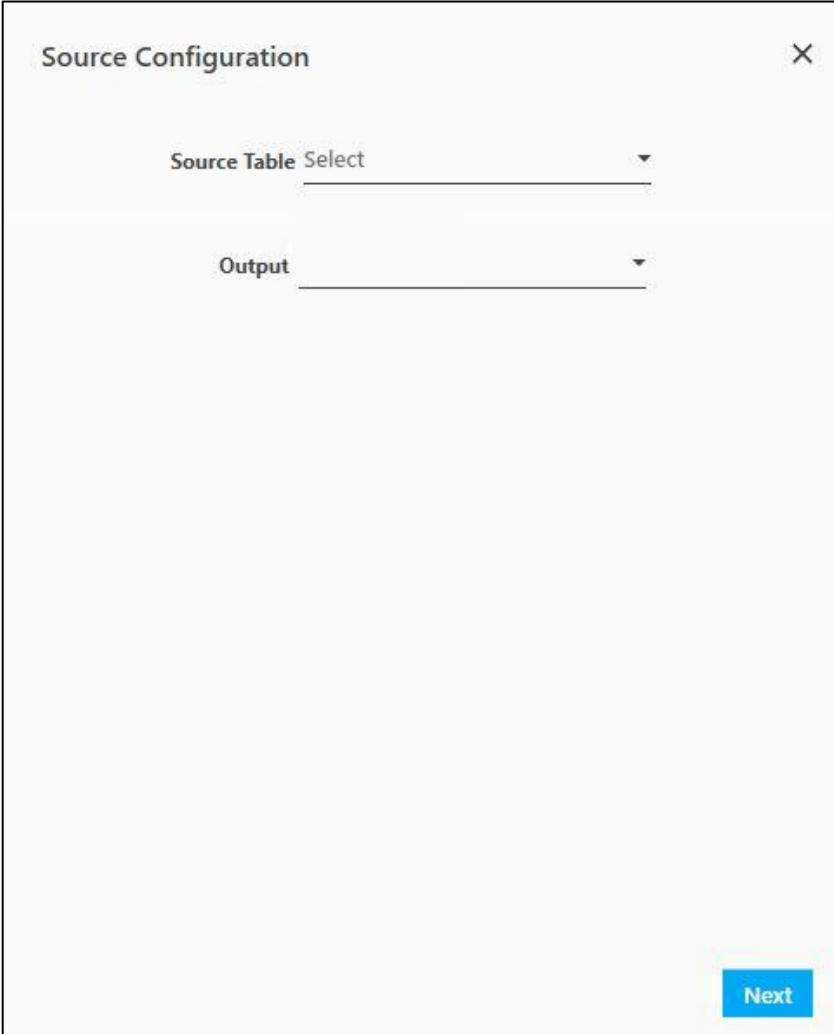
Source Configuration

This lets you configure the input and output data as per the selected Loop Type.

Click the  (Settings) icon to specify configurations related to the source file. The Source Configuration dialog box appears. Click any of the links below to know the detailed steps regarding each source type.

- [DataTable](#) - This option is used to read a data table and loop over its data. The **Preview** pane remains empty for a DataTable loop type whose data is available only at the runtime and not at design time.
- [Excel File](#) - This option is used to read an excel file and loop over its data.
- [JSON](#) - This option is used to loop over JSON file and iterate over JSON array and object.
- [N-Iterations](#) - This option is used to set the loop for a fixed number of times.
- [Plain/CSV TextFile](#) - This option is used to read text or CSV file and loop over its data.

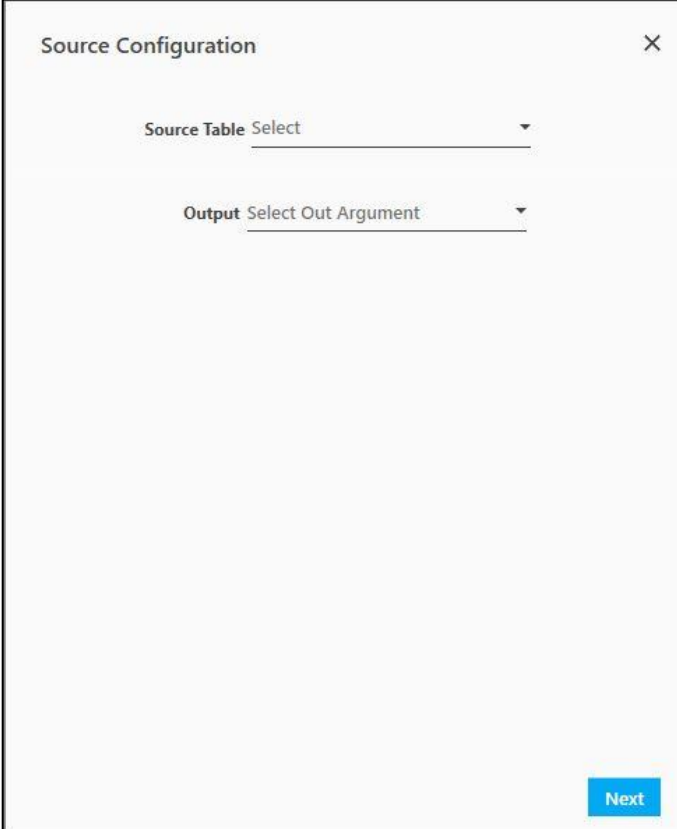
DataTable



The image shows a dialog box titled "Source Configuration" with a close button (X) in the top right corner. Inside the dialog, there are two dropdown menus. The first is labeled "Source Table" and has a "Select" button to its right. The second is labeled "Output" and also has a dropdown arrow to its right. At the bottom right of the dialog, there is a blue button labeled "Next".

- a. In the **Source Table** list, select the parameter holding the required data table. It can also be an output of another activity or process workflow which provides data table object like **Outlook Search E-Mail** activity. You must define the parameter in the **Parameter** bar to use this option.
- b. In the **Output** list, select an **Out** parameter of type **System.Data.DataRowView**. It is used to access the data at run time. You must define the parameter in the **Parameter** bar to use this option.
- c. Click **Next**. The **Column Mapping** dialog box appears.

Excel File



Source Configuration

Source Table Select

Output Select Out Argument

Next

- a. In the **Source Filepath** list, select the parameter holding the excel file path along with the file name and file extension. For example, D:\Demo\Advancedloop.xlsx . You must define the parameter in the **Parameter** bar to make it available for the selection.
- b. Select the **Has Header** check box to indicate if the first row of the excel file is the header.
- c. In the **Output** list, select an **Out** parameter of type **System.Data.DataRowView**. It is used to access the data at run time. You must define the parameter in the **Parameter** bar to use this option.
- d. Click **Next**. The **Column Mapping** dialog box appears.

JSON

Source Configuration

Input as Filepath

JSON Input Select

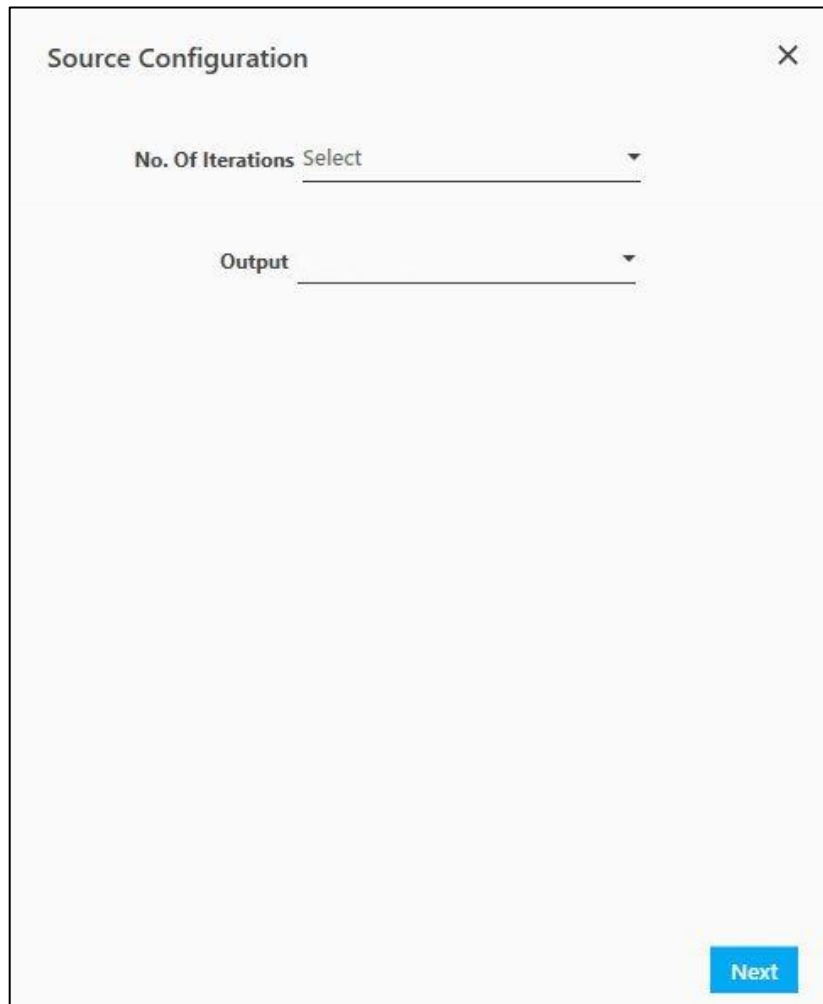
JSON Path Select

Output

Next

- a. Select the **Input as Filepath** check box, if you want provide a JSON file to loop over.
- b. In the **JSON Input** list, select the parameter holding the file path along with the file name and file extension of the JSON file if the **Input as Filepath** check box is selected, else, enter the JSON string. Example of JSON filepath is D:\Demo\Advancedloop.json, while a JSON string contains an array of values or an object. You must define the parameter in the **Parameter** bar to make the file path available for the selection.
- c. In the **JSON Path** list, select the parameter holding the JSONPath (the query language for JSON structure) on which iteration needs to be done. You must define the parameter in the **Parameter** bar to make it available for the selection.
- d. In the **Output** list, select an **Out** parameter of type **System.Data.DataRowView**. It is used to access the data at run time. You must define the parameter in the **Parameter** bar to use this option.
- e. Click **Next**. The **Column Mapping** dialog box appears.

N-Iterations



The image shows a dialog box titled "Source Configuration" with a close button (X) in the top right corner. Inside the dialog, there are two dropdown menus. The first is labeled "No. Of Iterations" and has the word "Select" to its right. The second is labeled "Output" and has a small downward arrow to its right. At the bottom right of the dialog, there is a blue button labeled "Next".

- a. In the **No. of Iterations** list, select the parameter holding the value of the number of times the loop must be execute. You must define the parameter in the **Parameter** bar to use this option.
- b. In the **Output** list, select an **Out** parameter of type **System.Data.DataRowView**. It is used to access the data at run time. You must define the parameter in the **Parameter** bar to use this option.
- c. Click **Next**. The **Column Mapping** dialog box appears.

Plain/CSV TextFile

Source Configuration

Source Filepath Select

Has Header

Delimiter Select

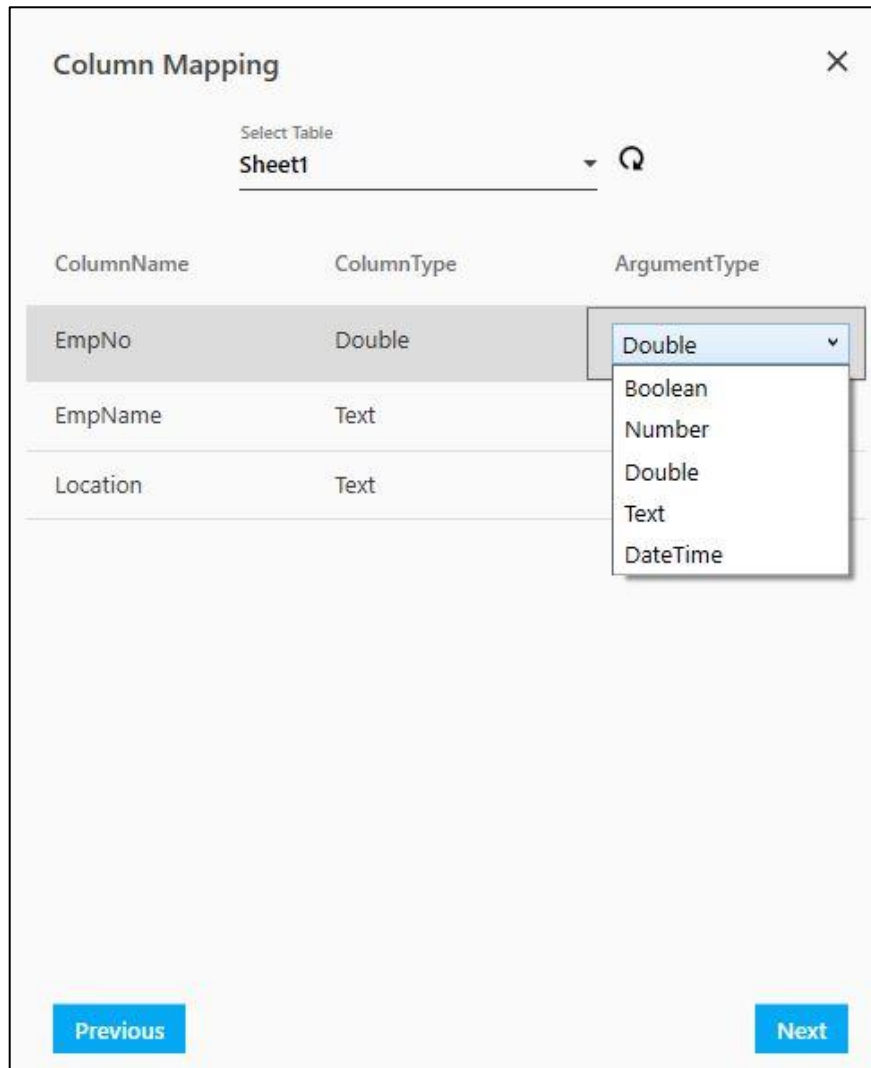
Output


Next

- a. In the **Source Filepath** list, select the parameter holding the CSV file path along with the file name and file extension. For example, D:\Demo\Advancedloop.csv. You must define the parameter in the **Parameter** bar to make it available for the selection.
- b. Select the **Has Header** check box to indicate if the first row of the CSV file is the header.
- c. In the **Delimiter** list, select the type of field separator used in the CSV file. Available options are - **Comma, Tab, Semicolon, Colon, and SingleSpace**.
- d. In the **Output** list, select an **Out** parameter of type **System.Data.DataRowView**. It is used to access the data at run time. You must define the parameter in the **Parameter** bar to use this option.
- e. Click **Next**. The **Column Mapping** dialog box appears.

Column Mapping

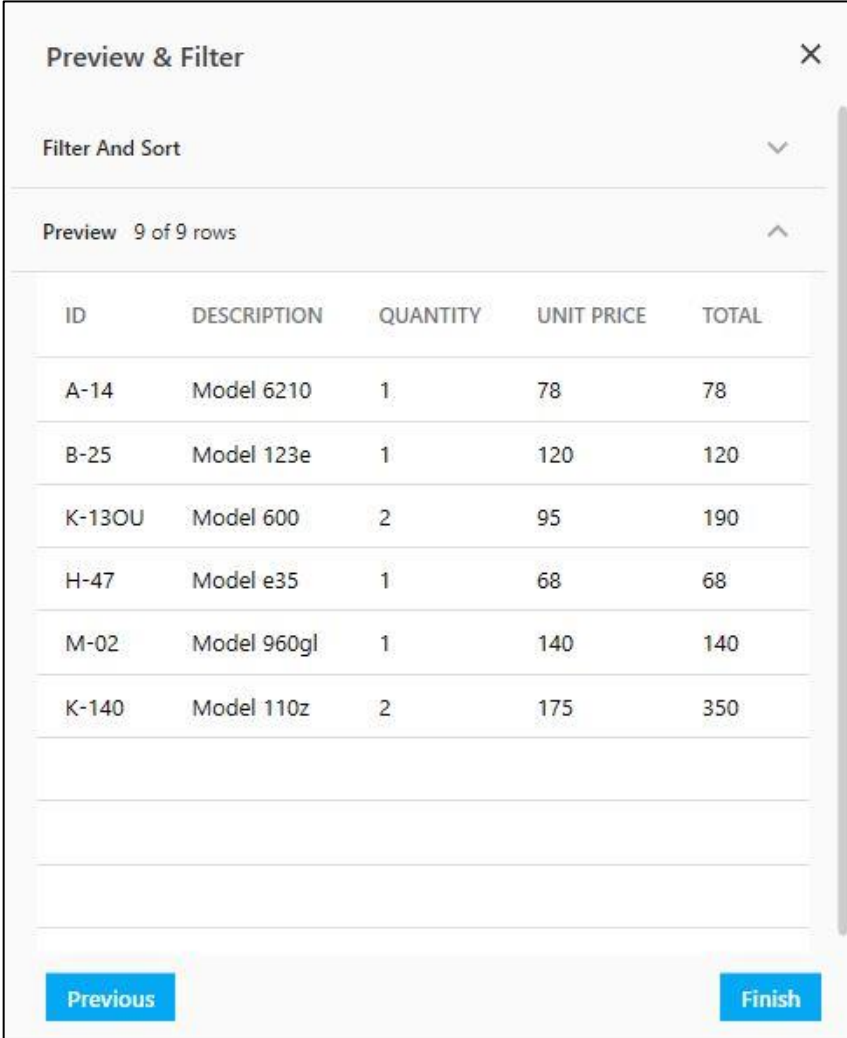
This lets you align the data of the source file with its correct data type by loading its schema. In the Column Mapping dialog box perform the following mapping:



- In the **Select Table** list, select the required table. The table schema appears and the column data gets mapped with its data type. You can click the  (Reload Schema) icon to reload or reset the column mapping.
- Click the **ArgumentType** column of the respective data to realign the required data type if the fetched data type is not correct.
- Click **Next**. The **Preview & Filter** dialog box appears.
- Click **Previous** to return to the **Column Mapping** dialog box.

Preview & Filter



It displays the preview of the data of the source file. You can apply filters and define the sorting criteria over the source data for faster processing.



ID	DESCRIPTION	QUANTITY	UNIT PRICE	TOTAL
A-14	Model 6210	1	78	78
B-25	Model 123e	1	120	120
K-13OU	Model 600	2	95	190
H-47	Model e35	1	68	68
M-02	Model 960gl	1	140	140
K-140	Model 110z	2	175	350

Preview Pane

You can view the total number of rows and the data available in the source file in the Preview pane. This is the data over which filter and sorting can be applied. A maximum of 100 rows can be viewed in the Preview pane.

Select any cell and use the arrow keys to navigate up, down, left or right in the preview table. Use the  (Collapse) and  (Expand) icon to hide or view the pane.

Filter & Sort

To filter and sort the data for desired output and for faster processing, you can define filter that is applied to the data before looping it.

Preview & Filter
✕

Filter And Sort
^

Filter: Expression here
⊕ ⊖

Expression ID
Field Name
Operator
Argument Name

Sort By: DESCRIPTION ▾
ASC ▾



No. of Rows: Select Int Argument ▾

Execute

Preview 9 of 9 rows
^

ID	DESCRIPTION	QUANTITY	UNIT PRICE	TOTAL
A-14	Model 6210	1	78	78
B-25	Model 123e	1	120	120
K-13OU	Model 600	2	95	190

Previous
Finish

- a. In the **Filter and Sort** pane, click the  (Expand) icon to expand the pane.
- b. Click the  (**Add Filter Expression**) icon to auto generate an **Expression ID** used to form the query expression for applying the filter. You can use the OR and the AND Boolean operators along with parenthesis to form complex queries. Only OR and AND operators are supported . By default, the AND operator is added with the expression Id. For the respective added expression:
 - i. . In the **Field Name** column, select the column name that you want to include in the expression Id. If the source data has a header, the header name is displayed; else, F1-Fn field names are available for selection.
- c. In the **Operator** column, select the criteria to match data based on the expression id applied. Available options are - **Equal, Not Equal, Like, =, !=, >, >=, < and ,<=**. See [Expression Forming Tips](#) to know more about forming the expression query.
- d. In the **Argument Name** column, select the parameter holding the value to map with the selected **Field Name**. The

argument type of the **ArgumentName** parameter and the **Field Name** parameter must be same. You must define the parameter and its value in the **Parameter** bar to use this option.

- e. Repeat step b through step iii to add more filter expression to form complex queries.

Preview & Filter ✕

Filter And Sort ^

Filter: ⊕ ⊖

Expression ID	Field Name	Operator	Argument Name
expression1	<input type="text" value="Field Name"/>	<input type="text" value="Operators"/>	<input type="text" value="Arguments"/>

Sort By:

No. of Rows:

Execute

Preview 9 of 9 rows ^

ID	DESCRIPTION	QUANTITY	UNIT PRICE	TOTAL
A-14	Model 6210	1	78	78
B-25	Model 123e	1	120	120

Previous
Finish

- f. In the **Sort By** list, select the column name to sort the data either in ascending or descending order. Available options are - **ASC** and **DSC**.
- g. In the **No. of Rows** list, select the parameter holding the value of the number of rows of **Int32** type to limit the number of rows of the source data to be looped over.
- h. Click **Execute** to view the short listed data as per the filter expression set.
- i. Click **Finish** to apply the configured filter and sorting criteria. You are taken back to the **Studio** menu.

The Advanced Loop activity is created with a default display name.

Expression Forming Tips

Wildcard Characters:

Use * or % wildcard at the beginning or at the end of the value set in the parameter defined for selection in the ArgumentName option. For example - *value, value* or *value*. You can not use the wildcard in the middle of the value. For example, va*lue is not supported.

Operator:

For String type FieldName, you can use wildcards with the Like operator. Like operator can be used to include only values that match a pattern with wildcards.

Examples of the wildcard expression used with the value:

- **j*** - the values starting with j.
- **%jo%** - the values that contain jo.
- **"j*"** // values that don't start with "j"
- If the value for a **Like** operator contains any of these special characters such as *, %, [, or], the characters must be placed within parenthesis. For example, **[*]**, **[%]**, **[[]** or **[][]**.
- **[*]*** - the values starting with *.
- **[[]*** - the values starting with [.

Advanced Loop Properties

The properties of an Advanced Loop activity are listed in the following table and can be edited in the Property grid on the right panel.

Property Name	Description
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity.</p> <p>If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.

Property Name	Description
Misc	
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	<p>Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.</p>
DisplayName	<p>The display name of the activity in the flowchart designer. By default, the name is set as Advanced Loop. You can change the name as required.</p>

9.7.2 Preview & Filter

Preview header displays the total number of rows in the document, after applying the filter and the total number of rows displayed on the screen. A Maximum of 100 rows is displayed on the preview screen. This screen displays the preview of data and is used to apply filters on the data.

To navigate horizontally in the table, select any cell and use arrow keys (left & right) to navigate left and right. Click on the "Collapse" icon to hide the preview pane.

Roll No	Name	Gender	IsHosteller
1	ADAM	M	True
2	AMELIA	F	False
3	DYLAN	M	True
4	EMILY	F	True
5	GEORGE	M	True
6	JOSHUA	M	True
7	KATIE	F	False
8	LIAM	M	True
9	LUCY	F	False
10	MICHAEL	M	True

Filter and Sort pane are used to apply a filter to data before looping over it. This results in a faster process as the filter is applied before looping over it.

There are three options in this pane.

- Filter – Create and apply filters on data using filter query generator.
- Sort By – Sort the data based on a single column (Ascending or Descending).
- No. Of Rows – Limit the number of rows that are fetched from the source.

Preview & Filter ✕

Filter And Sort ^

Filter: ⊕ ⊖

Expression ID	Field Name	Operator	Argument Name
<p>Sort By: Field Name ASC ▼</p>			
<p>No. of Rows: Select Int Argument ▼</p>			

Execute

Preview 15 of 15 rows ▼

Previous

Finish

- Filter:

Click on Add icon to **⊕** add a new filter and Delete **⊖** icon to delete the last added filter.

4. Every new filter is added as an expression in the filter textbox.
5. "AND" and "OR" operators are only used along with parentheses to make complex queries. By default, the "AND" operator is added with expressions.
6. Each filter has an auto-generated expression ID, use only these in the filter textbox.
7. Any other keyword or character other than expression ID, "AND", "OR" and parentheses throws a syntax error.
8. To apply a filter, click on the "Execute" button.

Scenario: Find the students who are male and hosteller, and their roll number is > 3.

Solution:

1. Build an expression where "rollNo" is an argument of type Int32.
2. "gender" is an argument of type String
3. "hosteler" is an argument of type Boolean.

Filter And Sort ^

Filter: ⊕ ⊖

Expression ID	Field Name	Operator	Argument Name
expression1	Roll No	>=	rollNo
expression2	Gender	Equals	gender
expression3	IsHosteller	=	hosteler

Sort By:

No. of Rows:

Execute

NOTE: For string type column, use wildcards with "Like" operator, which is used to include only values that match a pattern.

Similarly, to find all the students who are either male or hosteller, filter expression must be Expression2 OR expression3.

Wildcard characters:

- Use Characters * or %, at the beginning of a pattern as, "*value", at the end "value*", or at both "**value**".
- The wildcard in the middle of a pattern "va*lue" is **not allowed**.
-

NOTE: For string type column we can use wild cards with "Like" operator.

- **Operator LIKE** is used to include only values that match a pattern with wildcards. **Wildcard** character is * or %, it can be at the beginning of a pattern "*value", at the end "value*", or at both "**value**". Wildcard in the middle of a pattern "va*lue" is **not allowed**.

- "j*" // values that start with "j"
- "%jo%" // values that contain "jo"
- "j*" // values that don't start with "j"
- If a pattern in a LIKE clause contains any of these special characters * % [], those characters must be escaped in brackets [] like this [*], [%], [[] or [[]].
- "[*]" // values that starts with "*"
- "[[]]" // values that starts with "["

9.7.3 Step-By-Step Guide to Use Advanced Loop to Evaluate the Price of Items in Invoice

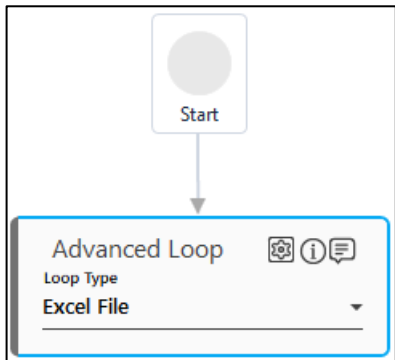
Let's create an example to find the list of Ids of items whose total price in the invoice excel file is greater than 100 but less than 300. We also want to sort the short-listed items based on the alphabetical ascending order of their respective Ids.


1. Create a process.
2. In the Parameter bar, create an **In** argument, **SourceID** of **Argument Type - String** and set the **Default Value** as **"D:\Sample Data\ItemsTable_ChronicalSolutions.xlsx"**. The invoice excel file is saved in the **D** drive of the local machine.
3. Create **In** arguments, **TotalPrice** and **RowCount** of **Argument Type - Int32** and set the Default Value as **100, 300** and **8** respectively. This lets us form the filter expression and limit the number of rows to be looped over.
4. Create an **Out** argument **OutputData** of **Argument Type - System.Data.DataRowView** to access and store the sorted data at the run time.

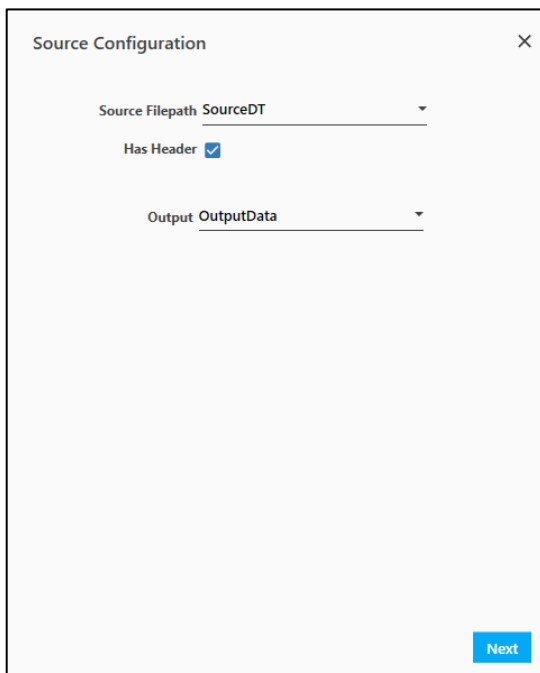
Name	Direction	Argument type	Default value
SourceDT	In	String	"D:\Sample Data\ItemsTable_ChronicalSolutions.xlsx"
OutputData	Out	DataRowView	Default value not supported
TotalPriceLL	In	Int32	100
RowCount	In	Int32	8
TotalPriceUL	In	Int32	300

5. From the Canvas Tools panel, add the Advanced Loop activity to the Flowchart designer on the Canvas.

6. In the **Loop Type** list, select **Excel File**.

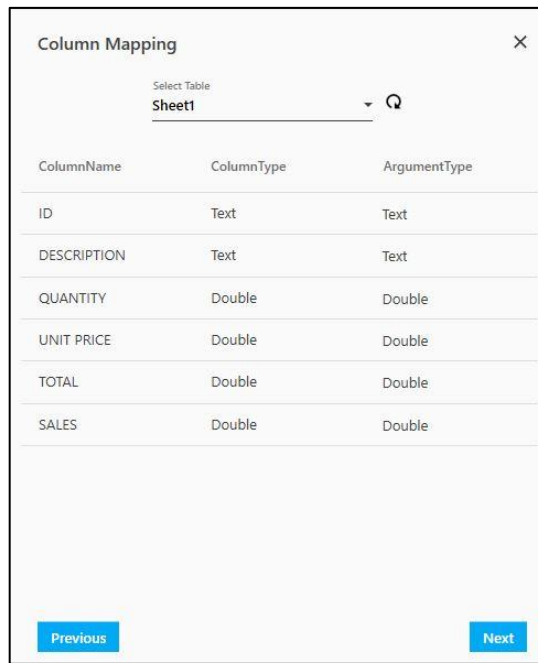


7. Click the  (Settings) icon to specify configurations related to the source file. The **Source Configuration** dialog box appears.

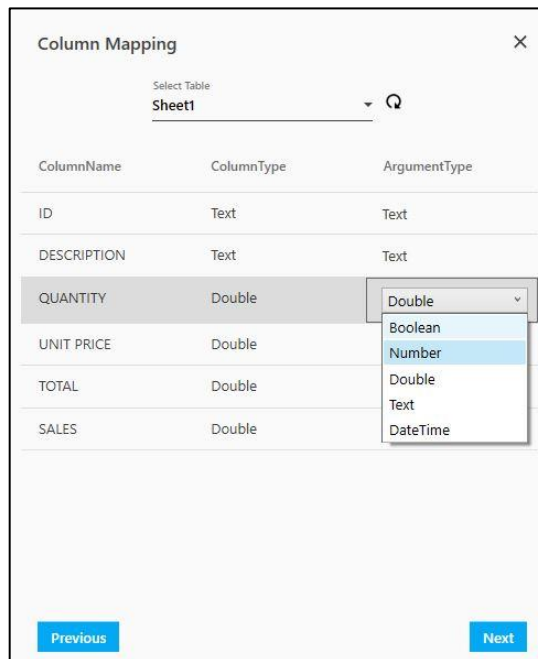


- 8. In the **Source Filepath** list, select the **SourceDT** argument created above to pass the excel file a sinput.
- 9. Select the **Has Header** check box to indicate that the first row of the excel file is the header data.
- 10. In the Output list, select the OutputData argument created above to access and store the short listed data at runtime.

11. Click **Next**. The **Column Mapping** dialog box appears. The default schema is displayed. The **QUANTITY**, **UNIT PRICE**, **TOTAL** and **SALES** are number type data, to reset their argument type:



In the ArgumentType column of each of the respective ColumnName entries, click and then select Number.



12. Click **Next**. The **Preview & Filter** dialog box appears. By default, the **Preview** pane displays the data of the source file.

Preview & Filter ✕

Filter And Sort ▼

Preview 8 of 8 rows ▲

ID	DESCRIPTION	QUANTITY	UNIT PRICE	TOTAL
A-14	Model 6210	1	78	78
B-25	Model 123e	1	120	120
K-13OU	Model 600	2	95	190
H-47	Model e35	1	68	68
M-02	Model 960gl	1	140	140
K-140	Model 110z	2	175	350

Previous
Finish

13. Expand the **Filter & Sort** pane, and then click the **+** (Add Filter Expression) icon to auto generate an **Expression ID** used to form the query expression for applying the lower limit filter on the total price.
- In the **Field Name** column, select **TOTAL** to include in the expression Id.
 - In the **Operator** column, select **>** to set the criteria to match data based on the lower limit of the total price set.
 - In the **Argument Name** column, select the **TotalPriceLL** argument created above.
14. Click the **+** (Add Filter Expression) icon to auto generate another **Expression ID** used to form the query expression for applying the upper limit filter on the total price.
- In the **Field Name** column, select **TOTAL** to include in the expression Id.
 - In the **Operator** column, select **<** to set the criteria to match data based on the upper limit of the total price set.

- c. In the **Argument Name** column, select the **TotalPriceUL** argument created above.

Preview & Filter ✕

Filter: ⊕ ⊖

Expression ID	Field Name	Operator	Argument Name
expression1	TOTAL ▾	> ▾	TotalPriceLL ▾
expression2	TOTAL ▾	< ▾	TotalPriceUL ▾

Sort By: ID ▾ ASC ▾

No. of Rows: RowCount ▾

Execute

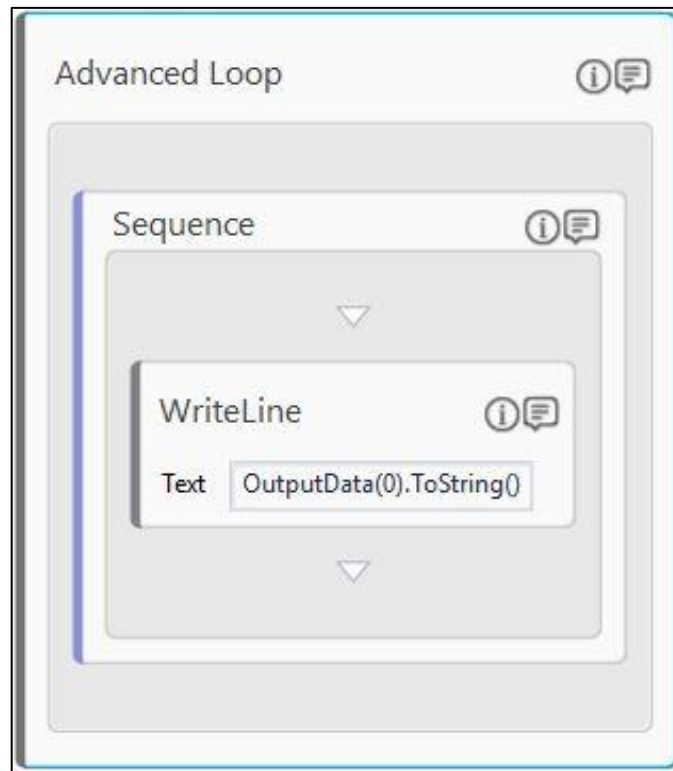
Preview 3 of 3 rows ⌵

ID	DESCRIPTION	QUANTITY	UNIT PRICE	TOTAL
B-25	Model 123e	1	120	120
K-13OU	Model 600	2	95	190
M-02	Model 960ql	1	140	140

Previous
Finish

- In the **Sort By** list, select ID and **ASC** to sort the data based on the ascending order of the ID of the items.
- In the **No. of Rows** list, select the **RowCount** argument created above to limit the number of rows of the source data to be looped over.
- Click **Execute** to view the short listed data as per the filter expression set.
- Click **Finish**. You are taken back to the Studio menu.

To view the output in Automation Studio, let's add WriteLine activity inside the Advanced Loop activity. You can assign this process to a robot if you want to execute this process outside Automation Studio.



19. In the **Text** field, enter **OutputData(0).ToString()** to convert the **System.Data.DataRowView** to **String** data type and then print the item Ids of the short listed item as per the filter expression set.
20. Save the process.
21. Setup the environment and then perform test run.



The console displays the item Ids in ascending order whose total price is greater than 100 but less than 300.

9.8 DataTable

DataTable tool is used to work with tabular data. This tool helps you to obtain and process the data to display in the table or grid format. Following activities are included in DataTable tool:

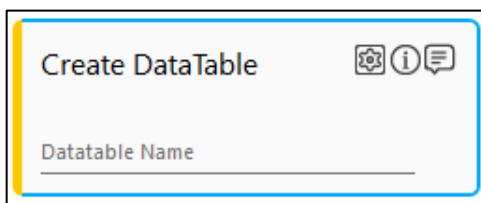
- [Create DataTable](#)
- [Add Row](#)
- [Remove Row](#)
- [Update Data](#)
- [Clear DataTable](#)
- [Get Data](#)
- [Import DataTable](#)
- [Export DataTable](#)
- [Sort DataTable](#)
- [Count Row Column](#)

9.8.1 Create DataTable

This activity allows you to create a data table. By using this activity, you can create a new customized data table in the Automation Studio without having dependency on any application such as, excel.


Using Create DataTable Activity

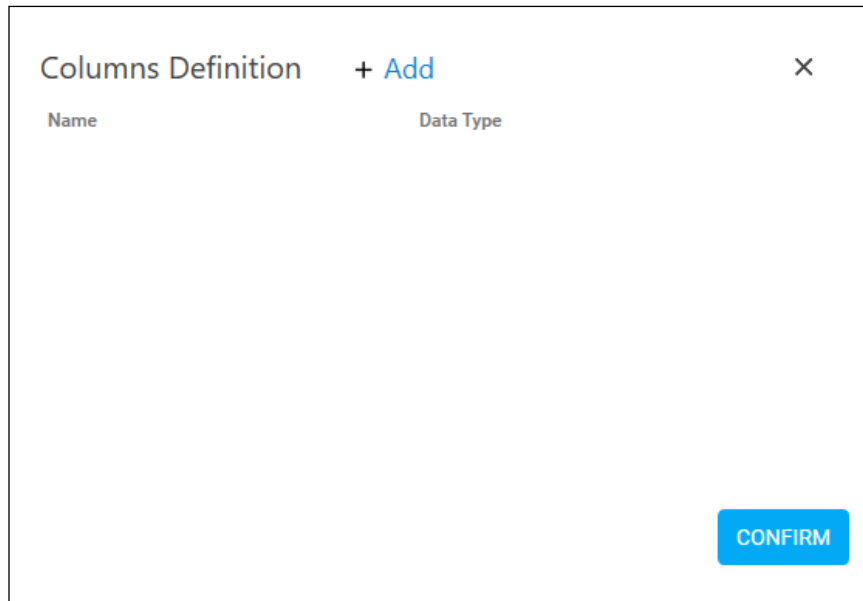
1. In the **Canvas Tools** pane, click **DataTable** to expand the tool and view the associated activities.
2. Drag the **Create DataTable** activity and drop on to the **Flowchart** designer area on the **Canvas**.



3. Enter a name for the datatable in the **Datatable Name** field. A datatable is created with the name provided in DataTable Name field.

Note: Special characters and space(s) are not allowed in the data table name except underscore. You can define the data table name in one single word or use underscore to separate two words, for example, Attendance, Data_Table_1, Demo_file.

4. Click the  (**Settings**) icon to add and define the columns in the Datatable. The **Columns Definition** window appears.

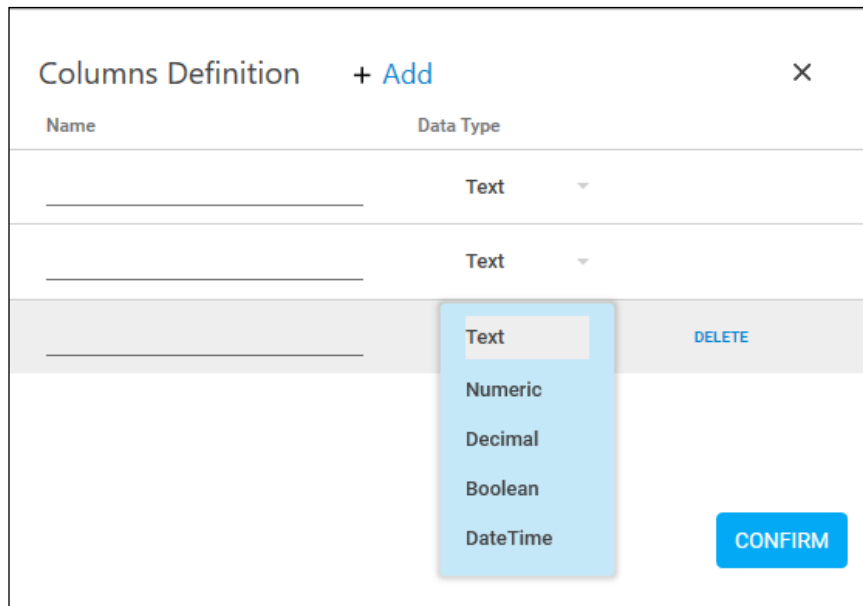


Name	Data Type
------	-----------

CONFIRM

Note: At least one column is required in the data table.

5. Click **Add** to add column headers and its associated Data Type in the Datatable.



Name	Data Type
	Text
	Text
	<ul style="list-style-type: none"> Text Numeric Decimal Boolean DateTime

CONFIRM

It helps you create and customize each new column you want to add to the data table. You can define the columns and select the type of values that the new column is going to accept.

- **Text** – Select this if column input is in string/text. For example, Student Name, Applicant Name.
- **Numeric** – Select this if column input is in whole number. For example, 6, 45.
- **Decimal** – Select this if column input is in decimal number. For example, 7.89, 1.34.

- **Boolean** – Select this if column input is either true or false.
 - **DateTime** – Select this if column input is in date and time format.
6. Click **CONFIRM** to save the datatable.

Create DataTable Properties

The properties of Create DataTable activity are listed in the following table and can be edited in the Properties grid on the right pane.

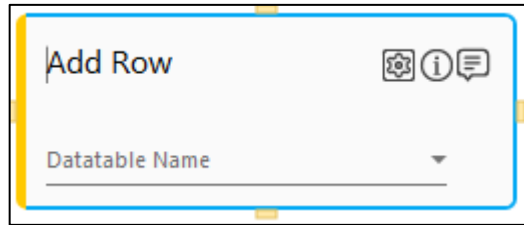
Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Create DataTable . You can change the name as required.
IsExtracted	Select this option to mark the activity as transferring the information to the bot in control tower, to check the data is extracted successfully from DataTable before proceeding with subsequent process.


9.8.2 Add Row

This activity allows you to add new row in the datatable.

Using Add Row Activity

1. In the **Canvas Tools** pane, click **DataTable** to expand the tool and view the associated activities.
2. Drag the **Add Row** activity and drop on to the **Flowchart** designer area on the **Canvas**.



3. In the **Datatable Name** list, select a datatable to which the row is to be added.
4. Click the  (**Settings**) icon to add a row in the existing datatable. The **Add Row Mapping** window appears.

Add Row Mapping ×

Name	Data Type	Value	If Default
Col1	Text	Dave	<input type="checkbox"/>
Col2	Numeric	20	<input checked="" type="checkbox"/>
Col3	Decimal	12.3	<input checked="" type="checkbox"/>
Col4	Boolean	true	<input type="checkbox"/>

CONFIRM

5. In the **Value** column, enter or select the value in the respective field.

In the Value list, select a parameter that holds the value. You must define the parameter in the Parameter bar to use this option. Alternatively, In the Value field, enter the value and select the If Default checkbox to set it as the default value.

6. Click **CONFIRM** to add new row in the DataTable.

Add Row Properties

The properties of Add Row activity are listed in the following table and can be edited in the Properties grid on the right pane.

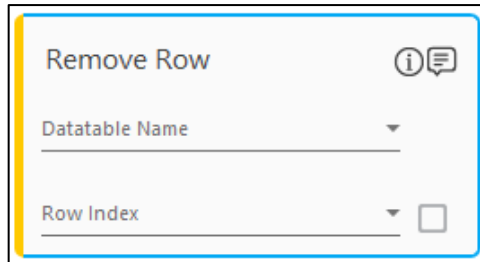
Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Add Row . You can change the name as required.

9.8.3 Remove Row

This activity allows you to delete a row from the datatable.

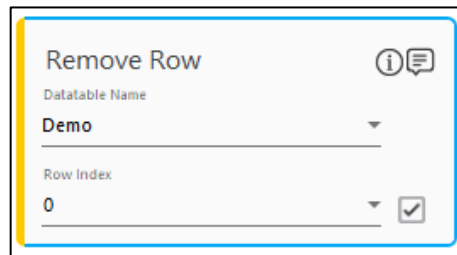
Using Remove Row Activity

1. In the **Canvas Tools** pane, click **DataTable** to expand the tool and view the associated activities.
2. Drag the **Remove Row** activity and drop on to the **Flowchart** designer area on the **Canvas**.



3. In the **Datatable Name** list, select a datatable from which the row needs to be removed.
4. In the **Row Index** field, enter or select the index of the row to be deleted.

In the Row Index list, select a parameter that holds the value. You must define the parameter in the Parameter bar to use this option. Alternatively, In the Row Index field, enter the value of the row index and select the If Default checkbox to set it as the default value.



Note: In Automation Studio, Row Index count start from 0. Row index should not be greater than the number of datatable rows.

Remove Row Properties

The properties of Remove Row activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity.</p> <p>If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	

Property Name	Usage
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Remove Row . You can change the name as required.

9.8.4 Update Data

This activity allows you to update the values in an existing datatable.

Using Update Data Activity

1. In the **Canvas Tools** pane, click **DataTable** to expand the tool and view the associated activities.
2. Drag the **Update Data** activity and drop on to the **Flowchart** designer area on the **Canvas**.

3. In the **Datatable Name** list, select the datatable that needs to be updated.
4. In the **Row Index** field, enter or select the index of the row to be updated.

In the Row Index list, select a parameter that holds the value. You must define the parameter in the Parameter bar to use this option. Alternatively, In the Row Index field, enter the value of the row index and select the If Default checkbox to set it as the default value.

Note: In Automation Studio, Row Index count starts from 0. Row index should not be greater than the number of datatable rows.

5. In the **Column Name** list, select the name of the column whose value needs to be updated.
6. Enter or select the argument in the **Argument** field.

In the Argument list, select a parameter that holds the value. You must define the parameter in the Parameter bar to use this option. Alternatively, In the Argument field, enter the value of the argument and select the If Default checkbox to set it as the default value.

Update Data Properties

The properties of Update Data activity are listed in the following table and can be edited in the Properties grid on the right pane.

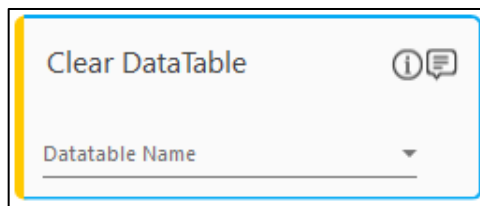
Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Update Data . You can change the name as required.

9.8.5 Clear DataTable

This activity allows you to clear all the data in the datatable.

Using Clear DataTable Activity

1. In the **Canvas Tools** pane, click **DataTable** to expand the tool and view the associated activities.
2. Drag the **Clear DataTable** activity and drop on to the **Flowchart** designer area on the **Canvas**.



3. In the **Datatable Name** list, select the datatable that needs to be cleared.

Clear Datatable Properties

The properties of Clear DataTable activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.

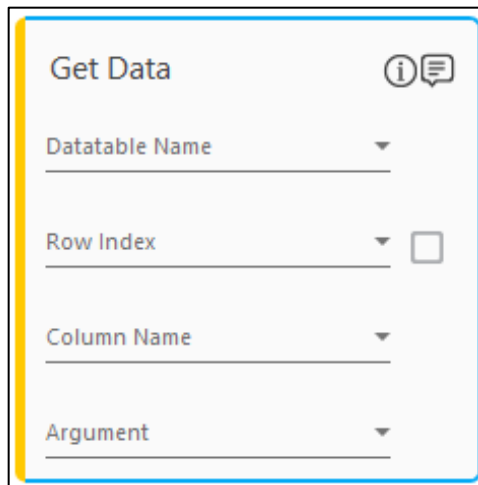
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Clear DataTable . You can change the name as required.

9.8.6 Get Data

This activity allows you to get value from a specific row in the datatable, according to a specified column.

Using Get Data Activity

1. In the **Canvas Tools** pane, click **DataTable** to expand the tool and view the associated activities.
2. Drag the **Get Data** activity and drop on to the **Flowchart** designer area on the **Canvas**.



3. In the **Datatable Name** list, select the datatable from which you want to retrieve the data.
4. In the **Row Index** field, enter or select the index of the row whose value is to be retrieved.

In the Row Index list, select a parameter that holds the value. You must define the parameter in the Parameter bar to use this option. Alternatively, In the Row Index field, enter the value of the row index and select the If Default checkbox to set it as the default value.

Note: In Automation Studio, Row Index count start from 0. Row index should not be greater than the number of datatable rows.

5. In the **Column Name** list, select the name of the column whose value is to be retrieved from the row.
6. In the **Argument** field, Select or enter the name of the argument in which value is set.
7. You can use any activity to view the result, such as, WriteLine, Write to Excel.

Get Data Properties

The properties of Get Data activity are listed in the following table and can be edited in the Properties grid on the right pane

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes, the application ignores any error while executing the activity. If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No.
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Get Data. You can change the name as required.

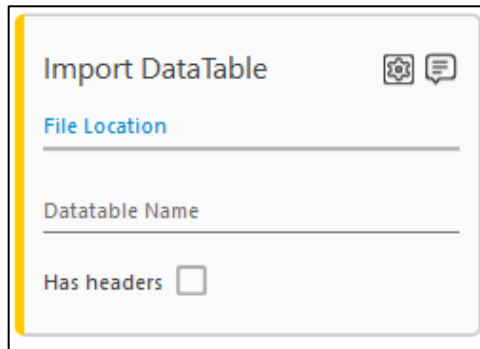
9.8.7 Import DataTable

This activity allows you to import data from excel or csv file into a datatable.

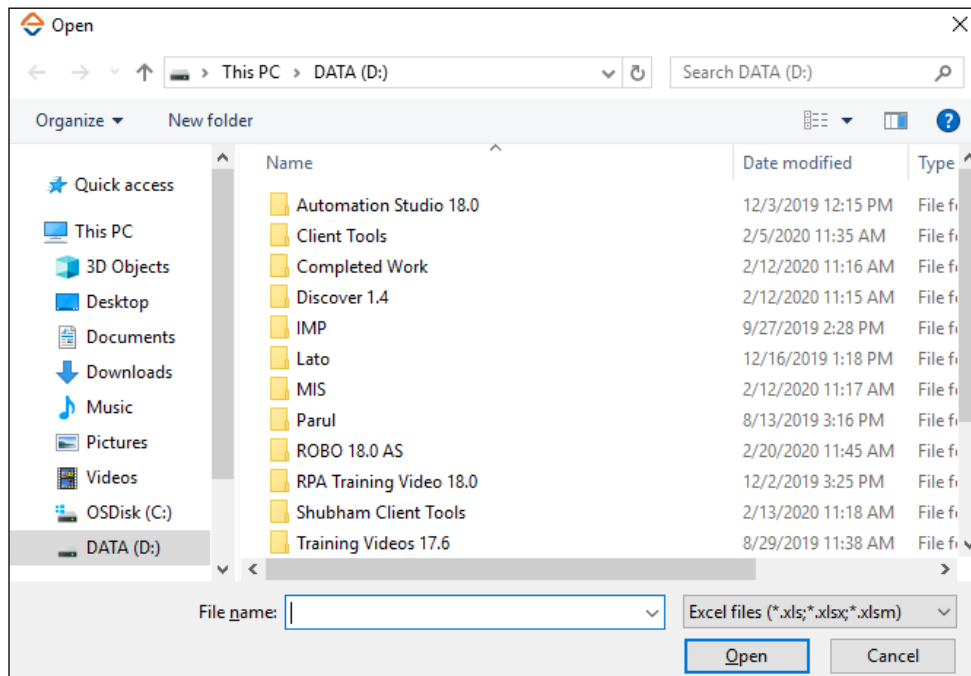
Using Import Datatable Activity

1. In the **Canvas Tools** pane, click **DataTable** to expand the tool and view the associated activities.

2. Drag the **Import DataTable** activity and drop on to the **Flowchart** designer area on the **Canvas**.




3. Click **File Location** to import datatable. The **Open** windows appears.



4. Navigate and select the required file, and then click **Open**.

Note: If your datatable is being extracted from an Excel File, the Excel Sheet field appears in Import DataTable activity. In the Excel Sheet list, select the sheet or worksheet which you want to import in Automation Studio.

5. Enter a name for the datatable in the **Datatable Name** field. A datatable is created and the selected file is imported in the datatable.
6. Select the **Has Headers** option, if you want to import datatable headers.
7. Click the  (**Settings**) icon, and then click the required option to configure the datatable. The available options are:

- a. **Configure Reader:** This opens a **Reader** window to configure rows.

If you want data from a specified section/range of the file to be imported in Automation Studio, enter or select the value in the **Row Start** and **Rows To Take** fields respectively.

Field	Value	Dropdown	Checkbox
Reader	ExcelReader		
Sheet Name	Sheet1		
Row Start	1	▼	<input checked="" type="checkbox"/>
Rows To Take	2147483647	▼	<input checked="" type="checkbox"/>

In the Row Start/Rows To Take field, enter the value of the row. Alternatively, In the Row Start/Rows To Take list, select a parameter that holds the value. You must define the parameter in the Parameter bar to use this option and clear the If Default checkbox.

- b. **Configure Headers:** This opens the **Headers Configuration** window to configure columns.

Name	Data Type	Column Number	Load
Applicant Name	Text	1	<input checked="" type="checkbox"/>
Physics	Text	2	<input checked="" type="checkbox"/>
Chemistry	Text	3	<input checked="" type="checkbox"/>
Maths	Text	4	<input checked="" type="checkbox"/>

CONFIRM

In Headers Configuration window, the following four entities are available:

- **Name** – Name of the header. You can change the name as per your requirement.
- **Data Type** – The type of data for the header. You can change the data type as per your requirement.
- **Column Number** – The column number of the header. This field cannot be edited.
- **Load** – Select the Load option, if you want to import the column else uncheck.

Note: If you have tick mark Has Header check box in Import DataTable activity, then DataTable Headings appears in Headers Configuration window.

Import DataTable Properties

The properties of Import DataTable activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes, the application ignores any error while executing the activity. If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No.
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Excel Input	
Sheet Name	The selected sheet name of the selected Excel is displayed.
File Details	
File Path	The file path of the selected excel is displayed in this field. You can also define the file path directly in this field. The format to define the file is D:\MST\Stock.xls.
File Type	The file type of the imported DataTable is stored in this field, such as, excel, csv, tab delimiters.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Import DataTable. You can change the name as required.
Output	
Column Count	Define the parameter in argument to store the value and use the stored value in the subsequent process. In this parameter, by default column count of an imported DataTable gets stored.
Row Count	Define the parameter in argument to store the value and use the stored value in the subsequent process. In this parameter, by default row count of an imported DataTable gets stored.

Property Name	Usage
Success Status	Define the parameter in argument to store the value and use the stored value in the subsequent process. In this property, you can store Boolean value. This value specifies whether the import is successful or not. As per the received value/results whether subsequent process is decided.

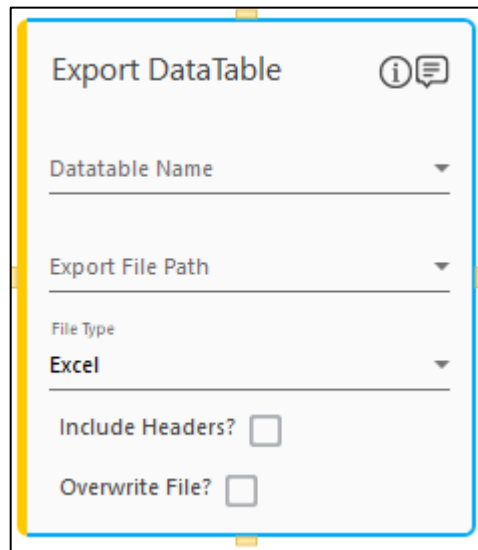
9.8.8 Export DataTable

This activity allows you to export datatable to excel, csv, or tab delimited file. There are some points to remember while using this activity:

- Ensure that the data is filtered, transformed, and harmonized in the desired format before using this activity.
- Ensure that the output folder/directory is created, and user has the adequate write permissions to execute this activity.

Using Export Datatable Activity

1. In the **Canvas Tools** pane, click **DataTable** to expand the tool and view the associated activities.
2. Drag the **Export DataTable** activity and drop on to the **Flowchart** designer area on the **Canvas**.



3. In the **Datatable Name** list, select a datatable which you want to export.
4. Select the file path from the **Export File Path** list. You must define the file path as argument in the Parameter bar. Refer **Parameter** section in the **AE-RPA-Engage-AutomationStudio-UserManual.pdf** if you want to know more about parameters and how to use it.

Note: You must define the file path along with the name and the extension, such as, D:\stock.xls.

5. Select the file type from the **File Type** list. The available options are Excel, CSV (comma separated value), and Tab delimited.

6. Select the **Include Headers?** check box to include the headers of the datatable in the exported datatable. This is helpful if you want to read and use the exported datatable independently.
7. Select the **Overwrite File?** check box to overwrite the file if a file with the same name exists in the defined path.

Note: If all the exported files need to be retained, then implement dynamic file naming to use a unique file name for each exported file.

Export DataTable Properties

The properties of Export DataTable activity are listed in the following table and can be edited in the Properties grid on the right pane.

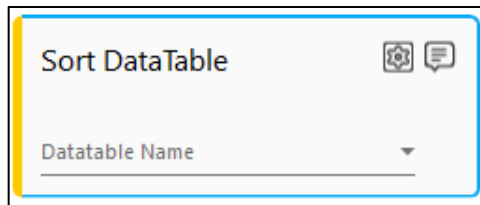
Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Export DataTable . You can change the name as required.


9.8.9 Sort DataTable

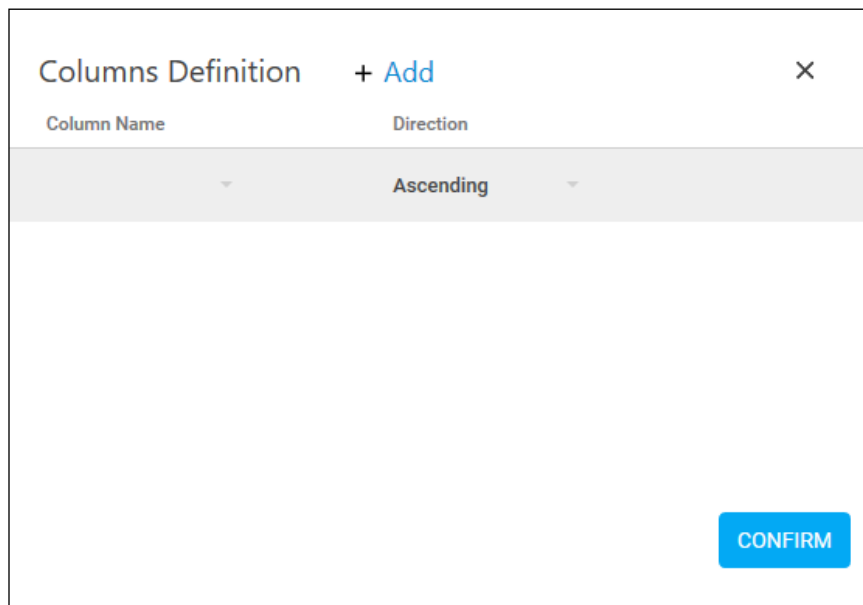
This activity allows you to sort the data in a datatable based on the values of the specified columns. The values of the specified column can be sorted in ascending or descending order only.

Using Sort DataTable

1. In the **Canvas Tools** pane, click **DataTable** to expand the tool and view the associated activities.
2. Drag the **Sort DataTable** activity and drop on to the **Flowchart** designer area on the **Canvas**.



3. In the **Datatable Name** list, select the datatable you want to sort.
4. Click the  (**Settings**) icon. The **Columns definition** window appears.



5. Click **Add** to define a column you want to sort.
6. In **Column Name**, select the column from the drop-down.
7. In the **Direction** column, select **Ascending or Descending** depending on the order in which the table is to be sorted.
8. Click **Confirm**.

Sort Datatable Properties

The properties of Sort DataTable activity are listed in the following table and can be edited in the Properties grid on the right pane

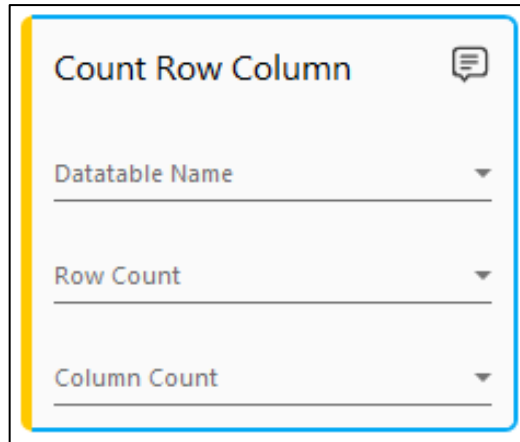
Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Sort DataTable . You can change the name as required.

9.8.10 Count Row Column

This activity counts the number of rows and columns of the datatable.

Using Count Row Column

1. In the **Canvas Tools** pane, click **DataTable** to expand the tool and view the associated activities.
2. Drag the **Count Row Column** activity and drop on to the **Flowchart** designer area on the **Canvas**.



3. In the **Datatable Name** list, select the datatable for which you want to count the rows and columns.
4. In the **Row Count** field, enter the variable name to store the row count of the datatable.
5. In the **Column Count** field, enter the variable name to store the column count of the datatable.

Count Row Column Properties

The properties of Count Row Column activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	

Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Count Row Column . You can change the name as required.

9.8.11 Example of Create DataTable, Add Row, Sort DataTable and Export DataTable

Let us consider creating a data table which comprises of student's marks and sort the data table by student name and export it to excel. The activities used in this example are Create DataTable, Add Row, Sort DataTable and Export DataTable.

To create a datatable for student's marks:

1. Create a new process.
2. Create an **In** argument, **Standard** and **AdmissionStatus**. Define **Argument Type** as **String**. These arguments are used to provide input and store the datatable output.
3. Define Default Value for Standard as 12th and AdmissionStatus as D:\Marks List.xlsx respectively.

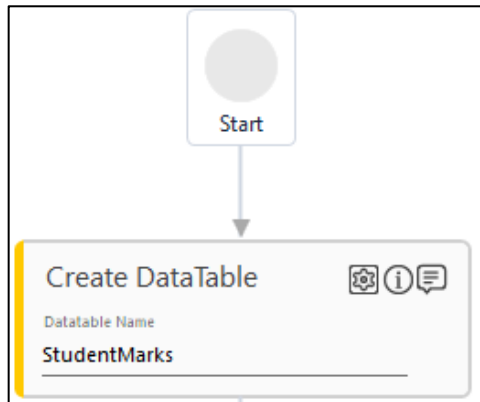
Name	Direction	Argument type	Default value
Standard	In	String	"12th"
AdmissionStatus	In	String	"D:\Marks List.xlsx"

Create Argument

Variables Arguments

4. From the Canvas Tools panel, add Create Datatable to the Flowchart designer on the Canvas

5. In the Datable Name field, enter StudentMarks.



6. Click the  (Settings) icon. The **Columns Definition** window appears.

Name	Data Type
Name	Text
Class	Text
Physics	Decimal
Chemistry	Decimal

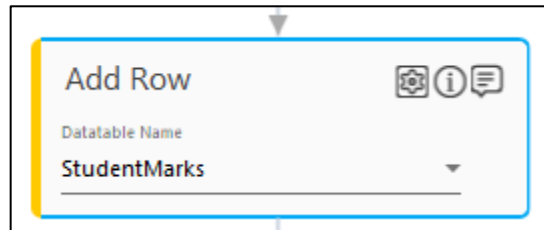
CONFIRM

- a. Click **Add**. Enter **Name** in the **Name** column, and select **Data Type** as **Text** from the list.
- b. Click **Add**. Enter **Class** in the **Name** column, and select **Data Type** as **Text** from the list.
- c. Click **Add**. Enter **Physics** in the **Name** column, and select **Data Type** as **Decimal** from the list.
- d. Click **Add**. Enter **Chemistry** in the **Name** column, and select **Data Type** as **Decimal** from the list.
- e. Click **Add**. Enter **Math** in the **Name** column, and select **Data Type** as **Decimal** from the list.

Note: Since the marks is captured in decimal, the data type selected for Physics, Chemistry, and Math is Decimal.

7. Click **CONFIRM**.
8. From the Canvas Tools panel, add Add Row to the Flowchart designer on the Canvas

9. In the Datable Name list, select StudentsMarks.



10. Click the  (Settings) icon. The **Add Row Mapping** window appears.

Name	Data Type	Value	If Default
Name	Text	Dave	<input checked="" type="checkbox"/>
Class	Text	Standard	<input type="checkbox"/>
Physics	Decimal	91	<input checked="" type="checkbox"/>
Chemistry	Decimal	95	<input checked="" type="checkbox"/>
Maths	Decimal	87	<input checked="" type="checkbox"/>

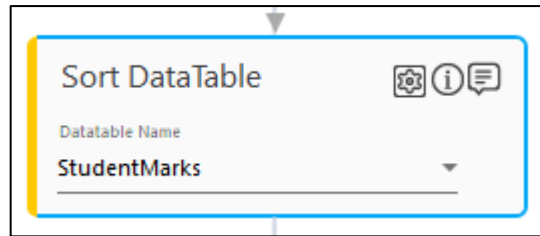
CONFIRM

- In the **Name** row, enter the value in the **Value** field.
- In the **Class** row, select **Standard** from the **Value** list and clear **If Default** checkbox.
- In the **Physics** row, enter the value in the **Value** field.
- In the **Chemistry** row, enter the value in the **Value** field.
- In the **Maths** row, enter the value in the **Value** field.
- Click **CONFIRM**.

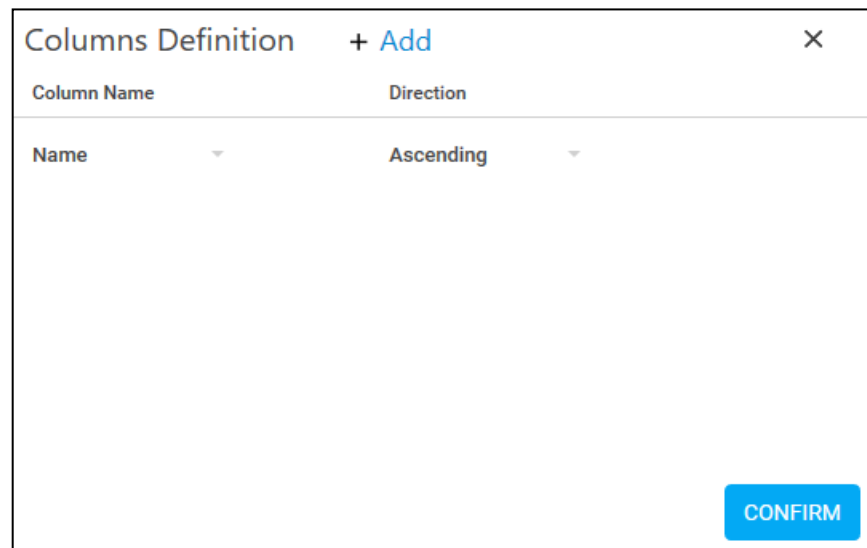
Note: In the Value field, enter the value and select the If Default checkbox to set it as the default value. Alternatively, In the Value list, select a parameter that holds the value. You must define the parameter in the Parameter bar to use this option.

- Follow steps 8 through 10 to add student's details.
- From the Canvas Tools panel, add Sort DataTable to the Flowchart designer on the Canvas

13. In the Datable Name list, select StudentMarks.



14. Click the  (Settings) icon. The **Columns Definition** window appears.



- In the **Column Name** list, select **Name**.
- In the **Direction** list, select **Ascending**.

Note: The student name is sorted in the alphabetical order.

15. Click **CONFIRM**.

To view the output of this example, use Export DataTable activity.

16. From the Canvas Tools panel, add Export DataTable to the Flowchart designer on the Canvas

- In the Datatable Name list, select StudentMarks.
 - In the **Export File Path** list, select **AdmissionStatus**. As defined in the arguments, the **AdmissionStatus** indicates the file path.
 - In the **File Type** list, select **Excel**.
 - Select Include Headers? and Overwrite File? Options.
17. Click **Save**. The **Save Process** window appears.
18. Enter the **Process Name**, and click **SAVE**.
19. Setup the environment, and then perform test run.

The output is generated and placed at the specified file path location. You can open the Marks List excel to view the data.

	A	B	C	D	E
1	Name	Class	Physics	Chemistry	Maths
2	Dave	12th	91	95	87
3	Donny	12th	78	89	90
4	Paul	12th	67	45	69
5	Ram	12th	89	98	99
6	Tim	12th	98	95	99
7					

9.9 Image

Image based automation is used to capture the image and perform actions on the captured image or read/extract text from the text based image. Following activities are included in Image tool:

- [Image Control](#)
- [Text Extractor](#)

Additionally, Automation Studio provides image mode to perform image based automation via multimodal interface to enhance user experience. See [Image Mode](#) for more information.

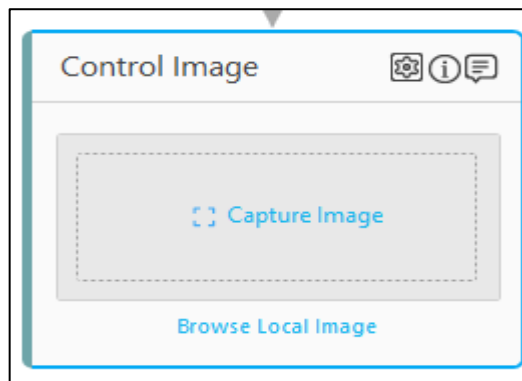
9.9.1 Image Control

This activity allows you to capture image and perform actions on the captured image.

Using Image Control

In the **Canvas Tools** pane, click **Image** to expand the tool and view the associated activities.



1. Drag the **Image Control** activity and drop on to the **Flowchart** designer area on the **Canvas**.



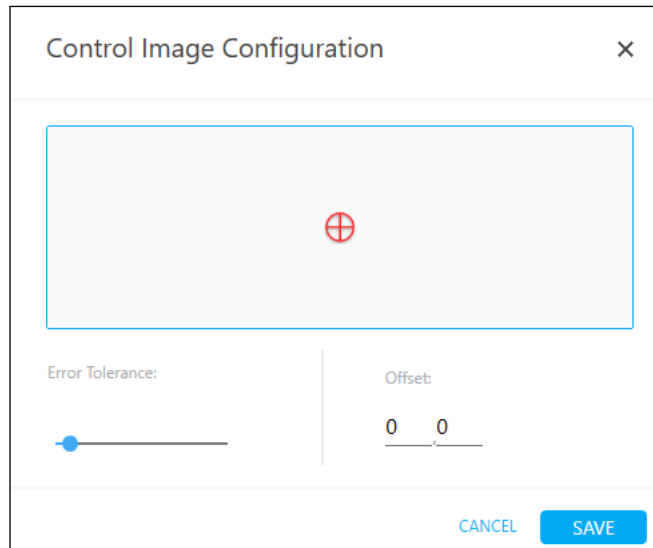
2. Click **Capture Image/Browse Local Image** to perform actions on the captured image as per your business requirement.

Click Capture Image and select the rectangular image area from the application in focus. The last focused application is captured in capture image.

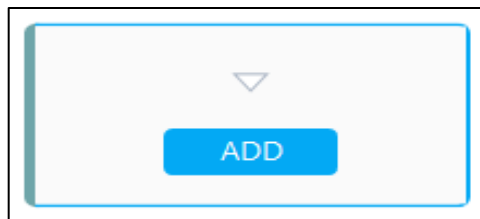
Click Browse Local Image to browse and attach the image from the local machine.

Additionally, you can use the  (Recapture Image) icon and the  (Delete) icon in the image activity to recapture or delete the image. These icons are displayed once the image is captured in the activity window.

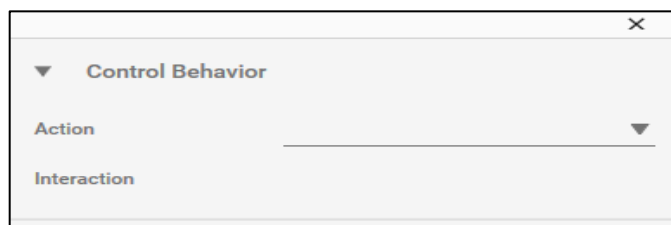
3. Click the  (**Settings**) icon. The **Control Image Configuration** window appears.



- **Error Tolerance:** Maximum acceptable error tolerance in the image matching while performing the image search during execution.
 - **Offset:** They are coordinates of the reference point selected during image capture with respect to the image. If required, alter the offset coordinates and save them from this screen.
 - **CANCEL:** Click **CANCEL** to cancel the changes.
 - **SAVE:** Click **SAVE** to save the changes.
4. Double-click Control Image Activity and click **ADD**. The interaction appears to configure the control behavior.



5. Click the  (**Settings**) icon. The **Control Behavior** window appears.



- In the **Action** list, select the action. Refer [image control field description](#) table to know more about actions and its details.

Note: You can add multiple interaction on the captured image. To add multiple interaction, click ADD and then click the Settings icon to select the action.

Image Control Field Description

The properties of Image Control fields are listed in the following table:

Property	Usage
Control Behavior	
Action	<p>Actions that can be performed on the image based application. Available options are:</p> <ul style="list-style-type: none"> ▪ Wait Until Exists - Waits till the image is found on the web page and is in focus. You can use this to avoid any delay that might occur for the image to appear. This interaction returns a Boolean value of true if the image is found and false if the image is not found. The Boolean value can be set to a variable and checked in a conditional statement to perform an appropriate action after finding the image. ▪ Wait Until Disappears - Waits until the image disappears from the web page. You can use this to avoid delay that might occur for the image to disappear. This interaction returns a Boolean value of true if the image is not found and false if it is found. The Boolean value can be set to a variable and checked in a conditional statement to perform an appropriate action after finding the image. ▪ Wait Until Clicked - Waits until the image is clicked. This interaction returns a Boolean value of true if the image is clicked and false if the image is not clicked. The Boolean value can be set to a variable and checked in a conditional statement to perform an appropriate action after finding the image. ▪ Click And Type - Allows to left click at the required area and provide the input value. Set a default value or pass the value using parameter defined in the Parameter bar. ▪ Left Click - Performs a left click specified at the offset point. ▪ Right Click - Performs a right click at the specified offset point. ▪ Left Double Click - Performs a left double click at the specified offset point. ▪ Right Double Click - Performs a right double click at the specified offset point. ▪ Mouse Over - Moves mouse pointer over the area at the specified offset point. ▪ Drag - Moves the draggable image based upon the mentioned vertical and horizontal coordinates. ▪ Horizontal Scroll - Searches for the image towards either right or left direction based on the speed of the scroll as per the option selected in the

Property	Usage
	<p>Scroll Direction and Scroll Action fields respectively.</p> <ul style="list-style-type: none"> ▪ Vertical Scroll - Searches for the image in the upward or downward direction based on the speed of the scroll as per the option selected in the Scroll Direction and Scroll Action fields respectively. ▪ Type Text - Allows entering the text where the cursor is present on the page. Set a default value or pass the value using parameter defined in the Parameter bar. ▪ Press Keys - Performs press action for the specified key. Apply modifiers like Control, Shift, Alt before the key press as specified in the fields Modifier One and Modifier Two. ▪ Key Down - Performs press of a single key specified on the keyboard. ▪ Key Up - Brings the pressed key back to normal position. It must be performed after the Key Down action to release the pressed key.
Interaction	The corresponding interaction set against the selected action.
Appears if Action selected is Wait Until Exists, Wait Until Disappears or Wait Until Clicked .	
Timeout (ms)	The wait time for the plugin until image is found or identified that it does not exists.
Poll Frequency (ms)	The frequency of checking the availability of the image and if the system is ready for the next action.
Appears if Action selected is Click And Type .	
Input Source	<p>The input value that must be entered. You must define a parameter in the Parameter bar to use this option.</p> <p>Select the check box beside Input Source and enter the value, if you want to set a default value.</p>
Appears if Action selected is Drag .	
Drag X	The vertical drag performed on the application window to drag the draggable image. The value must be entered in pixels.
Drag Y	The horizontal drag performed on the application window to drag the draggable image. The value must be entered in pixels.
Appears if Action selected is Horizontal Scroll or Vertical Scroll .	
Scroll Amount	The scroll speed (set in pixels) for each movement while searching for the image in the specified scrollable area.
Scroll Direction	<p>The direction of scroll for searching the image.</p> <p>Available options for Horizontal Scroll are Left and Right and for Vertical Scrolls are Up and Down.</p>

Property	Usage
Appears if Action selected is Type Text .	
Delay in Character (ms)	Triggers the delay when there is typing lag from the keyboard.
Input Source	The input value that must be entered. You must define a parameter in the Parameter bar to use this option. Select the check box beside Input Source and enter the value, if you want to set a default value.
Appears if Action selected is Press Keys .	
Key	The single key set as input from the keyboard.
Modifier One	Select the first modifier to create the key combination. Available options are - Control, Shift, Alt and Win (Windows) keys of the keyboard.
Modifier Two	Select the second modifier to create the key combination. Available options are- Control, Shift, Alt and Win (Windows) keys of the keyboard.
Appears if Action selected is Key Down or Key Up .	
Key	The single key set as input from the keyboard.
Misc	
Appears if Action selected is Wait Until Exists or Wait Until Disappears .	
Variable Name	User defined name of the variable that stores the captured value. This field appears only for some of the selected actions.

Image Control Properties

The properties of Image Control activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.

Property	Usage
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
CurrentWindowHandler	This property is used internally by the activities. No action required.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Control Image . You can change the name as required.
Use Fuzzy Matching	Select this option to turn fuzzy matching of the image. This matching ignores the aspect ratio, background color and resolution change. Use this flag, when basic image matching fails

9.9.2 Text Extractor

This activity allows you to extract the text using OCR.

9.9.3 Prerequisites

OneNote OCR capabilities is added via configuration of Automation Studio and Robots. To use this feature following assembly file is required:

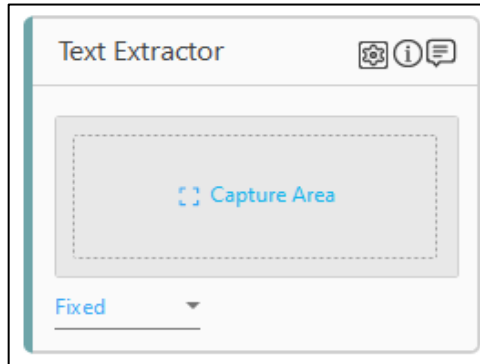
Component	Version
Interop.Microsoft.Office.Interop.OneNote.dll	1.1.0.2

To update assembly file:

1. Download the assembly file from [here](#) on your system.
2. Save the downloaded file at client-tools > AutomationStudio > **bin** > Plugins > OCR. If you download/access Automation Studio from the Admin module, you must save the required DLL at %localappdata% > **EdgeVerve** > **AutomationStudio** > **bin** > Plugins > OCR.
3. The assembly files are saved.

Using Text Extractor

1. In the **Canvas Tools** pane, click **Image** to expand the tool and view the associated activities.
2. Drag the **Text Extractor** activity and drop on to the **Flowchart** designer area on the **Canvas**.



3. Click **Capture Area** to capture the rectangle on screen on which OCR is to be performed.



If OCR Target is set to Desktop in the properties pane of the Text Extractor activity, then the last focused application is captured and shown for OCR area selection.

If OCR Target is set to File in the properties pane of the Text Extractor activity, then configure the image or define the image path in the Image File property in the properties pane.

Additionally, you can use the recapture image icon and the delete icon in the image activity to recapture or delete the image. These icons are displayed once the image is captured in the activity window.

4. In the **Fixed** list, select the usage mode as per your requirement.
 - If **OCR Target** is set to **Desktop**, following usage modes are available:
 - **Fixed:** In fixed mode, when the image is captured, a rectangular area is selected. The coordinates of the selected area are stored; no image is captured in this mode. The grey area in the activity acts as the image placeholder.
 - **Reference:** In the reference mode, two rectangular areas are selected to capture the image. The first selected rectangular area is stored as a template image which is used to perform search on screen during runtime. The second selected rectangular area is a region from where text is extracted using OCR. The coordinates of the selected area are stored for the second selection.

Additionally, you can capture the image using Browse local Image and browse the image on your local machine.


- If **OCR Target** is set to **File**, following usage modes are available:
 - **Fixed:** In fixed mode, when the image is captured, a rectangular area is selected from the given image file. The coordinates of the selected area are stored; no image is captured in this mode. The grey area in the activity acts as the

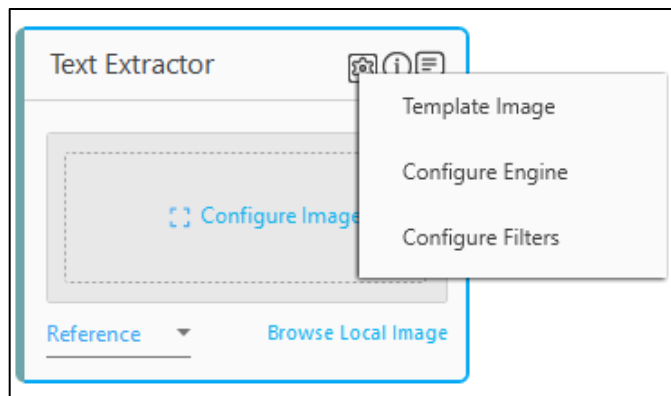
image placeholder.

- **Reference:** In the reference mode, two rectangular areas are selected from the given image file to capture the image. The first selected rectangular area is stored as a template image which is used to perform search on screen during runtime. The second selected rectangular area is a region from where text is extracted using OCR. The coordinates of the selected area are stored for the second selection.

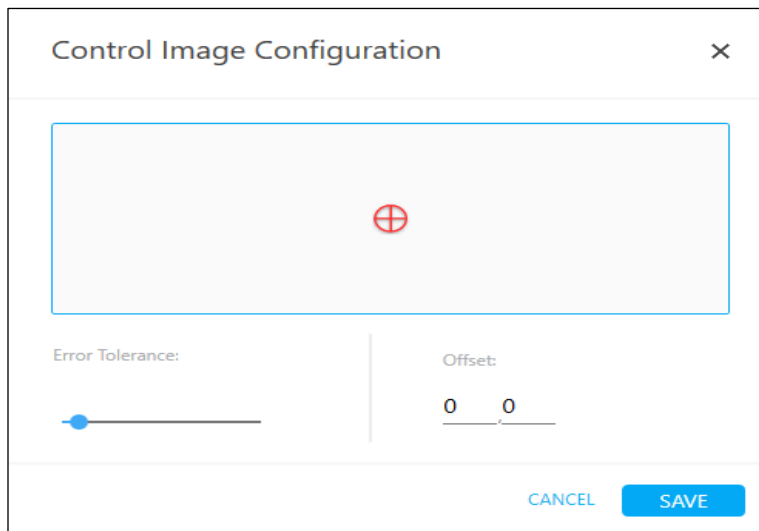
Additionally, you can capture the image using Browse local Image and browse the image on your local machine.

- **Full Image:** In Full Image mode, you can capture the full image to extract the text using OCR.

5. Click the  (**Settings**) icon, a list appears. Click the required option from the list.



- Template Image:** This option is available when usage mode is set as reference. It displays the template image which is used to perform search on screen during runtime.



- **Error Tolerance:** Maximum acceptable error tolerance in the image matching while performing the image search during execution.
- **Offset:** They are coordinates of the reference point selected during image capture with respect to the image. If required, alter the offset coordinates and save them from this screen.
- **CANCEL:** Click **CANCEL** to cancel the changes.

- **SAVE:** Click **SAVE** to save the changes.

b. **Configure Engine:** Configure the OCR engine used for text extraction.

OCR Engine	Tesseract
Configuration	ABBY Cloud OCR Google Vision API Microsoft Vision API One Note
Language	
Blacklisted Chars	Tesseract
Whitelisted Chars	
Data Type	Text
Variable Name	
Is CView	<input type="checkbox"/>

- **OCR Engine:** Select the required OCR Engine to convert the text into a machine-readable format. The configuration fields change as per the selected OCR Engine. Click any of the following link to know about the OCR Engine and their respective configuration details:
 - [Tesseract](#)
 - [ABBY Cloud API](#)
 - [Google Vision API](#)
 - [Microsoft Vision API](#)
 - [One Note](#)

Tesseract

This is a free software for OCR engine and available for various operating systems.

The properties of Tesseract are listed in the following table:

Property	Usage
Configuration	The selected OCR is displayed in this field.
Language	It Indicates the language of the targeted text. You can change or combine (two or more than two) languages, if required. For example, eng + Rus. Place the language data in the target data folder for any additional language for Tesseract engine.
Blacklisted Chars	The characters/digits that appears in the Blacklisted Chars field are ignored during the match, and other closest match are found. For example, 0123456789 are declared in the field than the mentioned characters are blacklisted, and 0 is not mistaken with O.
Whitelisted Chars	This field is also configured as Blacklisted Chars field, but the functionality of Whitelisted Chars is opposite to Blacklisted Chars. For example, 0123456789 are declared in the field than the mentioned characters are whitelisted, and 0 is not mistaken with O.
Data Type	The type of data used for text extraction in OCR. By default, data type is set to Text.
Variable Name	User defined name of the variable that stores the captured value. You can define the variable name here itself or predefine it in the Parameter bar.

ABBYY Cloud API

This is used to perform AI based image analysis. It is a third-party cognitive service. To consume this service, you need add a new application under CognitiveApps. See [Cognitive Services](#) for more details.

Property	Usage
Configuration	The selected OCR is displayed in this field.
ABBY Server URL	Enter the API URL.
Language	Specify the language.
Data Type	The type of data used for text extraction in OCR. By default, data type is set to Text.
Variable Name	User defined name of the variable that stores the captured value. You can define the variable name here itself or predefine it in the Parameter bar.

Google Vision API

This is used to perform AI based image analysis. It is a third-party cognitive service. To consume this service, you need add a new application under CognitiveApps. See cognitive services for more details.

Property	Usage
Configuration	The selected OCR is displayed in this field.
URL Of API	Enter the API URL.
Data Type	The type of data used for text extraction in OCR. By default, data type is set to Text.
Variable Name	User defined name of the variable that stores the captured value. You can define the variable name here itself or predefine it in the Parameter bar.

Microsoft Vision API

This is used to perform AI based image analysis. It is a third-party cognitive service. To consume this service, you need add a new application under CognitiveApps. See cognitive services for more details.

Property	Usage
Configuration	The selected OCR is displayed in this field.
URL	Enter the API URL.
Version	Enter the version of the API.
Is Handwritten	Select this option, if the image read is a handwritten image.
Data Type	The type of data used for text extraction in OCR. By default, data type is set to Text.
Variable Name	User defined name of the variable that stores the captured value. You can define the variable name here itself or predefine it in the Parameter bar.

One Note

OneNote OCR is a Microsoft OCR, and OneNote application installation is required to consume One Note OCR Engine.

To consume OneNote:

1. Download OneNote Interop dll as per prerequisite document.
2. Locate the following folder in your local machine "AutomationStudio\Plugins\OCR" where the Automation Studio is

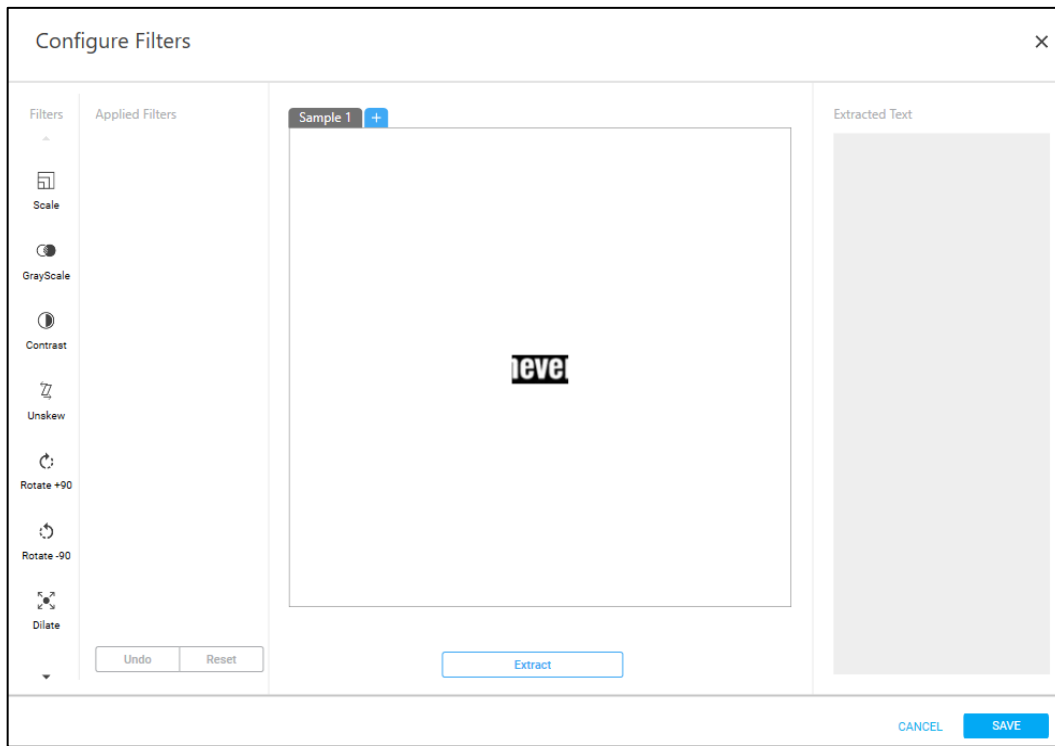
installed.

- Place the downloaded DLL at this location.

Property	Usage
Configuration	The selected OCR is displayed in this field.
Data Type	The type of data used for text extraction in OCR. By default, data type is set to Text.
Variable Name	User defined name of the variable that stores the captured value. You can define the variable name here itself or predefine it in the Parameter bar.

Note: A valid subscription is required to run third party cognitive services, such as, ABBYY, Google, Microsoft.

- C. **Configure Filters:** Apply preprocessing image filters to improve the accuracy of text extraction and test the correction of the OCR output. Following are the list of OCR filters used for text extraction.



- **Filters:** Configure the filters as per your requirement. Following are the list of available filters.
 - **Scale** – Use this filter to adjust the image/text in the right size. This filter scales the image by a factor of 2. The parameter to define the image is DPI.
 - **GrayScale** – Use this filter to convert the image in grayscale.
 - **Contrast** – Use this filter to adjust the contrast of the image. Contrast provides clarity in the image/text.
 - **Unskew** – Use this filter to correct the orientation of slightly skewed images to the horizontal or vertical axis, whichever is closet.
 - **Rotate +90** – Use this filter to rotate the image/text clockwise to 90 degree.
 - **Rotate -90** – Use this filter to rotate the image/text anti-clockwise to 90 degree.
 - **Dilate** – Use this filter to fill the holes and broken areas of the image.
 - **Erode** – Use this filter to reduce the brightness if the image.
 - **Exponential** – Use this filter to apply the System.Math.Exp function for each pixel in the image, clipping values as needed.
 - **Remove Noise** – Use this filter to remove noise from the background, such as, random variation of color and brightness.
 - **Invert** – Use this filter to reverse the saturation, brightness and hue values. The image is converted into photo negative.

- **Sharpen** – Use this filter to define the edges.
- **Smooth** – Use this filter to remove high spatial frequency noise.
- **Sepia** – Use this filter to convert the image in a brownish grey to dark yellowish-brown tone.
- **Applied Filters:** In this section, the list of applied filters is displayed.
 - **Undo:** Click **Undo** to cancel the selected filter.
 - **Reset:** Click **Reset** to unselect the applied filters.
- **Sample:** The selected image is displayed in this section. You can test the configured filters for correctness with maximum three sample. Below fields may vary as per your OCR Engine selected.
 - **Application Id** – Provide the Application Id to run the third-party cognitive service. This field is applicable if the selected OCR Engine is ABBYY Cloud Service.
 - **Password** – Provide the password for the mentioned Application ID to run the third-party cognitive service. This field is applicable if the selected OCR Engine is ABBYY Cloud Service.
 - **#NA#** - This field is not editable and not applicable.
 - **Subscription Key** – Provide the subscription key to run third-party cognitive service. This field is applicable for
 - **Extract** – Click Extract to view the extracted text or OCR output after preprocessing of the image.
- **Extracted Text:** Click Extract. The OCR output is displayed in this section.

Text Extractor Properties

The properties of Text Extractor activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property	Usage
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity.</p> <p>If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	

Property	Usage
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
CurrentWindowHandler	This property is used internally by the activities. No action required.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Text Extractor . You can change the name as required.
Use Fuzzy Matching	Select this option to turn fuzzy matching of the image. This matching ignores the aspect ratio, background color and resolution change. Use this flag, when basic image matching fails
Target	
Image File	Define the image path to perform the image OCR. The image file path is defined in string. For example: "D:\imageautomation\image1.jpeg"
OCR Target	Select OCR Target from the list to define how the image OCR is performed in the process. By default, this option is set to Desktop .

9.10 Files

File automations help you automate the tasks related to the files such as read the specific file, merge or split the files, search specified files. It helps you save the administration hours required to perform these tasks, enforce data use policies, prevent unauthorized applications to use the files, perform file management and storage. You can work with different file types such as PDF, MS Word and Excel files.

Following activities are included in Files tool:

- [Write To Excel](#)
- [Excel Loop](#)
- [Break Loop](#)
- [Use Case](#)
- [File Search](#)
- [File Read](#)
- [File Operations](#)
- [File Merger](#)

- [File Splitter](#)
- [Virus Scan](#)

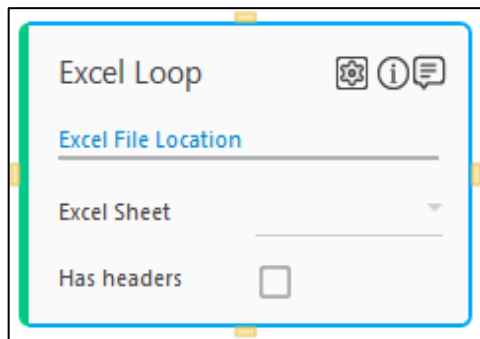
9.10.1 Write To Excel

Write To Excel activity is used within the Excel Loop activity to update specified variables into an excel sheet. Write to Excel activity can be used in the combination of Excel Loop activity. This activity cannot be used as a standalone activity to fetch data.

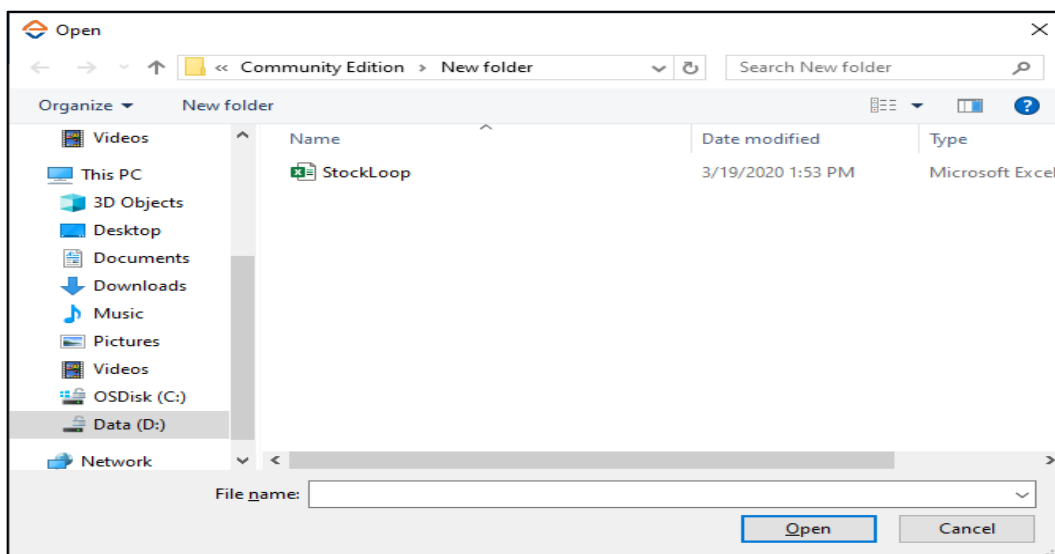
Using Write To Excel Activity

First, let's open the Studio and set up the Excel Loop activity.

1. In the **Canvas Tools** pane, click **Files** to expand the tool and view the associated activities.
2. Drag the **Excel Loop** activity and drop on to the **Flowchart designer** on the **Canvas**




3. Click **Excel File Location**. The **Open** window appears.



Select the excel file you want to automate and click Open.

4. In the **Excel Sheet** list, select the sheet or workbook you want to perform automate.

5. Select the **Has headers** check box if the selected excel has headers.
6. Click the  (**Settings**) icon. The **Output Parameters** dialog box appears to set data type and parameters.

Output Parameters ✕

Header Names	Data Type	Reuse Header Name <input checked="" type="checkbox"/> All	Parameter
Stock Name	Text	<input checked="" type="checkbox"/>	<u>StockName</u>
Stock Value	Text	<input checked="" type="checkbox"/>	<u>StockValue</u>
Risk	Text	<input checked="" type="checkbox"/>	<u>Risk</u>

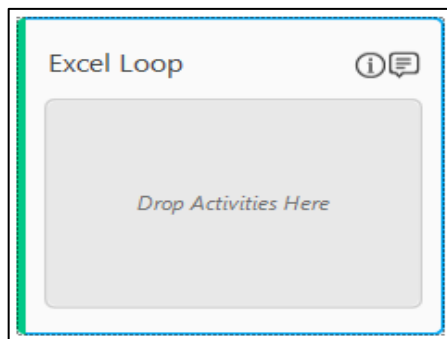
CONFIRM

Output Parameters dialog box have four columns:

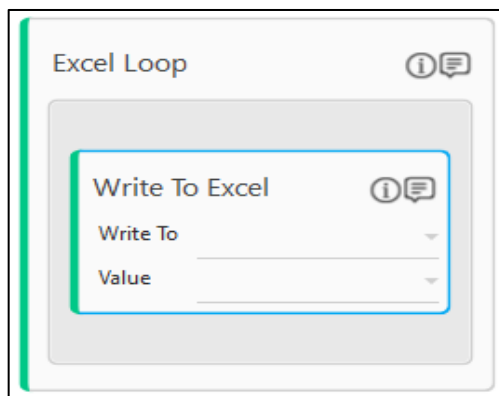
- a. **Headers Names:** In this column, you can view the headers name of the selected excel. This column is not editable.
 - b. **Data Type:** In this column, you can select the type of data for headers from the list. The available options are – Text, Numeric, Decimal, Boolean and DateTime.
 - **Text:** This data type is used to determine the value of the defined variable as text, such as, Dave, Stock.
 - **Numeric:** This data type is used to determine the value of the defined variable as Numeric, such as, 10, 88.
 - **Decimal:** This data type is used to determine the value of the defined variable as Decimal, such as, 10.01,9.04.
 - **Boolean:** This data type is used to determine the value of the defined variable as Boolean, such as, true, false.
 - **DateTime:** This data type is used to determine the value of the defined variable as date and time stamp, such as, 01122020;14:09:05.
 - c. **Reuse Header Name:** Select the **All** option if you want to reuse all the existing headers of the selected excel. You can even select the required headers individually by selecting the check box available in the headers row.
 - d. **Parameter:** In this column, headers parameter is available. You can edit the parameter name as per your requirement. These parameters are used to store the value and get the desired output. Parameter works like a variable.
7. Click **Confirm** to save the changes.

Now, let's add the Write To Excel activity within the Excel Loop activity.

8. Double-click the **Excel Loop** activity on the **Canvas**.



9. In the **Canvas Tools** pane, click **Files** to expand the tool and view the associated activities.
10. Drag the **Write To Excel** activity and drop on to the **Flowchart** designer on the **Canvas**, inside the **Excel Loop**.



11. In the **Write To** list, select the variable you want to update. If you use **New Excel Sheet** variable, a new worksheet is created within the excel sheet to populate the desired output.
12. In the **Value** list, select the variable in which output is stored while performing the automation process.

Write To Excel Properties

The properties of Write to Excel activity are listed in the following table and can be edited in the Properties grid on the right pane.

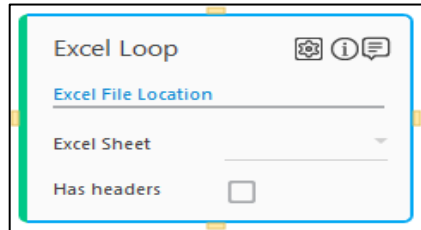
Property	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Write To Excel . You can change the name as required.

9.10.2 Excel Loop

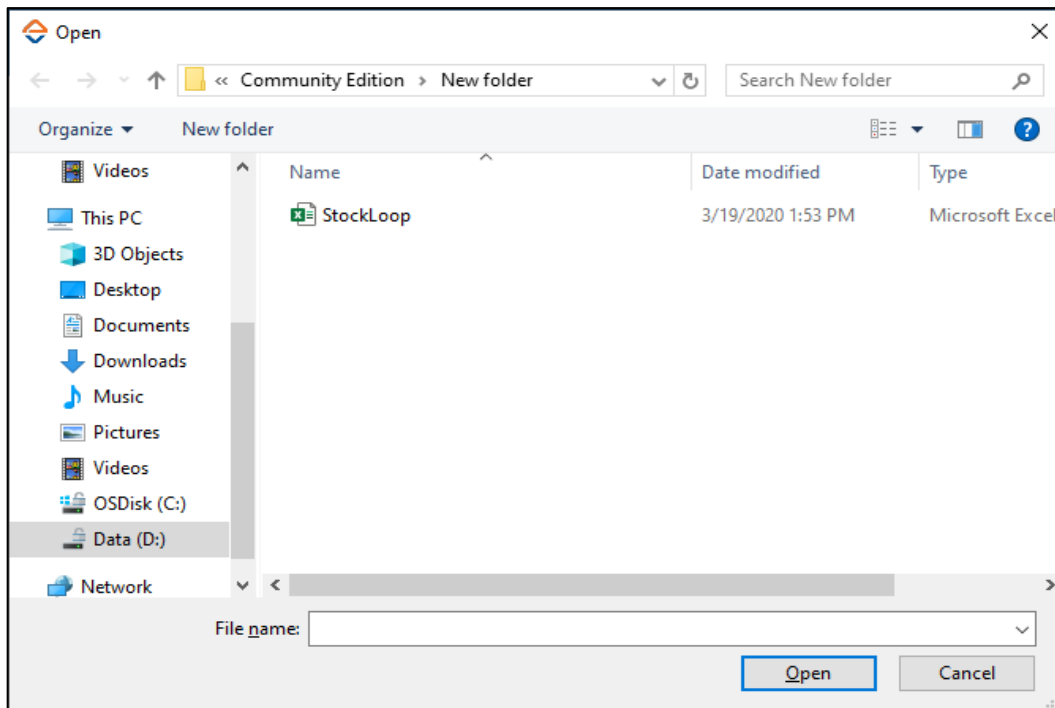
Excel loop activity is used to perform automation by reading values within an excel. In excel, there are number of rows and columns of the same nature and works in a loop where same set of actions are performed on each row available in the excel. Before using this activity, make sure you have an excel to read.

Using Excel Loop Activity

1. In the **Canvas Tools** pane, click **Files** to expand the tool and view the associated activities.
2. Drag the **Excel Loop** activity and drop on to the **Flowchart designer** on the **Canvas**



3. Click **Excel File Location**. The **Open** window appears.



Select the excel file you want to automate and click Open.

4. In the **Excel Sheet** list, select the sheet or workbook you want to perform automate.
5. Select the **Has headers** check box if selected excel has headers.

6. Click the  (**Settings**) icon. The **Output Parameters** dialog box appears to set data type and parameters.

Output Parameters ✕

Header Names	Data Type	Reuse Header Name <input checked="" type="checkbox"/> All	Parameter
Stock Name	Text	<input checked="" type="checkbox"/>	<u>StockName</u>
Stock Value	Text	<input checked="" type="checkbox"/>	<u>StockValue</u>
Risk	Text	<input checked="" type="checkbox"/>	<u>Risk</u>

CONFIRM

Output Parameters dialog box have four columns:

- a. **Headers Names:** In this column, you can view the headers name of the selected excel. This column is not editable.
 - b. **Data Type:** In this column, you can select the type of data for headers from the list. The available options are– Text, Numeric, Decimal, Boolean and DateTime.
 - **Text:** This data type is used to determine the value of the defined variable as text, such as, Dave, Stock.
 - **Numeric:** This data type is used to determine the value of the defined variable as Numeric, such as, 10, 88.
 - **Decimal:** This data type is used to determine the value of the defined variable as Decimal, such as, 10.01,9.04.
 - **Boolean:** This data type is used to determine the value of the defined variable as Boolean, such as, true, false.
 - **DateTime:** This data type is used to determine the value of the defined variable as date and time stamp, such as, 01122020;14:09:05.
 - c. **Reuse Header Name:** Select the **All** option if you want to reuse all the existing headers of the selected excel. You can even select the required headers individually by selecting the check box available in the headers row.
 - d. **Parameter:** In this column, headers parameter is available. You can edit the parameter name as per your requirement. These parameters are used to store the value and get the desired output. Parameter works like a variable.
7. Click **Confirm** to save the changes.

Excel Loop Properties

The properties of Excel Loop activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Loop Properties	
Excel File Location	The file path of the selected excel is displayed in this field. You can also define the file path directly in this field. The format to define the file is "D:\MST\Stock.xls".
Has Header	Select this option to use the existing headers of the selected excel.
Page Size	Defines the number of rows to be considered as a page in the Excel. Minimum value should be at least 1.
Range	Specified area is considered in the loop based on the range provided, for example, range A1:A15 consider column A and rows 1 to 15; Range A1:C5 consider Columns A,B and C and rows 1 to 5.
Save Option	Specify the Save Option for the results in the excel to be saved. There are four options to save the results – <ul style="list-style-type: none"> ▪ Cell: It saves the result after each cell. ▪ Row: It saves the result after each row. ▪ EndOfLoop: It saves the result once the entire loop is completed. ▪ None: It won't save the results in the excel. ▪
WorkSheet	Specify the name of the worksheet in the excel.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.

	In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Excel Loop . You can change the name as required.

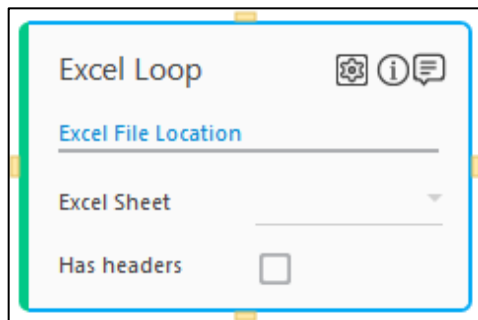
9.10.3 Break Loop

Break Loop activity is used within the Excel Loop activity or Advanced Loop activity to break an excel loop. This activity cannot be used as a standalone activity to break an excel loop.

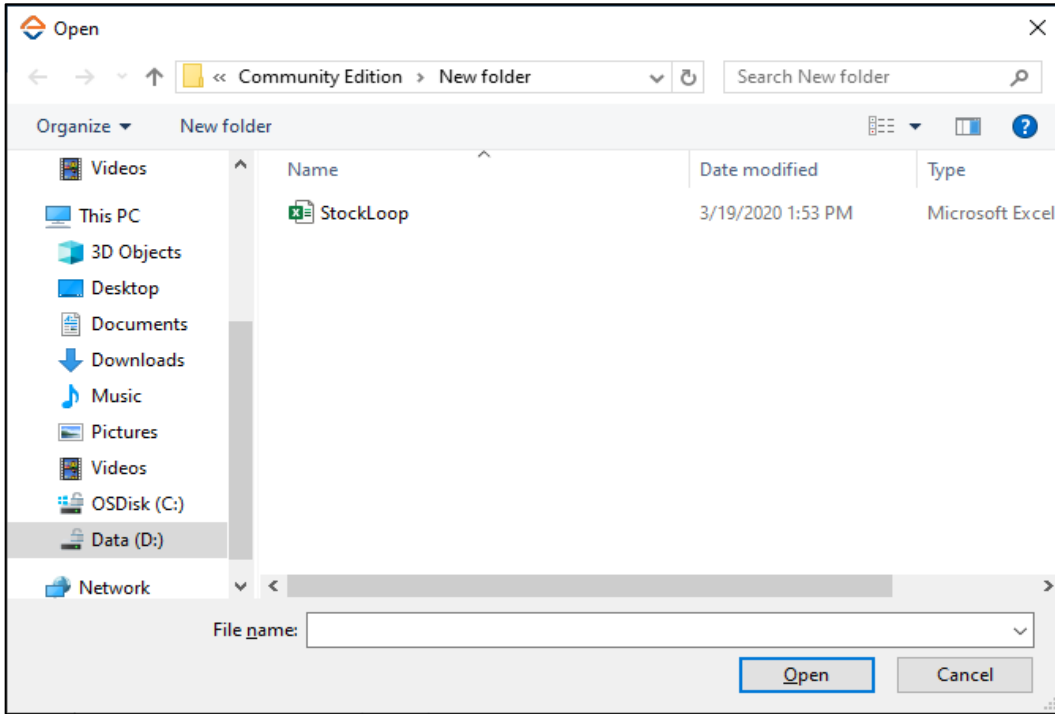
Using Break Loop

First, let's open the Studio and set up the Excel Loop activity.


1. In the **Canvas Tools** pane, click **Files** to expand the tool and view the associated activities.
2. Drag the **Excel Loop** activity and drop on to the **Flowchart designer** on the **Canvas**



3. Click **Excel File Location**. The Open window appears.



Select the excel file, you want to automate and click Open.

4. In the **Excel Sheet** list, select the sheet or workbook you want to automate.
5. Select the **Has headers** check box, if selected excel has headers.
6. Click the  (**Settings**) icon, an **Output Parameters** dialog box appears to set DataType and parameters.

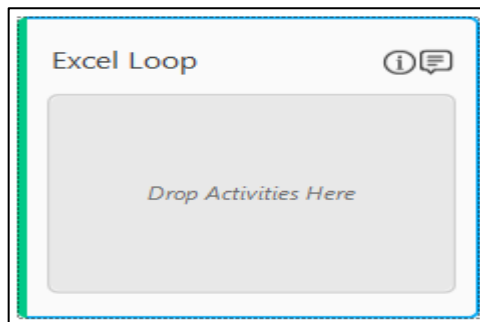


Output Parameters dialog box have four columns:

- a. **Headers Names:** In this column, you can view the headers name of the selected excel. This column is not editable.
 - b. **Data Type:** In this column, you can select the type of data for headers from the list. The available options are – Text, Numeric, Decimal, Boolean and DateTime.
 - **Text:** This data type is used to determine the value of the defined variable as text, such as, Dave, Stock.
 - **Numeric:** This data type is used to determine the value of the defined variable as Numeric, such as, 10, 88.
 - **Decimal:** This data type is used to determine the value of the defined variable as Decimal, such as, 10.01,9.04.
 - **Boolean:** This data type is used to determine the value of the defined variable as Boolean, such as, true, false.
 - **DateTime:** This data type is used to determine the value of the defined variable as date and time stamp, such as, 01/12/2020;14:09:05.
 - c. **Reuse Header Name:** Select the **All** option if you want to reuse all the existing headers of the selected excel. You can even select the required headers individually by selecting the check box available in the headers row.
 - d. **Parameter:** In this column, headers parameter is available. You can edit the parameter name as per your requirement. These parameters are used to store the value and get the desired output. Parameter works like a variable.
7. Click **Confirm** to save the changes.

Now, let's add the Break Loop activity within the Excel Loop activity.

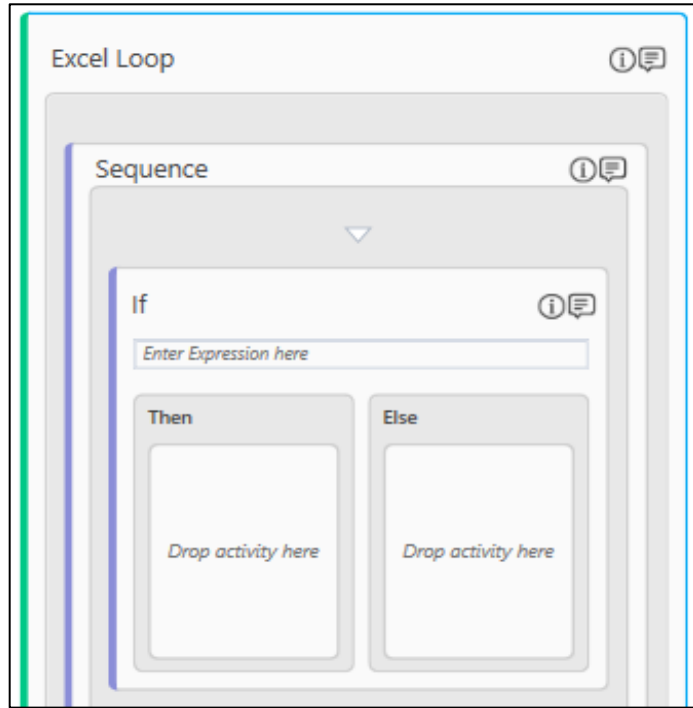
8. Double-click **Excel Loop** activity.



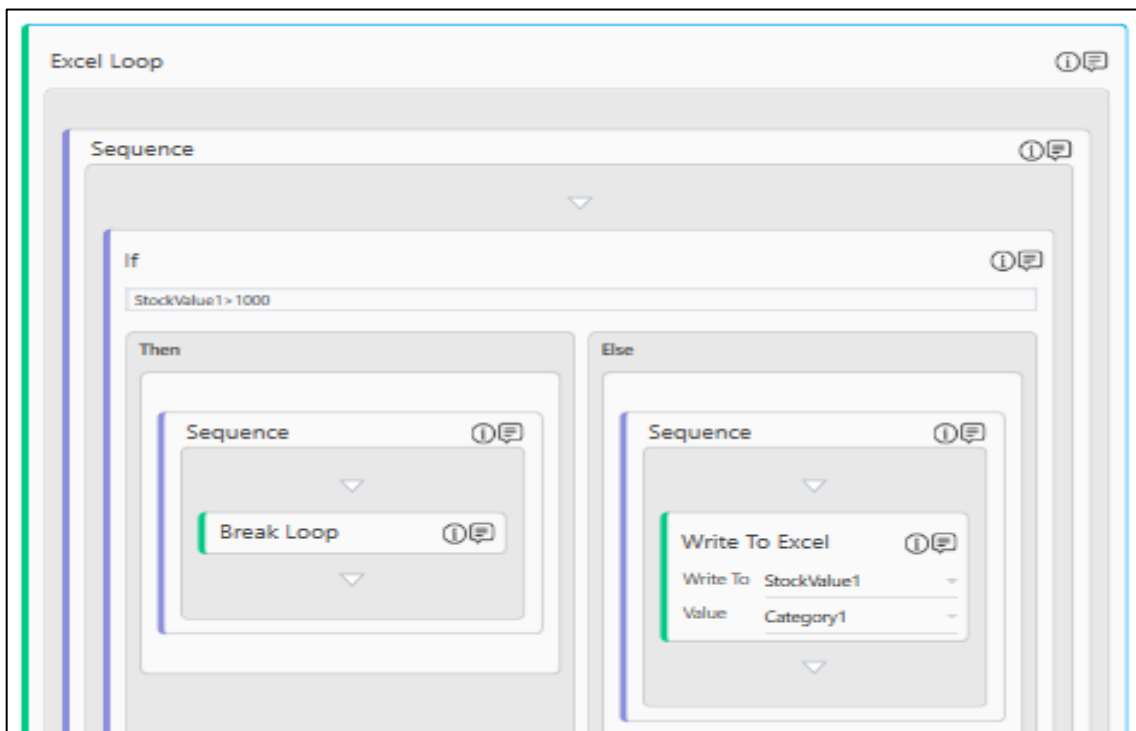
Note: In the steps below, we have included the If activity to define the end condition for break loop. You can use the Break Loop activity with any activity from automation studio, as per your business requirements.

9. In the **Canvas Tools** pane, click **Flow Controls** to expand the tool and view the associated activities.

10. Drag the **If** activity and drop on to the **Flowchart** designer on the **Canvas**, inside the **Excel Loop**.



11. Define **If-Then-Else** condition.
12. In the **Canvas Tools** pane, click **Files** to expand the tool and view the associated activities.
13. Drag the **Break Loop** activity and drop on to the **Flowchart** designer on the **Canvas** inside **Then** or **Else** condition, as per your process requirement.



Break Loop Properties

The properties of Break Loop activity are listed in the following table and can be edited in the Properties grid on the right pane.


Property Name	Usage
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity.</p> <p>If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Break Loop . You can change the name as required.

9.10.4 Use Case

Let us consider, you have an excel with company's name in one column and company's stock value in another column, you are required to update the stock prices on daily basis and sent it to the relevant team.

In Automation Studio, you can create a process to retrieve company's stock price.

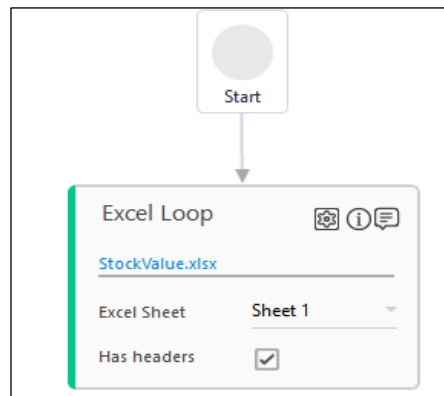
Prerequisite:

1. In the **Admin** menu, add a **WebApps** application.
2. In the Application Properties panel:
 - Enter the **Login URL** as <https://money.rediff.com/companies/Infosys-Ltd/13020007?query=Infosys+Ltd+>
 - Enter the **Search URL** as <https://money.rediff.com/companies/Infosys-Ltd/13020007?query=Infosys+Ltd+>
 - Enter Display Name as RediffMoney1.
 - Select preferred browser from the **Preferred Browser** list.
3. Click the  (**Save Properties**) icon to save the application details.

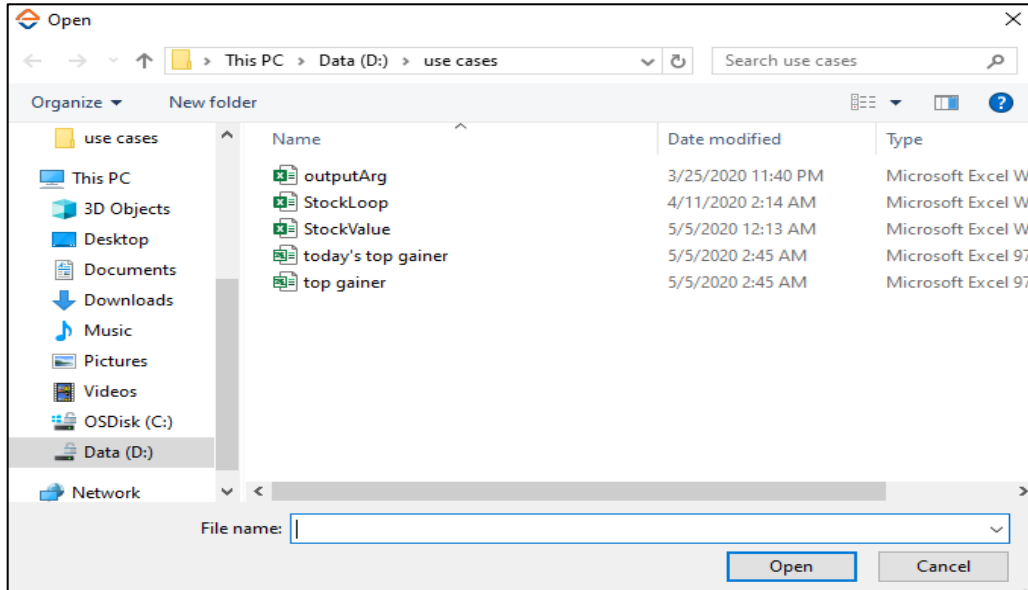
The WebApps application is created.

To automate the process of retrieving stock prices:


4. Create a new process.
5. From the Canvas Tools panel, add Excel Loop activity to the Flowchart designer on the Canvas

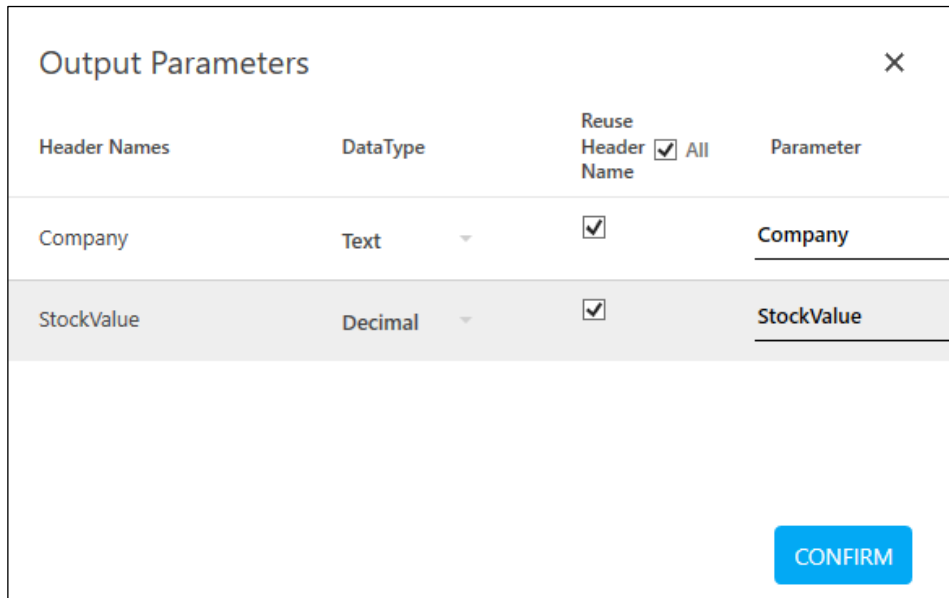


- Click **Excel File Location**. The **Open** window appears.



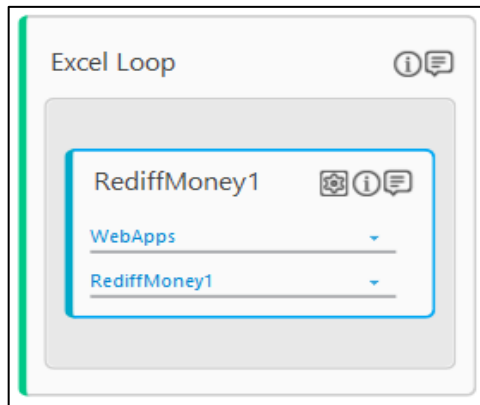
Select the **StockValue** file and click **Open**.



- In the **Excel Sheet** list, select **Sheet 1**.
 - Select **Has Headers** option.
6. Click the  (**Settings**) icon. The **Output Parameters** window appears.

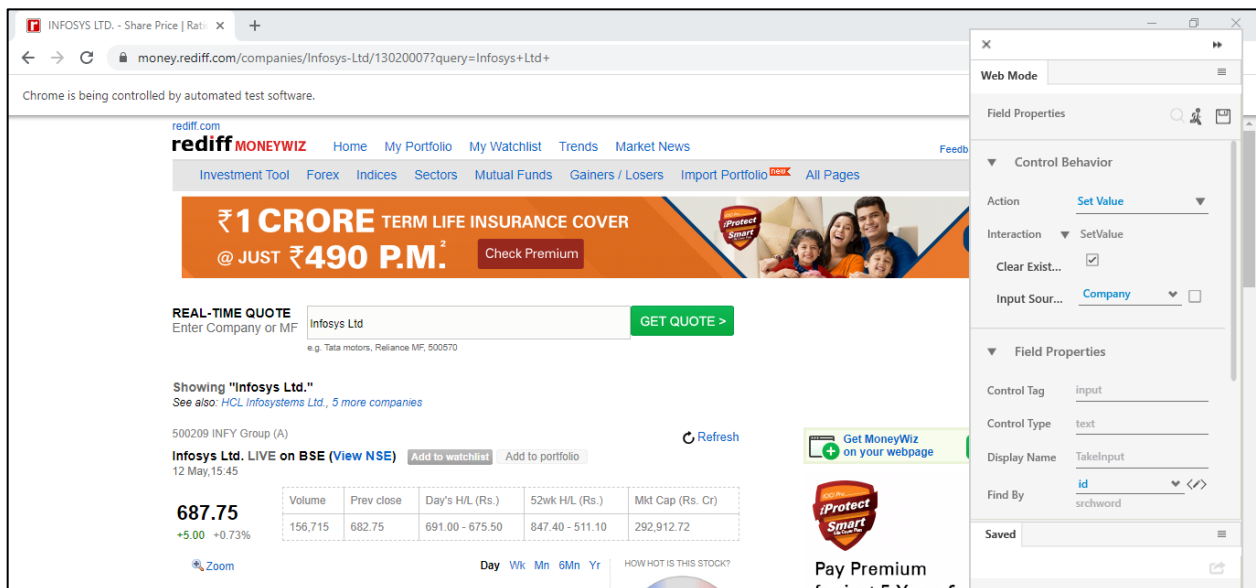



- Select **All** option to Reuse Header Name.
- For **Company**, select **DataType** as **Text** and **Parameter** as **Company**.
- For **StockValue**, select **DataType** as **Decimal** and **Parameter** as **StockValue**.
- Click **CONFIRM**.

7. Double-click **Excel Loop** activity on the **Canvas**.
8. From the Canvas Tools panel, add Application activity to the Flowchart designer on the Canvas

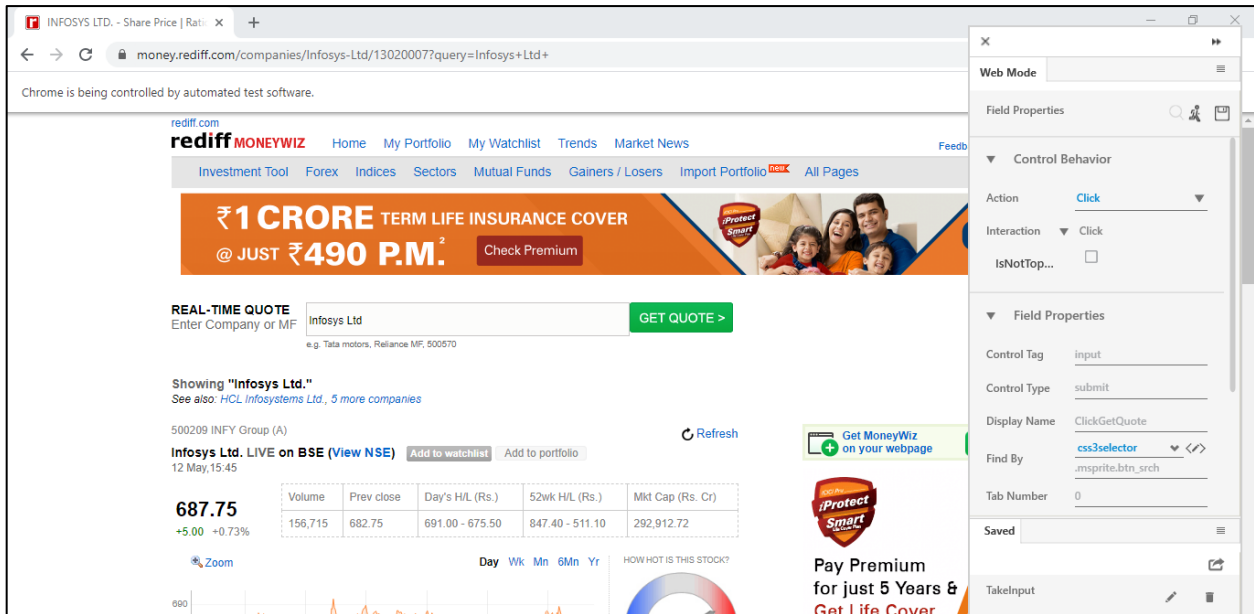


- In the Application Type list, select WebApps.
 - In the Select An Application list, select RediffMoney1.
9. Click the  (**Settings**) icon, a multimodal interface with selected application appears.
 10. In the **WEB MODE** interface, click the  (**Play**) icon.
 11. Select the **Search Textbox** field. The **Field Properties** pane of the **Web Mode** appears.



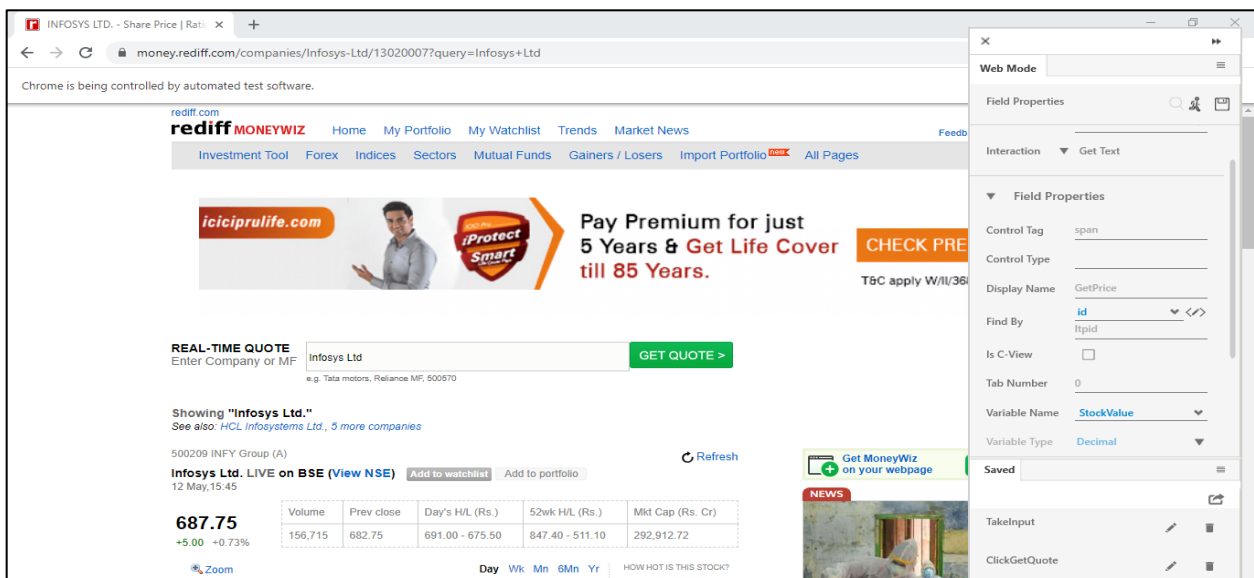
- In the **Action** list, select **Set Value**.
- Select Clear Existing Data option.
- In the **Input Source** list, select **Company**.
- Enter Display Name as TakeInput.
- Click the  (**Save Properties**) icon.

12. Select the **GET QUOTE** button. The **Field Properties** pane of the **Web Mode** appears.





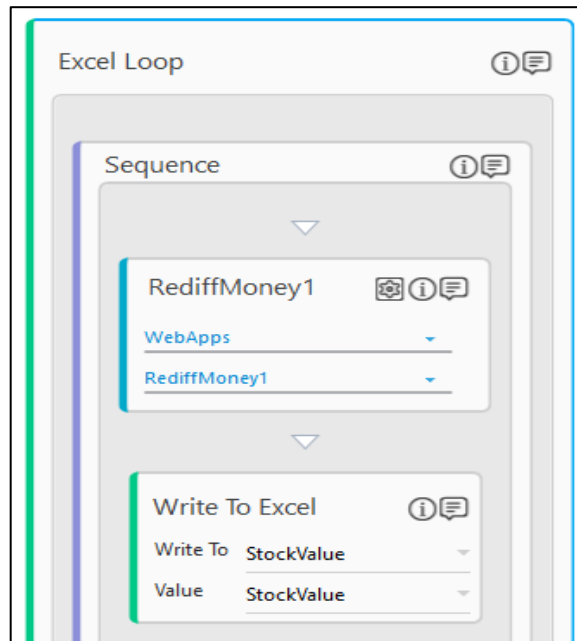
- In the **Action** list, select **Click**.
- Enter Display Name as ClickGetQuote.
- Click the **Save Properties** icon.

13. Select the Stock Value. A **Web Mode** pane appears.



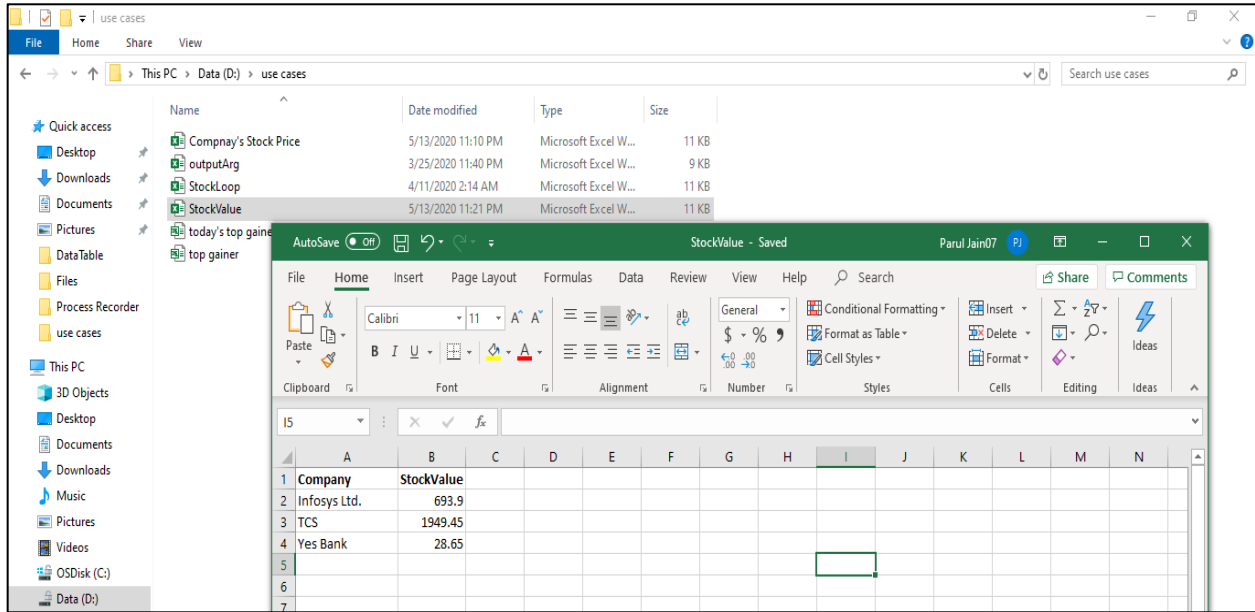
- In the **Action** list, select **Get Text**.
- Enter Display Name as GetPrice.
- In the **Variable Name** list, select **StockValue**.

- Click the  (**Save Properties**) icon .
14. Click the  (Save Configured fields and return to studio) icon.
 15. From the Canvas Tools panel, add Write to Excel activity to the Flowchart designer on the Canvas



- In the **Write To** list, select **StockValue**.
 - In the **Value** list, select **StockValue**.
16. Click **Save** and enter the **Process Name**.
 17. Setup the environment and then perform test run.

To view the output of this example, open StockValue excel.



9.10.5 Virus Scan

This activity uses the existing antivirus on the system to automate the scan of a file or a folder. The VirusScanProps.xml configuration file must be present in the Automation Studio > bin folder, else this activity does not work. The activity invokes the virus scanner client from the command line.

Prerequisites

To use the file Virus Scan feature, update the **VirusScanProps.xml** file that is present within the **AutomationStudio** folder.

To update the VirusScanProps.xml file:

1. Access AutomationStudio > bin and locate VirusScanProps.xml file.
2. Right click and click **Edit** to open the file.

```
<?xml version="1.0" encoding="utf-8" ?>
<!-- This file represents a fragment of a virus scan default values -->
<virusscan>
  <attributes>
    <attribute key='ClientScanEngineExePath' value="C:\Program Files (x86)\Symantec\Symantec Endpoint Protection\DoScan.exe"/>
    <attribute key='OpensScanWindow' value="true"/>
    <attribute key='PostArguments' value="&quot;"/>
    <attribute key='PreArguments' value="&quot;"/>
    <attribute key='VirusEngineOpenWindowProcessName' value="SavUI"/>
    <attribute key='WaitForTimeOutInMinutes' value="180"/>
    <attribute key='WaitTimeForFileMovementinMilliSecs' value="100"/>
  </attributes>
</virusscan>
```

The VirusScanProps.xml file opens.

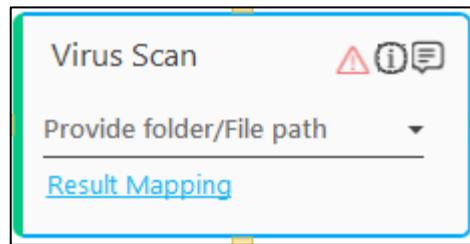
3. Update **ClientScanEngineExePath** with the path of .exe file of the antivirus installed in your server.

4. Update **WaitForTimeOutInMinutes** with the value of time out session in minutes.
5. Update **WaitTimeForFileMovementinMilliSecs** with the value of the wait time required for file movement in milliseconds.
6. Save and close the file.

The VirusScanProps.xml file is updated.

Using Virus Scan Activity

1. In the Canvas Tools pane, click **File** to expand the tool and view the associated activities.
2. Drag the **Virus Scan** activity and drop on to the Flowchart designer on the Canvas. The validation error symbol disappears when required inputs are provided.



3. In the **Provide folder/File path** list, select the parameter holding the source path of the file along with the file name or the path of the folder to perform the scan. You must define the parameter in the Parameter bar to store the value and make it available for the selection.
4. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

The Virus Scan activity with a default name is created.

Virus Scan Properties

The properties of Virus Scan activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as Virus Scan . You can change the name as required.

Step-By-Step Guide to Use Virus Scan to Invoke Existing Antivirus on the System to Automatically Scan a Folder

Let's see an example to invoke existing antivirus on the system to automatically scan a required folder.

Prerequisite:

Make sure the VirusScanProps.xml configuration file is present in the Automation Studio > bin folder, else the process would not work

To scan the required folder:

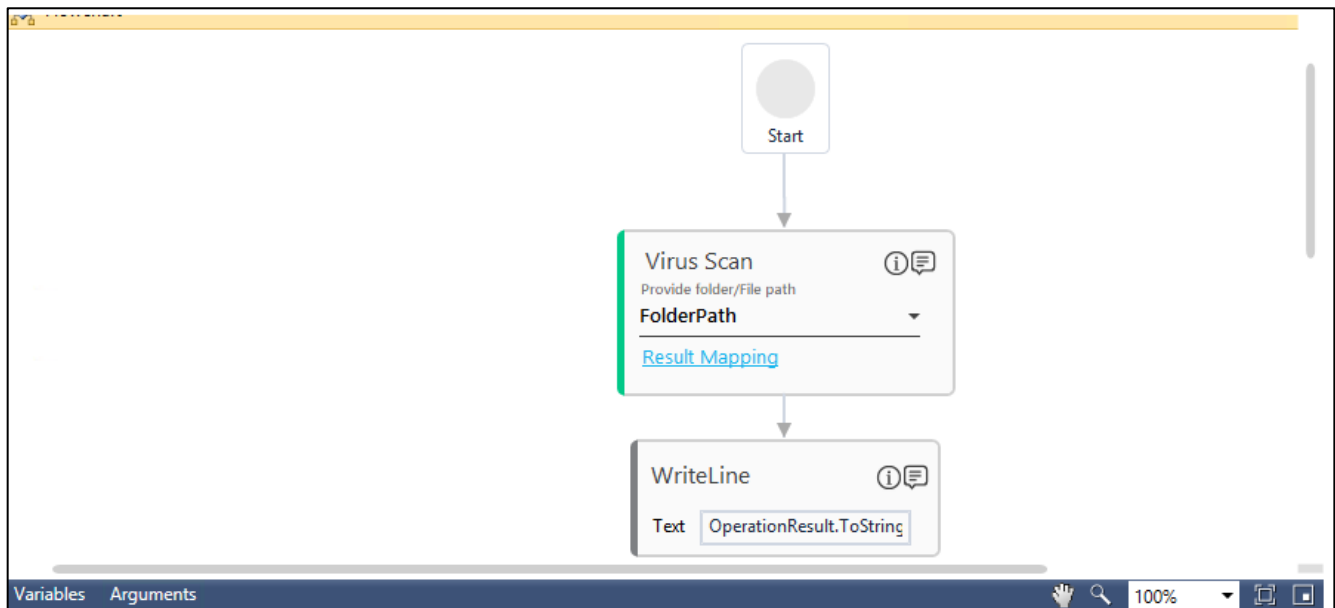
1. In the Canvas Tools pane, click **File** to expand the tool and view the associated activities.
2. Drag the **Virus Scan** activity and drop on to the Flowchart designer on the Canvas
3. In the **Parameter** bar, create an **In** argument, **FolderPath** of **String** type to pass the folder path of the required folder where the virus scan must be performed. Define the input value.

- In the **Parameter** bar, create an **In** argument, **OperationResult** of **Boolean** type to store the result of the operation performed.

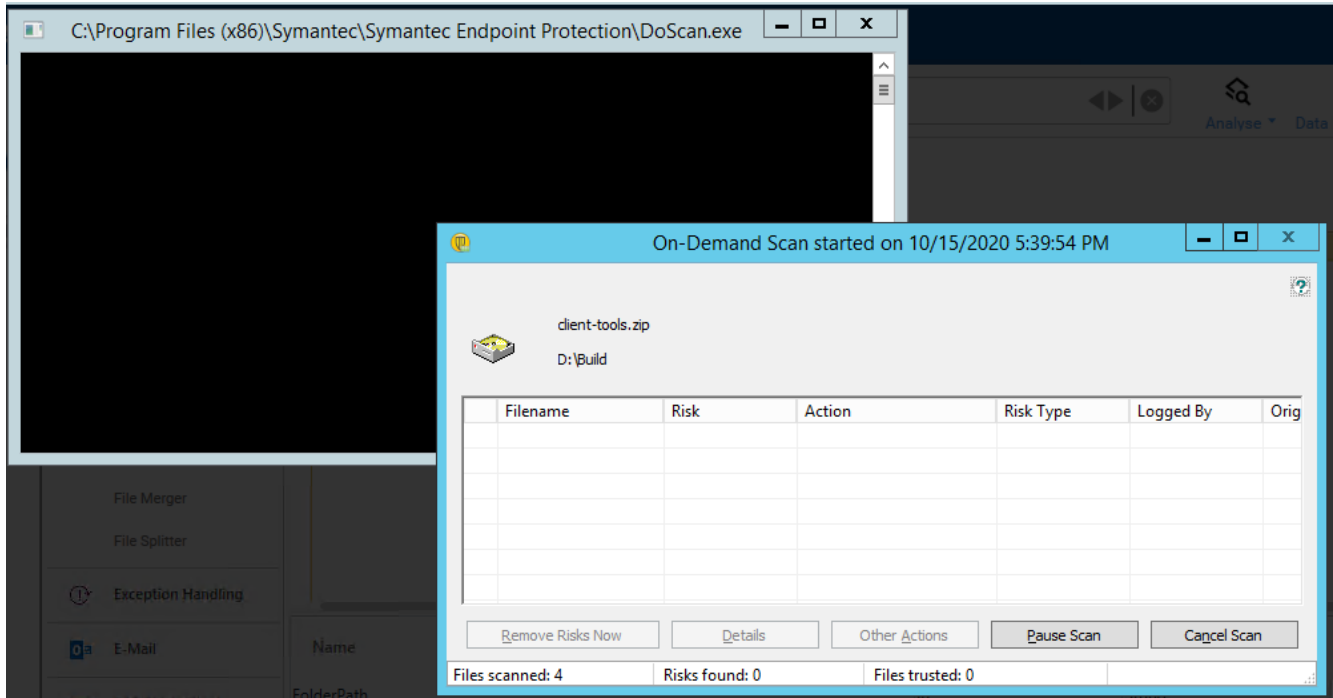
Name	Direction	Argument type	Default value
FolderPath	In	String	"D:\Build\Certificates"
OperationResult	In	Boolean	Enter a VB expression

Variables Arguments 100%

- In the **Provide folder/File path** list, select the **FolderPath** argument created above.
- Click the **Result Mapping** link. The **Output Mapping** list appears. Select the **OperationResult** argument created above. To view the output in Automation Studio, let's add **WriteLine** activity. You can publish, deploy and assign this process to a robot if you want to execute the process outside Automation Studio.
- Add a **WriteLine** activity and in the **Text** field, enter the **OperationResult** argument created above to store the result of search operation performed. Below is the sample automation process workflow created:



- Save the process.
- Setup the environment and then perform test run. Below is the sample screen shot of the command line invoking the virus scan and the virus scan performing the scan displaying the related details:



The Output console of Automation Studio displays the virus scan operation result:



9.10.6 File Splitter

This activity helps you to automate the task of splitting two or more files of same file type. The supported file types are PDF (up to version 1.6), MS Word (doc, docx, docm) and MS Excel (xlsx, xls, xlsx).

Word and PDF files can be split based on the specified range of page numbers while an Excel file can be split based on the specified sheet name or number of rows.

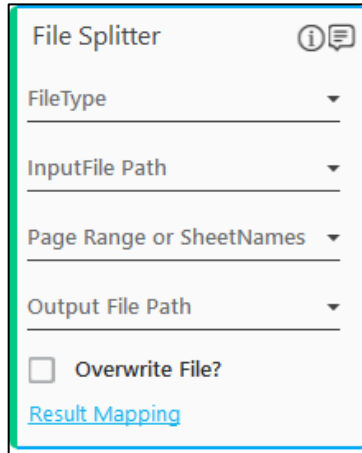
The page range must be in the num-num format. For example, page range from 13 to 51 must be specified as 13-51. Any other format of page range displays an error during process execution.

The sheet name must be provided as semi-comma separated values. For example, Sheet1;Sheet3, without any spaces between the names. The sheet names are case sensitive.

Using File Splitter Activity

1. In the Canvas Tools pane, click **File** to expand the tool and view the associated activities.

2. Drag the **File Splitter** activity and drop on to the Flowchart designer on the Canvas. The validation error symbol disappears when required inputs are provided.



3. In the **FileType** list, select the type of files that you want to split. Available options are:
 - [pdf](#)
 - [word](#)
 - [excel-sheetwise](#)
 - [excel-rowwise](#)

For pdf, word or excel-sheetwise FileType

- a. In the **InputFile Path** list, select the parameter holding the document file path along with the file name and its extension, for example, D:/documents/sample_doc.docx . You must define the parameters in the **Parameter** bar to store the value and make it available for the selection.
- b. In the **Page Range or SheetNames** list, select the parameter holding the page numbers of the word/pdf file and sheet name if the file is an excel. You must define the parameters in the **Parameter** bar to store the value and make it available for the selection.
- c. In the **Output File Path** list, select a parameter holding the output document file path along with the file name of the split file.

For excel-rowwise FileType

- a. In the **InputFile Path** list, select the parameter holding the document file path along with the file name and its extension. For example, D:/documents/sample_doc.docx . You must define the parameters in the **Parameter** bar to store the value and make it available for the selection.

The screenshot shows a configuration window titled "File Splitter". It contains the following elements:

- FileType:** A dropdown menu currently set to "excel-rowwise".
- InputFile Path:** A dropdown menu.
- Sheet Name:** A dropdown menu.
- Output Folder Path:** A dropdown menu.
- Header row number:** A dropdown menu.
- Split by:** A dropdown menu.
- Number of files to split into:** A dropdown menu.
- Remove styles:** An unchecked checkbox.
- Overwrite File?:** An unchecked checkbox.
- Result Mapping:** A blue hyperlink at the bottom.

- b. In the **Sheet Name** list, select the parameter holding the name of the sheet. The sheet name must be a single sheet name and not multiple sheets. You must define the parameters in the **Parameter** bar to store the value and make it available for the selection.
- c. In the **Output Folder Path** list, select the parameter holding the folder name to save the files after row wise split. You must define the parameters in the **Parameter** bar to store the value and make it available for the selection. The default value is set as **1**.
- d. In the **Header row number** list, select the parameter holding the number at which header column is present. The rows after the header row is used for splitting. You must define the parameters in the **Parameter** bar to store the value and make it available for the selection.
- e. In the **Split By** list, select the option by which you want to perform the split. You can split the file based on the **Number of Rows** or **Number of Files** specified.
- f. In the **Number of files to split into/Number of rows to split into** list, select the parameter holding the number of files or number of rows you want to split the file into. You must define the parameters in the **Parameter** bar to store the value and make it available for the selection.
- g. Select the **Remove styles** check box if you want to remove styles applied to the content of the file to simplify the data for splitting.
- h. Select the **Overwrite File?** check box to overwrite an existing file if you want to create the merged file with the name of an existing file.
- i. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

The File Splitter activity with a default name is created.

File Splitter Properties

The properties of File Splitter activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
AllowedNumberOfFoldersToBeCreatedAlternatively	Number of folder levels limit that can be created to place the output file . By default, the value is set to 2 . It is recommended to keep the folder level limit less than 5.
Breakpoint	Select this option to mark this activity as a pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as File Splitter . You can change the name as required.
HeaderRowNumber	The number at which header column is present. The rows after the header row is used for splitting.
NumOfFiles	The number of files you want to split the excel file into.
NumOfRows	The number of rows you want to split the excel file into.
NumParameter	The number of rows to split the excel row wise.
OutFolderPath	The folder name to save the files after row wise excel file split.

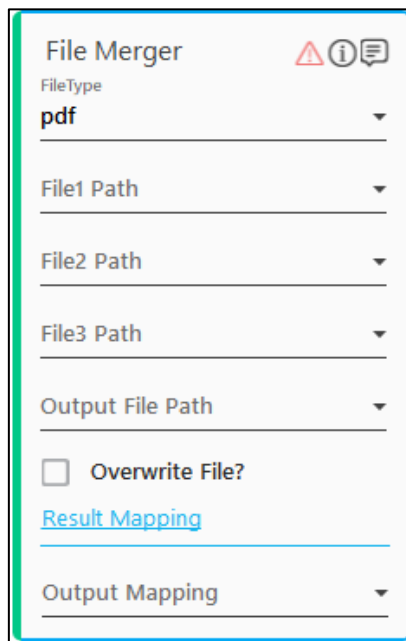
Property	Usage
PasteAsValues	Paste value or paste the formatted values while splitting the excel row-wise.
RowsOrFiles	Signifies if you want to split the excel file row wise or file wise.
SheetToSplitRowWise	Signifies if you want to split the excel file sheet wise or row wise.

9.10.7 File Merger

This activity helps you to automate the task of merging two or more files of same file type. The supported file types are PDF (up to version 1.6), MS Word and MS Excel.

Using File Merger Activity

1. In the **Canvas Tools** pane, click **File** to expand the tool and view the associated activities.
2. Drag the **File Merger** activity and drop on to the **Flowchart designer on the Canvas**. The validation error symbol disappears when required inputs are provided.



3. In the **FileType** list, select the type of files you want to merge. Available options are- **pdf**, **word** and **excel**.
4. In the **File1 Path**, **File2 Path** and **File3 Path** list, select the parameters holding the respective document file path along with the file name and file extension. For example, D:\documents\sample_doc.docx . You must define the parameters in the Parameter bar to store the respective values and make it available for the selection.
5. In the **Output File Path** list, select a parameter holding the file path along with the file name with which you want to save the merged file.

6. Select the **Overwrite File?** check box to overwrite an existing file if you want to create the merged file with the name of an existing file.
7. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

The File Merger activity with a default name is created.

File Merger Properties

The properties of File Merger activity are listed in the following table and can be edited in the **Properties** grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as File Merger . You can change the name as required.

Step-By-Step Guide to Use File Merger to Merge Different Excel Files

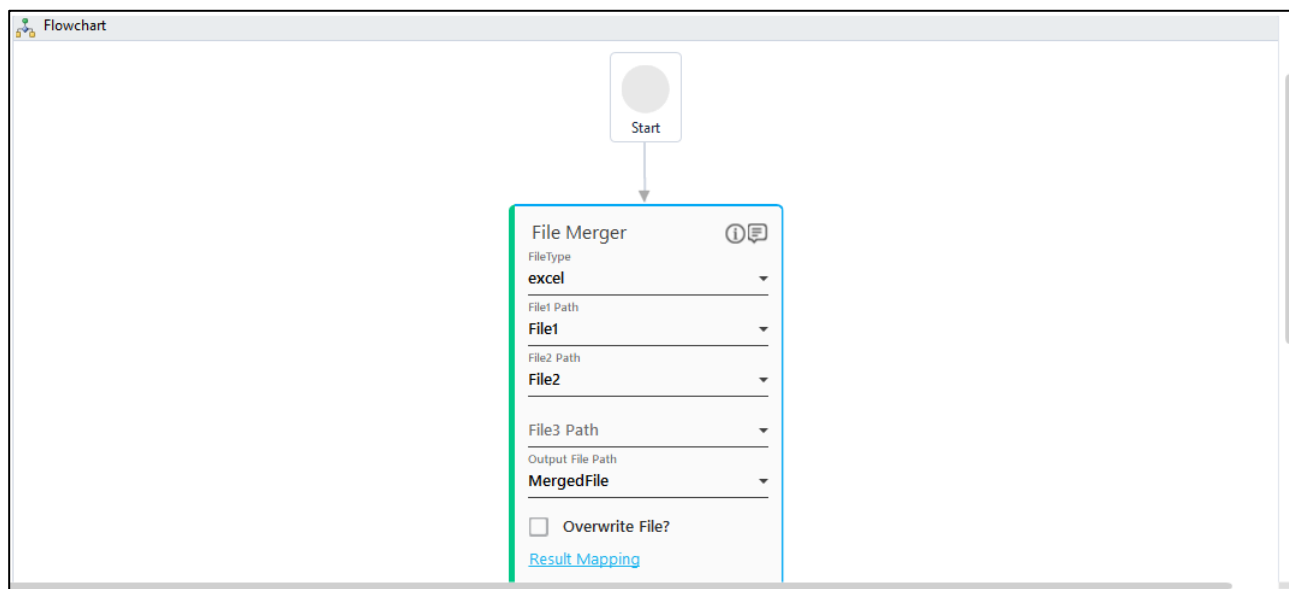
Let's see an example of merging two Excel files with different kind of data and see how the merged files looks like.

To merge two Excel files:

1. In the **Canvas Toolspane**, click **File** to expand the tool and view the associated activities.
2. Drag the **File Merger** activity and drop on to the **Flowchart designer on the Canvas**
3. In the **Parameter bar**, create **In** arguments of type **String** and define the input values:
 - **File1** - to pass the file path of the first file along with the file name and file extension as input.
 - **File2** - to pass the file path of the second file along with the file name and file extension as input.
 - **MergedFile** - to store the merged file. Define the destination folder and the file name along with the file extension to save the merged file with the specified name.

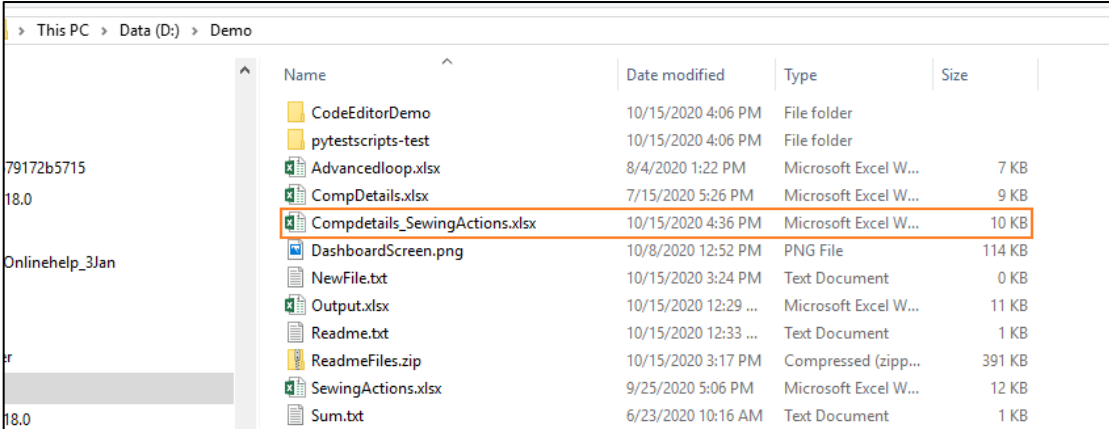
Name	Direction	Argument type	Default value
File1	In	String	"D:\Demo\CompDetails.xlsx"
File2	In	String	"D:\Demo\SewingActions.xlsx"
MergedFile	In	String	"D:\Demo\Compdetails_SewingActions.xlsx"

4. In the **FileType** list, select **excel**.
5. In the **File1 Path**, select the **File1** argument created above.
6. In the **File2 Path**, select the **File2** argument created above.
7. In the **Output File Path** list, select the **MergedFile** argument created above. Below is the sample automation process workflow created:



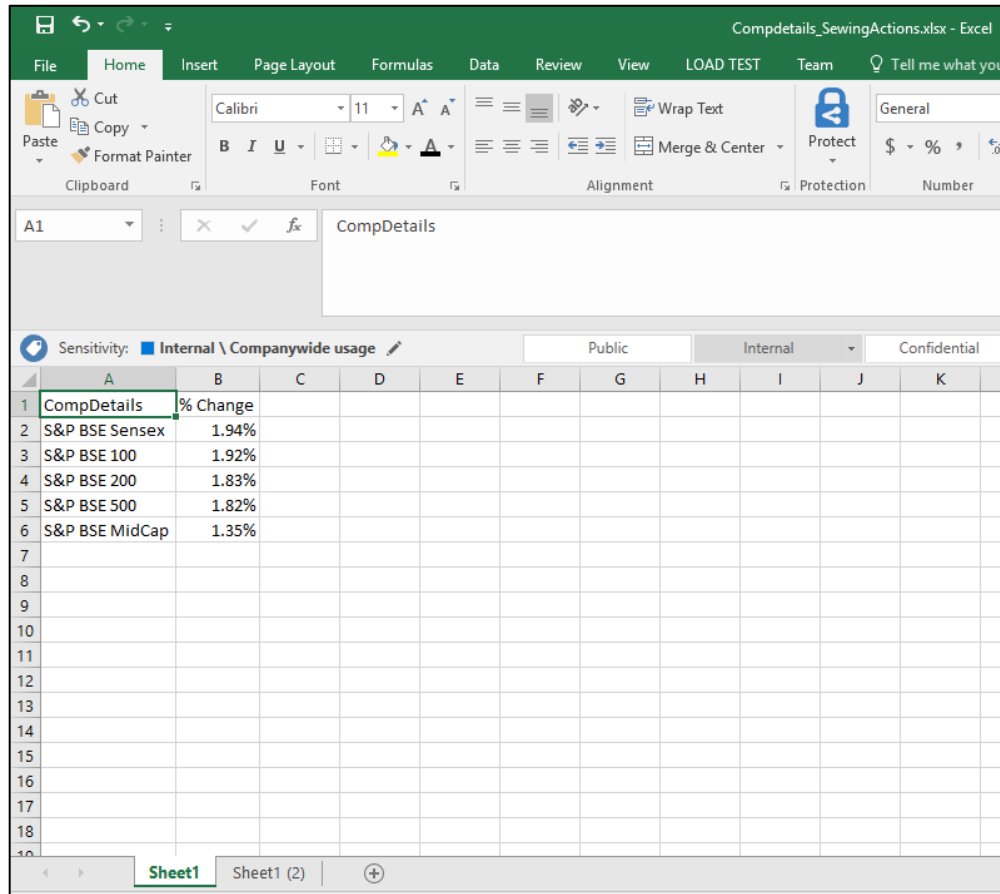
8. Save the process.
9. Setup the environment and then perform test run. You can publish, deploy and assign this process to a robot if you want to execute the process outside Automation Studio.

Observe that an Excel file, MergedFile.xlsx gets created at the specified location:



Name	Date modified	Type	Size
CodeEditorDemo	10/15/2020 4:06 PM	File folder	
pytestscripts-test	10/15/2020 4:06 PM	File folder	
Advancedloop.xlsx	8/4/2020 1:22 PM	Microsoft Excel W...	7 KB
CompDetails.xlsx	7/15/2020 5:26 PM	Microsoft Excel W...	9 KB
Compdetails_SewingActions.xlsx	10/15/2020 4:36 PM	Microsoft Excel W...	10 KB
DashboardScreen.png	10/8/2020 12:52 PM	PNG File	114 KB
NewFile.txt	10/15/2020 3:24 PM	Text Document	0 KB
Output.xlsx	10/15/2020 12:29 ...	Microsoft Excel W...	11 KB
Readme.txt	10/15/2020 12:33 ...	Text Document	1 KB
ReadmeFiles.zip	10/15/2020 3:17 PM	Compressed (zipp...	391 KB
SewingActions.xlsx	9/25/2020 5:06 PM	Microsoft Excel W...	12 KB
Sum.txt	6/23/2020 10:16 AM	Text Document	1 KB

The files merged files contains two different sheets corresponding to each of the Excel file:



9.10.8 File Operations

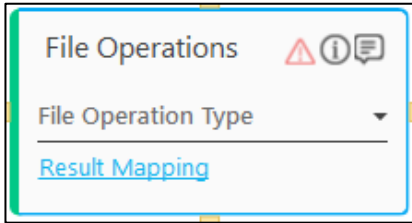
It allows you to automate file operations such as create, move, rename, zip and others. A few of the operations such as open operation, are limited to specific file types as the specific application would be required to open the file.

The application associated with the file type must be present in the system for the automation process workflow to perform the required file operation. For example, if you want to perform a file operation on an excel file, ensure that you have MS Excel installed on the system.

This activity can be used for files available on the network drive as well.

Using File Operations Activity

1. In the Canvas Tools pane, click **File** to expand the tool and view the associated activities.
2. Drag the **File Operations** activity and drop on to the Flowchart designer on the Canvas. The validation error symbol disappears when required inputs are provided.

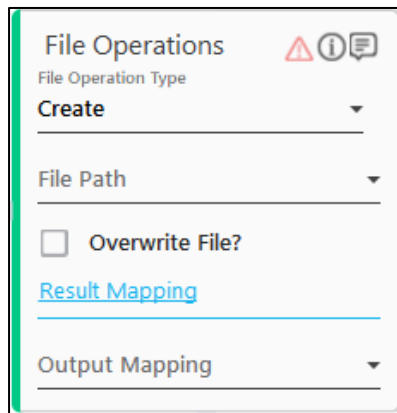


3. In the **File Operation Type** list, select the operation you want to perform. Click any of the link below to know the steps to perform the selected operation:

- [Create](#)
- [Copy](#)
- [Move](#)
- [Rename](#)
- [Open](#)
- [Save As](#)
- [Close](#)
- [Delete](#)
- [Zip](#)
- [Unzip](#)
- [Convert](#)
- [IsExist](#)

Create

It helps you to automate the creation of a new file and store it at a specified location, with a desired extension such as .doc, .docx, .txt, .jpg, .html and others.



1. In the **File Path** list, select the parameter holding the document file path along with the file name and file extension to create and save the file at the specified location with the specified name. For example, D:\documents\sampldoc.txt creates a new file **sampledoc** of **.txt** file type and saves it in the **documents** folder available at the **D** drive. You must define the parameters in the Parameter bar to store the value and make it available for the selection.
2. Select the **Overwrite File?** check box to overwrite an existing file, if you want to create the file with the name of an existing file.
3. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the

file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

Copy

It helps you to automate copying of one or multiple files from one location to another. You can use this operation to streamline high volume and complex files to copy from one location to another more efficiently, accurately and in less time.

You can copy a specific file from one location to another by providing the file path along with file name and file extension. For example, D:\Data\newfile.txt.

To copy multiple files from one location to another, you must provide only the file path. For example, D:\Data. You can either copy all the files in available in the folder or copy selective files of same file type by providing the file extension.

The screenshot shows a configuration window for 'File Operations'. The 'File Operation Type' is set to 'Copy'. The 'Source Path' field is empty. The 'File Extension' field is empty. The 'Destination Path' field is empty. The 'Append Parameters' field is empty. The 'Copy From Sub-Directories?' dropdown is set to 'true'. There is a 'Result Mapping' link at the bottom.

1. In the **Source Path** list, select the parameter holding the path of the file that needs to be copied. The path can either be the file location or file location along with the file name and its extension depending on the number of files you want to copy. You must define the parameter in the Parameter bar to store the value and make it available for the selection.
2. In the **File Extension** list, select the parameter holding the file name along with the file extension of the file or files that you want to copy. This is an optional field. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection. Following three combinations can be used.
 - *.*- to copy all the files from source location.
 - *.file extension - to copy all files with a specific extension. For example, *.xlsx is used if you want to copy all the excel files.
 - File name.File extension - to copy a file with a specific name and extension. For example, Results.xlsx.

3. In the **Destination Path** list, select the parameter holding the destination path of the file where it must be copied. For example, D:\UpdatedData. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection.
4. In the **Append Parameters** list, select the parameter holding the text that you want to add to the name of the file that is being copied. This is an optional field. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection.
5. In the **Copy From Sub-Directories?** list, select **true** to search for one or more files in the sub directories or sub folders of the source location path. By default, the value is set to **true**. If it is set to **false**, the required file is searched only in the specified source file path.
6. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

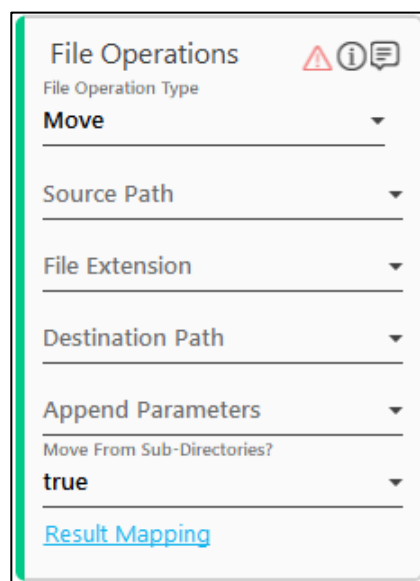
7.

Move

It helps you to automate moving of one or multiple files from one location to another. By automating the file transfer, you can prevent moving sensitive data to unwanted location and avoid end point vulnerabilities. It helps you to move multiple files with greater accuracy and in lesser time.

You can move a specific file from one location to another by providing the file path along with file name and file extension. For example, D:\Data\newfile.txt.

To move one or more files from one location to another, you must provide only the file path, for example, D:\Data.



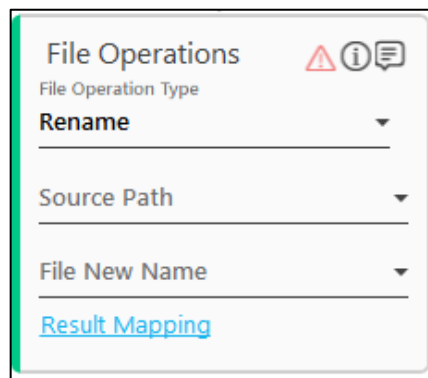
1. In the **Source Path** list, select the parameter holding the source path from where the file must be moved. The path can either be the file location or file location along with the file name and its extension depending on the number of files you

want to move. You must define the parameter in the Parameter bar to store the value and make it available for the selection.

2. In the **File Extension** list, select the parameter holding the file name along with the file extension of the file that you want to move. This is an optional field. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection. Following three combinations can be used.
 - ***.*** - to move all the files from source location.
 - ***.file extension** - to move all files with a specific extension like *.xlsx is used to copy all the excel files.
 - **File name.File extension** - to move a file with a specific name and extension. For example, Results.xlsx.
3. In the **Destination Path** list, select the parameter holding the destination file path. For example, D:\UpdatedData. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection.
4. In the **Append Parameters** list, select the parameter holding the text that you want to append to the name of the file or files that are being moved. This is an optional field. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection.
5. In the **Copy From Sub-Directories?** list, select **true** to search for the required file or files in the sub directories or sub folders of the source location path. By default, the value is set to **true**. If it is set to **false**, the required file or files are searched only in the specified source location path.
6. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

Rename

It helps you to automate renaming a file or a folder. Automating the task of renaming files or folders lets you handle multiple files in less time with greater accuracy and consistency.

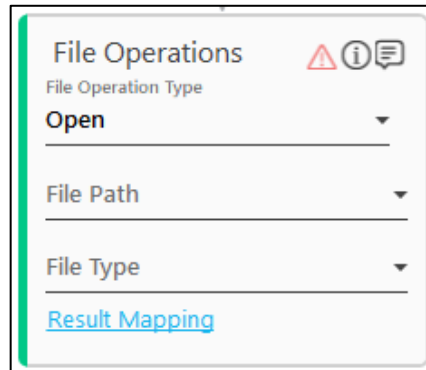


1. In the **Source Path** list, select the parameter holding the source path of the file along with the file name and file extension or the folder name that needs to be renamed. You must define the parameter in the Parameter bar to store the value and make it available for the selection.
2. In the **File New Name** list, select the parameter holding the name that you want to rename the file or the folder with. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection.

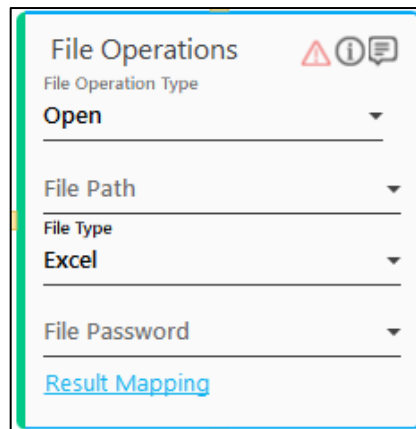
- Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

Open

It helps you to open a file automatically from a particular location.



- In the **File Path** list, select the Parameter holding the source path of the file with the file name and file extension. You must define the parameter in the Parameter bar to store the value and make it available for the selection.
- In the **File Type** list, select the type of file you want to open. Available options are- **Excel**, **Word** and **Others**.



- If the **File Type** selected is **Excel** or **Word**, the **File Password** list appears. In the **File Password** list, select the parameter holding the password to open the file if the file is password protected.
- Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

Save As

It helps you to automate saving an open file with a desired name and file extension. Automating the task of saving files prevents the possibility of losing any unsaved data due to system failure or human error.

The screenshot shows the 'File Operations' configuration panel. At the top, it says 'File Operations' with a warning, info, and help icon. Below that, 'File Operation Type' is set to 'Save As'. There are two dropdown menus: 'File Type' and 'Open Type'. Below these is a 'Destination Path' dropdown. At the bottom, there is a blue link labeled 'Result Mapping'.

1. In the **File Type** list, select the type of file you want to save. Available options are - **Word** and **Excel**.
2. In the **Open Type** list, choose if you want to open a **New** or an **Existing** file. **New** indicates that the file is not saved previously while **Existing** indicates that the file is already saved at a certain location.
3. If the **Open Type** selected is **New**, the **File Name** list appears.
 - In the **File Name** list, select the parameter holding the file name along with the file extension with which you want to save the file.

This screenshot is similar to the previous one, but the 'Open Type' dropdown is now set to 'New'. Consequently, a 'File Name' dropdown menu has appeared below the 'File Type' dropdown. The 'Destination Path' dropdown remains visible below it. The 'Result Mapping' link is still at the bottom.

4. If the **Open Type** list selected is **Existing**, the **Source Path** list appears.
 - In the **Source Path** list, select the parameter holding the source path of the file along with the file name and file extension of the file that you want to save.

5. In the **Destination Path** list, select the Parameter holding the destination path of the file along with the file name and file extension with which you want to save the file. The new name must not be same as the previous name.
6. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

Close

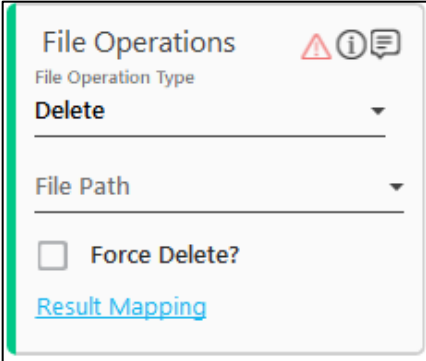
It helps you to automatically close a specified file. Automating the task of closing files can avoid losing data that might occur during some system failures if the file remains open.

1. In the **File Type** list, select the type of file you want to save. Available options are- **Word** and **Excel**.
2. In the **File Path** list, select the parameter holding the source path of the file with the file name and file extension. For example, D:\Data\newfile.txt. You must define the parameter in the Parameter bar to store the value and make it available for the selection.
3. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be

used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

Delete

It helps you to automatically delete the specified file. Automating the task of deleting files allows you to get rid of unwanted files without any human intervention, saving the disk space and time. Since the deletion of file happens in the background, other task can be performed simultaneously.



The screenshot shows a configuration window for a 'File Operations' task. The window has a title bar with a warning icon, an information icon, and a help icon. Below the title, there is a 'File Operation Type' dropdown menu with 'Delete' selected. Below that is a 'File Path' dropdown menu. At the bottom, there is a checkbox labeled 'Force Delete?' which is currently unchecked. A blue link labeled 'Result Mapping' is located at the bottom of the window.

1. In the **File Path** list, select the parameter holding the source path of the file with the file name and file extension. For example, D:\Data\newfile.txt. You must define the parameter in the Parameter bar to store the value and make it available for the selection.
2. Select the **Force Delete?** check box to forcefully delete a file that is in open state.
3. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

Zip

It helps you to automatically zip one or more files inside a folder. Automating the task of zipping files facilitates data organization and transfer accurately and efficiently. It saves the administrative time and effort required to manage multiple files transfer.

1. In the **Source Path** list, select the parameter holding the source path of the folder or the file for zipping. The path can either be the file location or file location along with the file name and its extension depending on whether you want to zip a folder or a file. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection.
2. In the **Destination Path** list, select the parameter holding the destination path where the selected file needs to be zipped. For example, D:\UpdatedData. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection.
3. In the **File Name** list, select the parameter holding the name of the zipped folder to be created. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection.
4. In the **Zip Format** list, select **.zip** as the format of the zip file/folder. The supported format is only **.zip**.
5. Select the **Password** check box if you want to password protect the created zipped folder. You must define the parameter, holding the password, in the **Parameter** bar and select the parameter in the **Password** list.
6. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

Unzip

It helps you to automatically extract archives. Automating the task of unzipping the compressed folder lets you unzip the file or folder without requiring a specific software to open the archive. It saves the time spent in manually extracting the files.

The screenshot shows a configuration window for 'File Operations'. At the top right, there are three icons: a warning triangle, an information 'i' icon, and a speech bubble icon. The main title is 'File Operations'. Below it, 'File Operation Type' is set to 'Unzip'. There are three dropdown menus: 'Source Path', 'Destination Path', and 'Password' (which has an unchecked checkbox next to it). At the bottom, there is a blue link labeled 'Result Mapping'.

1. In the **Source Path** list, select the parameter holding the source path of the folder or the file for unzipping. The path can either be the file location or file location along with the file name and its extension depending on whether you want to zip a folder or a file. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection.
2. In the **Destination Path** list, select the parameter holding the destination path where the selected file or files have to be unzipped. For example, D:\UpdatedData. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection.
3. Select the **Password** check box if you want to password protect the unzipped folder. You must define the parameter, holding the password, in the **Parameter** bar and select the parameter in the **Password** list.
4. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

Convert

It helps you to automatically convert one file type to another, thus having all the conversions consistent every time. It allows you have ideal file formats for security and shareability. It saves time and effort for ongoing, constant file conversion requirements.

File Operations (i) (m)

File Operation Type

Convert ▼

Source File Type ▼

Source File Path ▼

Destination Folder Path ▼

[Result Mapping](#)

In the Source File Type list, select the type of file you want to convert into different format. Available options are- Excel, CSV, TSV and Custom. You can use the Custom option to create a custom file format with specified separators.

Excel

It lets you convert a single sheet into a different file of specified file type. The sheet name must be a single sheet name and not multiple sheets.

File Operations (i) (m)

File Operation Type

Convert ▼

Source File Type

Excel ▼

Destination File Type ▼

Source File Path ▼

Sheet Name ▼

Destination Folder Path ▼

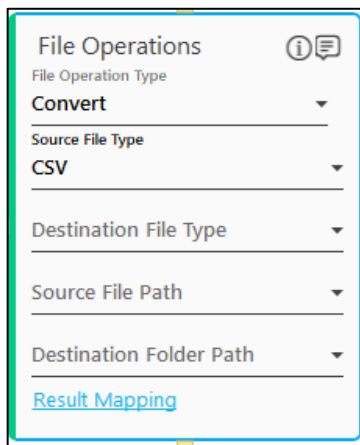
[Result Mapping](#)

1. In the **Destination File Type** list, select the desired format to which you want the source file to be converted. Available options are - **CSV, TSV** and **Custom**.
2. In the **Source File Path** list, select the parameter holding the source path of the file along with the file name. You must define the parameter in the Parameter bar to store the value and make it available for the selection.
3. In the **Sheet Name** list, select the parameter holding the name of the sheet that you want to convert into selected file type. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection. Leave this field blank, if you want to convert the entire excel file into a different format.

4. In the **Destination Separator** list, select the parameter holding the field separator to tell the operation where each field begins and ends in the converted file type. This field appears only when **Destination File Type** is **Custom**.
5. In the **Destination Folder Path** list, select the parameter holding the destination folder name where the converted file must be saved. For example, D:\UpdatedData. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection.
6. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

CSV/TSV

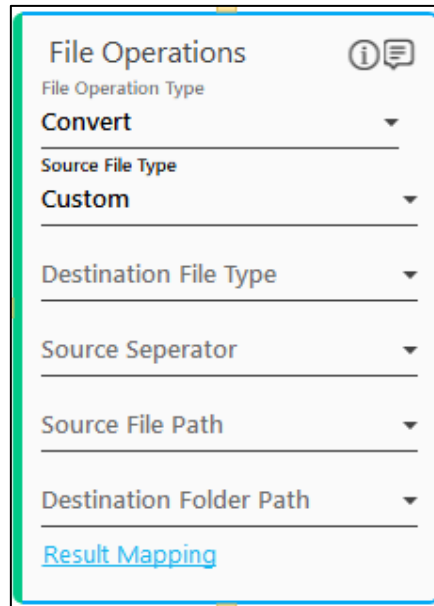
It lets you convert a file to Comma Separated (CSV) or Tab Separated Values (TSV) file format.



1. In the **Destination File Type** list, select the file type that you want to convert the selected file into. Available options are - **Excel, TSV** and **Custom**.
2. In the **Source File Path** list, select the parameter holding the source path of the file along with the file name. You must define the parameter in the Parameter bar to store the value and make it available for the selection.
3. In the **Destination Separator** list, select the parameter holding the desired field separator to tell the operation where each field begins and ends in the converted file type. This field appears only when **Destination File Type** is **Custom**.
4. In the **Destination Folder Path** list, select the parameter holding the destination folder name where the converted file must be saved. For example, D:\UpdatedData. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection.
5. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

Custom

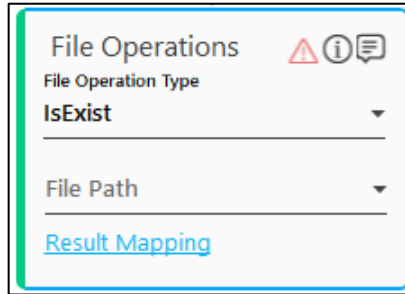
It lets you define your own file format.



1. In the **Destination File Type** list, select the file type that you want to convert the selected file into. Available options are - **CSV, Excel, TSV** and **Custom**.
2. In the **Source Separator** list, select the parameter holding the desired separator of the values such as \$, & and so on.
3. In the **Source File Path** list, select the parameter holding the source path of the file along with the file name and file extension. You must define the parameter in the Parameter bar to store the value and make it available for the selection.
4. In the **Destination Separator** list, select the parameter holding the field separator to indicate the operation where each field begins and ends in the converted file type. This field appears only when **Destination File Type** is **Custom**.
5. In the **Destination Folder Path** list, select the parameter holding the destination folder where the converted file must be saved. For example, D:\UpdatedData. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection.
6. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

IsExist

It helps you to automate checking whether a file exists at a particular location.



1. In the **File Path** list, select the parameter holding the source path of the file along with the file name and file extension. You must define the parameter in the Parameter bar to store the value and make it available for the selection.
2. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

File Operations Properties

The properties of File Operations activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.

Property Name	Usage
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as File Operations . You can change the name as required.

[Step-By-Step Guide to use File Operations to Create a File, Move it to Another Folder and Zip the Files](#)

Let's see an example of creating a .txt file, move all the .txt files available in the folder to a different folder and then zip the folder.

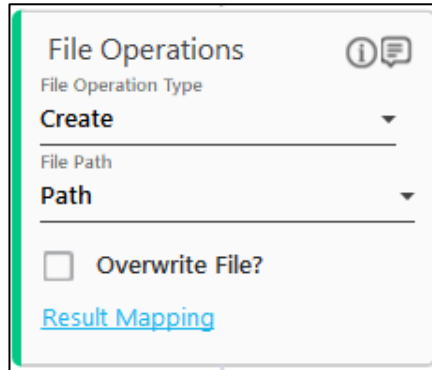
To create the .txt file, move and zip the folder:

1. In the Canvas Tools pane, click **File** to expand the tool and view the associated activities.
2. Drag the **File Operations** activity and drop on to the Flowchart designer on the Canvas
3. In the **File Operation Type** list, select **Create** operation.
4. In the **Parameter** bar, create an **In** argument, **Path** to pass the folder path as input for identifying the required file.

Name	Direction	Argument type	Default value
Path	In	String	"D:\SampleData\NewFile.txt"

Variables Arguments 100%

5. In the **File Path** list, select the **Path** argument created above.



6. Add another **File Operations** activity. Create the **In** arguments of type **String** and define the input details:

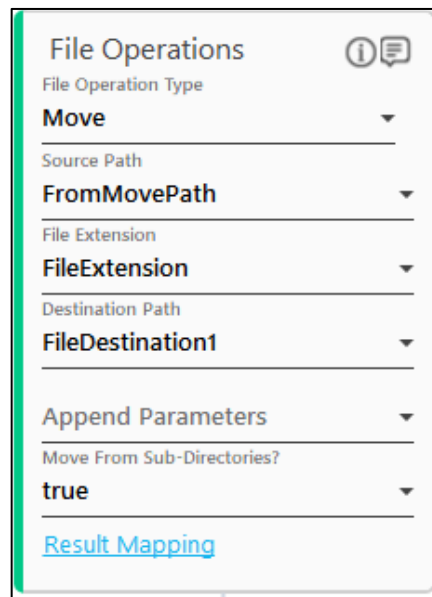
- **FromMovePath** - to pass the path of the folder from where all the .txt files must be moved.
- **FileExtention** - to pass the file format of the file that must be moved to other folder.
- **FileDestnation1** - to pass the path of the folder where all the .txt files must be moved.

↓ Name	Direction	Argument type	Default value
FileDestination1	In	String	"D:\Demo"
FileExtension	In	String	".*.txt"
FromMovePath	In	String	"D:\Sample Data"

7. In the **Source Path** list, select the **Path** argument created above.

8. In the **File Extension** list, select the **FileExtension** argument created above.

9. In the **Destination Path** list, select the **FileDestination1** argument created above.



10. Add one more **File Operations** activity. Create the **In** arguments of type **String** and define the input details:

- **FileDestination2** - to pass the path of the destination folder where zipped folder must be created.
- **ZipFileName** - to pass the name with which the zipped folder must be created.

Name	Direction	Argument type	Default value
Path	In	String	"D:\SampleData\NewFile.txt"
FileExtension	In	String	".txt"
FileDestination1	In	String	"D:\Demo"
FileDestination2	In	String	"D:\Demo"
ZipFileName	In	String	"ReadmeFiles"

Create Argument

Variables Arguments

100%

11. In the **Source Path** list, select the **FileDestination1** argument created above. This is path where the created file is moved.

12. In the **Destination Path** list, select the **FileDestination2** argument created above.

13. In the **File Name** list, select the **ZipFileName** argument created above.

14. In the **Zip Format** list, select **.zip** as the format of the zip file/folder.

File Operations

File Operation Type
Zip

Source Path
FileDestination1

Destination Path
FileDestination2

File Name Zip Format
ZipFileName **.zip**

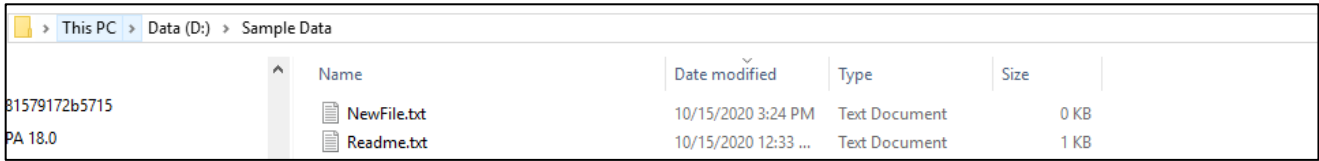
Password

[Result Mapping](#)

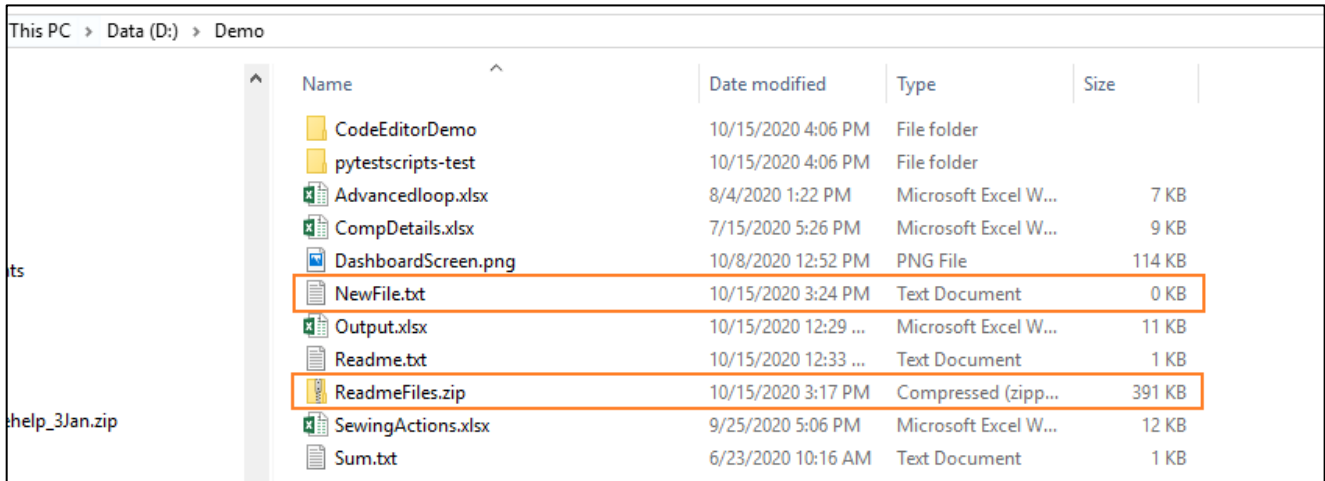
15. Save the process.

16. Setup the environment and then perform test run. You can publish, deploy and assign this process to a robot if you want to execute the process outside Automation Studio.

Observe that a .txt file with the specified name NewFile.txt gets created within the Sample folder.



The NewFile.txt along with other .txt files are then moved to the Demo folder and all the files get zipped and saved inside the same folder.

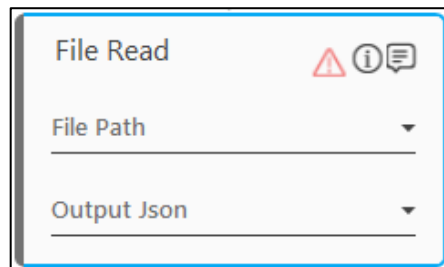


9.10.9 File Read

This activity allows you to automate the task of reading the text documents like DOCX and Rich Text format files. The entire document is read and the texts get stored as JSON strings. The JSON string can be further used as per the business requirement.

Using File Read Activity

1. In the Canvas Tools pane, click **File** to expand the tool and view the associated activities.
2. Drag the **File Read** activity and drop on to the Flowchart designer on the Canvas. The validation error symbol disappears when required inputs are provided.



3. In the **File Path** list, select the parameter holding the document file path along with the file name and file extension, for example, D:/documents/sample_doc.docx. You must define the parameter in the Parameter bar to store the value and

make it available for the selection.

- In the **Output Json** list, select a parameter to store the output of the processed document in the JSON string format. You must define the parameter in the **Parameter** bar to make it available for the selection.

The File Read activity with a default name is created.

File Read Properties

The properties of File Read activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as File Read . You can change the name as required.

Step-By-Step Guide to Use File Read to Read a .txt File and Store the Content as a JSON Fragment

Let's see an example of reading a .txt file and store the content as a JSON fragment.

To read the required file:

1. Create a new process.
2. In the Canvas Toolspane, click **File** to expand the tool and view the associated activities.
3. Drag the **File Read** activity and drop on to the Flowchart designer on the Canvas The validation error symbol disappears when required inputs are provided.
4. In the **Parameter** bar, create **In** arguments, **Path** and **Output** of type **String**. These arguments are used to pass the folder path as input for identifying the required file and store the output JSON fragment respectively.

Name	Direction	Argument type	Default value
Path	In	String	"D:\Sample Data\Readme.txt"
Output	In	String	Enter a VB expression

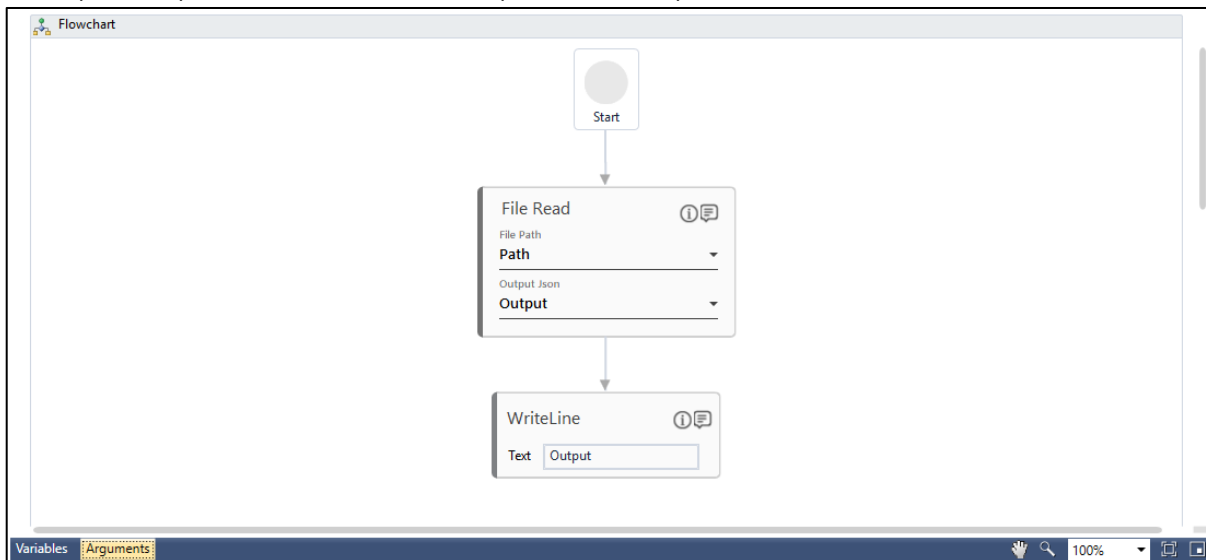
Create Argument

Variables Arguments

5. In the **File Path** list, select the Path argument created above.
6. In the **Output Json** list, select the Output argument created above.

To view the output in Automation Studio, let's add WriteLine activity. You can publish, deploy and assign this process to a robot if you want to execute the process outside Automation Studio.

7. Add a **WriteLine** activity and in the **Text** field, select the **Output** parameter created above to store the result of search operation performed. Below is the sample automation process workflow created:



8. Save the process.
9. Setup the environment and then perform test run. Below is the sample JSON fragment displayed in the **Output** console:

```
Output
[{"plainText":"Overview:\n\nThis BOT is used to get the free space of Drive(mentioned) in Local machine.\n\nInput parameters:\n\n1. DiskSpaceGB\n\nOutput:\n\nIn case of successful execution:\n\n1. IsSuccess: true\n\n2. Output: Display"}]
```

Studio Console Output Watch Errors Warnings Execution Stats Validation Error

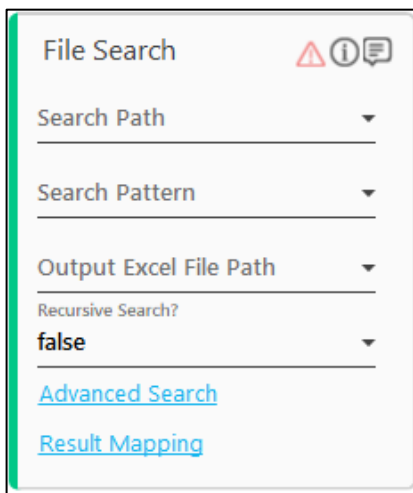
You can use the JSON activity to extract a relevant part of the response as per your requirement.

9.10.10 File Search

This activity allows you to automate searching a file or list of files available in a folder or network drive on the system. Using the file name, this activity recursively searches for the specified file(s) and stores the search result in an Excel file. You can perform the search based on the filters applied to the file parameters such as size, type, creation and modification date.

Using File Search Activity

1. In the **Canvas Toolspane**, click **File** to expand the tool and view the associated activities.
2. Drag the **File Search** activity and drop on to the **Flowchart designer on the Canvas**. The validation error symbol disappears when required inputs are provided.



3. In the **Search Path** list, select the parameter holding the source path where file search needs to be performed. For example, **D:\Data**. You must define the parameter in the Parameter bar to store the value and make it available for the selection.
4. In the **Search Pattern** list, select the parameter holding the file format condition to search the file. You must define the parameter in the **Parameter** bar to store the term and make it available for the selection. The supported formats for the search pattern are:
 - To search a particular file, provide the pattern as; **<file name.file extension>**. For example, **output.xlsx**.
 - To capture all files from a particular folder; provide the pattern as, **<*. *>**.
 - To search and capture all the files with a particular file extension; provide the pattern as, **<*.file extension>**. For

example, *.pdf.

- There could be more advanced search patterns depending upon your requirement such as, search all pdf files with a file name starting with 'A'. In such case, provide the pattern as <A*.pdf>.
5. In the **Output Excel File Path** list, select the parameter holding the path along with the excel file name and file extension to store the list of searched files . You must define the parameter in the **Parameter** bar to store the value and make it available for the selection.
 6. In the **Recursive Search** list, select **true** to search for the required file in the sub directories or sub folders of the source path. By default, the value is set to **false**.
 7. Click **Advanced Search** to expand and set the advance search options.

The screenshot shows a configuration window for 'File Search'. At the top right, there are three icons: a warning triangle, an information icon, and a help icon. The main area contains the following settings:

- Search Path**: A dropdown menu.
- Search Pattern**: A dropdown menu.
- Output Excel File Path**: A dropdown menu.
- Recursive Search?**: A dropdown menu currently set to **false**.
- Advanced Search**: A blue link that expands the search options.
- File Size limit in KB**: A dropdown menu.
- Creation Star** and **Creation End**: Two dropdown menus.
- Modified Fro** and **Modified To I**: Two dropdown menus.
- Result Mapping**: A blue link.
- Output Mapping**: A dropdown menu.

8. In the **File Size limit in KB** list, select the parameter holding the upper limit of the file size in KB. The files that are less in size than the specified file size are considered for the search. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection.

9. In the **Creation Start Date** list and **Creation End Date** list, select the parameters holding the date of file creation for searching the files created in the specified date range. For example, if the date specified for the **Creation Start Date** and **Creation End Date** is **06/21/2017** and **08/21/2017** respectively, then the process searches for the files created between 21st June 2017 and 21st August 2017. You must define the parameter in the **Parameter** bar to store the value and make it available for the selection.
10. In the **Modified From Date** list and **Modified To Date** list, select the parameters holding the date of file modification for searching the files modified in the specified date range. For example, if the date specified for the **Modified From Date** and **Modified To Date** is **11/15/2017** and **12/15/2017** respectively, then the process searches for the files modified between 11th December 2017 and 12th December 2017. You must define the parameters in the **Parameter** bar to store the values of the respective fields and make it available for the selection.
11. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the file operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

The File Search activity with a default name is created.

File Search Properties

The properties of File Search activity are listed in the following table and can be edited in the **Properties** grid on the right pane.

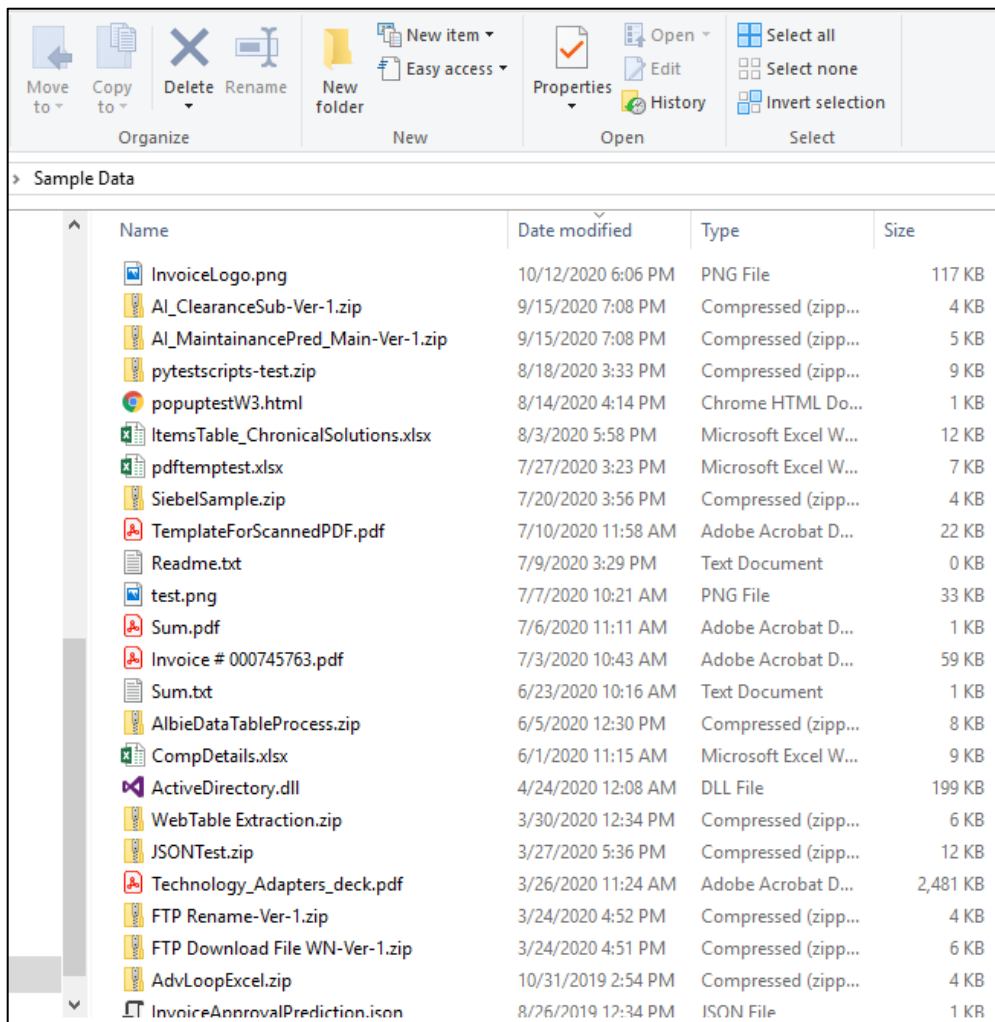
Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.

Property Name	Usage
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the flowchart designer area. By default, the name is set as File Search . You can change the name as required.

Step-By-Step Guide to Use File Search to Search for Files Modified During Last One Year With File Size Limit as 10KB

Let's see an example of searching Excel files modified during last one year whose file size limit is 10KB.

Below is the sample folder where we are looking for the required files that contains multiple files of different file format.



To search for the Excel files:

1. Create a new process.
2. In the **Canvas Toolspane**, click **File** to expand the tool and view the associated activities.
3. Drag the **File Search** activity and drop on to the **Flowchart designer on the Canvas** The validation error symbol disappears when required inputs are provided.
4. In the **Parameter** bar, create the following **In** arguments and define the required inputs:
 - **Path** - to pass the folder path as input for performing the search within the folder.
 - **Pattern** - to pass the file format as input for selecting the required files.
 - **OutputFile** - to define the folder path along with the excel file name and file extension to store the list of searched files.
 - **SizeLimit** - to pass the upper limit of the files as input for the files to be searched.
 - **ModifiedFrom** - to pass the from date of file modification as input.
 - **ModifiedTo** - to pass the to date of file modification as input.
 - **OperationResult** - to store the result of the file search operation performed.

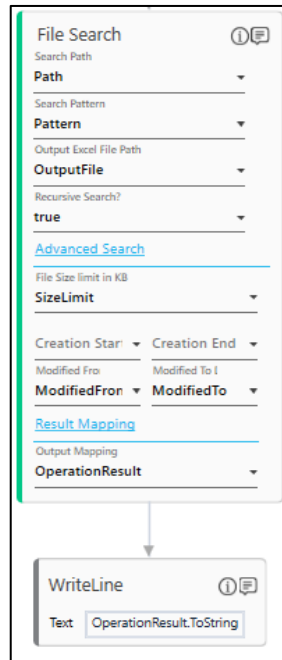
Name	Direction	Argument type	Default value
Path	In	String	"D:\Sample Data"
Pattern	In	String	"*.xlsx"
OutputFile	In	String	"D:\Demo\Output.xlsx"
SizeLimit	In	Int32	10
ModifiedFrom	In	String	"10/15/2019"
ModifiedTo	In	String	"10/15/2020"
OperationResult	In	Boolean	Enter a VB expression

5. In the **Search Path** list, select the **Path** argument created above.
6. In the **Search Pattern** list, select the **Pattern** argument created above.
7. In the **Output Excel File Path** list, select the **OutputFile** argument created above.
8. In the **Recursive Search** list, select true to search for the required file in the sub directories or sub folders of the source path.
9. Click **Advanced Search** to expand and set the advance search options.
10. In the **File Size limit in KB** list, select **SizeLimit** argument created above.
11. In the **Modified From Date** list and **Modified To Date** list, select the **ModifiedFrom** and **ModifiedTo** arguments respectively for searching the files modified in the specified date range.
12. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the **OperationResult** parameter.

To view the output in Automation Studio, let's add WriteLine activity. You can publish, deploy and assign this process to a robot if

you want to execute the process outside Automation Studio.

13. Add a **WriteLine** activity and in the **Text** field, enter the **OperationResult** argument created above to store the result of search operation performed. Below is the sample automation process workflow created:



14. Save the process.
15. Setup the environment and then perform test run. Below is the sample screen shot of the excel file that stores the list of searched files:

The screenshot shows an Excel spreadsheet titled 'Output.xlsx - Excel'. The spreadsheet contains a table with the following data:

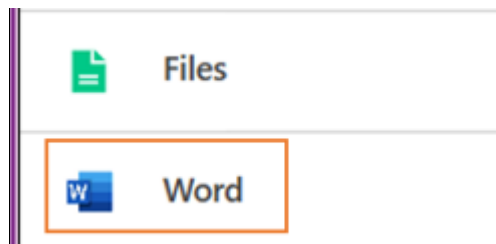
File Name	File Path	Owned By	Creation Date	Last Modified At	File Size	File Type
CompDetails.xlsx	D:\Sample Data\CompDetails.xlsx	ITLINFOSYS\robert.walton	7/6/2020 11:10	6/1/2020 11:15	9	.xlsx
pdftemptest.xlsx	D:\Sample Data\pdftemptest.xlsx	ITLINFOSYS\robert.walson	7/27/2020 15:23	7/27/2020 15:23	7	.xlsx

The Output console of Automation Studio displays the file search operation result:



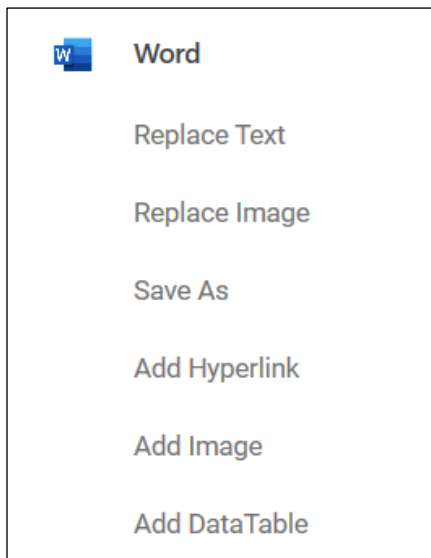
9.11 Word

The Word tool enables you to replace text or image, save the word file in various formats, add hyperlink, image, or data table at specified positions in the document. Both .doc and .docx file types are supported.



Following activities are included in Word tool:

- [Replace Text](#)
- [Replace Image](#)
- [Save As](#)
- [Add Hyperlink](#)
- [Add Image](#)
- [Add Data Table.](#)

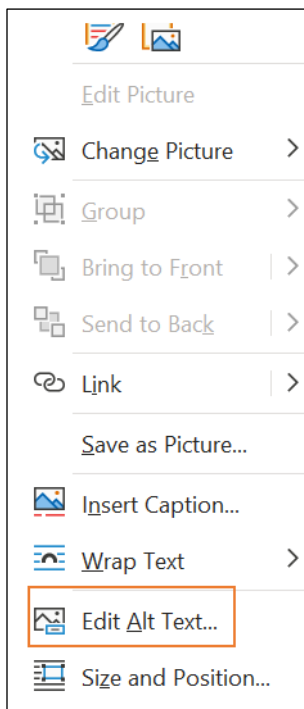


9.11.1 Prerequisites

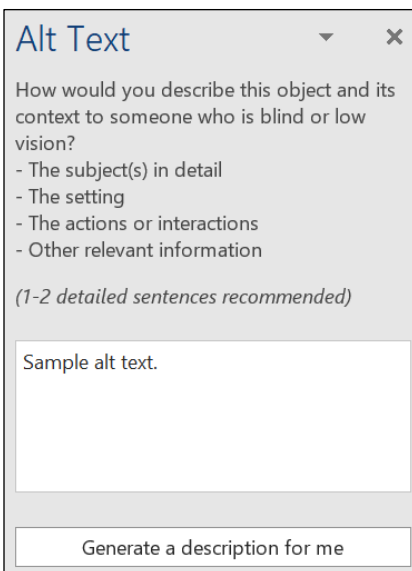
- Word file, that is required to be edited, must not be open at the time of process execution
- Word file, that is required to be edited, must not be in a read-only mode
- For **Replace Image** action, make sure the image has an **Alt Text**.

Adding Alt Text to an image:

1. Right click the image you want to add the alt text to and select **Edit Alt Text**.



2. Add the alternate text in the **Alt Text** window in the right panel and close the **Alt Text** window.



Alt Text

How would you describe this object and its context to someone who is blind or low vision?

- The subject(s) in detail
- The setting
- The actions or interactions
- Other relevant information

(1-2 detailed sentences recommended)

Sample alt text.

Generate a description for me

9.11.2 Replace Text

Replace Text activity is used to replace specific text in the document with the required text.

Using Replace Text Activity

1. Make sure the [prerequisites](#) for using this activity are met.
2. Create a new process.
3. In the **Canvas Tools** pane, click **Word** to expand the tool and view the associated activities.
4. Drag the **Replace Text** activity and drop on to the **Flowchart** designer on the **Canvas**.
5. Define all the input parameters as In arguments and output parameters as Out arguments in the Argument bar to make it available for the selection.
6. From the **File Path** dropdown list, select the In argument holding the required file path value, along with the file name and extension, of the required word document. The selected argument must be of String type.

- From the **Search For** dropdown list, select the **In** argument holding the text that must be searched and replaced in the word document. The selected argument must be of **String** type.

Note: The input value for this dropdown list is case-sensitive, and the parameter needs to be defined accordingly.

- From the **Replace With** dropdown list, select the **In** argument holding the text required to replace the searched text. The selected argument must be of **String** type.
- From the **ReplaceOption** dropdown list, select the required option.
 - Replace**: To replace the first occurrence of the specified text
 - ReplaceAll**: To replace all occurrences of the specified text

- Click the **Output Result** link and from the **Output Mapping** dropdown list, select the required **Boolean** type of Output parameter to map the status of the operation performed.

The **Replace Text** activity is configured.

9.11.3 Replace Image

Replace Image activity is used to replace specific image or images in the document with the required image.

Note: Microsoft validation applies on the image size.

Using Replace Image Activity

1. Make sure the [prerequisites](#) for using this activity are met.
2. Create a new process.
3. In the **Canvas Tools** pane, click **Word** to expand the tool and view the associated activities.
4. Drag the **Replace Image** activity and drop on to the **Flowchart** designer on the **Canvas**.
5. Define all the input parameters as **In** arguments and output parameters as **Out** arguments in the Argument bar to make it available for the selection.
6. From the **File Path** dropdown list, select the **In** argument holding the required file path value, along with the file name and extension, of the required word document. The selected argument must be of **String** type.

7. From the **Alternate Text** dropdown list, select the **In** argument holding the **Alt Text** of the image, required to be replaced. The argument must be of **String** type.

Note: Make sure alternate text is available for the image. For more information refer [Alt Text](#) section.

8. From the **Image Path** dropdown list, select the **In** argument holding the path of the image to be used as a replacement. The argument must be of **String** type.
Click the **Output Result** link and from the **Output Mapping** dropdown list, select the required **Boolean** type of Output parameter to map the status of the operation performed.

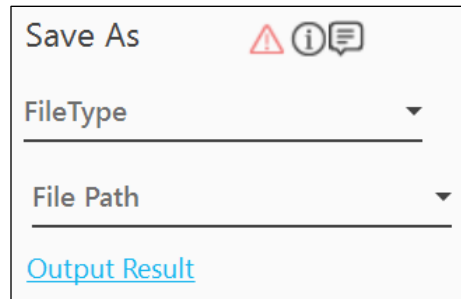
The **Replace Image** activity is configured.

9.11.4 Save As

Save As activity is used to make a copy of the document in required format.

Using Save As Activity

1. Make sure the [prerequisites](#) for using this activity are met.
2. Create a new process.
3. In the **Canvas Tools** pane, click **Word** to expand the tool and view the associated activities.
4. Drag the **Save As** activity and drop on to the **Flowchart** designer on the **Canvas**.
5. From the **File Type** dropdown list, select the required file type.



Available options for file types are:

- XMLDocument
 - MacroEnabledDocument
 - OldDocument
 - WordTemplate
 - WebPage
 - WebPageFiltered
 - RichText
 - PlainText
 - PDF
6. Define all the input parameters as **In** arguments and output parameters as **Out** arguments in the Argument bar to make it available for the selection.
 7. From the **File Path** dropdown list, select the **In** argument holding the required file path value, along with the file name and extension, of the required word document. The selected argument must be of **String** type.
 8. Click the **Output Result** link and from the **Output Mapping** dropdown list, select the required **Boolean** type of Output parameter to map the status of the operation performed.

Note: The new file gets saved at the same location as the old file.

The **Save As** activity is configured.

9.11.5 Add Hyperlink

Add Hyperlink activity is used to add a hyperlink in a document or to specified text in the document.

Using Add Hyperlink Activity

1. Make sure the [prerequisites](#) for using this activity are met.
2. Create a new process.
3. In the **Canvas Tools** pane, click Word to expand the tool and view the associated activities.
4. Drag the Add Hyperlink activity and drop on to the **Flowchart** designer on the **Canvas**.
5. Define all the input parameters as In arguments and output parameters as Out arguments in the Argument bar to make it available for the selection.
6. From the **File Path** dropdown list, select the In argument holding the required file path value, along with the file name and extension, of the required word document. The selected argument must be of **String** type.

The screenshot shows the configuration interface for the 'Add Hyperlink' activity. It features a title bar with the activity name and three icons (warning, information, help). Below the title bar are four dropdown menus, each with a downward arrow: 'File Path', 'Address', 'TextToDisplay', and 'RelativeInsert'. At the bottom of the configuration area is a 'Result Mapping' section, which is currently empty and highlighted with a blue border.

7. From the Address dropdown list, select the In argument holding the address of the hyperlink required to be added. The selected argument must be of String type.
8. From the **TextToDisplay** dropdown list, select the In argument holding the text that is required to be displayed, as a tooltip, for the hyperlink. The selected argument must be of String type.
9. From the **RelativeInsert** dropdown list, select the required value, to insert hyperlink, with respect to the document or specific text.

The screenshot shows the 'Add Hyperlink' dialog box with the following fields and options:

- Add Hyperlink** (with warning, info, and help icons)
- File Path** (dropdown menu)
- Address** (dropdown menu)
- TextToDisplay** (dropdown menu)
- RelativeInsert** (dropdown menu, highlighted with an orange box)
 - Document
 - Text

- a. To insert hyperlink with respect to the document: From the **DocPosition** dropdown list, select the position, at which, you want to insert the hyperlink. You can choose to insert the hyperlink at the **Start** or **End** of the document.

The screenshot shows the 'Add Hyperlink' dialog box with the following fields and options:

- Add Hyperlink** (with warning, info, and help icons)
- File Path** (dropdown menu)
- Address** (dropdown menu)
- TextToDisplay** (dropdown menu)
- RelativeInsert** (dropdown menu)
- Document** (dropdown menu, highlighted with an orange box)
 - DocPosition
 - Start
 - End

- b. To insert hyperlink with respect to the text: From the **Search Text** dropdown list, select the **In** argument holding the text to be searched. The argument must be of **String** type.

Note: The input value is case-sensitive, and the parameter needs to be defined accordingly.

The screenshot shows a configuration panel for adding a hyperlink. It contains several dropdown menus: 'File Path', 'Address', 'TextToDisplay', 'Text' (under the 'RelativeInsert' section), 'Search Text', 'Text Position', and 'Occurrence'. The 'Search Text' dropdown is highlighted with an orange border.

- c. From the **Text Position** dropdown list, select the position of the hyperlink. You can choose to insert the hyperlink before the text, after the text or you can replace the text with the hyperlink.

The screenshot shows the same configuration panel as above, but with the 'Text Position' dropdown menu open. The options listed are 'Text Position', 'InsertAfter', 'InsertBefore', and 'Replace'. The 'Text Position' option is highlighted with an orange border.

- d. From the **Occurrence** dropdown list, select the occurrence of the text you want to add the hyperlink to. You can choose to add it to the first, last or all occurrences of the text. You can also define an index of the occurrences by selecting **OccurrenceIndex**.

- e. From the **OccurrenceIndex** dropdown list, select the **In** argument, holding the required index of the text.

Note: RelativeInsert sometimes requires manual intervention or adjustments to position the hyperlink, depending on the size of the document and the word placements.

10. Click the **Result Mapping** link and from the **Output Mapping** dropdown list, select the **Out** argument which is made available to map the status of the operation performed. The selected argument must be of **Boolean** type.

The **Add Hyperlink** activity is configured.

9.11.6 Add Image

Add Image activity is used to add image in the document or at specified positions in the document.

Using Add Image Activity

1. Make sure the [prerequisites](#) for using this activity are met.
2. Create a new process.
3. In the **Canvas Tools** pane, click Word to expand the tool and view the associated activities.
4. Drag the Add Image activity and drop on to the **Flowchart** designer on the **Canvas**.

5. Define all the input parameters as In arguments and output parameters as Out arguments in the Argument bar to make it available for the selection.
6. From the **File Path** dropdown list, select the **In** argument holding the required file path value, along with the file name and extension, of the required word document. The selected argument must be of **String** type.

7. From the Image Path dropdown list, select the In argument holding the file path, of the image, that is to be inserted. The selected argument must be of String type.
8. From the **RelativeInsert** dropdown list, select the required value to insert image with respect to the document or specific text.

- a. To insert image with respect to the document: From the **Position** dropdown list, select the position you want the image inserted at. You can choose to insert the image at the **Start** or **End** of the document.

- b. To insert image with respect to the text: From the **Search Text** dropdown list, select the **In** argument holding the text to be searched. The argument must be of **String** type.

Note: The input value is case-sensitive, and the parameter needs to be defined accordingly.

The screenshot shows the 'Add Image' configuration panel with the following fields: File Path, Image Path, RelativeInsert, Text, Search Text (highlighted), Text Position, Occurrence, and a Result Mapping link.

- c. From the **Text Position** dropdown list, select the position of the image. You can insert the image after the text, before the text or you can replace the text with the image.

The screenshot shows the 'Add Image' configuration panel with the 'Text Position' dropdown menu open, displaying options: Text Position, InsertAfter, InsertBefore, and Replace.

- d. From the **Occurrence** dropdown list, select the occurrence of the text you want to add the Image to. You can choose to add it to the first, last or all occurrences of the text. You can also define an index of the occurrences by selecting **OccurrenceIndex**.

- e. From the **OccurrenceIndex** dropdown list, select the **In** argument, holding the required index of the text.

Note: RelativeInsert sometimes requires manual intervention or adjustments to position the image, depending on the size of the image.

9. Click the **Result Mapping** link and from the **Output Mapping** dropdown list, select the **Out** argument which is made available to map the status of the operation performed. The selected argument must be of **Boolean** type.

The **Add Image** activity is configured.

9.11.7 Add Data Table

Add Data Table activity is used to add a data table in the document or to specified positions in the document.

Using Add Data Table Activity

1. Make sure the [prerequisites](#) for using this activity are met.
2. Create a new process.
3. In the **Canvas Tools** pane, click **Word** to expand the tool and view the associated activities.
4. Drag the **Add Data Table** activity and drop on to the **Flowchart** designer on the **Canvas**.
5. From the **File Path** dropdown list, select the **In** argument holding the required file path value, along with the file name and extension, of the required data table. The selected argument must be of **String** type.

6. From the **DataTable Name** dropdown list, select the **In** argument holding the data table name. The selected argument must be of **Datatable** type.

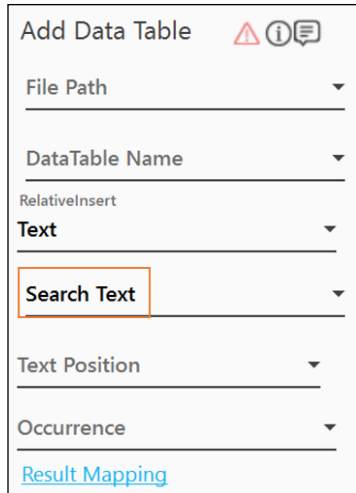
Note: The argument can be created only for an existing data table. For more information on using a data table, refer [DataTable](#) section in the AE-RPA-Engage Automation Studio Guide.pdf.

7. From the **RelativeInsert** dropdown list, select the required value to insert data table with respect to the document or text.

- a. To insert data table with respect to the document: From the **DocPosition** dropdown list, select the position, at which, you want to insert the data table. You can choose to insert the data table at the start or end of the document.

- b. To insert data table with respect to the text: From the **Search Text** dropdown list, select the **In** argument holding the text to be searched. The argument must be of **String** type.

Note: The input value is case-sensitive, and the parameter needs to be defined accordingly.



Add Data Table ⚠️ ⓘ 🗨️

File Path ▼

DataTable Name ▼

RelativeInsert

Text ▼

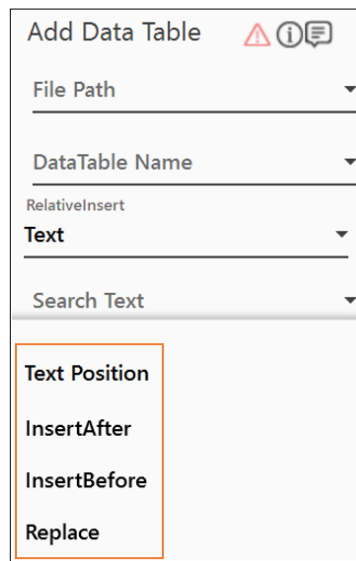
Search Text ▼

Text Position ▼

Occurrence ▼

[Result Mapping](#)

- c. From the Text Position dropdown list, select the position of the data table. You can insert the data table after the **text**, before the text or you can replace the text with the data table.



Add Data Table ⚠️ ⓘ 🗨️

File Path ▼

DataTable Name ▼

RelativeInsert

Text ▼

Search Text ▼

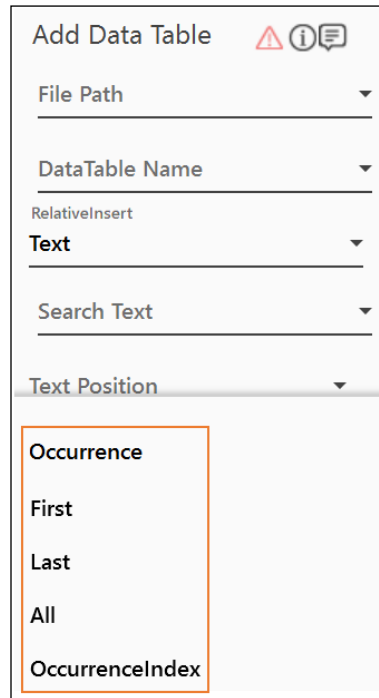
Text Position




InsertAfter

InsertBefore

Replace

- d. From the **Occurrence** dropdown list, select the occurrence of the text you want to add the data table to. You can choose to add it to the first, last or all occurrences of the text. You can also define an index of the occurrences by selecting **OccurrenceIndex**.



Add Data Table   

File Path

DataTable Name

RelativeInsert

Text

Search Text

Text Position

- Occurrence
- First
- Last
- All
- OccurrenceIndex

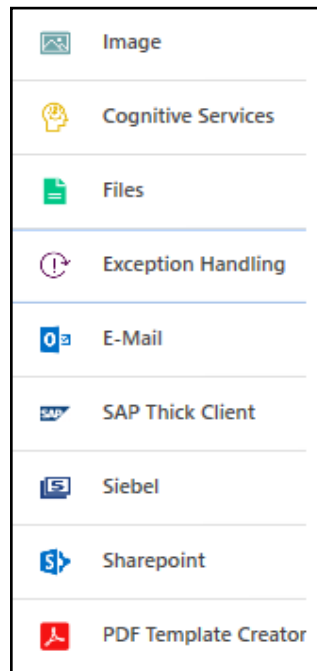
- e. From the **OccurrenceIndex** dropdown list, select the In argument, holding the required index of the text.
8. Click the **Result Mapping** link and from the **Output Mapping** dropdown list, select the **Out** argument which is made available to map the status of the operation performed. The selected argument must be of **Boolean** type.

The Add Data Table activity is configured.

9.13 Exception Handling

While executing an automation request, potential errors may happen, and various actions might be performed to identify/troubleshoot them. E.g., running a sequence of activities, or ending the process gracefully, or even ignoring the error, as and when they occur.

To support these actions in CE a component of Exception Handling is added in Automation Studio's Canvas Tool. This component consists of Throw and TryCatch block.



9.13.1 TryCatch

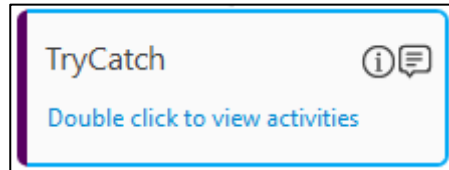
It helps you to configure steps which potentially raises an exception and steps to handle the exception.

The Try Catch activity contains a Try activity, a Catch activity and a Finally activity. The steps or actions that can cause exception are configured in the Try activity block. If the Try activity throws an exception, the exception is matched with the exception defined in the Catch activity block. If the exception matches, the steps or actions configured in the Catch activity block are executed. The Finally activity is executed irrespective of occurrence of an exception. Steps or actions in the Finally activity block are configured to be executed when either the Try activity or any required action in the Catch activity completes. If the exception does not match, the Finally activity block is not executed and the exception is passed to the parent activity.

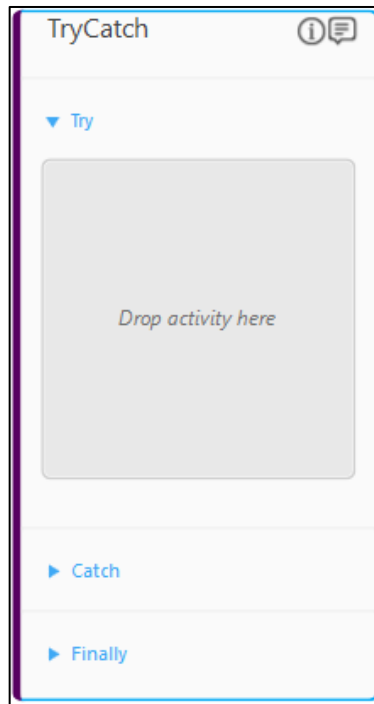
The parent activity (an activity block, for example, an Application activity) has an option to consume the error by setting its Ignore Error property to Yes. The automation process workflow is terminated if the parent activity at the root level of the workflow does not consume the error.

Using Try Catch Activity

1. In the **Canvas Tools** pane, click **Exception Handling** to expand the tool and view the associated activities.
2. Drag the **Try Catch** activity and drop on to the **Flowchart** designer on the **Canvas**.

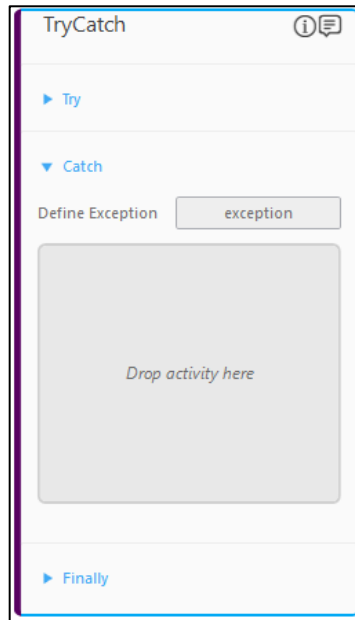


3. Double click the **TryCatch** activity block to configure the required steps or the actions. By default, the **Try** activity is expanded.

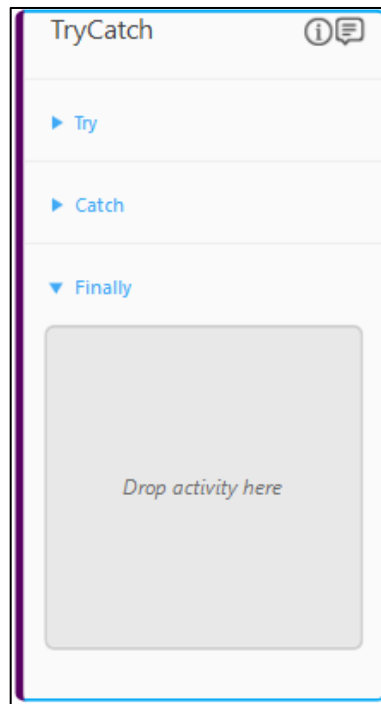


4. In the **Try** activity block, drag and drop the activity that can potentially throw an exception. Whenever the exception occurs, the workflow moves to the **Catch** activity block.

5. Click and expand the **Catch** activity block.



6. In the **Define Exception** field, enter the exception message that appears in the **Try** activity block.
7. In the **Catch** activity block, drag and drop the activity to handle the exception that appears in the **Try** activity block.
8. Click and expand the **Finally** activity block.



9. In the **Finally** activity block, drag and drop the activity that must be executed when either **Try** or **Catch** activity completes. The **Try Catch** activity is configured.

Try Catch Activity Properties

The properties of Try Catch activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
DisplayName	The display name of the activity in the Flowchart designer area. By default, the name is set as TryCatch. You can change the name as required.

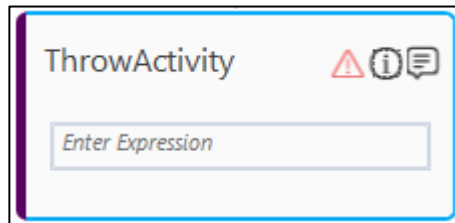
9.13.2 Throw

It helps you to end the execution of the automation process workflow and displays an exception message for the same. The exception message gets logged in the log file of Automation Studio.

You can use the Throw activity with or without the Try Catch activity.

Using Throw Activity

1. In the **Canvas Tools** pane, click **Exception Handling** to expand the tool and view the associated activities.
2. Drag the **Throw** activity and drop on to the **Flowchart** designer on the **Canvas**. The validation error symbol disappears when required inputs are provided.



3. In the **Enter Expression** field, enter the message that you want to display as the exception. The text must be entered in double quotes.
4. The **Throw** activity with a default name is created.

Throw Activity Properties

The properties of Throw activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the Flowchart designer area. By default, the name is set as ThrowActivity. You can change the name as required.
ExceptionText	The message to display as the exception. Alternatively, you can specify the message in the Enter Expression field of the activity block. The message specified in the Properties grid reflects in the activity block and vice versa.

9.13.3 Ignore Error Support

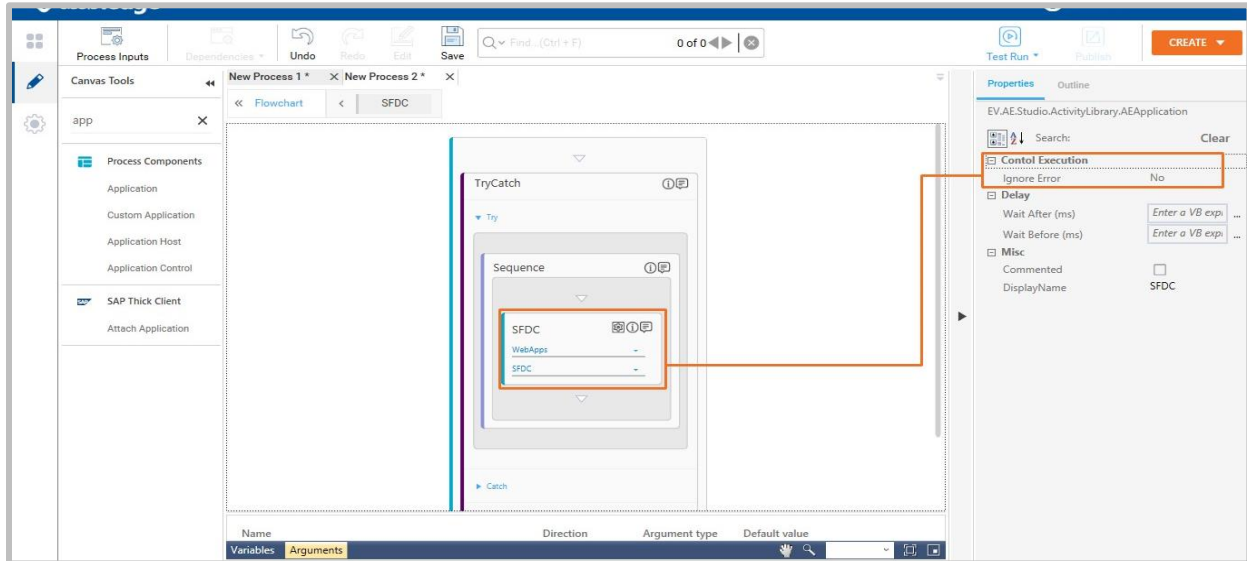
Provision is provided in the Studio to set error Ignore Error property in the activities.

The error can be ignored at a particular step/component. The utility is provided to ignore the error with the default value as *Ignore Error = "No"*.

The Ignore Error is set either at an activity level or field level.

Activities with "ignored error" do not alter the process state even if it fails.

If the activity fails, Step/Activity status is set to "ErrorIgnored", and if Ignore Error is set to "Yes", it is displayed in Kibana reporting.



Subsequent activities under the same parent block are executed if Ignore Error is set to "Yes", and activity fails.

If an error occurs in an activity where Ignore Error is set to "No", then an error is thrown for this activity, and subsequent activities under the same parent block are not executed.

- Ignore error property is available for all custom activities including container activities like Application, Excel loop, etc.
- Ignore error property is not available for built-in activities like WriteLine, Delay, etc.

Parent (a custom container activity, for example, an application block) of the activity has an option to consume the error by setting its Ignore Error property to "Yes".

The workflow process is terminated if the activity/parent block at workflow root level, does not consume the error.

9.14 E-Mail

E-Mail automation helps you perform e-mail operations related to SMTP and MS Outlook by automation the required action using out-of-box activities provided by Automation Studio.

Automating the e-mail operations lets you save the time, automatically trigger recipient specific e-mails, improve business and marketing strategies and so on.

Following are the available activities:

- [SMTP Mail](#)
- [Outlook Mail](#)

9.14.1 SMTP Mail

It allows you to automate certain e-mail operations such as drafting e-mails and send them, search for the required e-mails based on various search criteria, read e-mails and download attachments, send reply, forward or delete the e-mails, using the SMTP server.

Message ID or Trigger ID are used to perform certain SMTP operations like search and read an e-mail. Message ID is a unique identifier associated with an e-mail in the mailbox that helps to select the required e-mail to perform the intended operation.

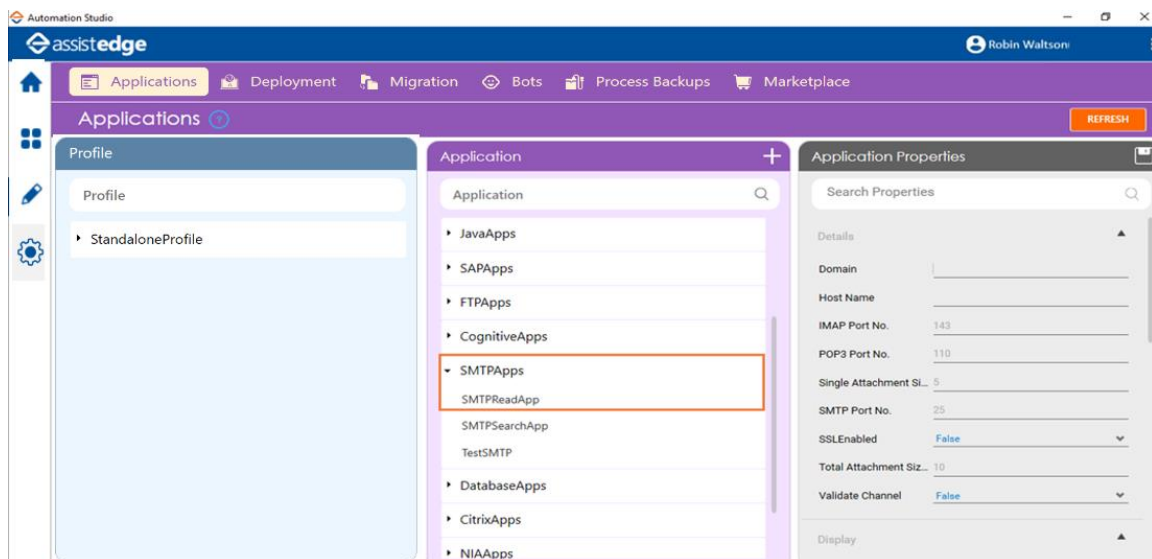
Note: This activity must be used inside an Application activity or the system displays an error. This lets you establish a connection between Automation Studio and the configured SMTP server.

Prerequisite


You must configure the SMTP sever in Automation Studio before you start configuring the steps of the automation process workflow. This establishes the connection between the SMTP server and Automation Studio to perform the automation.

Below are the minimum required properties for configuring the intended SMTP server. If you want to define remaining properties, refer SMTPApp in the Admin Menu.

- In the **Admin** menu, add an application of **Application Type- SMTPApps**. The mandatory fields are highlighted with red box.

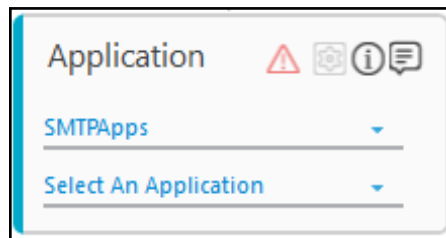


- In the **Domain** field, enter the domain name of the SMTP server.
- In the **Host Name** field, enter the host name or the IP address of the SMTP server.
- The **IMAP Port No.** and the **POP3 Port No.** fields are auto filled with the default values, **143** and **110** respectively. You can edit the port number if the server is configured using different port number. If IMAP port number is not known, use the default value for configuring the SMTP application.

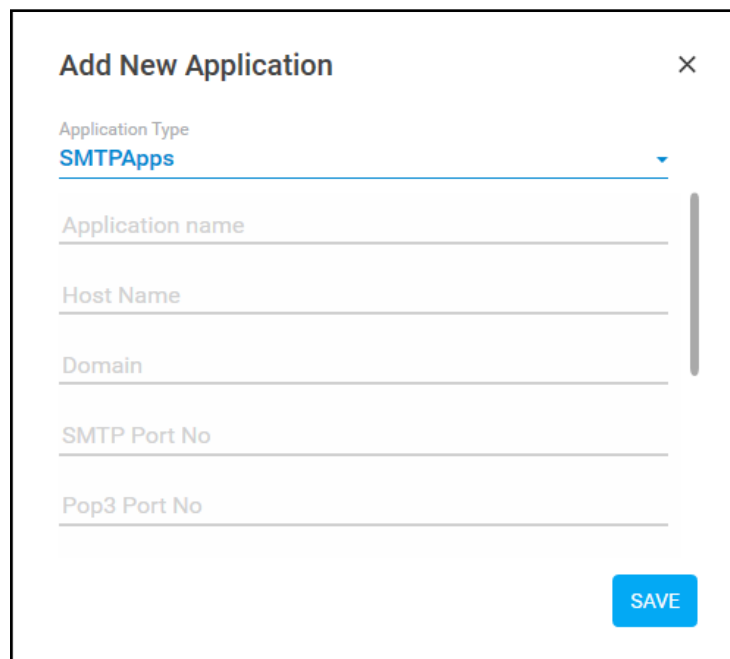
9. Click the  (**Save Properties**) icon to save the application.
10. The SMTP server is configured.

Using SMTP E-Mail Activity

1. In the **Canvas Tools** pane, click **Process Components** to expand the tool and view the associated activities.
2. Drag the **Application** activity and drop on to the **Flowchart** designer on the **Canvas**.
3. The validation error symbol disappears when required inputs are provided.

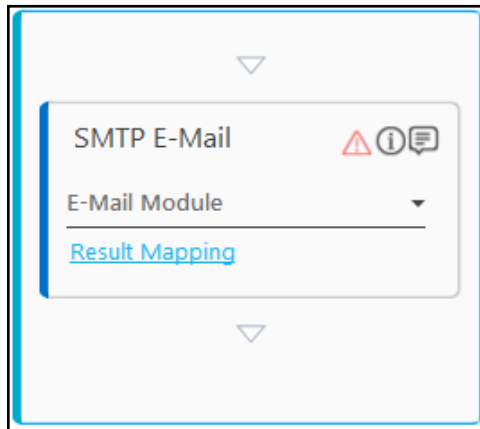


4. In the **Application Type** list, select **SMTPApps**. You must have at least one application added for it to appear in the list.
5. In the **Select An Application** list, select the configured SMTP server where you want to perform automation. Alternatively, you can add a new web application at this point in time. To add web application:
6. In the Select An Application list, click Add New Application. The Add New Application dialog box appears.



- a. In the **Application name** field, enter the desired name of the SMTP server you want to add. The name must not contain any special character or space.
- b. In the **Host Name** field, enter the host name or the IP address of the SMTP server.
- c. In the **Domain** field, enter the domain name of the SMTP server.

- d. In the **SMTP Port No** field, enter the port used to connect to the SMTP server. The default value of the SMTP port is 25.
 - e. In the **Pop3 Port No** field, enter the port number of the POP3 protocol. The default value of the POP3 port is 110.
 - f. In the **IMAP Port No** field, enter the port number of the IMAP protocol. It is an advanced protocol compared to POP3 protocol for mail server connection. The default value of IMAP port is 143.
 - g. In the **Single Attachment Size Limit (MB)** field, enter the maximum file size of the attachment allowed to be sent in an e-mail using SMTP, specified in MB. By default, the single attachment size is 5 MB.
 - h. In the **Total Attachment Size Limit (MB)** field, enter the total file size of the attachment allowed to be sent in an e-mail using SMTP, specified in MB. By default, the total attachment size is 10 MB.
 - i. In the **SSL Enables** list, select **True** if SSL is enabled for the SMTP server, else select **False**.
 - j. In the **ValidateChannel** list, select **True** if the user Id used to connect to the server is same as the from e-mail address in **Send E-Mail** activity, else select **False**.
 - k. Click **SAVE**. The SMTP server is configured
7. Double click the **Application** activity, drag the **SMTP E-Mail Activity** and drop inside the **Application** activity.



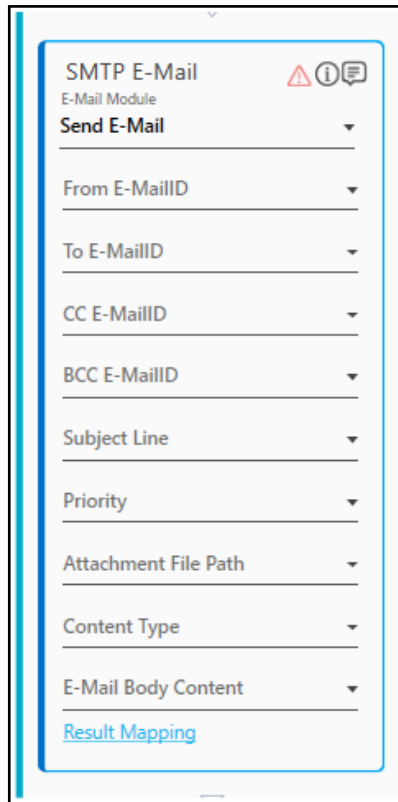
8. In the **E-Mail Module** list, select the operation you want to perform. Click any of the link below to know the steps to perform the selected operation:
- [Send E-Mail](#)
 - [Get EntryId](#)
 - [Read E-Mail](#)
 - [Move E-Mail](#)
 - [Search E-Mail](#)

Note: The Getting EntryId operation can be directly used without the Application activity.

Send E-Mail

It helps you to create a new e-mail, attach a file, mention the priority of the e-mail, and then send the e-mail to intended recipients using SMTP server.

You can add multiple senders and recipients separated by semi-colon (;). All the files to be attached, must be available and accessible.



- In the **From E-MailID** list, select the parameter holding the e-mail Ids to be used for sending the e-mail. The Ids must be semi-colon (;) separated values. You must define the parameter in the **Parameter** bar to make them available for the selection.
- In the **To E-MailID** list, select the parameter holding the e-mail Ids to which the e-mail must be sent. The Ids must be semi-colon (;) separated values. You must define the parameters in the **Parameter** bar to make them available for the selection.
- In the **CC E-MailID** list, select the parameter holding the e-mail Ids of the recipients whom you want to copy on the e-mail publicly. The Ids must be semi-colon (;) separated values. You must define the parameters in the **Parameter** bar to make them available for the selection.
- In the **BCC E-MailID** list, select the parameter holding the e-mail Ids of the recipients whom you want to copy on the e-mail privately. The Ids must be semi-colon (;) separated values. You must define the parameters in the **Parameter** bar to make them available for the selection.

Note: It is mandatory to select at least any one among To E-MailID, CC E-MailID and BCC E-MailID list.

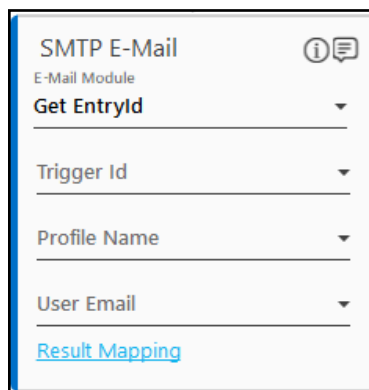
- e. In the **Subject Line** list, select the parameter holding the subject of the e-mail to be sent. You must define the parameters in the **Parameter** bar to make them available for the selection.
- f. In the **Priority** list select the priority of the e-mail to be sent as **High**, **Normal** or **Low**.
- g. In the **Attachment File Path** list, select the parameter holding the list of files to be attached to the e-mail. The list of files must be separated by a semi-colon (;). You must enter the file path along with file name and extension. You must define the parameters in the **Parameter** bar to make them available for the selection.
- h. In the **Content Type** list, select the type of content of the e-mail body. Available options are - **HTML** and **PlainText**. If none of the options is selected, **PlainText** is considered by default.
- i. In the **E-Mail Body Content** list, select the parameter holding the body of the e-mail. The body of the e-mail is plain text or HTML. Images are embedded in the HTML only if they are present online and if they are available and accessible to the intended recipient. Signature is automatically added if it is configured in the Outlook. You must define the parameters in the **Parameter** bar to make them available for the selection.
- j. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

Get EntryId

This activity is helpful if you want to use a load generator along with the trigger and still use Outlook. If a load balancer performs the e-mail operations, it uses an e-mail trigger and in return a Trigger ID is created. A Trigger ID comprises multiple fields such as Message ID, Subject, From, To, Date and so on.

Use this activity to retrieve the EntryID from the Message ID and other fields present in the Trigger ID. An EntryID is a unique Id field generated by the messaging storage system of Outlook application. Whenever an item is created in a folder, it is assigned a new EntryID.

Note: This activity is not dependent on the SMTP application and can be used outside the Application activity.



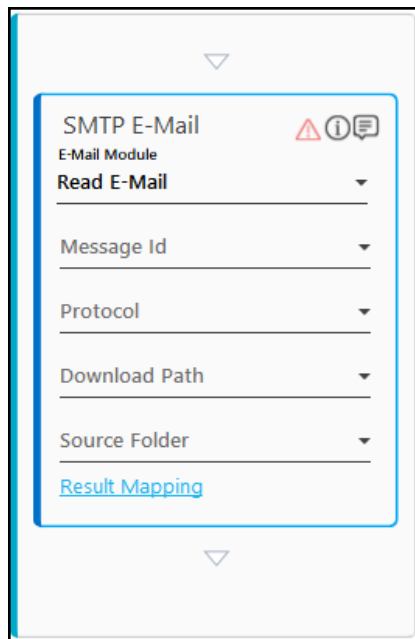
- k. In the **Trigger Id** list, select the parameter holding the **Trigger ID** from which the **EntryID** must be retrieved. You must define the parameters in the Parameter bar to make them available for the selection.

- l. In the **Profile Name** list, select the parameter holding the Outlook profile name consisting of the account and other setting details. You must define the parameter in the Parameter bar to make them available for the selection.
- m. In the **User Email** list, select the parameter holding the e-mail id whose **EntryID** must be retrieved.
- n. Click **Result Mapping** link, the **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. If the operation is successful, it returns true and if the operation fails, it returns false. It can be used as an input for the next action that you want to perform depending upon the success or failure of the operation. You must define the parameters in the **Parameter** bar to make them available for the selection.

Read E-Mail

It helps you to read the e-mail selected as per the Message ID and saves the message as a folder, at the specified location. This operation works with both Message ID and Trigger ID. Based on the input pattern, the type of ID provided is recognized by the system.

The folder gets created in the drive with the specified Message ID as the folder name. All the content of the selected e-mail along with the attachments is saved inside the created folder.



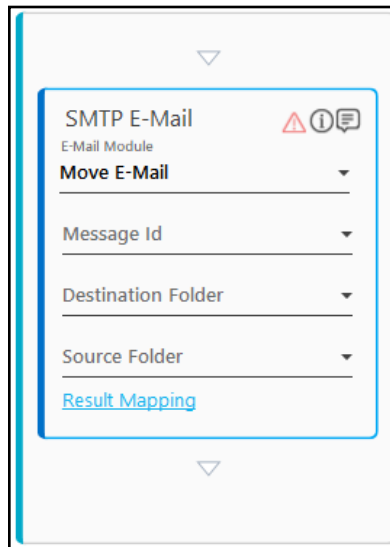
- o. In the **Message Id** list, select the parameter holding the **Message ID** for selecting the e-mail to be read. Based on the input pattern, the system automatically identifies if the input is a message Id or a trigger Id. You must define the parameters in the Parameter bar to make them available for the selection.
- p. In the **Protocol** list, select the type of protocol used to configure the SMTP server. Available options are **IMAP** and **POP3**.
- q. In the **Download Path** list, select the parameter holding the drive or folder location where the e-mail must be saved. The drive must be accessible. You must define the parameters in the **Parameter** bar to make them available for the selection.

- r. In the **Source Folder** list, select the parameter holding the folder name of the mailbox from where the e-mail must be read. this field disappears if the **Protocol** selected is **POP3**. POP3 is an old protocol, by default, the inbox is used as the source folder. You must define the parameters in the **Parameter** bar to make them available for the selection.
- s. Click **Result Mapping** link, the **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. If the operation is successful, it returns true and if the operation fails, it returns false. It can be used as an input for the next action that you want to perform depending upon the success or failure of the operation. You must define the parameters in the **Parameter** bar to make them available for the selection.

Move E-Mail

It helps you to move an e-mail, selected as per the Message ID, from one folder to another in the mailbox. This operation works with both Message ID and Trigger ID. Based on the input pattern, the type of ID provided is recognized by the system.

This activity uses the IMAP protocol by default. A Pop3 is an older protocol and does not support moving an e-mail from one location to another.



- a. In the **Message Id** list, select the parameter holding the **Message ID** for selecting the e-mail to be moved. You must define the parameters in the Parameter bar to make them available for the selection. Refer **Parameter** section in the **AE-RPA-Engage-AutomationStudio-UserManual.pdf** guide if you want to know more about parameters and how to use it.
- b. In the Destination Folder list, select the parameter holding the folder name of the mailbox to which the e-mail must be moved. You must define the parameters in the Parameter bar to make them available for the selection.
- c. In the Source Folder list, select the parameter holding the folder name of the mailbox from where the e-mail must be moved. By default, the source folder is inbox. You must define the parameters in the Parameter bar to make them available for the selection.
- d. Click the Result Mapping link. The Output Mapping list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it

available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

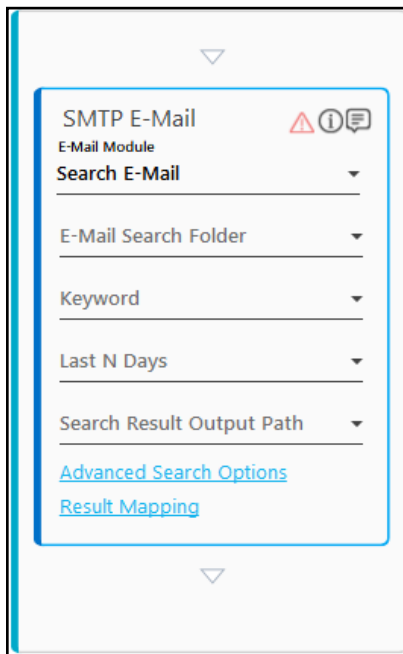
Search E-Mail

It helps you search one or more e-mails as per the specified search criteria. This operation uses the IMAP protocol to search the intended e-mails. The search criteria are applied as Boolean AND operator. You must select at least one search criteria for this operation to work. The advance search options such as subject, date range, sender of the e-mail and others, are available to set different search criteria as per your requirement.

This activity becomes the trigger point for all the operations that use Message ID or Trigger ID for selecting the intended e-mails.

All e-mails matching the search criteria, are stored in the specified Excel file (XLSX format) with their respective Message IDs. Use Excel Loop activity to iterate the data saved in the Excel file. The Message ID is read in the loop to perform operations such as Read or, Move. For example, to read e-mails based on a condition like only e- mails from a particular sender or a specific subject, use the Search E-Mail operation in SMTPMail activity to search for specific e-mails, and then perform the Read E-mail operation.

Note: Message ID is derived from the Trigger ID for a Load Generator process workflow.



- e. In the **E-Mail Search Folder** list, select the parameter holding the folder name where the required e-mail is present. The folder must be present inside the specified root folder. For example, **Inbox** or **Inbox\CustomFolder**. The default value is inbox. You must define the parameters in the Parameter bar to make them available for the selection. In the **Keyword** list, select the parameter holding the generic keyword for search such as the subject line, body, from address, to address, cc address, attachment name, and attachment content. You must define the parameters in the **Parameter** bar to make them available for the selection.

- f. In the **Last N Days** list, select the parameter holding the last n number of days, for example, 2 searches for e-mails received in the last two days. You must define the parameters in the **Parameter** bar to make them available for the selection.
- g. In the **Search Result Output Path** list, select the parameter holding the excel file path where the search result is written. The file gets automatically created if it is not present at the target location.
- h. In the **Search Result Output(Json)** list, select a parameter to store the details of the e-mail in JSON format. The Json data contains sender e-mail Id, received time, subject, body, attachment path and attachment name. You must define a **String** type parameters in the **Parameter** bar to make them available for the selection.
- i. Click **Advanced Search Options** link to set the advance search criteria. The **Outlook Search - Advanced** dialog box appears.

The screenshot shows a dialog box titled "SMTP Search - Advanced" with a close button (X) in the top right corner. The dialog is organized into several sections, each with a title and a list of dropdown menus:

- Keyword**: A single dropdown menu.
- E-Mail Search Folder** and **Last N Days**: Two dropdown menus side-by-side.
- Filters**: A section containing four dropdown menus: "E-Mail Read Status", "HasAttachment", "Received Start Date", and "Received End Date".
- Content**: A section containing five dropdown menus: "To E-MailID", "From E-MailID", "Body Content", "Subject Line", and "Attachment Name".

A blue button labeled "RESET" is located at the bottom right of the dialog box.

- i. In the **Keyword** list, select the parameter holding the generic keyword for search such as the subject line, body, from address, to address, cc address, attachment name, attachment content. You must define the parameters in the **Parameter** bar to make them available for the selection. The parameter selected in the **Search** activity block reflects in the **SMTP Search - Advanced** dialog box and vice versa.
- ii. In the **E-Mail Search Folder** list, select the parameter holding the folder name where the required e-mail is present. The folder must be present inside the specified root folder, for example, **Inbox** or **Inbox\CustomFolder**. You must define the parameters in the **Parameter** bar to make them available for the selection. The parameter selected in the **Search** activity block reflects in the **SMTP Search - Advanced** dialog box and vice versa.
- iii. In the **Last N Days** list, select the parameter holding the last n number of days, for example, 2 searches for e-mails

received in the last two days. You must define the parameters in the **Parameter** bar to make them available for the selection. The parameter selected in the **Search** activity block reflects in the **SMTP Search - Advanced** dialog box and vice versa.

- iv. In the **E-Mail Read Status** list, select the filter criteria to search the required e-mail. Available options are - **All, Read** and **Unread**.
- v. In the **HasAttachment** list, select **Yes** if you want to filter e-mails that have attachments. Select **No**, if you want to filter e-mails that do not have attachments.
- vi. In the **Received Start Date** list and **Received End Date** list, select the parameter holding the start date and the end date respectively. The dates must be in the mm/dd/yy format. It forms the date range to filter the required e-mails that are received in the specified date range. Both the lists must be selected together.
- vii. In the **To E-MailID** list, select the parameter holding the e-mail Id of the receiver to search the e-mails based on the receiver of the e-mail. You must define the parameters in the **Parameter** bar to make them available for the selection.
- viii. In the **From E-MailID** list, select the parameter holding the e-mail Id of the sender to search the e-mails based on the sender of the e-mail. You must define the parameters in the **Parameter** bar to make them available for the selection.
- ix. In the **Body Content** list, select the parameter holding the body content of the e-mail to filter the e-mail based on the body content. You must define the parameters in the **Parameter** bar to make them available for the selection.
- x. In the **Subject Line** list, select the parameter holding the subject of the e-mail to filter the e-mail based on the subject line. You must define the parameters in the **Parameter** bar to make them available for the selection.
- xi. In the **Attachment Name** list, select the parameter holding the name of the attachment to filter the e-mail based on the attachment name. You must define the parameters in the **Parameter** bar to make them available for the selection.
- xii. Click **RESET** to reset and enter the details again, else close the dialog box to save the advance search options.
- j. Click **Result Mapping** link, the **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. If the operation is successful, it returns true and if the operation fails, it returns false. It can be used as an input for the next action that you want to perform depending upon the success or failure of the operation. You must define the parameters in the **Parameter** bar to make them available for the selection.

The respective SMTP operations are configured.

SMTP Mail Properties

The properties of SMTP Mail activity are listed in the following table and can be edited in the Properties grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity.

Property Name	Usage
	<p>If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the Flowchart designer area. By default, the name is set as SMTP E-Mail . You can change the name as required.
Root Folder Name	The name of the root folder where the SMTP operations must be performed.

9.14.2 Outlook Mail

It allows you to automate outlook operations related to e-mail such as create e-mails and send them, search for the required e-mails based on various search criteria, read e-mails and download attachments, send reply, forward or delete the e-mails.

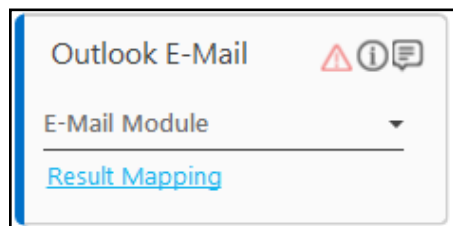
For establishing a connection with the Outlook application, credentials of the logged-in user are used. However, outlook itself may have mail-boxes of multiple users or multiple accounts.

Prerequisite

Outlook 2013 or higher version must be installed on the system.

Using Outlook Mail Activity

1. In the **Canvas Tools** pane, click **E-Mail** to expand the tool and view the associated activities.
2. Drag the **Outlook Mail** activity and drop on to the **Flowchart** designer on the **Canvas**.
3. The validation error symbol disappears when the required inputs are provided.



4. In the **E-Mail Module** list, select the operation you want to perform. Click any of the link below to know the steps to perform the selected operation:
 - [Send E-Mail](#)
 - [Reply E-Mail](#)
 - [Forward E-Mail](#)
 - [Read E-Mail](#)
 - [Search E-Mail](#)
 - [Delete E-Mail](#)
 - [Move E-Mail](#)
 - [Follow Up E-Mail](#)

Note: For more complex solutions using the Outlook such as cross-referencing items in folders, EntryID can be used as it is a unique Id field generated by the messaging storage system of Outlook application. Whenever an item is created in a folder, it is assigned a new EntryID. However, it changes if an item is moved to a different folder or if an item is exported and then imported (even to the same folder).

Send E-Mail

It helps you to create a new e- mail, attach a file, specify the priority of the e-mail, and then send the e-mail to intended recipients.

You can add multiple recipients separated by a semi-colon (;). The attachment file size upper limit is set during configuration, in the OutlookAutomationProps.xml file. All the files to be attached, must be available and accessible. Apart from the file size set in the configuration file, other Outlook restrictions related to the file size and type apply.

The screenshot shows a configuration window for 'Outlook E-Mail'. At the top right, there are three icons: a warning triangle, an information 'i' icon, and a speech bubble icon. Below the title, it says 'E-Mail Module'. The main area contains a list of dropdown menus for various email parameters: 'Send E-Mail', 'Profile Name', 'Sender E-MailID', 'To E-MailID', 'CC E-MailID', 'BCC E-MailID', 'Subject Line', 'Priority', 'Attachment file path', and 'E-Mail Body Content'. At the bottom, there is a blue link labeled 'Result Mapping'.

- a. In the **Profile Name** list, select the parameter holding the Outlook profile name consisting of the account and other setting details, that must be used for the operation. By default, the profile name is **Outlook** in the Outlook application. You must define the parameter in the **Parameter** bar to make them available for the selection.
 - i. If the Outlook application is already open and the profile name is not provided in the activity, the operation is performed on the opened profile.
 - ii. If the Outlook application is already open and the profile name provided in the activity, the current opened Outlook application is closed and the specified profile is used to re-open Outlook application.
 - iii. If an Outlook application is not opened and the profile name is not provided in the activity, the default profile name is considered to open the Outlook application and perform the required operation.
 - iv. If an Outlook application is not opened and the profile name is provided in the activity, the specified profile name is considered to open the Outlook application and perform the required operation.
- b. In the **Sender E-MailID** list, select the parameter holding the e-mail Id to be used to send the e-mail. You must define the parameters in the **Parameter** bar to make them available for the selection.
- c. In the **To E-MailID** list, select the parameter holding the e-mail Ids to which the e-mail must be sent. The Ids must be semi-colon (;) separated and the values may contain internal as well as H2O Ids. You must define the parameters in the **Parameter** bar to make them available for the selection.
- d. In the **CC E-MailID** list, select the parameter holding the e-mail Ids of the recipients whom you want to copy on the e-

mail publicly. The Ids must be semi-colon (;) separated and the values may contain internal as well as external Ids. You must define the parameters in the **Parameter** bar to make them available for the selection.

- e. In the **BCC E_MailID** list, select the parameter holding the e-mail Ids of the recipients whom you want to copy on the e-mail privately. The Ids must be semi-colon (;) separated and the values may contain internal as well as external Ids. You must define the parameters in the **Parameter** bar to make them available for the selection.
- f. In the **Subject Line** list, select the parameter holding the subject of the e-mail to be sent. You must define the parameters in the **Parameter** bar to make them available for the selection.
- g. In the **Priority** list, select the priority of the e-mail to be sent as **High, Normal, or Low**.
- h. In the **Attachment File Path** list, select the parameter holding the list of files to be attached to the e-mail. The list of files must be separated by a semi-colon (;). You must enter the file path along with file name and extension. You must define the parameters in the **Parameter** bar to make them available for the selection.
- i. In the **E-Mail Body Content** list, select the parameter holding the body of the e-mail. The body of the e-mail is plain text or HTML. Images are embedded in the HTML only if they are present online and if they are available and accessible to the intended recipient. Signature is automatically added if it is configured in the Outlook. You must define the parameters in the **Parameter** bar to make them available for the selection.
- j. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

Reply E-Mail

It helps you to reply to the e-mail selected as per the EntryID field present in the Outlook for the given profile. The reply can be sent to the sender or to all the recipients marked on the e-mail. You can attach a file, mention the priority of the e-mail, and then send the e-mail to intended recipients where RE is automatically added to the subject of the e-mail.

The attachment file size upper limit is set during the configuration in the OutlookAutomationProps.xml file. Apart from the file size set in the configuration file, other Outlook restrictions related to the file size and type apply.

The screenshot shows a configuration window for 'Outlook E-Mail'. At the top, there are three icons: a warning triangle, an information 'i' icon, and a speech bubble icon. Below the title, it says 'E-Mail Module' and 'Reply E-Mail' with a dropdown arrow. The following fields are listed with dropdown arrows: Profile Name, E-Mail Entry ID, CC E-MailID, BCC E-MailID, Priority, Attachment file path, E-Mail Body, Reply All?, Result Mapping (a blue link), and Output Mapping.

- k. In the **Profile Name** list, select the parameter holding the Outlook profile name consisting of the account and other setting details, that must be used for the operation. By default, the profile name is **Outlook** in the Outlook application. You must define the parameter in the Parameter bar to make them available for the selection.
 - i. If the Outlook application is already open and the profile name is not provided in the activity, the operation is performed on the opened profile.
 - ii. If the Outlook application is already open and the profile name provided in the activity, the current opened Outlook application is closed and the specified profile is used to re-open Outlook application.

- iii. If an Outlook application is not opened and the profile name is not provided in the activity, the default profile name is considered to open the Outlook application and perform the required operation.
- iv. If an Outlook application is not opened and the profile name is provided in the activity, the specified profile name is considered to open the Outlook application and perform the required operation.
- l. In the **E-Mail Entry Id** list, select the parameter holding the **EntryID** for selecting the e-mail to be replied. You must define the parameters in the **Parameter** bar to make them available for the selection.
- m. In the **CC E-MailID** list, select the parameter holding the e-mail Ids of the recipients whom you want to copy on the e-mail publicly. The Ids must be semi-colon (;) separated and the values may contain internal as well as external Ids. You must define the parameters in the **Parameter** bar to make them available for the selection.
- n. In the **BCC E-MailID** list, select the parameter holding the e-mail Ids of the recipients to whom you want to copy on the e-mail privately. The Ids must be semi-colon (;) separated and the values may contain internal as well as external Ids. You must define the parameters in the **Parameter** bar to make them available for the selection.
- o. In the **Priority** list select the priority of the e-mail to be sent as **High, Normal, or Low**.
- p. In the **Attachment File Path** list, select the parameter holding the list of files to be attached to the e-mail. The list of files must be separated by a semi-colon (;). You must enter the file path along with file name and extension. All the files to be attached, must be available and accessible. Apart from the file size set in the configuration file, other outlook based file size and type restrictions apply. You must define the parameters in the **Parameter** bar to make them available for the selection.
- q. In the **E-Mail Body Content** list, select the parameter holding the body of the e-mail. The body of the e-mail is plain text or HTML. Images are embedded in the HTML only if they are present online and if they are available and accessible to the intended recipient. Signature is automatically added if it is configured in the Outlook. You must define the parameters in the **Parameter** bar to make them available for the selection.
- r. In the **Reply All?** list, select **true** if you want use the reply all feature of Outlook. If **false** is selected, the reply is sent only to the sender of the e-mail.
- s. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

Forward E-Mail

It helps you to forward the e-mail selected as per the EntryID field present in the outlook for the given profile. You can attach more files apart from the existing attachments that gets added automatically, mention the priority of the e-mail, and then send the e-mail to intended recipients where FW is automatically added to the subject of the e-mail.

The attachment file size upper limit is set during the configuration in the OutlookAutomationProps.xml file. Apart from the file size set in the configuration file, other Outlook restrictions related to the file size and type apply.

- a. In the **Profile Name** list, select the parameter holding the Outlook profile name consisting of the account and other setting details, that must be used for the operation. By default, the profile name is **Outlook** in the Outlook application. You must define the parameter in the Parameter bar to make them available for the selection.
 - i. If the Outlook application is already open and the profile name is not provided in the activity, the operation is performed on the opened profile.
 - ii. If the Outlook application is already open and the profile name provided in the activity, the current opened Outlook application is closed and the specified profile is used to re-open Outlook application.
 - iii. If an Outlook application is not opened and the profile name is not provided in the activity, the default profile name is considered to open the Outlook application and perform the required operation.
 - iv. If an Outlook application is not opened and the profile name is provided in the activity, the specified profile name is considered to open the Outlook application and perform the required operation.

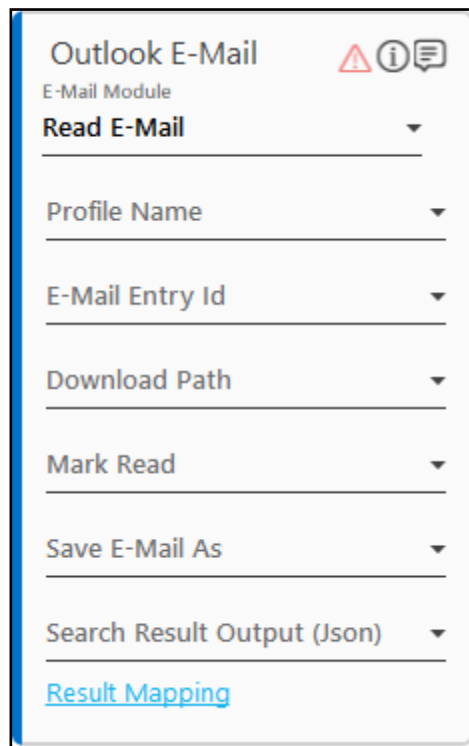
- b. In the **E-Mail Entry Id** list, select the parameter holding the **EntryID** for selecting the e-mail to be forwarded. You must define the parameters in the **Parameter** bar to make them available for the selection.
- c. In the **Forward To E-MailID** list, select the parameter holding the e-mail Ids to whom you want to forward the e-mail. The Ids must be semi-colon (;) separated values that may contain internal as well as external Ids. You must define the parameters in the **Parameter** bar to make them available for the selection.
- d. In the **CC E-MailID** list, select the parameter holding the e-mail Ids of the recipients whom you want to copy on the e-mail publicly. The Ids must be semi-colon (;) separated and the values may contain internal as well as external Ids. You must define the parameters in the **Parameter** bar to make them available for the selection.
- e. In the **BCC E-MailID** list, select the parameter holding the e-mail Ids of the recipients whom you want to copy on the e-mail privately. The Ids must be semi-colon (;) separated and the values may contain internal as well as external Ids. You must define the parameters in the **Parameter** bar to make them available for the selection.
- f. In the **Priority** list select the priority of the e-mail to be sent as **High, Normal, or Low**.
- g. In the **Attachment File Path** list, select the parameter holding the list of files to be attached to the e-mail. The list of files must be separated by a semi-colon (;). You must enter the file path along with file name and extension. All the files to be attached, must be available and accessible. Apart from the file size set in the configuration file, other outlook based file size and type restrictions apply. You must define the parameters in the **Parameter** bar to make them available for the selection.
- h. In the **E-Mail Body Content** list, select the parameter holding the body of the e-mail. The body of the e-mail is plain text or HTML. Images are embedded in the HTML only if they are present online and if they are available and accessible to the intended recipient. Signature is automatically added if it is configured in the Outlook. You must define the parameters in the **Parameter** bar to make them available for the selection.
- i. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

Read E-Mail

It helps you to read the e-mail selected as per the EntryID field present in the Outlook for the given profile and download it in the specified format. It saves the message either as a .msg format file or as a folder, containing the subject, mail body, and attachment files.

If the .msg format is created, the e-mail is saved with the subject line as the file name. In case the file already exists at the target location with the same name, the received data time stamp is appended in the file name.

If the folder is created, it is saved with the EntryID as the folder name.



- a. In the **Profile Name** list, select the parameter holding the Outlook profile name consisting of the account and other setting details, that must be used for the operation. By default, the profile name is **Outlook** in the Outlook application. You must define the parameter in the Parameter bar to make them available for the selection.
 - i. If the Outlook application is already open and the profile name is not provided in the activity, the operation is performed on the opened profile.
 - ii. If the Outlook application is already open and the profile name provided in the activity, the current opened Outlook application is closed and the specified profile is used to re-open Outlook application.
 - iii. If an Outlook application is not opened and the profile name is not provided in the activity, the default profile name is considered to open the Outlook application and perform the required operation.
 - iv. If an Outlook application is not opened and the profile name is provided in the activity, the specified profile name is considered to open the Outlook application and perform the required operation.
- b. In the **E-Mail Entry Id** list, select the parameter holding the **EntryID** for selecting the e-mail to be read. You must

define the parameters in the **Parameter** bar to make them available for the selection.

- c. In the **Download Path** list, select the parameter holding the folder location where the e-mail must be saved. The folder must be accessible. You must create a directory if the directory is not present. There must be privileges available to create the directory. You must define the parameters in the **Parameter** bar to make them available for the selection.
- d. In the **Mark Read** list, select **true** if you want to mark the e-mail as read in the Outlook.
- e. In the **Save E-Mail As** list, select:
 - MSG if you want to save the e-mail as .msg format file.
 - Folder if you want to save the e-mail as a folder or
 - All if you want to save the e-mail as both .msg and folder.
- f. In the **Search Result Output(Json)** list, select a parameter to store the details of the e-mail in Json format. The JSON data contains sender e-mail Id, received time, subject, body, attachment path and attachment name. You must define a **String** type parameters in the **Parameter** bar to make them available for the selection.
- g. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

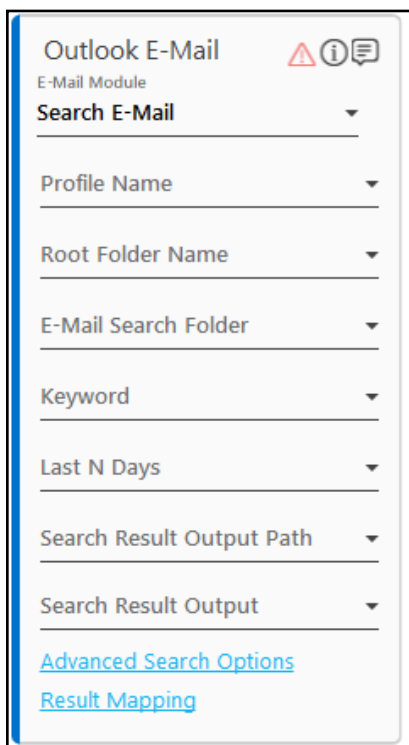
Search E-Mail

It helps you to search for one or more e-mails as per the specified search criteria. This operation can search in any of the Outlook folders that is present under any of the account. The search criteria is applied as a Boolean AND operator. You must select at least one search criteria for this operation to work. The advance search options such as subject, date range, sender of the e-mail and others, are available to set different search criteria as per your requirement.

This operation searches the folder present at any level in Outlook, apart from the root level, as long as the level is a folder that contains items. If a parent folder is selected, all the sub-folders are searched for the required e-mail.

If there are duplicate folder names, the first occurrence of the folder with the specified name is searched. It is recommended not to have duplicate folders in outlook.

Additionally, you can search a specific account that is added in the Outlook.



- h. In the **Profile Name** list, select the parameter holding the Outlook profile name consisting of the account and other setting details, that must be used for the operation. By default, the profile name is **Outlook** in the Outlook application. You must define the parameter in the Parameter bar to make them available for the selection.
 - i. If the Outlook application is already open and the profile name is not provided in the activity, the operation is performed on the opened profile.
 - ii. If the Outlook application is already open and the profile name provided in the activity, the current opened Outlook application is closed and the specified profile is used to re-open Outlook application.
 - iii. If an Outlook application is not opened and the profile name is not provided in the activity, the default profile name is considered to open the Outlook application and perform the required operation.
 - iv. If an Outlook application is not opened and the profile name is provided in the activity, the specified profile name is considered to open the Outlook application and perform the required operation.
- i. In the **Root Folder Name** list, select the parameter holding the root folder in the Outlook or Archives to perform the search. For example, username@domain.com. You must define the parameters in the **Parameter** bar to make them available for the selection.
- j. In the **E-Mail Search Folder** list, select the parameter holding the folder name where the required e-mail is present. The folder must be present inside the specified root folder. For example, **Inbox** or **Inbox\CustomFolder**. You must define the parameters in the **Parameter** bar to make them available for the selection.
- k. In the **Keyword** list, select the parameter holding the generic keyword for search such as the subject line, body, from address, to address, cc address, attachment name and attachment content. You must define the parameters in the **Parameter** bar to make them available for the selection.

- i. In the **Last N Days** list, select the parameter holding the last n number of days for which you want to perform the search. For example, **2** searches for e-mails received in the last two days. You must define the parameters in the **Parameter** bar to make them available for the selection.
- m. In the **Search Result Output Path** list, select the parameter holding the excel file path where the search result is written. The file automatically gets created if it is not present at the target location.
- n. In the **Search Result Output(Json)** list, select a parameter to store the details of the e-mail in Json format. The Json data contains sender e-mail Id, received time, subject, body, attachment path, and attachment name. You must define a **String** type parameters in the **Parameter** bar to make them available for the selection.
- o. Click the **Advanced Search Options** link to set the advance search criteria. The **Outlook Search - Advanced** dialog box appears.

Outlook Search - Advanced [X]

Keyword [v] Root Folder Name [v]
 E-Mail Search Folder [v] Last N Days [v]

Filters

E-Mail Read Status [v] HasAttachment [v]
 Received Start Date [v] Received End Date [v]
 Last N E-mails [v]

Content

To E-MailID [v] From E-MailID [v]
 Body Content [v] Subject Line [v]
 Attachment Name [v] Attachment Content [v]

RESET

- i. In the **Keyword** list, select the parameter holding the generic keyword for search such as the subject line, body, from address, to address, cc address, attachment name, attachment content. You must define the parameters in the **Parameter** bar to make them available for the selection. The parameter selected in the **Search** activity block reflects in the **Outlook Search - Advanced** dialog box and vice versa.
- ii. In the **Root Folder Name** list, select the parameter holding the root folder in the Outlook or Archives to perform the search, for example, username@domain.com. You must define the parameters in the **Parameter** bar to make them available for the selection. The parameter selected in the **Search** activity block reflects in the **Outlook Search - Advanced** dialog box and vice versa.
- iii. In the **E-Mail Search Folder** list, select the parameter holding the folder name where the required e-mail is present. The folder must be present inside the specified root folder, for example, **Inbox** or **Inbox\CustomFolder**. You must

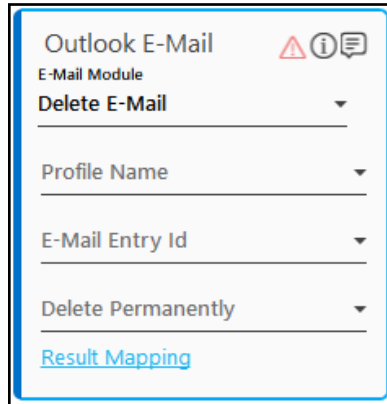
define the parameters in the **Parameter** bar to make them available for the selection. The parameter selected in the **Search** activity block reflects in the **Outlook Search - Advanced** dialog box and vice versa.

- iv. In the **Last N Days** list, select the parameter holding the last n number of days. For example, if you specify **2**, it searches for e-mails received in the last two days. You must define the parameters in the **Parameter** bar to make them available for the selection. The parameter selected in the **Search** activity block reflects in the **Outlook Search - Advanced** dialog box and vice versa.
 - v. In the **E-Mail Read Status** list, select the filter criteria to search the required e-mail. Available options are - **All**, **Read** and **Unread**.
 - vi. In the **HasAttachment** list, select **Yes** if you want to filter e-mails that have attachments. Select **No**, if you want to filter e-mails that do not have attachments.
 - vii. In the **Received Start Date** list and **Received End Date** list, select the parameter holding the start date and the end date respectively. The dates must be in the mm/dd/yy format. It forms the date range to filter the required e-mails that are received in the specified date range. Both the lists must be selected together.
 - viii. In the **Last N E-mails** list, select the parameter holding the last n number of e-mails, for example, **2** searches for the last two e-mails received. You must define the parameters in the **Parameter** bar to make them available for the selection.
 - ix. In the **To E-MailID** list, select the parameter holding the e-mail Id of the receiver to search the e-mails based on the receiver of the e-mail. You must define the parameters in the **Parameter** bar to make them available for the selection.
 - x. In the **From E-MailID** list, select the parameter holding the e-mail Id of the sender to search the e-mails based on the sender of the e-mail. You must define the parameters in the **Parameter** bar to make them available for the selection.
 - xi. In the **E-Mail Body Content** list, select the parameter holding the body content of the e-mail to filter the e-mail based on the body content. You must define the parameters in the **Parameter** bar to make them available for the selection.
 - xii. In the **Subject Line** list, select the parameter holding the subject of the e-mail to filter the e-mail based on the subject line. You must define the parameters in the **Parameter** bar to make them available for the selection.
 - xiii. In the **Attachment Name** list, select the parameter holding the name of the attachment to filter the e-mail based on the attachment name. You must define the parameters in the **Parameter** bar to make them available for the selection.
 - xiv. In the **Attachment Content** list, select the parameter holding the name of the attachment to filter the e-mail based on the content inside the attachment. You must define the parameters in the **Parameter** bar to make them available for the selection.
 - xv. Click **RESET** to reset and enter the details again, else close the dialog box to save the advance search options.
- p. Click **Result Mapping** link, the **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. If the operation is successful, it returns true and if the operation fails, it returns false. It can be used as an input for the next action that you want to perform depending upon the success or failure of the operation. You must define the parameters in the **Parameter** bar to make them

available for the selection.

Delete E-Mail

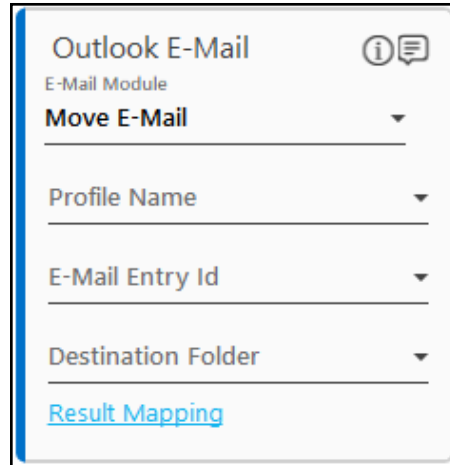
It helps you to delete an e-mail selected as per the EntryID field present in the Outlook for the given profile. You can soft delete (move the specified e-mail in the Deleted Items folder) or hard delete (permanently delete) an e-mail. Only a single e-mail can be deleted at a time.



- a. In the **Profile Name** list, select the parameter holding the Outlook profile name consisting of the account and other setting details, that must be used for the operation. By default, the profile name is **Outlook** in the Outlook application. You must define the parameter in the Parameter bar to make them available for the selection.
 - i. If the Outlook application is already open and the profile name is not provided in the activity, the operation is performed on the opened profile.
 - ii. If the Outlook application is already open and the profile name provided in the activity, the current opened Outlook application is closed and the specified profile is used to re-open Outlook application.
 - iii. If an Outlook application is not opened and the profile name is not provided in the activity, the default profile name is considered to open the Outlook application and perform the required operation.
 - iv. If an Outlook application is not opened and the profile name is provided in the activity, the specified profile name is considered to open the Outlook application and perform the required operation.
- b. In the **E-Mail Entry Id** list, select the parameter holding the **EntryID** for selecting the e-mail to be deleted. You must define the parameters in the **Parameter** bar to make them available for the selection.
- c. In the **Delete Permanently** list, select **true** if you want to delete the specified e-mail permanently. You can select **false** to perform soft delete of the e-mail.
- d. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

Move E-mail

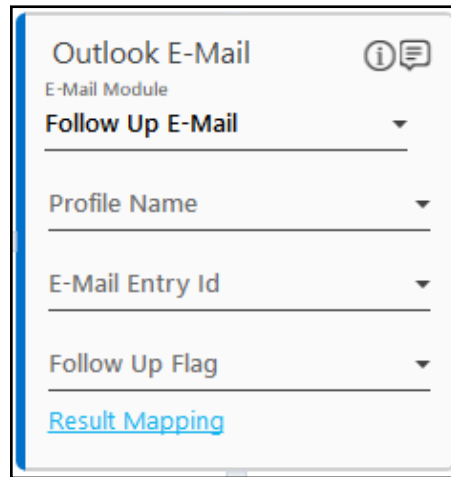
It helps you to move an e-mail selected as per the EntryID field present in the Outlook for the given profile, to the specified folder. Only a single e-mail can be moved at a time.



- a. In the **Profile Name** list, select the parameter holding the Outlook profile name consisting of the account and other setting details, that must be used for the operation. By default, the profile name is **Outlook** in the Outlook application. You must define the parameters in the Parameter bar to make them available for the selection.
 - i. If the Outlook application is already open and the profile name is not provided in the activity, the operation is performed on the opened profile.
 - ii. If the Outlook application is already open and the profile name provided in the activity, the current opened Outlook application is closed and the specified profile is used to re-open Outlook application.
 - iii. If an Outlook application is not opened and the profile name is not provided in the activity, the default profile name is considered to open the Outlook application and perform the required operation.
 - iv. If an Outlook application is not opened and the profile name is provided in the activity, the specified profile name is considered to open the Outlook application and perform the required operation.
- b. In the **E-Mail Entry Id** list, select the parameter holding the **EntryID** for selecting the e-mail to be moved. You must define the parameters in the **Parameter** bar to make them available for the selection.
- c. In the **Destination Folder** list, select the parameter holding the folder name to which the e-mail must be moved.
- d. Click **Result Mapping** link, the **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. If the operation is successful, it returns true and if the operation fails, it returns false. It can be used as an input for the next action that you want to perform depending upon the success or failure of the operation. You must define the parameters in the **Parameter** bar to make them available for the selection.

Follow Up E-Mail

It helps you to mark a Follow Up flag to the e-mail selected as per the EntryID field present in the outlook for the given profile.



- a. In the **Profile Name** list, select the parameter holding the Outlook profile name consisting of the account and other setting details, that must be used for the operation. By default, the profile name is **Outlook** in the Outlook application. You must define the parameter in the **Parameter** bar to make them available for the selection.
 - i. If the Outlook application is already open and the profile name is not provided in the activity, the operation is performed on the opened profile.
 - ii. If the Outlook application is already open and the profile name provided in the activity, the current opened Outlook application is closed and the specified profile is used to re-open Outlook application.
 - iii. If an Outlook application is not opened and the profile name is not provided in the activity, the default profile name is considered to open the Outlook application and perform the required operation.
 - iv. If an Outlook application is not opened and the profile name is provided in the activity, the specified profile name is considered to open the Outlook application and perform the required operation.
- b. In the **E-Mail Entry Id** list, select the parameter holding the **EntryID** for selecting the e-mail to be moved. You must define the parameters in the **Parameter** bar to make them available for the selection.
- c. In the **Follow Up Flag** list, select the follow-up flag to remind you at a later time. Available options are - **Today, Tomorrow, This Week, Next Week, No Date** and **Mark Complete**. All the available flags work as per the options available in the Outlook application.
- d. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

The respective Outlook operations are configured.

Outlook Mail Properties

The properties of Outlook Mail activity are listed in the following table and can be edited in the Properties grid on the right pane.

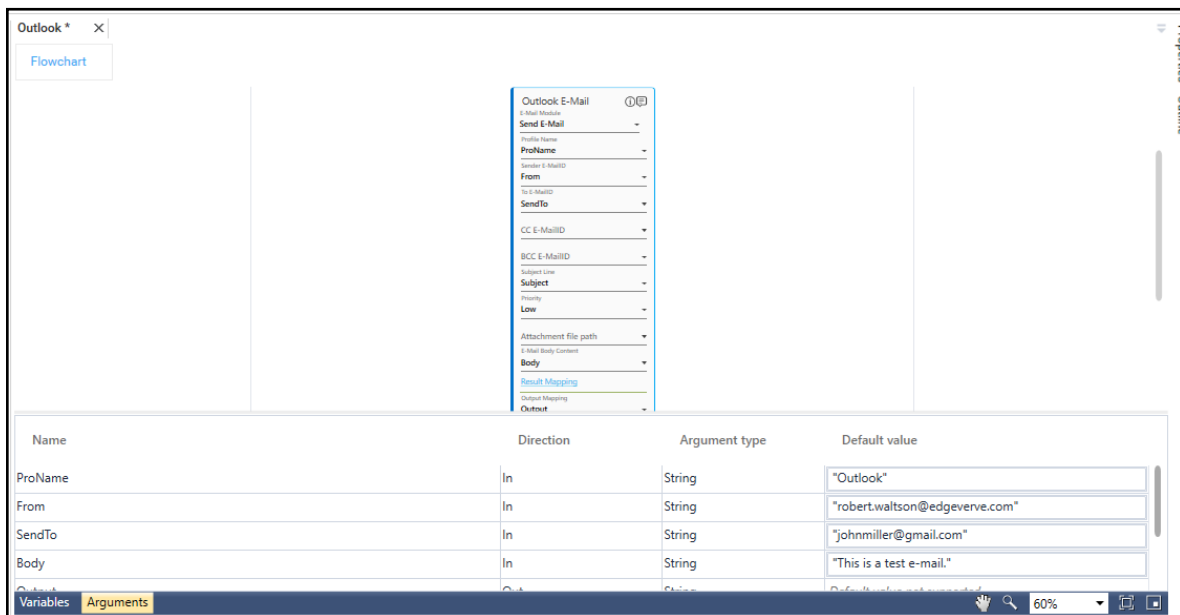
Property Name	Usage
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity.</p> <p>If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the Flowchart designer area. By default, the name is set as Outlook E-Mail . You can change the name as required.
Root Folder Name	The name of the root folder where the Outlook operations must be performed.

Example of Outlook Mail

Let's create an example to automate sending e-mail from an Outlook account to an external recipient.

To automate sending an e-mail:

1. Create a new process.
2. From the **Canvas Tools** panel, add the Outlook E-Mail activity to the **Flowchart** designer on the **Canvas**.
3. In the Parameter bar, create **In** arguments:
 - a. **ProName** - to store the profile name of the Outlook application
 - b. **From** - to store the e-mail Id of the sender
 - c. **SendTo** - to store the e-mail Id of the recipient
 - d. **Subject** - to store the subject line of the e-mail
 - e. **Body** - to store the body content of the e-mail
4. In the **Parameter** bar, create **Out** argument, **Output** to store the return value of the success or failure of the operation.



5. In the **Profile Name** list, select the **ProName** argument, created above.
6. In the **Sender E-MailID** list, select the **From** argument, created above.
7. In the **To E-MailID** list, select the **SendTo** argument, created above.
8. In the **E-Mail Body Content** list, select the **Body** argument, created above.
9. Click **Result Mapping** link. The **Output Mapping** list appears. Select Output argument, created above.

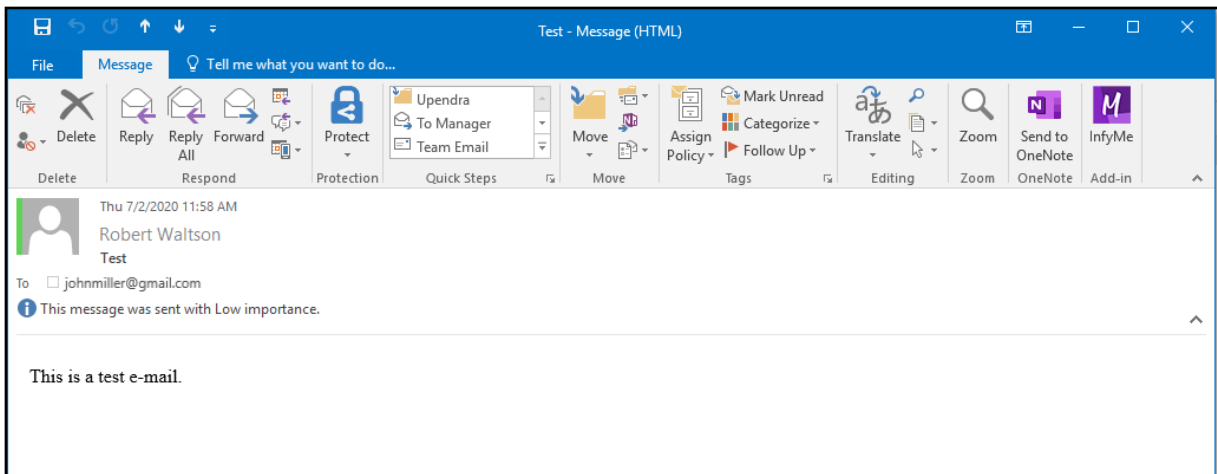
To view the output in Automation Studio, let's add WriteLine activity that returns True, if the e-mail sending is successful. You can assign this process to a robot, if you want to execute this process outside Automation Studio.

1. Add a **WriteLine** activity.
2. Save the process.
3. Setup the environment and then perform test run.

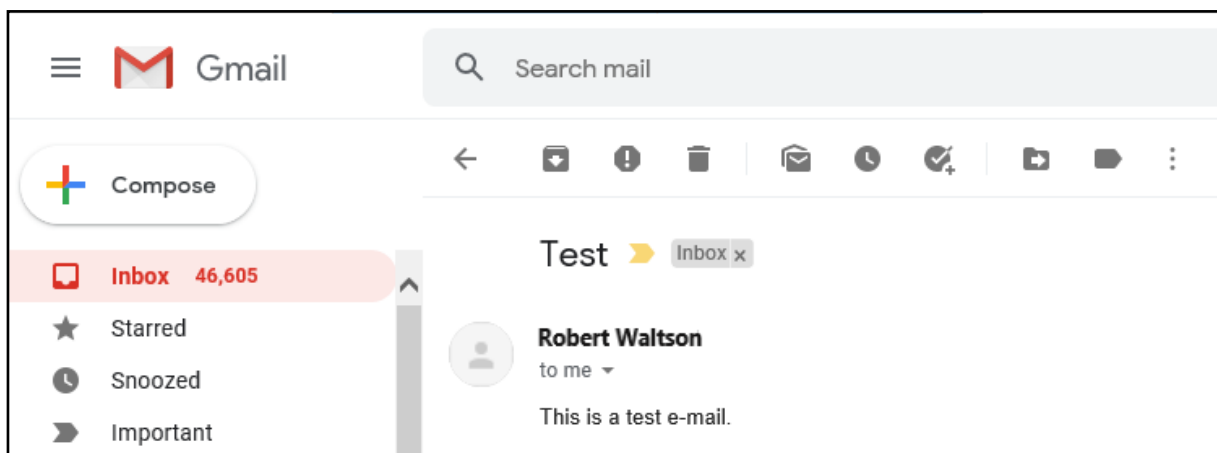
The Output console displays True that signifies that the e-mail is sent successfully.



Below is a sample e-mail sent from Outlook application:



Below is the sample e-mail received on Gmail application:



9.15 SAP Thick Client

The SAP Thick Client allows you to automate SAP thick client applications. A thick client provides full functionality of the application without having a server communication.

Note: SAP Automation is considered as normal web automation. It not dependent upon AE platform SAP automation settings.

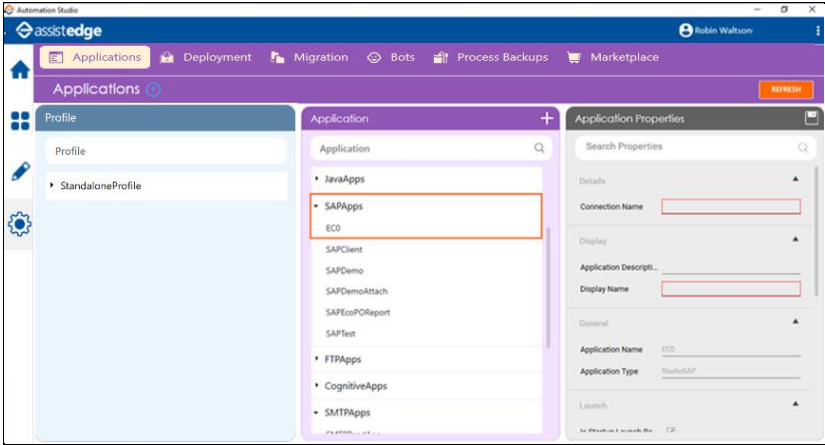

You can record steps performed and even fetch values from a SAP application using the following activities:

- [SAP Recorder](#)
- [GUI Field Extractor](#)
- [Attach Application](#)

9.15.1 Prerequisites

Following are the prerequisites required for using the SAP Thick Client activities:

Requirement	Description								
Enterprise SAP Connection	You must have an active enterprise SAP connection.								
Assembly Files	<p>Save the following assembly files in the Automation Studio folder saved on your system:</p> <table border="1"> <thead> <tr> <th>Component</th> <th>Version</th> </tr> </thead> <tbody> <tr> <td>Interop.SAPFEWSELib.dll</td> <td>SAP ECC 6/ECC 4 and above</td> </tr> <tr> <td>Interop.SapROTWr.dll</td> <td>SAP ECC 6/ECC 4 and above</td> </tr> <tr> <td>stdole.dll</td> <td>SAP ECC 6/ECC 4 and above</td> </tr> </tbody> </table> <p>To save the files:</p> <ol style="list-style-type: none"> 1. Download the mentioned assembly files from SAP website on your system. 2. Save Interop.SAPFEWSELib.dll, Interop.SapROTWr.dll and stdole.dll in client-tools > AutomationStudio > bin > ThirdPartyDLLs folder. If you download/access Automation Studio from the Admin module, you must save the required DLLs at %localappdata% > EdgeVerve > AutomationStudio > bin > ThirdPartyDLLS folder. 3. Save Interop.SAPFEWSELib.dll and Interop.SapROTWr.dll in the client-tools > AutomationStudio > bin > SAPEngine folder. If you download/access Automation Studio from the Admin module, you must save the required DLLs at %localappdata% > EdgeVerve > AutomationStudio > bin > SAP 	Component	Version	Interop.SAPFEWSELib.dll	SAP ECC 6/ECC 4 and above	Interop.SapROTWr.dll	SAP ECC 6/ECC 4 and above	stdole.dll	SAP ECC 6/ECC 4 and above
Component	Version								
Interop.SAPFEWSELib.dll	SAP ECC 6/ECC 4 and above								
Interop.SapROTWr.dll	SAP ECC 6/ECC 4 and above								
stdole.dll	SAP ECC 6/ECC 4 and above								

Requirement	Description
<p>Configure SAP application in Automation Studio</p>	<p>folder.</p> <p>The intended SAP application must be configured in Automation Studio. Below are the minimum required properties for configuring SAP application. If you want to define remaining properties, refer SAPApps in the Application tab of Admin menu.</p> <p>To configure the SAP application:</p> <ol style="list-style-type: none"> In the Admin menu, add an application of Application Type- SAPApps.  <ol style="list-style-type: none"> In the Application Properties panel: <ul style="list-style-type: none"> In the Connection Name field, enter the name of the active enterprise SAP connection. In the Display Name field, enter a name of the SAP application that you want to configure. Click the  (Save Properties) icon to save the application details. The SAPApps application is configured.

9.15.2 Addition of SAP Application

Addition of SAP application requires an application name and SAP Connection name.

9.15.3 SAP Recorder

This activity records the steps performed on a SAP thick client application such as:

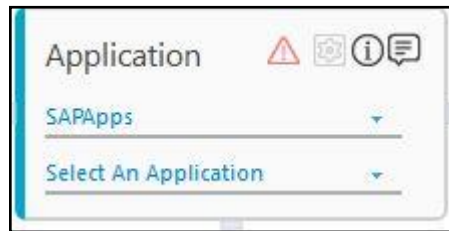
- Entering the transaction code,

- Navigating through the application,
- Setting data in the applications,
- Clicking different controls and searching data,
- Comparing data fields on the screen,
- Data extraction, and
- Others

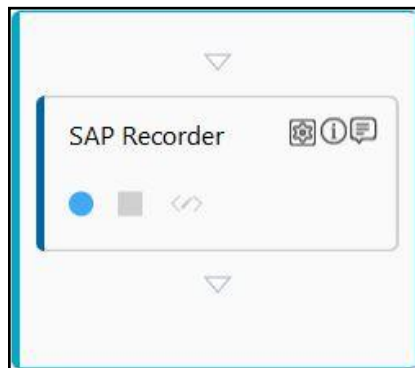
This activity must be used inside an Application activity or the system displays an error.



Using SAP Recorder Activity

1. Make sure that the [prerequisites](#) for using SAP Recorder activity is met.
2. In the **Canvas Tools** pane, click **Process Components** to expand the tool and view the associated activities.
3. Drag the **Application** activity and drop on to the **Flowchart** designer on the **Canvas**.



4. In the **Application Type** list, select **SAPApps**. Application type that has at least one application added to it appears in the list.
5. In the **Select An Application** list, select the SAP application you want automate. You can add a SAP application that is already configured in the Automation Studio or add a new application at this level.
6. Double click the **Application** activity to open the activity container.
7. In the **Canvas Tools** pane, click **SAP Thick Client** to expand the tool and view the associated activities.
8. Drag the **SAP Recorder** activity and drop in the **Application** activity.

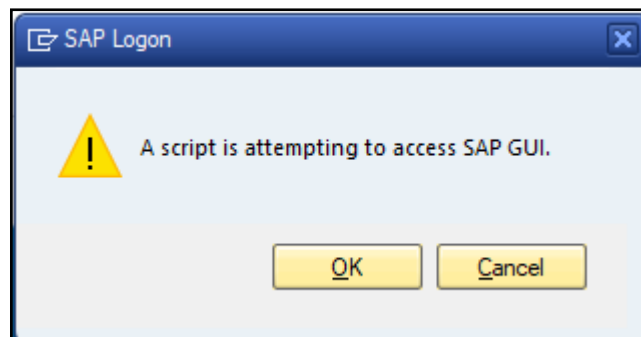


- Click the  (**Start Recording**) icon to record the steps performed in the SAP application. You can click the  (**Start Recording**) icon to overwrite the current recording and to re-record the steps. Here we are taking an example of logging into a SAP application.

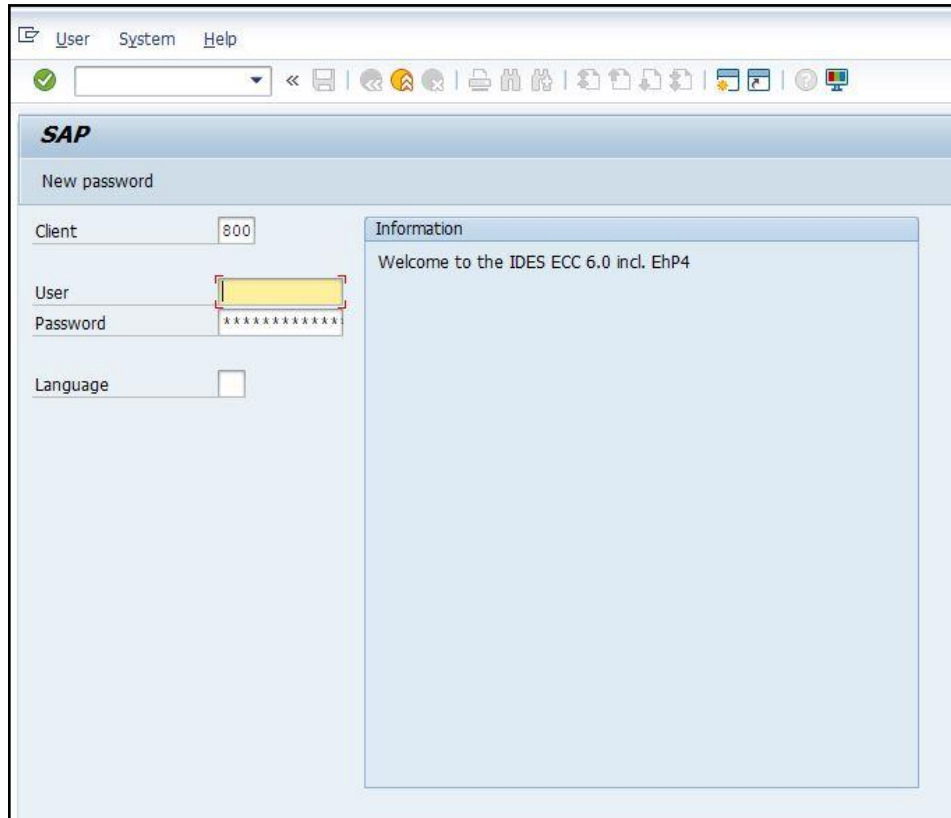
Note:


- Launch SAP application before you start using the activity else the Start Recording icon disabled.
- You must keep the relevant page of SAP application open before you start recording.

The SAP Logon pop message appears.





- Click **OK** to close the message box to start recording the steps. The available SAP application window appears. Perform all the steps that you want to automate for the recorder to capture.



11. In the **SAP Recorder** activity, click the  (**Stop Recording**) icon to stop capturing the steps. As a result of recording the steps in the SAP application, related scripts are created in the **Script Editor**.



12. You can use the  (**Edit Script**) icon to view or to edit the recording. Click **CONFIRM** once edited.
13. Click the  (**Settings**) icon and then click **Input Mapping** to map corresponding input fields with parameters of the **SAP Recorder** activity.

Input Field	Select Argument	If Default	Default Value
rowindex	<input type="text"/>	<input checked="" type="checkbox"/>	0
input3	<input type="text"/>	<input type="checkbox"/>	
input4	<input type="text"/>	<input type="checkbox"/>	

CONFIRM

The SAP Input Mapping dialog box appears. The Input Field column signifies the fields captured in the SAP application. The Default Value fields hold the values that you entered as part of the automation steps captured.

14. In the **Select Argument** list, select the argument you want to map with the corresponding input field. You must define the parameter in the [Parameter](#) bar with the input values for mapping. Clear the **If Default** check box for the fields you do not want to set the value captured as the input parameter for the automation process.

15. Click **CONFIRM**.

The fields are configured and the Application activity for Application Type-SAPApps is created.

SAP Recorder Properties

The properties of SAP Recorder activity are listed in the following table and can be edited in the Property grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	When you set this option to Yes , the application ignores any error while executing the activity. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	

Property Name	Usage
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the Flowchart designer. By default, the name is set as SAP Recorder . You can change the name as required.
SessionId	A numeric property to indicate which session id of the SAP is used to perform the action. It is used in case of multiple session scenarios of SAP. For example, when a new session is opened on some action in first SAP session. Default value is set to 0 .

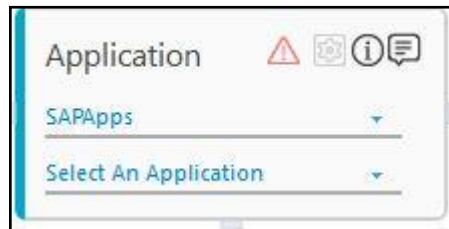
9.15.4 GUI Field Extractor

This activity allows you to fetch data from a SAP thick client application.

This activity must be used inside an Application activity or the system displays an error.

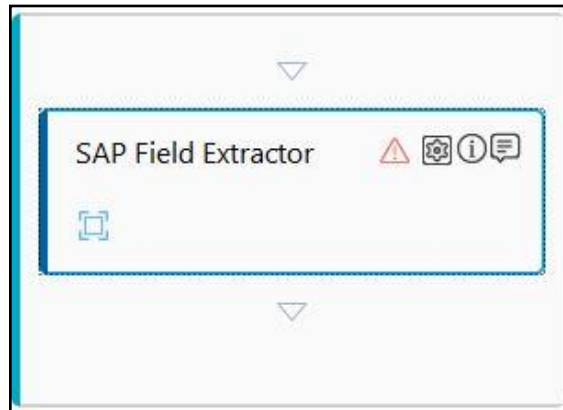
Using GUI Field Extractor Activity


1. Make sure that the [prerequisites](#) for using SAP Recorder activity is met.
2. In the **Canvas Tools** pane, click **Process Components** to expand the tool and view the associated activities.
3. Drag the **Application** activity and drop on to the **Flowchart** designer area.



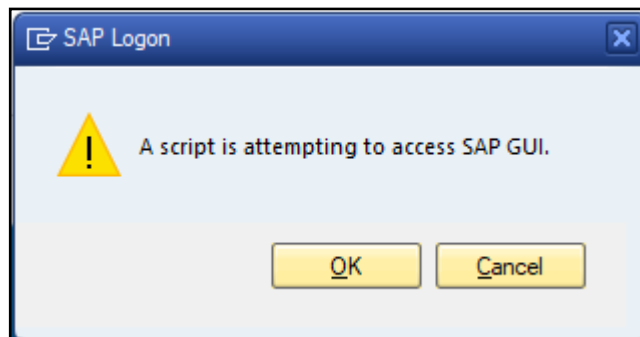
4. In the **Application Type** list, select **SAPApps**. Application type that has at least one application added to it appears in the drop-down list.
5. In the **Select An Application** list, select the SAP application you want to automate. You can add a SAP application that is already configured in Automation Studio or add a new application at this level.
6. Double click the **Application** activity to open the activity container.
7. In the **Canvas Tools** pane, click **SAP Thick Client** to expand the tool and view the associated activities.

8. Drag the **GUI Field Extractor** activity and drop in the **Application** activity.



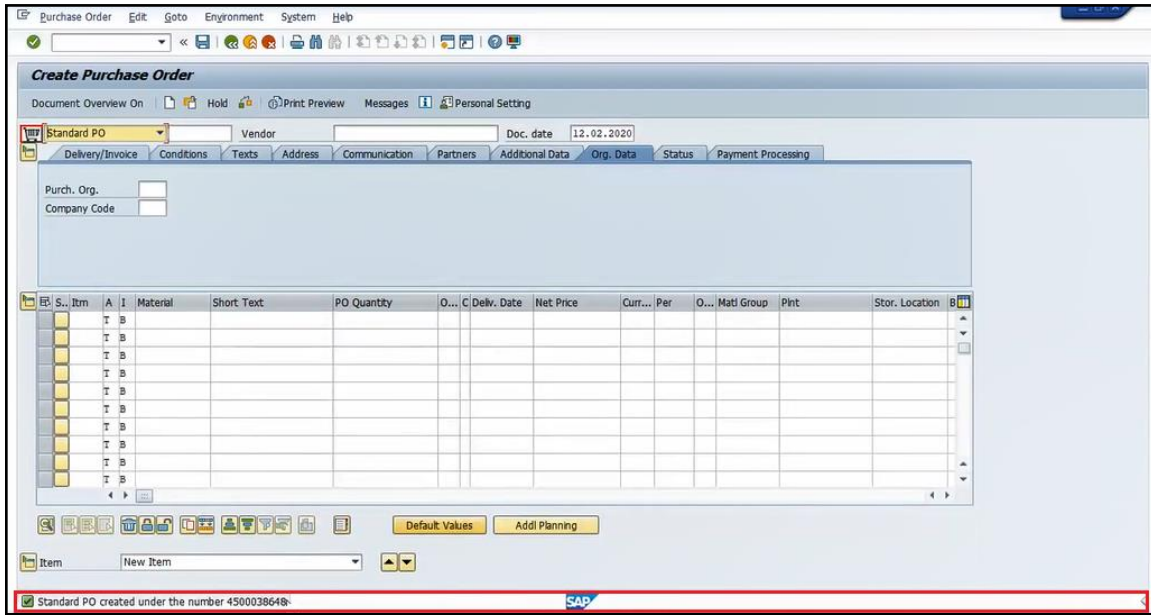
9. Click the  (**Capture a Field**) icon to capture the steps that you want to perform within the SAP application.

- Note:
- Launch SAP application before you start using the activity else the Capture a Field icon remains disabled.
- You must keep the relevant page of SAP application open before capturing the steps.




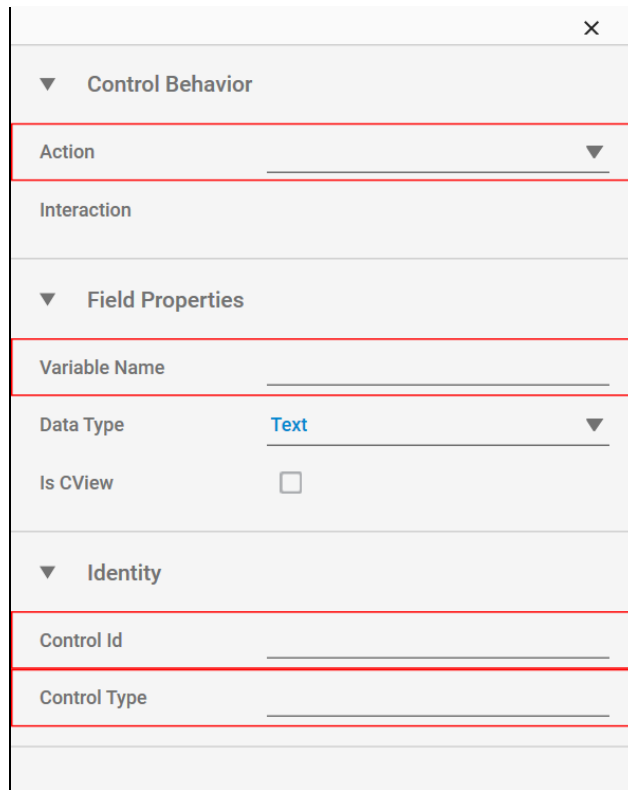
The SAP Logon pop message appears.

10. Click **OK** to close the message box and to configure the field.



The SAP application appears. The area where the cursor is present gets highlighted within red box.

11. Move the cursor to the area you want to capture and then click.
12. In the **GUI Field Extractor** activity, click the  (**Settings**) icon. The **Field Properties** panel appears. The fields that are mandatory to be filled are highlighted with red box.



13. Enter details of all the mandatory fields and other relevant fields as per your requirement.
14. Close the panel to save the details entered.

The fields are configured and the GUI Field Extractor activity is created.

Note: A single GUI Field Extractor activity captures a single field of a SAP application. You need to use multiple GUI Field Extractor activities to capture multiple field of a SAP application

GUI Control Behavior Field Properties

The properties of Control Behavior of GUI Field Extractor are listed in the following table and can be edited in the Control Behavior pane.

Property Name	Usage
Control Behavior	
Action	<p>Actions that can be performed on a web based application. Available options are:</p> <ul style="list-style-type: none"> ▪ Control Exists- Checks availability of the UI element on a SAP window. ▪ Extract Value- Extracts the value of the selected area. ▪ Extract Cell Value- Extracts the value of the selected cell. ▪ Get Row Count- Provides the total count of the row. ▪ Get Node Text - Provides the text value of the node.
Interaction	The corresponding interaction set against the selected action.
Field Properties	
Variable Name	User defined name of the variable that stores the captured value.
Data Type	The type of the data captured.
Identify	
Control Id	Identifier of the UI element. It is useful for locating elements that has a unique id associated with it.
Control Type	Type of the UI element selected.

GUI Field Extractor Properties

The properties of GUI Field Extractor activity are listed in the following table and can be edited in the Property grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity.</p> <p>If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the Flowchart designer. By default, the name is set as SAP Field Extractor . You can change the name as required.
SessionId	<p>A numeric property to indicate which session id of the SAP is used to perform the action. It is used in case of multiple session scenarios of SAP. For example, when a new session is opened on some action in first SAP session.</p> <p>Default value is set to 0.</p>

9.15.5 Attach Applications

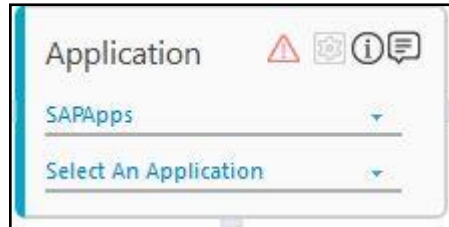
It allows you to bring the available SAP application in focus to perform the subsequent action in the SAP application.

This activity must be used inside an Application activity or the system displays an error.

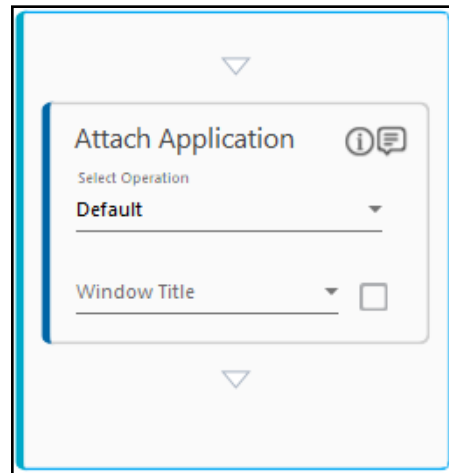
Note: When SAP application is configured in the Admin menu of Automation Studio, make sure the Is Startup Launch Required checkbox is cleared.

Using Attach Application Activity

1. Make sure that the [prerequisites](#) for using SAP Recorder activity are met.
2. In the **Canvas Tools** pane, click **Process Components** to expand the tool and view the associated activities.
3. Drag the **Application** activity and drop on to the **Flowchart** designer area.



4. From the **Application Type** list, select **SAPApps**. Application type that has at least one application added to it appears in the drop-down list.
5. In the **Select An Application** drop-down list, select the SAP application you want to perform automation on. You can add or configure a SAP application that is already added in Automation Studio or add a new application at this level.
6. Double click the **Application** activity to open the activity container.
7. In the **Canvas Tools** pane, click **SAP Thick Client** to expand the tool and view the associated activities.
8. Drag the **Attach Application** activity and drop in the **Application** activity.



9. In the **Select Operation** list, select the criteria to search for the required application in focus. Available options are:
 - **Default**- selects the window based on the default window name.
 - **Equals**- selects the window based on the exact match with the window name.
 - **Contains**- selects the window based on the text present in the window name.
 - **StartsWith**- selects the window based on the starting text of the window name.
 - **EndsWith**- selects the window based on the text with which the window name ends.
10. In the **Window Title** list, select a parameter that holds the title of the window of the application. You must define the parameter in the **Parameter** bar to use this option. Alternatively, in the **Window Title field**, enter the title of the SAP

application window and select the **Is Default Value** check box to set it as the default value of the title.

The Attach Application for the intended SAP application window is created. You can add an activity just below the Attach Application activity to perform the next action on the application in focus.

Attach Application Properties

The properties of an Attach Application activity are listed in the following table and can be edited in the Property grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the Flowchart designer. By default, the name is set as Attach Application . You can change the name as required.

9.16 SharePoint Activities

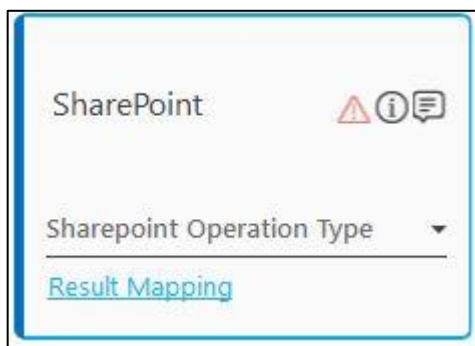
This activity allows you to automate the actions that can be performed on the Sharepoint site like upload, download, rename or delete the required file.

To access the Sharepoint site for performing automation, credential of the current logged in user of the system is used. Single Sign On (SSO) is not supported currently.

The Sharepoint 2010 version can be automated using the out-of-box activity. While you can use the [Web Mode](#) of the Application activity of Automation Studio to automate operations of the browser based Sharepoint site of any version.

9.16.1 Using Sharepoint Operations Activity

1. In the **Canvas Tools** pane, click **Sharepoint** to expand the tool and view the associated activities.
2. Drag the **Sharepoint** Operations activity and drop on to the **Flowchart** designer on the **Canvas**. The validation error symbol disappears when required inputs are provided.



3. In the **Sharepoint Operations Type** list, select the operation that you want to perform. Click any of the links below to know the details related to the supported Sharepoint operations:
 - [Upload](#)
 - [Download](#)
 - [Copy](#)
 - [Move](#)
 - [Delete](#)

9.16.2 Upload

The screenshot shows a configuration form for a SharePoint activity. The form has a title bar with 'SharePoint' and three icons (warning, info, help). Below the title bar are several dropdown menus:

- Sharepoint Operation Type: **Upload**
- Sharepoint Base Url: (empty)
- SharePoint Upload Folder Location: (empty)
- Source Folder Location: (empty)
- Source File Names: (empty)
- Comments: (empty)
- Force to Overwrite?: **False**

At the bottom of the form, there is a link labeled 'Result Mapping'.

1. In the **Sharepoint Base Url** list, select the parameter holding the URL of the Sharepoint site with the complete path of the folder in which you want to upload the file is located. For example, if the folder is located under **Mysite** you must provide the URL as, **http://Server/sites/Mysite**. You must define the parameter in the [Parameter](#) bar with the Sharepoint URL to make it available for the selection. This is a mandatory field.

Note: Do not add the pagename.aspx in the URL, else, the activity tries to search for the correct URL and logs it into its log file. However, the accuracy of finding the correct URL is not guaranteed and the process fails with an incorrect URL.

2. In the **SharePoint Upload Folder** Location list, select the parameter holding the path of the Sharepoint folder where you want to upload the file. For example, **/project/test/testing**. If the folder does not exist, it gets created at the mentioned path. The path mentioned must be accessible with the required permissions to create the folder. The permitted limit of the depth of sub folder that can be created inside a folder is defined in the **SharepointAutomationProps.xml** file and can be edited if required. You must define the parameter in the **Parameter** bar with the Sharepoint URL to make it available for the selection. This is a mandatory field.
3. In the **Source Folder Location** list, select the parameter holding the path of the folder along with the folder name from where the file must be uploaded. For example, **D:\Demo\SharePoint Folder**. You must define the parameter in the **Parameter** bar with the folder path to make it available for the selection. This is a mandatory field.
4. In the **Source File Names** list, select the parameter holding the file name along with the file extension or a pattern to select the required files. You can upload multiple files by providing the semi-colon (;) separated values. For example, **sample_doc.docx;*.pdf**. You must define the parameter in the **Parameter** bar with the file name or the pattern to make it available for the selection.
5. In the **Comments** list, select the parameter holding the checkin comments related to the uploaded files. You must define the parameter in the **Parameter** bar with the checkin comments to make it available for the selection.
6. In the **Force to Overwrite?** list, select True if you want to checkout and overwrite an existing file creating a new version of

the file. If **Flase** is selected, the files are not overwritten.

7. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

9.16.3 Download

1. In the **Sharepoint Base Url** list, select the parameter holding the URL of the Sharepoint site with the complete path of the folder from which you want to download the file. For example, if the folder is located under **Mysite** you must provide the URL as, **http://Server/sites/Mysite**. You must define the parameter in the [Parameter](#) bar with the Sharepoint URL to make it available for the selection. This is a mandatory field.

Note: Do not add the pagename.aspx in the URL, else, the activity tries to search for the correct URL and logs it into its log file. However, the accuracy is of finding the correct URL is not guaranteed and the process fails with an incorrect URL.

2. In the **SharePoint Source Folder Location** list, select the parameter holding the path of the Sharepoint folder from where you want to download the file. For example, **/project/test/testing**. If the folder does not exist, it gets created at the mentioned path. The path mentioned must be accessible with the required permissions to create the folder. The permitted limit of the depth of sub folder that can be created inside a folder is defined in the **SharepointAutomationProps.xml** file and can be edited if required. You must define the parameter in the **Parameter** bar with the Sharepoint URL to make it available for the selection. This is a mandatory field.
3. In the **Source** File Names list, select the parameter holding the file name along with the file extension or a pattern to select the required files. You can download multiple files by providing the semi-colon (;) separated values. For example, **sample_doc.docx;*.pdf**. You must define the parameter in the **Parameter** bar with the file name or the pattern to make it available for the selection.

4. In the **Download Location** list, select the parameter holding the path of the folder along with the folder name where the file must be downloaded. For example, **D:\Demo\SharePoint Folder**. You must define the parameter in the **Parameter** bar with the folder path to make it available for the selection. This is a mandatory field.
5. In the **Force to Overwrite?** list, select True if you want to overwrite an existing file. If **Flase** is selected, the files are not overwritten.
6. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

9.16.4 Copy/Move

1. In the **Sharepoint Base Url** list, select the parameter holding the URL of the Sharepoint site with the complete path of the folder from where you want to copy or move the file. For example, if the folder is located under **Mysite** you must provide the URL as, **http://Server/sites/Mysite**. You must define the parameter in the [Parameter](#) bar with the Sharepoint URL to make it available for the selection. This is a mandatory field.

Note: Do not add the `pagename.aspx` in the URL, else, the activity tries to search for the correct URL and logs it into its log file. However, the accuracy of finding the correct URL is not guaranteed and the process fails with an incorrect URL.

2. In the **SharePoint Folder Location** list, select the parameter holding the path of the Sharepoint folder from where you want to copy or move the file. For example, **/project/test/testing**. You must define the parameter in the **Parameter** bar with the Sharepoint URL to make it available for the selection. This is a mandatory field.
3. In the **Source File Names** list, select the parameter holding the file name along with the file extension or a pattern to select the required files. You can copy or move multiple files by providing the semi-colon (;) separated values. For example,

sample_doc.docx;*.pdf. You must define the parameter in the **Parameter** bar with the file name or the pattern to make it available for the selection. If you leave this field empty all the files present in the specified folders are copied or moved as per the selected operation.

- In the **Sharepoint Destination Folder** list, select the parameter holding the path of the Sharepoint folder where you want to copy or move the file. For example, **http://Sharepoint/sites/Mysite**. You must define the parameter in the Parameter bar with the folder path to make it available for the selection. This is a mandatory field.
- Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

9.16.5 Delete

- In the **Sharepoint** Base Url list, select the parameter holding the URL of the Sharepoint site with the complete path of the folder from where you want to delete the file. For example, if the folder is located under **Mysite** you must provide the URL as, **http://Server/sites/Mysite**. You must define the parameter in the [Parameter](#) bar with the Sharepoint URL to make it available for the selection. This is a mandatory field.

Note: Do not add the pagename.aspx in the URL, else, the activity tries to search for the correct URL and logs it into its log file. However, the accuracy of finding the correct URL is not guaranteed and the process fails with an incorrect URL.

- In the **SharePoint Folder Location** list, select the parameter holding the path of the Sharepoint folder from where you want to delete the file. For example, **/project/test/testing**. This is a mandatory field.
- In the **Source File Names** list, select the parameter holding the file name along with the file extension or a pattern to select the required files. You can delete multiple files by providing the semi-colon (;) separated values. For example, **sample_doc.docx;*.pdf**. You must define the parameter in the **Parameter** bar with the file name or the pattern to make it available for the selection. If you leave this field empty all the files present in the specified folders are deleted.

4. Click the **Result Mapping** link. The **Output Mapping** list appears. Select the available parameter to map the status of the operation performed. Both Boolean and String type is supported. You must pre-define the parameter to make it available for the selection. If the operation is successful, it returns true and if the operation fails it returns false. It can be used as an input for the next action that you want to perform in the automation process workflow, depending upon the success or failure of the file operation performed.

The respective Sharepoint operations are configured.

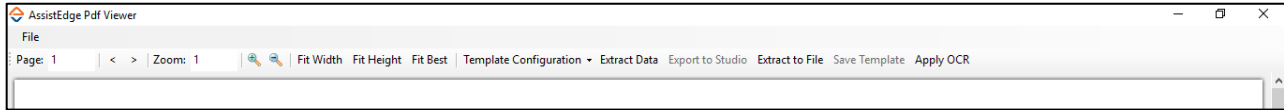
9.16.6 SharePoint Operations Properties

The properties of Sharepoint Operations activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes, the application ignores any error while executing the activity. If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No.
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the Flowchart designer. By default, the name is set as Sharepoint Operations . You can change the name as required.

9.17 PDF Template Creator

It allows you to configure a PDF template used to extract a block of data and tabular data from a searchable PDF file. Additionally, you can use the OCR engine available with this tool to extract data from a scanned PDF file.



The following functionality are available in the menu bar of AssistEdge Pdf Viewer screen of the PDF Template Creator canvas tool.

- **Page** - Enter the page number in the field to go to the required page. Use the > (**Next Page**) and the < (**Previous Page**) arrows to navigate to the next or previous page of the PDF file.
- **Zoom** - Zooms In or Out the PDF file for a better view of the content available in the document.
- **Fit Width** - Fits the PDF file to the window width.
- **Fit Height** - Fits the PDF file to the window height.
- **Fit Best** - Fits the PDF file to view the entire content of the page.
- **Template Configuration** - Allows you to configure a pdf template for data extraction. Available options are:
 - **Create New Template** - Creates a new PDF template.
 - **Load Existing Template** - Opens an already existing PDF template.

Note: Ensure the PDF format used for extraction is the same as the format used for template configuration. For example, if the template configuration happens using a PDF created with an inverted axis, the PDF used for extraction must have been created using an inverted axis as well; else, extraction fails.

- **Extract Data** - Displays or saves the extracted data as per the type of data extracted. See Extract Data and Export Option section to know the details.
- **Export to Studio** - Exports the configured template to the Studio menu. See Extract Data and Export Option section to know the details.
- **Extract to File** - Exports the extracted data into a file. See Extract Data and Export Option section to know the details.
- **Save Template** - Saves the configured template at %localappdata% > EdgeVerve > AutomationStudio > ProtonFiles > PdfRepository folder.
- **Apply OCR** - Runs the OCR engine on the scanned PDF file for data extraction. See Using OCR for Extracting Data section to know the related details.

Prerequisite

Microsoft Visual C++ 2015 version X86 must be installed on the system.

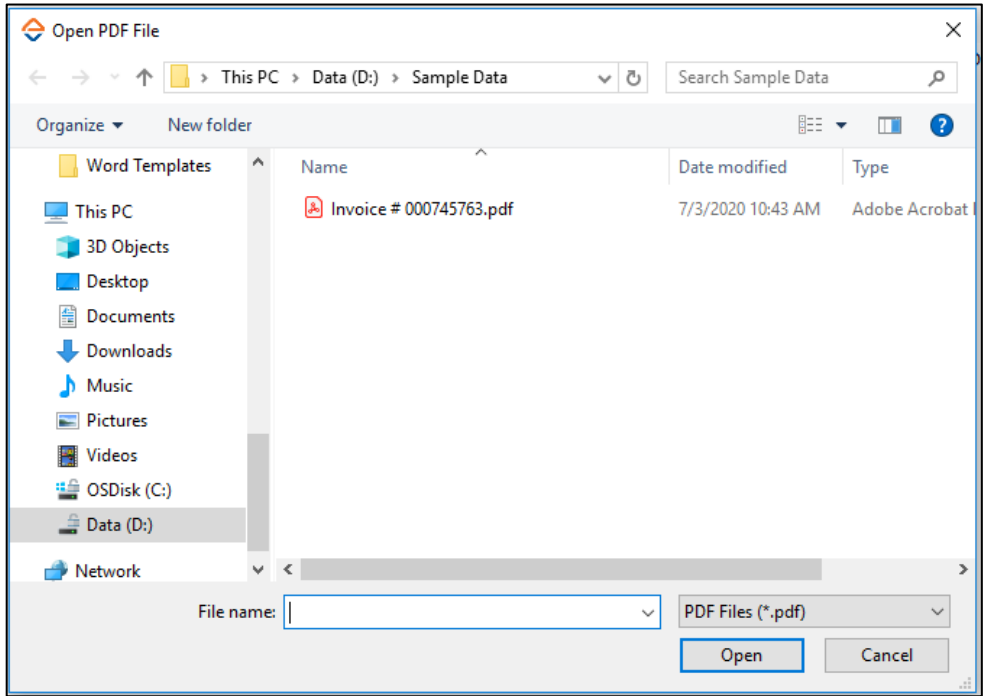
Using PDF Template Creator Activity

Depending on whether the PDF is searchable or scanned, the steps to create the PDF template differs. Click any of the links below to know the steps in detail:

- [Extracting Data From a Searchable PDF](#)
- [Extraction of Data From a Scanned PDF](#)

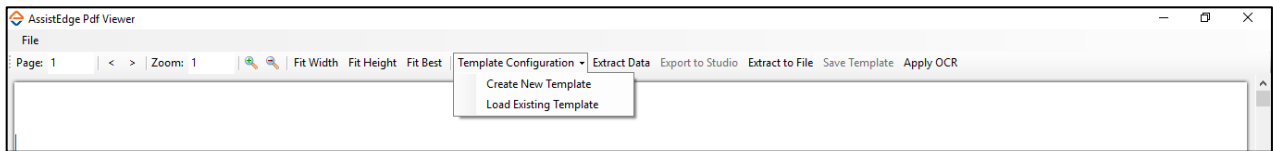
Extracting Data From a Searchable PDF

1. In the **Canvas Tools** pane, click **PDF Template Creator** to launch the tool. The **Open PDF File** dialog box appears.



2. Browse for the required file and click **Open**. The **AssistEdge PDF Viewer** along with the selected PDF file appears.
3. Click **Template Configuration**, and then click **Create New Template** to create a new template or click **Load existing template** to view, edit or use and an existing template. A configuration pane appears on the right side of the **AssistEdge PDF Viewer**. By default, the **Configuration Type** selected is **Reference Configuration**.

Note: You can use the existing PDF template with PDF Extractor activity as well. See PDF Extractor for related details.



Click any of the links below to know the steps in details for the selected configuration type:

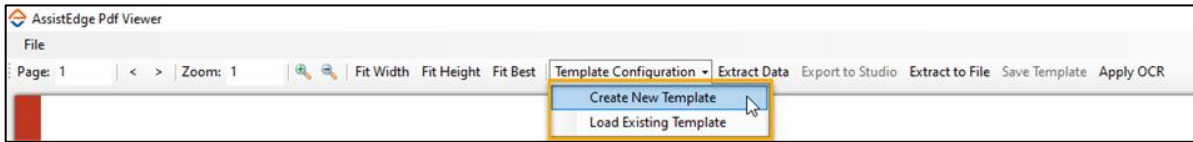
- [Reference Configuration](#)
- [Field Configuration](#)

Reference Configuration

You can use the PDF Template Creator to launch the pdf tool and create a process for automated PDF extraction.

To create a new template,

4. Click the **PDF Template Creator** activity in Automation Studio.
5. Select the pdf file for which you want to automate the process.
6. Click Template configuration.



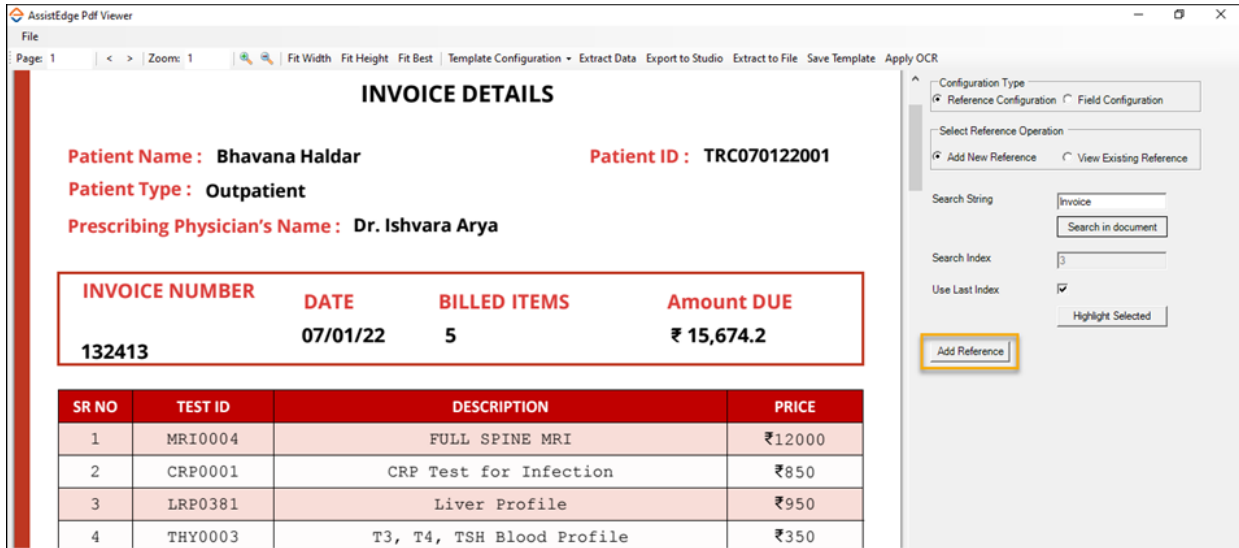
- a. Menu Items:
 - i. **Template Configuration:** To configure the template for PDF automation
 1. **Create New Template:** To create a new template
 2. **Load Existing Template:** To use existing template
 - ii. **Save Template:** Save the created template in the proprietary format
 - iii. **Export to Studio:** Export, the template to the Studio, to create the pdf extractor activity.
 - iv. **Configuration Type:** Choose to configure one or many references or one or many fields (tables) using the two options
 - v. **Select Reference Operation:** Any new reference or existing reference/ field is accessible by choosing one of the two options.
7. Add new references or view existing references.
8. Click **Export to Studio** to export the template into the Automation Studio.

Add New Reference

1. Select Add New Reference and search a string for which you want to automate the extraction.
2. Enter the required index number in the Search Index field.
3. Specifying index number is useful when there are multiple references of a string in the PDF file.
4. Select the Use Last Index to search a string from last index that is last occurrence of the reference string in the PDF file.

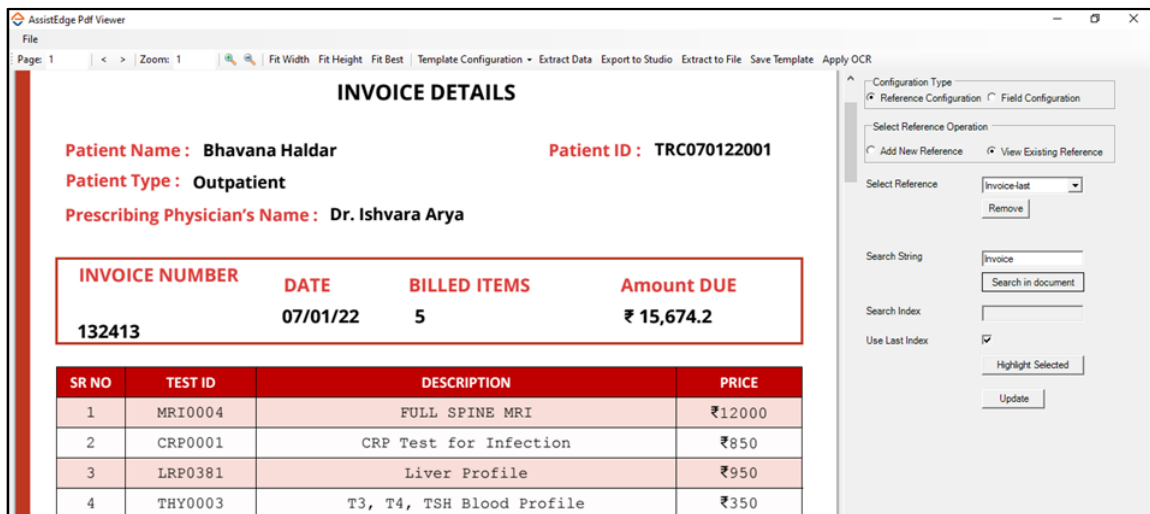
Note: Selecting the Use Last Index option disables the option of specifying index number in the Search Index field.

5. Click **Highlight Selected** to highlight the string based on selected option that is either **Search Index** or **Use Last Index**.
6. Click Add Reference.



View Existing Reference

7. Click **View Existing Reference** to view all the existing references.
You can also update or remove existing references.
8. Click **Remove** to delete the existing reference.
9. Click **Search in document** to search a string.
10. Specify index number in the **Search Index** field.
11. Select the **Use Last Index** option to search a string from last index that is last page of the PDF file.
12. Click **Highlight Selected** to highlight the string based on selected option that is either **Search Index** or **Use Last Index**.
13. Click **Update** to update existing reference.



Field Configuration

It lets you create or modify fields to extract a block of data or tabular data from the PDF file.

Fields are a set of configurations that include reference points, rectangular boundaries or even excluded areas to effectively extract the required data. For precision, you can provide related coordinates of the area from where the data must be extracted.

The screenshot shows the 'Field Configuration' dialog box. At the top, under 'Configuration Type', 'Field Configuration' is selected. Below this, 'Select Field Operation' has 'Add New Field' selected. There are three tabs: 'Selection Area', 'Excluded Areas', and 'Column Configuration'. The 'Selection Area' tab is active. It contains a 'Name' text box, three checkboxes for 'Is Tabular Field', 'Remove Newline Chars', and 'Output new file per page', and a dropdown menu for 'Area Selection Type' set to 'Fixed Rectangle'. Below these is a 'Draw Area' button. Further down are two sections for corner coordinates: 'Bottom Left Corner' and 'Top Right Corner', each with 'X', 'Y', and 'Page Number' input fields. At the bottom of the dialog are 'Add Field' and 'Highlight Current Field' buttons.

1. In the **Configuration Type** group, select **Field Configuration** to configure fields in the PDF template for data extraction.
2. In the **Select Field Operation** group, select the type of field you want to add. You can choose the **Add New** Field option to add a new field or choose the **View Existing Fields** option to edit the existing fields. You can configure three different areas to extract block of data or tabular data. Click any of the links below to know the detailed steps:
 - [Selection Area](#)
 - [Excluded Area](#)
 - [Column Configuration](#)

Note: The instructions given below are valid for adding a new field or editing an existing field. However, if you want to edit an existing field, upon selecting View Existing Fields option, the Select drop down appears with the list of configured fields. You can select the required field which you want to edit. The configuration details are auto populated in the fields and can be edited, if required.

In case you want to remove an existing field, select the required field, and then click Remove Field.

Selection Area

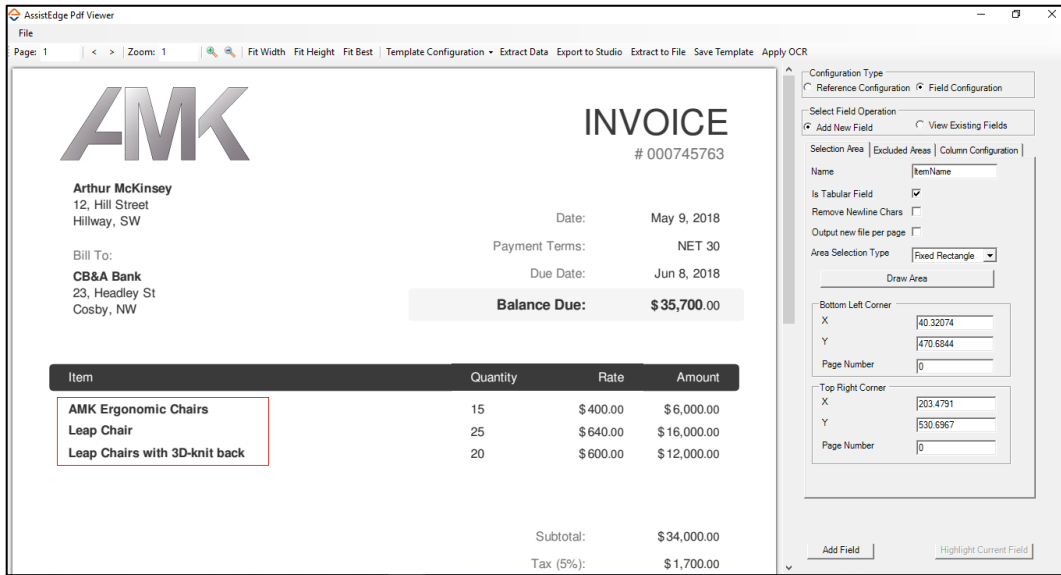
It lets you mark the area for data extraction.

3. In the **Name** field, enter the desired name for the data to be marked.
4. Select **Is Tabular Field** check box if the data to be marked is tabular data.
5. Select **Remove Newline Chars** check box if there is a need to remove any newline characters while extracting the data. Removal of the newline characters reflects in the output excel and the data is extracted as a single line data.
6. Select **Output new file** per page check box if the output for data extracted from each page is preferred as a new file.
7. In the **Area Selection Type** list, select the type of boundary, from static or dynamic corners, to mark the area for selection. Available options are- **Fixed Rectangle, One Corner, Two Corners and Three Corners.**

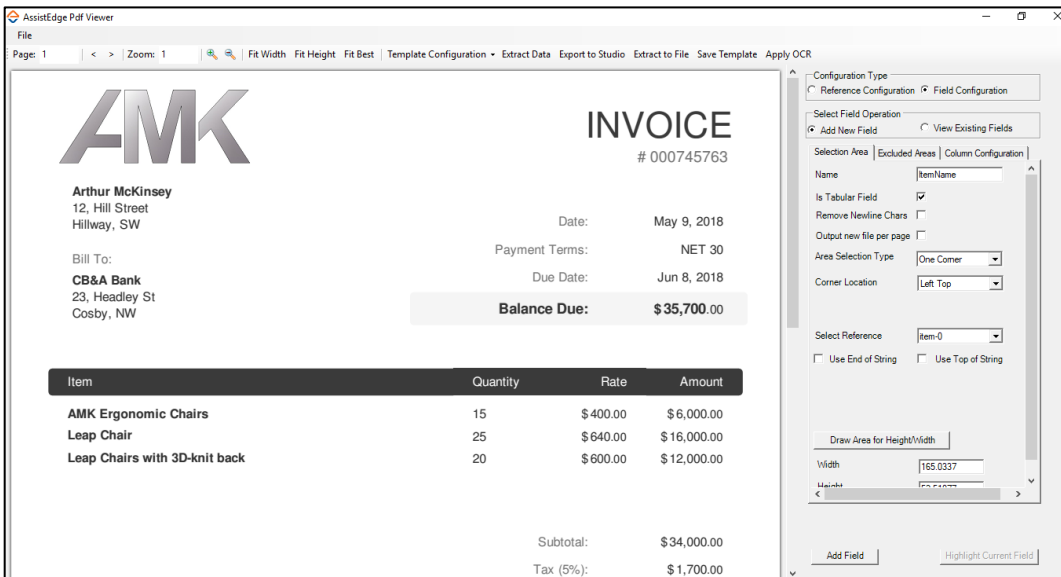
Note:

- **Fixed Rectangle** grants static position while the other three are dynamic position.
- Except for the **Fixed Rectangle** option, you must first define the **Reference Configuration**. This allows you to configure the offset points to let the tool know the rectangular boundaries for data extraction.
- **Three Corners** option is the ideal way to extract the tabular data.

- a. If Area Selection Type is Fixed Rectangle:
 - i. Click **Draw Area** and select the area that you want to mark for data extraction. The area gets highlighted within a box. The values of the **X** (horizontal) and **Y** (vertical) axis of the **Bottom Left Corner** and **Top Right Corner** groups get auto filled as per the area selected. Additionally, the **Page Number** field is auto filled. You can edit the auto filled values, if required.



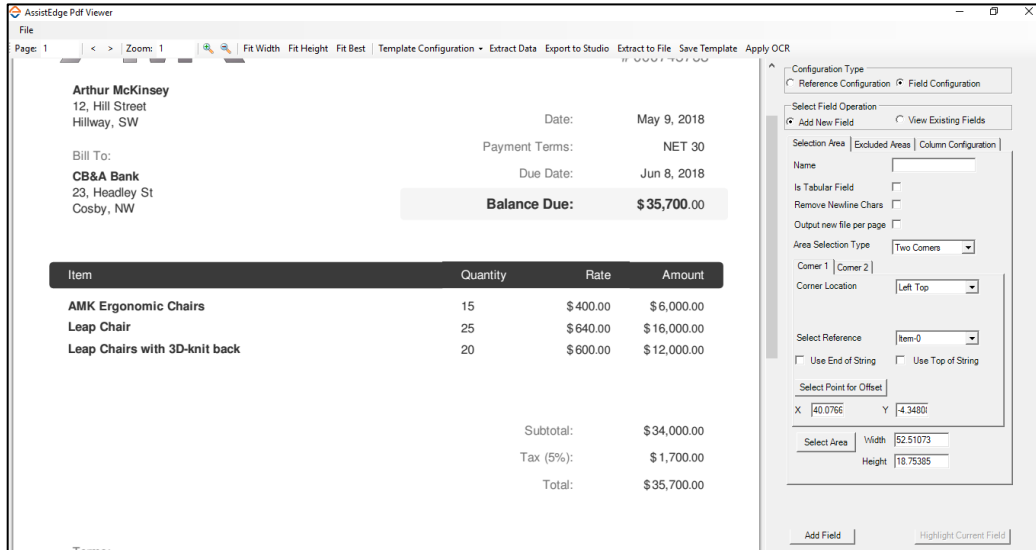
- ii. Click **Add Field** to confirm the configuration and add the field.
 - 1. If **Area Selection Type** is **One Corner**, you can manually define offset of the one corner to form a rectangular boundary along with the reference points:



- iii. In the **Corner Location** list, define the location of the field boundary. Available options are - **Left Bottom, Left Top, Right Bottom** and **Right Top**. By default, **Left Top** is selected.
- iv. In the **Select Reference** list, select the required reference string.
- v. Select the **Use End of String** check box to either start or stop data extraction when the end of the string is encountered.
- vi. Select the **Use Top of String** check box to indicate if the reference string is part of the data extracted.
- vii. Click Draw **Area for Height/Width**, and then manually draw the height and width of the boundary. The area gets highlighted with a red box. The **Width** and **Height** fields are auto-filled. You can edit the auto-filled values if required.

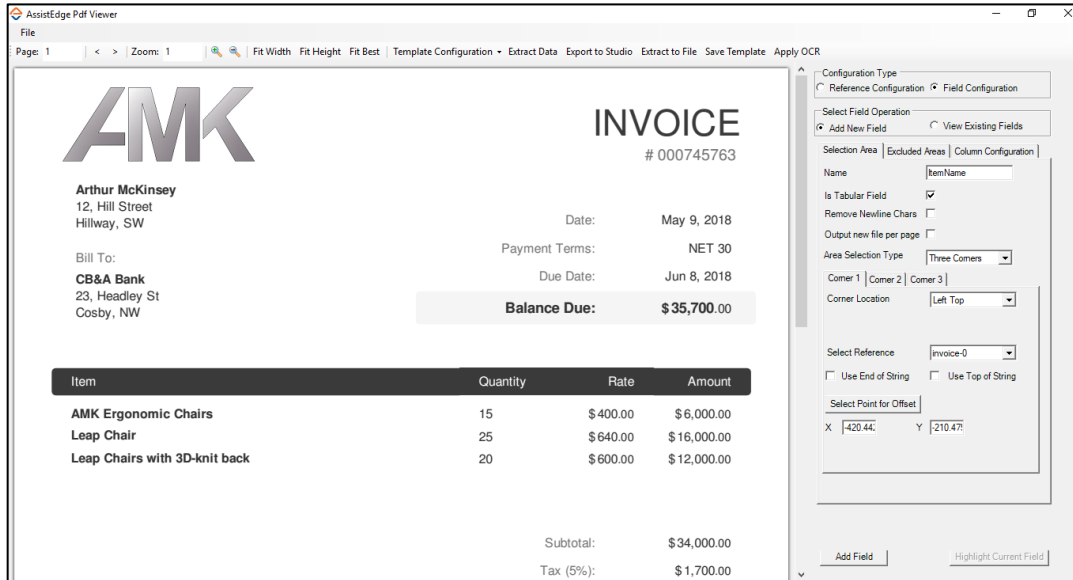
viii. Click **Add Field** to confirm and add the field.

1. If **Area Selection Type** is **Two Corner**, you can manually define offset of the two corners to form a rectangular boundary along with the reference points:



- In the **Corner 1** tab, define the location of field boundary in the **Corner Location** list. Available options are - **Left Bottom, Left Top, Right Bottom, and Right Top**. By default, **Left Top** is selected.
- In the **Select Reference** list, select the required reference string.
- Select the **Use End of String** check box to either start or stop data extraction when the end of the string is encountered.
- Select the **Use Top of String** check box to indicate if the reference string is part of the data extracted.
- Click **Select Point for Offset** and then choose a point from where the boundary of the first corner can be drawn. This auto-fills the values of the offset of the selected corner. Alternatively, you can manually enter the values of the **X** and **Y**-axis to define the offset of the selected corner. Zoom in the actual pdf file and choose the relevant points to fill the **X** and **Y**-axis fields.
- Click **Select Area**, and then manually draw the height and width of the boundary. The area gets highlighted with a red box. The **Width** and **Height** fields are auto filled. You can edit the auto filled values if required.
- Repeat step 1 through step 6 to configure **Corner 2** tab.

- xvi. Click **Add Field** to confirm and add the field.
1. If **Area Selection Type** is **Three Corner**, you can manually define offset of the three corners to form a rectangular boundary along with the reference points:



- xvii. In the **Corner 1** tab, define the location of field boundary in the **Corner Location** list. Available options are - **Left Bottom, Left Top, Right Bottom, and Right Top**. By default, **Left Top** is selected.
- xviii. In the **Select Reference** list, select an existing reference string.
- xix. Select the **Use End of String** check box to either start or stop data extraction when the end of the string is encountered.
- xx. Select the **Use Top of String** check box to indicate if the reference string is part of the data extracted.
- xxi. Click **Select Point for Offset** and then choose a point to define the boundary of the first corner. This auto-fills the values of the offset of the selected corner. Alternatively, you can manually enter the values of the **X** and **Y**-axis to define the offset of the selected corner. Zoom in the actual pdf file and choose the relevant points to fill the **X** and **Y**-axis fields.
- xxii. Repeat step 1 through step 5 to configure **Corner 2** and **Corner 3** tabs respectively.

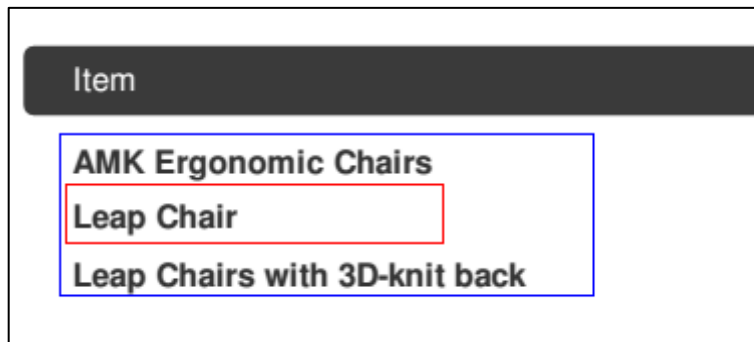
Once the area is selected, you can mark required area to exclude from the data extraction.

Excluded Area

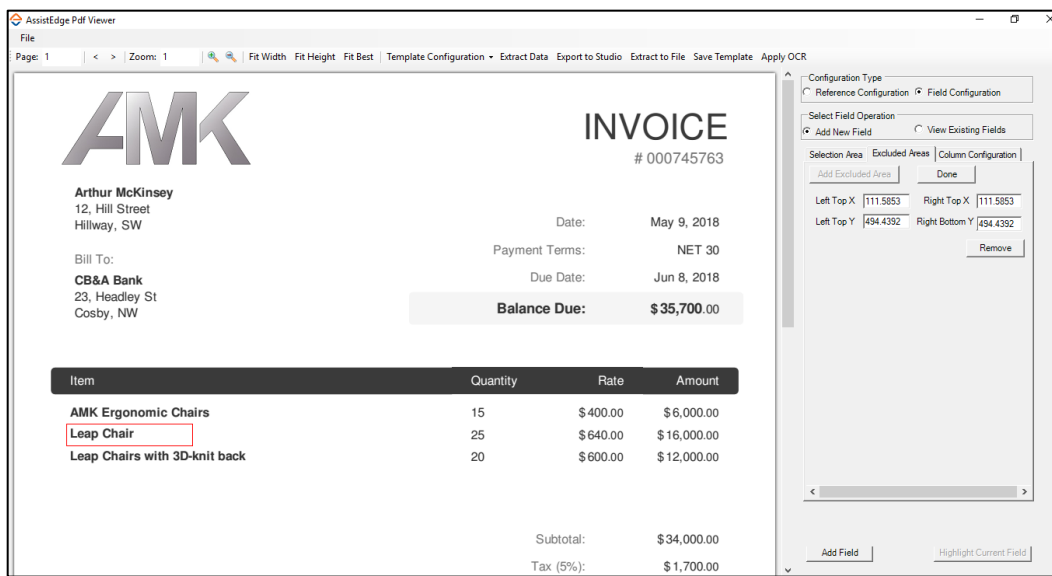
In this section you can configure the area that you want to exclude from data extraction.

If a table spans over multiple pages, configure the area spanning from the footer of current page to the header of next page, to be excluded from the extraction.

You can mark other areas to enable excluding a particular section from extraction as depicted in the below screen shot. The blue box highlights the area marked for data extraction while the red box highlights the area that must be excluded from extraction.



1. Click the **Excluded Area** tab.
2. Click the **Add Excluded Area** button then select the area on the PDF file to auto fill the **Left Top X, Left Top Y, Right Top X,** and **Right Top Y** coordinates of the area to exclude. You can edit the values if required.

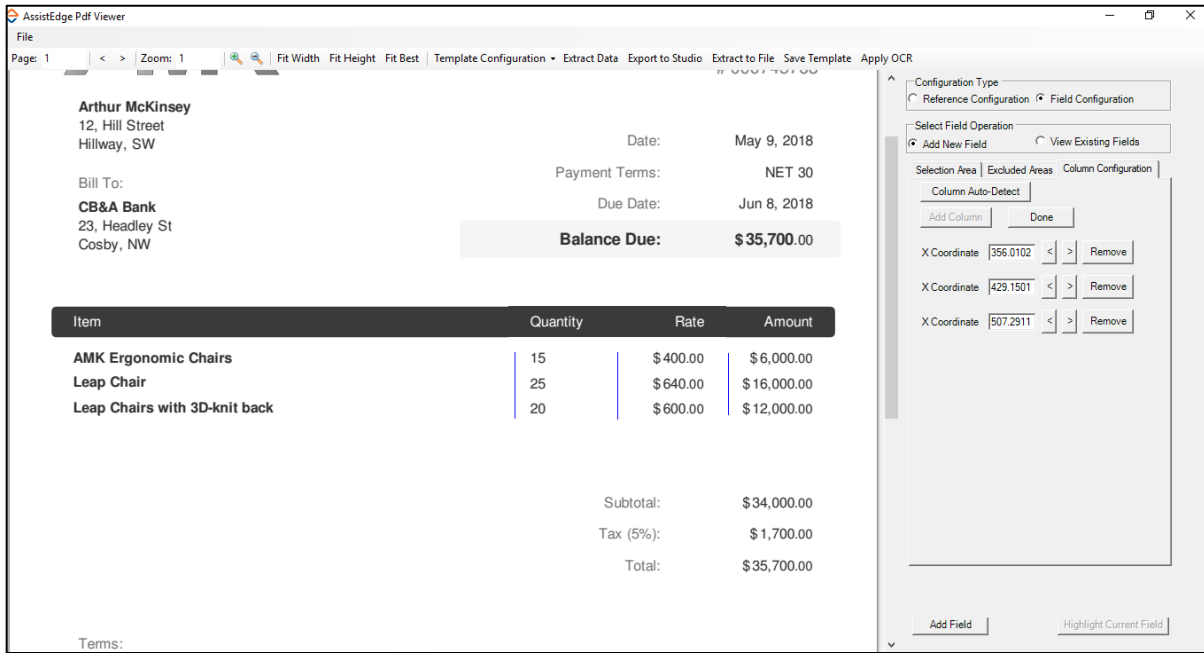


3. Click **Done**.
4. Repeat step I to iii to mark other areas to exclude them from the data extraction. The data is marked for exclusion from extraction.

Column Configuration

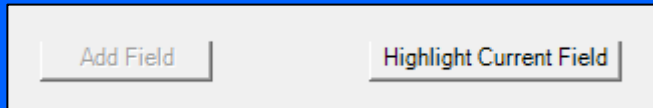
In this section you can define the column format in which the extracted tabular data must be saved in the .CSV file after extraction. If you do not mark the columns of the tabular data, the extracted data gets saved in a single row in the .CSV file.

1. Click the Column Configuration tab.
2. Click **Column Auto-Detect** to automatically identify the columns. Alternatively, click **Add Column** to manually mark the columns. Draw vertical lines between the tabular fields that you want to mark for column configuration.



3. Click **Done** to save the defined column configuration. The column configuration is done.

Note: You can view the marked area (added or edited) upon selecting View Existing Fields option. The Select drop-down field appears with the list of configured fields. You can select the required field which you want to view or edit.



Click Highlight Current Field button. This highlights the marked or updated area for data extraction. This button is not available while creating or modifying the fields.

- 4. Click **Add Field** to confirm and add the configured field. This button remains disabled if you want to edit an existing field.
- 5. Click **Save Template**. The **Save As** dialog box appears. The dialog box does not appear if you are saving an existing field.
- 6. Provide the desired file name, and then click **Save**.
The PDF template gets saved at the %localappdata% > EdgeVerve> AutomationStudio > ProtonFiles> PdfRepository folder.

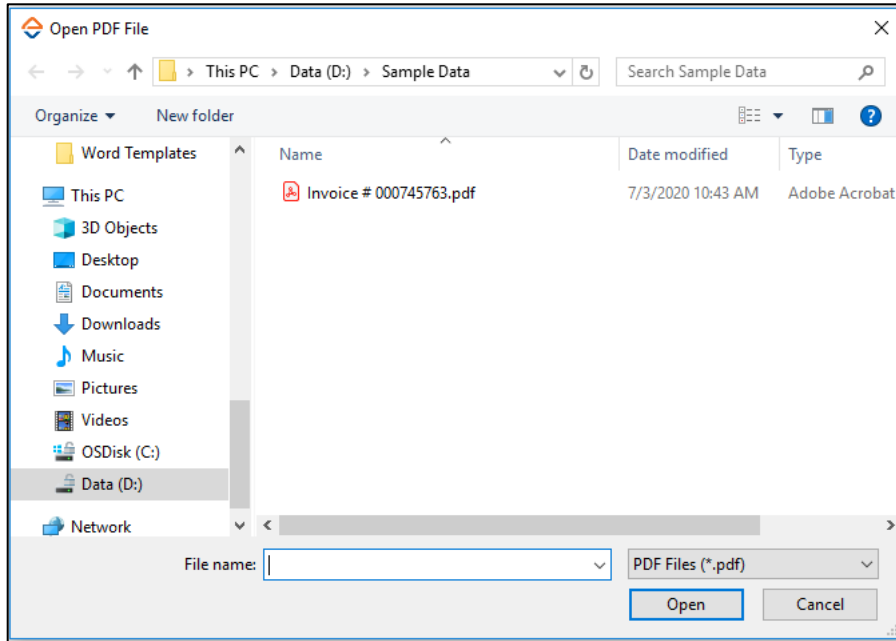
Extracting Data From a Scanned PDF

You can use the OCR (Optical Character Recognition) technology, available in Automation Studio, to automatically extract text and images from a scanned PDF file.

Additionally, you can use the Apply OCR functionality manually, available in the AssistEdge Pdf Viewer window, to extract scanned data that may be available in the searchable PDF file.

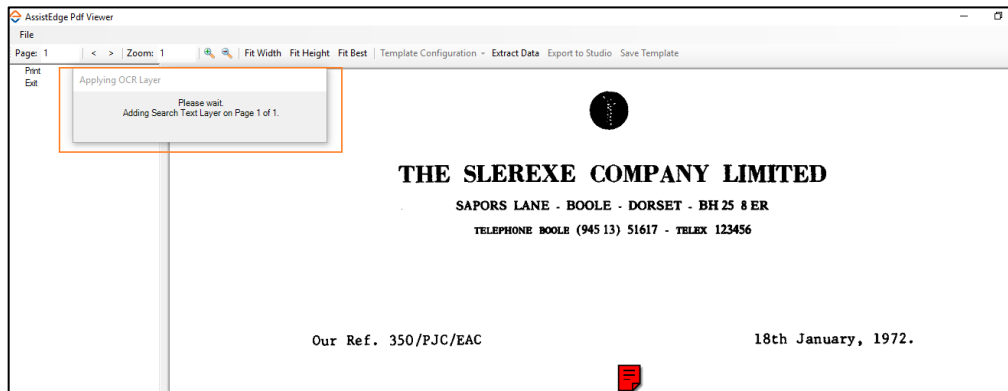
To extract data using OCR:

1. In the **Canvas Tools** pane, click **PDF Template Creator** to launch the tool. The **Open PDF File** dialog box appears.

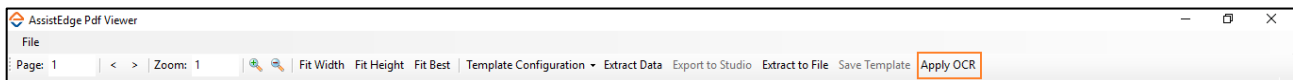


2. Browse for the required file and click **Open**. The **AssistEdge PDF Viewer** along with the selected PDF file appears.

- If it is a scanned PDF file, Automation Studio automatically runs the OCR engine and scans the document.



- If it is a searchable PDF file with some scanned data, click **Apply OCR**. Automation Studio runs the OCR engine and scans the document.



3. Click **Template Configuration** and then click **New Template**. A configuration pane appears on the right side of the **AssistEdge PDF Viewer**. By default, **Configuration Type** selected is **Reference Configuration**. Click any of the links below to know the steps in details for the selected configuration type

- [Reference Configuration](#)
- [Field Configuration](#)


Reference Configuration

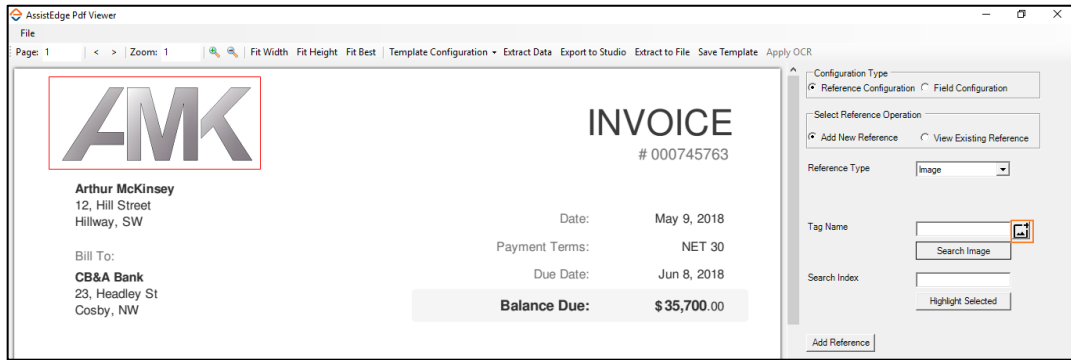
It lets you configure a reference point for data extraction. With the help of reference points, you can indicate the start and the end point of the area within which the data must be extracted. You can configure a string or image as a reference point. You can update or remove the references from the PDF template.

1. In the **Reference Type** list, select Text to extract text or select Image to extract image from the scanned document.

Note: The instructions given below are valid for adding a new reference or editing an existing reference. However, if you want to edit an existing reference, upon selecting View Existing References option, the Select Reference drop-down field appears with the list of configured references. You can select the required reference that you want to edit. The configuration details are auto populated in the fields and can be edited if required. If there is no reference available, by default, -0 appears in Select Reference list.

In case you want to remove an existing field, select the required field, and then click Remove.

- a. If you want to extract text, refer the steps mentioned in Reference Configuration section to know the details.
- b. If you want to use the image as the reference, follow the below mentioned steps:
 - i. Click the  icon, and then select the image to mark for extraction.



- c. In the **Tag Name** field, enter the image name and then click **Search Image**. This highlights the images with the entered name.
- d. In the **Search Index** text box, enter the occurrence of the image and then click **Highlight Selected**. This keeps the blue box highlight at the specified occurrence of the image. If the image entered, occurs at multiple instances, you need to specify which instance of the occurrence of the string you want to mark. By default, the first occurrence of the string is considered for marking.
- e. Click **Add Reference** to add the marked image as reference. This button is not available if you are editing an existing reference.
- f. Click **Update** to update the edited reference. This button is not available if you are adding a new reference.

The image is added or updated as a reference for data extraction.

Field Configuration

It lets you create or modify fields to extract a block of data or tabular data from the PDF file.

Fields are a set of configurations that include reference points, rectangular boundaries or even excluded areas to effectively extract the required data. For precision, you can provide related coordinates of the area from where the data must be extracted. See Field Configuration section to know the details to add or edit the field configuration.

In a scanned PDF file, you can apply filters and process the scanned PDF to enhance the quality of the extracted data.

- 2. Select **View Existing Fields** option. The **Select** drop down appears with the list of configured fields.

Configuration Type
 Reference Configuration Field Configuration

Select Field Operation
 Add New Field View Existing Fields

Select

Selection Area | Excluded Areas | Column Configuration |

Name

Is Tabular Field

Remove Newline Chars

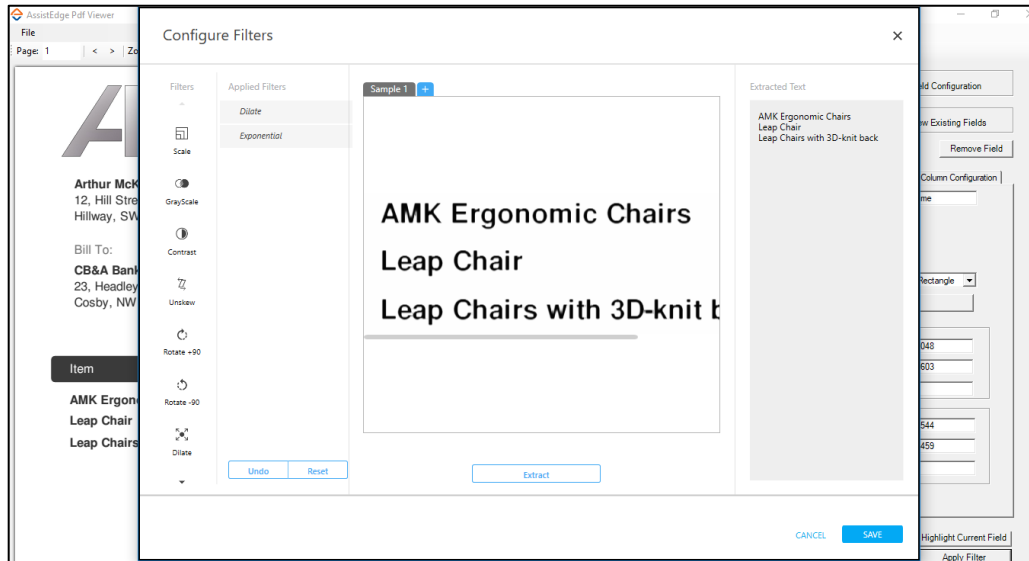
Output new file per page

Area Selection Type

Bottom Left Corner
X
Y
Page Number

Top Right Corner
X
Y
Page Number

- 3. You can select the required field, and then click **Apply Filter**. The **Configure Filters** dialog box appears with the selected area and the available filters.



4. Choose the required filter. It starts appearing in the **Applied Filters** pane. You can use **Undo** to remove the last applied filter and **Reset** to remove all the filters.
5. Click **Extract** to view a sample of the data that would be extracted.
6. Click **Save** to save the configuration. You are directed back to the **AssistEdge Pdf Viewer**.
7. Click **Save Template**. The **Save As** dialog box appears. The dialog box does not appear if you are editing an existing configuration.
8. Provide the desired file name and click **Save**.

The PDF template gets saved at the %localappdata% > EdgeVerve > AutomationStudio > ProtonFiles > PdfRepository folder.

See Implementation Tips section to know how a configured template can be edited for performing advanced operations like extracting tabular data that is available over multiple pages.

Note:

- Template defined for scanned PDF files cannot be used for searchable PDF files.
- Scanned documents with handwritten data is not supported.
- Extraction algorithm works on how the text is indexed in a file. However, in a few PDF files, what appears to be a part of one string may not be its index order. For example, consider the sample data below:
- Shipping Methods:
- FedEx 706797210999 FedEx 706797211002
- The data is indexed as, Shipping Methods: FedEx 706797210999 706797211002, and not as, Shipping Methods: FedEx 706797210999 FedEx 706797211002, hence the index value must be entered correctly for the relevant data to be extracted.

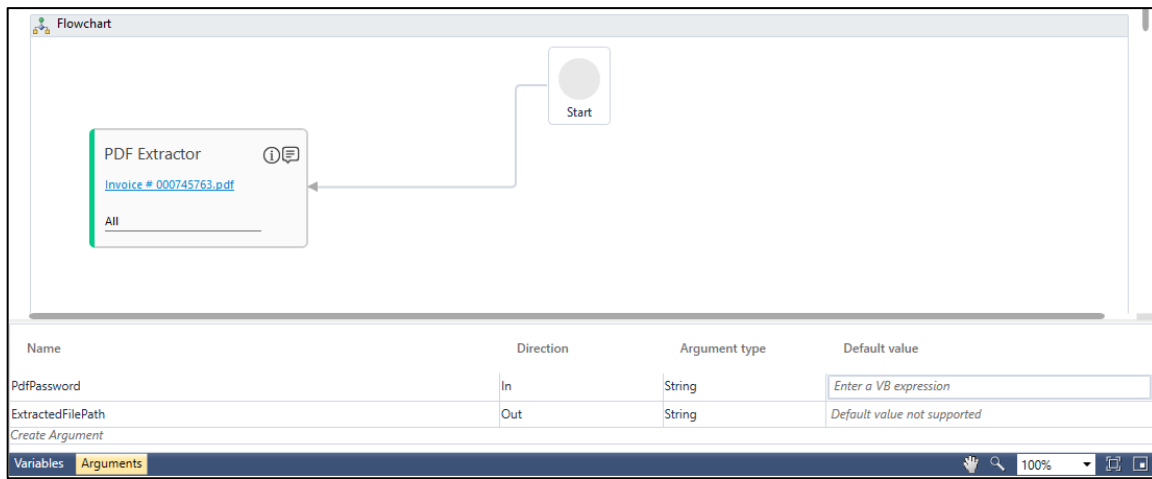
Extract Data and Export Options

You must first save the template to extract the data. You can extract data from the configured template by using Extract Data option.

Using the Extract Data option, you can view a sample of extracted block of data in the AssistEdhe PDF Viewer window itself, however, the tabular data gets saved in a .CSV excel file. By default, the .CSV file gets saved in the %localappdata% > EdgeVerve> AutomationStudio folder.

The Extract to File options exports the extracted data in a file and saves the file location in the ExtractedFilePath argument that gets created automatically with the help of the PDF Extractor activity. The PDF Extractor activity is created and the PdfPassword and ExtractedFilePath parameters gets automatically created in the Arguments pane of the Parameter bar.

This option is helpful if you want to convert the entire document or certain set of pages into text. See PDF Extractor activity to know how to use it.



Use the Export to Studio option to export the PDF template to the Studio menu. It creates the PDF Extractor activity with the PageReferences, PdfPassword and PageCount parameters getting automatically created in the Arguments pane. See PDF Extractor activity to know how to use it.

The screenshot shows a flowchart with a 'Start' node connected to a 'PDF Extractor' activity. The activity has two input arguments: 'Invoice # 000745763.pdf' and 'PDFTestTemplate.dtmp'. Below the flowchart is a table of arguments for the PDF Extractor activity.

Name	Direction	Argument type	Default value
ItemName	Out	String	Default value not supported
PageReferences	Out	DataTable	Default value not supported
PdfPassword	In	String	Enter a VB expression
PageCount	Out	Int32	Default value not supported
Create Argument			

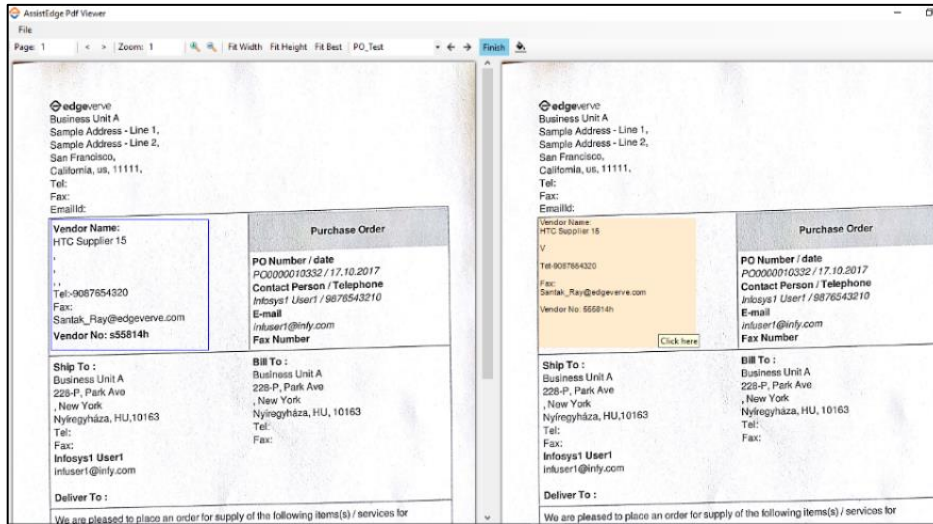
At the bottom of the interface, there are tabs for 'Variables' and 'Arguments', and a zoom level of 100%.

You can use the In argument, PdfPassword, to pass the password of the excel file to make it password protected. The Out arguments, PageReferences and PageCount can be used to store. Another default Out argument, ExtractedFilePath, stores the location where the file gets stored after Export to File option is used.

Assisted View

You can compare the data extracted from a scanned PDF file with the original document in a comparison view in Automation Studio. Once the PDF template is exported to Automation Studio, you can select the Compare Result check box available in the Properties grid of the PDF Extractor activity.

While performing Test Run in Automation Studio, a comparative view of the data extracted from the scanned document and the original document is displayed. This enables you to validate the data and make the required changes in the template, if any.



Confidence Score

If you are using a scanned document for data extraction, a default argument, ConfidenceScore, gets created in the Argument pane after the template is exported to the Studio menu. It is used to get the confidence value of all the extracted fields. It can be useful in deciding whether the extracted fields need manual correction. It compares the original data and the scanned data to provide the confidence score.

You can add a WriteLine activity and enter `Convert.ToString(ConfidenceScore.Rows(0)("Field Name"))` in the Text field to view the score in Automation Studio.

Name	Direction	Argument type	Default value
f3	Out	String	Default value not supported
table1	Out	String	Default value not supported
PageReferences	Out	DataTable	Default value not supported
PdfPassword	In	String	Enter a VB expression
PageCount	Out	Int32	Default value not supported
ConfidenceScore	Out	DataTable	Default value not supported

Implementation Tips

- Configuration scenarios for workaround during development:
- The saved PDF template file available at `%localappdata% > EdgeVerve > AutomationStudio > ProtonFiles > PdfRepository` folder, contains some configuration fields which can be edited in the notepad and used for

advanced operations. For example, in tabular form, where the data spread over multiple pages, a table with separate rows must be created instead of extracting the entire table. To create a table with separate rows, edit the **ExtractPagewise** field of the PDF template as **true**. This parameter exists at the end of the template.

- Following are configuration fields that can be edited:
 - **ExtractPagewise** - This field is used to append multiple tabular outputs to a single CSV file.
 - **AutoAddSpaces** and **Space Width** - What appears as a space, may not be a space character in the file. It could be two sequential characters placed at a distance to give an illusion of space. By default, if there is no space character in the file, the extracted output appears without space. You can set the **AutoAddSpaces** field to true and define the **Space Width** flag to let the extractor add a space character whenever the distance between two characters exceeds the space width value in PDF units. Use the corner draw area button to estimate the required space width.
 - **Output mapping** - This field can be used to map to another variable of Boolean or String type.
 - **IsNewRowNewField** - Set this field to true to extract a table with one column. Data flows from top to bottom and then to the right, instead of the common scenario of left to right and then top to bottom.
 - Password related configurations:
 - The extraction of password protected PDF files is supported in Automation Studio. Whenever the template is exported to Automation Studio, **PdfPassword** parameter is created in the **Arguments** pane.

Name	Direction	Argument type	Default value
ItemName	Out	String	Default value not supported
PageReferences	Out	DataTable	Default value not supported
PdfPassword	In	String	Enter a VB expression
PageCount	Out	Int32	Default value not supported
Create Argument			

Variables Arguments 100%

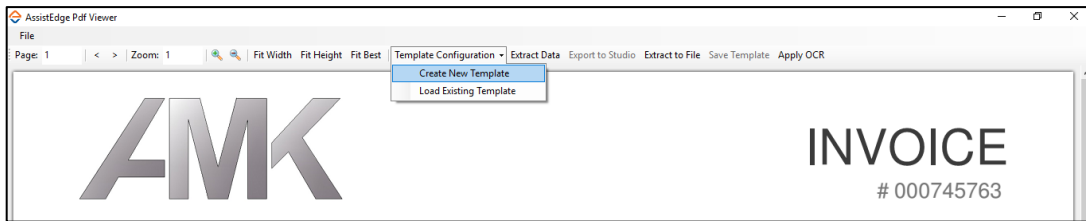
- Do not assign any default value to the argument as it gets stored in the process file. It is recommended to create an automation process workflow using PDF Extractor activity and use the process as a re-use process where the **PdfPassword** parameter is configured.
- About Backward compatibility:
 - Templates created in previous versions of Automation Studio cannot be viewed or edited in the current version. However, the automation process workflow configured using previous versions of Automation Studio can be executed in the current version.
 - Troubleshooting steps for various scenarios:
 - a. If the studio activity fails, check for the relevant logs. These logs are located inside the Logs folder in the build path (or Automation Studio parent folder).
 - b. Check if the file path provided for the PDF is correct.
 - c. Check if the template being used is relevant.
 - d. Open the PDF file in Google Chrome to check, if the search string and index combination are as intended.
 - e. Load the file and template and check if the intended area is correctly highlighted

Step-By-Step Guide to Use PDF Template Creator to Extract Tabular Data From The PDF File

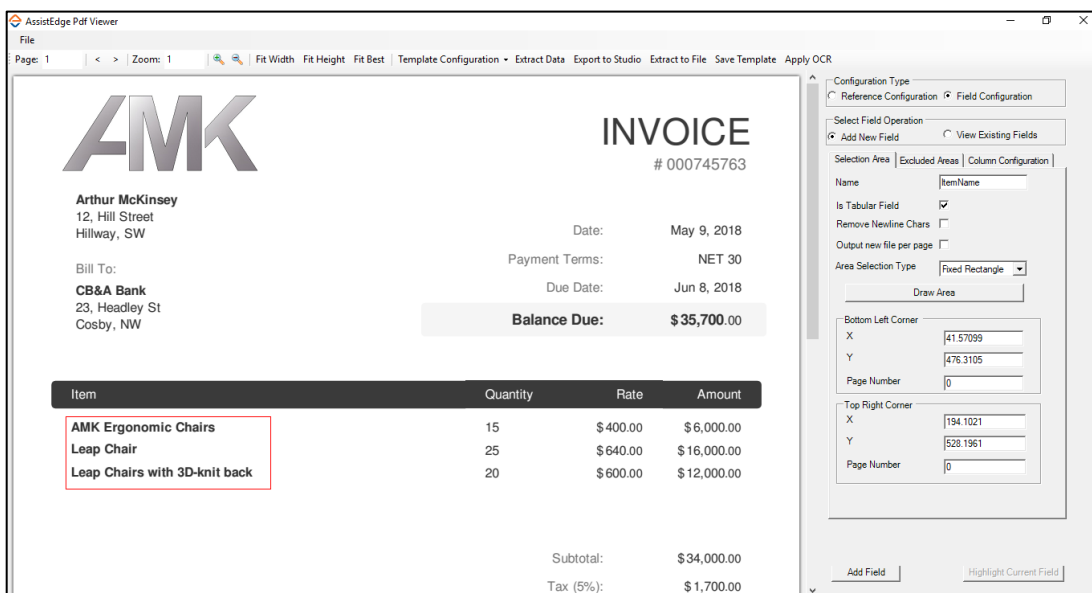
Let's see an example of creating a PDF template which is then used to extract tabular data from the PDF file. The example demonstrates tabular data extraction using the Fixed Rectangle, One Corner, Two Corner and Three Corner Area Selection Type.

Using Area Selection Type as Fixed Rectangle

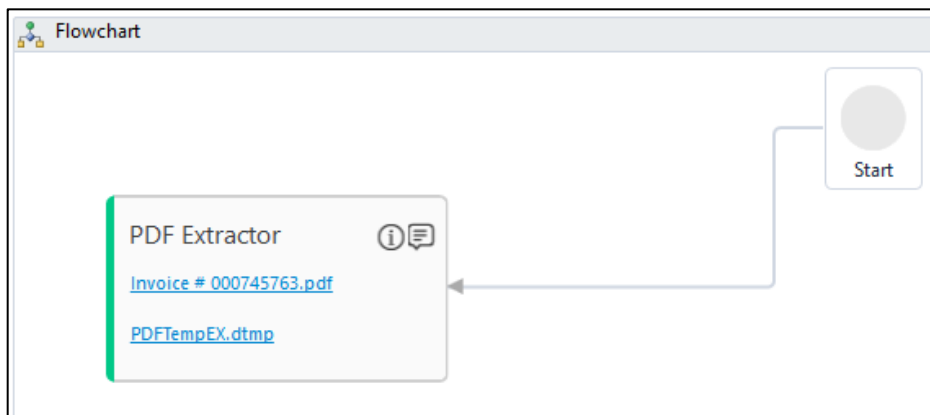
1. In the Canvas Tools panel, click PDF Template Creator.
2. Browse for the required PDF file and click **Open**. The **AssistEdge PDF Viewer** along with the selected PDF file appears.
3. In the menu bar of the **AssistEdge Pdf Viewer** screen of the **PDF Template Creator** canvas tool, click **Template Configuration**. Select **Create New Template** to create a new template. A configuration pane appears on the right side of the **AssistEdge PDF Viewer** screen.



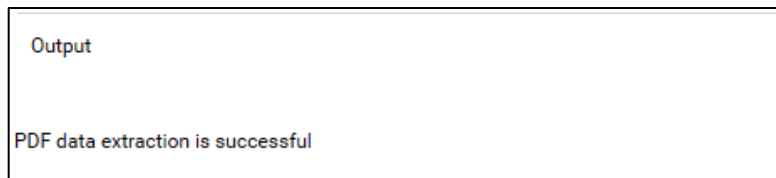
4. In the **Configuration Type** group, select **Field Configuration** to configure the fields in the PDF template for tabular data extraction.
5. In the Selected Area tab, enter ItemNames as the name of the field of the tabular data to be marked.
6. Select the **Is Tabular Field** check box to indicate marking of tabular data for extraction.
7. In the Area Selection Type list, select Fixed Rectangle.
8. Click **Draw Area** and mark the tabular data. The area gets highlighted within a box. The values of the **X** (horizontal) and **Y** (vertical) axis of the **Bottom Left Corner** and **Top Right Corner** groups along with the **Page Number** field get auto filled as per the area selected.



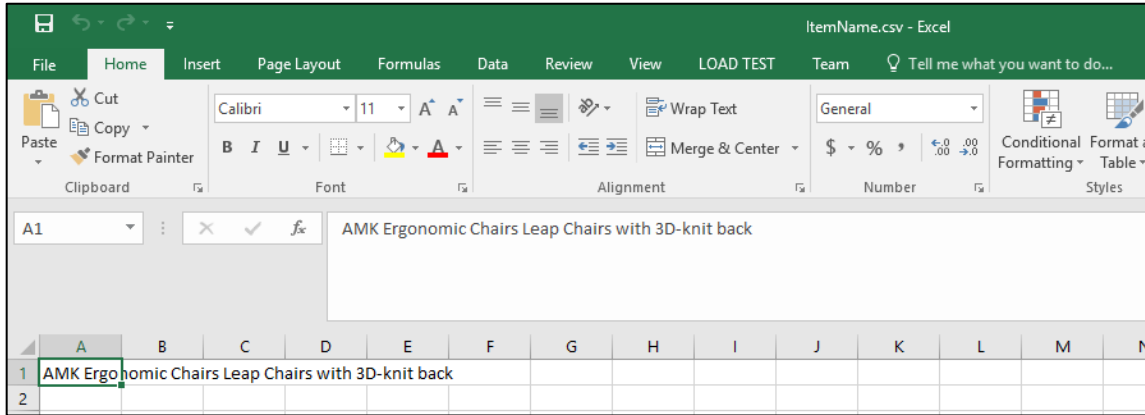
9. Click the **Excluded Area** tab.
10. Click **Add Excluded Area** button, and then select the second entry of the tabular data to exclude from extraction. The **Left Top X, Left Top Y, Right Top X, and Right Top Y** coordinates of the marked area get auto-filled.
11. Click **Done**.
12. Click **Add Field**. The marked area is added as a field.
13. Click **Save Template**. The **Save As** dialog box appears.
14. Provide the desired file name and click **Save**.
15. Click **Export to Studio**. This creates a **PDF Extractor** activity with the saved configurations. The specified PDF file name and the saved PDF template name is displayed in the **PDF Extractor** activity. If you click the displayed names, you are taken to the file location of respective files.



16. Save the process. To view the output in Automation Studio, set up the environment and perform test run. You can assign this process to a robot if you want to execute this process outside Automation Studio. A message for successful data extraction is displayed in the Output console of Automation Studio.

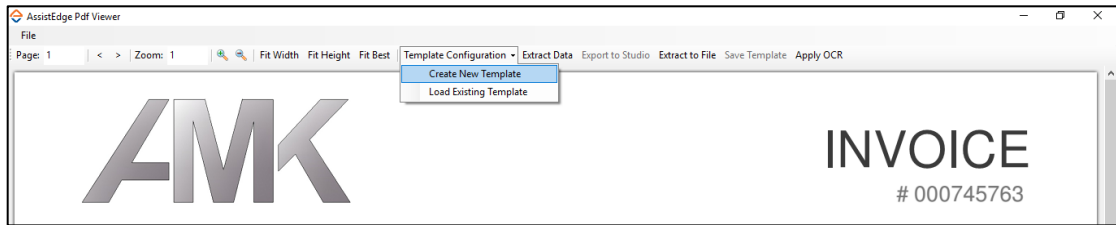


17. The .CSV file with the extracted data gets created at **%localappdata% > EdgeVerve > AutomationStudio** folder.

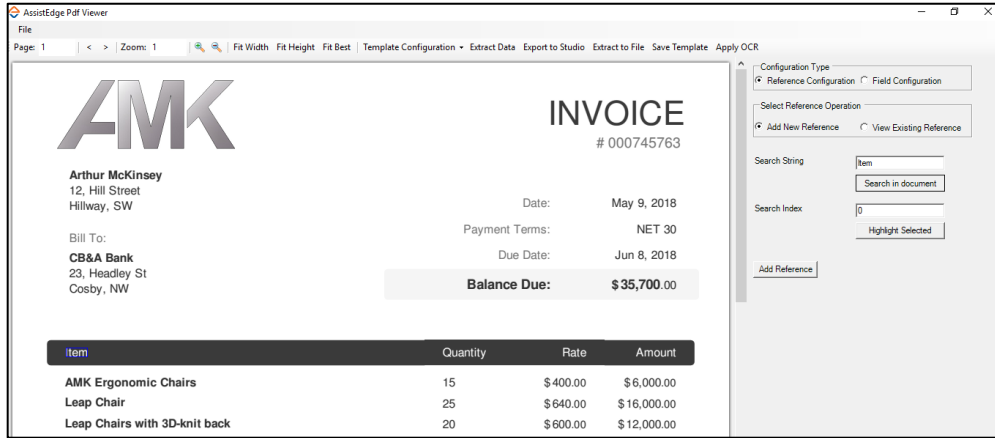


Using Area Selection Type as One Corner

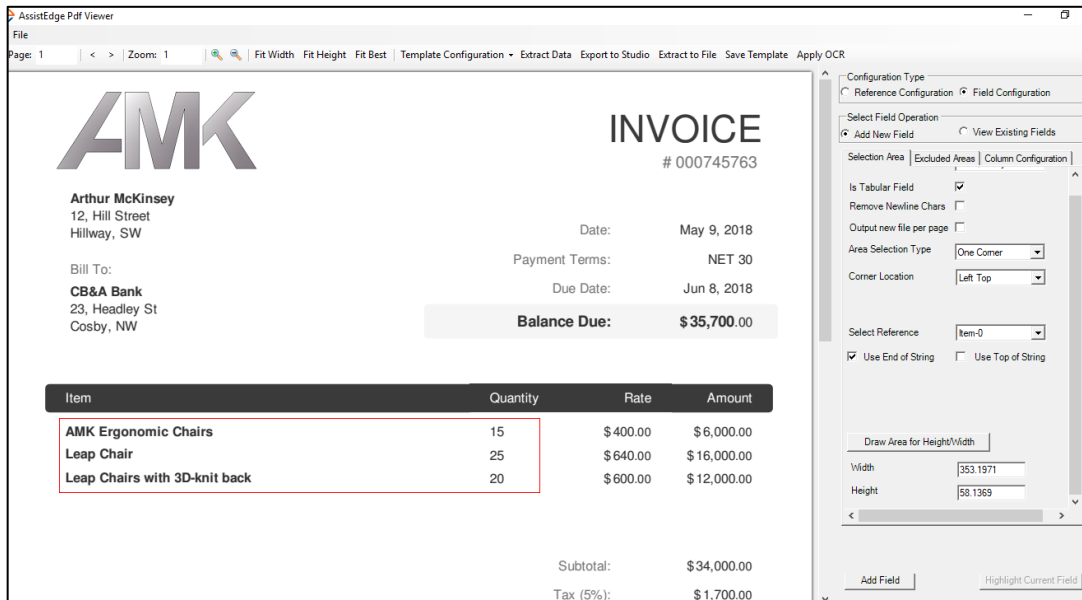
- 18. In the **Canvas Tools** panel, click PDF Template Creator.
- 19. Browse for the required PDF file and click **Open**. The **AssistEdge PDF Viewer** along with the selected PDF file appears.
- 20. In the menu bar of the **AssistEdge Pdf Viewer** screen of the **PDF Template Creator** canvas tool, click **Template Configuration**. Select **Create New Template** to create a new template. A configuration pane appears on the right side of the **AssistEdge PDF Viewer** screen.



- 21. In the **Configuration Type** group, select **Reference Configuration** to configure references in the PDF template for tabular data extraction.
- 22. In the Select Reference Operation group, select the Add New Reference option.
- 23. In the **Search String** text box, enter **Item** to mark this string for adding as a reference and then click **Search in document**. This highlights the relevant strings with a blue box across the PDF file.
- 24. In the **Search Index** text box, enter 0 as the index of the string, and then click **Highlight Selected**. This keeps the blue box highlight at the specified occurrence of the string.

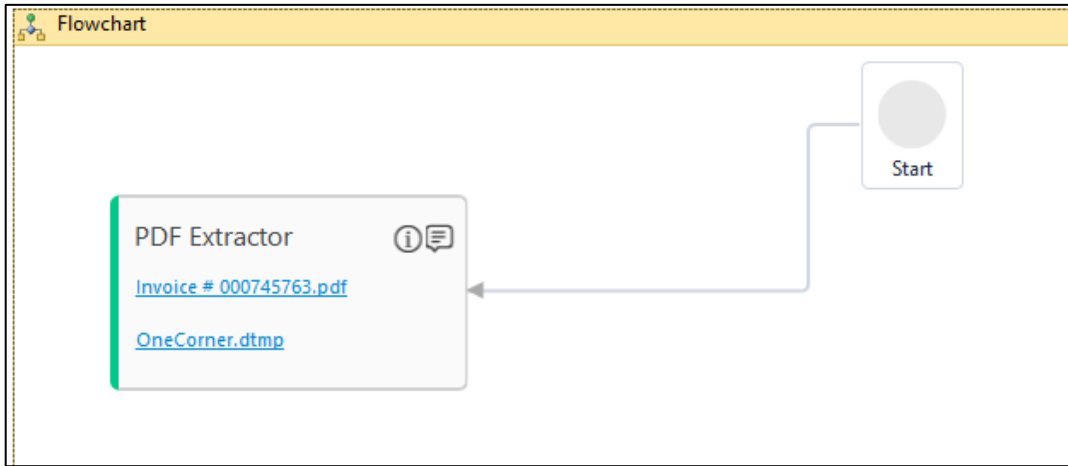


25. Click **Add Reference** to add the marked string as reference.
26. In the **Configuration Type** group, select **Field Configuration** to configure the fields in the PDF template for tabular data extraction.
27. In the **Selected Area** tab, enter **ItemQuantity** as the name of the field of the tabular data to be marked.
28. Select the **Is Tabular Field** check box to indicate marking of tabular data for extraction.
29. In the Area Selection Type list, select One Corner.
30. In the **Corner Location** list, select **Left Top** option.
31. Click **Draw Area for Height/Width**, and then manually draw the height and width of the boundary from the top left corner of the tabular data. The area gets highlighted with a red box. The **Width** and **Height** fields are auto-filled.

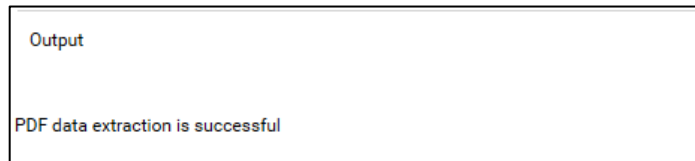


32. Click **Add Field** to confirm and add the field.
33. Click **Save Template**. The **Save As** dialog box appears.
34. Provide the desired file name and click **Save**.
35. Click **Export to Studio**. This creates a **PDF Extractor** activity with the saved configurations. The specified PDF file name and

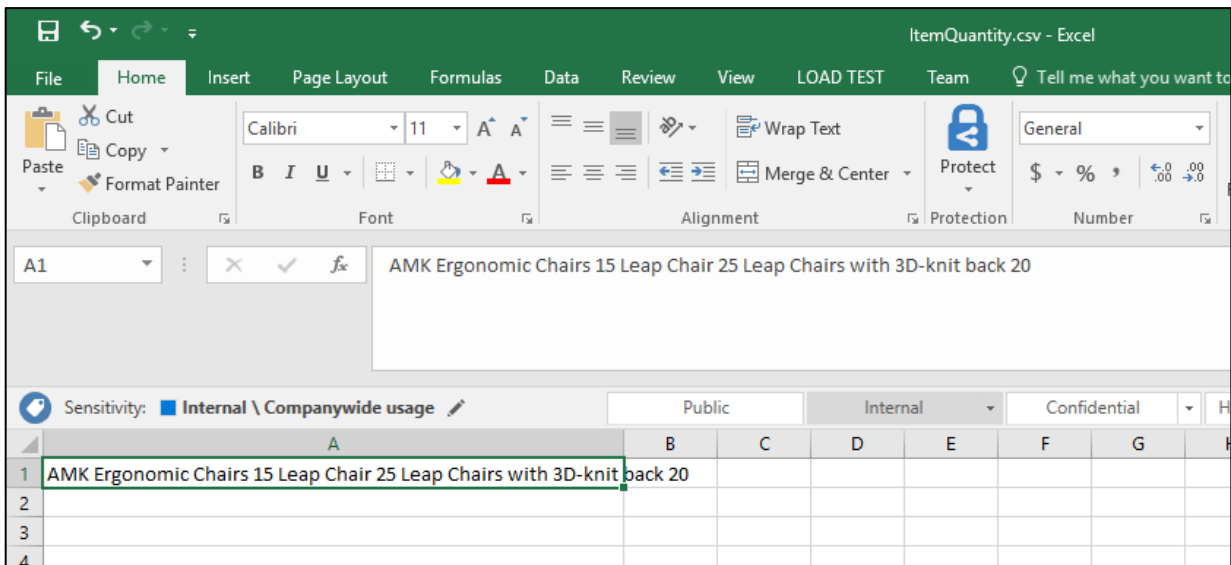
the saved PDF template name is displayed in the **PDF Extractor** activity. If you click the displayed names, you are taken to the file location of respective files.



36. Save the process. To view the output in Automation Studio, set up the environment and perform test run. You can assign this process to a robot if you want to execute this process outside Automation Studio. A message for successful data extraction is displayed in the **Output** console of Automation Studio.



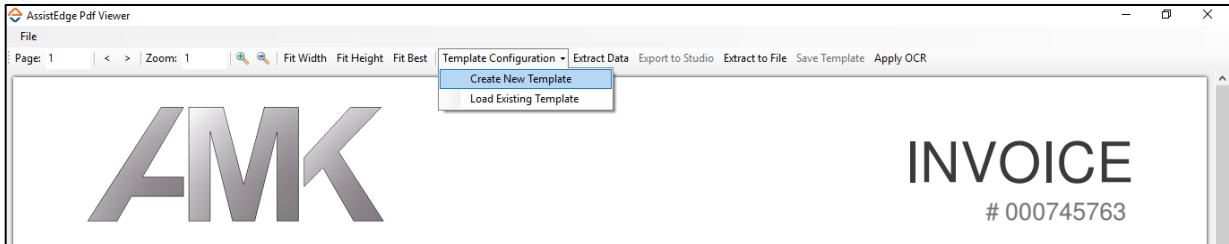
37. The .CSV file with the extracted data gets created at %localappdata% > **EdgeVerve**> **AutomationStudio** folder.



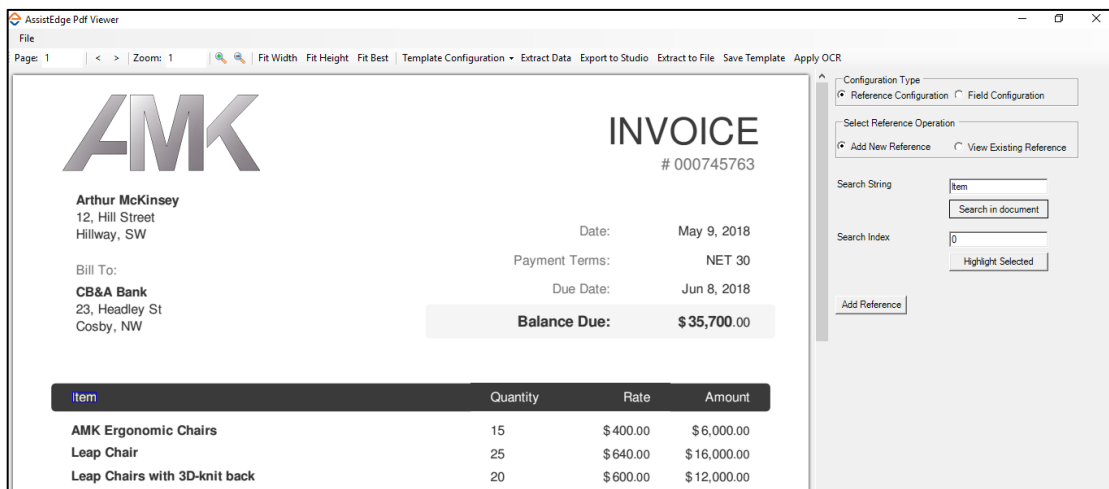
Using Area Selection Type as Two Corner

1. In the **Canvas Tools** panel, click PDF Template Creator.

- Browse for the required PDF file and click **Open**. The **AssistEdge PDF Viewer** along with the selected PDF file appears.
- In the menu bar of the **AssistEdge Pdf Viewer** screen of the **PDF Template Creator** canvas tool, click **Template Configuration**. Select **Create New Template** to create a new template. A configuration pane appears on the right side of the **AssistEdge PDF Viewer** screen.

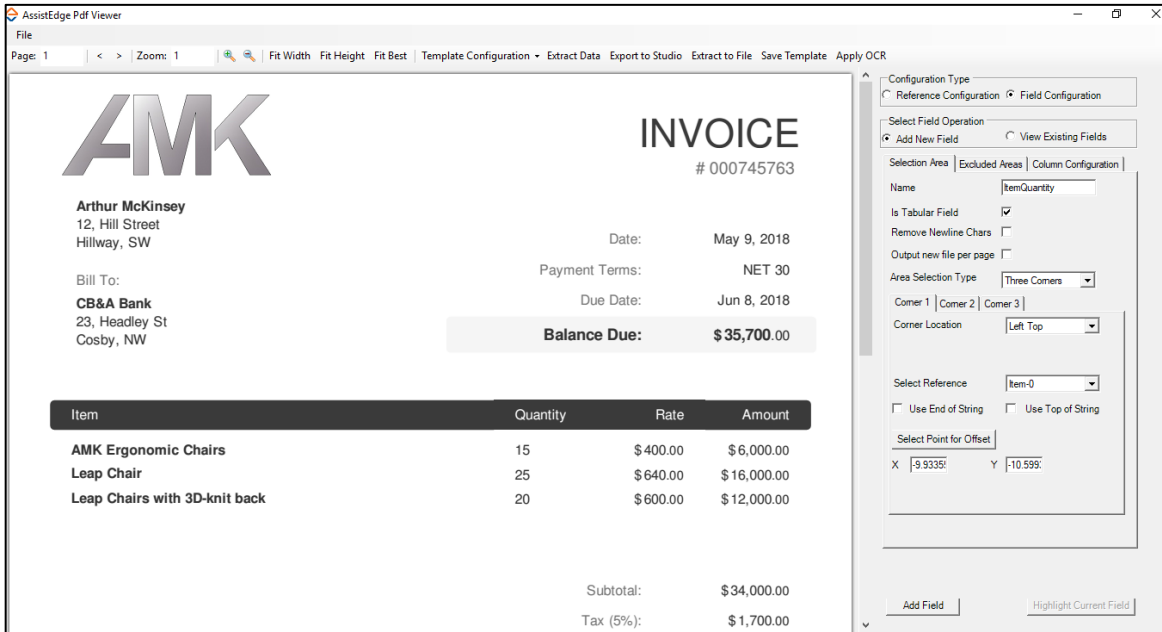


- In the **Configuration Type** group, select **Reference Configuration** to configure references in the PDF template for tabular data extraction.
- In the **Select Reference** Operation group, select the **Add New Reference** option.
- In the **Search String** text box, enter **Item** to mark this string for adding as a reference and then click **Search in document**. This highlights the relevant strings with a blue box across the PDF file.
- In the **Search Index** text box, enter 0 as the index of the string, and then click **Highlight Selected**. This keeps the blue box highlight at the specified occurrence of the string.

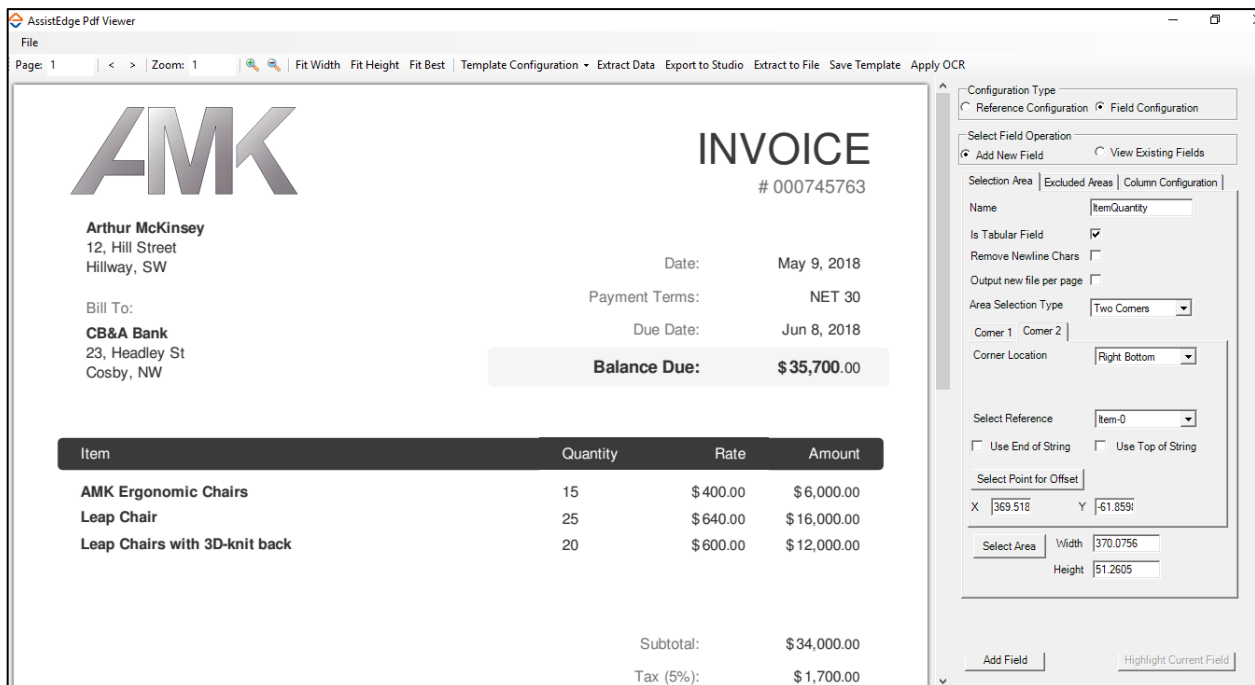


- Click **Add Reference** to add the marked string as reference.
- In the **Configuration Type** group, select **Field Configuration** to configure the fields in the PDF template for tabular data extraction.
- In the **Selected Area** tab, enter **ItemQuantity** as the name of the field of the tabular data to be marked.
- Select the **Is Tabular Field** check box to indicate marking of tabular data for extraction.
- In the Area Selection Type list, select Three Corner.
- In the **Corner 1** tab, select **Left Top** option to define the location of field boundary of the first corner.
- In the **Select Reference** list, select **Item-0** reference string defined above.

- Click **Select Point for Offset** and then choose a point from where the boundary of the first corner can be drawn. This auto-fills the values of the offset of the selected corner.

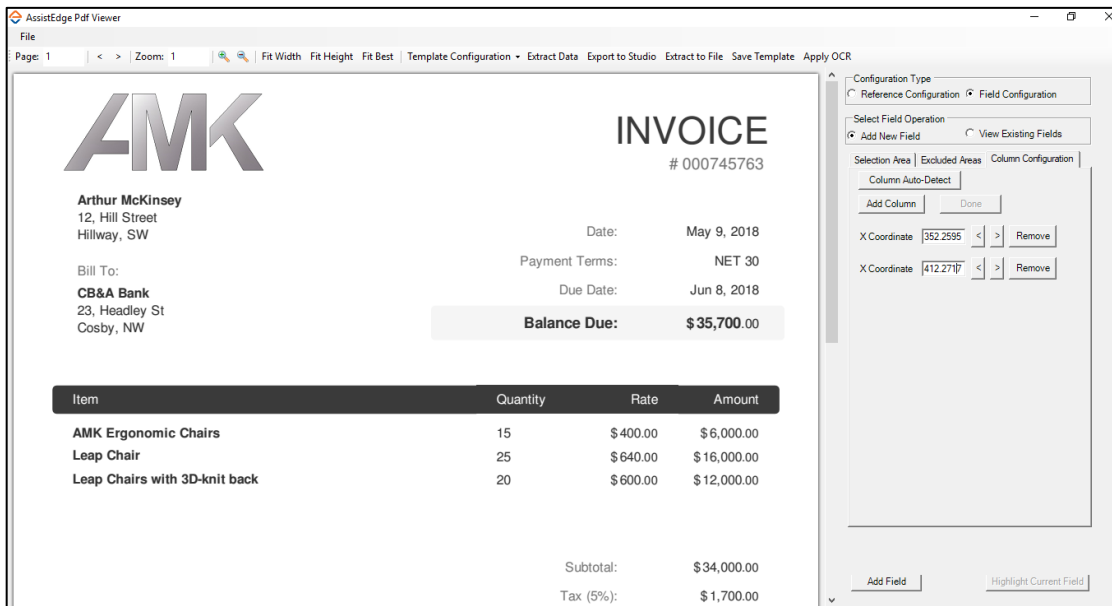


- In the **Corner 2** tab, select **Left Bottom** option to define the location of field boundary of the second corner.
- In the **Corner Location** list, select **Left Top** option.
- Click **Draw Area for Height/Width**, and then manually draw the height and width of the boundary from the top left corner of the tabular data. The area gets highlighted with a red box. The **Width** and **Height** fields are auto-filled.

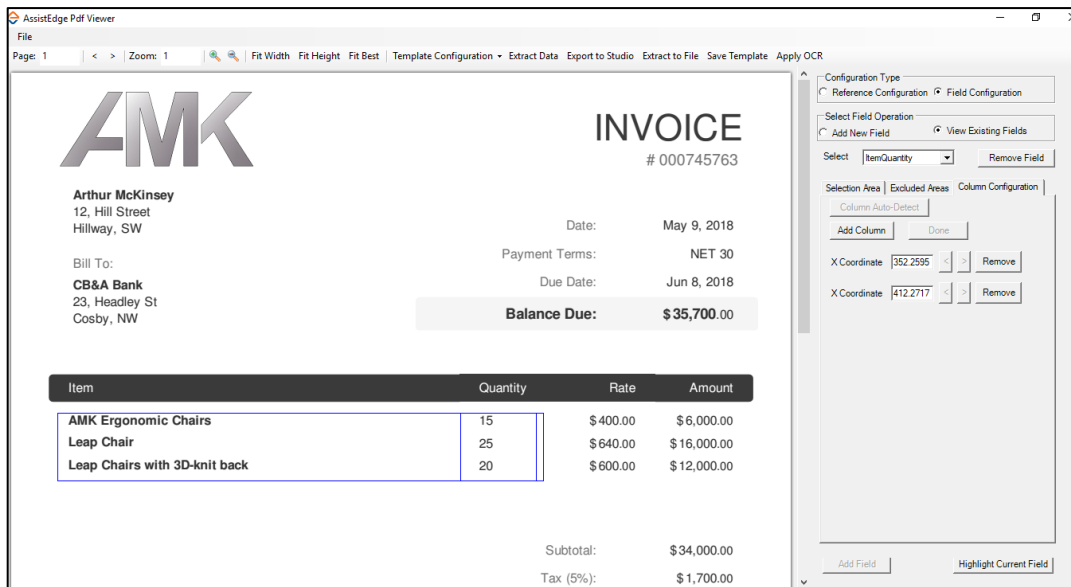


- Click the **Column Configuration** tab.

- Click **Add Column** to manually mark the columns. Draw vertical lines between the tabular fields that you want to mark for column configuration.

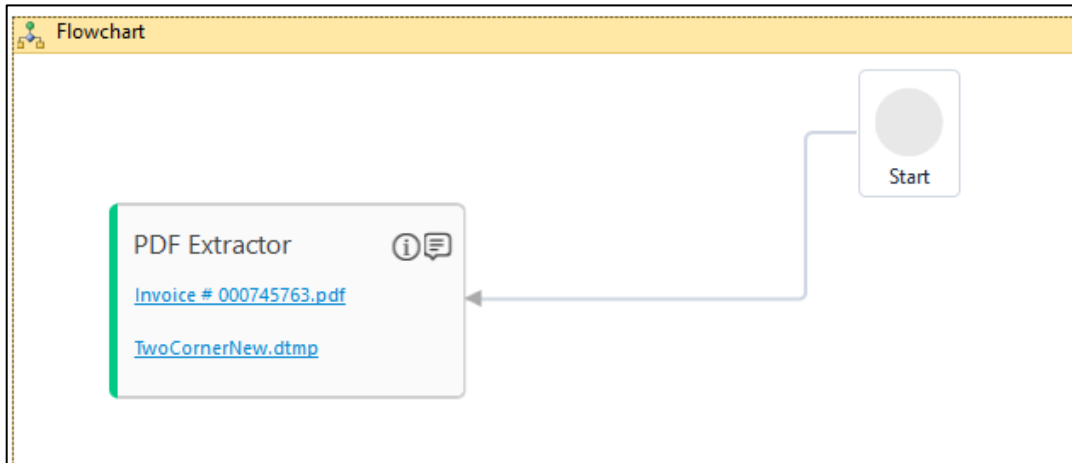


- Click **Done** to save the defined column configuration.
- Click **Add Field** to confirm and add the field.
- In the **Select Field Operation** group, select **View Existing Fields** option to view and verify the area marked for extraction.
- In the **Select** drop-down list, select **ItemQuantity**.



- Click **Highlight Current Field** button. This highlights the marked or updated area for data extraction.
- Click **Save Template**. The **Save As** dialog box appears.
- Provide the desired file name and click **Save**.

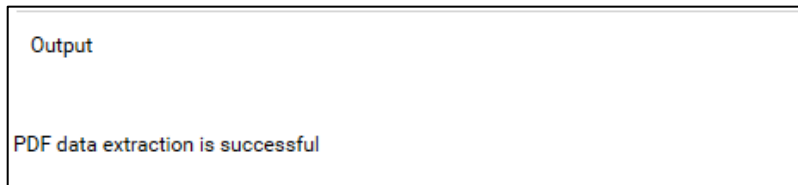
- 28. Click **Export to Studio**. This creates a **PDF Extractor** activity with the saved configurations. The specified PDF file name and the saved PDF template name is displayed in the **PDF Extractor** activity. If you click the displayed names, you are taken to the file location of respective files.



- 29. **Save** the process.

To view the output in Automation Studio, set up the environment and perform test run. You can assign this process to a robot if you want to execute this process outside Automation Studio.

A message for successful data extraction is displayed in the Output console of Automation Studio.



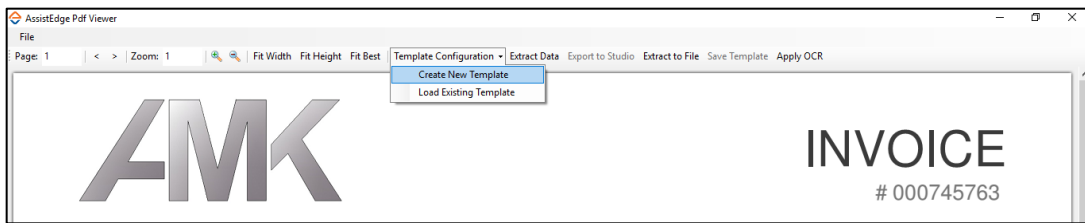
The .CSV file with the extracted data gets created at %localappdata% > EdgeVerve > AutomationStudio folder.

The screenshot shows an Excel spreadsheet titled 'ItemQuantity.csv - Excel'. The ribbon includes 'File', 'Home', 'Insert', 'Page Layout', 'Formulas', 'Data', 'Review', 'View', 'LOAD TEST', 'Team', and 'Tell me'. The 'Home' tab is active, showing options for Cut, Copy, Paste, and Format Painter. The font is set to Calibri, size 11. The active cell is A1, containing the text 'AMK Ergonomic Chairs'. Below the ribbon, there is a 'Sensitivity' dropdown set to 'Internal \ Companywide usage'. The spreadsheet data is as follows:

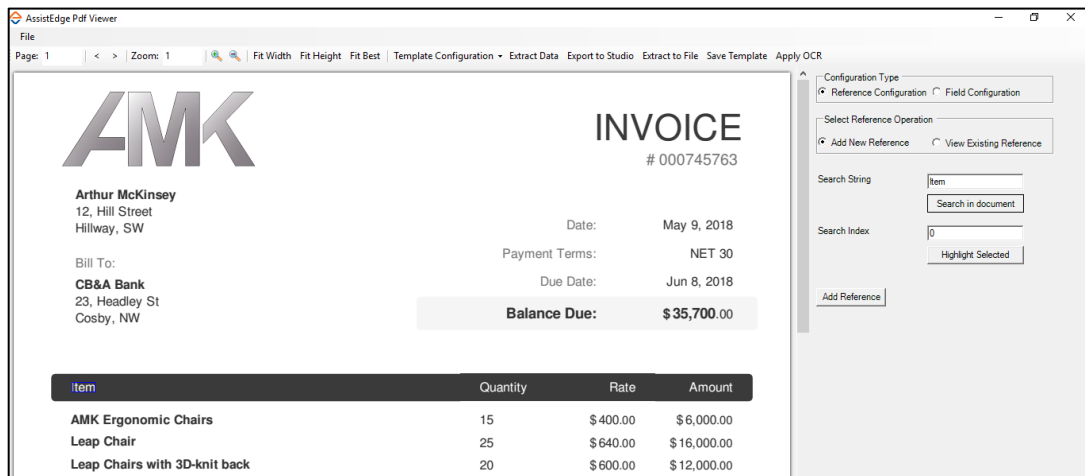
	A	B	C	D	E	F	G	H	I
1	AMK Ergonomic Chairs	15							
2	Leap Chair	25							
3	Leap Chairs with 3D-knit back	20							
4									

Using Area Selection Type as Three Corner

1. In the **Canvas Tools** panel, click PDF Template Creator.
2. Browse for the required PDF file and click **Open**. The **AssistEdge PDF Viewer** along with the selected PDF file appears.
3. In the menu bar of the **AssistEdge Pdf Viewer** screen of the **PDF Template Creator** canvas tool, click **Template Configuration**. Select **Create New Template** to create a new template. A configuration pane appears on the right side of the **AssistEdge PDF Viewer** screen.

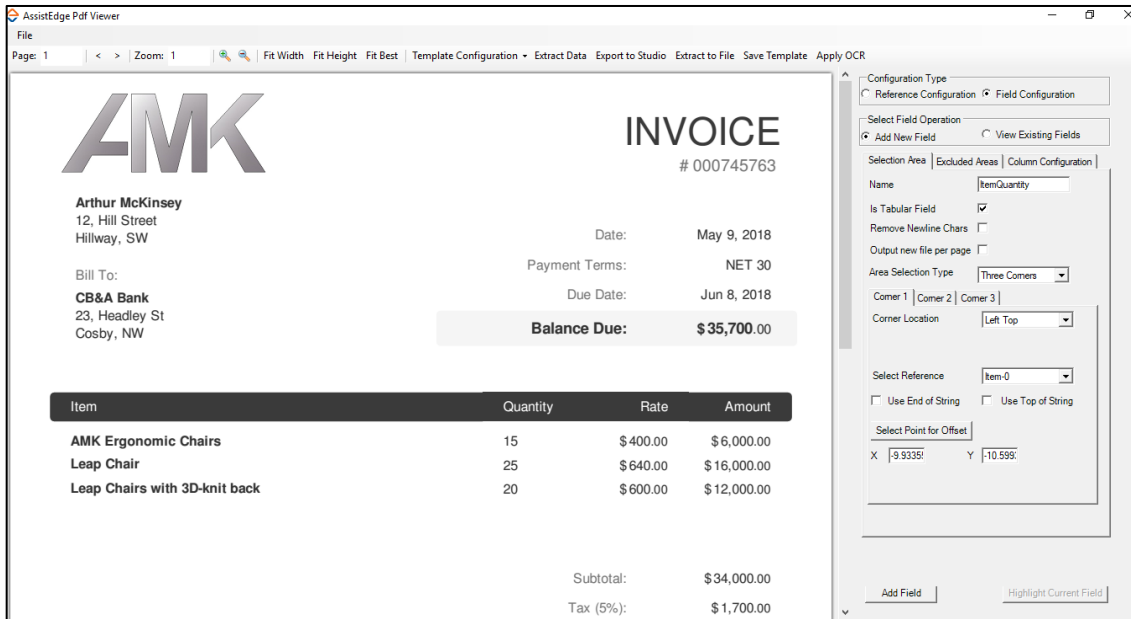


4. In the **Configuration Type** group, select **Reference Configuration** to configure references in the PDF template for tabular data extraction.
5. In the Select Reference Operation group, select the Add New Reference option.
6. In the **Search String** text box, enter Item to mark this string for adding as a reference and then click **Search in document**. This highlights the relevant strings with a blue box across the PDF file.
7. In the **Search Index** text box, enter 0 as the index of the string, and then click **Highlight Selected**. This keeps the blue box highlight at the specified occurrence of the string.

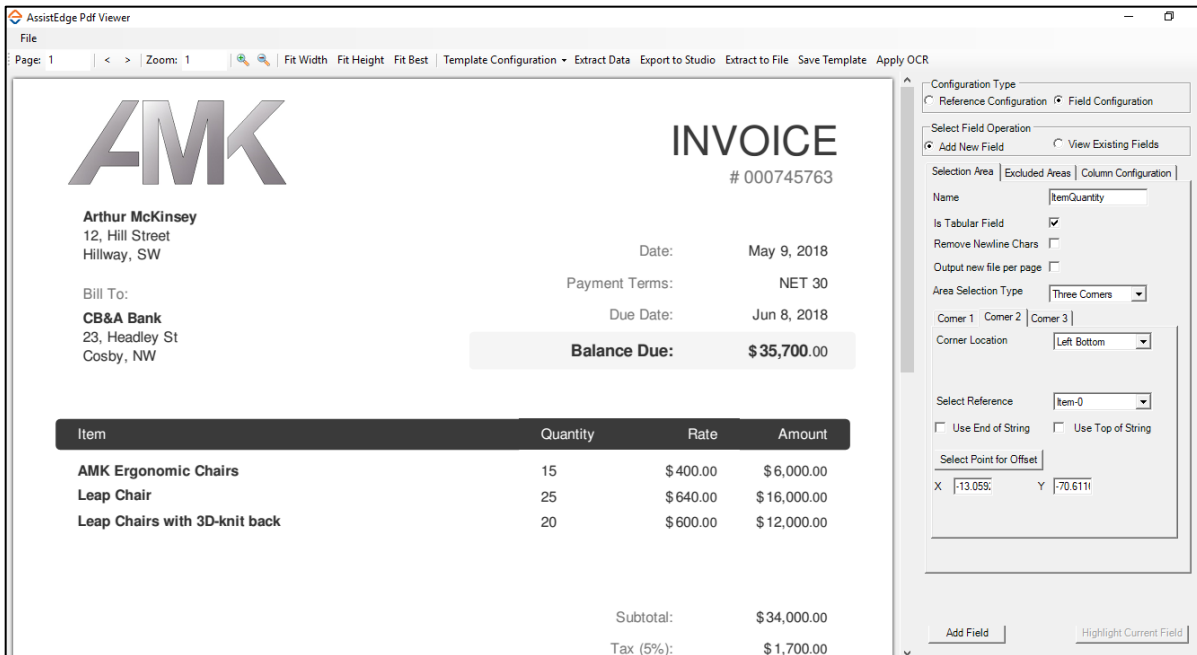


8. Click **Add Reference** to add the marked string as reference.
9. In the **Configuration Type** group, select **Field Configuration** to configure the fields in the PDF template for tabular data extraction.
10. In the **Selected Area** tab, enter **ItemQuantity** as the name of the field of the tabular data to be marked.
11. Select the **Is Tabular Field** check box to indicate marking of tabular data for extraction.
12. In the Area Selection Type list, select Three Corner.

13. In the **Corner 1** tab, select **Left Top** option to define the location of field boundary of the first corner.
14. In the Select Reference list, select **Item-0** reference string defined above..
15. Click **Select Point for Offset** and then choose a point at the top left area around the required table from where the boundary of the first corner can be drawn. This auto-fills the values of the offset of the selected corner.

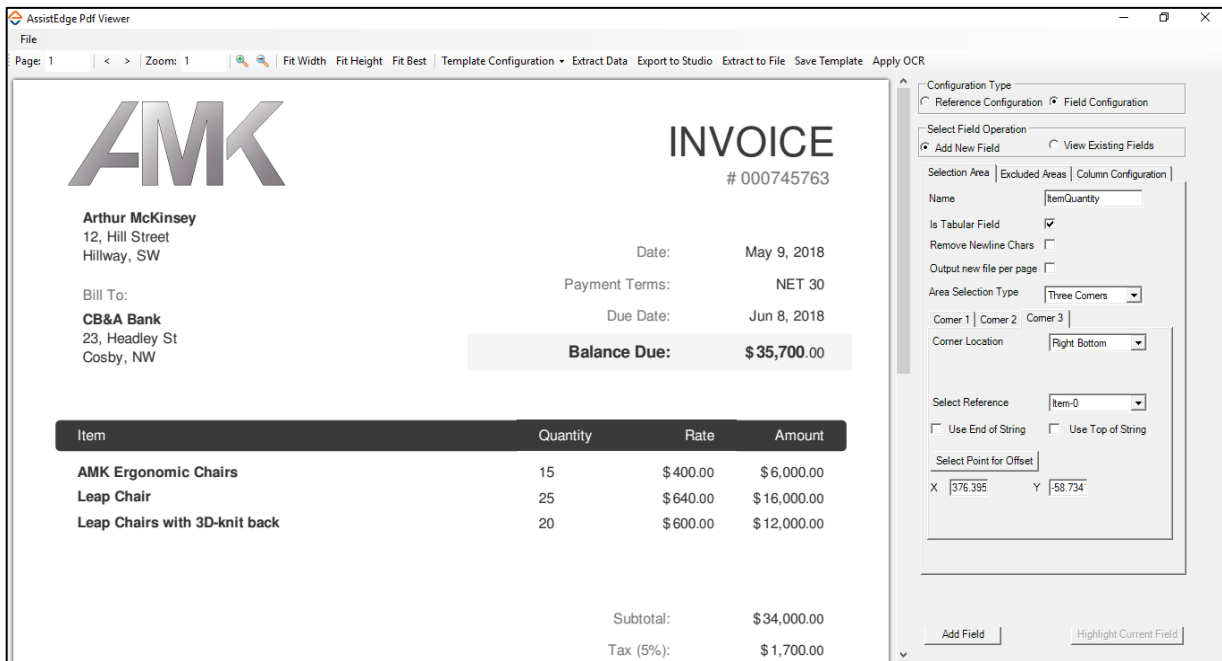


16. In the **Corner 2** tab, select **Left Bottom** option to define the location of field boundary of the first corner.
17. Click **Select Point for Offset** and then choose a point at the bottom left area around the required table from where the boundary of the second corner can be drawn. This auto-fills the values of the offset of the selected corner.

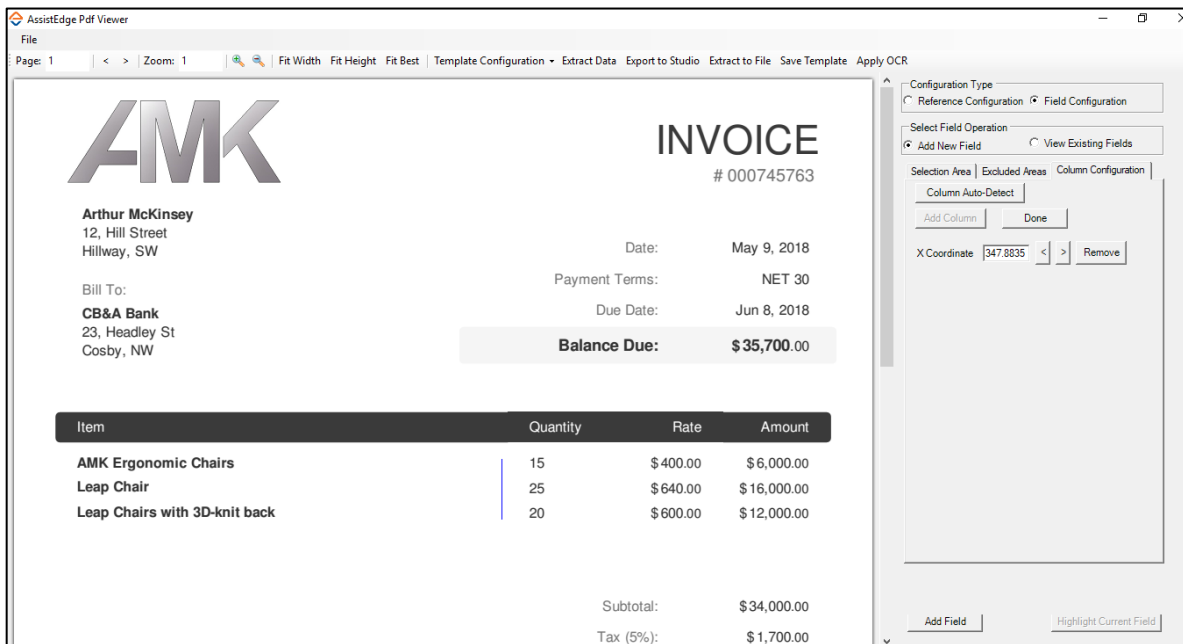


18. In the **Corner 3** tab, select **Right Bottom** option to define the location of field boundary of the first corner.

- Click **Select Point for Offset** and then choose a point at the bottom right area around the required table from where the boundary of the third corner can be drawn. This auto-fills the values of the offset of the selected corner.

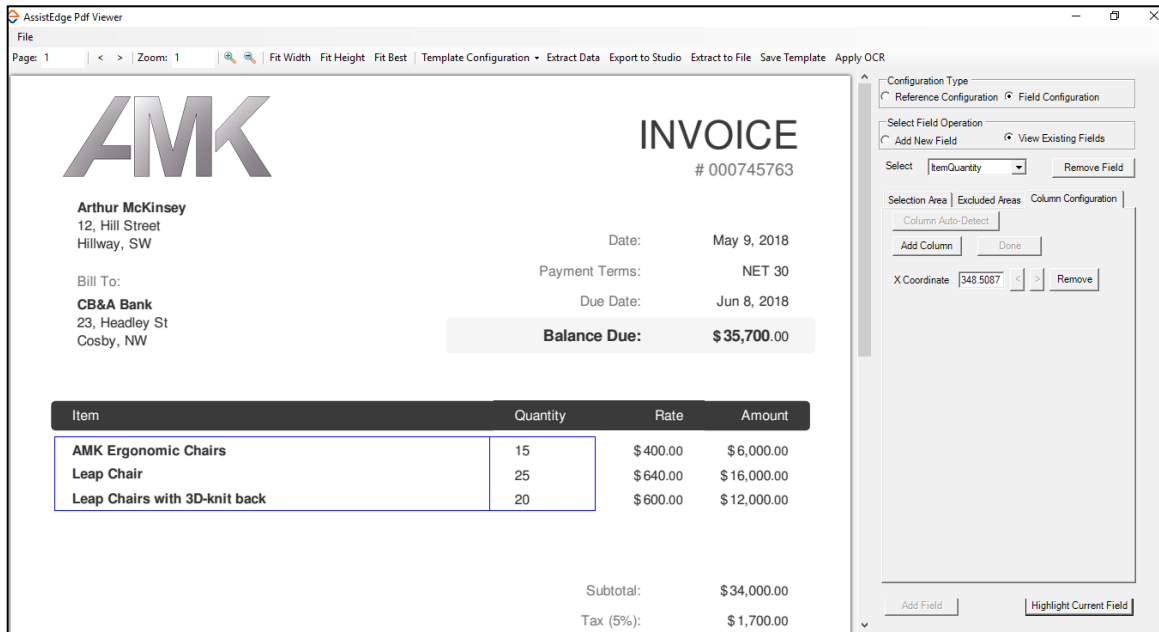


- Click the Column Configuration tab.
- Click **Add Column** to manually mark the columns. Draw vertical lines between the tabular fields that you want to mark for column configuration.



- Click **Done** to save the defined column configuration.
- Click **Add Field** to confirm and add the field.
- In the **Select Field Operation** group, select **View Existing Fields** option to view and verify the area marked for extraction.

25. In the Select drop-down list, select **ItemQuantity**.

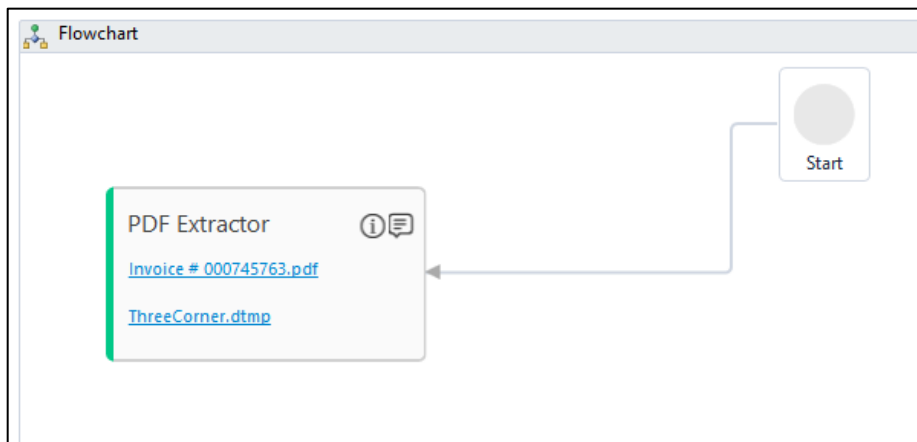


26. Click **Highlight Current Field** button. This highlights the marked or updated area for data extraction.

27. Click **Save Template**. The **Save As** dialog box appears.

28. Provide the desired file name and click **Save**.

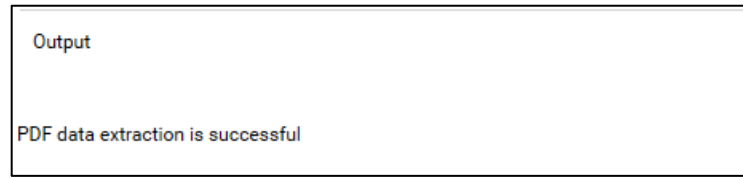
29. Click **Export to Studio**. This creates a **PDF Extractor** activity with the saved configurations. The specified PDF file name and the saved PDF template name is displayed in the **PDF Extractor** activity. If you click the displayed names, you are taken to the file location of respective files.



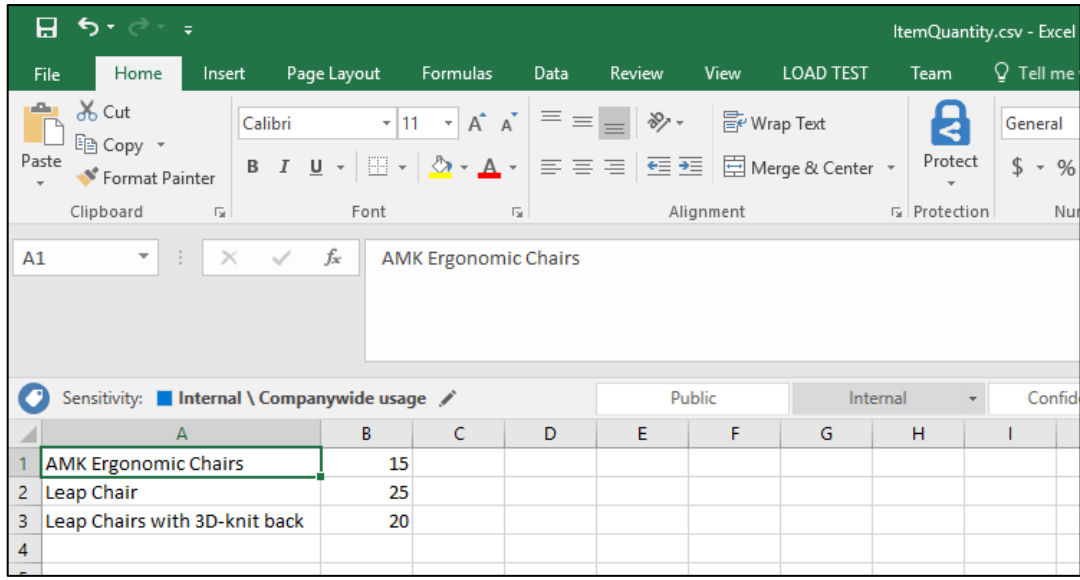
30. Save the process.

To view the output in Automation Studio, set up the environment and perform test run. You can assign this process to a robot if you want to execute this process outside Automation Studio.

A message for successful data extraction is displayed in the **Output** console of Automation Studio.



The .CSV file with the extracted data gets created at %localappdata% > EdgeVerve> AutomationStudio folder.



9.18 Excel Recorder

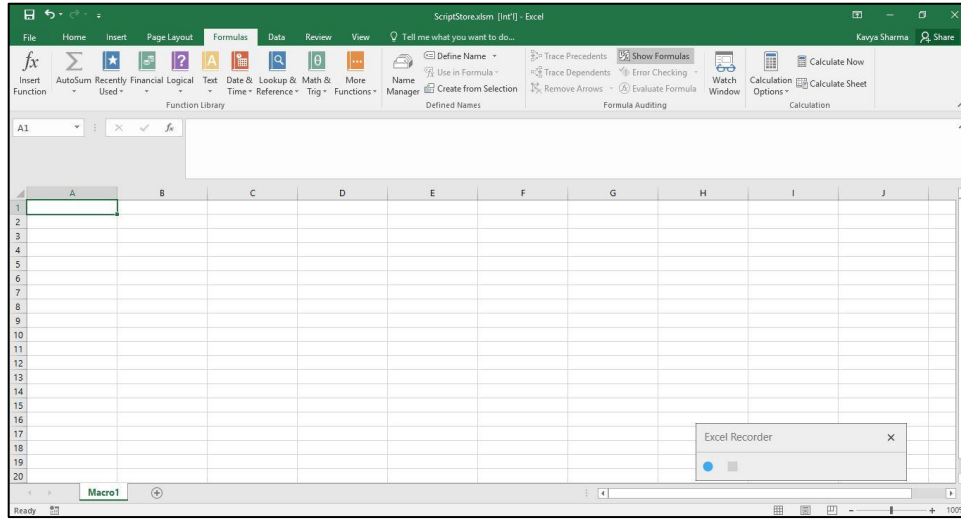
Excel recorder is a powerful excel automation tool that captures everything that the Excel itself can offer through its own Macro recorder. It records the steps and operations performed in Microsoft Excel application such as working across different excel files or worksheets and performing a repetitive task over the excel file.

9.18.1 Prerequisites

Ensure in the Trust Centre settings options of the Excel application, the Enable all macros and Trust access to the VBA project object model check boxes are selected.



9.18.2 Using the Excel Recorder

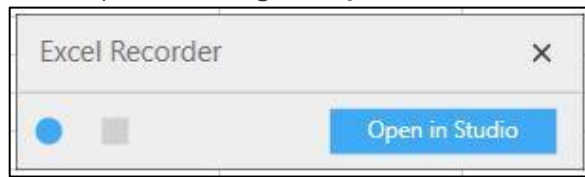
1. In the **Canvas Tools** pane, click Excel Recorder to launch the tool. The **ScriptStore.xlsm** excel file along with the **Excel Recorder** interface appears.



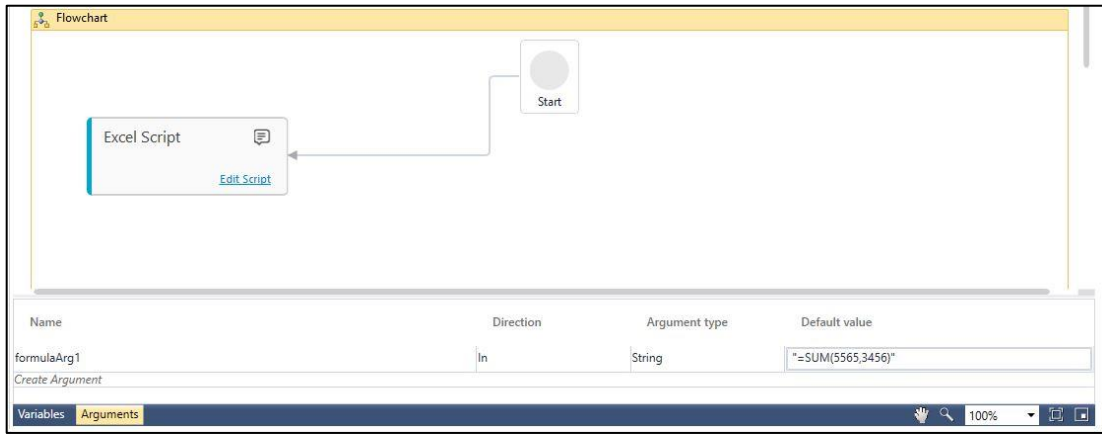
Note:

- You can save the opened .xlsm file to perform the required operation or open an existing excel file by navigating through File option of the opened excel file.
- You can open an existing file once the excel file and the Excel Recorder interface appears.
- Before you open a different file, you must start the recording.
- When you select an entire Excel sheet by using Ctrl+A+A and copy paste the selection in another sheet, workbook, or application, you may observe the System.
- Exception:ScriptStoreException: Out of memory error in the Automation Studio log file. To fix this error, see **RPA Troubleshooting Guide.pdf**.

2. Click the  (**Start Recording**) icon to start the recording and perform the steps to record for automation. For example, you can input the formula in a cell or access the formula from the Formulas section. You can use the formatting options in the similar way. The recording icon turns grey.
3. Click the  (**Stop Recording**) icon to stop the recording. The **Open in Studio** button appears.



4. Click **Open in Studio** button to return to the **Canvas**. An **Excel Script** activity is created with a link to the script of the steps recorded. It automatically creates arguments related to the operations performed in the excel file with default values. You can edit the name of the created argument. Below is sample activity and the related arguments that got created automatically:



The Excel Recorder activity is created with a default name.

Editing Script

Click the Edit Script link to edit the scripts of the steps recorded. The scripts captured opens in the Notepad application. Below is a sample script:

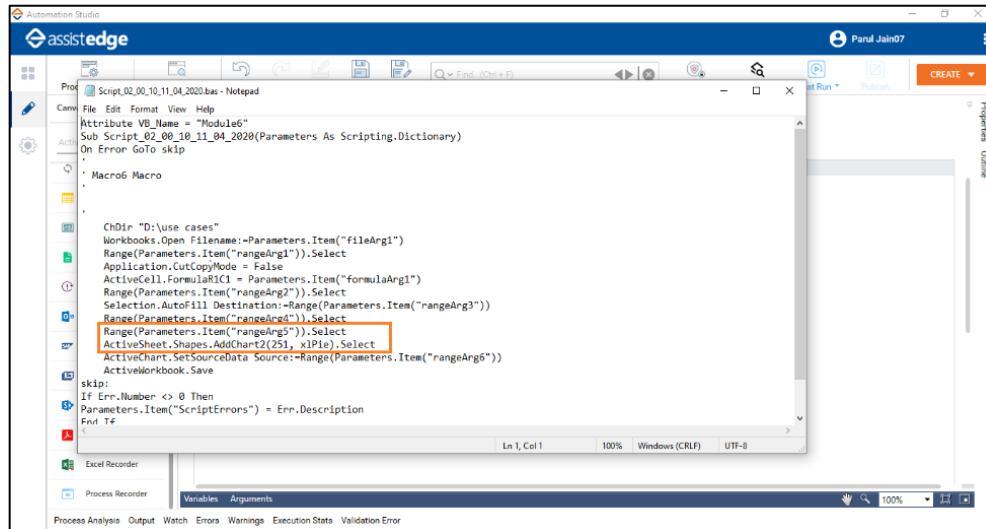
```
Script_15_49_00_03_08_2020.bas - Notepad
File Edit Format View Help
Attribute VB_Name = "Module5"
Sub Script_15_49_00_03_08_2020(Parameters As Scripting.Dictionary)
On Error GoTo skip
'
' Macro6 Macro
'
'
ActiveCell.FormulaR1C1 = Parameters.Item("formulaArg1")
skip:
If Err.Number <> 0 Then
Parameters.Item("ScriptErrors") = Err.Description
End If
End Sub
```

You can configure the dynamic part of a static recording using the Edit Script option.

While recording the excel operations, if the static fields are captured by the cell value and not by action, you can change those steps into generic dynamic functions by editing the scripts. Alternatively, capture the static fields using activate cells, rather than selecting them to make it dynamic during the recording itself. For more information, search about Excel VBA on the web.

Usage Tips

- Find last row of the column:
- Offset the cell value and make the selection of the call as comment using the Edit Script options. Below is an example of finding the last row:



- Input large script formulas:
- If large text-based formulas like VLOOKUP are not captured during recording, then input them in the edit script area.
- Build a complete use case
- Record small steps using the **Excel Recorder** activity, and then combine them using the **Reuse Process** activity to form an automation process workflow.

Automation using Excel Recorder

The following list of operations have been tested while the list is not restricted to the same:

Excel 2010	Test for Excel 2010
Extract	<ul style="list-style-type: none"> ▪ Extract a cell value and provide as input to other activity or automation process workflow ▪ One-time extraction of the entire column, deletion of columns and insertion of columns ▪ Extract cell value for a column till blank cell
Copy	<ul style="list-style-type: none"> ▪ Copy a cell value from one sheet to another sheet ▪ Copy a column from one sheet to another sheet ▪ Copy entire sheet and paste it in a new sheet or new excel file ▪ Copy a selected range of cells (column and rows)
Formulas	<ul style="list-style-type: none"> ▪ VLOOKUP: Paste the VLOOKUP formula in a cell ▪ Concatenate: Two columns can be concatenated, and the value can be pasted in a new column ▪ Sum: For entire column or selected range of cells inside the column ▪ Count: For entire column or selected range of cells inside the column ▪ Paste formula in a blank cell or selected range of cells
Macro	Invoke an existing Macro
Sort	Sort the entire column date wise
New Sheet	Create of a new sheet (tab)
Filter	<ul style="list-style-type: none"> ▪ Enable filter for all columns ▪ Filter, select labels (one or multiple) and then extract the entire range ▪ Clear all filters
Create New Excel	<ul style="list-style-type: none"> ▪ Create a new excel file and name it ▪ Naming convention where the name is appended with the date

Excel 2010	Test for Excel 2010
Save	<ul style="list-style-type: none"> ▪ Save option of the excel file ▪ Save As option after downloading the excel file from the SAP application
Paste	<ul style="list-style-type: none"> ▪ Paste the text ▪ Paste Special: Right click Paste Special and paste Values (V) ▪ Paste in the selected range

9.18.3 Excel Scripts Properties

The properties of Excel Scripts activity are listed in the following table and can be edited in the Property grid on the right pane.

Property Name	Usage
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity. If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the Flowchart designer. By default, the name is set as Excel Scripts . You can change the name as required.
Name	The default name with which the script gets saved. You can change the name as required.
Path	The default file path where the script file gets saved. You can change the location as required.

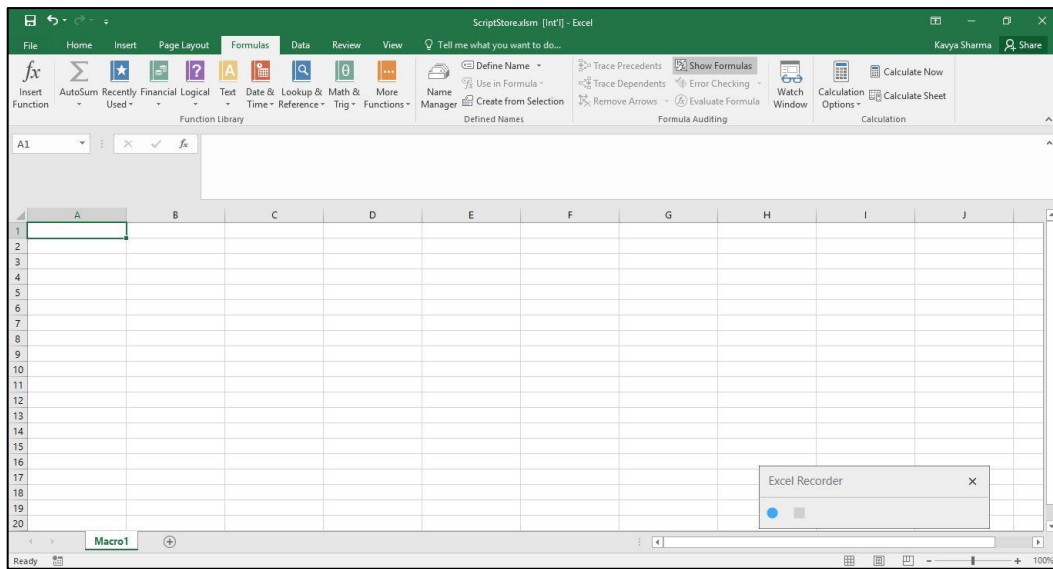
9.18.4 Step-By-Step Guide to Use Excel Recorder to Calculate the Invoice Total


Let's create an example of automating the process of calculating the invoice total of the customer bill available in an excel file.

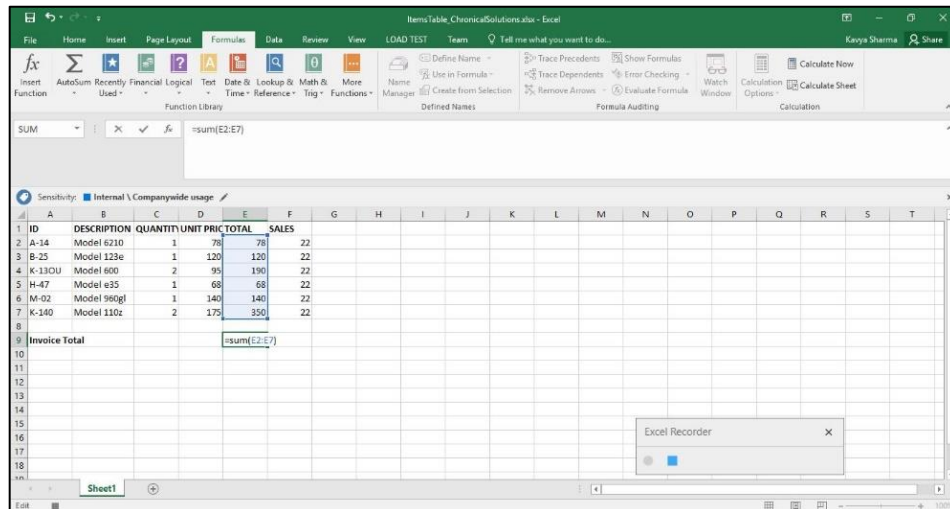
Prerequisite:


In the Trust Centre settings options of the Excel application, the Enable all macros and Trust access to the VBA project object model check boxes are selected.

1. In the **Canvas Tools** panel, click **Excel Recorder**. The **ScriptStore.xlsm** excel file along with the **Excel Recorder** interface appears.



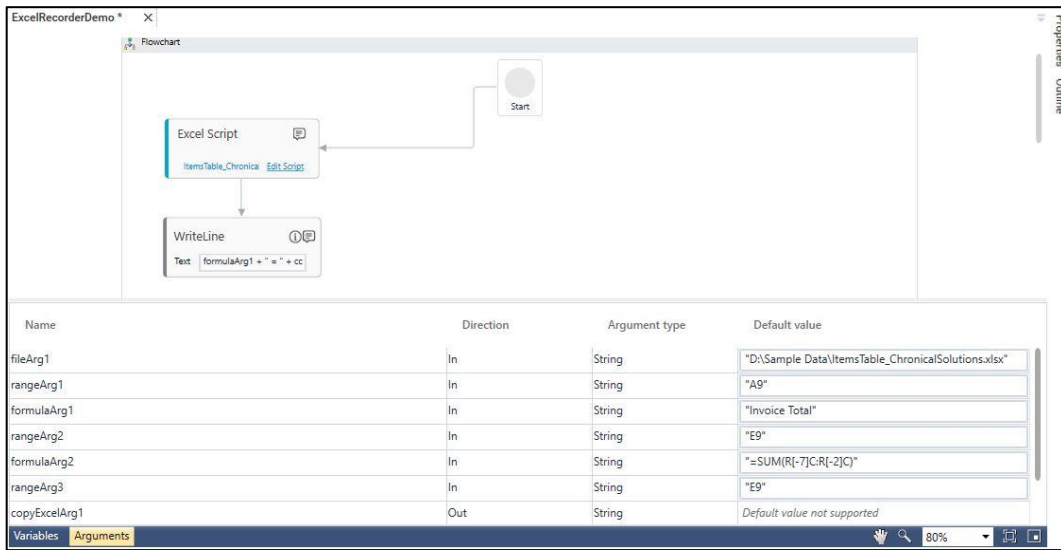
2. Click the  (**Start Recording**) icon to start the recording and perform the steps to record for automation.
3. In the **Menu** bar of the **ScriptStore.xlsm** excel file, click **File**.
4. Click **Open. Browse** and select the required customer bill excel file. The excel file appears.



5. Enter the text **Invoice Total** at the end of the rows.
6. Calculate the sum of the **Total** column of the items mentioned in the invoice excel and press Enter. The total amount appears.
7. Copy the cell where the invoice total is displayed.
8. Click the  (**Stop Recording**) icon to stop the recording. The **Open in Studio** button appears.



9. Click **Open in Studio** button to return to the **Canvas**. An **Excel Script** activity is created with a link to the script of the steps recorded. It automatically creates arguments related to the operations performed in the excel file with default values. You can edit the name of the created argument.



10. Save the process.
To view the calculated invoice total in Automation Studio, let's add **WriteLine** activity. You can assign this process to a robot, if you want to execute the process outside Automation Studio.
11. Add a **WriteLine** activity below the **Excel Script** activity and in the **Text** field, enter **formulaArg1 + " " + copyExcelArg1**
12. Setup the environment and then perform test run.



The **Output** console displays the calculated invoice total.

9.19 Process Recorder

Process Recorder is used to record a business process, either in full or in small parts, and convert it into an automation process flowchart.

Process recorder supports web and windows desktop applications. Supported browsers includes Internet Explorer, Chrome, MS Edge, and Firefox. Windows desktop application includes all the windows exe, such as, Word, Notepad, Excel, calculator. It does not support JAVA, Mainframe and SAP applications.

Process recorder supports cross application recording, so, you can record any business process which includes web and windows desktop applications in one process.

9.19.1 Prerequisites

As a prerequisite of process recorder activity, you need to [add browser extension](#) and [configure application](#) in Studio to perform the process recordings.

Add Browser Extension

Add the required browser extension to perform the recording of the respective browsers.

- [Chrome extension](#)
- [Firefox extension](#)
- [Edge extension](#)

To add Chrome extension:

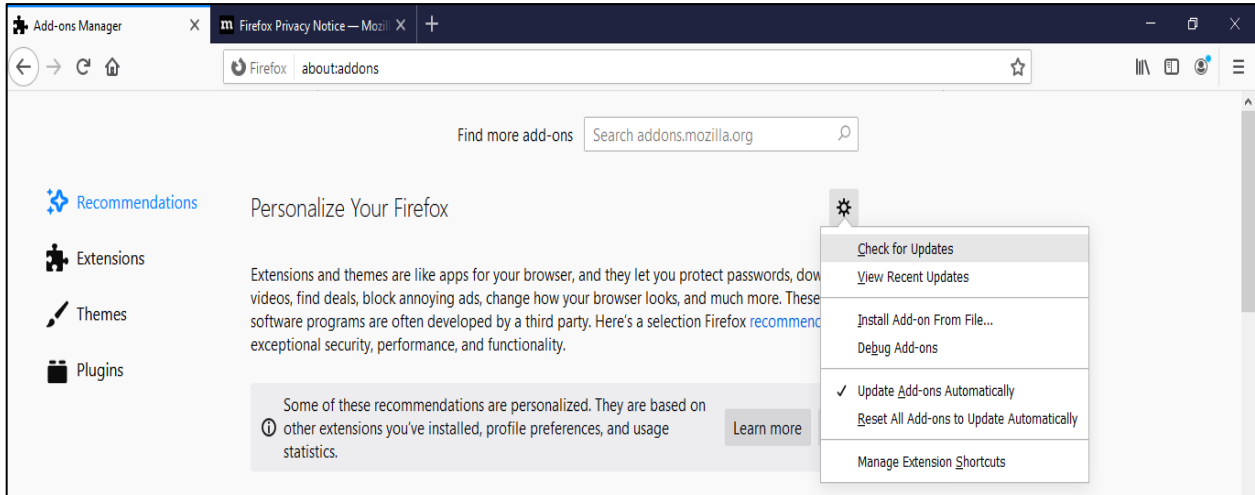
Access the Chrome webstore [AssistEdge Record and Playback extension - Chrome Web Store \(google.com\)](#) and follow the mentioned steps to add the AssistEdge process recorder extension.

To add Firefox extension:

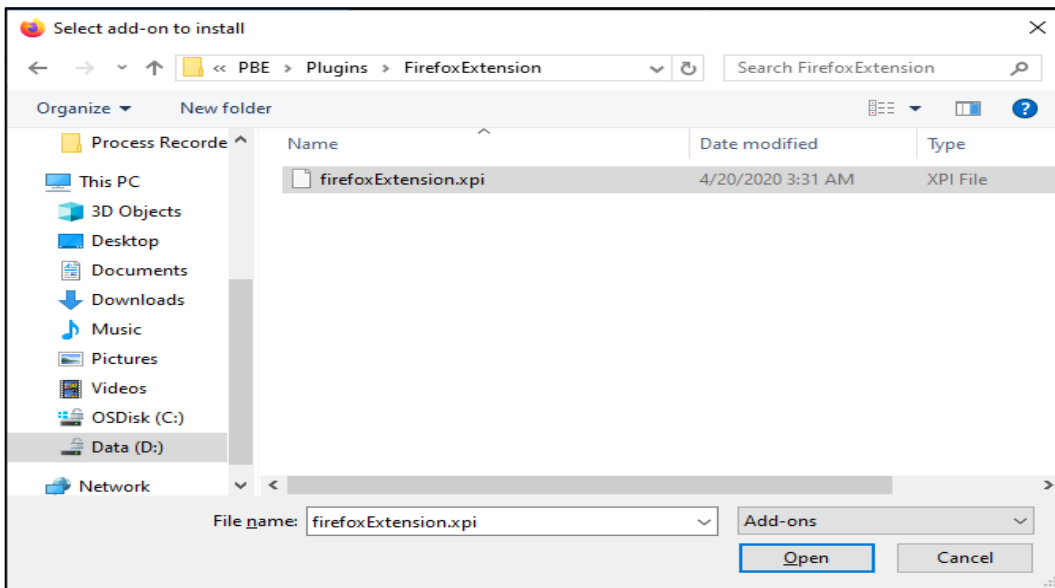
1. Open the Firefox browser and type **about:addons** in the address bar.



2. Click the  (**Settings**) icon, a list appears.

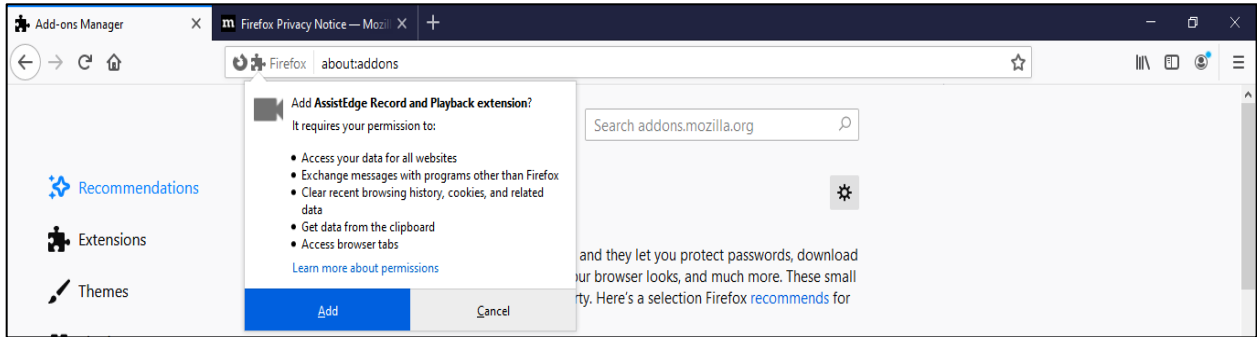


3. Click Install Add-on From File.... The select add-on to install window appears.



4. Select firefoxExtension.xpi from client-tools > AutomationStudio > bin> PBE > Plugins > FirefoxExtension location (or, %localappdata% > EdgeVerve > AutomationStudio > bin > PBE > Plugins > FirefoxExtension, If you download/access Automation Studio from the Admin module) and click Open.

5. Click **Add**.



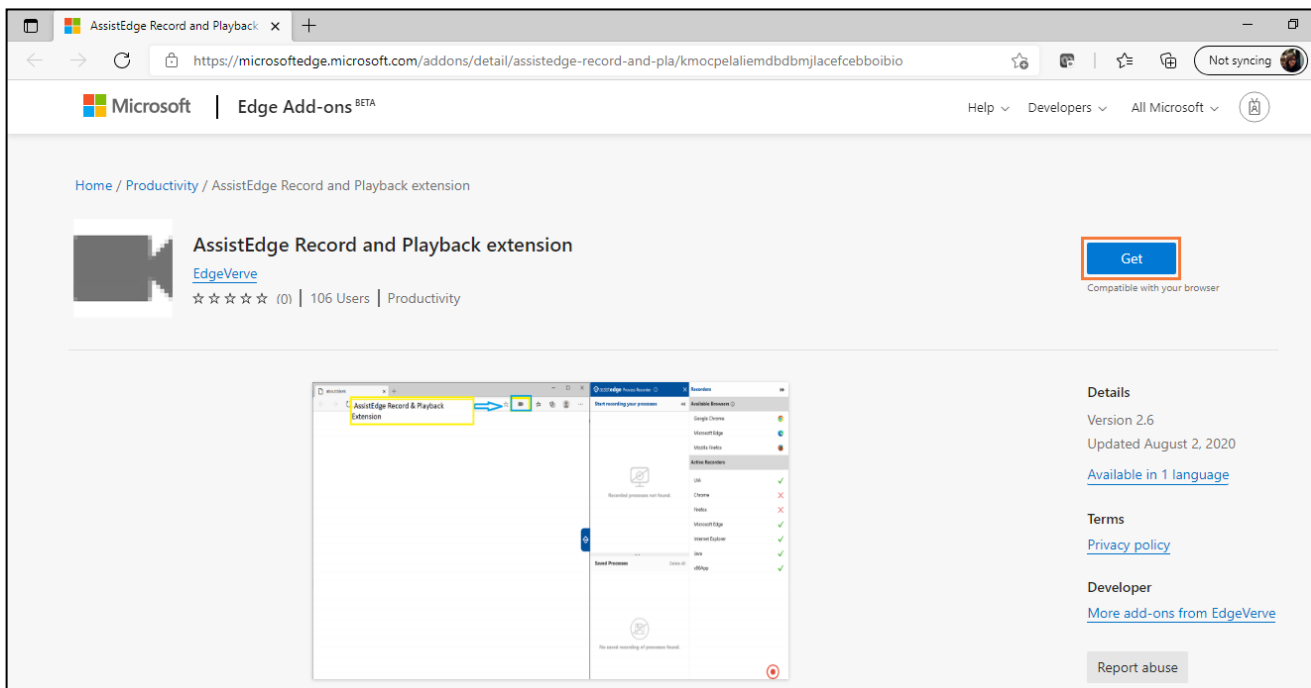
6. The  (camera) icon is displayed on the address bar.



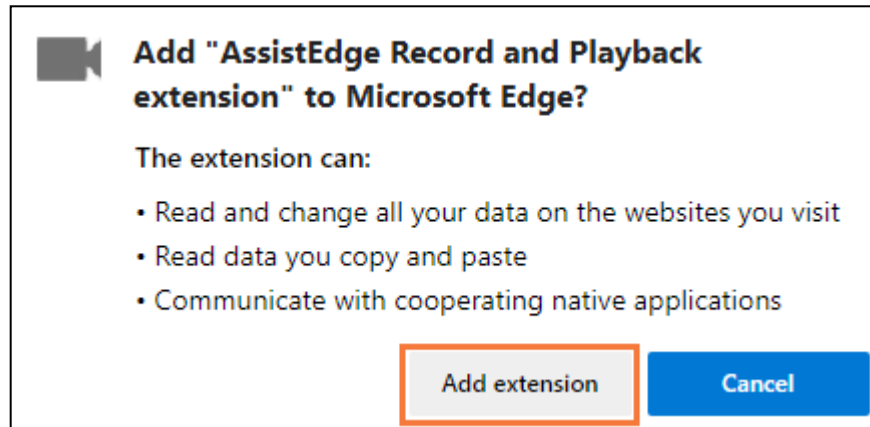
To add Edge extension:


7. Microsoft does not allow downloading the extension for Edge browser. Until this feature is available use the below mentioned link in the Microsoft Edge browser to add and access the Microsoft Edge Extension:

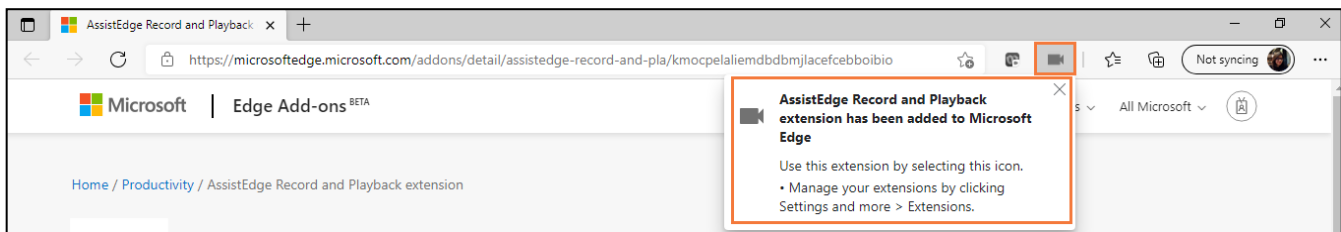
<https://microsoftedge.microsoft.com/addons/detail/kmocupelaliemdbdbmjlacefcebboibio>



8. Click **Get**. The Add "AssistEdge Record and Playback extension" to Microsoft Edge? appears.



9. Click Add extension.
10. The  (camera) icon is displayed on the address bar.



Simultaneously, a message AssistEdge Record and Playback extension has been added to Microsoft Edge appears in a pop-up window.

Configure Application

Before proceeding with the process recorder activity, you need to configure application in the admin tab of the Automation Studio as per your process requirement. Automation Studio supports three types of applications for recording:

- [WindowsApps](#)
- [ExcelApps](#)
- [WebApps](#)

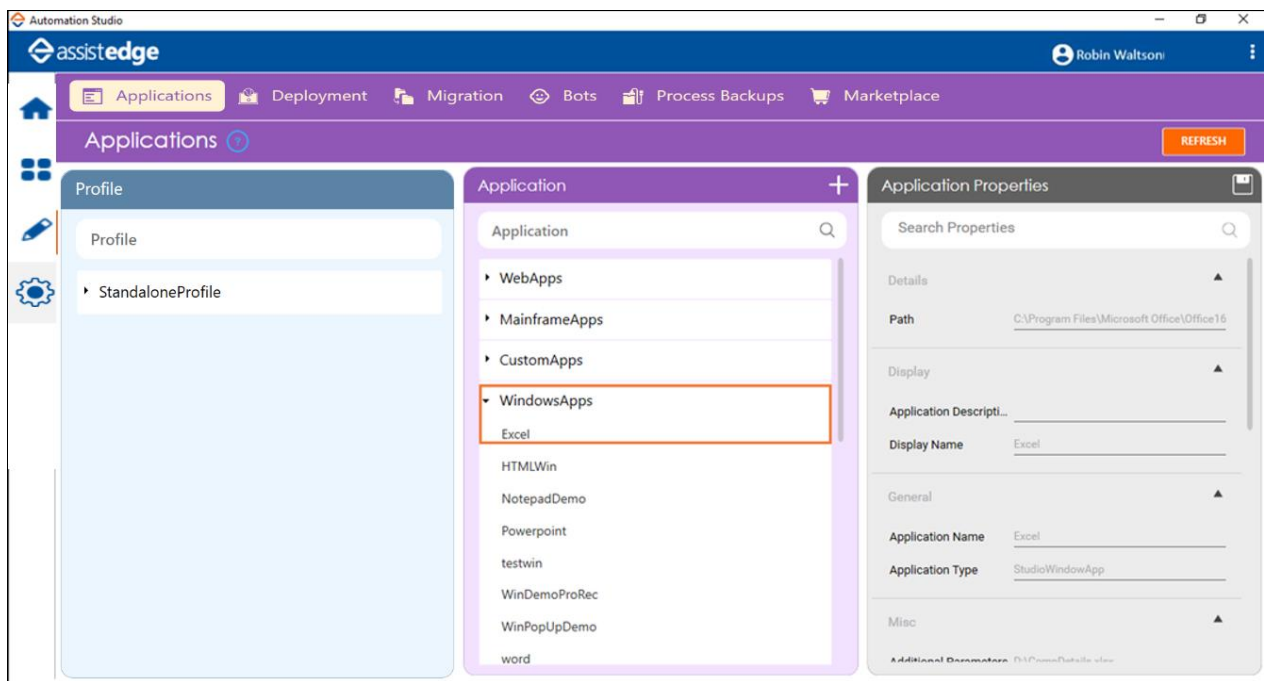
WindowsApps

You can perform process recorder activity on windows application, such as, word, PowerPoint, notepad.

To add windows application:

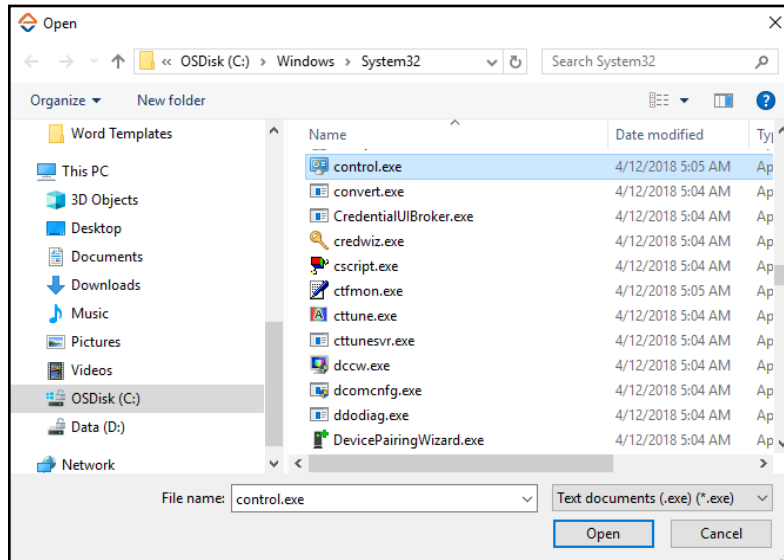
Below are the minimum required properties for configuring the intended windows application. If you want to define remaining properties, refer [WindowsApp](#) in the Admin Menu.

1. In the **Admin** menu, add an application of **Application Type- WindowsApps**. The **Add Application** dialog box appears.



2. In the **Application Name** field, enter a desired name of the EXE application you want to add. The name must not contain any special character or space.

3. Click **ADD** and browse for the windows application you want to configure. Click **Open**.



The WindowsApps application is configured.

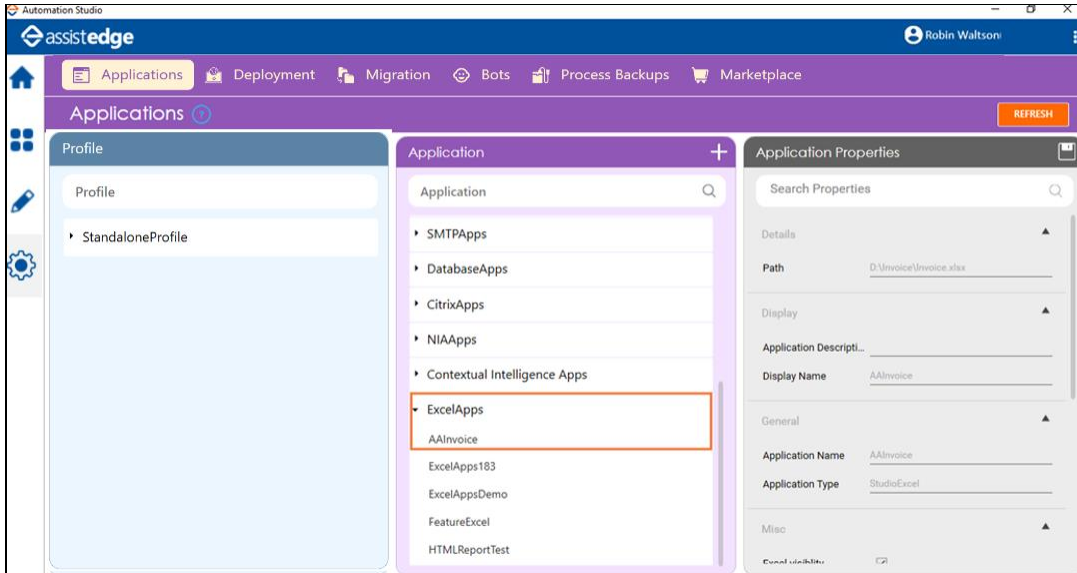
ExcelApps

You can perform process recorder activity on excel application.

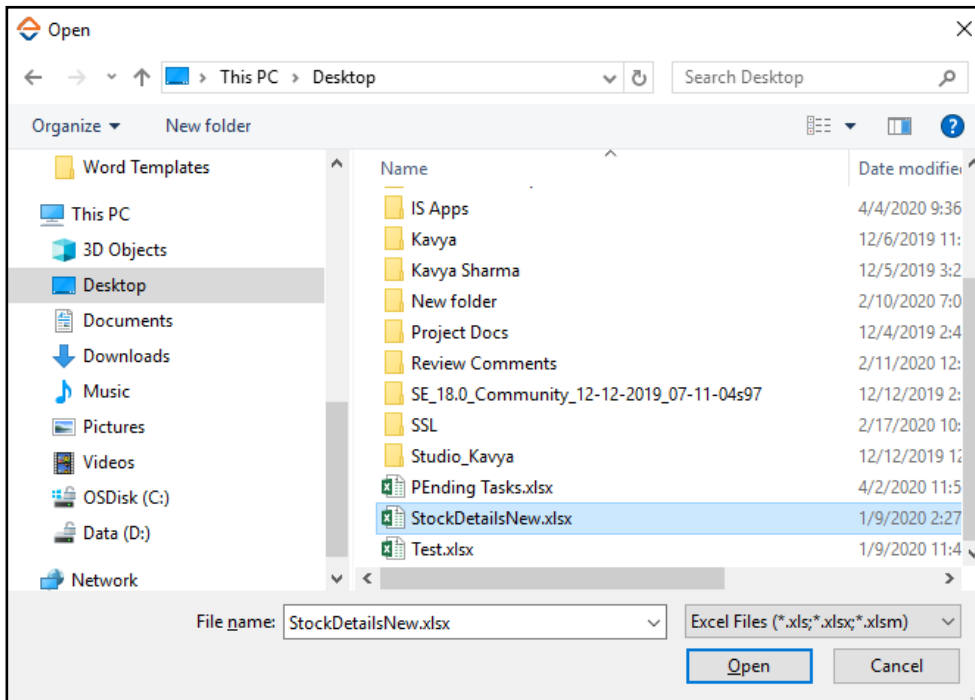
To add excel application:

Below are the minimum required properties for configuring the intended excel application. If you want to define remaining properties, refer [ExcelApps](#) in the Admin Menu.

1. In the **Admin** menu, add an application of **Application Type- ExcelApps**. The Add **Application** dialog box opens.



2. In the **Application Name** field, enter a desired name of excel application you want to add. The name must not contain any special character or space.
3. Click **ADD**. Navigate and select the excel file you want to configure. Click **Open**.



The ExcelApps application is configured.

Note: Make sure excel file is already created on your system.

WebApps

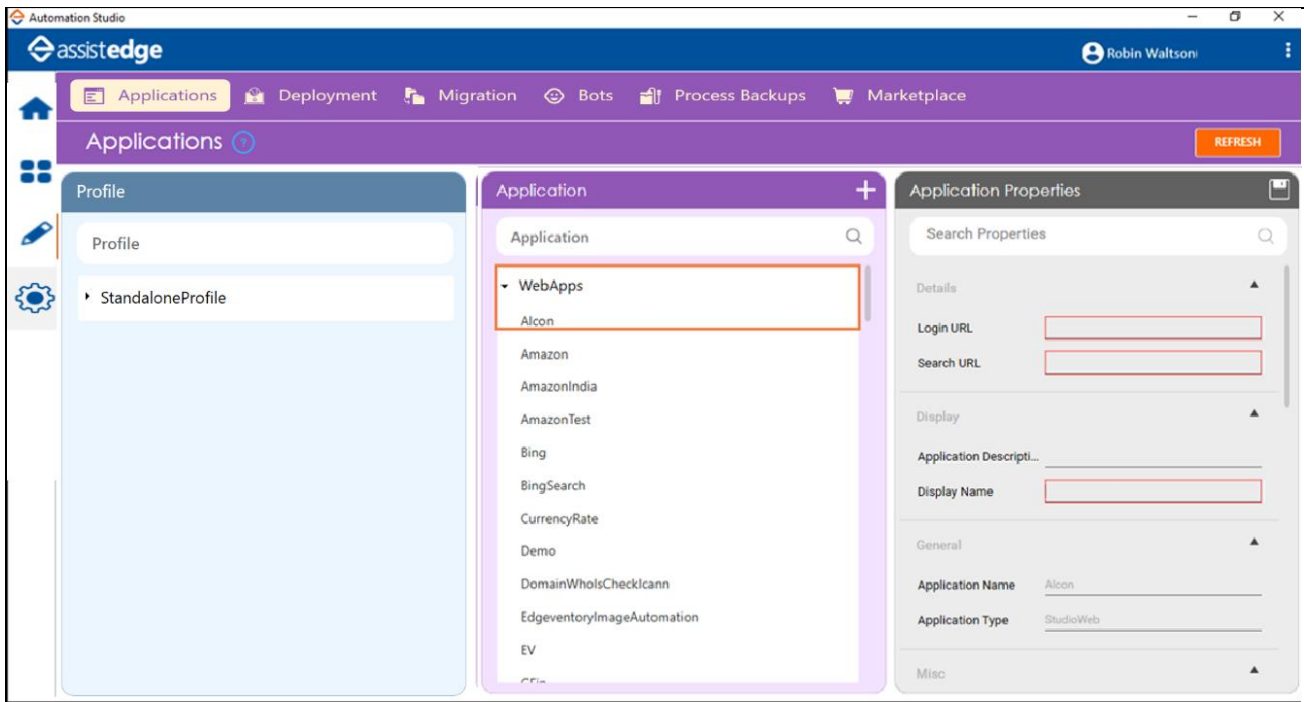
You can perform process recorder activity on web applications, such as, Internet Explorer, Chrome.


To add web application:

Below are the minimum required properties for configuring WebApps. If you want to define remaining properties, refer Add Application.

Below is the minimum required properties for configuring the intended web application. If you want to define remaining properties, refer [WebApps](#) in the Admin Menu.

1. In the **Admin** menu, add an application of **Application Type- WebApps**.

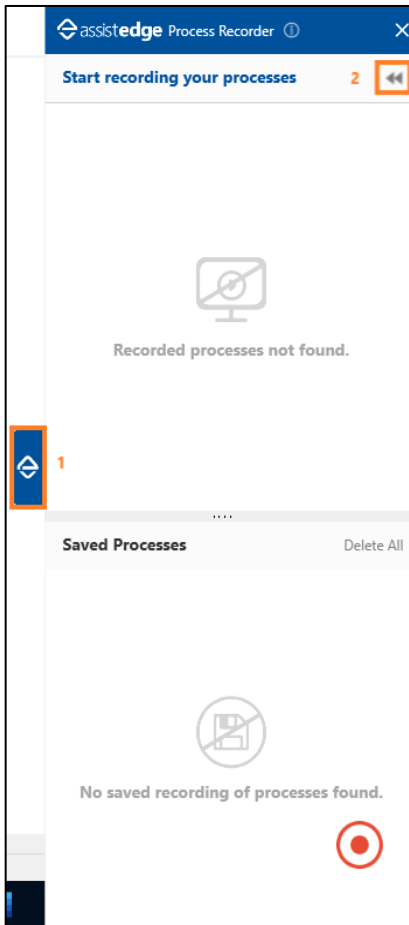


2. In the Application Properties panel:
 - In the **Login URL** field, enter the login URL of the web application you want to access.
 - In the **Search URL** field, enter the URL of the page where you want to perform the automation post login.
 - In the **Display Name**, enter a desired name of the web application.
3. Click the  (**Save Properties**) icon to save the application.


The WebApps application is configured.


9.19.2 Process Recorder Usage

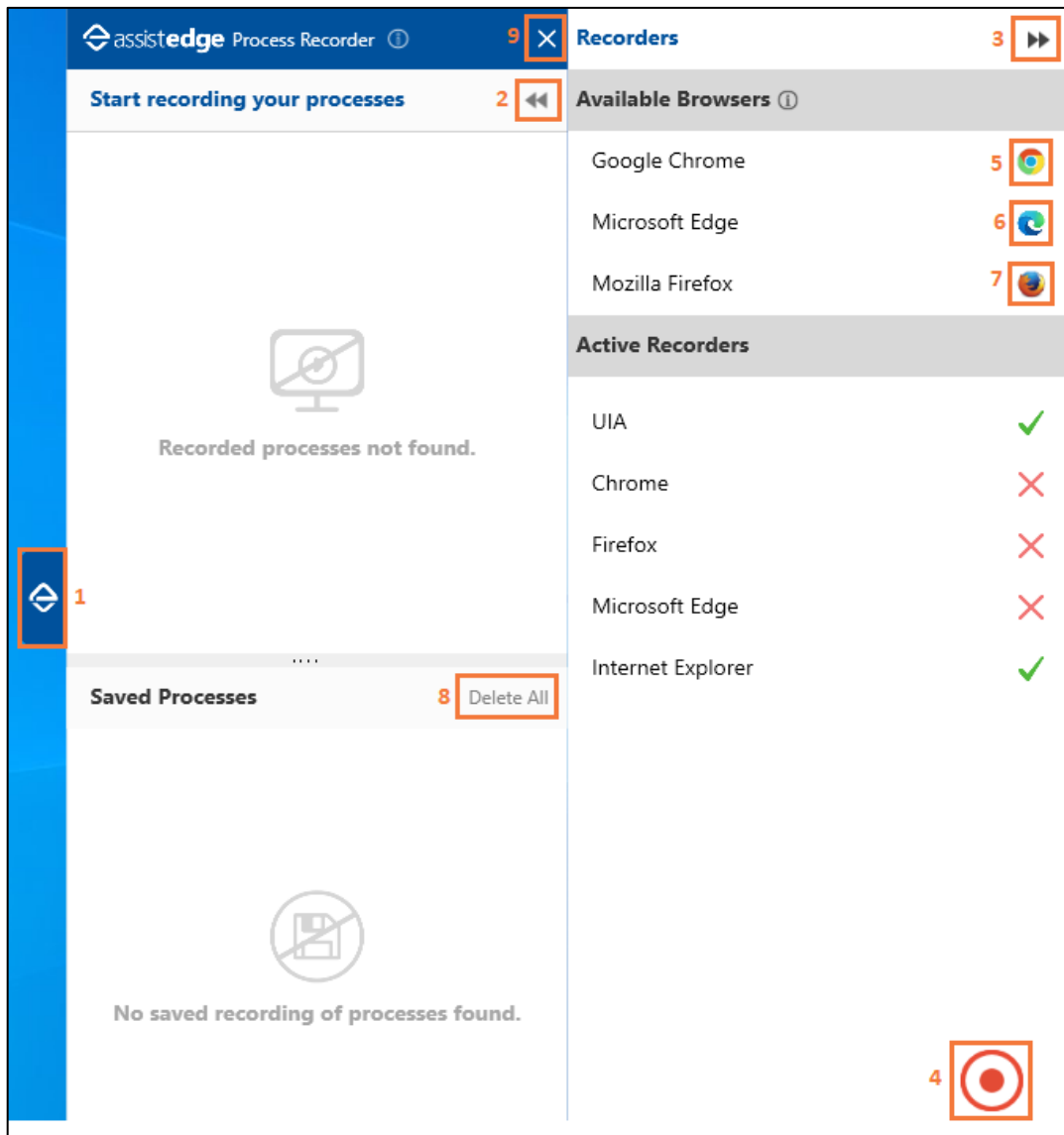
Click Process Recorder on the Canvas, a vertical tray appears on the right-hand side of the screen.







Following are the features of the process recorder tray:


1. Click the  (EdgeVerve) logo to minimize the window to the right side of the screen.

- Click the  (**Launcher**) icon to launch the complete process recorder tray. Initially, Start recording your processes and Saved Processes is launched.




- Click the  (**Collapse Settings View**) icon to minimize the Available Browsers and Active Recorders.
- Click the  (**Start Recording**) icon to start the process recorder for recording a process.
 - Once the **Start Recording** button is clicked, the button is turned into blue and in a pause sign.
 - Click the **Pause** button to pause the recording, it will go back to red and in a play sign.
 - Click the **Play** button to resume recording.
 - To stop recording, long hold the button.
- Click the  (**Google Chrome**) icon in the Available Browser list to launch and activate the browser. Once activated it will be listed as an active recorder as depicted for Internet Explorer. The green right icon indicates the browser is active.
- Click the  (**Microsoft Edge**) icon in the Available Browser list to launch and activate the browser. Once activated it will




be listed as an active recorder as depicted for Internet Explorer. The green right icon indicates the browser is active.

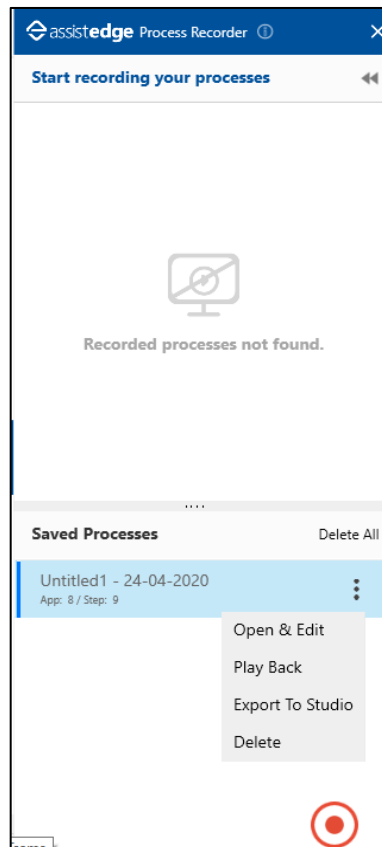
7. Click the  (**Mozilla Firefox**) icon in the Available Browser list to launch and activate the browser. Once activated it will be listed as an active recorder as depicted for Internet Explorer. The green right icon indicates the browser is active.
8. Click **Delete all** to delete all the saved processes.
9. Click **close** to close the process recorder tray and go back to the Automation Studio screen.

9.19.3 Using Process Recorder


1. Click **Process Recorder**, the tray opens.
2. Click the  (**Start Recording**) icon and start recording your process.

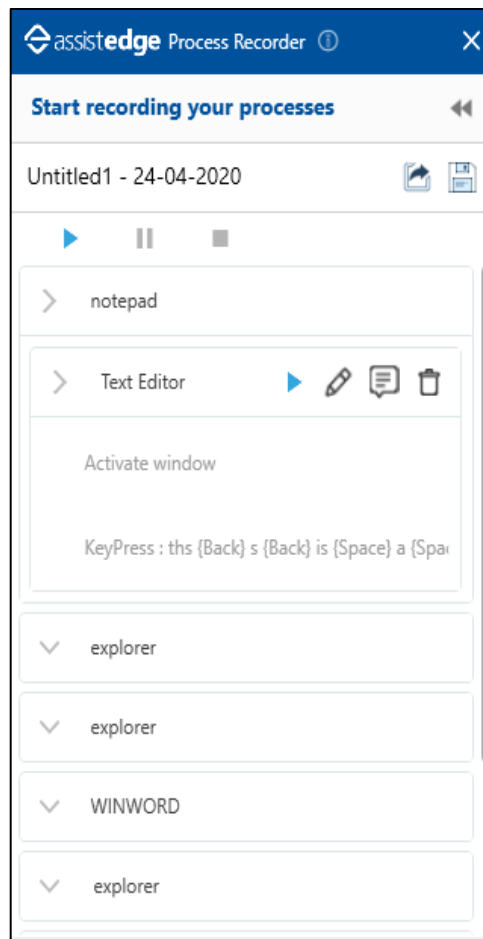
Note: Make sure required application for process recording are open on your system in the background before proceeding with the Start Recording.

3. Click the  (**Pause Recording**) icon to pause the recording and then click the  (**Start Recording**) icon to resume recording, if required.
4. Long hold the  (**Stop Recording**) button to stop process recording. Recorded Process is available in the **Saved Processes** section.











Following are the available options/features in the Saved Process section:

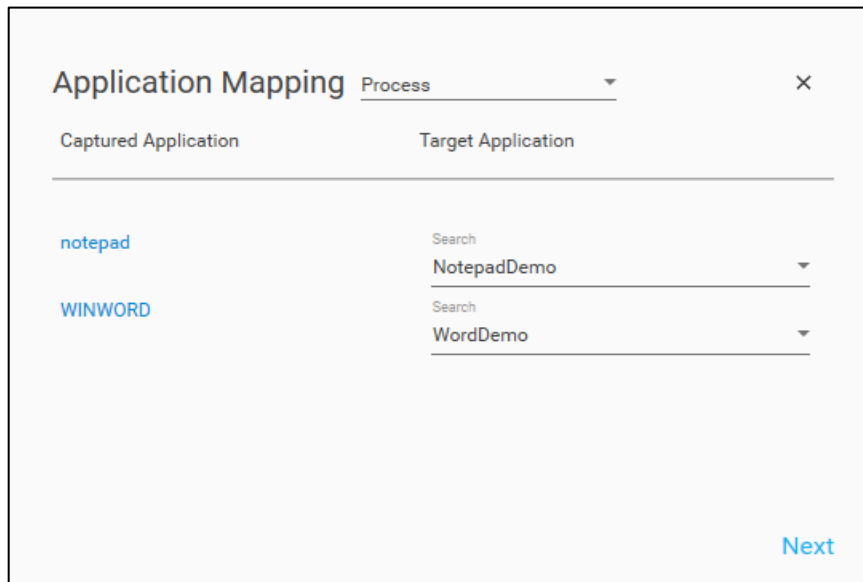
- Click the  (three dot) icon to view the available options.
 - Click **Open & Edit** to view the recorded process in the **Start recording your process** section.
 - Click **Play Back** to view the entire recorded automation.
 - Click **Export To Studio** to export the recorded process on the **Flowchart** designer area on the **Canvas**.
 - Click **Delete** to delete the recorded process.
 - Click **Delete All** to delete all the saved processes in the process recorder activity.
5. Click **Open & Edit** to open the recorded process and its step in the **Start recording your process** section.



Following are available options/features of Start recording your processes section on the process recorder tray:

- Double-click the **Name Field** to edit the process name.
- Click the  (**Save Process**) icon to save the changes performed in the recorded steps.
- Click the  (**Export To Studio**) icon to export the recorded process on the **Flowchart** designer area on the **Canvas**.
- Click the  (**Arrow**) icon to expand/collapse the recorded steps.

- Click the  (**Play Automation**) icon to view the automation of the recorded step.
 - Click the  (**Edit Semantic**) icon to edit the recorded step name.
 - Click the  (**Edit Annotation**) icon to add a footnote for the recorded step.
 - Click the  (**Delete**) icon to delete the recorded step.
6. Click the  (**Export To Studio**) icon to export the recorded process on the **Flowchart** designer area on the **Canvas**.
 7. The **Application Mapping** window appears. In this window, you need to map the Captured Application window to the Target Application window.

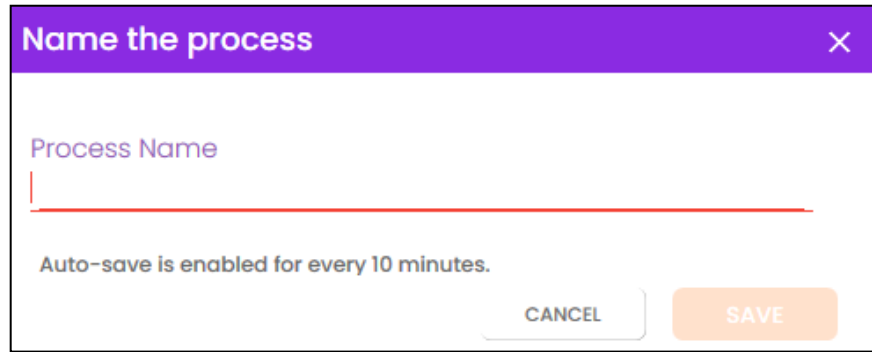


- a. Select the **Application Mapping** type from the list. There are three types of application mapping:
 - **Process** - Automation process workflow in Automation Studio forms the workflow where you can design a process you want to automate.
 - **SignIn Process** - A sign in process (also known as a log in process) is used to access an application upon authentication. In a typical scenario, a username and password are required to sign in to an application. You can use a SignIn Process to automate the repetitive task of signing in to an application.
 - **Reset Process** - With the help of a reset process, you can change the current state of the application involved in your process to the initial state of it.
- b. Map the Captured Application to the Target Application.
- c. Click **Next**.

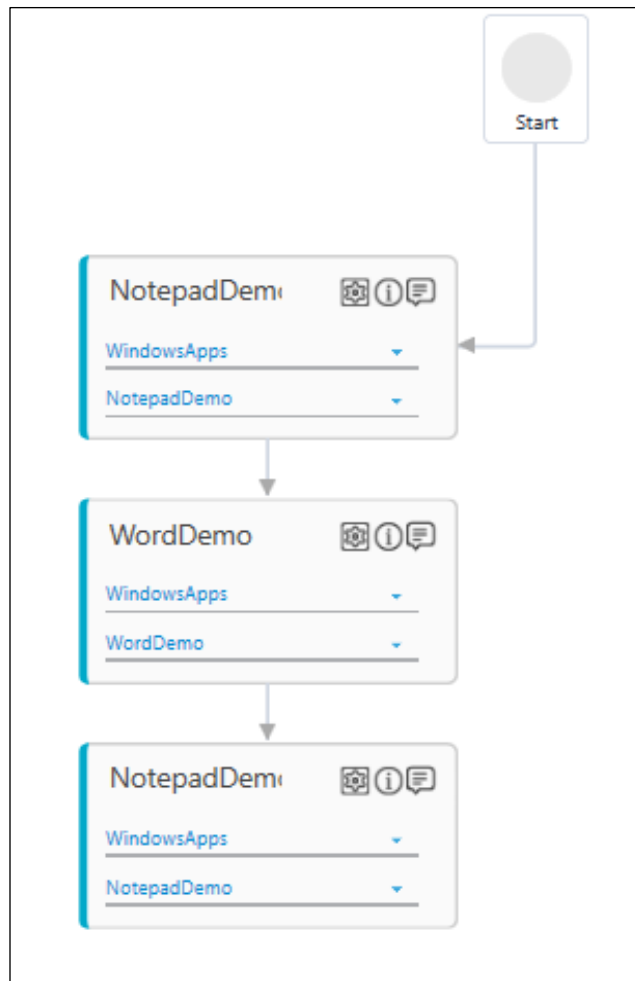
8. The **Field Mapping** window appears. In this window, you need to map the arguments/parameters to the **Name Field**.

Field Name	Data Type	Show to Agent	User Input
NotepadDemo			
inputField1	Text	<input type="checkbox"/>	<input checked="" type="checkbox"/>
inputField4	Text	<input type="checkbox"/>	<input checked="" type="checkbox"/>
WordDemo			
inputField2	Text	<input type="checkbox"/>	<input checked="" type="checkbox"/>
inputField3	Text	<input type="checkbox"/>	<input checked="" type="checkbox"/>

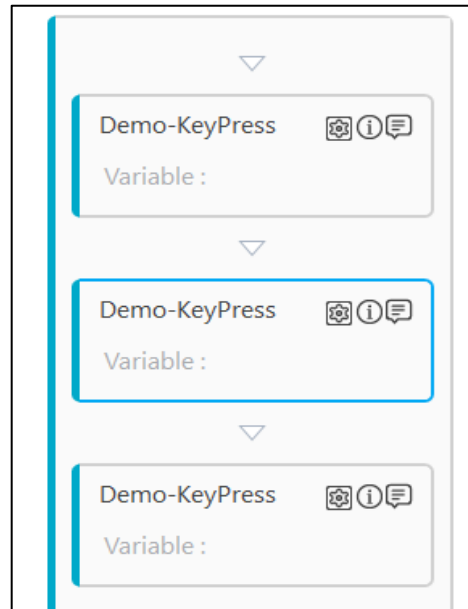
- a. **Name Field:** You can edit the Field Name as per your requirement.
 - b. **Data Type:** Select the Data Type for Field Name from the list.
 - **Text** – Select this if field names input is in string/text. For example, Student Name, Applicant Name.
 - **Numeric** – Select this if field name input is in whole number. For example, 6, 45.
 - **Decimal** – Select this if field name input is in decimal number. For example, 7.89, 1.34.
 - **Boolean** – Select this if field name input is either true or false.
 - **TableData** – Select this if field name input is in tabledata format.
 - **DataTable** – Select this if field name input is in datatable format.
 - **DateTime** – Select this if field name input is in date and time format.
 - c. **Show to Agent:** This option is selected when argument is copied while recording.
 - d. **User Input:** This option is selected when argument is typed or pasted while recording.
9. Click **OK**.
10. The **Name the process** appears. In the **Process Name** field, enter the name of the process of your choice and then, click **SAVE**.




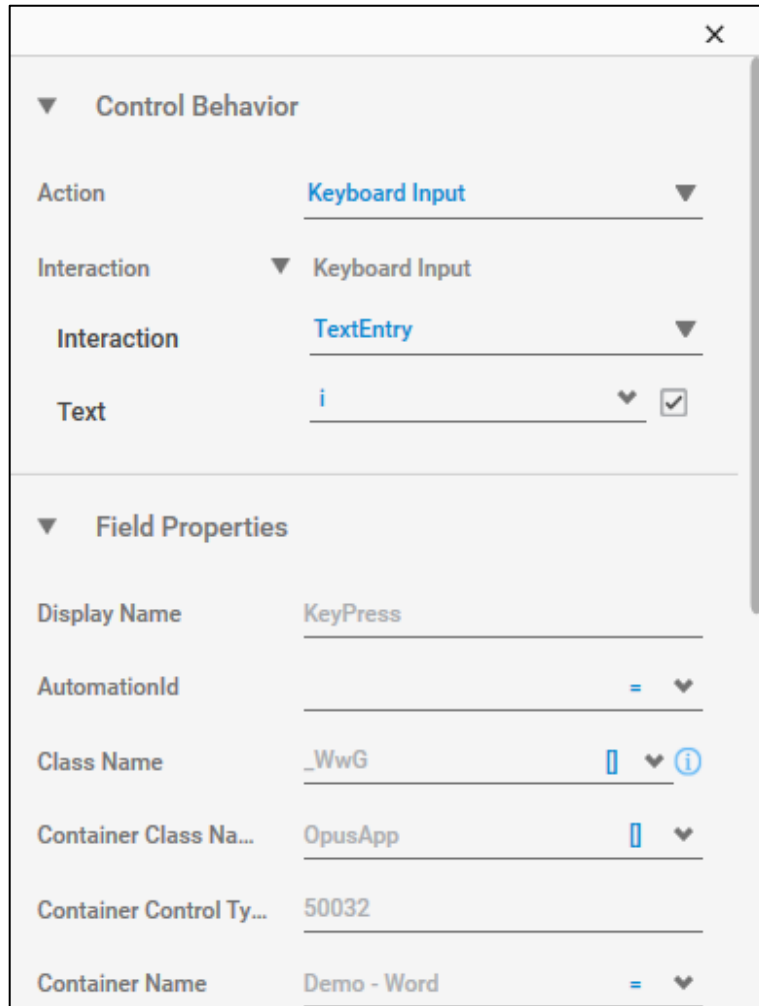
11. The recorded process is exported on the **Flowchart** designer area on the **Canvas**.



12. Additionally, you can view the configuration of the recorded steps and perform changes in the configuration as per your requirement to get the desired results. Double-click any exported application to drill down the steps.



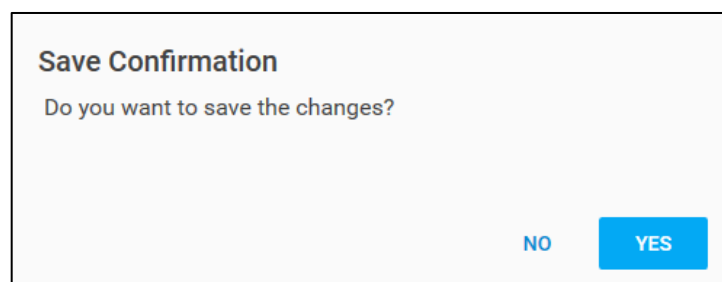
13. Click the  (**Settings**) icon to make changes in the captured interaction. In the **Action** list, you can change the type of the action on the recorder interaction. The **Field Properties** and **Misc Properties** may vary as per the selected action on the recorded interaction. Refer [Web Mode Properties](#) and [Windows Mode Properties](#), for Control Behavior, Field Properties and Misc Properties for more details.



The screenshot shows a configuration window with two main sections: Control Behavior and Field Properties. The Control Behavior section includes fields for Action (Keyboard Input), Interaction (Keyboard Input), another Interaction (TextEntry), and Text (i). The Field Properties section includes fields for Display Name (KeyPress), AutomationId, Class Name (_WwG), Container Class Name (OpusApp), Container Control Type (50032), and Container Name (Demo - Word).

Note: You can configure any new interaction in the process recorder activity without performing re-recording of the business process. Refer [Web Application](#) and [Windows Application](#) for more details.

14. Click the **X (Close)** icon. A confirmation message appears.
15. Click **YES** to save the changes.




Save Confirmation
Do you want to save the changes?

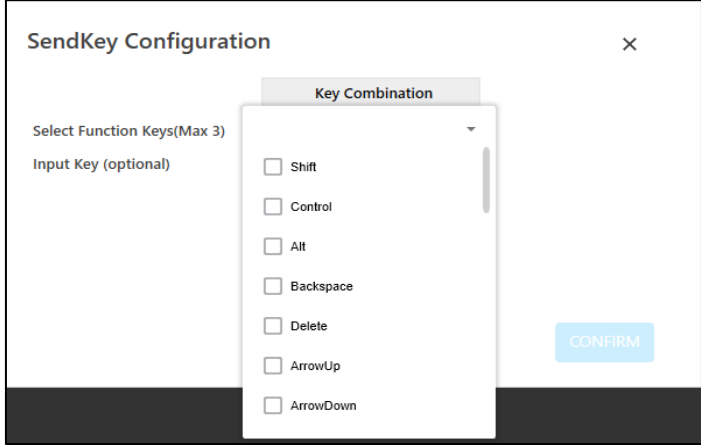
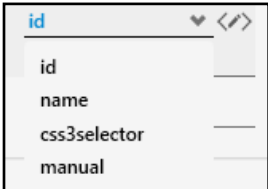
NO YES


9.19.4 Web Mode Properties


The properties of Web Mode are listed in the following table:

Property	Usage
Control Behavior	
Action	<p>In the Action list, Select the Interaction.</p> <ul style="list-style-type: none"> ▪ Control Exists - Checks if the UI element exists at the required area. It is useful when working with dynamics controls. ▪ Wait Until Exists - Waits till the UI element is found. Using this action avoids usage of delay in between the activity to address page loading time. This interaction returns a Boolean value of true or false on success or failure of finding the image. The Boolean value can be set to a variable and checked in a conditional statement to perform an appropriate action after finding the image. ▪ Click - Allows to left click the selected UI element. ▪ Open Context Menu - Use this option, in case the control opens a context menu on click. ▪ Double Click - Performs a double clicks the selected UI element. ▪ Get Value - Extracts the current value of the selected UI element such as input, select, textbox. ▪ Get Attribute - Extracts the value of the attribute of the selected UI element. ▪ Get Text - Extracts the value of the attribute of the selected UI element. ▪ Mouse Over - Hovers the mouse over the selected UI element. ▪ Scroll By - Scrolls to particular section of the web page horizontally or vertically. ▪ Scroll Into View - Scrolls to bring the application area viewable on screen of the web browser. <p>Set Value - Sets the user provided value of the attribute of the UI element and stores it as the value of the parameter. You must define the parameter in the Parameter bar before you start capturing the element.</p> <ul style="list-style-type: none"> ▪ Set Attribute - Sets the user provided attribute for the specified UI element and stores the provided value of the attribute into a parameter. You must define the parameter in the Parameter bar before you start capturing the element.

Property	Usage
	<ul style="list-style-type: none"> ▪ Send Keys - Captures UI element input from the keyboard in the selected UI element. The input can be a combination of keys or text input from the keyboard. Use this option when a keystroke needs to be automated on a specific web element and not on the overall application. ▪ Shift Click - Opens the selected URL in a new window. It appears only when you want to capture action on a button or a link.
Interaction	The corresponding interaction set against the selected action
Appears if Action selected is Scroll By.	
XAxis	Vertical scroll on the web page. The value must be entered in pixels.
YAxis	Horizontal scroll on the web page. The value must be entered in pixels.
Appears if Action selected is Scroll Into View.	
Scroll Tos	<p>Determines the area till where the scroll must be done. Available options are:</p> <ul style="list-style-type: none"> ▪ Start ▪ Center ▪ End <p>Note: Internet Explorer does not support Centre option, so configuring center for ScrollTo displays exception at runtime.</p>
Appears if Action selected is Set Value, Set Attribute or Get Attribute.	
Attribute Name	Sub attributes or the properties of the html attribute selected.
Clear Existing Data	Select this option to clear the existing value.
Appears if Action selected is Set Value, Set Attribute.	
Input Source	The input value that must be provided. You must define a parameter in the Parameter bar to use this option. Select the checkbox beside Input Source and enter the input value, if you want to set a default value.
Appears if Action selected is SendKeys.	
Keys	<p>The key sent as input from the keyboard. It can be a combination of keys or a text.</p> <p>To configure the keys:</p> <ol style="list-style-type: none"> 1. Once Action selected is SendKeys, click the  (Configure) icon. The SendKey Configuration dialog box appears.

Property	Usage
	<div style="text-align: center;">  </div> <ol style="list-style-type: none"> 2. In the Select Funtion Keys(Max 3) list, select the keys from the keyboard to create the key combination. You can select upto maximum of three keys. 3. In the Input Key(optional) filed, enter an alphabetical or a numeric key to create the combination. This is an optional field. 4. Click CONFIRM. <p>The configuration for SendKeys action is done.</p> <p style="text-align: center;">Appears if Action selected is Shift Click.</p>
Dock Window	When a new browser window is opened on click of a link or a button on the web page, this button enables the new window listed in the multimodal UI. You can select one of the windows in the multimodal UI to configure controls on the intended window.
Tab Name	Name of the new window when it is docked in Engage application.
Target URL	URL of the new window that opens when Shift Click is selected in Action .
Field Description	
Control Tag	User specified identifier of the selected UI element.
Control Type	Type of the UI element selected.
Display Name	User specified display name of the action configured.
Find By	<p>Locator of the UI element on a web application.</p> <p>Identifying the correct element forms one of the first step towards creating an automation workflow. This isit changes depending on the properties of the UI elements defined. Commonly available options are:</p> <div style="text-align: center;">  </div>

Property	Usage
	<ul style="list-style-type: none"> ▪ Id - Identifier of the UI element. It is useful for locating elements that has a unique Id associated with it. ▪ name - Field name of the control. Each element on the web page has a name assigned to it which can be used to select the intended element. ▪ css3selector - Part of the CSS rule that uses selectors such as tags, class, pseudo-element, pseudo-class, combination of selectors and other wide range of selectors to identify the UI element. ▪ css3xpath - Depending on the html tagging, css3xpath can also be available in this list. It uses the Xpath expression to locate the UI element. ▪ manual - Provides manual way of identifying the UI element. You can use this option when the selected set of properties available for the UI element are not able to identify the required it at automation runtime. <p>You can configure backup selectors to avoid situations where the automation process is unable to find the primary selector. This is not a mandatory step. Follow the instructions to configure the backup selectors:</p> <ol style="list-style-type: none"> 1. Click the  (Configure) icon. The Configure Selectors dialog page appears. <div data-bbox="539 1171 1205 1562" style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <div style="display: flex; justify-content: space-between;"> Configure Selectors × </div> <div style="margin-top: 10px;"> <p>Order Selectors</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> id <input type="checkbox"/> name <input type="checkbox"/> css3selector <input type="checkbox"/> manual </div> <div style="margin-top: 10px;"> <p>Value</p> <hr style="width: 80%; margin-left: 20px;"/> </div> <div style="text-align: right; margin-top: 10px;"> CONFIRM </div> </div> 2. Select the additional selectors. <ul style="list-style-type: none"> • If the selector is manual:

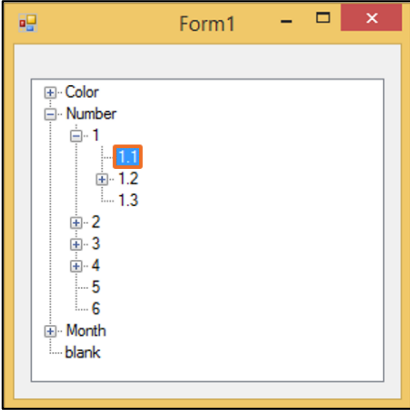
Property	Usage
	<div data-bbox="532 300 1211 695" data-label="Image"> </div> <p>a. Click the  (pencil) icon to configure the manual selector. The Configure Selector dialog box appears.</p> <div data-bbox="521 831 1224 1245" data-label="Image"> </div> <p>b. In the Select Attribute list, select the attribute of the manual selector. As per the HTML structure and the properties of the UI element defined, the attributes are displayed in this list.</p> <p>c. In the Value list, either select a parameter containing the value of the selected attribute or select the check box to set a default value.</p> <p>d. In the Condition list, select the condition of the selector. Available options are- Equals, Contains, Starts With and Ends With.</p> <p>e. The Generated Selector field displays the default value generated, if any. You can click Add to add this value in the Selector field.</p> <p>f. Click CONFIRM.</p> <p>3. In the Value field, the captured selector is displayed.</p> <p>4. You can drag the selector and drop it to change the sequence as per the priority of the selector you want to set. At the time of process execution, the selector is considered as per the sequence set, the top most having the highest priority while the the last one having the least priority for locating the</p>

Property	Usage
	UI element. 5. Click CONFIRM to save the configuration.
Appears if Action selected is Control Exists, Wait Until Exists, Get Value, Get Attribute, Get Text	
Tab Number	The window tab on which you want to perform automation.
Variable Name	User defined name of the variable that stores the captured value.
Variable Type	Type of the variable defined. Available options are: <ul style="list-style-type: none"> ▪ Text - A sequence of character, either as a constant or a variable. ▪ Numeric - An integer type that range from negative through positive. ▪ Decimal - An exact numeric value defined by its precision and scale. ▪ Boolean - A data type used for making decision. Can have only two values- true or false. ▪ Data Table - A data type that stores tabular data.
Misc Properties	
FrameNo	Number of the frame in a web page where you want to perform the action. Automation Studio automatically detects the frame number based on the area selected during configuration.
Item Index	Index number of the selected item on the web page.
Item Offset	The distance between the target item and the anchor.

9.19.5 Windows Mode Properties

The properties of Windows Mode are listed in the following table:

Property	Usage
Control Behavior	
Action	In the Action list, Select the Interaction. <ul style="list-style-type: none"> ▪ Click - Allows to left click the selected UI element. ▪ Select - Selects a UI element. ▪ Get Text - retrieves text from the selected UI element. a. Set Text - sets user provided text input to the selected UI element and stores it into a parameter. You must define the parameter in the Parameter bar before you start capturing the UI element. ▪ Get Data From Table - Reads complete data of a DataGrid Table control to a

Property	Usage
	<p>DataTable and then if required to an excel/CSV file. Following are the steps involved to read the data.</p> <ol style="list-style-type: none"> b. Create an argument of type System.Data.DataTable, before extracting the UI element. c. Click WINDOWS MODE to extract the Data Grid Table control. d. Select Get Data from the Table. e. Set the Variable Name as the argument created in the first step. f. Save the configuration. g. Use Export Data Table activity to write data to excel <ul style="list-style-type: none"> ▪ Select Tree Node - Selects a node of a tree control using this interaction. Specify the input in a specific format. For example, for selecting node 1.1, enter the input value as Number/1/1.1.  <ul style="list-style-type: none"> ▪ Expand Tree Node - Expands a node of the tree control. ▪ Select RadioButton – Selects a radio button using this interaction. ▪ Get RadioButton Status - Fetches the radio button status. It returns true or false. ▪ Get ComboBox Value - Reads the currently selected combo box value. ▪ Perform Combobox Select Value - Selects the specified value from the drop-down list. ▪ Expand/Collapse - Allows to expand or collapse the selected UI element. ▪ Toggle CheckBox Action - Toggles the current state of a checkbox. ▪ Get CheckBox Status - Returns the current selection status of a checkbox. ▪ Set DataGrid Value By Index - Sets the value to a DataGrid cell based on its index. ▪ Get DataGrid Value By Index - Retrieves data grid value by its index. Specify the cell index from which the input is needed. ▪ Scroll Horizontal - Performs horizontal scroll based on the input provided on the

Property	Usage
	<p>percentage scale from 0 to 100.</p> <ul style="list-style-type: none"> ▪ Scroll Vertical - Performs vertical scroll based on the input provided on the percentage scale from 0 to 100. ▪ Mouse Click - allows mouse simulation. It requires the application to be in focus. You can choose to perform Left Click, Right Click or Double Click available from the drop-down list. ▪ Keyboard Input - captures UI element input from the keyboard in the selected UI element. This interaction allows performing different keyboard input combinations with the help of Single Key, Double Keys, and Triple Keys and TextEntry options. ▪ Set Focus - brings the UI element in focus before performing any action on it. ▪ If Control Exists - checks availability of the UI element on the windows application. It is used while working with dynamic controls. ▪ Legacy Interaction - Allows to configure UI element of the Windows application which supports a legacy pattern.
Interaction	The corresponding interaction set against the selected action.
Appears if Action selected is Horizontal Scroll or Vertical Scroll .	
Expand/Collapse	Performs horizontal/vertical scroll based on the input provided on the percentage scale from 0 to 100.
Percent	Allows to expand or collapse the selected UI element.
Appears if Action selected is Keyboard Input and Interaction selected is TextEntry .	
Text	Allows user input as a text.
Appears if Action selected is Mouse Click .	
Mouse Action	<p>Allows user input using the mouse. Available options are:</p> <ul style="list-style-type: none"> ▪ LeftClick - Performs a left click specified at the offset point. ▪ DoubleClick - Performs a double click specified at the offset point. ▪ RightClick - Performs a right click specified at the offset point.
Appears if Action selected is Get DataGrid Value by Index or Set DataGrid Value By Index .	
Column Number	Specify the cell index from which the input is needed.
Row Number	Specify the cell index from which the input is needed.
Appears if Action selected is Keyboard Input and Interaction selected is DoubleKeys or TripleKeys .	
Key	The single key set as input from the keyboard.
Key	The second key that is set as input from the keyboard along with the first key.

Property	Usage
Key	The third key that is set as input from the keyboard along with the first and the second key.
Field Description	
Display Name	User specified display name of the windows element selected.
AutomationId	Unique identifier for the automation element in the automation tree. See Handling Dynamic Controls section to know more about the usage of this field.
Class Name	The class name of the UI element as defined by the UI element developer. See Handling Dynamic Controls section to know more about the usage of this field
Container Class Name	The container class name of the UI element as defined by the developer. See Handling Dynamic Controls section to know more about the usage of this field.
Container Control Type	The control type of the container of the UI element as defined by the UI element developer.
Container Name	The container name of the UI element as defined by the developer. See Handling Dynamic Controls section to know more about the usage of this field
Control Index	The index of the current UI element within the entire set of elements with the same values of automation Id, class name and UI element name as that of the current element.
Control Name	The control name of the UI element as defined by the UI element developer. See Handling Dynamic Controls section to know more about the usage of this field.
Control Type	The control type of the UI element as defined by the developer.
Search By Control Order No.	Signifies if the UI element needs to be searched based on the UI element order number. Control order number is the index of the current element within the entire set of elements with the same value of UI element Id as that of the current element.
Control Order No.	Control order number is the index of the current element within the entire set of elements with the same value of UI element Id as that of the current element.
Appears if Action selected is Get Text .	
Variable Name	User defined name of the variable that stores the captured value.
Variable Type	Type of the variable defined. Available options are: <ul style="list-style-type: none"> ▪ Text - A sequence of character, either as a constant or a variable. ▪ Numeric - An integer type that range from negative through positive. ▪ Decimal - An exact numeric value defined by its precision and scale. ▪ Boolean - A data type used for making decision. Can have only two values- true or false. ▪ Data Table - A data type that stores tabular data.
Appears if Action selected is Mouse Click .	

Property	Usage
Offset From Control	It is the distance between the UI element and the anchor.
Misc Properties	
Max wait for input idle	The maximum time interval in milliseconds for which the windows plugin waits on launch before it starts executing the configured interactions.
Retry Count	Number of times, the plugin retries to find a UI element, if not found.
Retry Interval (ms)	The time interval in milliseconds for which the plugin waits before it attempts for a retry.

9.19.6 Process Recorder Properties

The properties of Process Recorder activity are listed in the following table and can be edited in the Properties grid on the right pane.

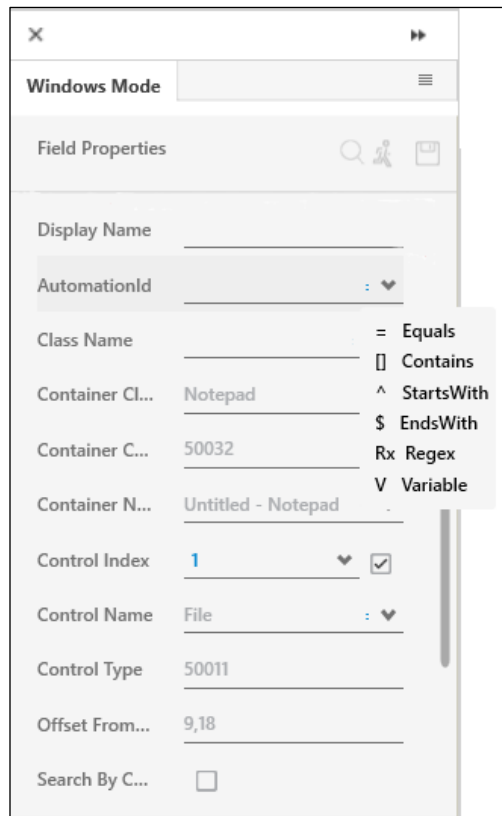
Property	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After (ms)	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before (ms)	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the Flowchart designer area. By default, the name is set as Target Application Name . You can change the name as required.


9.19.7 Handling Dynamic Controls

Automation Studio provides different identification criteria and Parametrization of the value of the dynamic field properties of the UI elements. You can reconfigure some of the auto captured field properties of such UI elements to create a more effective way of identifying the indented UI element. For such controls, remove the dynamic part of the string and use an appropriate option for UI element identification. For example, if a Class Name field is recognized as AssistEdge_Studio_20180205083009, the later part of the string, that is a timestamp, is dynamic and can be removed. Retain only the static part and reconfigure the field properties using the suitable option.

Following are the different field properties that can be reconfigured against their respective values:

- AutomationId
- Class Name
- Container Class Name
- Container Name
- Control Name



Use the  (drop down arrow) to reconfigure the identification criteria. Available options are:



- **Equals** - Identifies the UI element based on the exact match with the value of the selected field property.
- **Contains** - Identifies the UI element based on the string present in the value of the selected field property.
- **StartsWith** - Identifies the UI element based on the starting string of the value of the selected field property.

- **EndsWith** - Identifies the UI element based on the ending string of the value of the selected field property.
- **Regex** - Identifies the UI element with the string matching the value of the selected field property as per the identification pattern defined through the provided regular expression such as a.b, *txt and others.
- **Variable** - Identifies the UI element as per the parametrization of the value of the selected field property. Parametrization allows to run the identification process over and over again using different values. With parameterization, windows controls with looping and assignment activities, are used.




9.19.8 Step-By-Step Guide to Use Process Recorder to Print the Population of a City in Notepad From a Web page



Let's create an example to automate a process to print the Pune, India population in notepad from a webpage.

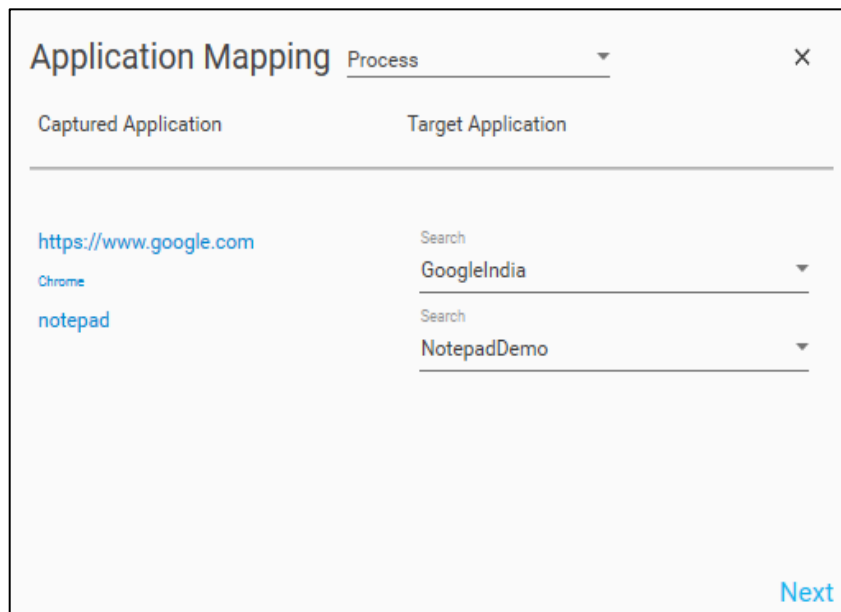
Prerequisite:

1. In the **Admin** menu, add an application of **Application** type - **WebApps**.
2. In the Application Properties panel:
 - In the Login URL and Search URL field, enter `https://www.google.com/`.
 - In the Display Name field, enter GoogleIndia.
 - In the Preferred Browser list, select Chrome.
3. Click the  (**Save Properties**) icon to save the application details.
4. In the **Admin** menu, add an application of **Application** type - **WindowsApps**.
5. In the Application Properties panel:
6. In the **Path** field, enter the exe file location of the notepad. For example, `C:\Windows\System32\notepad.exe`.
7. In the Display Name field, enter NotepadDemo.
8. Click the  (**Save Properties**) icon to save the application details.
The web and windows applications are created.

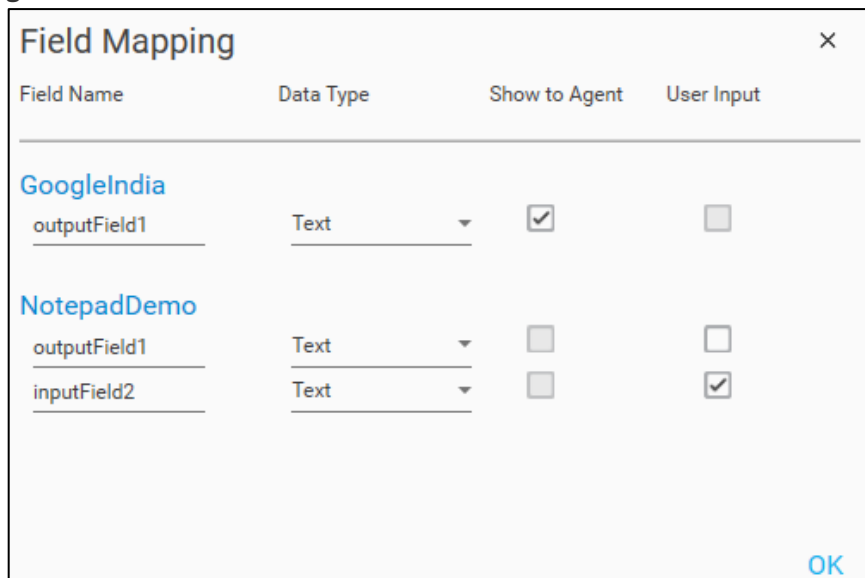
To create a process with Process Recorder activity:

9. Open Notepad.
10. In the **Canvas Tools** panel, Click Process Recorder.
11. Click the  (**Launcher**) icon to expand the Process Recorder tray.
12. Click the  (**Google Chrome**) icon to launch and activate the browser.
13. Click the  (**Start Recording**) icon to record the process.
14. In the browser address bar field, enter `www.google.com`.
15. In the search field, enter **Pune Population 2020** and press enter.

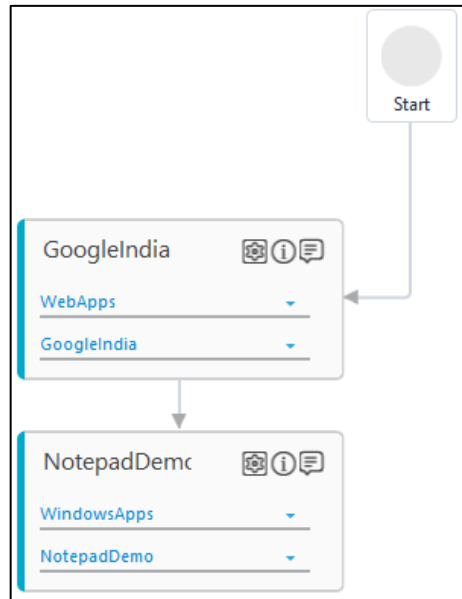
16. Copy the Pune population and paste it in the **Notepad**.
17. Enter **Pune Population 2020** in the Notepad after the pasted text.
18. Long hold the  (**Pause Recording**) icon to stop the recording.
19. Click the  (**Export To Studio**) icon to export the recorded process to the Automation Studio.
 - In the **Application Mapping** window, perform the following steps:
In the **Application Mapping** list, select Process.
 - In the Captured Application and Target Application column, for https://www.google.com chrome, select GoogleIndia and for notepad, select NotepadDemo.
20. Click **Next**.



21. In the **Field Mapping** window, click OK.



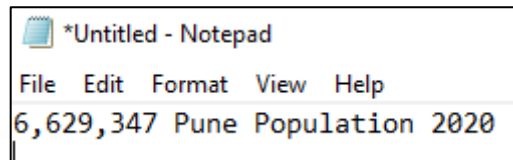
The process is exported to the **Flowchart** designer on the **Canvas**.



22. Save the process.

23. Set up the environment and perform test run.

The output is displayed in the Notepad.



9.20 Peoplesoft

Automation of tasks on Peoplesoft 9.2 on its FluidUI is possible. Automation on Peoplesoft is treated in the same way as any other web page.

A web application needs activity to be created on RPA with the Peoplesoft URL with a preferred browser and used as a regular web application.

9.21 Siebel

Oracle's Siebel CRM tool comes in two variants:

- High Interactivity (HI)
- Open UI.

The HI variant of Siebel CRM renders the user interface as a collection of HTML frames and uses ActiveX.

Launching and signing in to Siebel applications is done using Studio's existing SignIn Process functionality with the Siebel HI web application.



AssistEdge provides capability to support tool based automation of Siebel CRM High Interactivity application to configure a business flow.

Following version of Siebel HI is supported:

- Base Release Version: Siebel CRM 8.1.x / 8.2.x
- Application Version: 15.8.0.0 SIA
- Schema Version: 50.13.13.0

The web enhancements in AssistEdge is capable of supporting tool based automation of Siebel CRM Open UI application to configure a business flow.

The following components in Automation Studio's Canvas Tools support Siebel HI tool based automation.

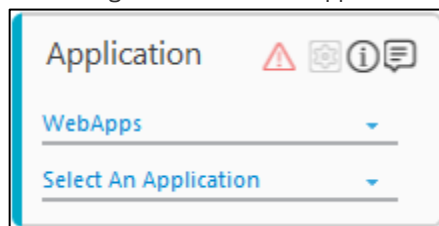


9.21.1 Navigation

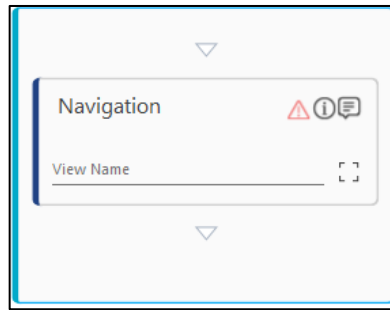
It helps to automate navigation to screen tabs, link bars, group of views and group of Detail views of the Siebel HI application using the name of the view. Siebel HI UI provides control for tabs and link bars; hence, using the screen view helps to navigate to the desired location within the Siebel CRM application.


Using Navigation Activity

1. In the **Canvas Tools** pane, click **Process Components** to expand the tool and view the associated activities.
2. Drag the **Application** activity and drop on to the **Flowchart** designer on the **Canvas**.
3. In the **Application Type** list, select WebApps.
4. In the **Select An Application** list, select the configured Siebel CRM application.



5. Double-click the **Application** activity to open the activity block.
6. In the **Canvas Tools** pane, click **Siebel** to expand the tool and view the associated activities.
7. Drag the **Navigation** activity and drop inside the **Application** activity. The validation error symbol disappears when the required inputs are provided.



8. Navigate to the desired view in the docked Siebel CRM application in Automation Studio, and then click the  (**Get View Name**) button to automatically fetch the view name. If the application does not open or the view name is not retrieved, enter the view name manually in the **View Name** field.

Note: To know the view name, navigate to the required screen of the Siebel CRM application manually, and then access Help > About View > View name.

The Navigation activity to navigate to the desired Siebel CRM screen is created.

Navigation Activity Properties

The properties of a Navigation activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	

Property Name	Usage
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the Flowchart designer. By default, the name is set as Navigation . You can change the name as required.

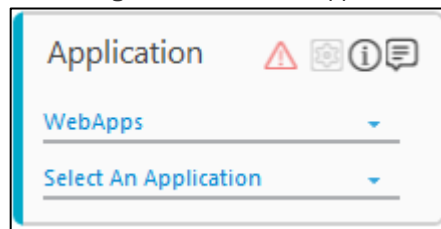
9.21.2 Siebel Operations

It helps to automate Siebel operations performed on a business component such as insert, delete, update, query and so on.

Note: You must first navigate to the applet screen where you want to perform the Siebel operation. You can use the Navigation activity to navigate to the required screen.

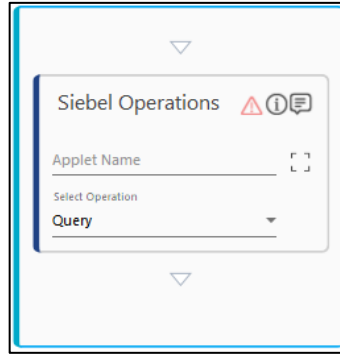
Using Siebel Operations Activity


1. In the **Canvas Tools** pane, click **Process Components** to expand the tool and view the associated activities.
2. Drag the **Application** activity and drop on to the **Flowchart** designer on the **Canvas**.
3. In the **Application Type** list, select **WebApps**.
4. In the **Select An Application** list, select the configured Siebel CRM application.



5. Double-click the **Application** activity and navigate to the desired applet. You can use the **Navigation** activity for the required navigation. See the [Use Case](#) for better understanding.
6. In the **Canvas Tools** pane, click **Siebel** to expand the tool and view the associated activities.

7. Drag the **Siebel Operations** activity and drop inside the **Application** activity. The validation error symbol disappears when the required inputs are provided.



8. Click the  (**Get Applet Name**) button to automatically fetch the applet name. If the required applet name is not retrieved, enter the applet name manually in the **Applet Name** field.

Note: To know the applet name, navigate to the required Siebel CRM application screen manually and then access Help > About View > Applets name.

On the Siebel CRM screen, multiple applets can be present. You must choose the applet where you want to perform the automation.

9. In the **Select Operation** list, select the Siebel operation you want to perform in the specified applet. It displays a list of commonly used Siebel operations. By default, **Query** is selected. Available options are:
- **Query** - Searches the specified record.
 - **Go** - Clicks the Go button of the specified grid or applet.
 - **New** - Clicks the New button of the specified applet.
 - **Copy** - Copies the selected record in the grid.
 - **Add** - Adds a new record (row) in the specified grid.
 - **Ok** - Clicks the Ok button of the specified applet.
 - **Delete** - Deletes the selected record from the specified grid.
 - **New Task** - Clicks the Next button of the specified applet.
 - **Previous Task** - Clicks the Previous button of the specified applet.
 - **MoveNextRow** - Moves and selects the next record in the set of records filtered in the grid.
 - **Expand Tree** - Expands the specified tree applet. Once the tree is expanded, use the [Set Field Value](#) activity to write and the [Get Field Value](#) activity to read the desired value of columns available in the expanded tree.
 - The **Siebel Operations** activity is created.

Siebel Operations Activity Properties

The properties of a Siebel Operations activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the Flowchart designer. By default, the name is set as Siebel Operations . You can change the name as required.

9.21.3 Drilldown

It helps to automate drilling down a field in the selected grid of the applet and view the related details.

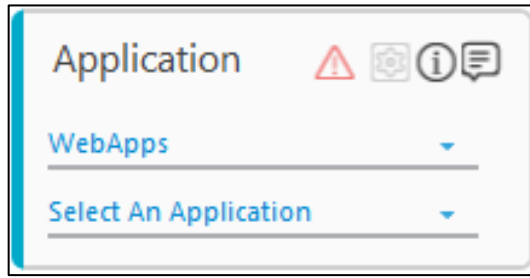
It clicks the specified field name of the top row of the grid; hence, use proper filtering operations to narrow down the displayed records to one row and then use this activity to click the required field.

Note: You must first navigate to the screen to select the desired field to drill down. You can use the Navigation activity to navigate to the required screen.

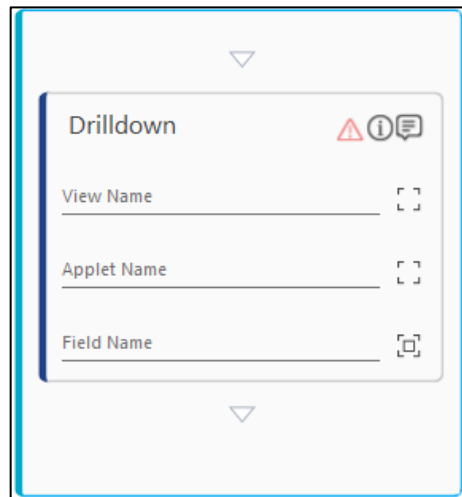
Using Drilldown Activity




1. In the **Canvas Tools** pane, click **Process Components** to expand the tool and view the associated activities.

2. Drag the **Application** activity and drop on to the **Flowchart** designer on the **Canvas**.
3. In the **Application Type** list, select **WebApps**.
4. In the **Select An Application** list, select the configured Siebel CRM application.



5. Double click the **Application** activity and navigate to the desired view. You can use the Navigation activity for the required navigation. See the Use Case for better understanding.
6. In the **Canvas Tools** pane, click Siebel to expand the tool and view the associated activities.
7. Drag the **Drilldown** activity and drop inside the **Application** activity. The validation error symbol disappears when required inputs are provided.



8. Click the  (**Get View Name**) button to automatically fetch the view name. If the view name is not retrieved, enter the view name manually in the **View Name** field.
9. Navigate to the desired applet in the docked Siebel CRM application in Automation Studio, and then click the  (**Get Applet Name**) button to automatically fetch the applet name. If the applet name is not retrieved, enter the applet name manually in the **Applet Name** field.
10. Click the  (**Get Field Name**) button. The Available Fields dialog box appears with the list of available field names.

Note:

- To know the view name, navigate to the required screen of the Siebel CRM application manually and then access **Help > About View > View** name.
- To know the applet name, navigate to the required screen of the Siebel CRM application manually and then access **Help > About View > Applets** name.
- On the Siebel CRM screen, multiple applets can be present. You must choose the applet where you want to perform the automation.

Field Display Name	Field Value	Is Visible
New	Row Status	True
SR #	SR Number	True
Account	Account	True
Team Space	TeamSpaceExistenceState	True
Last Name	Contact Last Name	True
Owner	Owner	True
Priority	Priority	True
Recommendation	Recommendation	True

CONFIRM

11. Select the required field name where you want to set the parameter. However, in few instances, not all field names are displayed or the field names are not retrieved and a pop up message **Method ColumnDisplayed is not allowed here** appears. In such cases, you must enter the field name manually.

Note: To know the field name:

- In the Siebel CRM application, click the applet where the required field is present.
- Click the **Query** button.
- Press **F12** to open the **Console** of Siebel CRM screen.
- Enter any value in the required field name, and then click Go.
- In the Console, enter the Java expression **<theApplication().ActiveApplet().BusComp.GetSearchExpr(>** and then click enter.
- The field name appears in the square bracket. You can use this field name to enter it manually in the activity.

The **Drilldown** activity is created.

Drilldown Activity Properties

The properties of a Drilldown activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the Flowchart designer. By default, the name is set as Drilldown . You can change the name as required.

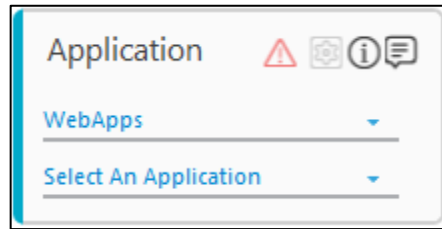
9.21.4 Set Field Value

It helps to automate setting a value in the specified field or grid of the applet in the Siebel CRM application such as, a service request or an account name.

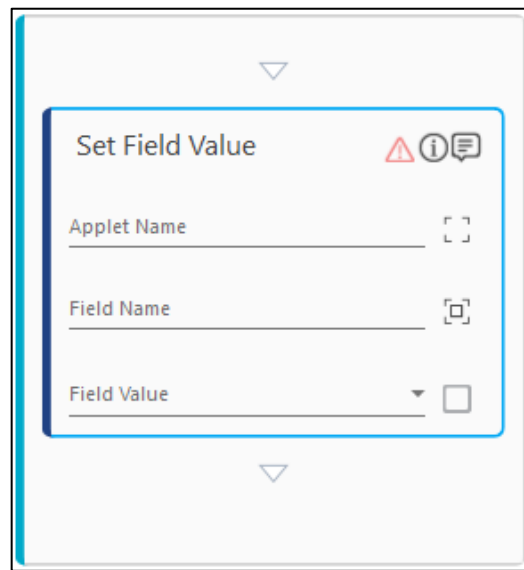
Note: You must first navigate to the screen where you want to set the required value. You can use the Navigation activity to navigate to the required screen.

Using Set Field Value Activity

1. In the **Canvas Tools** pane, click **Process Components** to expand the tool and view the associated activities.
2. Drag the **Application** activity and drop on to the **Flowchart** designer on the **Canvas**.
3. In the **Application Type** list, select **WebApps**.
4. In the **Select An Application** list, select the configured Siebel CRM application.



5. Double-click the **Application** activity and navigate to the desired applet. You can use the **Navigation** activity for the required navigation. See the [Use Case](#) for better understanding.
6. In the **Canvas Tools** pane, click **Siebel** to expand the tool and view the associated activities.
7. Drag the **Set Field Value** activity and drop inside the **Application** activity. The validation error symbol disappears when the required inputs are provided.



8. Click the  (**Get Applet Name**) button to automatically fetch the applet name. If the required applet name is not retrieved, enter the applet name manually in the **Applet Name** field.

Note:

- To know the view name, navigate to the required screen of the Siebel CRM application manually and then access **Help > About View > Applets** name.
- On the Siebel CRM screen, multiple applets can be present. You must choose the applet where you want to perform the automation.

9. Click the  (**Get Field Name**) button. The **Available Fields** dialog box appears with the list of available field names.

Field Display Name	Field Value	Is Visible
New	Row Status	True
SR #	SR Number	True
Account	Account	True
Team Space	TeamspaceExistenceState	True
Last Name	Contact Last Name	True
Owner	Owner	True
Priority	Priority	True
Recommendation	Recommendation	True

CONFIRM

10. Select the required field name and click **CONFIRM**. However, at few instances, not all field names are displayed or the field names are not retrieved at all and a pop up message **Method ColumnDisplayed is not allowed here** appears. In such cases, you must enter the field name manually.

Note: To know the field name:

- In the Siebel CRM application, click the applet where the required field is present.
- Click the **Query** button.
- Press **F12** to open the **Console** of Siebel CRM screen.
- Enter any value in the required field name, and then click Go.
- In the Console, enter the Java expression **<theApplication().ActiveApplet().BusComp.GetSearchExpr(>** and then click enter.
- The field name appears in the square bracket. You can use this field name to enter it manually in the activity. Few field names, mostly radio buttons, are not retrieved using this method. You must retrieve such field names from the Siebel back end or by using the Application Host activity in Automation Studio. See the Example of Application Host- Windows section of the Application Host activity to know the detailed steps.

Note:

- Define a reference string before you configure fields, and then define the table which dynamically, along with the reference string, locates the area of data extraction. See Add New Reference to know how to add a reference string.
- Table boundaries and columns must be defined in a manner where the string is not cut off. This may affect data extraction.

Note: For Text data type input, the Data Table data type are also available for selection:

- If you have migrated a process from a previous version of AssistEdge, the process input data defined in the Configuration screen is available in the Subsequent Process's Arguments Mapping screen.
- The subsequent process input data from the previous version of AssistEdge is displayed in the Subsequent Process's Arguments Mapping screen only after the migrated process is opened in the Edit

11. In the **Field Value** list, select the parameter holding the field value in the specific format of the specified field. You must define the parameter in the Parameter bar to use this option. Alternatively, enter the required value and select the **Default Value** check box to set it as the default value.

The **Set Field Value** activity is created.

Set Field Value Activity Properties

The properties of a **Set Field Value** activity are listed in the following table and can be edited in the **Properties** grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the Flowchart designer. By default, the name is set as Set Field Value . You can change the name as required.

9.21.5 Get Field Value

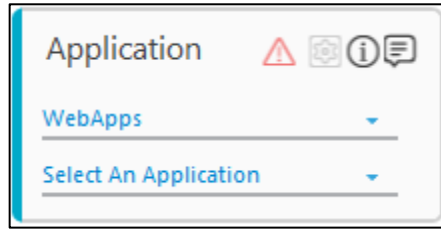
It helps to automate extracting field value from the specified field or grid of the applet present on the Siebel CRM screen such as, details of a service request or an account name. It stores the extracted data in a string parameter, that can be used for further processing in the automation process workflow.

Note: You must first navigate to the screen from where you want to retrieve the field value. You can use the Navigation activity to navigate to the required screen.

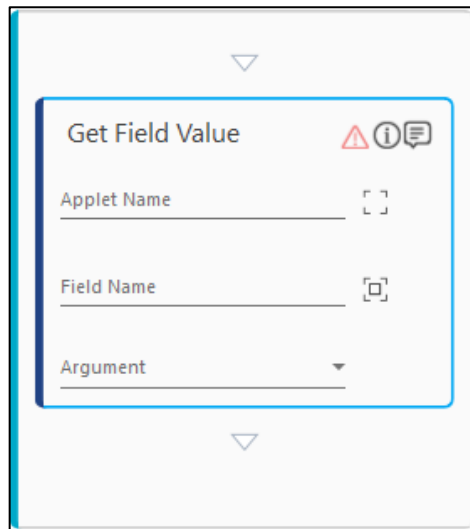
Using Get Field Value Activity


1. In the **Canvas Tools** pane, click **Process Components** to expand the tool and view the associated activities.
2. Drag the **Application** activity and drop on to the **Flowchart** designer on the **Canvas**.
3. In the **Application Type** list, select **WebApps**.

4. In the **Select An Application** list, select the configured Siebel CRM application.



5. Double-click the **Application** activity and navigate to the desired applet. You can use the **Navigation** activity for the required navigation. See the [Use Case](#) for better understanding.
6. In the **Canvas Tools** pane, click **Siebel** to expand the tool and view the associated activities.
7. Drag the **Get Field Value** activity and drop inside the **Application** activity. The validation error symbol disappears when the required inputs are provided.



8. Click the  (**Get Applet Name**) button to automatically fetch the applet name. If the required applet name is not retrieved, enter the applet name manually in the **Applet Name** field.

Note:

- To know the view name, navigate to the required screen of the Siebel CRM application manually and then access **Help > About View > Applets** name.
- On the Siebel CRM screen, multiple applets can be present. You must choose the applet where you want to perform the automation.

9. Click the  (**Get Field Name**) button. The **Available Fields** dialog box appears with the list of available field names.

Field Display Name	Field Value	Is Visible
New	Row Status	True
SR #	SR Number	True
Account	Account	True
Team Space	TeamSpaceExistenceState	True
Last Name	Contact Last Name	True
Owner	Owner	True
Priority	Priority	True
Recommendation	Recommendation	True

CONFIRM

10. Select the required field name and click **CONFIRM**. However, at few instances, not all field names are displayed or the field names are not retrieved at all and a pop up message **Method ColumnDisplayed is not allowed here** appears. In such cases, you must enter the field name manually.

Note: To know the field name:

- In the Siebel CRM application, click the applet where the required field is present.
- Click the **Query** button.
- Press **F12** to open the **Console** of Siebel CRM screen.
- Enter any value in the required field name, and then click Go.
- In the Console, enter the Java expression `<theApplication().ActiveApplet().BusComp.GetSearchExpr(>` and then click enter.
- The field name appears in the square bracket. You can use this field name to enter it manually in the activity.

11. In the **Argument** list, select a parameter to store the extracted field value. You must define the parameter in the Parameter bar to use this option.

The **Get Field Value** activity is created.

Get Field Value Activity Properties

The properties of a Get Field Value activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the Flowchart designer. By default, the name is set as Get Field Value . You can change the name as required.

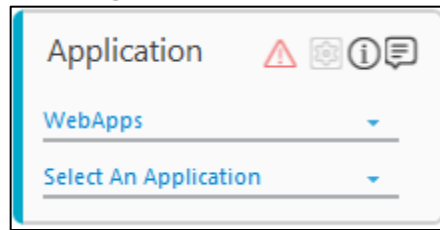
9.21.6 Fetch Grid Data

It helps to automate read the value of the desired grid from a single or multiple rows of the selected applet. You can filter the grid using other Siebel activities and fetch the required grid data.

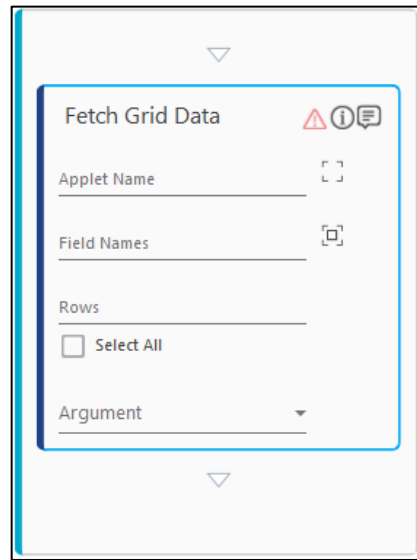
Note: You must first navigate to the screen from where you want to retrieve the field value. You can use the Navigation activity to navigate to the required screen.

Using Set Field Value Activity

1. In the **Canvas Tools** pane, click **Process Components** to expand the tool and view the associated activities.
2. Drag the **Application** activity and drop on to the **Flowchart** designer on the **Canvas**.
3. In the **Application Type** list, select **WebApps**.
4. In the **Select An Application** list, select the configured Siebel CRM application.



5. Double-click the **Application** activity and navigate to the desired applet. You can use the **Navigation** activity for the required navigation. See the [Use Case](#) for better understanding.
6. In the **Canvas Tools** pane, click **Siebel** to expand the tool and view the associated activities.
7. Drag the **Fetch Grid Data** activity and drop inside the **Application** activity. The validation error symbol disappears when the required inputs are provided.

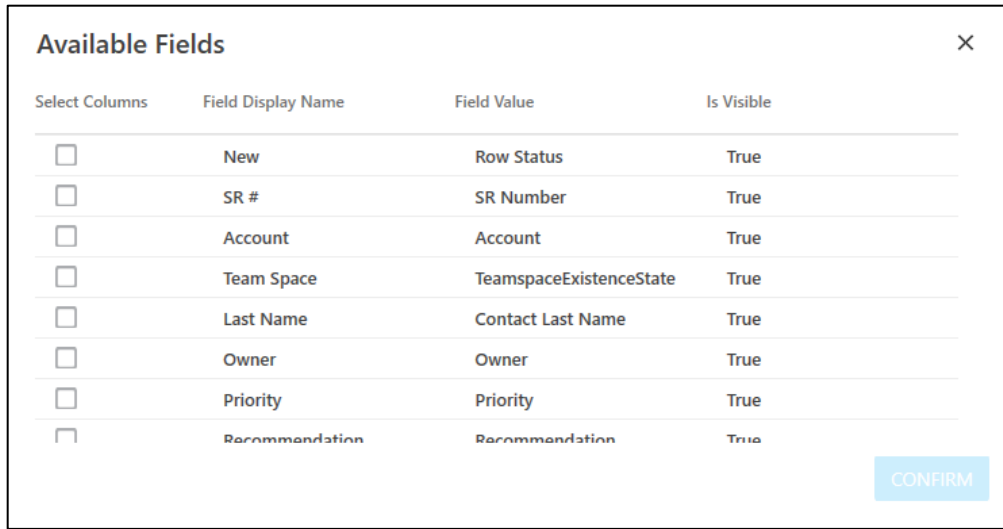


8. Click the  (**Get Applet Name**) button to automatically fetch the applet name. If the required applet name is not retrieved, enter the applet name manually in the **Applet Name** field.

Note:

- To know the view name, navigate to the required screen of the Siebel CRM application manually and then access **Help > About View > Applets** name.
- On the Siebel CRM screen, multiple applets can be present. You must choose the applet where you want to perform the automation.

- Click the  (**Get Field Name**) button. The **Available Fields** dialog box appears with the list of available field names.



- Select the required field name and then click **CONFIRM**. You can select one or multiple field names, as per your requirement.
- In the **Rows** field, enter the specific number of rows to fetch the data from the grid. You can select the Select All check box to retrieve all rows from the grid.
- In the **Argument** list, select the parameter of the data table type to store the extracted grid. You must define the parameter in the Parameter bar to use this option.
The **Fetch Grid Data** activity is created. The **Get Field Value** activity is created.

Fetch Grid Data Activity Properties

The properties of a Fetch Grid Data activity are listed in the following table and can be edited in the Properties grid on the right panel.

Property Name	Usage
Control Execution	
Ignore Error	When this option is set to Yes , the application ignores any error while executing the activity. If set to NA , it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception. By default, this option is set to No .
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.

Property Name	Usage
Misc	
Breakpoint	Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected. In large or complex processes, breakpoints help in identifying the error, if any.
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the Flowchart designer. By default, the name is set as Get Field Value . You can change the name as required.
IsFieldValueDefault	Indicates if all rows must be retrieved from the grid. Alternatively, you can specify if you want to retrieve all the rows from a grid by selecting Select All check box of the activity block. The value specified in the Properties grid reflects in the activity block and vice versa.
RowCount	Indicates the number of rows that must be retrieved from the grid. Alternatively, you can specify the number of rows in the Rows field of the activity block. The value specified in the Properties grid reflects in the activity block and vice versa.
SelectedColumns	Indicates the field name that you want to retrieve form the grid. Alternatively, you can specify the field name in the Field Names field of the activity block. The value specified in the Properties grid reflects in the activity block and vice versa.

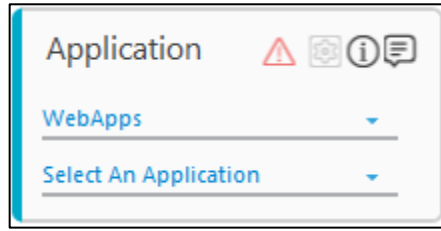
9.21.7 Open Task

It helps to automate the operations related to Tasks of the Siebel CRM applicatio using the task name.

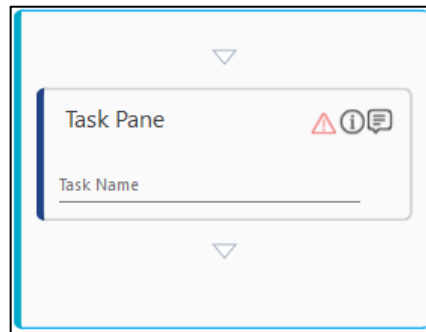
Using Open Task Activity

1. In the **Canvas Tools** pane, click **Process Components** to expand the tool and view the associated activities.
2. Drag the **Application** activity and drop on to the **Flowchart** designer on the **Canvas**.
3. In the **Application Type** list, select **WebApps**.

4. In the **Select An Application** list, select the configured Siebel CRM application.



5. Double click the **Application** activity to open the activity block.
6. In the **Canvas Tools** pane, click **Siebel** to expand the tool and view the associated activities.
7. Drag the **Open Task** activity and drop inside the **Application** activity. The validation error symbol disappears when the required inputs are provided.



8. In the **Task Name** field, enter the task name of the required **Task**.

Note: To know the task name, navigate to the required screen of the Siebel CRM application manually and then access Site Map > Administration - Web Services. Search tasks in the Find option, and then copy the task name of the desired Task.

The Open Task activity is created.

You can use the WEB MODE to configure the click action on the Next, Previous and other buttons of the displayed Task. See the [Web Mode](#) section to know how to configure the steps.

Open Task Activity Properties

The properties of the Open Task activity are listed in the following table and can be edited in the Properties grid on the right panel.


Property Name	Usage
Control Execution	
Ignore Error	<p>When this option is set to Yes, the application ignores any error while executing the activity. If set to NA, it bypasses the exception (if any) to let the automation flow continue; however, it marks the automation status as failure, in case of an exception.</p> <p>By default, this option is set to No.</p>
Delay	
Wait After	Specify the time delay that must occur after the activity is executed. The value must be in milliseconds.
Wait Before	Specify the time delay that must occur before the activity is executed. The value must be in milliseconds.
Misc	
Breakpoint	<p>Select this option to mark this activity as the pause point while debugging the process. At this point, the process freezes during execution allowing you to examine if the process is functioning as expected.</p> <p>In large or complex processes, breakpoints help in identifying the error, if any.</p>
Commented	Select this option to mark this activity as inactive in the entire process. When an activity is commented, it is ignored during the process execution.
DisplayName	The display name of the activity in the Flowchart designer. By default, the name is set as Open Task . You can change the name as required.

9.21.8 Step-By-Step Guide to Use Siebel to Search for Service Request on the Oracle Siebel CRM HI Application

Let's automate navigating to All Service Request link of the Service Request tab of the Oracle Siebel CRM HI application and search for the specified service request using Query. We should be able to click Go to view the details of the specified service request and then extract its summary details. We are going to drilldown the service request, fetch grid data of some of the fields of displayed Activities and view the extracted data in an excel file.

Note: The Siebel activity stores the extracted grid data in a data table. You must export the data table in to a viewable format, such as excel.

Prerequisite:




1. In the **Admin** menu, add an application of **Application Type** - WebApps.
2. In the Application Properties panel:
 - a. Enter the **Login URL** of the Siebel CRM application.
 - b. Enter the **Search URL** of the Siebel CRM application.
 - c. Enter the **Display Name**. In this example, **SiebelHI** is entered.
3. Click the  (**Save Properties**) icon to save the application details.
4. Enter other details as per your requirement.
The Siebel CRM application is configured.

To automate steps as per the scenario:

5. Create a [SignIn](#) Process for the configured Siebel CRM application. [Publish](#) and [Deploy](#) the **Sign In process** to use it in the relevant automation process workflow created for the configured Siebel CRM application.
6. Create a data table argument, **ActivityDetails** to store the extracted grid data.
7. Create the string arguments:
 - **Summary**, to store the field value extracted from the **Summary** field.
 - **FilePath**, to store the exported data table values. Specify the file location along with the file name and file extension.

Name	Direction	Argument type	Default value
ActivityDetails	In	DataTable	<i>Enter a VB expression</i>
Summary	In	String	<i>Enter a VB expression</i>
FilePath	In	String	"D:\Sample Data\Out\SiebelOutput.xlsx"
LoginId	In	String	<i>Enter a VB expression</i>
Password	In	String	<i>Enter a VB expression</i>


8. From the **Canvas Tools** panel, add **Application** activity to the **Flowchart** designer on the **Canvas**.


9. In the **Application Type** list, select **WebApps**.
10. In the **Select an Application** list, select the configured Siebel application.
11. Double click the **Application** activity and add the **Navigation** activity inside the **Application** activity.
12. In the View Name field, enter All Service Request List View view name to navigate to the All Service Request link of the Service Request tab.
13. Set up the environment and test run the process. This launches the Siebel application and automatically navigates to the desired screen. This step helps you to auto retrieve the required information such as the applet name using the docked application in Automation Studio.
14. Add the **Siebel Operations** activity below the **Navigation** activity.
15. Click the  (**Get Applet Name**) button to auto fill the applet name.
16. In the **Select Operation** list, select **Query** to initiate searching the desired service request.
17. Add the **Set Field Value** activity below the **Siebel Operations** activity.
18. Click the  (**Get Applet Name**) button to auto fill the applet name.
19. Click the  (**Get Field Name**) button. The **Available Fields** dialog box appears with the list of available field names.

Available Fields ✕

Field Display Name	Field Value	Is Visible
New	Row Status	True
SR #	SR Number	True
Account	Account	True
Team Space	TeamSpaceExistenceState	True
Last Name	Contact Last Name	True
Owner	Owner	True
Priority	Priority	True
Recommendation	Recommendation	True

CONFIRM

20. Select **SR #** and then click **CONFIRM**. You are taken back to the **Studio** menu.
21. In the **Field Value** field, enter a service request and select **Is Default Value** to set the service request number as the default value.
22. Add **Siebel Operations** activity below the **Set Field Value** activity.
23. Click the  (**Get Applet Name**) button to auto fill the applet name.
24. In the **Select Operation** list, select **Go** to perform the **Go** operation that displays the details related to the specified service request.
25. Add **Get Field Value** activity below the **Siebel Operations** activity.

26. Click the  (**Get Applet Name**) button to auto fill the applet name.

27. Click the  (**Get Field Name**) button. The **Available Fields** dialog box appears with the list of available field names.

Field Display Name	Field Value	Is Visible
Priority	Priority	True
Recommendation	Recommendation	True
Sub Area	INS Sub-Area	True
Summary	Abstract	True
SR Type	SR Type(New)	True
Area	INSArea	True
Status(icon)	HelpDesk SR Resolution	False
Site	Account Location	False


CONFIRM


28. Select **Summary** and then click **CONFIRM**. You are taken back to the Studio menu.

29. In the **Argument** list, select the **Summary** argument created above. To view the extracted Summary details in Automation Studio, let's add **WriteLine** activity.

30. Add a **WriteLine** activity below the **Get Field Value** activity.

31. Add **Drilldown** activity below the **Writeline** activity to drilldown the displayed service request.

32. Click the  (**Get View Name**) button to auto-fill the view name.

33. Click the  (**Get Applet Name**) button to auto-fill the applet name.


34. Click the  (**Get Field Name**) button. The **Available Fields** dialog box appears with the list of available field names.

Field Display Name	Field Value	Is Visible
New	Row Status	True
SR #	SR Number	True
Account	Account	True
Team Space	TeamSpaceExistenceState	True
Last Name	Contact Last Name	True
Owner	Owner	True
Priority	Priority	True
Recommendation	Recommendation	True

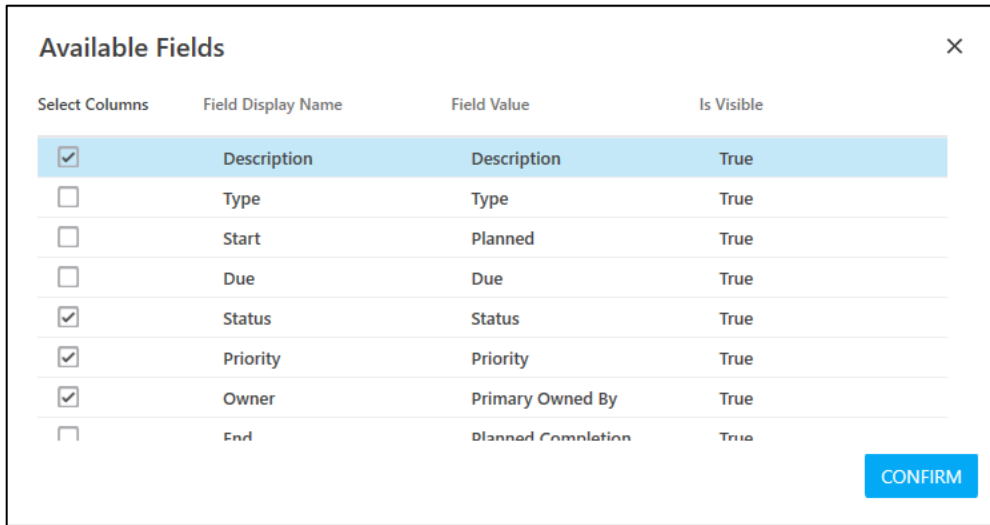
CONFIRM

35. Select **SR #** and then click **CONFIRM**. You are taken back to the Studio menu.

36. Add **Fetch Grid Data** activity below the **Drilldown** activity.

37. Click the  (**Get Applet Name**) button to auto-fill the applet name.

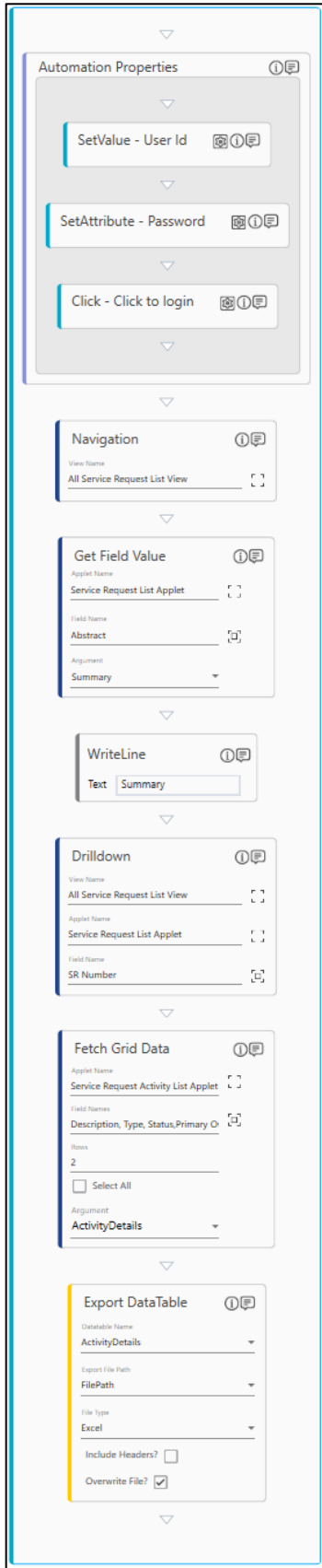
38. Click the  (**Get Field Name**) button. The **Available Fields** dialog box appears with the list of available field names.



Select Columns	Field Display Name	Field Value	Is Visible
<input checked="" type="checkbox"/>	Description	Description	True
<input type="checkbox"/>	Type	Type	True
<input type="checkbox"/>	Start	Planned	True
<input type="checkbox"/>	Due	Due	True
<input checked="" type="checkbox"/>	Status	Status	True
<input checked="" type="checkbox"/>	Priority	Priority	True
<input checked="" type="checkbox"/>	Owner	Primary Owned By	True
<input type="checkbox"/>	End	Planned Completion	True

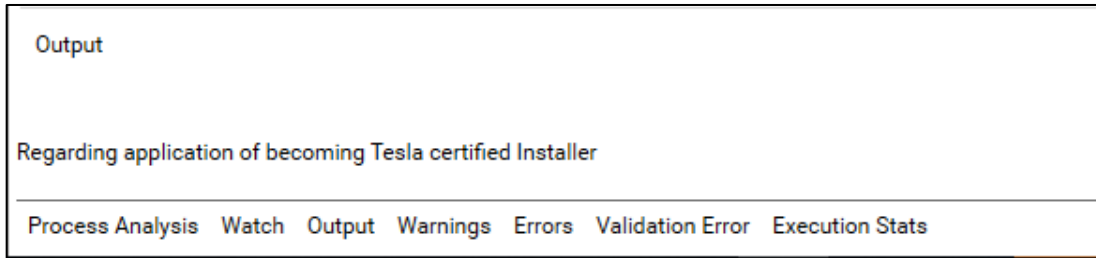
39. Select **Description, Status, Priority** and **Owner** field names to extract their respective grid data and then click **CONFIRM**. You are taken back to the **Studio** menu.
40. In the **Row** field, enter **2**. This indicates the tool to extract the specified grid data from two rows.
41. In the **Argument** list, enter the **ArgumentDetails** argument to store the extracted grid data.
42. Add **Export DataTable** activity to store and view the extracted grid data in an excel.
43. In the **Datatable** Name list, select the **ActivityDetails** argument.
44. In the **Export File Path** list, select the **FilePath** argument created above.
45. In the **File Type** list, select **Excel**.

4.6. Save the process. Below is the screen shot of how the process workflow looks:

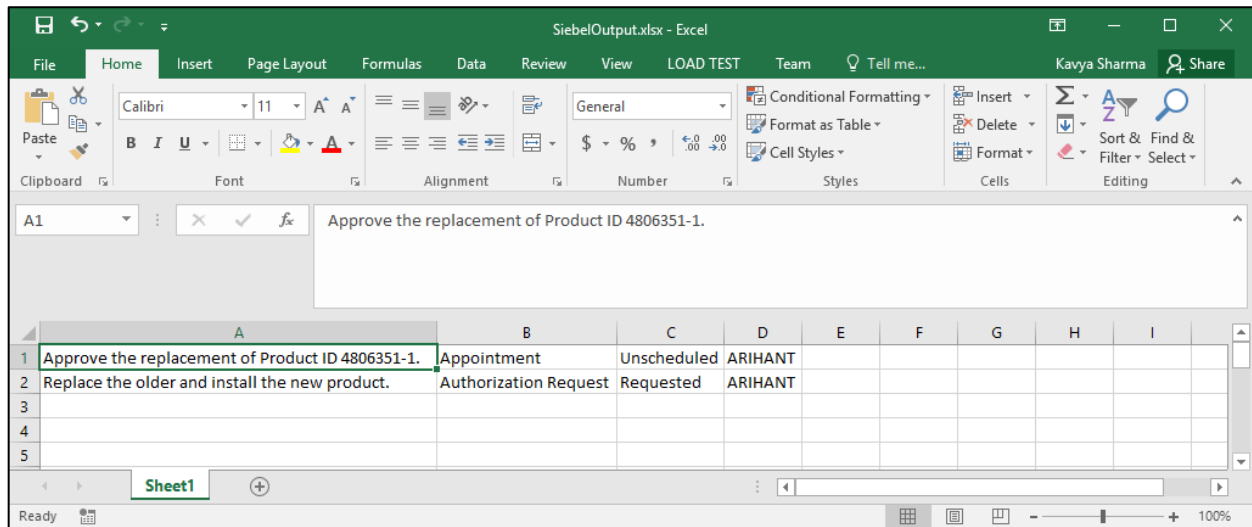


To view the output in Automation Studio, reset and then [set up the environment](#). Perform the [test run](#). You can assign this process to a robot, if you want to execute this process outside Automation Studio.

The extracted **Summary** field value is displayed in the **Output** console of Automation Studio.



The data table with the extracted grid data is exported into an Excel file and is saved at the specified location.



9.22 Automation of Oracle E-Business Suite using Micro-bots

AssistEdge RPA Studio supports automation of various operations on Java-based Oracle Forms. Following Micro-bots are currently available for automation of Oracle forms through AssistEdge RPA Design Studio:

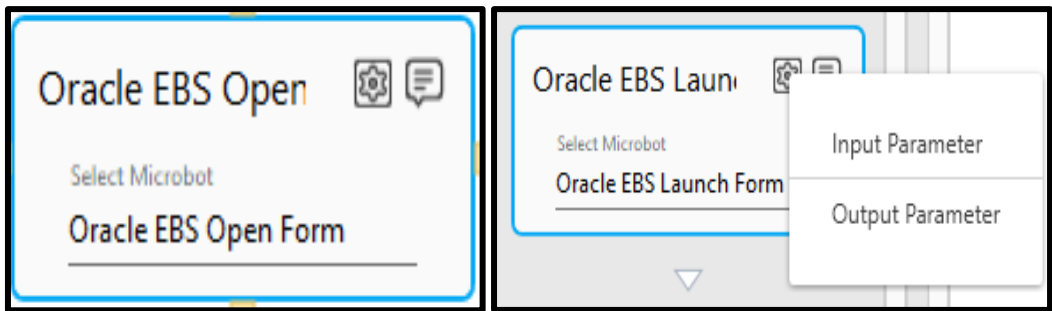
- Oracle EBS Open Form
- Oracle EBS Menu Options
- Oracle EBS Toolbar Options
- Oracle EBS Set Text
- Oracle EBS Set Radio Button
- Oracle EBS Get Text
- Oracle EBS Get Radio Button
- Oracle EBS Select Tab
- Oracle EBS Click Hyperlink

- Oracle EBS Switch Form
- Oracle EBS Click Button
- Oracle EBS Start Mapping Server
- Oracle EBS Stop Mapping Server

Micro-bots are available as part of the build and are imported to AssistEdge Design Studio. All micro-bots are available under the Microbot activity of AssistEdge RPA once imported.

9.22.1 Oracle EBS Open Form

Automate the opening of an Oracle form with the help of Oracle EBS Open Form microbot.



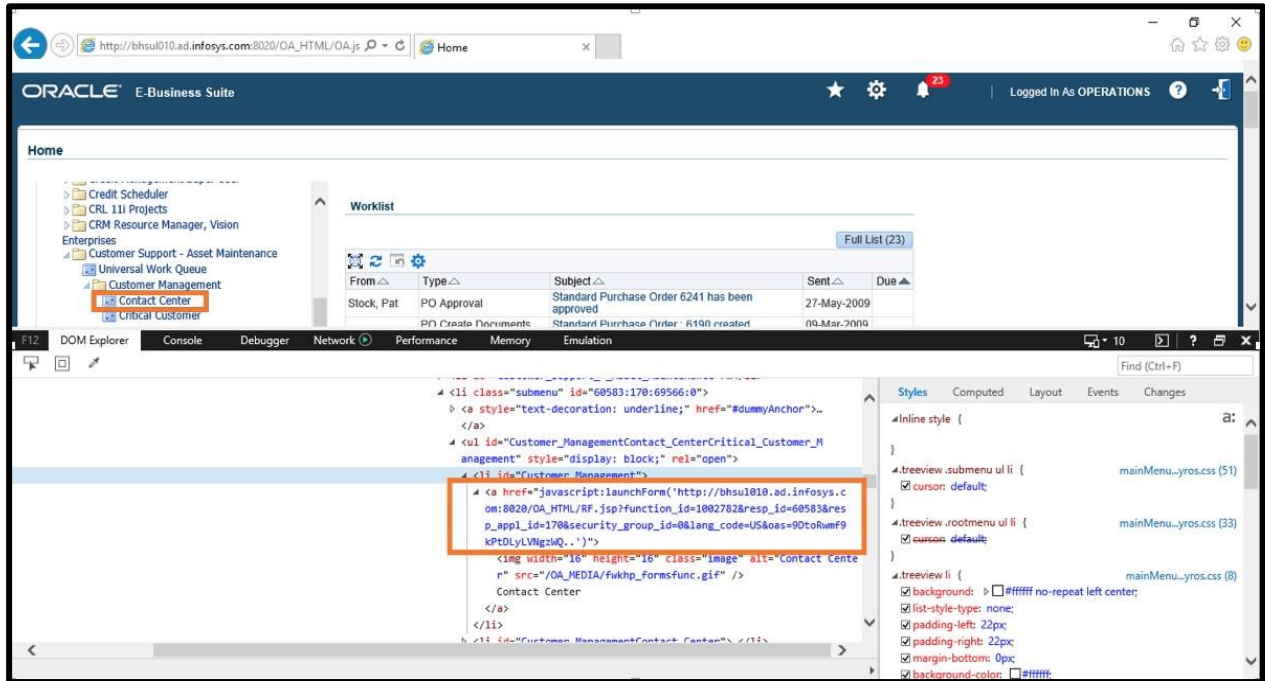
Input Parameters

Argument	Argument Type	Map to Parent	If Default	Default Value
FormUrl	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
InUrl	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
EbsWindowName	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>

Provide default value or mapping to argument FormUrl

CONFIRM

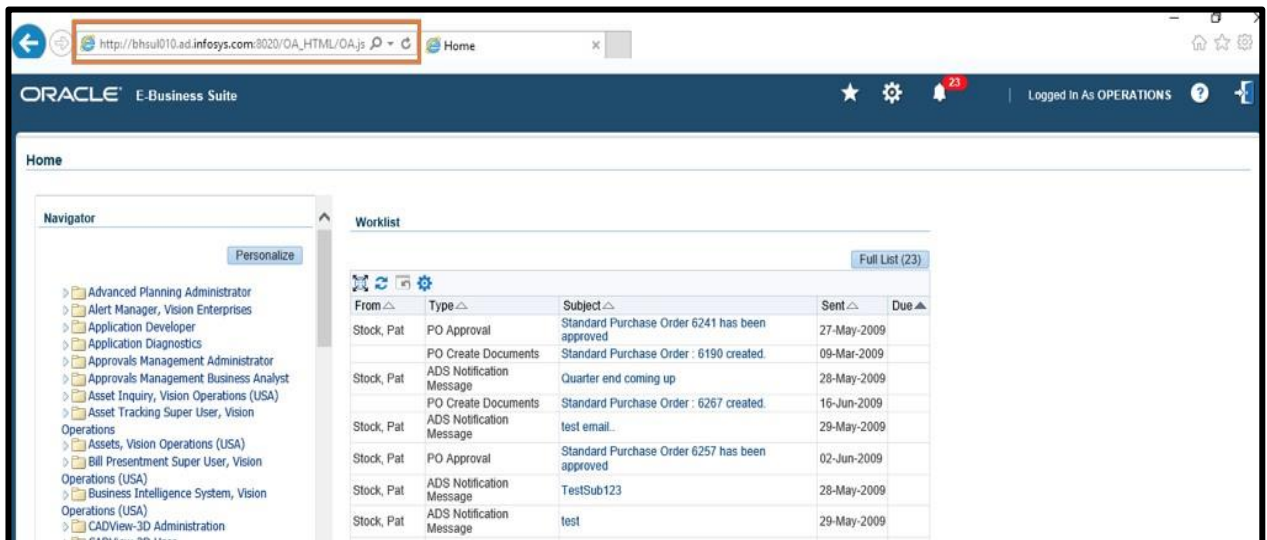
- **Form URL:** The input to this field is the URL of the form to be launched.



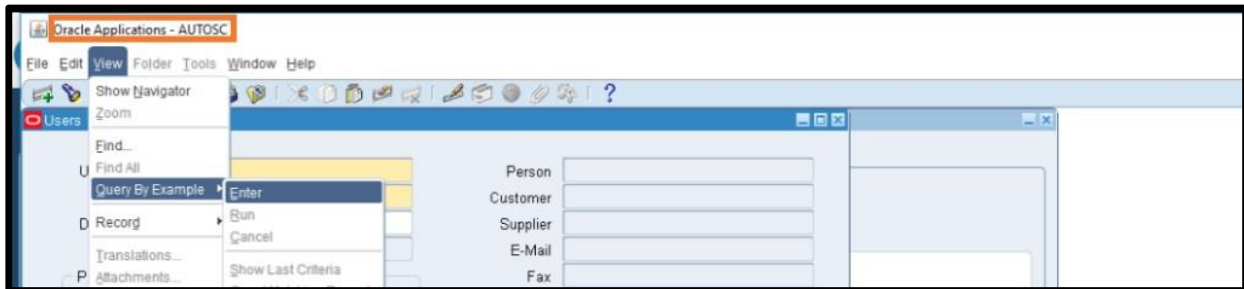
As seen in the screenshot, the form URL is

http://bhsul010.ad.infosys.com:8020/OA_HTML/RF.jsp?function_id=1002782&resp_id=60583&resp_appl_id=170&security_group_id=0&lang_code=US&oas=9DtRwrmf9kPtDLyLVNgzWQ..

- **In URL:** The input to this field is the URL of the page that immediately comes up after logging in to the Oracle EBS application.



- **EBS Window Name:** The input to this field is the name of the window under which the Oracle form opens.



Output parameters

Microbot Mappings ✕

Argument	Argument Type	Map to Parent	If Default	Default Value
Status	Boolean	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>
StatusMessage	Text	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>
ApplicationWindow Text		<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>
PortNumber	Numeric	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>

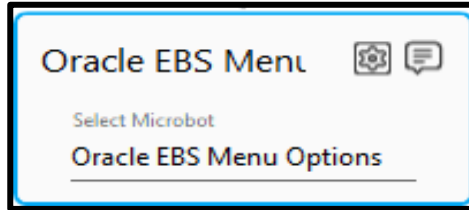
Provide default value or mapping to argument Status

CONFIRM

- **Status:** This parameter gets the output as either True or false depending upon whether or not the Microbot successfully completed the operation.
- **StatusMessage:** This parameter would get the output as either Success Message or a specific error message depending on whether the Microbot successfully completed the operation or it failed due to an exception.
- **Port Number:** This parameter gets the port number as an output. Java applet runs on this port.
- **Application Window:** This parameter gets the Application Window name as an output.

9.22.2 Oracle EBS Menu Options

Automate selecting of a Menu option, with the help of Oracle EBS Menu Options microbot.



Input Parameters

- **Window Name:** The input for this field is the name of the window under which the Oracle form opens.

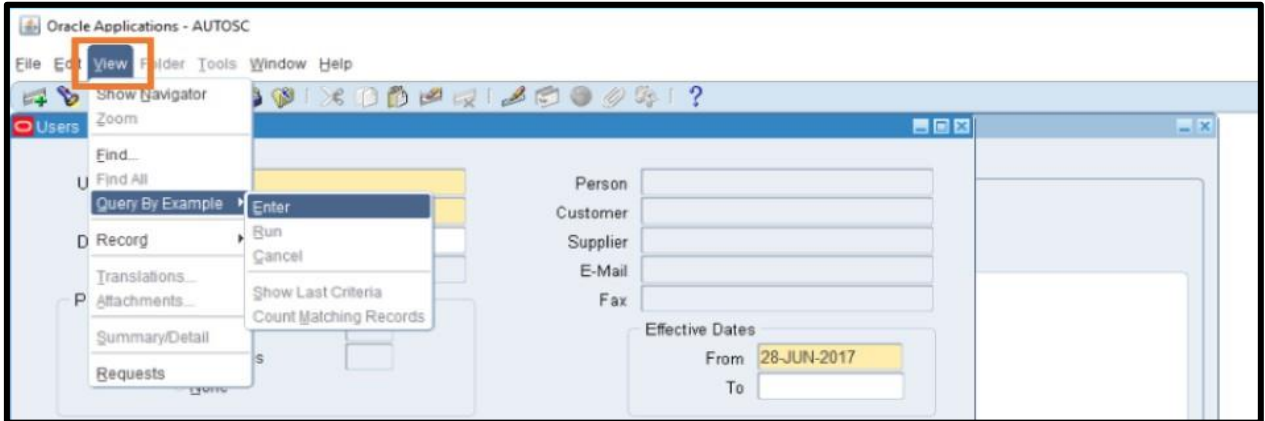
WindowName	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
ApplicationPort	Numeric	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
MainMenu	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
SubMenu	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>

Provide default value or mapping to argument WindowName

CONFIRM

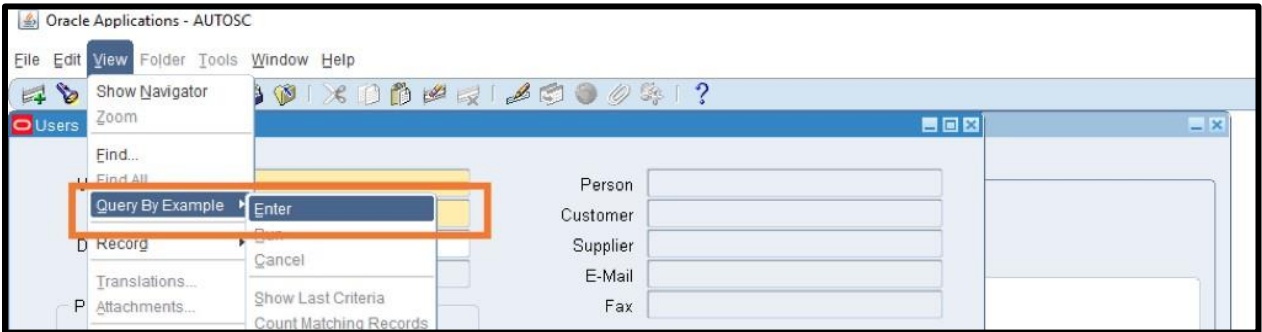
- **ApplicationPort:** This parameter takes the value captured through Port Number field in Open Form microbot's outputparameter. Java applet runs on the application port and the application port is dynamic.

- **Main Menu:** The input to this field is the main menu option that is selected. E.g., **"View"**.



- **Sub Menu:** The input to this parameter is the sub menu option to be selected. If there are multiple levels in the sub-menu, values are entered in a forward slash separated format. E.g., **"Query By Example/Enter"**.

47.



Output Parameters

Microbot Mappings ✕

Argument	Argument Type	Map to Parent	If Default	Default Value
StatusMessage	Text	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>
Status	Boolean	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>

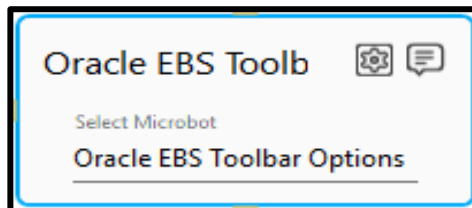
Provide default value or mapping to argument StatusMessage

CONFIRM

- Status: This parameter gets the output as either True or false depending on the successful completion of the operation or failed due to an exception by the Microbot.
- StatusMessage: This parameter gets the output as either Success Message or a specific error message depending upon the successful completion or failed due to an exception.

9.22.3 Oracle EBS Toolbar Options

From the Oracle form, select “Oracle EBS Toolbar Options” microbot.



Input Parameters

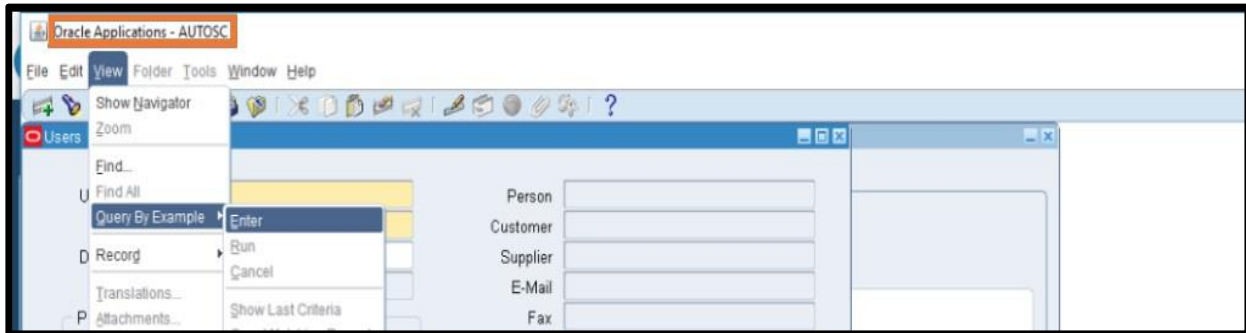
Microbot Mappings ✕

Argument	Argument Type	Map to Parent	If Default	Default Value
WindowName	Text	<input type="text" value=""/>	<input type="checkbox"/>	<input type="text" value=""/>
ApplicationPort	Text	<input type="text" value=""/>	<input type="checkbox"/>	<input type="text" value=""/>
FormName	Text	<input type="text" value=""/>	<input type="checkbox"/>	<input type="text" value=""/>
ToolbarControlName	Text	<input type="text" value=""/>	<input type="checkbox"/>	<input type="text" value=""/>

Provide default value or mapping to argument WindowName

CONFIRM

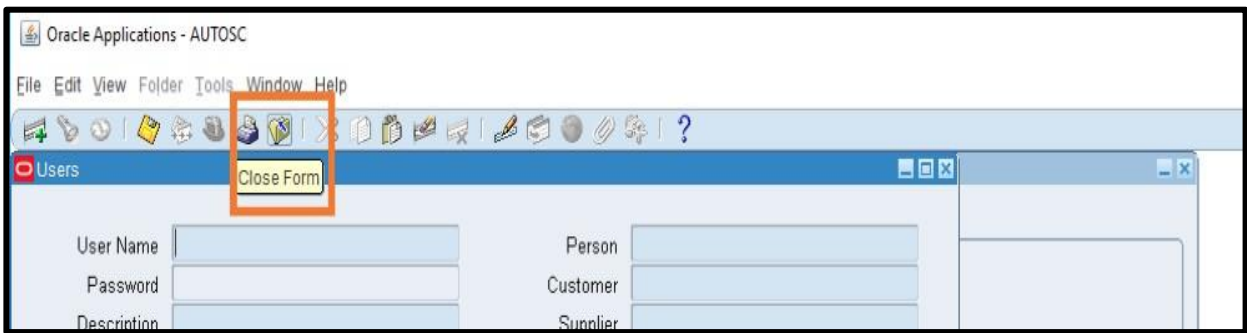
- **Window Name:** The input to this field is the name of the window under which the Oracle form opens.



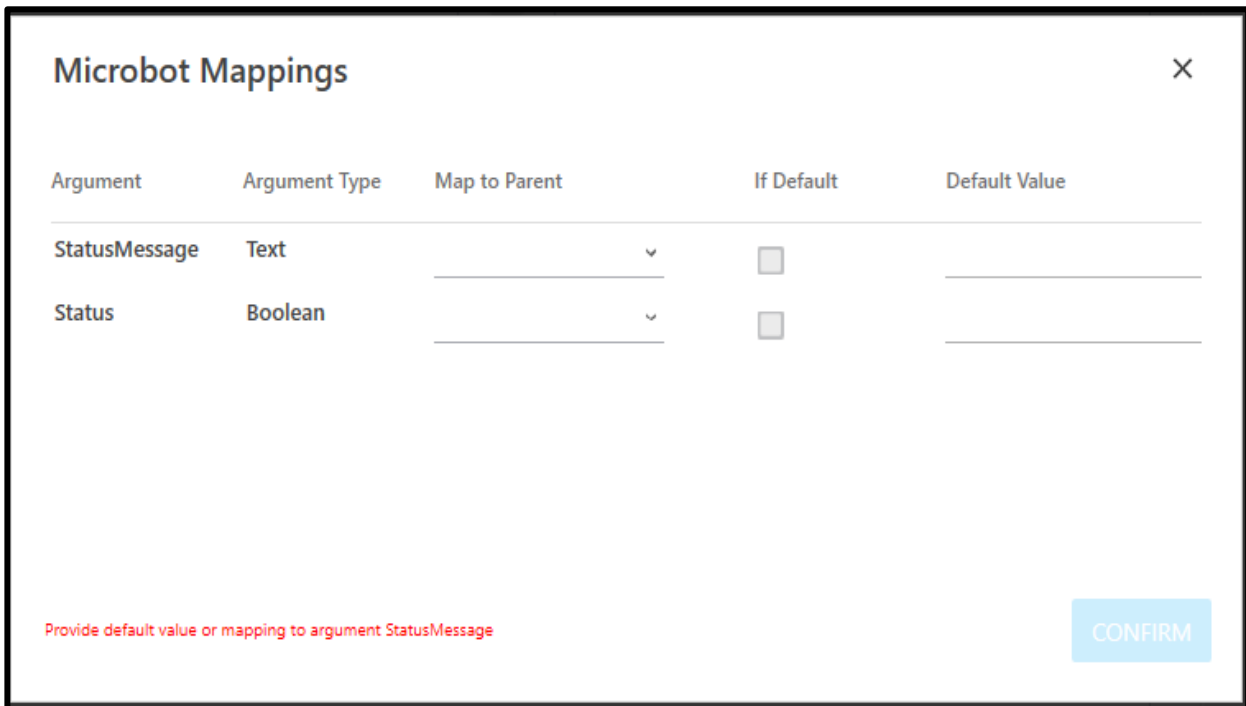
- **ApplicationPort:** This parameter takes the value captured through a Port Number field in the Open Form microbot’s output parameter. Java applet runs on this port, and it is dynamic.
- **FormName:** The Input to this parameter is the name of the form on which operation is performed.



- **ToolBarControlName:** The input to this parameter is the name of the toolbar option that is selected. For example: "CloseForm".



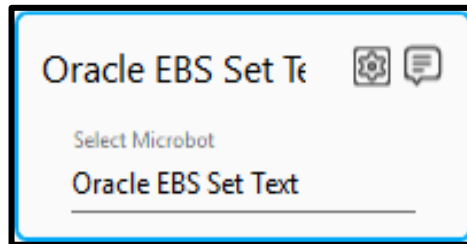
Output Parameters



- **Status:** This parameter would get the output as either True or false based on the status of operation completion by the Micro-bot.
- **StatusMessage:** This parameter gets the output as either Success Message or a specific error message depending upon whether the Microbot successfully completed the operation or failed due to an exception.

9.22.4 Oracle EBS Set Text

Automate the setting of the text in a form field in an Oracle form with the help of Oracle EBS Set Text microbot.



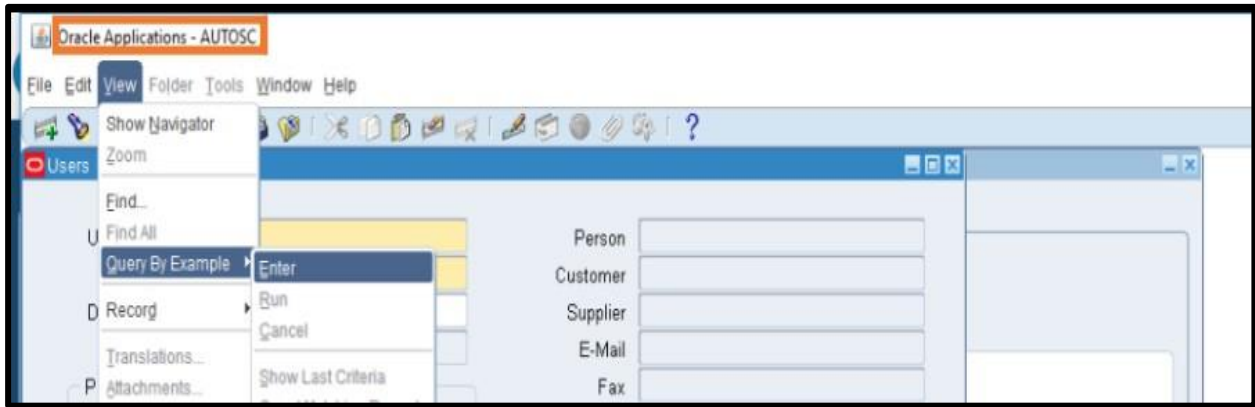
Input Parameters

Argument	Argument Type	Map to Parent	If Default	Default Value
WindowName	Text	<input type="checkbox"/>	<input type="checkbox"/>	
ApplicationPort	Numeric	<input type="checkbox"/>	<input type="checkbox"/>	
TextFieldId	Text	<input type="checkbox"/>	<input type="checkbox"/>	
InputString	Text	<input type="checkbox"/>	<input type="checkbox"/>	

Provide default value or mapping to argument WindowName

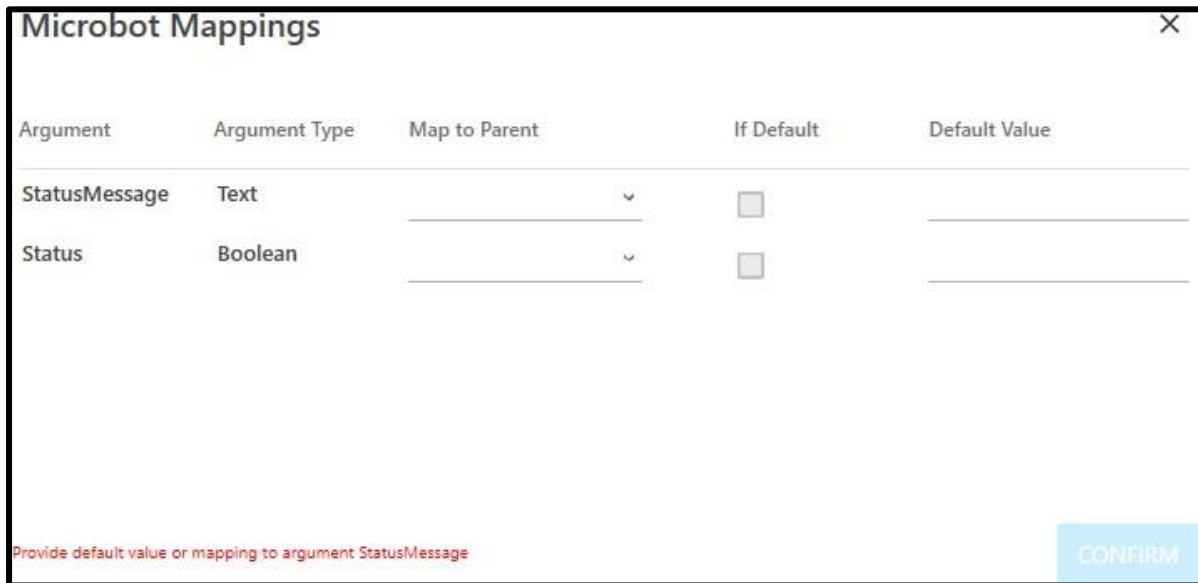
CONFIRM

- **Window Name:** The input to this field is the name of the window under which the Oracle form opens.



- ApplicationPort: This parameter takes the value captured through Port Number field in Open Form microbot’s output parameter. Java applet runs on this port, and it is dynamic.
- TextFieldId: The input for this field is the search path of the form field where the text is set.
- InputString: The input for this field is the value that is provided as input to the text field.

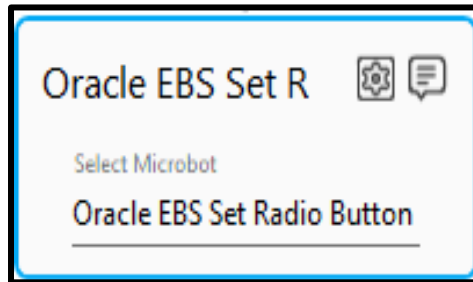
Output Parameters



- Status: This parameter gets the output as either True or false based on the Microbot if it successfully completed the operation or not
- StatusMessage: This parameter gets the output as either Success Message or a specific error message depending upon whether Microbot successfully completed the operation or failed due to an exception.

9.22.5 Oracle EBS Set Radio Button

Automate the setting of a radio button or a check the box in the Oracle form with the help of Oracle EBS Set Radio Button microbot.



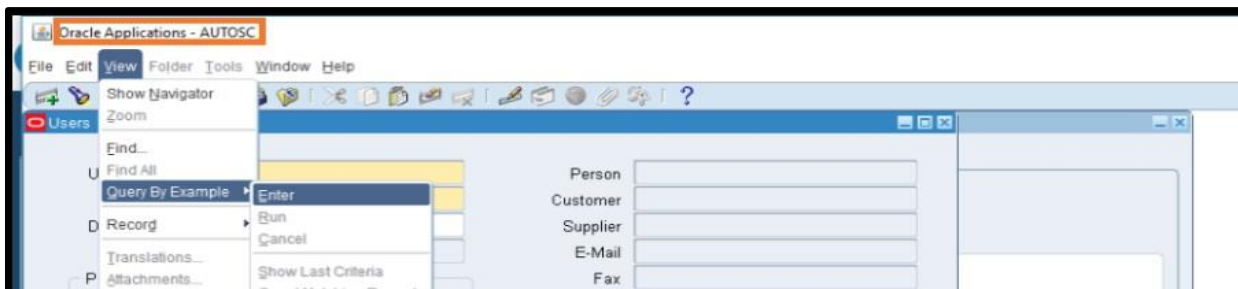
Input Parameters

Argument	Argument Type	Map to Parent	If Default	Default Value
WindowName	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
ApplicationPort	Numeric	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
FormName	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
RadioButtonId	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>

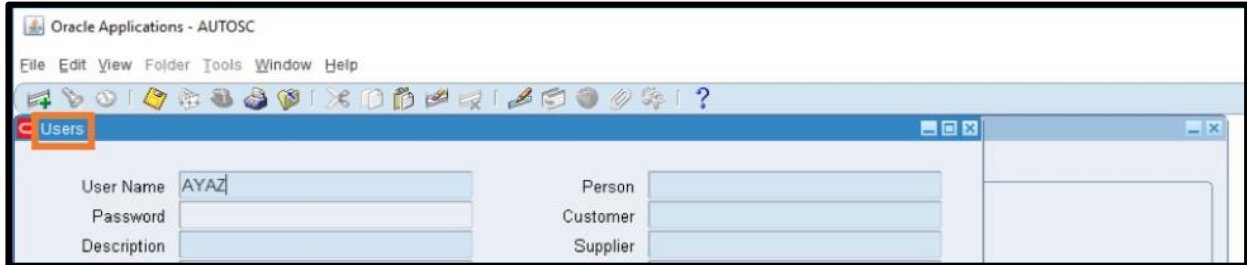
Provide default value or mapping to argument WindowName

CONFIRM

- **Window Name:** The input to this field is the name of the window under which the oracle form opens.



- **ApplicationPort:** This parameter takes the value captured through Port Number field in Open Form microbot's output parameter. Java applet runs on this port, and it is dynamic.
- **FormName:** The Input to this parameter is the name of the form on which operation is performed.



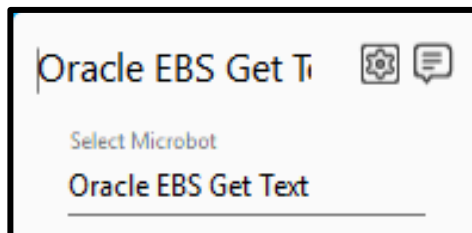
- **RadioButtonID:** The input for this field is search path of the radio button or the checkbox.

Output Parameters

- **Status:** This parameter gets the output as either True or false depending on the Microbot if it could successfully complete the operation.
- **StatusMessage:** This parameter gives the output as either Success Message or a specific error message depending upon the Microbot if it successfully completed the operation or failed due to an exception.

9.22.6 Oracle EBS Get Text

Automate the reading of the text from a form field in an Oracle form with the help of Oracle EBS Get Text microbot.



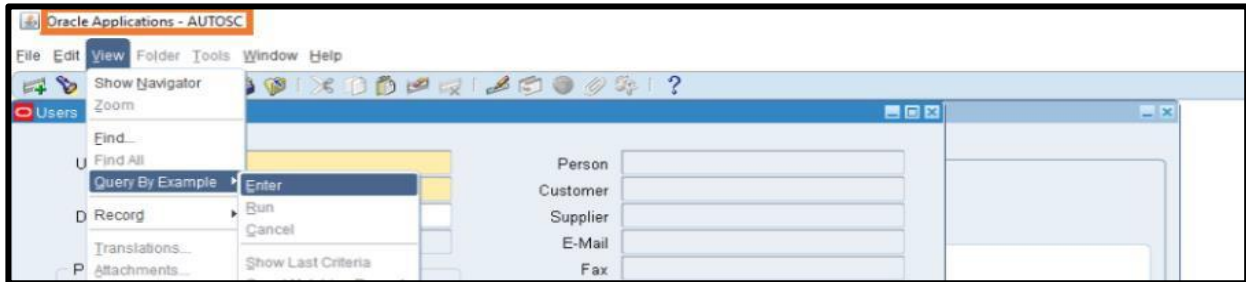
Input Parameters

Argument	Argument Type	Map to Parent	If Default	Default Value
WindowName	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
ApplicationPort	Numeric	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
TextFieldId	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>

Provide default value or mapping to argument WindowName

CONFIRM

- **Window Name:** The input to this field is the name of the window under which the Oracle form opens.



- **ApplicationPort:** This parameter takes the value captured through the Port Number field in Open Form microbot’s output parameter. Java applet runs on this port and is dynamic.
- **TextFieldId:** The input to this field is the search path of the form field.

Output Parameters

Microbot Mappings ✕

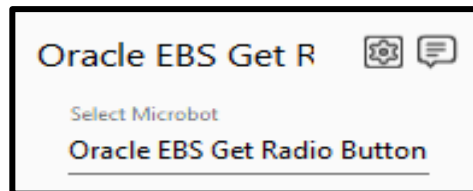
Argument	Argument Type	Map to Parent	If Default	Default Value
Status	Boolean	<input type="checkbox"/>	<input type="checkbox"/>	
StatusMessage	Text	<input type="checkbox"/>	<input type="checkbox"/>	
OutputString	Text	<input type="checkbox"/>	<input type="checkbox"/>	

CONFIRM

- **Status:** This parameter gets the output as either True or false depending upon whether or not the Microbot successfully completed the operation.
- **StatusMessage:** This parameter gets the output as either True or false depending on the Microbot if it successfully completed the operation.
- **OutputString:** This parameter gets the output as the value present in the form field is being read.

9.22.7 Oracle EBS Get Radio Button

Automate the status of the reading of a radio button or a checkbox in an Oracle form with the help of Oracle EBS Get Radio Button microbot.



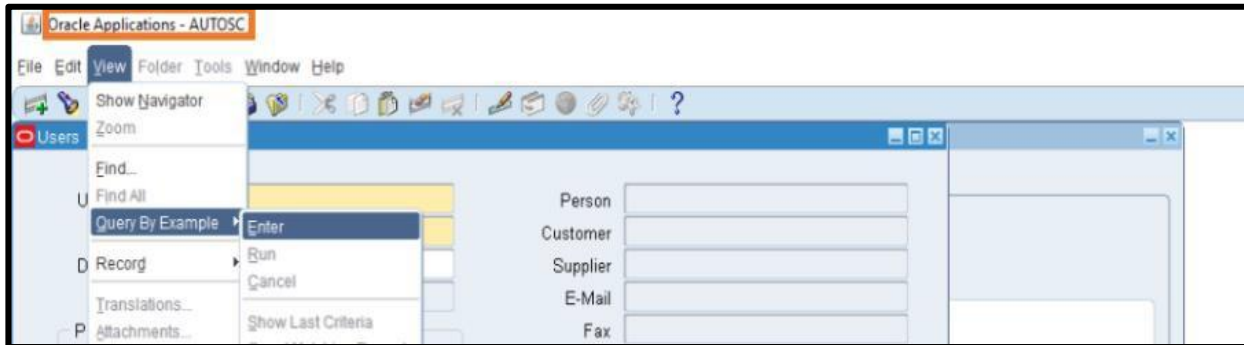
Input Parameters

Argument	Argument Type	Map to Parent	If Default	Default Value
WindowName	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
ApplicationPort	Numeric	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
RadioButtonId	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
FormName	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>

Provide default value or mapping to argument WindowName

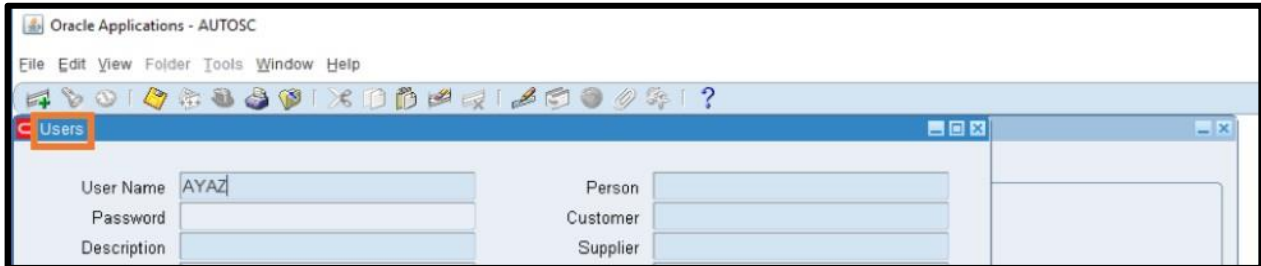
CONFIRM

- **Window Name:** The input for this field is the name of the window under which the Oracle form opens.

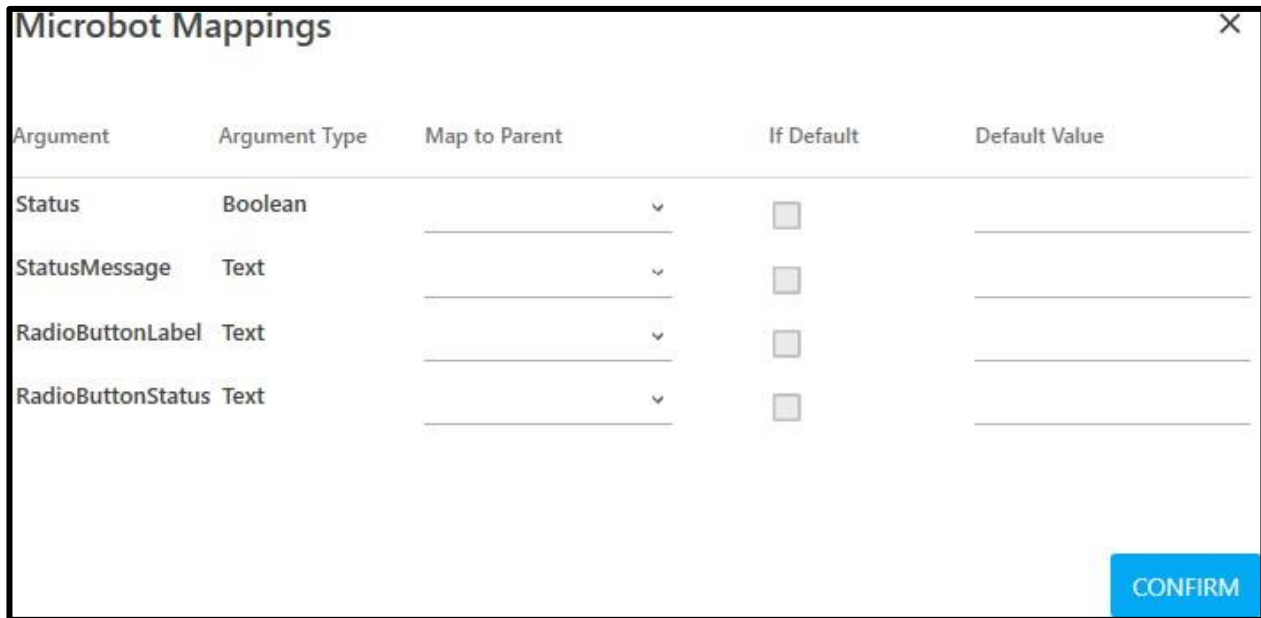


- **ApplicationPort:** This parameter takes the value captured through Port Number field in Open Form microbot's output parameter. Java applet runs on this port and is dynamic.

- RadioButtonID: The input to this field is search the path of the radio button or the checkbox.
- FormName: The Input to this parameter is the name of the form on which the operation is performed.



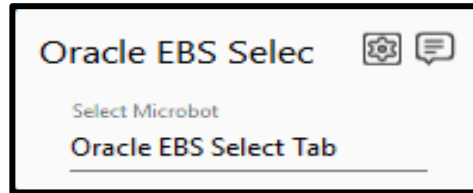
Output Parameters



- Status: This parameter gets the output as either True or false depending on whether the Microbot successfully completed the operation or not.
- StatusMessage: This parameter gets the output as either Success Message or a specific error message depending upon whether Microbot successfully completed the operation or it failed due to an exception.
- RadioButtonLabel: This parameter gets the output as the label of the radio button status is being read.
- RadioButtonStatus: This parameter gets the output as a value of the radio button. If the radio button is checked, then the value returns to be true if the radio button is not checked, the value returns false.

9.22.8 Oracle EBS Select Tab

Automate the selection of a tab in an Oracle form with the help of Oracle EBS Select Tab microbot.



Input Parameters

Argument	Argument Type	Map to Parent	If Default	Default Value
ComponentId	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
TabName	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
PortNo	Numeric	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>

Provide default value or mapping to argument ComponentId

CONFIRM

- Component ID: The input to this parameter is the search path of the tab component to be selected.
- Tab Name: The input to this parameter is the name of the selected tab.
- PortNo: This parameter picks up the value captured through Port Number field in Open Form microbot's output parameter. Java applet runs on this and is dynamic.

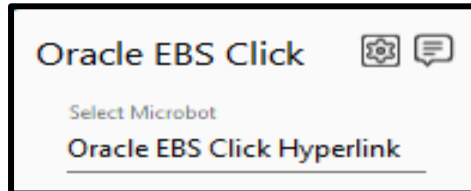
Output Parameters

- Status: This parameter gets the output as either True or false depending on whether Microbot could successfully complete the operation or not.
- StatusMessage: This parameter gets the output as either Success Message or a specific error message depending upon whether Micro-bot successfully completed the operation or it failed due to an exception.

9.22.9 Oracle EBS Click Hyperlink

Automate the “click” on a hyperlink in the Oracle form with the help of Oracle EBS Click Hyperlink microbot.

This capability is used along with the get text capability. If a form field contains a hyperlink which is already read using another bot, then Oracle EBS Click hyperlink microbot it utilized to launch the hyperlink.



Argument	Argument Type	Map to Parent	If Default	Default Value
Url	Text	<input type="checkbox"/>	<input type="checkbox"/>	

Provide default value or mapping to argument Url

CONFIRM

Input Parameters

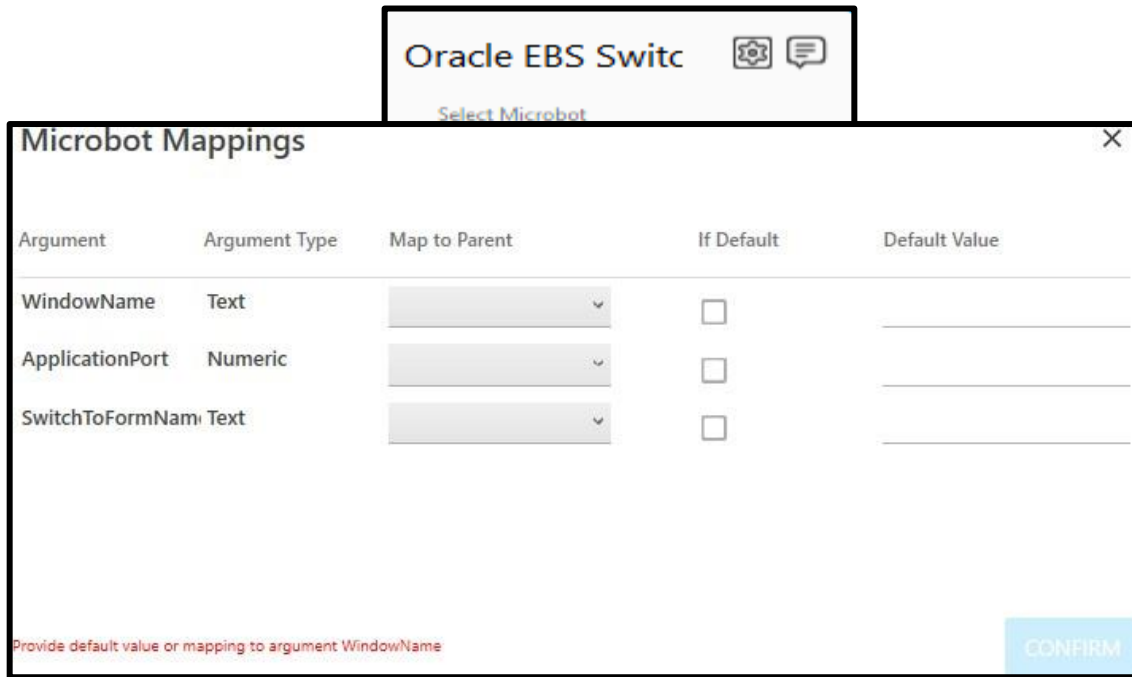
URL: The input to this parameter is the URL/hyperlink which is launched.

Output Parameters

- Status: This parameter gets the output as either True or false depending upon whether the Microbot successfully completed the operation or not.
- StatusMessage: This parameter gets the output either as a Success Message or a specific error message depending on whether the Microbot successfully completed the operation or failed due to an exception.

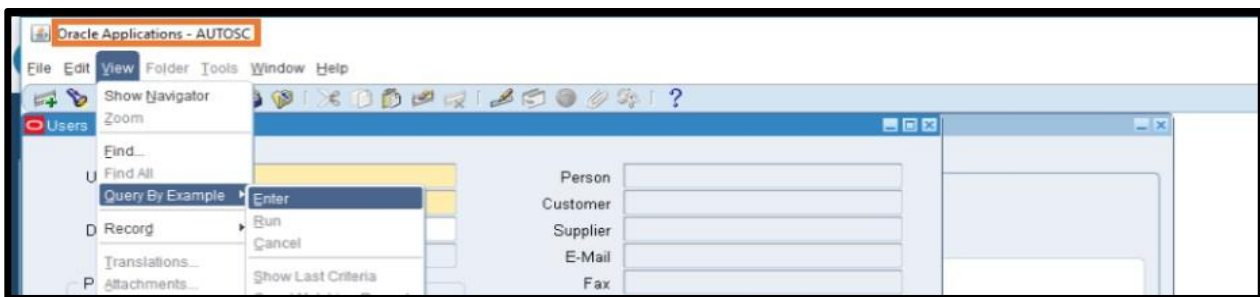
9.22.10 Oracle EBS Switch Form

Automate the switching of Oracle Forms with the help of Oracle EBS Switch Form microbot.



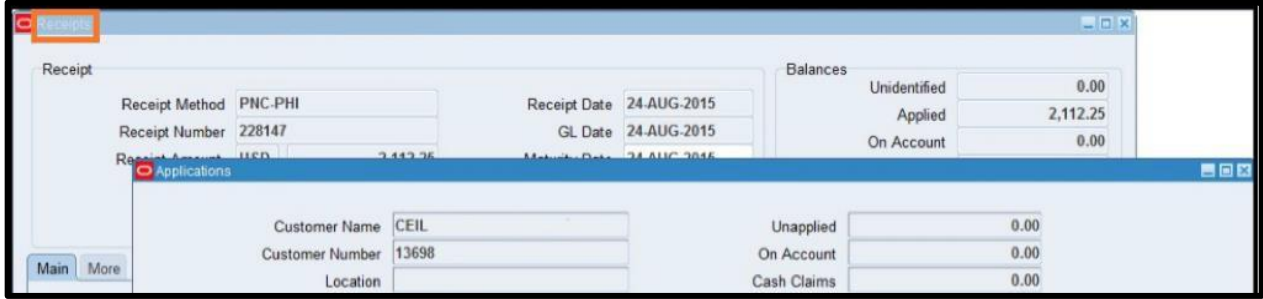
Input Parameters

- **Window Name:** The input for this field is the name of the window under which the Oracle form opens.



- **ApplicationPort:** This parameter takes the value captured through Port Number field in Open Form microbot's output parameter. Java applet runs on this port and is dynamic.

- **SwitchToFormName:** The input to this parameter is the name of the form where control is transferred.

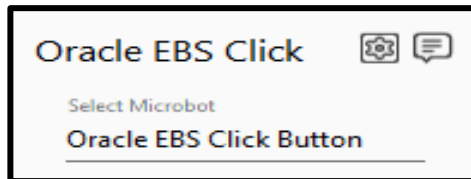


Output Parameters

- **Status:** This parameter gets the output as either True or False depending upon whether the Microbot successfully completed the operation or not.
- **StatusMessage:** This parameter gets the output as either Success Message or a specific error message depending upon the Microbot successfully completed the operation or failed due to an exception.

9.22.11 Click a Button

Automate the clicking of a button in an Oracle form with the help of Oracle EBS Click Button microbot.



Argument	Argument Type	Map to Parent	If Default	Default Value
ComponentId	Text	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
PortNo	Numeric	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>

Provide default value or mapping to argument ComponentId

CONFIRM

Input Parameters

- ComponentId: The input to this parameter is the oform ID of the button component.
- PortNo: This parameter takes the value, captured through Port Number field in Open Form microbot's output parameter. Java applet runs on this port and is dynamic.

Output Parameters

- Status: This parameter gets the output as either True or false depending upon whether the Microbot successfully completed the operation or not
- StatusMessage: This parameter gets the output as a Success Message, or a specific error message depending upon if the Microbot completed the operation, or it failed due to an exception.

9.22.12 Oracle EBS Start Mapping Server

Automate the start mapping server with Oracle EBS Start Mapping Server microbot. This Microbot is a prerequisite for toolbar and menu options capability and must be placed before the Oracle EBS Open Form Microbot in a process. There are no input parameters for this Microbot.

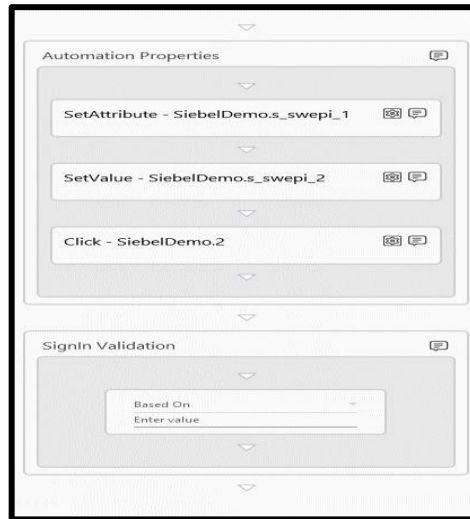
9.22.13 Oracle EBS Stop Mapping Server

Automate stopping of the mapping server with the help of Oracle EBS Stop Mapping Server microbot. This Microbot must be placed at the end of a process. There are no input parameters for this Microbot.

9.23 Automation of Siebel HI Application using Microbots

Oracle Siebel CRM tool comes in two variants – High Interactivity, and Open UI. The high interactivity renders the user interface as a collection of HTML frames and uses ActiveX controls.

Launching and signing in to the Siebel application is done using the existing Studio's SignIn Process functionality with the Siebel HI web application.



AE provides a basic capability to support automation of Siebel CRM HI application to configure business flow through Microbots.

The underlying system of this Microbot development is Siebel IP15 version, high interactive UI, and ecommunications module.

9.23.1 SiebelHINav

This Microbot is used to navigate to screen tabs, link bars, and views of Siebel HI variant via sitemap. Siebel high interactive UI doesn't provide controls for tabs and link bars.

Therefore, Site Map is used for navigation. Sitemap does not have unique id or labels; hence an index of occurrence is provided along with id or label as input.

Following are the details of the parameters to be used while configuring this Microbot:

Input Parameters

- InputURL: This is string data type. Provide the URL to get the IE instance on which the Microbot runs.
- NavControlId: This is string data type. Provide the HTML control Id of the link of the sitemap page. E.g., to navigate to My Accounts link of Accounts tab, get the
- HTML control ID of Accounts -> My Accounts link in site map (E.g., s_smi_0148). The index parameter is used, in

case of same control IDs for different links.

- NavTabName: This is string data type. Provide the label (tab name, link bar name, or view name) of the link present on the sitemap page.
- Index: This is an integer data type. Index is used for multiple occurrences of the same HTML control ID or labels on the sitemap page.

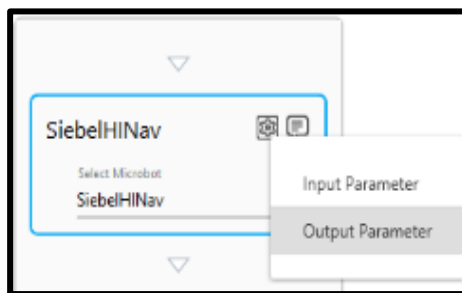
The default value is 1, which indicates the first occurrence of the control ID.

Argument	Argument Type	Map to Parent	If Default	Default Value
InputURL	Text	<input type="text"/>	<input checked="" type="checkbox"/>	http://evblaeseapw491/ec
NavControlId	Text	<input type="text"/>	<input checked="" type="checkbox"/>	s_smi_046
Index	Numeric	<input type="text"/>	<input checked="" type="checkbox"/>	5
NavTabName	Text	<input type="text"/>	<input checked="" type="checkbox"/>	Service Requests - All Serv

CONFIRM

Output Parameters

- OutURL: This is string data type. It provides the active URL at the completion of activity of navigating to the desired location.
- NavSuccess: This is a Boolean data type which confirms whether the navigation activity was successful.



9.23.2 SiebelHIUIButtonClick

This Microbot is used to click on a particular button in web-dialog of Siebel HI variant (like Query, Go, Next, Complete, etc.). Pop-ups and clicking on any grid element are currently not supported.

Following are the details of the parameters to be used while configuring this Microbot:

Argument	Argument Type	Map to Parent	If Default	Default Value
OutURL	Text	URL <input type="checkbox"/>	<input type="checkbox"/>	
NavSuccess	Boolean	StatusSuccess <input type="checkbox"/>	<input type="checkbox"/>	

Input Parameter

- InURL: This is an input parameter with string data type. Provide the URL to get the IE instance on which the Microbot runs. It is either taken from the output of previous Microbot or uses a generic static variable declared in the process.
- ControlID: This is string data type. Provide the HTML control Id of the element to be clicked.
- Label: This is an input parameter with string data type. Provide the label of the button (like Menu, New, and Query, etc.) to be clicked. If ControlId and labeled both are provided, ControlId gets the preference.

In case of duplicate labels in a web-dialog, provide an index of occurrence.

- Index: This is an input parameter with integer data type. It is used for multiple occurrences of the same HTML control ID or labels on sitemap page exist.



The default value is 1, indicating the first occurrence of the control ID.

Output Parameters

- OutURL:** This is an output parameter with string data type. It provides the active URL at the completion of the activity (clicking on the desired button).

Argument	Argument Type	Map to Parent	If Default	Default Value
OutURI	Text	URL	<input type="checkbox"/>	
InURI	Text	URL	<input type="checkbox"/>	
Objectid	Text		<input checked="" type="checkbox"/>	s_2_1_13_0_mb
ButtonName	Text		<input checked="" type="checkbox"/>	Query
Index	Numeric		<input checked="" type="checkbox"/>	1

9.23.3 SiebelHIUIGetGridCount

This Microbot is used to expand the grid and get the number of records in that grid in the Siebel high interactive UI.

Any operation on a grid requires stitching of two Microbots; first Microbot to get the number of columns in the grid, and another microbot to perform operations like selecting the desired row or modifying a particular column (parameter) in the row. This Microbot is used to identify the count of rows in a grid.

Following are the details of the parameters to be used while configuring this Microbot:

Input Parameter

- **InURL:** This is a string data type. Provide URL to get the IE instance on which the Microbot runs. It either takes from the output of previous Microbot or uses a generic static variable declared in the process.
- **ControlID:** This is a string data type. Provide control ID of the element which does the paging activity. Control ID is the paging control object ID on the top right corner of the grid to be traversed.

Argument	Argument Type	Map to Parent	If Default	Default Value
InURI	Text	URL	<input type="checkbox"/>	
ExpandCtrlId	Text		<input checked="" type="checkbox"/>	s_2_1_18_0_mb

Output Parameters

- **RecCount:** This is an output parameter with the numeric data type. It provides the number of records (count of rows) in the selected grid at the completion of the activity.

Argument	Argument Type	Map to Parent	If Default	Default Value
RecCount	Numeric	RowCount	<input type="checkbox"/>	

9.23.4 SiebelHIUIUpdateGridCell

This Microbot is used in selecting specific row/record in a grid and updating a particular column (parameter) in a grid of Siebel high interactive UI.

Any operation on a grid requires stitching of two Microbots; first Microbot to get the number of columns in the grid, and another microbot to perform operations like selecting the desired row or modifying a particular column (parameter) in the row.

The SiebelHIUIGetGridCount Microbot is used to identify the count of rows in a grid, and this Microbot is used to perform the operation of selecting the desired row or modifying a particular column (parameter) in the row.

Following are the details of parameters to be used while configuring this Microbot:

Input Parameter

- InURL: This is a string data type. Provide URL to get the IE instance on which the Microbot runs. The URL is either taken from the output of previous Microbot or uses a generic static variable declared in the process.
- GridID: This is a string data type. Provide the object ID/control ID of the desired grid.
- RecCount: This is a numeric data type. It provides the number of records (count of rows) in the selected grid at the completion of the activity. This field is taken from the output of the SiebelHIUIGetGridCount Microbot.
- SearchColNo: This is an integer data type. Provide the column number (place of the parameter from extreme left) in the grid, where the desired value of that parameter is searched (search criteria). This must be the column number of the primary key. The numbering starts from 1.
- SearchCriteria: This is a string data type. Provide the value to be searched in a particular column (parameter) of the grid to get the desired record in the list. If the column (parameter) has more than one row with the same value (search criteria), then it always considers the first record found.
- UpdateColNo: This is an integer data type. Provide the column number (place of the parameter from extreme left) in a grid whose value needs to be updated. If this parameter is not provided, the operation is performed on searchColNo.
- UpdateValue: This is a string data type. Provide the value to be updated in the desired cell (UpdateColNo). If this parameter is not provided, the row is selected.

Argument	Argument Type	Map to Parent	If Default	Default Value
SearchCriteria	Text		<input checked="" type="checkbox"/>	DSSDC
GridId	Text		<input checked="" type="checkbox"/>	s_2_1
UpdateValue	Text		<input checked="" type="checkbox"/>	
RecCount	Numeric	RowCount	<input type="checkbox"/>	
UpdateColNo	Numeric		<input checked="" type="checkbox"/>	3
SearchColNo	Numeric		<input checked="" type="checkbox"/>	3

CONFIRM

Output Parameters

- RecordFound: This is an output parameter with the numeric data type. It provides the value 0 or 1 depending on the success of search criteria. If no record is found, this value is "0."
- CellData: This is an output parameter with text data type and provides the search criteria value.

Argument	Argument Type	Map to Parent	If Default	Default Value
RecordFound	Numeric	RecFound	<input type="checkbox"/>	
CellData	Text	data_1	<input type="checkbox"/>	

9.23.5 SiebelHIWebDialogUpdate

This Microbot is used to automate setting the value of a particular parameter/element in web-dialog in the Siebel high interactive UI.

Following are the details of the parameters to be used while configuring this Microbot:

- InURL: This is an input parameter with string data type. Provide URL to get the IE instance on which the Microbot runs. It is either taken from the output of previous Microbot or uses a generic static variable declared in the process.
- Controllid: This is an input parameter with string data type. Provide the HTML control Id of the element which needs to be updated.
- Label: This is an input parameter with string data type. Provide the label of the parameter updated. If Controllid and label both are provided, Controllid gets preference. In case of duplicate labels in a web-dialog, provide an index of occurrence.
- Index: This is an input parameter with integer data type. It is used when there are multiple occurrences of the same HTML control ID or labels on the site map page. If the default value is 1, it is the first occurrence of the control ID.
- InputValue: This is an input parameter with string data type. It takes the value to be put in the selected parameter to update it.

Argument	Argument Type	Map to Parent	If Default	Default Value
Label	Text		<input checked="" type="checkbox"/>	Account:
Index	Numeric		<input checked="" type="checkbox"/>	1
InURL	Text	URL	<input type="checkbox"/>	
InputValue	Text		<input checked="" type="checkbox"/>	DSSDC
Controllid	Text		<input checked="" type="checkbox"/>	


9.23.6 SiebelHIReadWebDialogElement

This Microbot is to automate reading the value of a particular parameter/element in web-dialog in the Siebel high interactive UI.

Following are the details of parameters to be used while configuring this Microbot:

Input Parameters

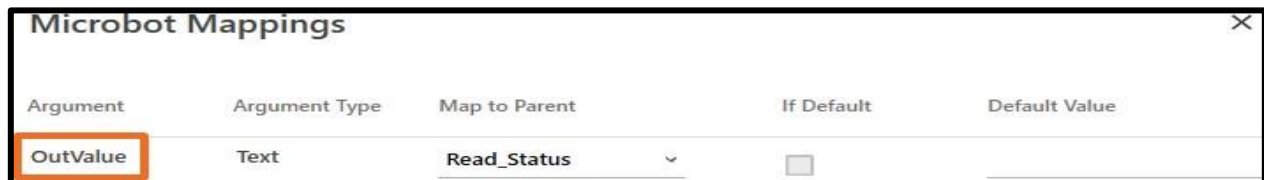
- **InURL:** This is an input parameter with string data type. Provide URL to get the IE instance on which the Microbot runs. It is taken either from the output of previous Microbot or uses a generic static variable declared in the process.
- **ControllId:** This is an input parameter with string data type. Provide the HTML control Id of the element/parameter to read the values.
- **Label:** This is an input parameter with string data type. Provide the label of the parameter whose value needs to be read. If ControllId and label both are provided, ControllId gets the preference. In case of duplicate labels in a web-dialog, provide an index of occurrence.
- **Index:** This is an input parameter with integer data type. It is used when multiple occurrences of the same HTML control ID or labels on sitemap page exist. If the default value is 1, it is the first occurrence of the control ID.



Argument	Argument Type	Map to Parent	If Default	Default Value
Label	Text		<input checked="" type="checkbox"/>	Status:
Index	Numeric		<input checked="" type="checkbox"/>	1
InURL	Text	URL	<input type="checkbox"/>	
ControllId	Text		<input checked="" type="checkbox"/>	

Output Parameters

- **OutValue:** This is an output parameter with string data type. It provides the value captured for the desired parameter.



Argument	Argument Type	Map to Parent	If Default	Default Value
OutValue	Text	Read_Status	<input type="checkbox"/>	

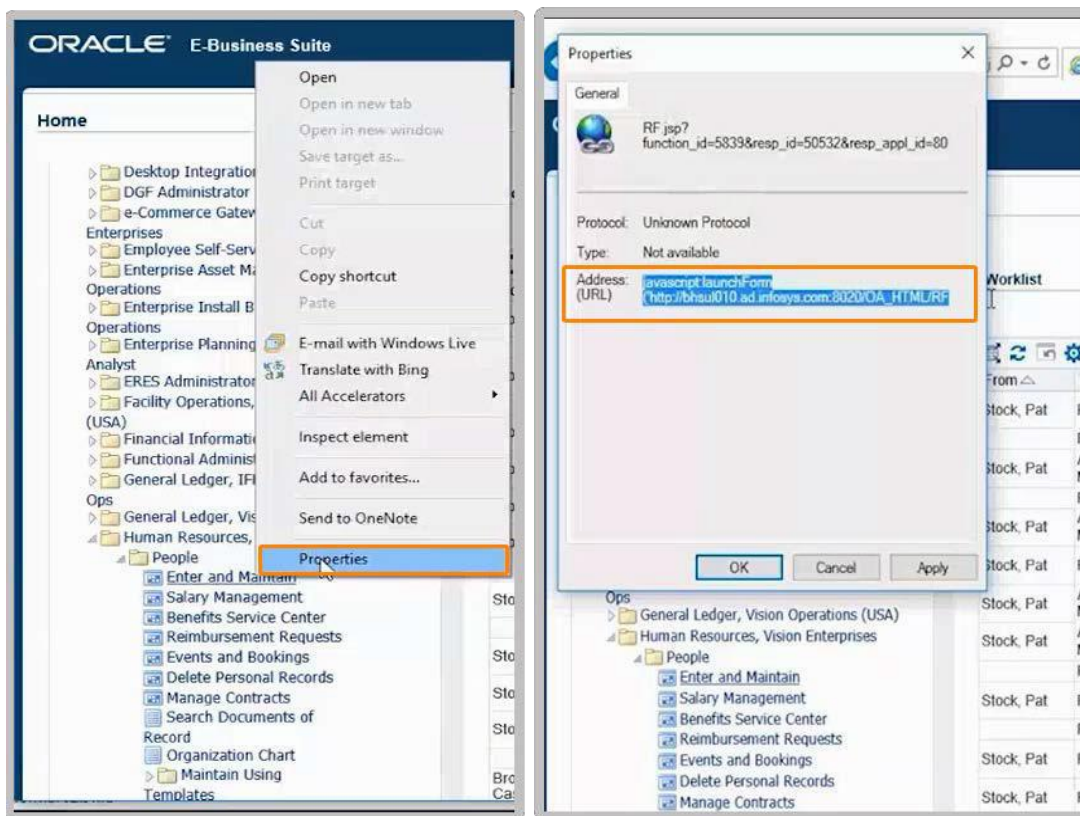
9.24 Oracle EBS Codeless Configuration

AssistEdge RPA Studio supports automation of the Java-based Oracle Forms in a codeless manner throughout of the box adapters. Following are the available configurations.

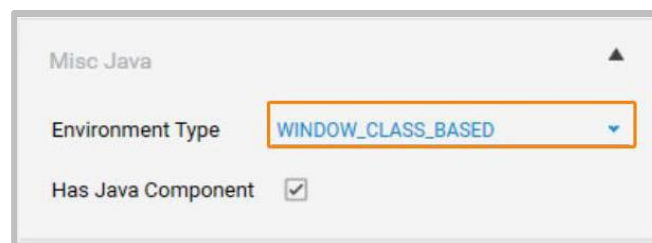
9.24.1 Launch an Oracle EBS Form through Web Navigator

Automate the opening of an Oracle form through Oracle EBS web navigator. Following are the steps to perform the automation.

- Add an application specific to the Oracle form in the Admin-->Applications section of AssistEdge Automation Studio.



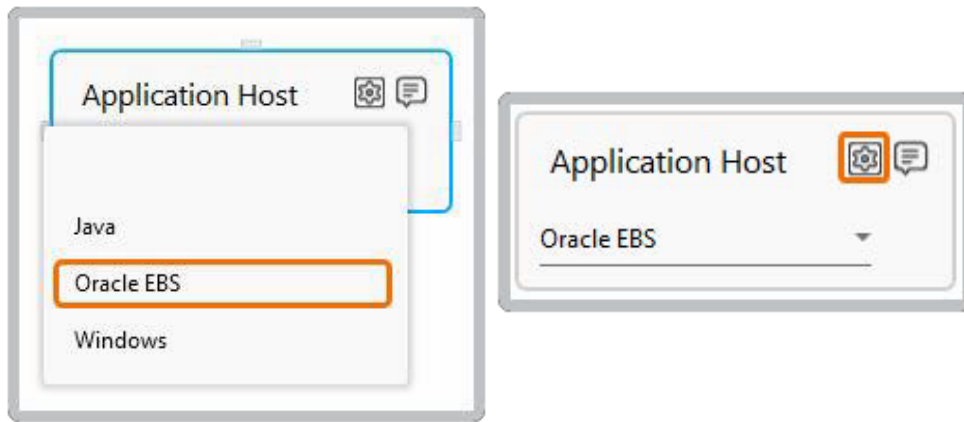
- URL of a specific form is extracted from the properties of the form from the web navigator.
- Look for “Misc Java” properties, while defining the properties of the application.



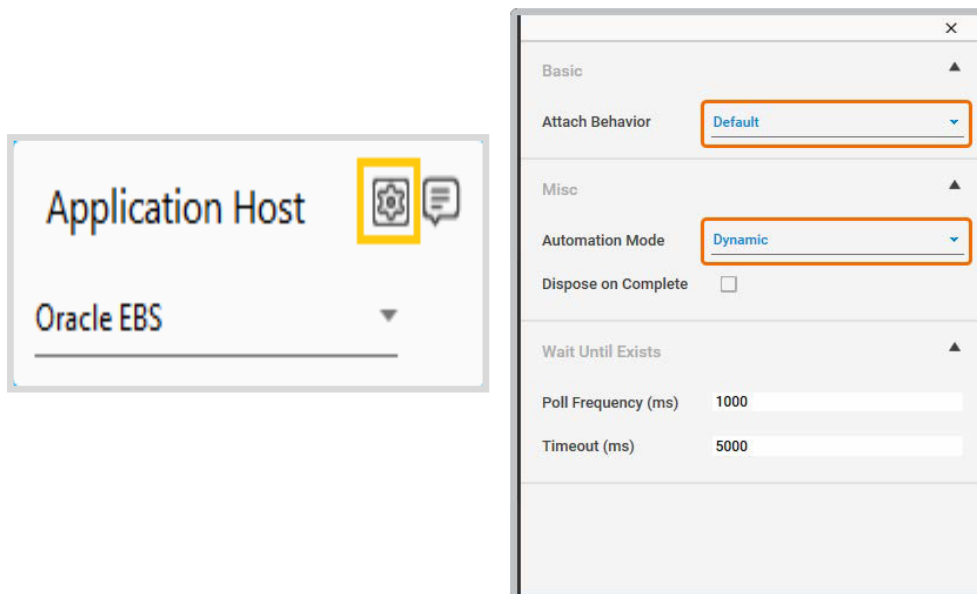
- Select the “Has Java Component” Check-box.
- Select the correct value for environment type from the drop-down.
- Save the application.
- The saved application is used as part of the process. The specific form is launched during the Setup Environment.

9.24.2 Application Host - Setting Up the Oracle EBS Window for Automation

The “Application Host” activity in the Process Components section is used to set up the form for automation. The Application Host activity must be inside the “Application” Activity.

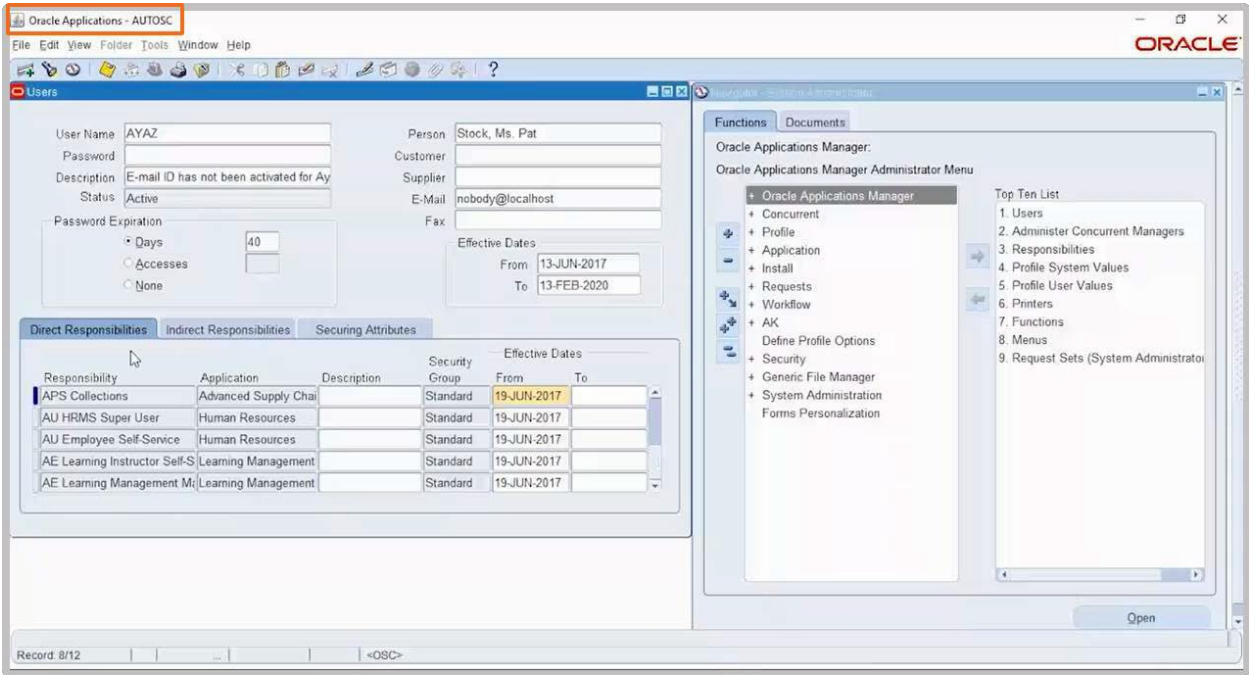


- Select Oracle EBS from the drop-down.
- Click on the settings options of the Application. Click on setup environment for Oracle and Siebel applications. Once the environment set up is complete, use the activity automation configuration for the launched application of Oracle and Siebel. In the lower versions of Automation Studio, the settings button performs this functionality.



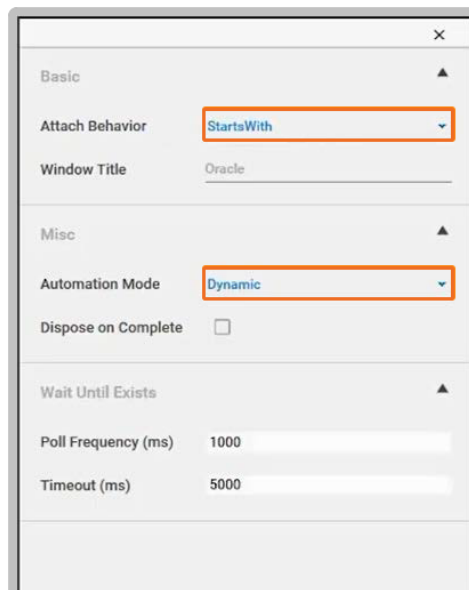
- Attach Behavior is initially set to “Default”. Change to a different option such as “Equals”, “Contains”, “Starts with”, or “Ends with” for improved accuracy.

E.g. In the form, the window name is Oracle Applications – AUTOSC. As an example, select Attach behavior as “Starts with” and



Window Title as “Oracle”.

- The Automation Mode is selected as Dynamic or static depending upon the type of component ID to be extracted while performing the automation.

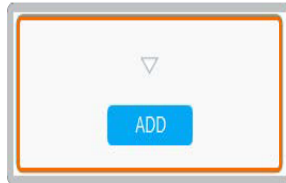


9.24.3 Application Control – Basics

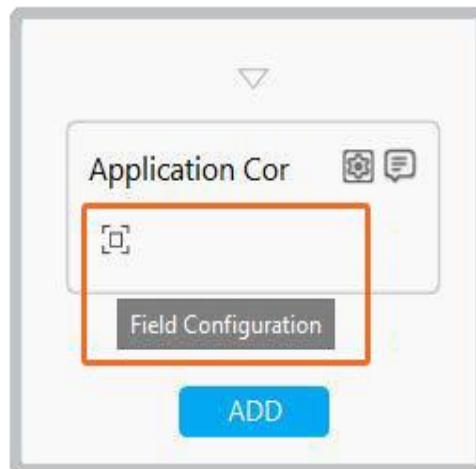
The “Application Control” activity is used to automate the actions on Oracle EBS forms.

Following are few of the key points for using this activity.

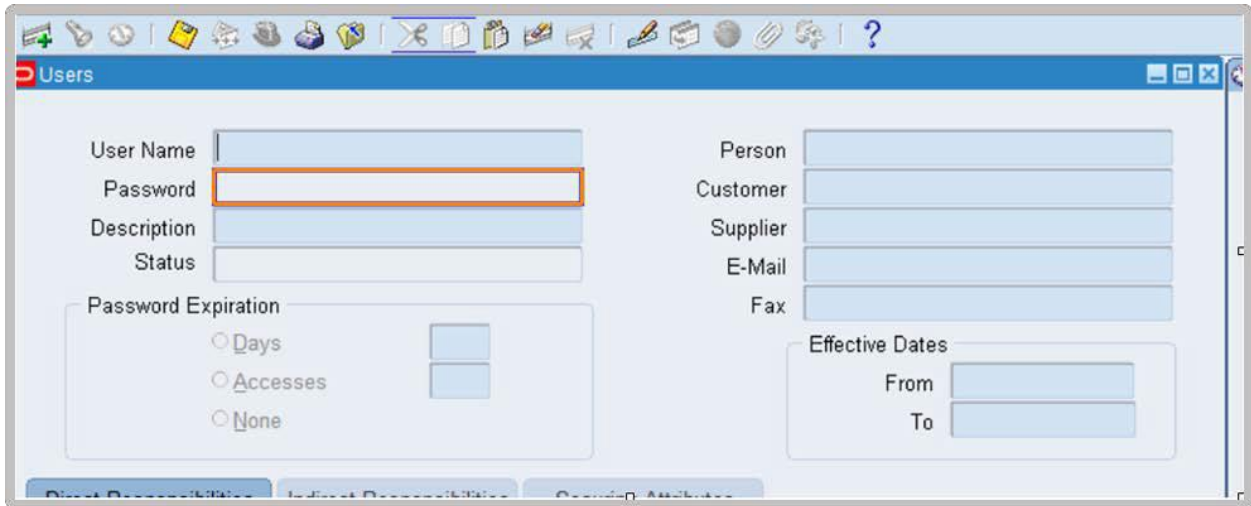
- The “Application Control” activity must be inside the “Application Host” activity.
- The “Application Control” activity is added by using “Add” button inside the “Application Host” activity.



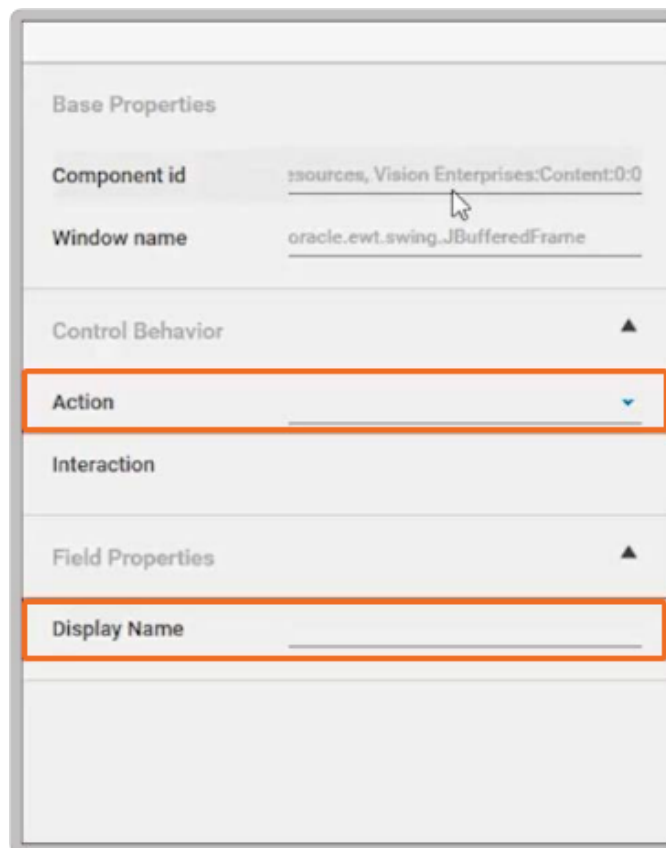
- Drag and drop the “Application Control” activity from “Process Components” to use it within other components like “Excel loop”.
- Click on “Field Configuration” button to start extracting the component to be automated.



- After clicking on the Field configuration button, point the cursor to the component to be automated. The component is highlighted in blue while it is being pointed at. E.g., Password field in the form.

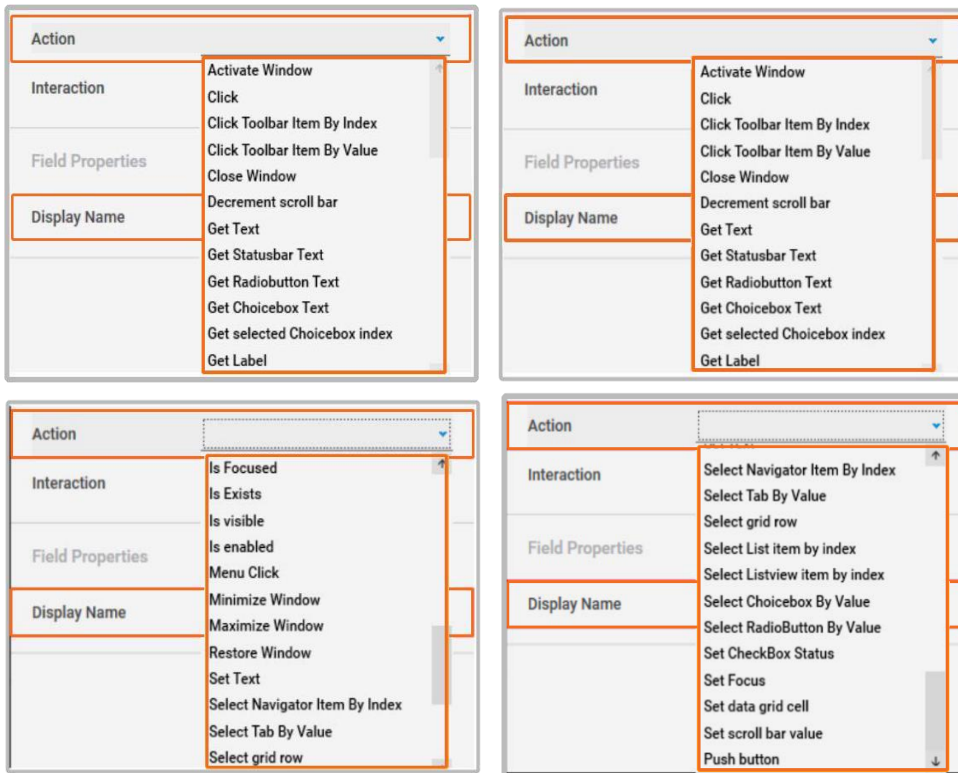


- Click on the highlighted component and, the ID of the component is extracted and the settings pop-up opens over the "Application Control" activity.



The static part of component ID is replaceable with an argument to use a dynamic value as part of the component ID. Right click on the component ID field to view and select the available arguments.

- Select one of the available actions from the drop-down.




- Once the action button is selected, fill the rest of the fields which appears as per the selected action.

9.24.4 Launch an Oracle EBS Form using Java-based Navigator

Automate the opening of an Oracle form through Oracle EBS Java-based navigator.

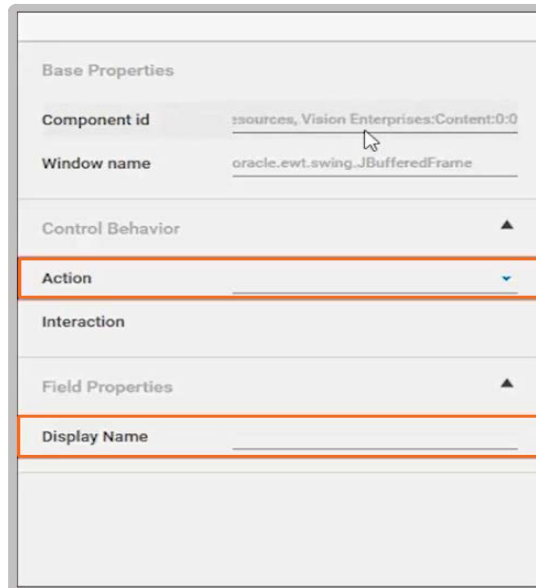
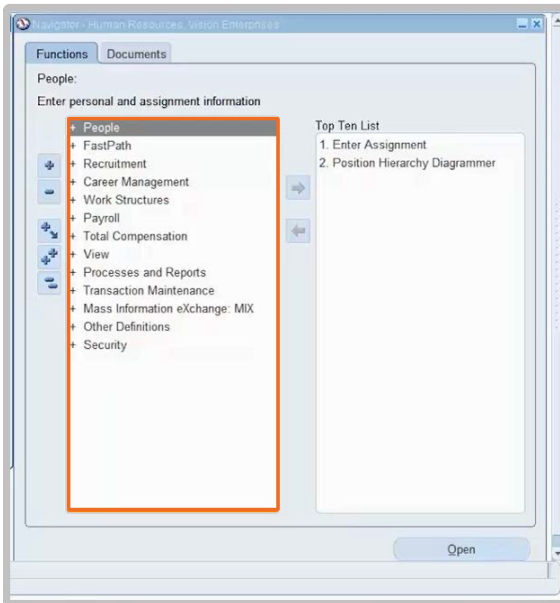
Following are the steps involved in the automation.

- Add an "Application Control" Activity to the flow.
- Click on the Field Extraction  button to extract the ID of the navigator.

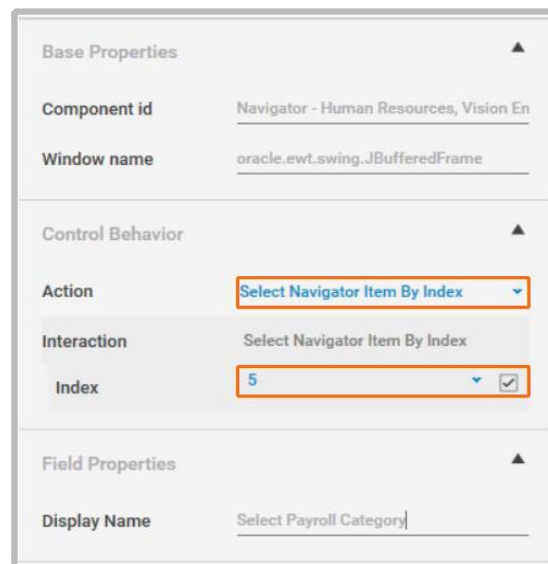
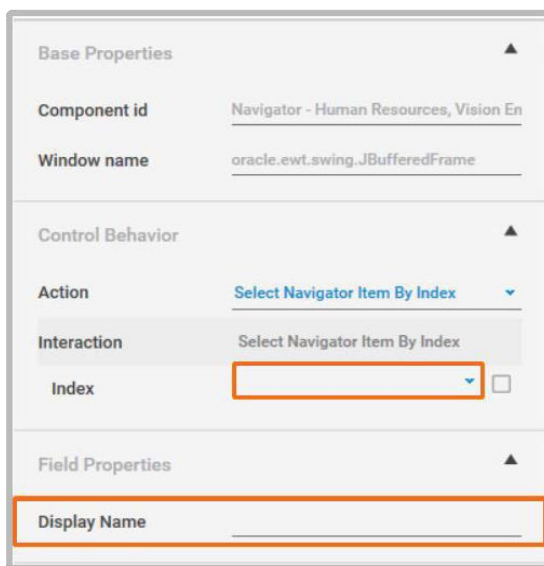
Scenario: Open Payroll >> Payment Methods

Solution:

- Select the navigator area

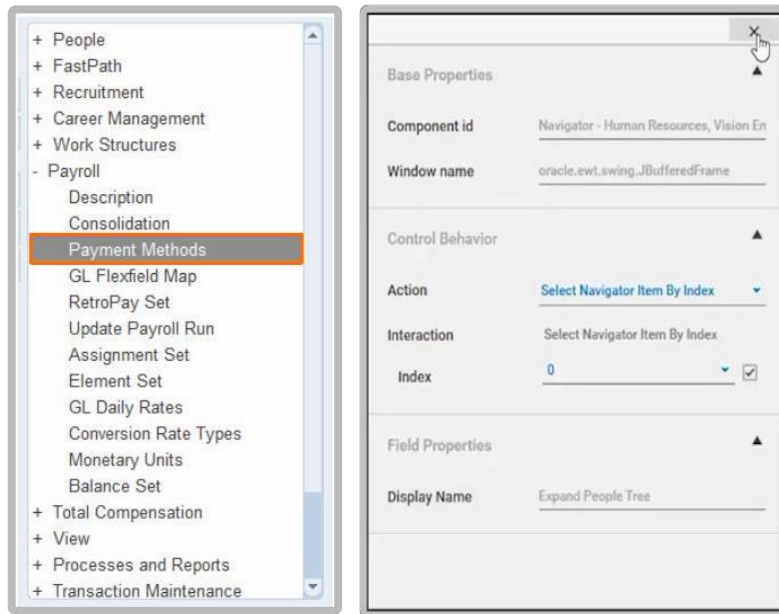


- Once the component ID is captured, select the action as "Select Navigator Item by Index".

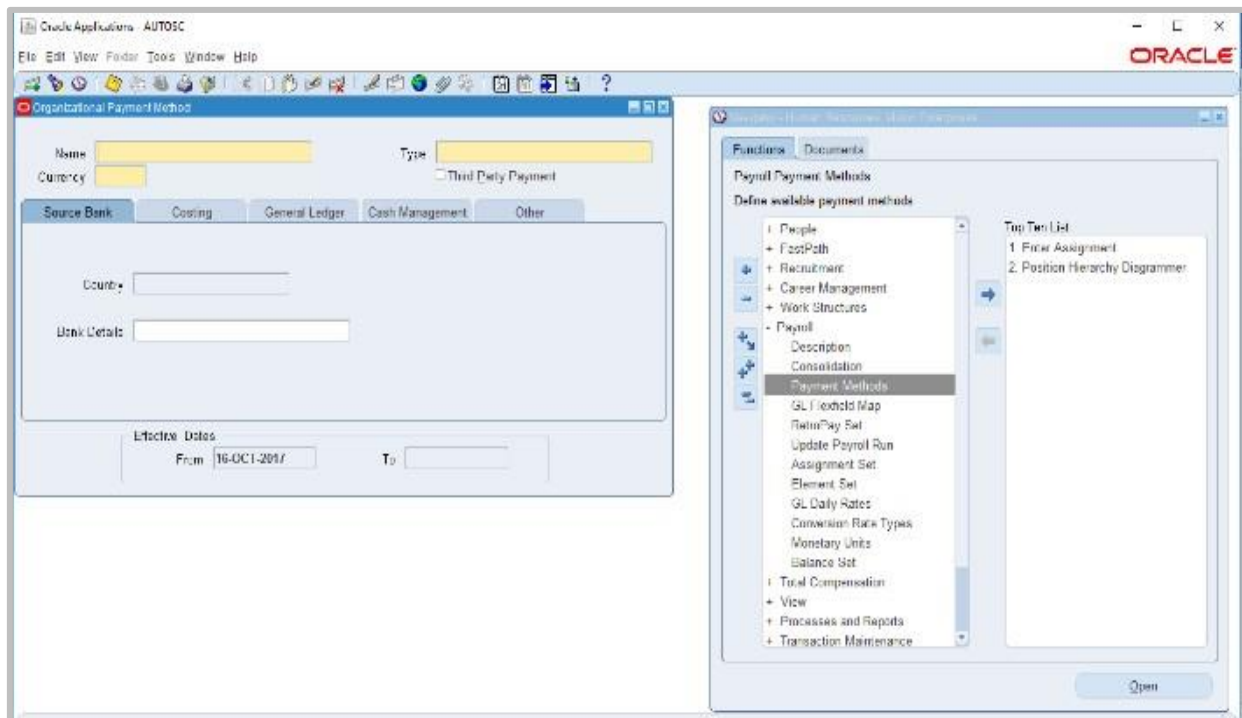


- Enter 5 as a value in the index field as the index for "Payroll" option is "5".
- Enter an appropriate "Display Name" for the activity.
- Configure automation for clicking the open button.

- Configure selecting “Payment Methods” option from navigator in the way similar to “Payroll” option.




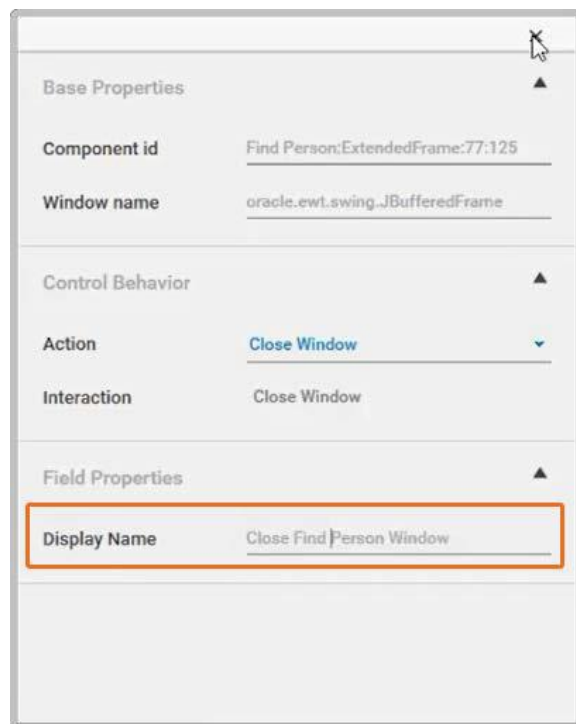
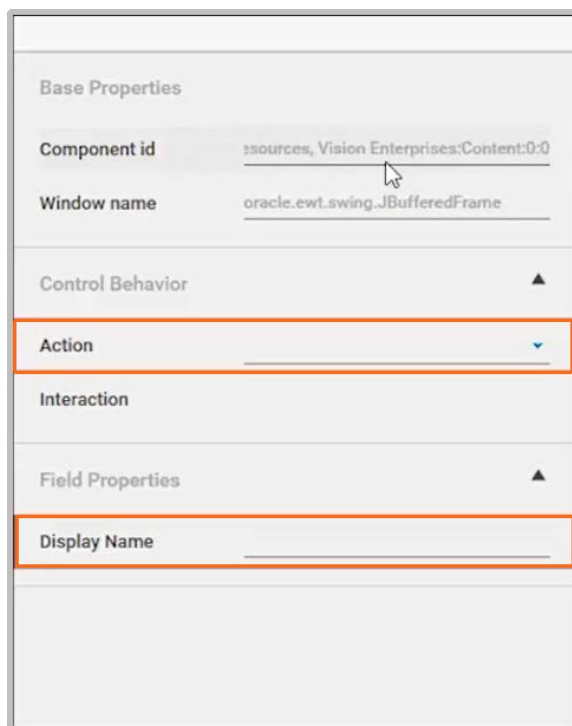
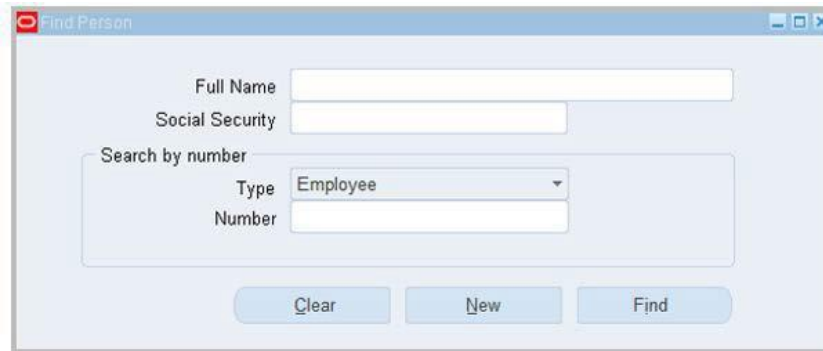
- Configure automation for clicking the open button.
- Execution of the above steps results in opening the Payroll Payment form.



9.24.5 Close, Minimize or Maximize the Form


Automate the minimizing, maximizing and closing of an Oracle EBS form through the following steps:

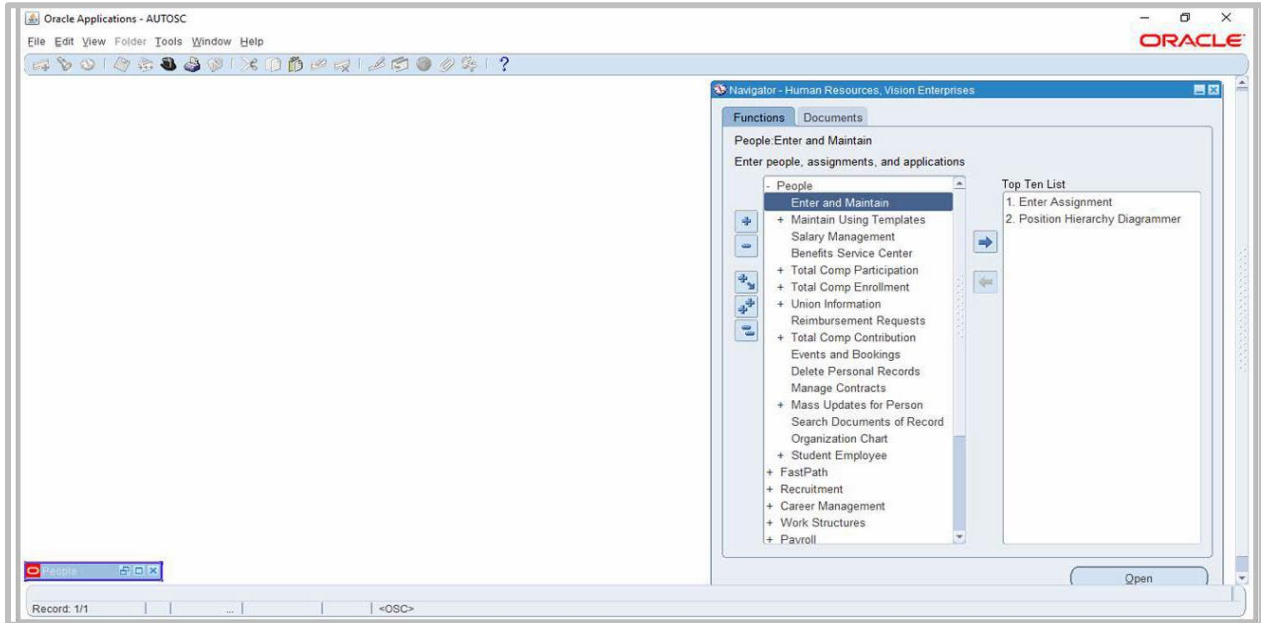
- Add an **"Application Control"** Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.



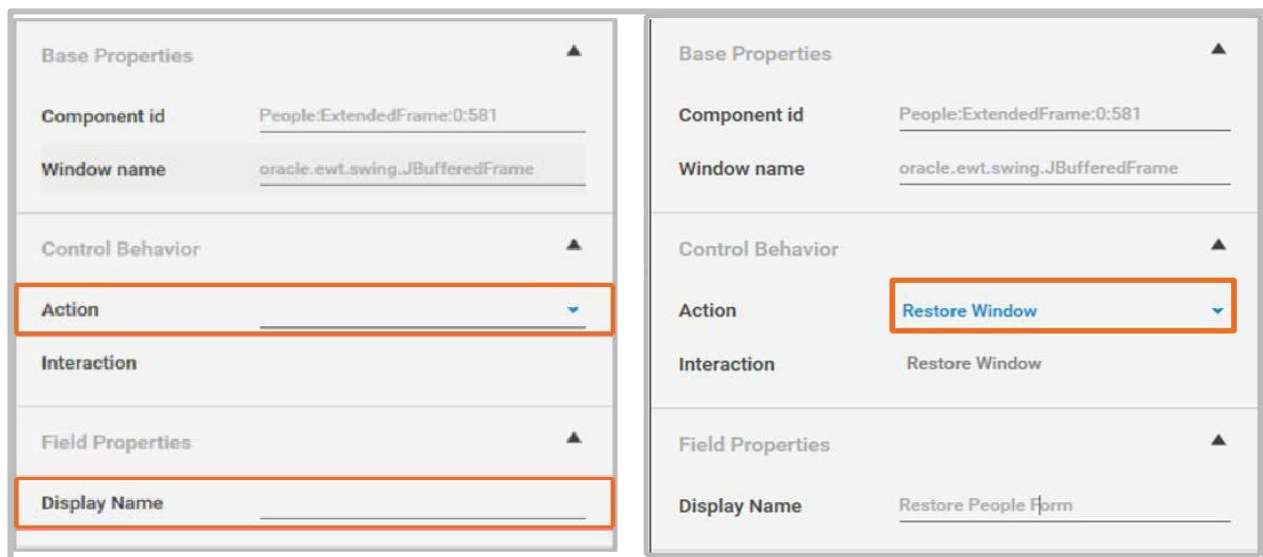
- Point the mouse to the window at which the action must be taken to extract its component ID.
- Select the relevant action to be taken and enter an appropriate display name.
- Execution of the above steps results in closing the Oracle EBS form.

Automate the restoring of a minimized Oracle EBS form through the following steps.

- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the window at which the action must be taken to extract its component ID.




- Select the relevant "action" to be taken and enter an appropriate display name.

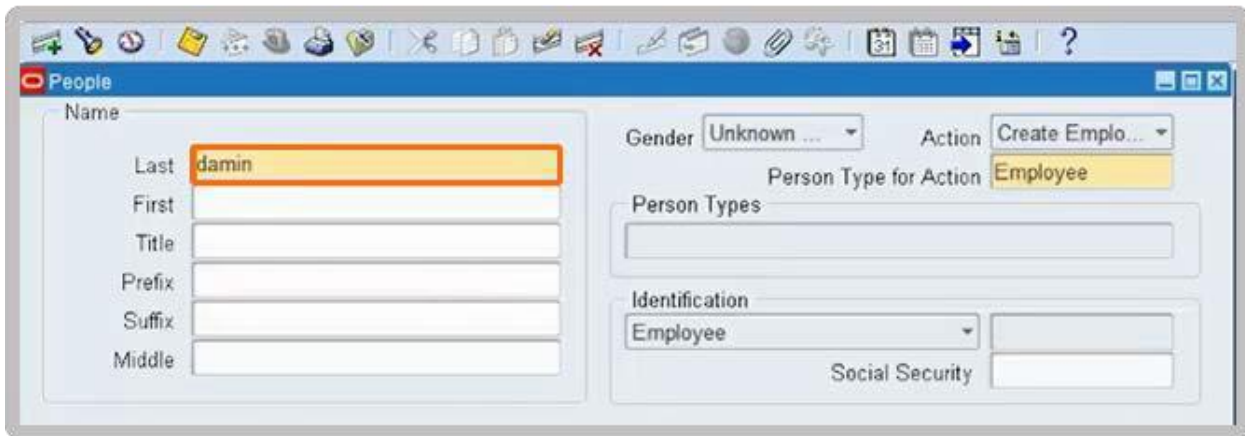


- Execution of the above steps results in restoring the form.

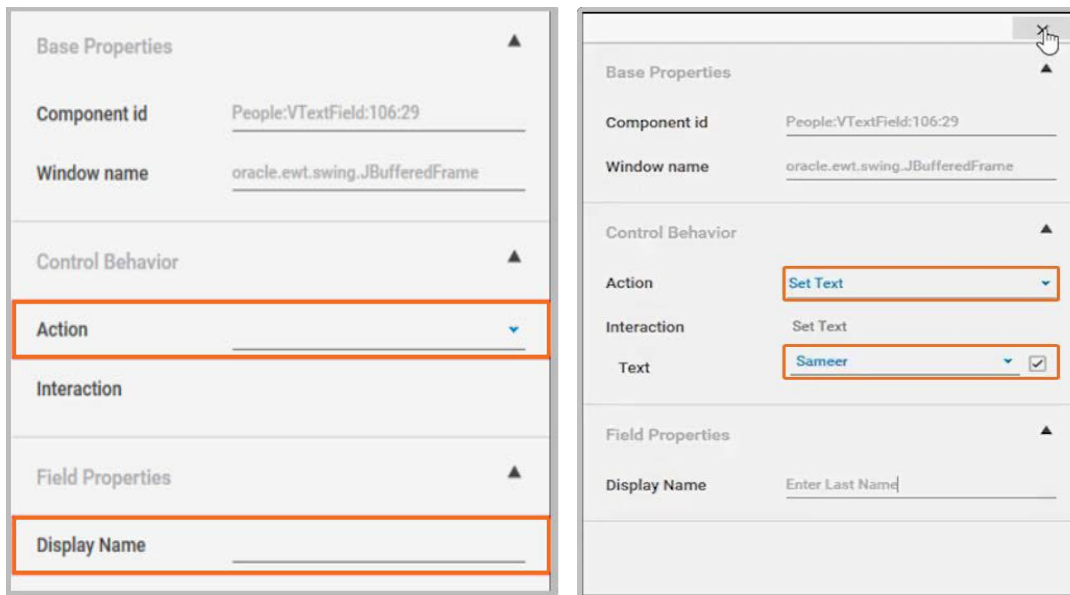
9.24.7 Set Text

Automate the setting of the text in an Oracle EBS form through the following steps.

- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the component at which the action must be taken, to extract its component ID.




- Select the relevant action to be taken from the list of values and enter an appropriate display name.

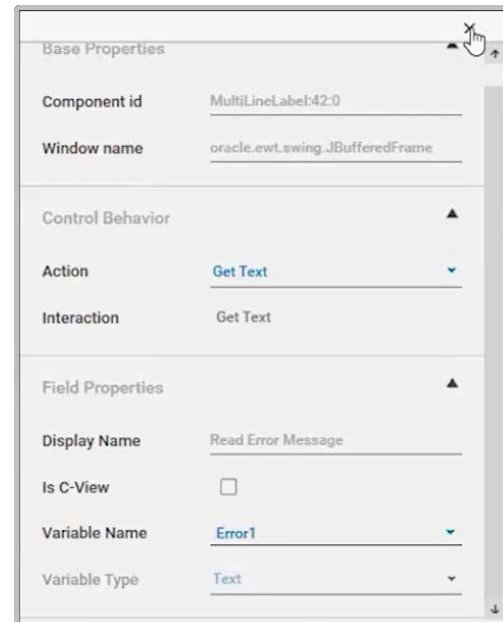
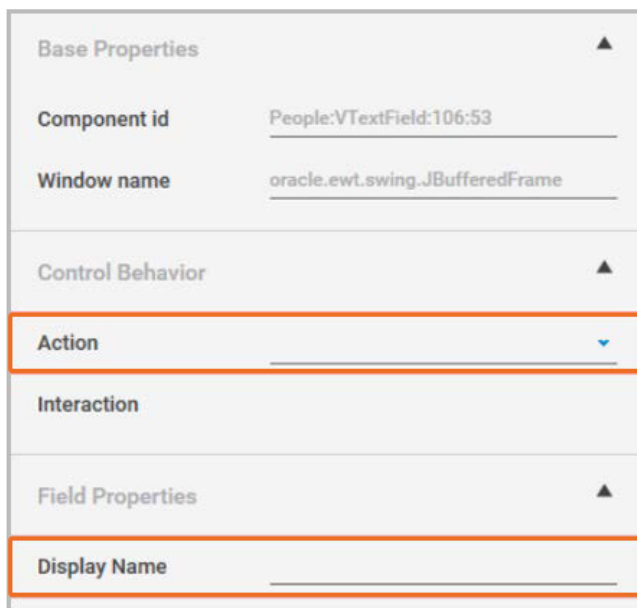
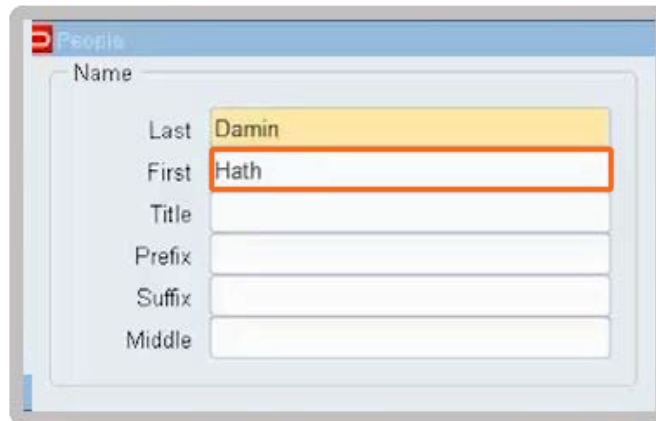


- Execution of the above steps results in setting the text in the corresponding text box.

9.24.8 Get Text

Automate reading of the text in an Oracle EBS form through the following steps.


- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the component at which the action must be taken, to extract its component ID.

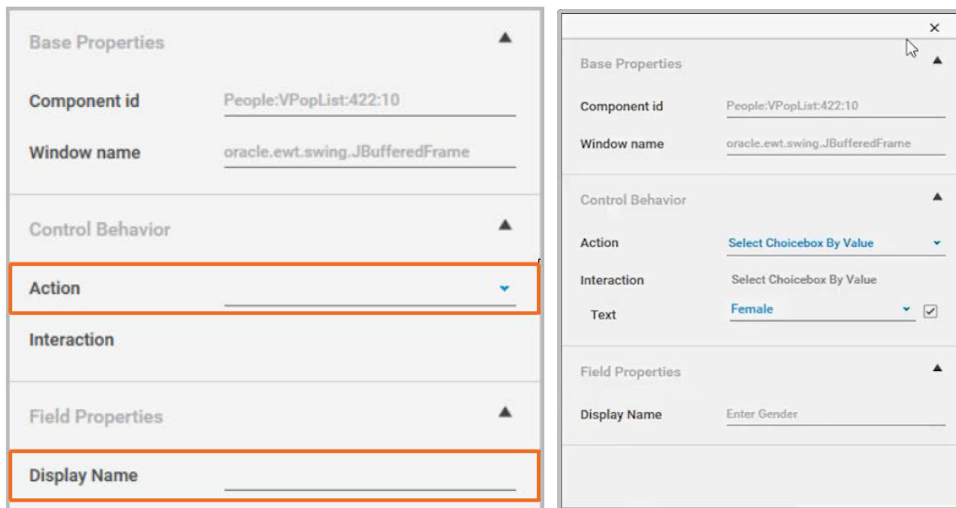
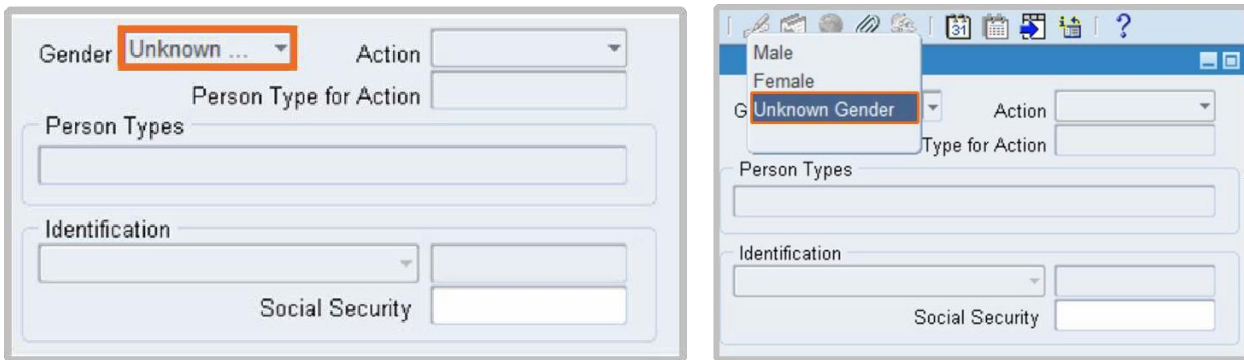


- Select the relevant action to be taken, variable name, variable type and enter an appropriate display name.
- Execution of the above steps results in reading the text from the corresponding field in an Oracle EBS form.

9.24.9 Select Value from the Choice box/Drop-down

Automate the value selection from choice-box/dropdown in an Oracle EBS form by following the below mentioned steps:


- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the component at which the action must be taken, to extract its component ID.

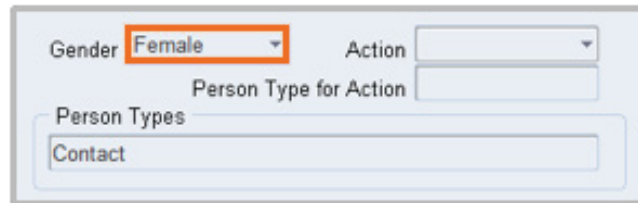


- Select the relevant action to be taken and enter an appropriate display name.
- Execution of the above steps results in selecting a value from corresponding choice-box/dropdown in an Oracle EBS form.

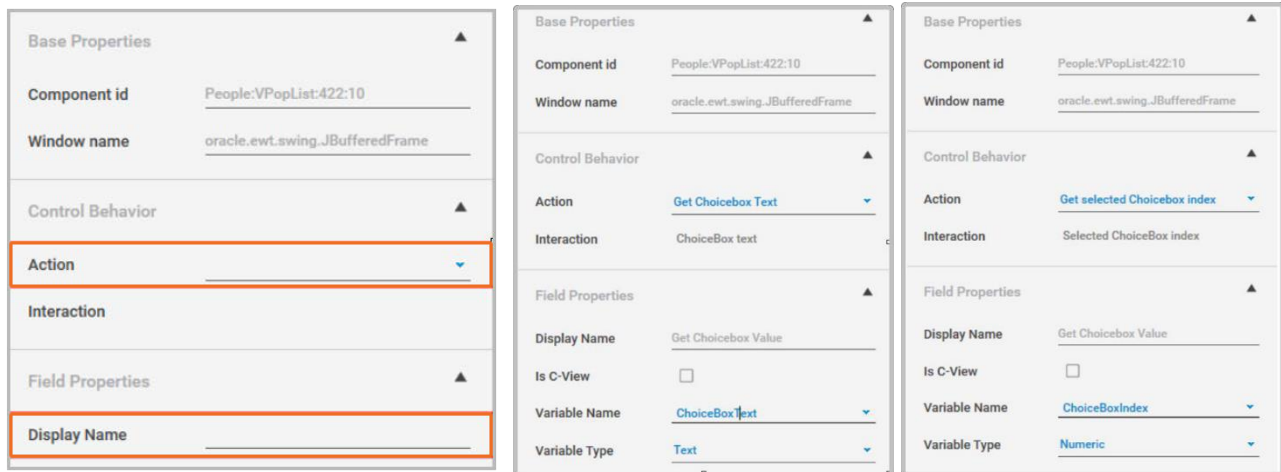
9.24.10 Read Selected Choice-box Value

To automate the reading of the text/index of the selected option in the choice-box in an Oracle EBS form, follow the below mentioned steps:

- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.




- Point the mouse to the component at which the action must be taken, to extract its component ID.
- Select the relevant action to be taken and enter an appropriate display name.

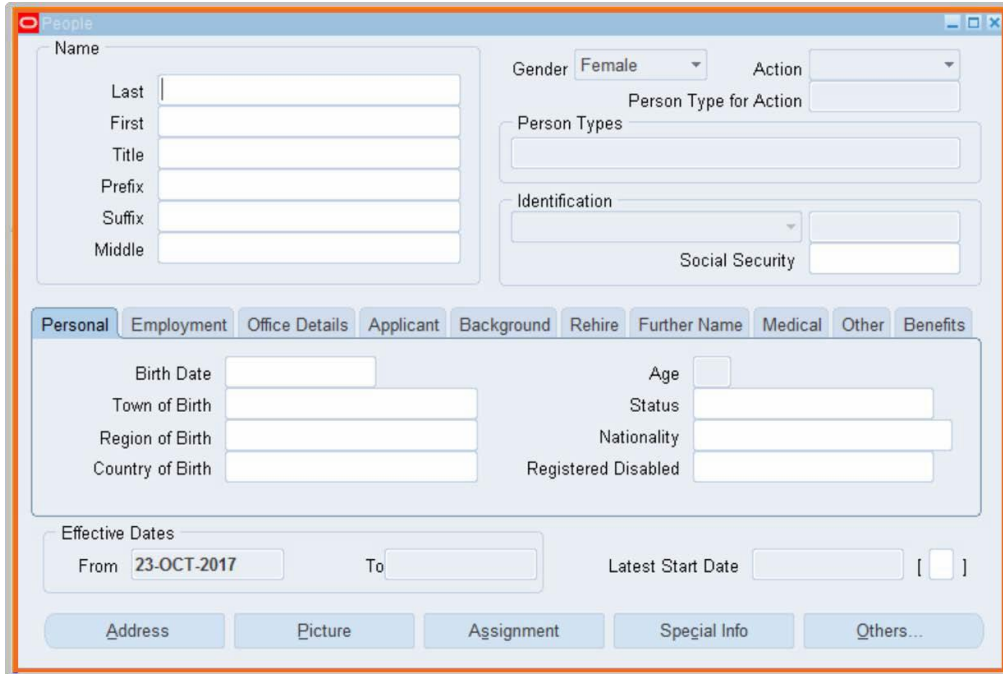


- Execution of the above steps results in reading the text or the index of the corresponding checkbox value in an Oracle EBS form.

9.24.11 Activate an Oracle Form

Automate the “activating of an Oracle EBS form” by following the below mentioned steps.

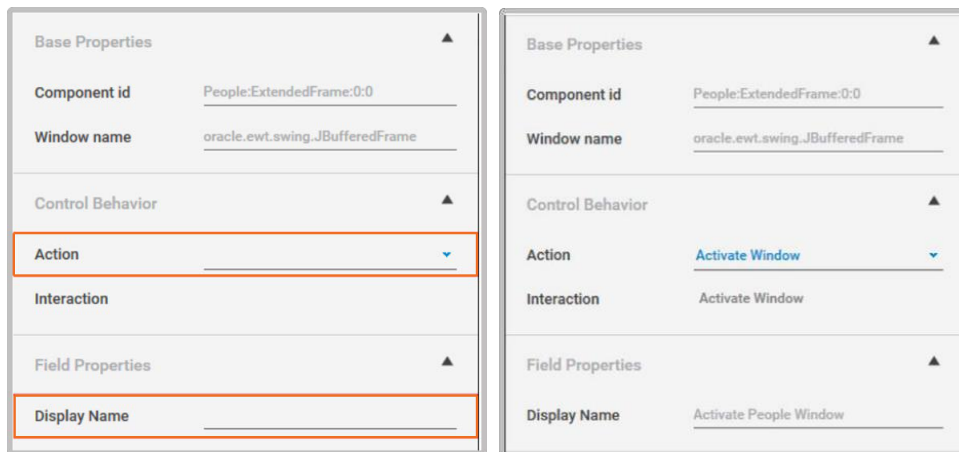
- Add an “**Application Control**” Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the window at which the action must be taken, to extract its component ID.



The screenshot shows the Oracle People form window. The window title is "People". It contains several sections:

- Name:** Last, First, Title, Prefix, Suffix, Middle.
- Gender:** Female (dropdown).
- Action:** (dropdown).
- Person Type for Action:** (text field).
- Person Types:** (text field).
- Identification:** (dropdown).
- Social Security:** (text field).
- Navigation tabs:** Personal, Employment, Office Details, Applicant, Background, Rehire, Further Name, Medical, Other, Benefits.
- Personal Information:** Birth Date, Town of Birth, Region of Birth, Country of Birth, Age, Status, Nationality, Registered Disabled.
- Effective Dates:** From 23-OCT-2017, To (text field), Latest Start Date (text field) with a checkbox.
- Buttons:** Address, Picture, Assignment, Special Info, Others...

- Select the relevant action to be taken and enter an appropriate display name.




The two screenshots show the configuration tool for an activity. The left screenshot shows the configuration tool with the "Action" dropdown menu open and the "Display Name" field highlighted with a red box. The right screenshot shows the configuration tool with the "Action" dropdown menu set to "Activate Window" and the "Display Name" field set to "Activate People Window".

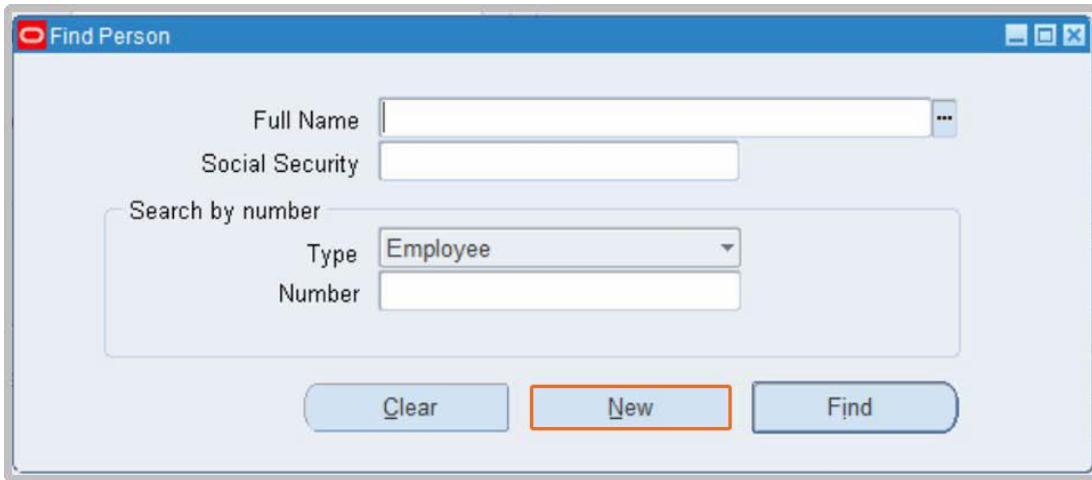
Property	Value
Component id	People:ExtendedFrame:0:0
Window name	oracle.ewt.swing.JBufferedFrame
Action	Activate Window
Interaction	Activate Window
Display Name	Activate People Window

- Execution of the above steps results in activating an Oracle form/window.
- This capability is useful in cases where an action must be taken on a form which is currently not in focus.

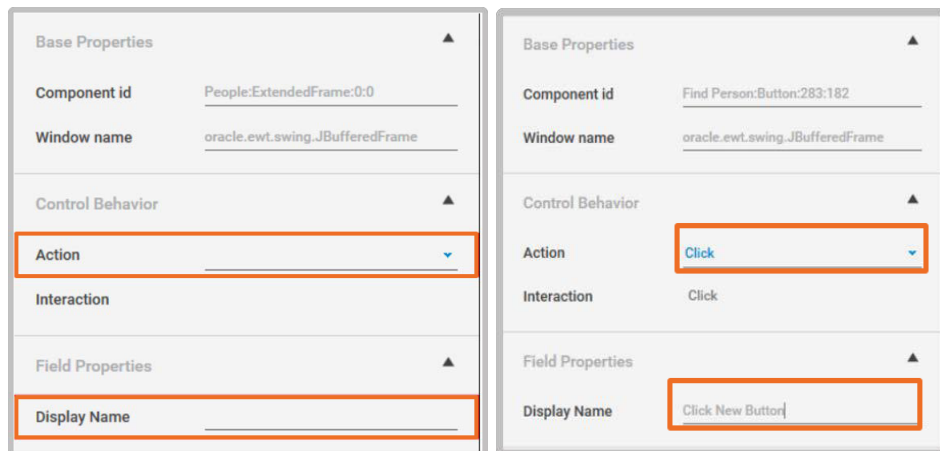
9.24.12 Click

Automate to click a component such as buttons in an Oracle EBS form, by following the below mentioned steps.

- Add an “**Application Control**” Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the component at which the action must be taken, to extract its component ID.




- Select the relevant action to be taken and enter an appropriate display name.

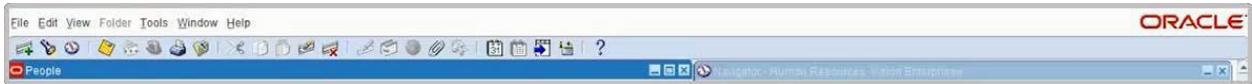


- Execution of the above steps results in clicking of a button in an Oracle EBS form.

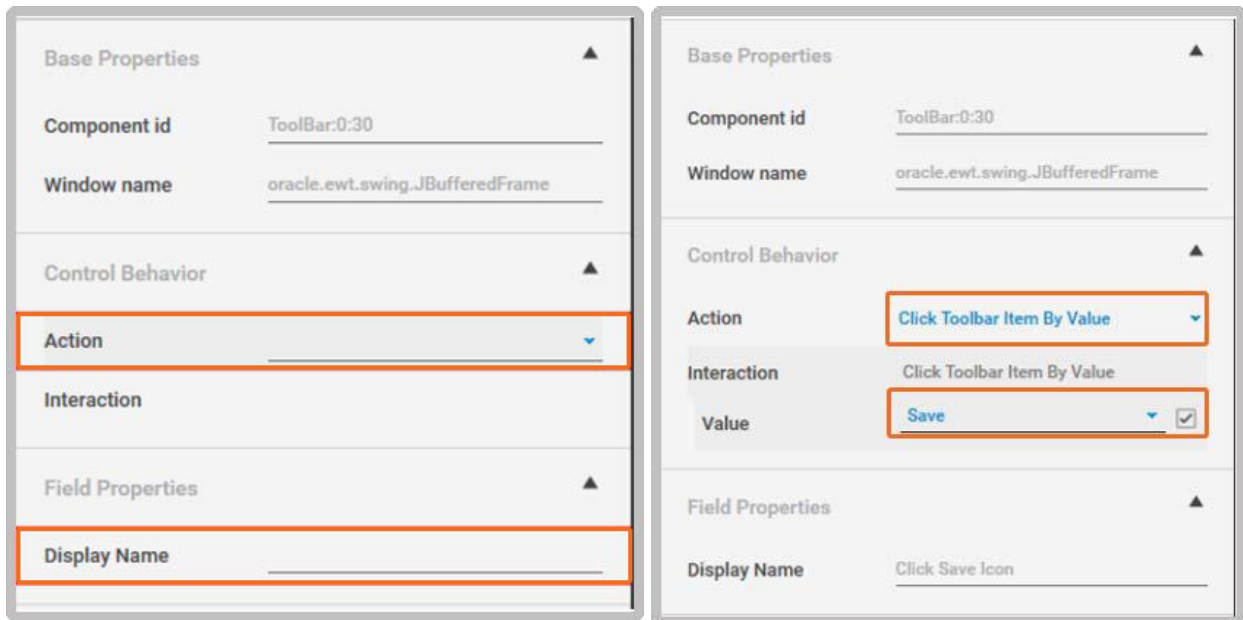
9.24.13 Select Toolbar Item by Value

Automate selection of a toolbar item by value in an Oracle EBS form, by following the below mentioned steps.

- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the component at which the action must be taken, to extract its component ID.




- Select the relevant action to be taken and enter an appropriate display name.
- In the value field, exact text to be displayed on the tooltip must be provided, as input.

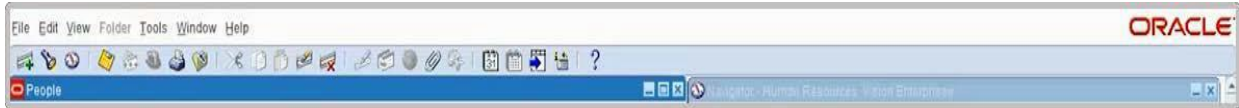


- Execution of the above steps results in clicking the corresponding toolbar item.

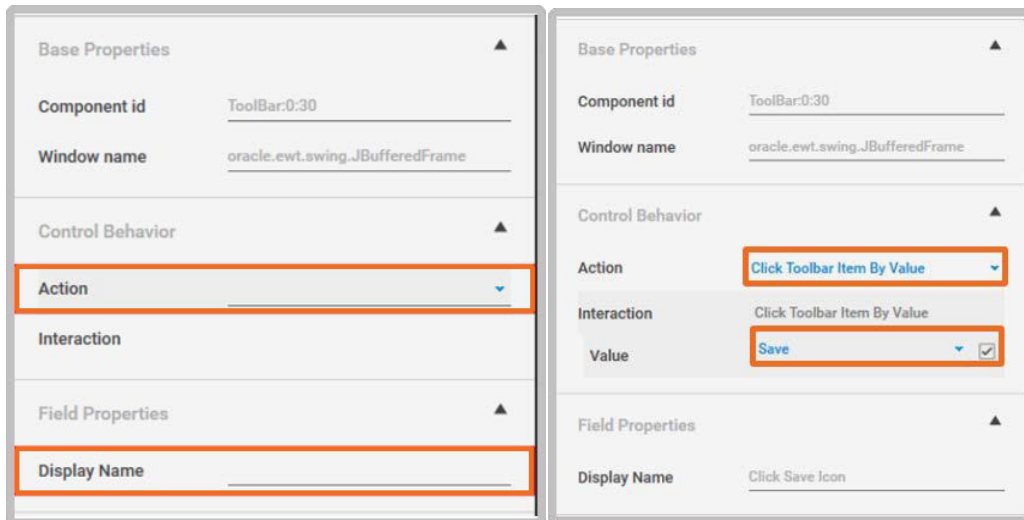
9.24.14 Select Toolbar Item by Index

Automate selecting a toolbar item by index in an Oracle EBS form by following the below mentioned steps.

- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse at the component on which the action must be taken, to extract its component ID.



- Select the relevant action to be taken and enter an appropriate display name.
- In the value field, the index of the relevant item must be provided as input. E.g., to click on the Find icon, provide




the input as

- Execution of the above steps results in clicking the corresponding toolbar item.

9.24.15 Select Tab by Value

Automate switching of tabs in an Oracle EBS form by following the below mentioned steps.

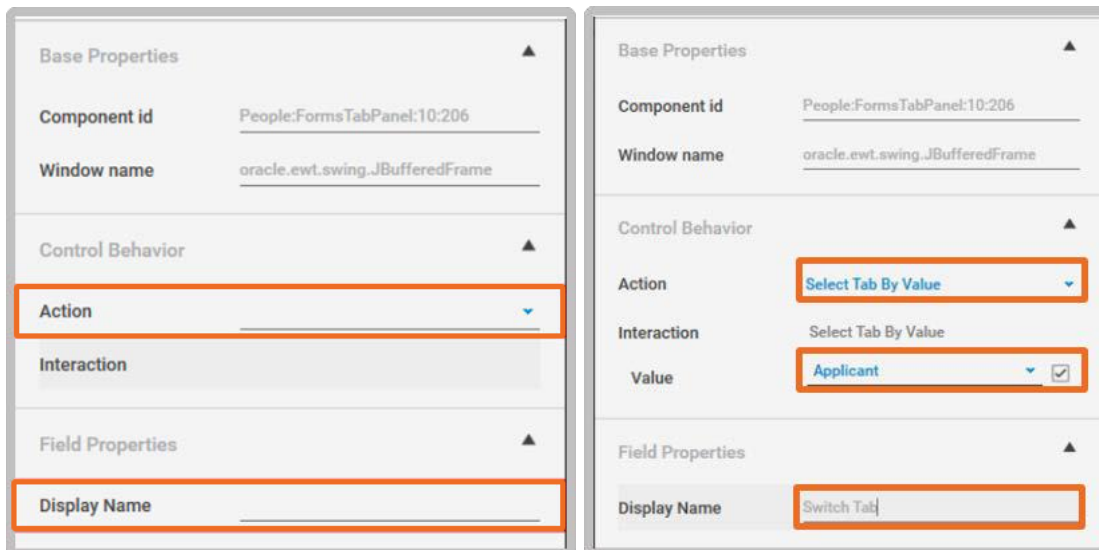
- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the component on which the action must be taken, to extract its component ID.



The screenshot shows a horizontal tabbed interface with the following tabs: Personal, Employment, Office Details, Applicant, Background, Rehire, Further Name, Medical, Other, and Benefits. Below the tabs are several input fields for personal information:

- Birth Date
- Town of Birth
- Region of Birth
- Country of Birth
- Age
- Status
- Nationality
- Registered Disabled

- Select the relevant action to be taken and enter an appropriate display name.
- Provide the tab name as an input to the value field.



The first screenshot shows the configuration tool with the following fields highlighted in orange:

- Action:** (Empty dropdown)
- Display Name:** (Empty text field)


The second screenshot shows the configuration tool with the following fields highlighted in orange:

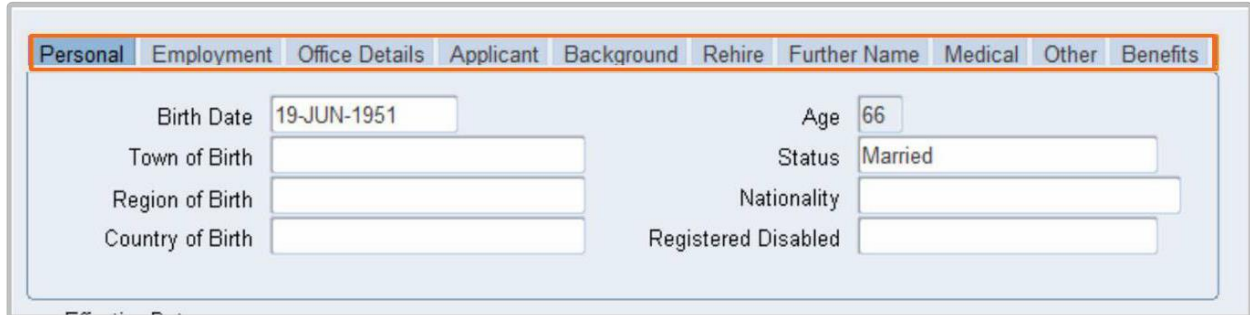
- Action:** Select Tab By Value
- Value:** Applicant
- Display Name:** Switch Tab

- Execution of the above steps results in clicking the corresponding tab.

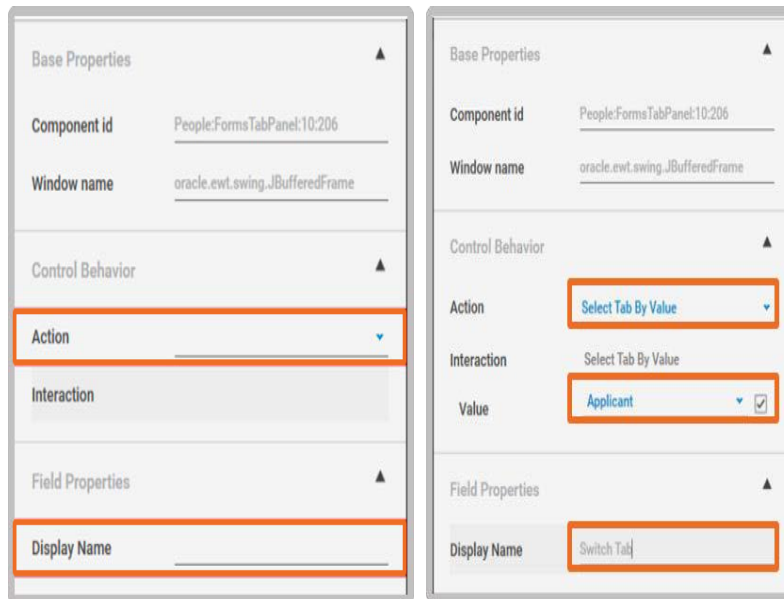
9.24.16 Read Selected Tab Name

Automate reading the name of the selected tab in an Oracle EBS form, by following the below mentioned steps.

- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the component at which the action must be taken, to extract its component ID.




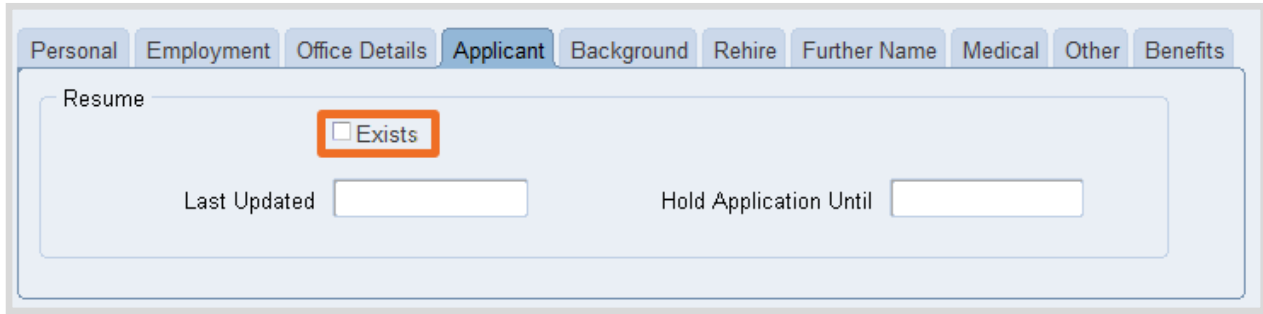
- Select the relevant action to be taken, variable name, variable type and enter an appropriate display name.



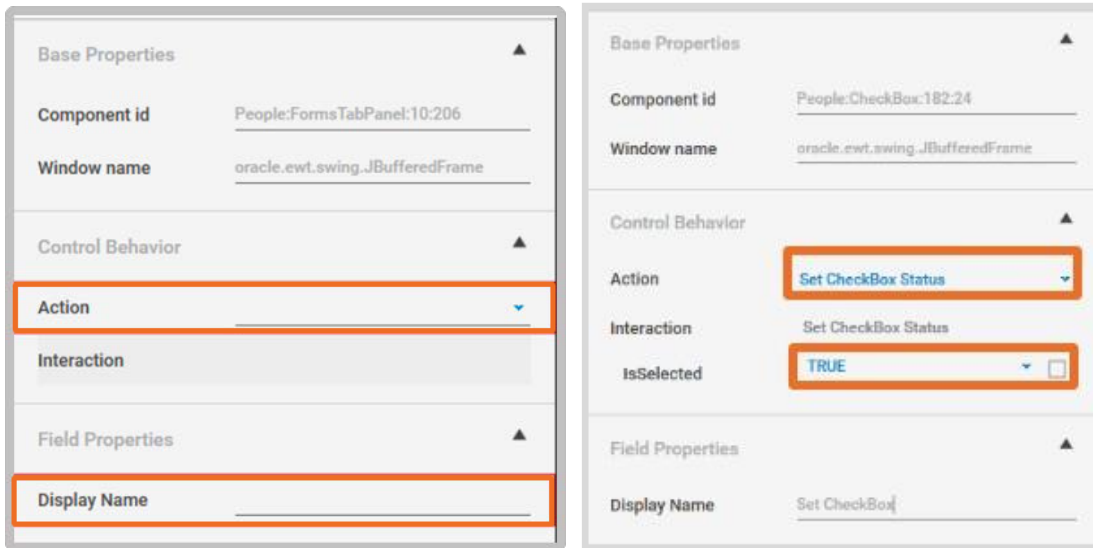
- Execution of the above steps results in reading the name of the corresponding tab.

Automate the setting/checking of a check-box in an Oracle EBS form by following the below mentioned steps.

- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the component at which the action must be taken, to extract its component ID.




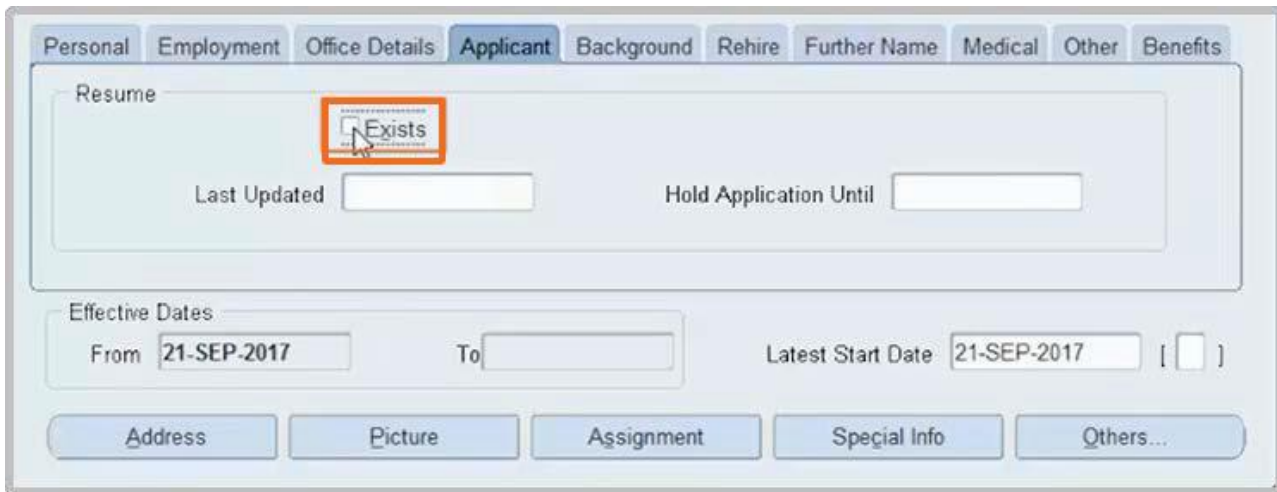
- Select the relevant action to be taken and enter an appropriate display name.



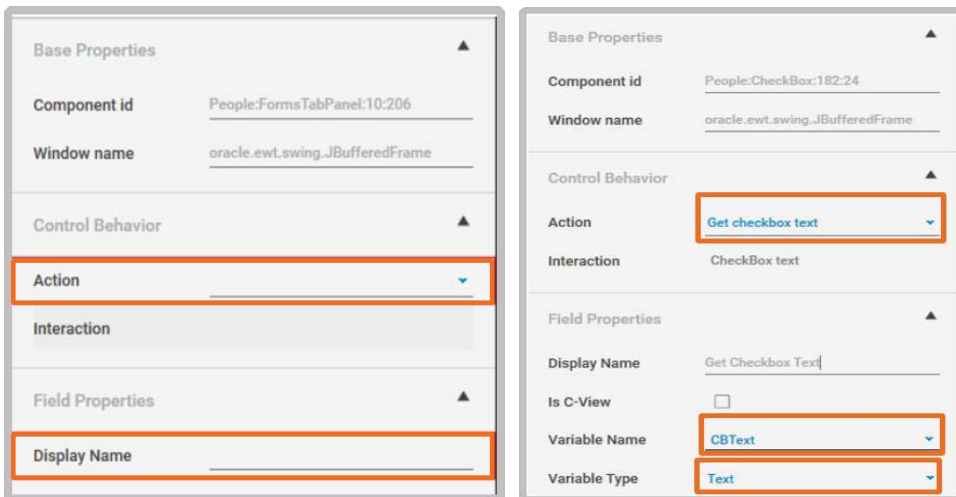
- Provide a relevant value in the "IsSelected" field.
- Execution of the above steps results in setting/checking the corresponding check-box.

Automate reading the status of a check-box in Oracle EBS form by following the below mentioned steps.

- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the component at which the action must be taken, to extract its component ID.




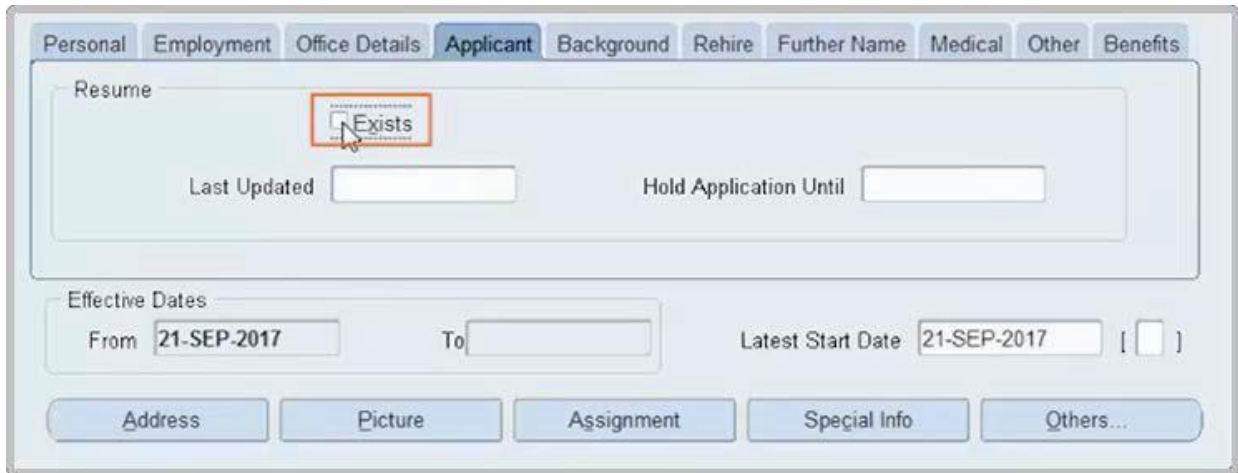
- Select the relevant action to be taken, variable name, variable type and enter an appropriate display name.



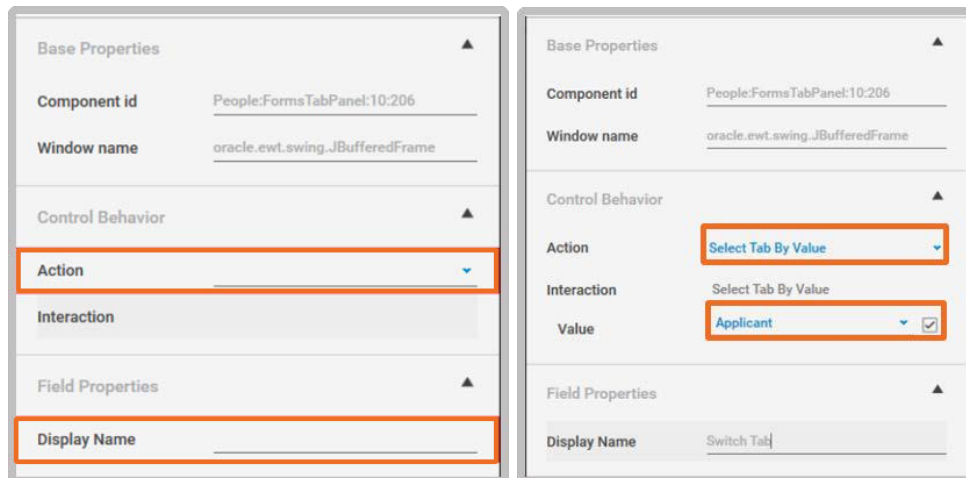
- Execution of the above steps results in getting the status of the corresponding check-box.

Automate reading the text part of the check-box in an Oracle EBS form by following the below mentioned steps.

- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the component at which the action must be taken, to extract its component ID.




- Select the relevant action to be taken, variable name, variable type and enter an appropriate display name.

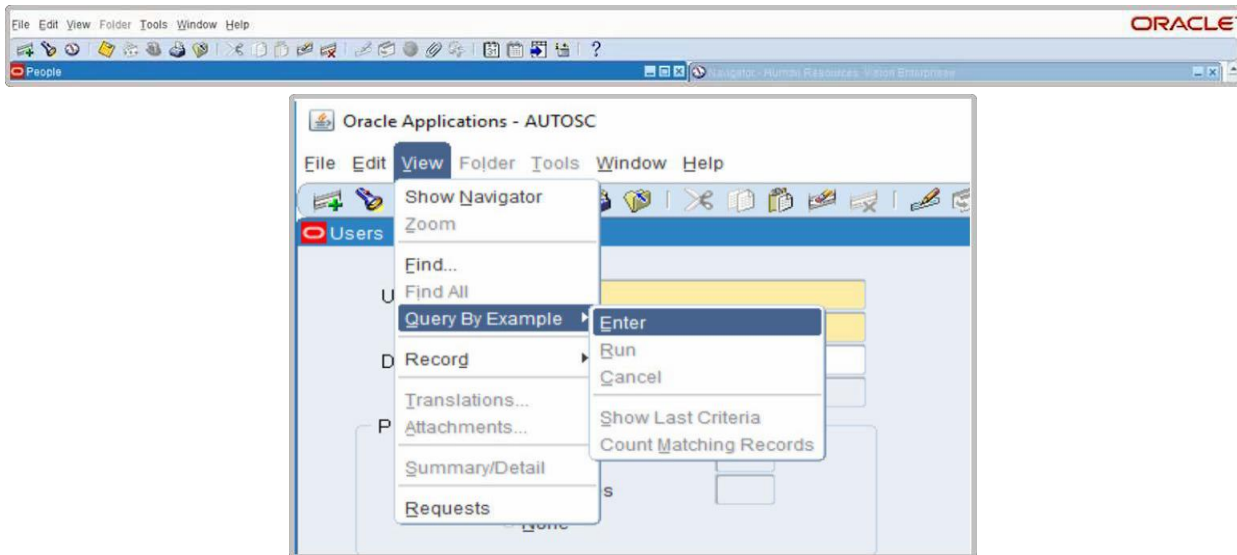


- Execution of the above steps results in getting the text part of the corresponding check-box.

9.24.20 Click Menu Options

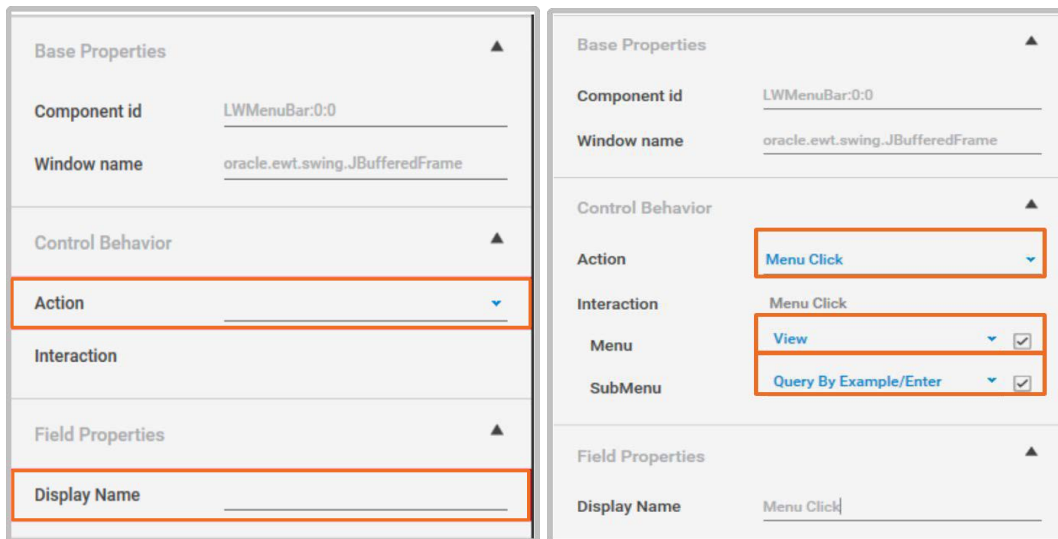
Automate clicking of menu options in an Oracle EBS form by following the below mentioned steps.

- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the component on which the action must be taken, to extract its component ID.




- Select the relevant action to be taken and enter an appropriate display name. Provide Menu and Sub Menu values as input. Menu field has Main Menu value as input. Sub-Menu has values of multiple sub menu levels separated by a forward slash (/).

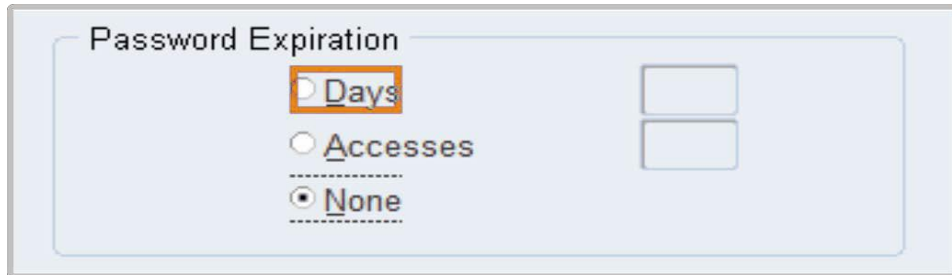
E.g., for View>>Query By Example>>Enter, the input for Menu is View and, Sub-Menu is Query By Example/Enter.



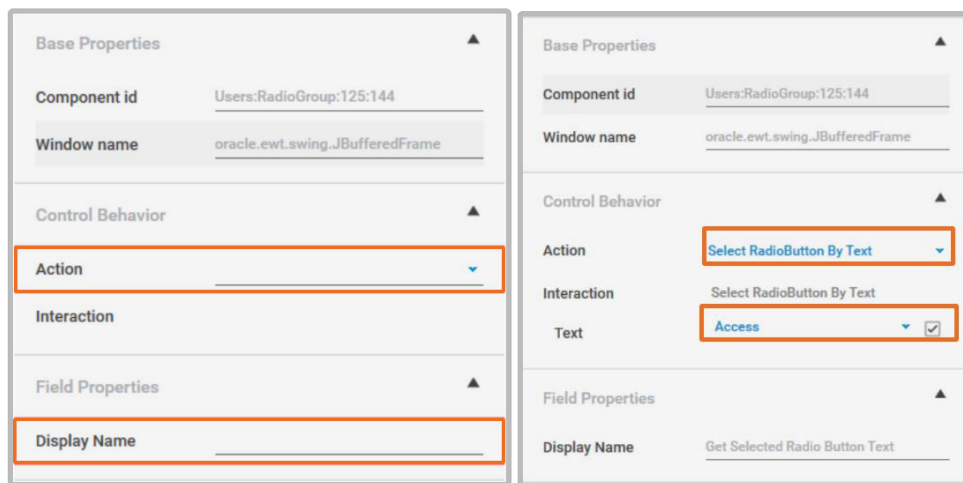
- Execution of the above steps results in clicking the corresponding Menu options.

Automate the setting of a radio button in an Oracle EBS form by following the below mentioned steps:

- Add an **"Application Control"** Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to any one of the radio buttons in the radio button set at which the action must be taken, to extract its component ID.




- Select the relevant action to be taken and enter an appropriate display name.

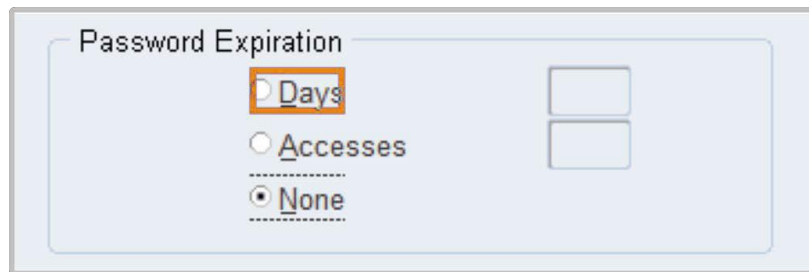


- Execution of the above steps results in the setting of the corresponding radio button in the radio button set.

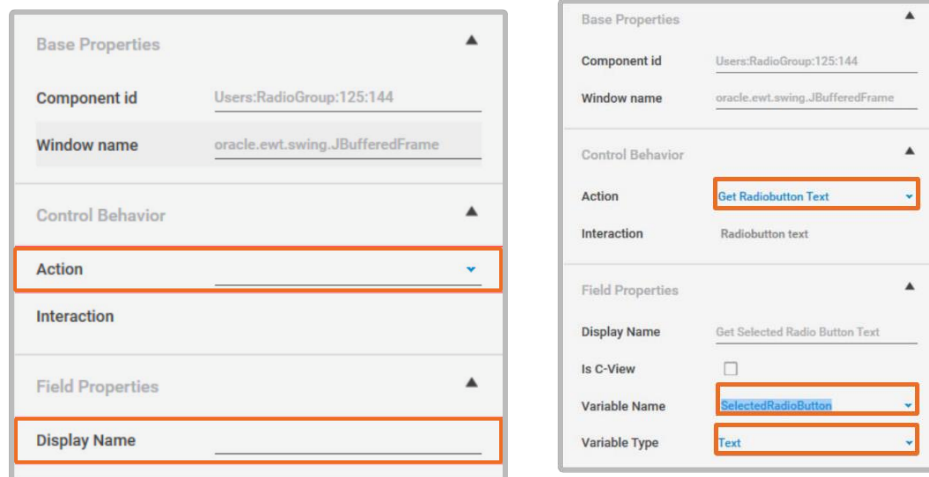
9.24.22 Get Selected Radio Button Text

Automate reading the text of a selected radio button in the radio button set in an Oracle EBS form by following the below mentioned steps.

- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to any one of the radio buttons in the radio button set at which the action must be taken, to extract its component ID.



- Select the relevant action to be taken, variable name, variable type and enter an appropriate display name.




- Execution of the above steps results in getting the text of a selected radio button in the radio button set.

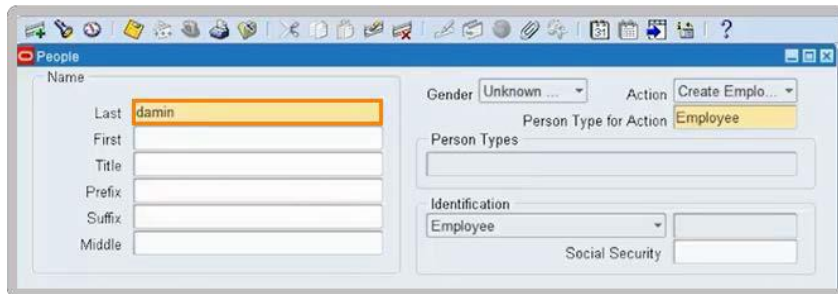
9.24.23 Check Component State

Automate checking the current state of an Oracle EBS form. Before configuring the automation, check for the following conditions:

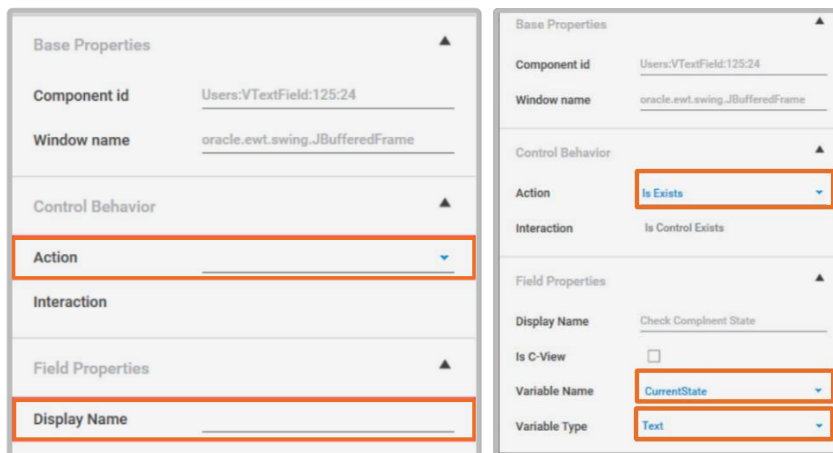
- Is Exists
- Is Enabled
- Is Focused
- Is Visible

Perform the automation by following the below mentioned steps.

- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the window at which the action must be taken, to extract its component ID.




- Select the relevant action to be taken, variable name, variable type and enter an appropriate display name.

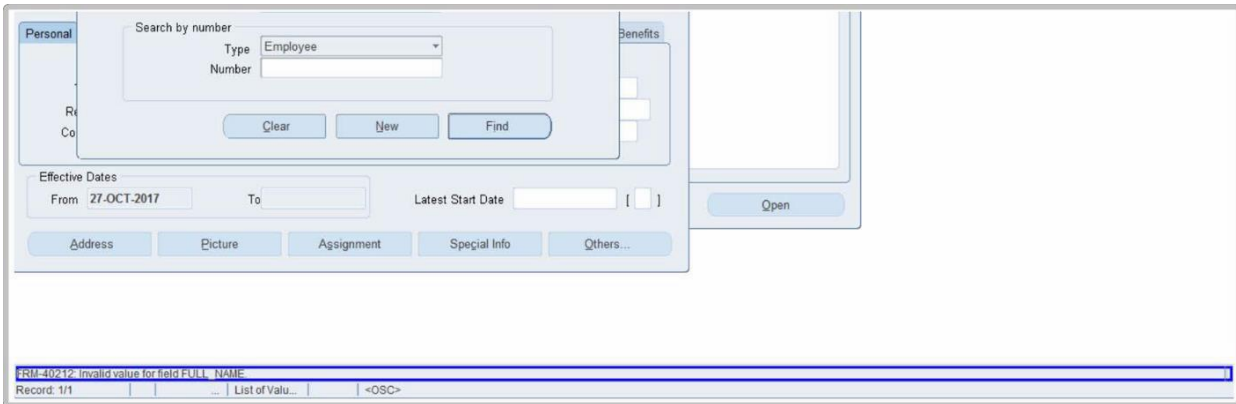


- Execution of the above steps results in getting the current state of the extracted component. E.g., If Is Visible is selected as an action, then the response is True if the component is visible on the form otherwise the response is False.

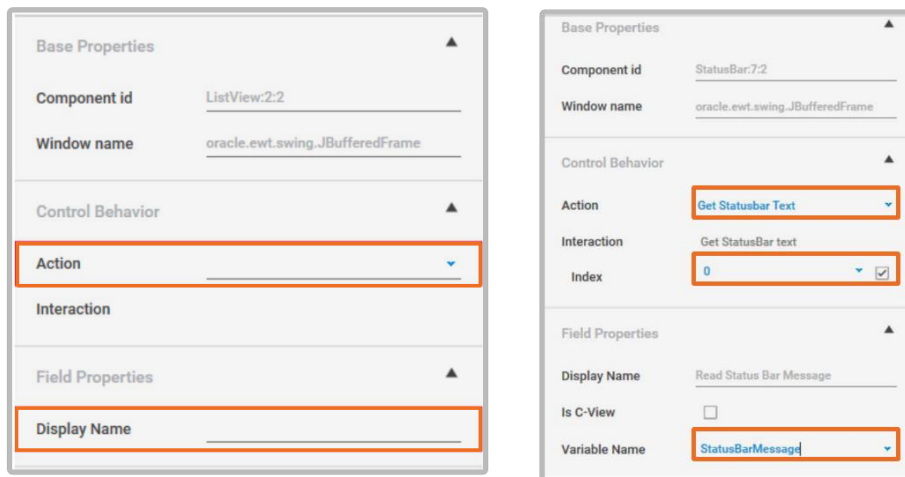
9.24.24 Read Status Bar Message

Automate reading of message on the status bar in an Oracle form by following the below mentioned steps.

- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the component at which the action must be taken, to extract its component ID.




- Select the relevant action to be taken, variable name, variable type and enter an appropriate display name.

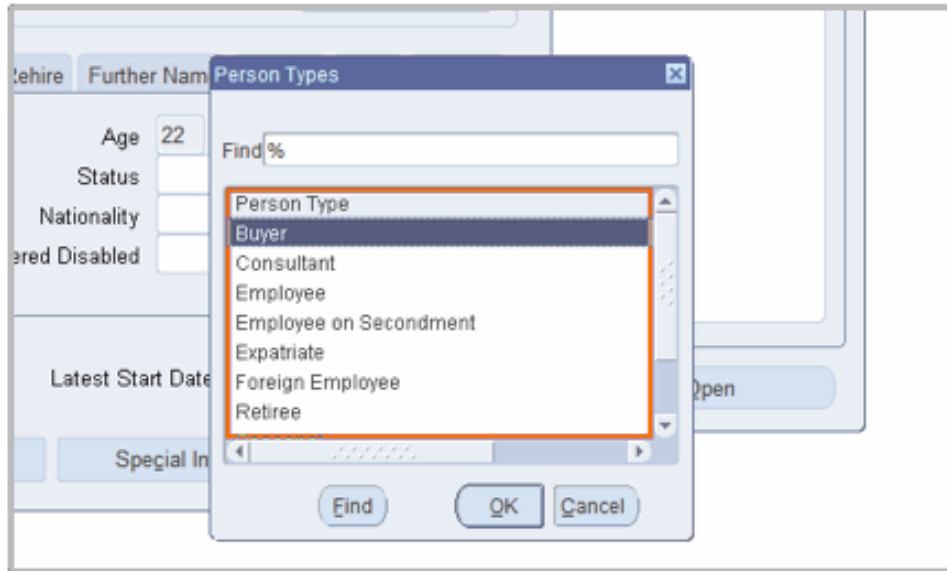


- Select the index as "0" to read the message.
- Execution of the above steps results in reading the status bar message.

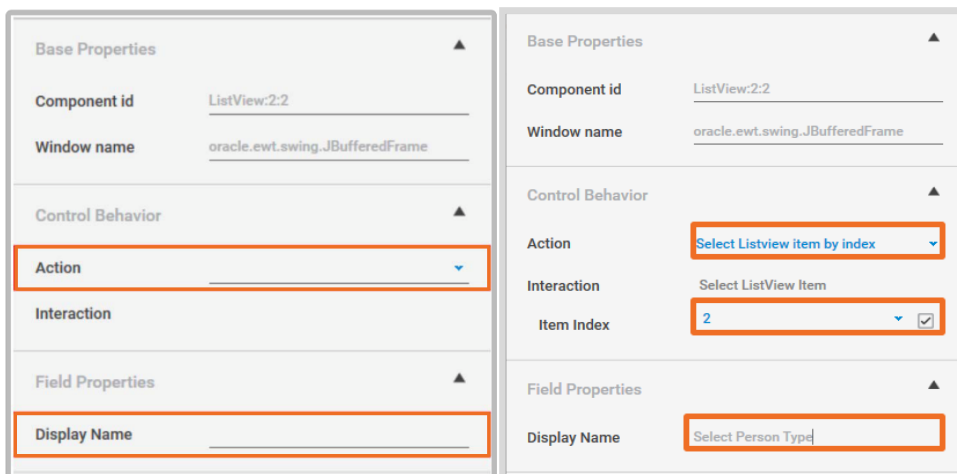
9.24.25 Select List-view Item by Index

Automate the selection of a list-view item by its index in an Oracle form by following the below mentioned steps.

- Add an “Application Control” Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.



- Point the mouse to the component at which the action must be taken, to extract its component ID.
- Select the relevant action to be taken and enter an appropriate display name. Provide an index of the item to be selected as input to the Item index field. E.g., index of item “Employee” is 2




- Execution of the above steps results in selecting the corresponding item in list-view.

9.24.26 Scroll Operations

Automate the following different actions on a scroll in an Oracle EBS form.

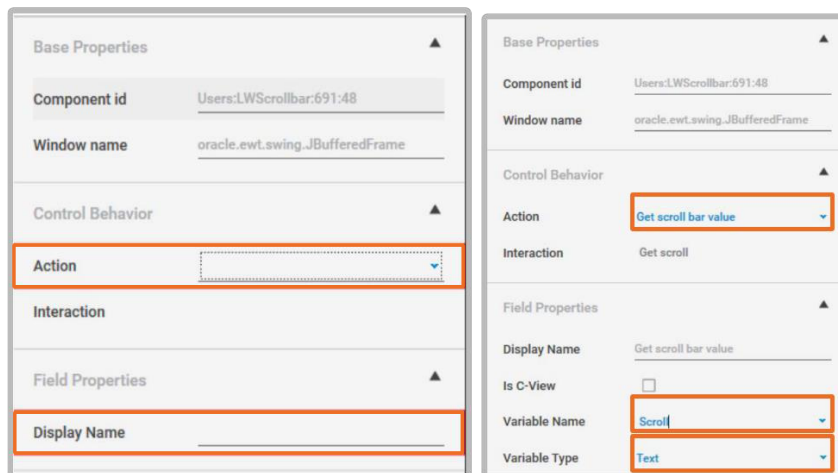
- Get Scroll Bar Value – get the current position of the scroll bar in the table.
- Set Scroll Bar Value – set the position of the scroll bar at the desired position.
- Increment Scroll Bar – increment the scroll bar by a value x, to be provided as an input.
- Decrement Scroll Bar – decrement the scroll bar by a value x, to be provided as an input.

Perform the automation by following the below-mentioned steps.

- Add an “Application Control” Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the component at which the action must be taken, to extract its component ID.



Responsibility	Application	Description	Security Group	Effective Dates	
				From	To
System Administrator	System Administrator	http://www.flipkart.i	Standard	13-JUN-2017	31-JUL-2017
AU General Ledger Super Us	General Ledger		Standard	19-JUN-2017	
AU Candidate Offers	Oracle iProcurement		Standard	19-JUN-2017	
AE HR Web Administrator	Oracle iProcurement		Standard	19-JUN-2017	
ABM Intelligence ADA	Activity Based Manage	test1	Standard	20-JUN-2017	20-JUN-2017



Base Properties

Component id: Users.LWScrollbar:691:48

Window name: oracle.ewt.swing.JBufferedFrame

Control Behavior

Action: Get scroll bar value

Interaction: Get scroll

Field Properties

Display Name: Scroll

Base Properties

Component id: Users.LWScrollbar:691:48

Window name: oracle.ewt.swing.JBufferedFrame

Control Behavior

Action: Get scroll bar value

Interaction: Get scroll

Field Properties

Display Name: Get scroll bar value

Is C-View:

Variable Name: Scroll

Variable Type: Text

- Select the relevant action to be taken and enter an appropriate display name.

Base Properties	
Component id	Users:LWScrollbar:691:48
Window name	oracle.ewt.swing.JBufferedFrame
Control Behavior	
Action	Decrement scroll bar
Interaction	Decrement scroll
Index	1 <input checked="" type="checkbox"/>
Field Properties	
Display Name	Decrement Scroll


Base Properties	
Component id	Users:LWScrollbar:691:48
Window name	oracle.ewt.swing.JBufferedFrame
Control Behavior	
Action	Set scroll bar value
Interaction	Set scroll
Value	1 <input checked="" type="checkbox"/>
Field Properties	
Display Name	Set scroll bar value

Base Properties	
Component id	Users:LWScrollbar:691:48
Window name	oracle.ewt.swing.JBufferedFrame
Control Behavior	
Action	Increment scroll bar
Interaction	Increment scroll
Index	1 <input checked="" type="checkbox"/>
Field Properties	
Display Name	Increment Scroll

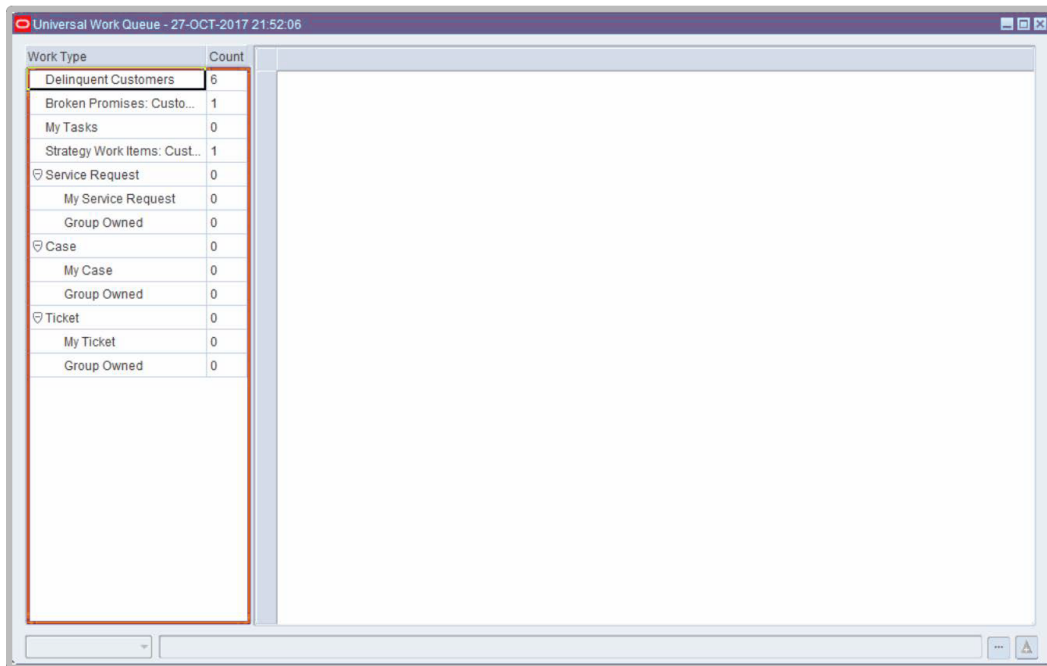
- Execution of the above steps results in the corresponding action on the scroll bar.

9.24.27 Select Grid Row

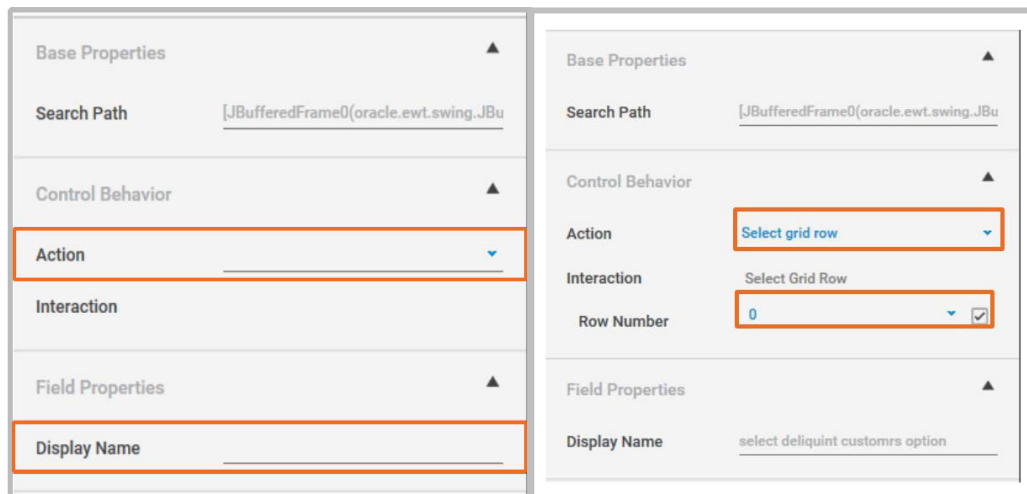
Automate selecting a grid row in an Oracle form by following the below mentioned steps.

- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.

- Point the mouse to the component at which the action must be taken, to extract its component ID.




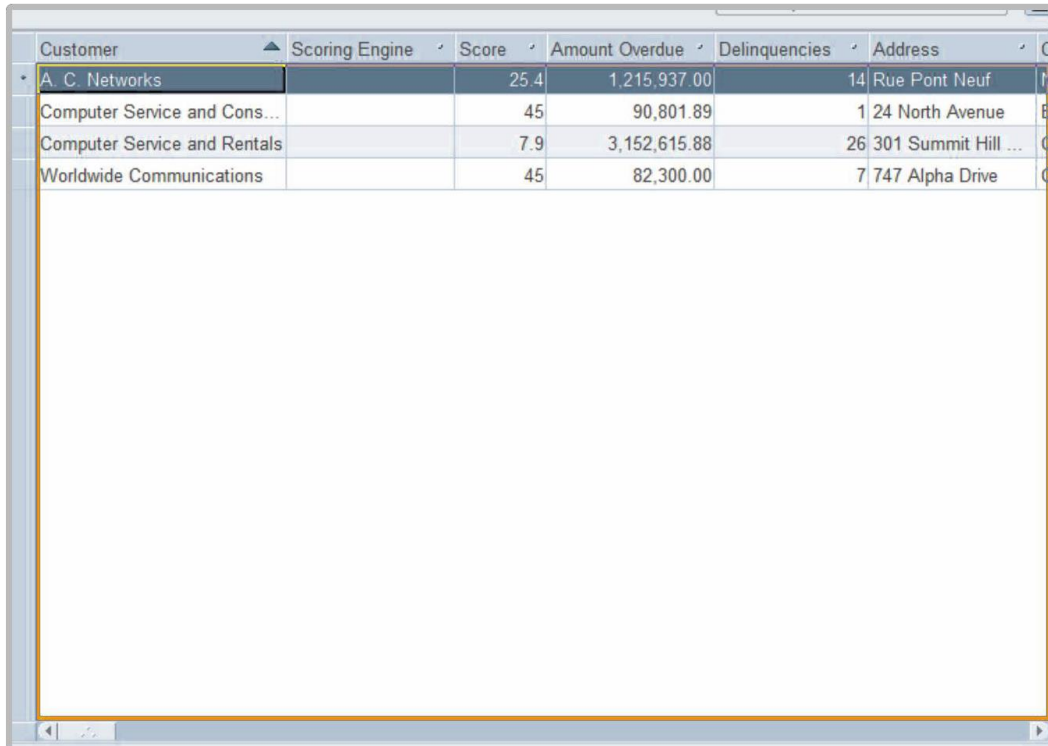
- Select the relevant action to be taken, row number and enter an appropriate display name.



- Execution of the above steps results in clicking the corresponding row in the grid.

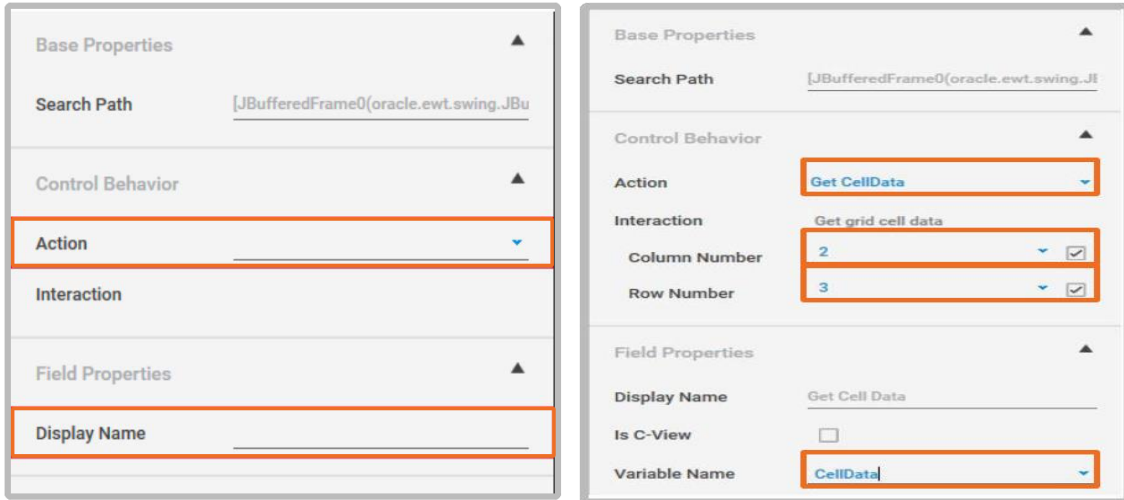
Automate getting or setting the value of a grid cell in an Oracle form by following the below mentioned steps.

- Add an "Application Control" Activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the component at which the action must be taken, to extract its component ID.

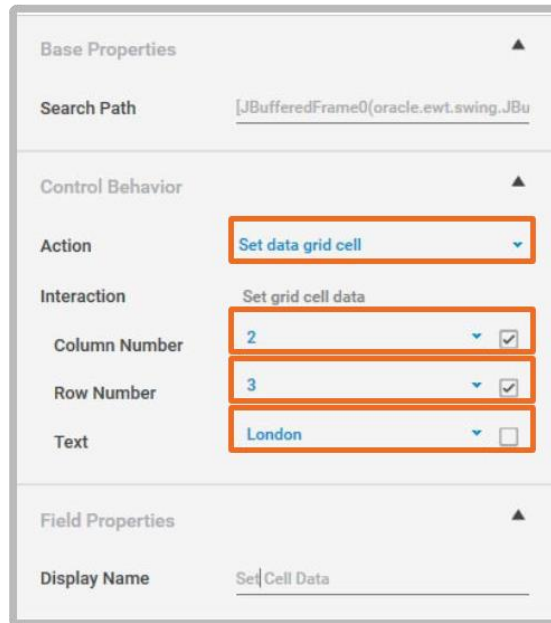


Customer	Scoring Engine	Score	Amount Overdue	Delinquencies	Address
A. C. Networks		25.4	1,215,937.00	14	Rue Pont Neuf
Computer Service and Cons...		45	90,801.89	1	24 North Avenue
Computer Service and Rentals		7.9	3,152,615.88	26	301 Summit Hill ...
Worldwide Communications		45	82,300.00	7	747 Alpha Drive

- Select the relevant action to be taken, row number, column number, input value and enter an appropriate display name.



The image shows two screenshots of a configuration interface. The left screenshot shows the 'Action' dropdown menu set to 'Get CellData' and the 'Display Name' field empty. The right screenshot shows the 'Action' dropdown menu set to 'Get CellData', 'Interaction' set to 'Get grid cell data', 'Column Number' set to 2, 'Row Number' set to 3, 'Display Name' set to 'Get Cell Data', and 'Variable Name' set to 'CellData'.



The image shows a screenshot of a configuration interface. The 'Action' dropdown menu is set to 'Set data grid cell', 'Interaction' is set to 'Set grid cell data', 'Column Number' is set to 2, 'Row Number' is set to 3, 'Text' is set to 'London', and 'Display Name' is set to 'Set Cell Data'.


- Execution of the above steps results in getting or setting the data of the corresponding cell.

9.24.29 Other Grid Operations

Automate the following grid operations in an Oracle form.

- Get Row Count – get the count of a total number of rows in the grid.
- Get Visible Row Count – get the count of visible rows in the grid.
- Get Column Count – get the count of a total number of columns in the grid.
- Get Visible Column Count – get the count of visible columns in the grid.

Perform the automation on these grid operations by following the below mentioned steps.

- Add an “Application Control” activity to the flow.
- Click on Field Extraction  button to extract the ID of the navigator.
- Point the mouse to the component at which the action must be taken, to extract its component ID.

9.25 Appendix

9.25.1 Arguments in Automation Studio

An argument in automation Studio is also used to stores data but the scope of an argument is not limited. It can be used to pass data across different automation process workflow. This implies that an argument created within one workflow is available for selection within another workflow. It can be a string, integer, Boolean, null or other data type depending on the requirement. You can view the list of available variable type in the Arguments pane.

The Argument Fields

The fields of an argument in Studio are listed in the following table:

Field Name	Description
Name	The name of the argument created. If you do not assign a name, argument with a default name is created.
Direction	Defines the direction of the argument. Available options are: <ul style="list-style-type: none"> ▪ In- stored input data ▪ OutArgument- stores output data ▪ InOutArgument- can be used to store input data as well as output data ▪ Property – this option is not used in Automation Studio currently. It displays an error if

Field Name	Description
	used in the process workflow.
Argument Type	<p>Defines the type of argument created. Available options are:</p> <ul style="list-style-type: none"> ▪ String- a sequence of character, either as a constant or a variable. ▪ Boolean-a datatype used for making decision. Can have only two values- true or false. ▪ Int32-an integer type that range from negative through positive. ▪ Object- an abstract datatype that holds data of any kind. ▪ Array of [T]- a group or collection of same datatypes. ▪ Browse for Types...- other types of data. Automation studio supports, .Net variable datatype. <p>Default value is set as String.</p>
Default Value	The value that you want to assign to the argument.

About EdgeVerve

EdgeVerve Systems Limited, a wholly owned subsidiary of Infosys, is a global leader in developing digital platforms, assisting clients to unlock unlimited possibilities in their digital transformation journey. Our purpose is to inspire enterprises with the power of digital platforms, thereby enabling our clients to innovate on business models, drive game-changing efficiency and amplify human potential. Our platforms portfolio across Automation (AssistEdge), Document AI (XtractEdge), and Supply Chain (TradeEdge) helps inspire global enterprises to discover & automate processes, digitize & structure unstructured data and unlock the power of the network by integrating value chain partners. EdgeVerve, with a deep-rooted entrepreneurial culture, our innovations are helping global corporations across financial services, insurance, retail, consumer & packaged goods, life sciences, manufacturing telecom and utilities, and more.

EdgeVerve. Possibilities Unlimited.

www.edgeverve.com