# Plugin Reference Guide1

## (Release 3.0)

# Plugins Compatibility

- Plugins in Release 3.0 are compatible with AutomationEdge Release 6.0.0 onwards.

# Plugins Technical Reference

AutomationEdge Plugins technical reference includes three guides,

- AutomationEdge_R3.0_Plugin_Reference_Guide1
- AutomationEdge_R3.0_Plugin_Reference_Guide2
- AutomationEdge_R3.0_Plugin_Reference_Guide3

# Contents

# I. Workflow Steps

# 1 GUI Automation

GUI Automation plugin steps are used for automation of web and windows applications, on Windows and Linux based OS.

The following browser versions are supported for GUI Automation/GUI Spy in Release 3.0.

| Browser Support | OS | Chrome* | Firefox | IE |
|---|---|---|---|---|
| Release 3.0 | Windows | 75 to 91 | 83 to 89 | 8 to11 |
| | Linux | 80 to 91 | 83 to 89 | - |

*Chrome Drivers are available on EPD. Download the required version and the System Administrator can upload it via File Management menu, on AutomationEdge Server.

**Steps to configure Chrome driver:**

Upload the latest browser drivers for GUI Automation plugin as follows,

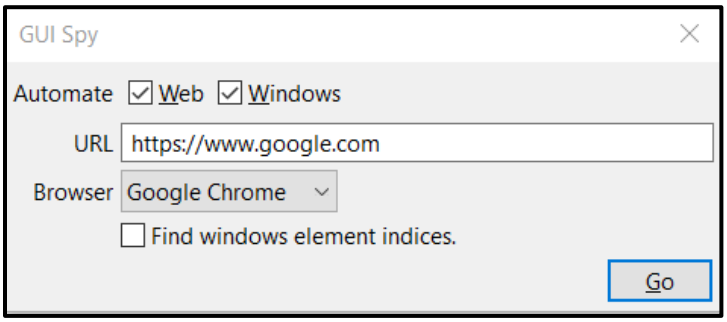
i. Upload the latest browser drivers (e.g. <CHROME<VERSION>.zip>) for GUI Automation Plugin - using Plugin Management on AutomationEdge UI. (Refer section - File Management in AutomationEdge_System_Administrator_Guide_R6.0.0).

ii. Once uploaded the driver is available for both Process Studio and Agent. For Process Studio, additionally, perform Plugin Sync operation so that the drivers are available for automation.

**GUI Spy**

GUI Spy is used to build workflows with GUI Automation plugin steps for web and windows applications.

The GUI Spy pop-up window gives options to operate in Web, Windows or Web + Windows modes.



**References:**

For a description on how to use GUI Spy refer to Appendix 2: GUI Spy Tutorial

For sample workflows refer to,

- Project 8: GUI Spy for Web Applications
- Project 8: GUI Spy Recorder
- Project 8: GUI Spy for Windows Applications

in the AutomationEdge_R7.0.0_ProcesStudio_Activity_Guide.

**Compatibility:**

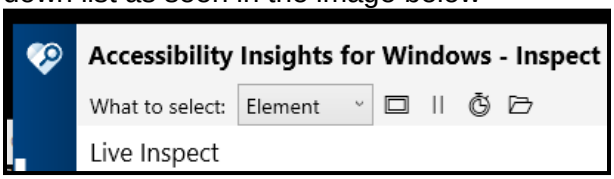
- Windows 7/8.1/10, Windows Server 2008/2012/2016

**Prerequisites:**

- Microsoft .NET Framework 4.5 or above

**Check Plugin Support:**

- You may check if your target application elements are supported by Windows Application GUI Plugin steps.
- Verify the target application elements against “Accessibility Insights for Windows” application.
- You may check only for Element accessibility. Choose criterion Element from the drop down list as seen in the image below




- Get more details in the following video, https://www.youtube.com/watch?v=BIu9ONdMGGg

**Notes:**

- Use GUI Spy to build a workflow with GUI Automation plugin steps for Windows. Refer to Appendix 2: GUI Spy Tutorial (including best practices/workarounds).
- AutomationEdge Workflows with GUI Automation Windows steps should be marked Sequential.

Following are the Plugin steps,

## 1.1 Close Web Window

### 1.1.1 Description

The Close Web Window step closes the current browser window in GUI Automation.

### 1.1.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.2 Element Action

### 1.2.1 Description

Windows: Element Action step performs an action on the specified element (based on the Element Property chosen).

### 1.2.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |
| 2 | ActionType | Possible Action Types are,<br>• Get Property Value |

| | | |
|---|---|---|
| | | • Wait Until (some condition (element property value) is satisfied). |
| 3 | Windows Path | Windows Path is the relative path of the desktop application on which element exists. It is captured by Windows Spy in the Generate Step dialog. It is then populated in the step configuration upon Step Generation. You may also enter or modify Windows path manually on the 'Step Generation' window or step configuration.<br><br>Windows path may be provided as a String or regex.<br>Sample of Windows Path as string:<br>w[Desktop]/w[Application Name]<br>or<br>Sample of Windows Path as regex:<br>w[Desktop]/w[/Application N/], w[Desktop]/w[/Application .*/],<br>If you are providing partial application name then put it as a regex between / and /. Also if the Application name changes dynamically it is a good idea to provide it as a regex. |
| 4 | Criteria | 'Criteria' is the criterion to locate element on Windows Desktop. It is captured by Windows Spy in the Generate Step dialog. Upon Step Generation it is populated in the step configuration. You may change criteria by choosing from the drop down list in 'Step Generation' window or step configuration.<br><br>Note: It is similar as in GUI Automation plugin steps. |
| 5 | Criteria Value | Specify relevant criteria value for the criteria provided above. |
| 6 | Element index(Starts with 0) | For Multiple elements with same criteria, specify index of the element you want to locate.<br>Provide index 0 if you want first element from the located elements, increment index by 1 for the next elements. |
| 7 | Element Property | The possible Element Properties are as follows,<br>• Is Exist – for all elements,<br>• Is Checked – for checkbox,<br>• Is Visible – for all elements,<br>• Is Enable – for all elements,<br>• Is Selected – for all elements but specifically designed for radio button<br><br>Note: |

| | | |
|---|---|---|
| | | • Is Checked Element Property is not supported for Radio button elements.<br>• With element property – Is Exist or Is Visible avoid 'Absolute Path' criteria.<br>• In a workflow it is recommended to first check if element exists (Wait Until – Is Exist) followed by Get Property Values steps. |
| 8 | Reverse above Element Property Condition | Enable checkbox to reverse the Element Property chosen above. |
| | Dynamic fields: | |
| 9 | Output Field | This field is visible if Get Property Value is chosen in Action Type.<br>Specify an output field name to hold the output of step execution (The output is True/False). |
| 10 | Timeout (In Seconds) | This field is visible if Wait Until is chosen in Action Type.<br>Specify a Timeout value for the Element Property chosen. |

| Common Buttons: | | |
|---|---|---|
| No. | Field Name | Description |
| | Buttons: | |
| 1 | OK | On click of this button field values are checked. If any required field values are missing then validation error message is displayed.<br><br>If all the required field values are provided then it will save the field values. |
| 2 | Cancel | Click this button to cancel the window without saving any values. |
| 3 | Button: Edit Element | Click Edit Element to start spying and change step details as desired.<br>Opening spy from step is restricted to the scope of the step. i.e. Only relevant elements are detected with active spy. Use Hotkeys ('Shift' or '~') to detect the element. We may call it as a 'Step Restricted Spy' for understanding purposes. e.g. Suppose spy is started from 'Element Action' step. Now spy is restricted to elements that support 'Element Action'. All elements that support 'Element Action' will be highlighted in blue color and the rest will be highlighted in 'Red' color upon hover. Notice Step dropdown is disabled and value selected is 'Element Action' |

## 1.3 Exit Browser

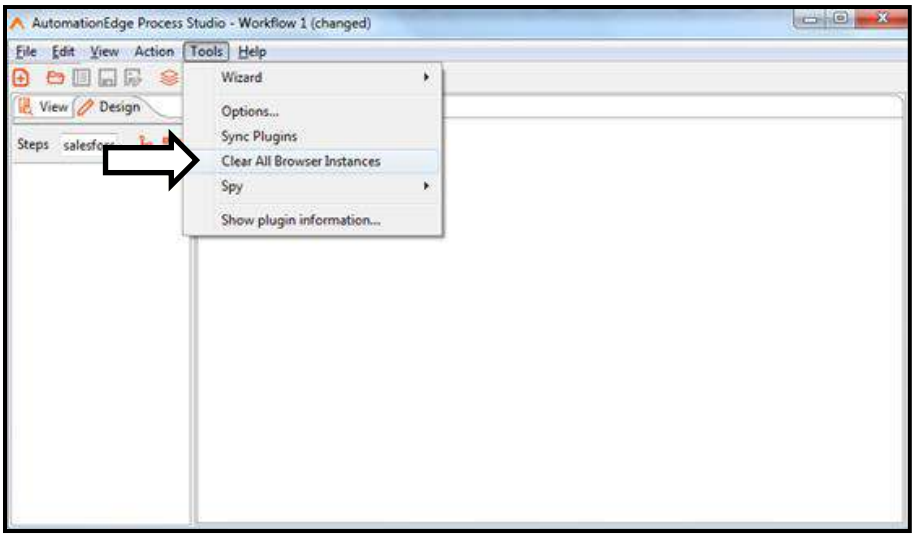
### 1.3.1 Description

The Exit Browser step explicitly closes the entire browser session for the driver specified in the GUI Automation.

Note: When a GUI Automation workflow is run on an agent the entire browser session implicitly closes after the end of the workflow even if Exit browser is not put at the end of the workflow.

However, When a GUI Automation workflow is run from Process Studio the entire browser session does not close implicitly.

Browser terminates abnormally then you need to clear browser driver instance before you can start a new browser. Navigate to Tools→ Clear All Browser Instances as shown below.




### 1.3.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.4 Get Current Window Title

### 1.4.1 Description

The Get Current Window Title GUI Automation plugin step gets the title of the current browser window.

### 1.4.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |
| 2 | Web Browser Name | You can select any one Web Browser Name from the given list (BROWSER1, BROWSER1). You must select the same Web Browser Name which was selected in Start Browser Step. |
| 3 | Window title field | Specify a fieldname to hold the retrieved Window Title. The default value is windowTitle. |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.5 Get Value

### 1.5.1 Description

Windows: Get Value plugin step gets value of element type textbox.

### 1.5.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |

| Read Parameters Tab: | | |
|---|---|---|
| No. | Field Name | Description |
| 1 | Tabular Columns: | |
| i | Windows Path | Windows Path is the relative path of the desktop application on which element exists. It is captured by Windows Spy in the Generate Step dialog. It is then populated in the step configuration upon Step Generation. You may also enter or modify Windows path manually on the 'Step Generation' window or step configuration.<br><br>Windows path may be provided as a String or regex.<br>Sample of Windows Path as string:<br>w[Desktop]/w[Application Name]<br>or<br>Sample of Windows Path as regex:<br>w[Desktop]/w[/Application N/], w[Desktop]/w[/Application .*/],<br>If you are putting partial application name then put ii as a regex between / and /. Also if the Application name changes dynamically it is a good idea to provide it as a regex. |
| ii | Criteria | 'Criteria' is the criterion to locate element on Windows Desktop. It is captured by Windows Spy in the Generate Step dialog. Upon Step Generation it is populated in the step configuration. You may change criteria by choosing from the drop down list in 'Step Generation' window or step configuration.<br><br>Note: It is similar as in GUI Automation plugin steps. |
| iii | Criteria Value | Specify relevant criteria value for the criteria provided above. |

| | | |
|---|---|---|
| iv | Element index(Starts with 0) | For Multiple elements with same criteria, specify index of the element you want to locate.<br>Provide index 0 if you want first element from the located elements, increment index by 1 for the next elements. |
| V | Read Type | Choose a Read Types from the drop down list,<br>• 'Value' –from text box or<br>• 'Text' -from label<br><br>Note: Get value is not applicable to encrypted values like password fields. |
| vi | Output field name | Specify an Output field to hold the value of the textbox element. |
| 2 | Timeout(In seconds) | Specify a Timeout to find the element on the Desktop application. |

| Common Buttons: | | |
|---|---|---|
| No. | Field Name | Description |
| | Buttons: | |
| 1 | OK | On click of this button field values are checked. If any required field values are missing then validation error message is displayed.<br><br>If all the required field values are provided then it will save the field values. |
| 2 | Cancel | Click this button to cancel the window without saving any values. |
| 3 | Edit Element | Click Edit Element to start spying.<br>Opening spy from step is restricted to the step, i.e. Only relevant elements are detected from active spy. Use Hotkeys to detect the element ('Shift' or '~'). We may call it as a 'Step Restricted Spy' for understanding purposes. E.g. Suppose spy is started from 'Get Value' step. Now spy is restricted to elements that support 'Get Value'. All elements that are supported will be highlighted in blue color and others will be highlighted in 'Red' color upon hover. Step dropdown is disabled and value selected is 'Get Value'. |

## 1.6 Inject Java Script

### 1.6.1 Description

Inject java Script is used to inject java script in a GUI automation workflow. Sometimes web control with web locators doesn't react well as is unable to perform the action. At such times this step can be very useful to perform web actions.

Inject Java Script automatically gets the web handle for the Web Browser chosen in Web Browser Name selection which is not available automatically in other scripting steps.

### 1.6.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list,<br>Basically you must select the same Web Browser Name which was selected in Start Browser Step.<br>Inject Java Script automatically gets web handle from the selection. |
| | Left Pane: | |
| 2 | Get Input Fields | By clicking on this it will give a list of all input fields that can be used as a part of Inject Java Script. |
| | Code Snippets: | |
| 3 | Sample Code | Double click Sample Code to populate sample code in the Inject Java Script right hand pane. Here you will find an example to make fields available to next step.<br><br>Partial pseudo code for making multiple fields available to next step,<br>`var FieldName = "FieldName";`<br>`var FieldValue = "FieldValue";`<br><br>`var FieldName2 = "FieldName2";`<br>`var FieldValue2 = "FieldValue2";`<br><br>`return "<<>>"+FieldName+"::"+FieldValue+"<<>>;<<>>"+FieldName2+"::"+FieldValue2+"<<>>";` |
| | Right Pane: | |
| 4 | Inject Java Script | Write the Java Script in the space provided.<br><br>Following, is a sample Java Script showing how to set value and click a button. |

| | | |
|---|---|---|
| | | In this case the code sets values of username and password and clicks a login button.<br>`var chkPrint = document.getElementsByName('uid')[0].value = "mngr169997";`<br>`document.getElementsByName('password')[0].value = "YvynEtA";`<br>`document.getElementsByName('btnLogin')[0].click();` |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.7 Launch Application

### 1.7.1 Description

Windows Application GUI: Launch Application plugin step starts new application at runtime.

### 1.7.2 Configurations

| Input Tab: | | |
|---|---|---|
| No. | Field Name | Description |
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |
| 2 | Path of Application | Specify the full path (including full path and filename with extention) of an application to launch. |
| 3 | Button: Browse | Click the button to browse the Application path. |
| 4 | Checkbox:<br>Fail if error occurs | Enable checkbox to enable the Properties for application validation below. |
| | Properties for validation: | The following fields are enabled only if 'Fail if error occurs' checkbox is enabled.<br>Provide values in the fields below to be validated. |
| 5 | Windows Path/Title | Windows Path and Title are explained below,<br>• **Windows Path** is the relative path of the desktop application on which element exists. Enter Windows path manually in Step Configuration. |

| | | |
|---|---|---|
| | | Windows path may be provided as a String or regex.<br>Sample of Windows Path as string:<br>w[Desktop]/w[Application Name]<br>or<br>Sample of Windows Path as regex:<br>w[Desktop]/w[/Application N/], w[Desktop]/w[/Application .*/],<br>If you are putting partial application name then put ii as a regex between / and /. Also if the Application name changes dynamically it is a good idea to provide it as a regex.<br><br>• **Window Title** is the title of the Windows Application. (i.e. Application Name)<br><br>Note:<br>• When specifying 'Window Path/Title' make sure that no other application, window or folder with the same title is open, else it may lead to confusion at runtime.<br>• If you get an error like "Window is not available or not loaded in specified time", make sure that the Windows Path/Title, Path and application startup time are correctly specified. |
| 6 | Application startup time (Minutes) | Specify a waiting period in minutes for application startup as specified in the Windows Path/Title above.<br><br>Note: You may manually check time taken by the application to load and come up with a better approximation of Application startup time. |
| 7 | Checkbox:<br>close application on failure | You may enable this checkbox to close the application if the validations above are not successful. |

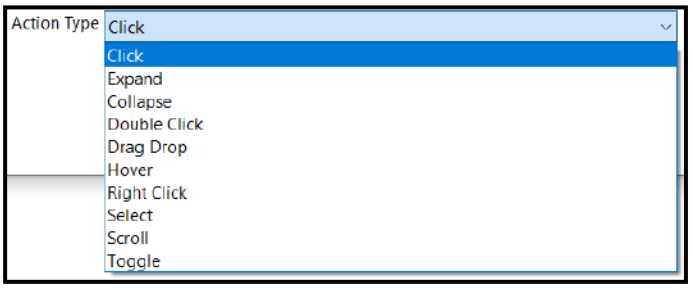
| Common Buttons: | | |
|---|---|---|
| No. | Field Name | Description |
| | Buttons: | |
| 1 | OK | On click of this button field values are checked. If any required field values are missing, then validation error message is displayed.<br><br>If all the required field values are provided then it will save the field values. |

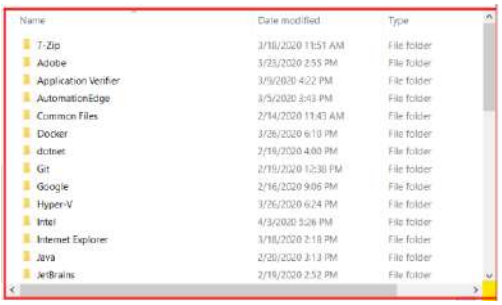
| | | |
|---|---|---|
| 2 | Cancel | Click this button to cancel the window without saving any values. |

## 1.8 Mouse Action

Windows Application GUI: Mouse Action plugin step performs simple mouse actions such as hover, click etc. to complex operations such as drag and drop.

### 1.8.1 Configurations

| Input Tab: | | |
|---|---|---|
| No. | Field Name | Description |
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |
| 2 | Action Type | Select an Action Type from the list below,<br>  <br>• Hover, Click, Double Click, Right Click actions are available with all element types.<br>• Expand, Collapse are useful for tree items, Dropdown element. In case expand is not available you may use Click Action Type.<br>• Toggle is useful for setting value to checkbox.<br>• Scroll option is available for scrollable elements.<br>You need to detect the immediate window containing the scroll bar, possibly by hovering on the intersection of the right and bottom scroll bars as marked in yellow below. |

| | | |
|---|---|---|
| | |  <br>• If Action type is Drag Drop first Spy the Source Element and generate step. In the step click on any field in the Target Element and click Edit Destination Element button to spy the destination element.<br><br>**Note:** In Mouse action In case drag drop is not working you may use the Set Value step. |
| | **Source Element:** | |
| 3 | Windows Path | Windows Path is the relative path of the desktop application on which element exists. It is captured by Windows Spy in the Generate Step dialog. It is then populated in the step configuration upon Step Generation. You may also enter or modify Windows path manually on the 'Step Generation' window or step configuration.<br><br>Windows path may be provided as a String or regex.<br>Sample of Windows Path as string:<br>w[Desktop]/w[Application Name]<br>or<br>Sample of Windows Path as regex:<br>w[Desktop]/w[/Application N/], w[Desktop]/w[/Application .*/],<br>If you are putting partial application name then put ii as a regex between / and /. Also if the Application name changes dynamically it is a good idea to provide it as a regex. |
| 4 | Criteria | 'Criteria' is the criterion to locate element on Windows Desktop. It is captured by Windows Spy in the Generate Step dialog. Upon Step Generation it is populated in the step configuration. You may change criteria by choosing from the drop down list in 'Step Generation' window or step configuration.<br><br>Note: It is similar as in GUI Automation plugin steps. |
| 5 | Criteria Value | Specify relevant criteria value for the criteria provided above. |

| | | |
|---|---|---|
| 6 | Element index(Starts with 0) | For Multiple elements with same criteria, specify index of the element you want to locate.<br>Provide index 0 if you want first element from the located elements, increment index by 1 for the next elements. |
| | | |
| | **Destination Element:** | The section is enabled only for the Action Type Drag Drop. |
| 1 | Windows Path | Windows Path is the relative path of the desktop application on which element exists. It is captured by Windows Spy in the Generate Step dialog. It is then populated in the step configuration upon Step Generation. You may also enter or modify Windows path manually on the 'Step Generation' window or step configuration.<br><br>Windows path may be provided as a String or regex.<br>Sample of Windows Path as string:<br>w[Desktop]/w[Application Name]<br>or<br>Sample of Windows Path as regex:<br>w[Desktop]/w[/Application N/], w[Desktop]/w[/Application .*/],<br>If you are putting partial application name then put ii as a regex between / and /. Also if the Application name changes dynamically it is a good idea to provide it as a regex. |
| 2 | Criteria | 'Criteria' is the criterion to locate element on Windows Desktop. It is captured by Windows Spy in the Generate Step dialog. Upon Step Generation it is populated in the step configuration. You may change criteria by choosing from the drop down list in 'Step Generation' window or step configuration.<br><br>Note: It is similar as in GUI Automation plugin steps. |
| 3 | Criteria Value | Specify relevant criteria value for the criteria provided above. |
| 4 | Element index(Starts with 0) | For Multiple elements with same criteria, specify index of the element you want to locate.<br>Provide index 0 if you want first element from the located elements, increment index by 1 for the next elements. |
| 5 | Timeout(In seconds) | Specify a Timeout to find the element on the Desktop application. |

Common Buttons:

| No. | Field Name | Description |
|---|---|---|
| | Buttons: | |
| 1 | OK | On click of this button field values are checked. If any required field values are missing then validation error message is displayed.<br><br>If all the required field values are provided then it will save the field values. |
| 2 | Cancel | Click this button to cancel the window without saving any values. |
| 3 | Button:<br>Edit Source Element/<br>Edit Destination Element | Click Edit Element to start spying.<br><br>Either Edit Source Element or Edit Destination Element button is visible depending on any field chosen on 'Source element' or Destination element' group.<br><br>Opening spy from step is restricted to the step, i.e. Only relevant elements are detected from active spy. Use Hotkeys ('Shift' or '~') to detect the element. We may call it as a 'Step Restricted Spy' for understanding purposes. E.g. is restricted to elements that support 'Mouse Action'. All elements that are supported will be highlighted in blue color and others will be highlighted in 'Red' color upon hover. Step dropdown is disabled and value selected is 'Mouse Action'. |

## 1.9 Set Value

Windows: Set Value plugin step sets value to element of type – Text box, Text area, document or pane.

### 1.9.1 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |

| Write Parameter Tab | | |
|---|---|---|
| No. | Field Name | Description |
| 1 | Tabular Columns: | |
| i | Windows Path | Windows Path is the relative path of the desktop application on which element exists. It is captured by Windows Spy in the Generate Step dialog. It is then populated in the step configuration upon Step Generation. You may also enter or modify Windows path manually on the 'Step Generation' window or step configuration.<br><br>Windows path may be provided as a String or regex.<br>Sample of Windows Path as string:<br>w[Desktop]/w[Application Name]<br>or<br>Sample of Windows Path as regex:<br>w[Desktop]/w[/Application N/], w[Desktop]/w[/Application .*/],<br>If you are putting partial application name then put ii as a regex between / and /. Also if the Application name changes dynamically it is a good idea to provide it as a regex. |
| ii | Criteria | 'Criteria' is the criterion to locate element on Windows Desktop. It is captured by Windows Spy in the Generate Step dialog. Upon Step Generation it is populated in the step configuration. You may change criteria by choosing from the drop down list in 'Step Generation' window or step configuration.<br><br>Note: It is similar as in GUI Automation plugin steps. |
| iii | Criteria Value | Specify relevant criteria value for the criteria provided above. |
| iv | Element index(Starts with 0) | For Multiple elements with same criteria, specify index of the element you want to locate.<br>Provide index 0 if you want first element from the located elements, increment index by 1 for the next elements. |
| v | Write Type | Specify a Write type – Append Text or Set New Text |
| vi | Input Text | Specify the text to set value of the element- textbox or text area |
| 2 | Timeout(In Seconds) | Specify a Timeout to find the element on the Desktop application. |

| Common Buttons: | | |
|---|---|---|
| No. | Field Name | Description |
| | Buttons: | |
| 1 | OK | On click of this button field values are checked. If any required field values are missing then validation error message is displayed.<br><br>If all the required field values are provided then it will save the field values. |
| 2 | Cancel | Click this button to cancel the window without saving any values. |
| 3 | Edit Element | Click Edit Element to start spying.<br>Opening spy from step is restricted to the step, i.e. Only relevant elements are detected from active spy. Use Hotkeys (‘Shift’ or ’~’) to detect the element. We may call it as a ‘Step Restricted Spy’ for understanding purposes. E.g. Suppose spy is started from ‘Set Value’ step. Now spy is restricted to elements that support ‘Set Value’. All elements that are supported will be highlighted in blue color and others will be highlighted in ‘Red’ color upon hover. Step dropdown is disabled and value selected is ‘et Value’. |

## 1.10 Start Browser

### 1.10.1 Description

The Start Browser step initiates the web browser and navigates to the URL specified in GUI automation.

### 1.10.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list. |
| 2 | Browser | Web browser to initialize. Choose from a drop down list containing,<br>• Mozilla<br>• Chrome |

| | | |
|---|---|---|
| | | • Internet Explorer -with Ignore IE Protected mode checkbox (Refer Note 3 below) |
| 3 | Web URL | URL to navigate after web browser is initialized. |
| 4 | Default download directory | Any browser has a default directory path where all files are downloaded. This feature allows users to overwrite the default download directory. Once a new Default download directory is specified, any file downloaded will be stored in the specified directory.<br><br>This setting is only affective for the current browser instance. This option is not available for Internet Explorer. |
| 5 | Maximize Browser | Checkbox to specify if browser window is to be maximized. By default, the browser is not maximized. |
| 6 | Start Node | Checkbox to specify if this step is a start node (no other input steps are connected to this step) |
| 7 | Clear all browser instances | Clear All Browser Instances option is available in Start Browser in addition to the process studio -> tools menu.<br>When a GUI Automation workflow is run from Process Studio the entire browser session does not close implicitly,<br>• Leaving the browser open even when there is an error during development and browser is manually or abnormally terminated.<br>• In case you manually terminate a browser the browser driver is still initialized and when you run the workflow the start step will give an error as the browser is already initialized.<br>• Similarly in case Browser terminates abnormally then you need to clear browser driver instance before you can start a new browser session.<br><br>Enable Clear All Browser Instances checkbox to Clear All Browser Instances before starting the browser. |
| 8 | Delay (in milliseconds) | Maximum global delay set for the entire session. |
| 9 | Notify on browser version change | Enable this checkbox to notify Workflow Administrator and Tenant Administrator if valid emails and SMTP is configured. |
| 10 | Page load timeout(in milliseconds) | The page load timeout in milliseconds for any web page across the workflow. Else the workflow errors out. The page load timeout can be overwritten by the Web URL Navigation step. |

| | | |
|---|---|---|
| 11 | Delay (in milliseconds) | The total delay in milliseconds for the entire workflow to execute. Delay to locate and perform any action on any web element step, else the workflow errors out |
| 12 | Retry count | User can set retry count number and that will be globally used in all steps of GUI Automation for retrying on error. Whenever the execution of GUI Automation step ends up with an error it will retry execution number of time specified in start browser.<br><br>If one doesn't want to retry for some specific steps, retry count can be set to 0 by using 'Set Retry Count' Step. Use of 'Web Set Retry Count' is to modify the count specified in Start Browser. Steps after Web Set Retry Count will get newly set Retry Count.<br><br>Following steps are excluded from the functionality of Retry Count:<br><br>1. Begin<br>2. End<br>3. CloseWindow<br>4. Robot Handling<br>5. SetDelay<br>6. SetRetryCount<br>7. URL navigation |

Notes:

1. In case of multiple Initialize Web Driver steps in single process/workflow, make sure different drivers are selected.
2. If delay is specified as 0, default considered is 30 seconds.
3. In case of IE, starting version 7, notion of 4 Security zones; namely Intranet, Internet, Trusted Sites and Restricted Sites was introduced. The following two scenarios are to be noted in Web GUI automation using IE.
    1. The default behavior in Web GUI automation using IE is to check "Enable Protected Mode" settings for all the zones and throw an error when this setting is different for any two zones.
    2. In Web GUI automation with IE When "Enable Protected Mode" settings are different for different security zones and you navigate from one URL in one zone to another URL in different zone having a different setting for "Enable Protected Mode", IE internally launches a new process and from that point on, automation fails as it loses the process handle.

In case of Web GUI automation 'Ignore IE Protected Mode Settings' checkbox has been introduced in Web-GUI "Start Browser" step and Web-Spy to handle the two scenarios above.

1. When "Ignore Protected Mode Settings" is CHECKED in Web GUI "Start Browser" step and Web-Spy. "Enable Protected Mode" setting check for different zones is ignored and it starts without error no matter what "Enable Protected Mode" settings for different zones are.
2. We already have the feature to start GUI Automation automation with an initial Browser URL. If during entire automation, browser doesn't navigate to some other URL but URL specified as initial Browser URL, zone switching never happens and automation works fine.

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.11 Web Actions

### 1.11.1 Description

The Web Actions step gives the option to perform several actions on a specific element on the web page in GUI automation. When 'drag and drop' action is selected the configurations for destination element is enabled.

### 1.11.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |
| 2 | Action* | Click action to be performed on the web page element. |
| | Source Element: | |
| 1 | Criteria* | Locate element on web page |
| 2 | Criteria Value | Relevant criteria value has to be specified. |

| | | |
|---|---|---|
| 3 | Multiple elements with same criteria | Check this option if there are multiple web elements in the web page having same criteria value you have specified. By default it will be unchecked. |
| 4 | Element index | If you have checked Multiple **elements with same criteria** specify index of the element you want to locate.<br>Select option **FIRST** if you want first element from the located element and select **LAST** if you want last element. |
| 5 | Use Coordinates | Enable checkbox to use coordinate options. This option is disabled for drag and drop Action. |
| 6 | X-Coordinate Value | Specify the value of X-Coordinate to move the cursor to specified x-coordinate of the located element. |
| 7 | Y-Coordinate Value | Specify the value of Y-Coordinate to move the cursor to specified y-coordinate of the located element. |
| | Destination Element: | When ‘drag and drop’ action is selected the configurations for destination element is enabled. The configurations available are same as 1-4 of source element as above. |

## *Action

The table below contains the description of each Action.

| No. | Field Name | Description |
|---|---|---|
| 1 | Click | Perform single click on the web element. |
| 2 | Right Click | Perform right click on the web element. |
| 3 | Double Click | Perform double click on the web element. |
| 4 | Drag and Drop | Will perform drag and drop operation. |
| 5 | Move to element | Will move focus to the specified web element. |
| 6 | Click and Hold | Clicks (without releasing) in the middle of the element. |

## *Criteria

The table below contains the description of criteria.

| No. | Field Name | Description |
|---|---|---|

| | | |
|---|---|---|
| 1 | Xpath | Xpath defines the hierarchy of the tags through which you can reach to a particular element. For example /html/body/div[1]/div/div[3]/div[2]/div/div[2]/div/div/div[2]/div[2]/div[1]/div[1]/div[1]/span/span |
| 2 | Id | Id attribute of a web element can be best way to identify an element if it is present and is going to constant and unique. There are cases however when it is either absent or its value changes every time you access the application in the browser |
| 3 | Name | Name is another attribute of a web element which can be used to identify an element. |
| 4 | Link Text | If a HTML page contains a hyperlink element <*a* href=www.google.com>Search On Google</a>, you can identify this href element by providing complete text 'Search On Google' as the Link value. |
| 5 | Partial Link text | This criteria is same as Link, just that, in this case, you can give partial string for finding a match |
| 6 | Tag Name | Every HTML element has a tag associated with it. The element can be located using that tag name. For example, textbox has a tag <input>. |
| 7 | Class Name | Class name is the css class associated with an HTML element. If there is a div element MyDiv , you can locate it by giving class name 'someclass'. |
| 8 | Css Selector | This criteria is useful when an HTML element has multiple associated css classes. Consider an HTML element . To identify this div element you can use Css selector as ".btn.primary-btn.submit" |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.12 Web Alert

### 1.12.1 Description

The Web Alert step gives the functionality of handling java script generated alerts in GUI automation.

### 1.12.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |
| 2 | Alert Operations* | Specify the alert operation to perform |
| 3 | Value | By default this field will be disabled and will be enabled if **SetValue** alert operation is chosen. Through this field you can specify the value you want to set in the alert. |
| 4 | Output field name | By default this field will be disabled this will be enabled only when you select **GetValue** alert operation. This allows you to define field name in which alert's text will be stored. |

*Alert Operations

The table below contains the description of each Alert Operation**.**

| No. | Field Name | Description |
|---|---|---|
| 1 | Accept | Select this option when you want to click "OK/Yes" button of the alert. |
| 2 | Dismiss | This option will dismiss the alert which is appeared on the web page. |
| 3 | SetValue | If any alert is asking you to enter some value in it you can select this option and specify the value in **Value** field which is entered in the alert. |
| 4 | GetValue | If you want to retrieve alert text choose this option and it will store the retrieved value in the field you have specified in the **Output field name**. |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.13 Web Click

### 1.13.1 Description

The Web Click step is used to click web element like button, hyper link, image and div. It is basically a single click on the specified web element.

### 1.13.2 Configurations

| No. | Field | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |
| 2 | Criteria* | Locate element on web page |
| 3 | Criteria Value | Relevant criteria value has to be specified. |
| 4 | Multiple elements with same criteria | Check this option if there are multiple web elements in the web page having same criteria value you have specified. By default it will be unchecked. |
| 5 | Element index<br>(Starts with 0) | If you have checked Multiple **elements with same criteria** specify index of the element you want to locate.<br>Select option **FIRST** if you want first element from the located element and select **LAST** if you want last element. |
| 6 | Number of clicks | Put the number of times web click occurs. It should be greater than zero. Default is 1. |
| 7 | Click Options | Choose a click option from the following three radio buttons. By default, Normal click is selected.<br>◉ Normal click<br>○ Press enter<br>○ Click using java script |

*Criteria

The table below contains the description of criteria.

| No. | Field | Description |
|---|---|---|
| 1 | Xpath | Xpath defines the hierarchy of the tags through which you can reach to a particular element. For example /html/body/div[1]/div/div[3]/div[2]/div/div[2]/div/div/div[2]/div[2]/div[1]/div[1]/div[1]/span/span |
| 2 | Id | Id attribute of a web element can be best way to identify an element if it is present and is going to constant and unique. There are cases however when it is either absent or its value changes every time you access the application in the browser |
| 3 | Name | Name is another attribute of a web element which can be used to identify an element. |
| 4 | Link Text | If a HTML page contains a hyperlink element <*a* href=www.google.com>Search On Google</a>, you can identify this href element by providing complete text 'Search On Google' as the Link value. |
| 5 | Partial Link text | This criteria is same as **Link**, just that, in this case, you can give partial string for finding a match. |
| 6 | Tag Name | Every HTML element has a tag associated with it. The element can be located using that tag name. For example, textbox has a tag <input>. |
| 7 | Class Name | Class name is the css class associated with an HTML element. If there is a div element MyDiv , you can locate it by giving class name 'someclass'. |
| 8 | Css Selector | This criteria is useful when an HTML element has multiple associated css classes. Consider an HTML element . To identify this div element you can use Css selector as ".btn.primary-btn.submit" |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.14 Web Composite

### 1.14.1 Description

The Web Composite step is a singular step that can be used to perform several Web GUI actions.

Note: The step will perform only the basic actions. For the advanced requirements the original steps need to be used.

### 1.14.2 Configurations

| No. | Field | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |
| | Actions Tab Table: | |
| 2 | Actions | Choose the Action to be performed from the drop down list.<br><br>The following, actions can be performed using the Web Composite step.<br><br>• Get Value<br>• Set Value<br>• Click<br>• Right Click<br>• Double Click<br>• Click and Hold<br>• Move to Element<br>• Alert SetValue<br>• Alert Getvalue<br>• Alert Accept<br>• Alert Dismiss<br>• Element Displayed<br>• Element Enabled<br>• Element Selected<br>• Page Scroll<br>• Wait Until Alert Present<br>• Wait Until Element Visible<br>• Wait Until Element Present<br><br>The following steps are not supported in<br>• Start Browser<br>• Exit Brower<br>• Close Web Window<br>• Robot Handling<br>• Web Drop Down Selection<br>• Web File Upload |

| | | |
|---|---|---|
| | | • Web Loop Table<br>• Web URL Navigation<br>• Web Set Delay<br>• Web Actions → Drag and Drop |
| 3 | Criteria* | Choose the element locator to locate element on web page. |
| 4 | Criteria Value | Specify the relevant criteria value of the element locator to search with. |
| 5 | Multiple elements with same criteria | If there are multiple elements on the same page/frame with same criteria select Yes. |
| 6 | Element index(Starts with 0) | Element index(Starts with 0) is enabled if Multiple **elements with same criteria** is marked as Yes. Need to specify the index of element to access.Select option **FIRST** if you want first element from the located element and select **LAST** if you want last element. |
| 7 | Value | Provide an optional value depending on the action chosen. (Such as if Web Click is chosen as action put the number of times web click occurs. It should be greater than zero. Default is 1; Provide value for Web Set Value). |
| 8 | Output field name | Specify an optional new output fieldname to hold value given out by the action depending on the action chosen, example Web Get Value. |
| 9 | Delay(in milliseconds) | Specify delay for the particular action. If empty, it is defaulted to the value set by Start Browser or latest Web Set Delay step. This is an implicit delay. |
| 10 | Description | Provide an optional description depending on the action chosen. |

## *Criteria

The table below contains the description of criteria.

| No. | Field | Description |
|---|---|---|
| 1 | Xpath | Xpath defines the hierarchy of the tags through which you can reach to a particular element. For example /html/body/div[1]/div/div[3]/div[2]/div/div[2]/div/div/div[2]/div[2]/div[1]/div[1]/div[1]/span/span |
| 2 | Id | Id attribute of a web element can be best way to identify an element if it is present and is going to constant and unique. There are cases however when it is either absent or its value changes every time you access the application in the browser |
| 3 | Name | Name is another attribute of a web element which can be used to identify an element. |

| | | |
|---|---|---|
| 4 | Link Text | If a HTML page contains a hyperlink element <*a* href=www.google.com>Search On Google</a>, you can identify this href element by providing complete text 'Search On Google' as the Link value. |
| 5 | Partial Link text | This criteria is same as **Link**, just that, in this case, you can give partial string for finding a match. |
| 6 | Tag Name | Every HTML element has a tag associated with it. The element can be located using that tag name. For example, textbox has a tag <input>. |
| 7 | Class Name | Class name is the css class associated with an HTML element. If there is a div element MyDiv , you can locate it by giving class name 'someclass'. |
| 8 | Css Selector | This criteria is useful when an HTML element has multiple associated css classes. Consider an HTML element . To identify this div element you can use Css selector as ".btn.primary-btn.submit" |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.15 Web Download Image

### 1.15.1 Description

The Web Download Image step downloads a Web GUI image.

### 1.15.2 Configurations

| No. | Field | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |
| 2 | Criteria | Locate element on web page |
| 3 | Criteria value | Relevant criteria value has to be specified. |

| | | |
|---|---|---|
| 4 | Multiple elements with same criteria | Check this option if there are multiple web elements in the web having same criteria value you have specified. By default it will be unchecked. |
| 5 | Element index(Starts with 0) | If you have checked Multiple elements with same criteria specify index of the element you want to locate.<br>Select option FIRST if you want first element from the located element and select LAST if you want last element. |
| 6 | Download Directory | The directory where the image file will be downloaded. The default directory is ${Internal.Entry.Current.Directory} i.e. that is the same location where the workflow is saved. |
| 8 | Output file path | Specify a field to hold the Output file path including the file path along with name of the image file downloaded. |
| 9 | Add filenames to result | If checkbox is enabled, then, if any image or attachment is downloaded, it is added to result; and reflected in the Requests results as a link to download the image/attachment. |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.16 Web Dropdown Selection

### 1.16.1 Description

The Web Dropdown Selection step is used to select value of dropdown list on the web page for Web GUI automation.

### 1.16.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |

| | | |
|---|---|---|
| 2 | Criteria* | Locate element on web page |
| 3 | Criteria Value | Relevant criteria value has to be specified. |
| 4 | Multiple elements with same criteria | Check this option if there are multiple web elements in the web page having same criteria value you have specified. By default it will be unchecked. |
| 5 | Element index | If you have checked Multiple elements with same criteria specify index of the element you want to locate.<br>Select option FIRST if you want first element from the located element and select LAST if you want last element. |
| 6 | Select by* | Method to select the dropdown value |
| 7 | Selection value | Relevant value need to be specified for the method selected in Select by |

## *Criteria

The table below contains the description of criteria**.**

| No. | Field Name | Description |
|---|---|---|
| 1 | Xpath | Xpath defines the hierarchy of the tags through which you can reach to a particular element. For example /html/body/div[1]/div/div[3]/div[2]/div/div[2]/div/div/div[2]/div[2]/div[1]/div[1]/div[1]/span/span |
| 2 | Id | Id attribute of a web element can be best way to identify an element if it is present and is going to constant and unique. There are cases however when it is either absent or its value changes every time you access the application in the browser |
| 3 | Name | Name is another attribute of a web element which can be used to identify an element. |
| 4 | Link Text | If a HTML page contains a hyperlink element <a href=www.google.com>Search On Google</a>, you can identify this href element by providing complete text ‘Search On Google’ as the Link value. |
| 5 | Partial Link text | This criteria is same as **Link**, just that, in this case, you can give partial string for finding a match |
| 6 | Tag Name | Every HTML element has a tag associated with it. The element can be located using that tag name. For example, textbox has a tag <input>. |

| | | |
|---|---|---|
| 7 | Class Name | Class name is the css class associated with an HTML element. If there is a div element MyDiv , you can locate it by giving class name 'someclass'. |
| 8 | Css Selector | This criteria is useful when an HTML element has multiple associated css classes. Consider an HTML element . To identify this div element you can use Css selector as ".btn.primary-btn.submit" |

*Select by

The table below contains the description of each Select by methods**.**

| No. | Field Name | Description |
|---|---|---|
| 1 | Index | Suppose located dropdown has 3 options "**A,B,C**", now if you want to select "**A**" give the index value as "0" and so on. |
| 2 | Value | Every item in html dropdown has a value attribute associated with it. If the Value attribute of desired option is known select this option. |
| 3 | Visible Text | Visible Text is the value of the option which user can see on the web page. |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.17 Web Element Condition

### 1.17.1 Description

The Web Element Condition step checks specified condition of web page element and returns a Boolean value in Web GUI automation.

### 1.17.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |
| 2 | Condition* | Condition to check on web page element. |
| 3 | Criteria* | Locate element on web page |
| 4 | Criteria Value | Relevant criteria value has to be specified. |
| 5 | Output field name | Name of the output field, which contains the evaluated Boolean value. |

#### *Condition

The table below contains the description of each Condition.

| No. | Field Name | Description |
|---|---|---|
| 1 | Is displayed | Check if web page element is displayed. |
| 2 | Is enabled | Check if web page element is enabled. |
| 3 | Is selected | Check if web page element is selected. |

#### *Criteria

The table below contains the description of criteria.

| No. | Field Name | Description |
|---|---|---|
| 1 | Xpath | Xpath defines the hierarchy of the tags through which you can reach to a particular element. For example /html/body/div[1]/div/div[3]/div[2]/div/div[2]/div/div/div[2]/div[2]/div[1]/div[1]/div[1]/span/span |
| 2 | Id | Id attribute of a web element can be best way to identify an element if it is present and is going to constant and unique. There are cases however when it is either absent or its value changes every time you access the application in the browser |
| 3 | Name | Name is another attribute of a web element which can be used to identify an element. |

| | | |
|---|---|---|
| 4 | Link Text | If a HTML page contains a hyperlink element <a href=www.google.com>Search On Google</a>, you can identify this href element by providing complete text 'Search On Google' as the Link value. |
| 5 | Partial Link text | This criteria is same as Link, just that, in this case, you can give partial string for finding a match |
| 6 | Tag Name | Every HTML element has a tag associated with it. The element can be located using that tag name. For example, textbox has a tag <input>. |
| 7 | Class Name | Class name is the css class associated with an HTML element. If there is a div element MyDiv , you can locate it by giving class name 'someclass'. |
| 8 | Css Selector | This criteria is useful when an HTML element has multiple associated css classes. Consider an HTML element . To identify this div element you can use Css selector as ".btn.primary-btn.submit" |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.18 Web File Upload

### 1.18.1 Description

The Web File Upload step is used for uploading a file with absolute path (browse dialog), on a web page in Web GUI automation.

### 1.18.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |
| 2 | File path | Absolute file path for upload |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.19 Web Get Value

### 1.19.1 Description

The Web Get Value step allows you to retrieve text of any web element (including dropdown). The retrieved value will be stored in a field value. The step also allows multiple criteria to retrieve text of multiple web elements in Web GUI automation.

### 1.19.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |
| 2 | Criteria* | Locate element on web page (including radio button and checkbox) |
| 3 | Criteria value | Relevant criteria value has to be specified. |
| 4 | Element is dropdown | Select **Yes/No** based on element is dropdown or not. |
| 5 | Multiple elements with same criteria | Select **Yes** if there are multiple web elements in the web page having same criteria value you have specified, else select **No** |
| 6 | Element index | If you have selected **Yes** for Multiple elements with same criteria, specify index of the element you want to locate. Select option FIRST if you want first element from the located element and select LAST if you want last element. |
| 7 | Output field name | Here you can define process studio output field in which retrieved value will be stored. |

*Criteria

The table below contains the description of criteria.

| No. | Field Name | Description |
|---|---|---|
| 1 | Xpath | Xpath defines the hierarchy of the tags through which you can reach to a particular element. For example /html/body/div[1]/div/div[3]/div[2]/div/div[2]/div/div/div[2]/div[2]/div[1]/div[1]/div[1]/span/span |
| 2 | Id | Id attribute of a web element can be best way to identify an element if it is present and is going to constant and unique. There are cases however when it is either absent or its value changes every time you access the application in the browser |
| 3 | Name | Name is another attribute of a web element which can be used to identify an element. |
| 4 | Link Text | If a HTML page contains a hyperlink element <a href=www.google.com>Search On Google</a>, you can identify this href element by providing complete text ‘Search On Google’ as the Link value. |
| 5 | Partial Link text | This criteria is same as Link, just that, in this case, you can give partial string for finding a match |
| 6 | Tag Name | Every HTML element has a tag associated with it. The element can be located using that tag name. For example, textbox has a tag <input>. |
| 7 | Class Name | Class name is the css class associated with an HTML element. If there is a div element MyDiv , you can locate it by giving class name ‘someclass’. |
| 8 | Css Selector | This criteria is useful when an HTML element has multiple associated css classes. Consider an HTML element . To identify this div element you can use Css selector as “.btn.primary-btn.submit” |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.20 Web Loop Table

### 1.20.1 Description

The step is used to traverse through an HTML table in Web GUI automation. Either entire table or specific columns can be the output.

The Web Loop Table step is designed to be used within a loop. On drag n drop of this new step, Continue step is also added on the canvas and the validations are same as the Start Loop-Continue Loop steps.

Each row of the table is processed for a single iteration (The rows to be processed are specified in the Loop Tab as All or specific rows or based on a condition). Processing of all the rows specified for a single iteration completes a loop.

The input row and additional fields are passed in the false path after all the rows (as specified in the Loop Tab) are processed.

**Validations:**

- If the workflow is using switch case for path decisions inside loop or error handling, all of them must end with Loop Continue Step.
- Both the steps must be present and connected properly in the workflow.
- Infinite loop will not be verified.
- Cannot have Blocking Steps in the loop
- Steps like Merge Join, Join Rows which wait for all rows before execution start will not be allowed in the loop.
- Row generating steps like Microsoft Excel Input are not allowed in a loop.

### 1.20.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |

| Web Table Tab: | | |
|---|---|---|
| No. | Field Name | Description |
| 2 | Selection Options | This is an implicit field with only one option, Entire Table with Specific Columns. |
| 3 | Criteria* | Choose the element locator from the drop down list. |
| 4 | Criteria value | Provide the value of the element locator to search with. |
| 5 | Multiple elements with same criteria | Check this option if there are multiple web elements in the web page having same criteria value you have specified. By default it will be unchecked. |
| 6 | Element index | If you have checked Multiple elements with same criteria specify index of the element you want to locate. |

| | | |
|---|---|---|
| | | You may also select option **FIRST** if you want first element from the located element and select **LAST** if you want last element. |
| | Get Values Table: | |
| 7 | Table Column Index | Specify column index to be retrieved. For multiple columns, mention column index on different rows. |
| 8 | Output Field Name | New field to store the retrieved value of the respective column for each row. |

| Loop Tab: | | |
|---|---|---|
| No. | Field Name | Description |
| 1 | Conditional Loop? | Checkbox to activate the Conditions block, else Number of rows will is active. |
| 2 | Number of rows | Specify the number of rows to pass through the loop. Values are in a dropdown,<br>• All Rows<br>• First<br>• Last<br>Also a text box is provided to enter a numeric value *N*, in case First or Last is selected. |
| | Table: Additional Fields | This table is used to create user defined additional fields. Additional fields are added as columns to the row. They are permanent and visible even after the loop is over.<br><br>Note: Any fields created within a loop are not permanently available after loop execution. Only Additional fields specified in Loop Start are available even after loop execution. |
| 3 | Name | Provide names of all additional fields you want to define. |
| 4 | Type | Select a data type for the additional field. |
| 5 | Format | Select a format for the data Type selected above. |
| 6 | Default value | Provide a default value for the additional field. |
| 7 | Set empty string? | Select Y or N.<br>If default value is not provided set ‘Set empty string?’ to Y. This sets a null value as an empty string. |
| 8 | Block: Conditions | Conditions Block is a set of five fields. These fields can be populated with Previous field, Addition fields or Static values. |

| | | |
|---|---|---|
| | For more detailed description with screen shots refer Start Loop step. | Click the field to populate a value as a toggle or from a popup that comes up.<br>The first field specifies a toggle for a condition. Leave it default to blank or click once to toggle it to NOT.<br>The second is the field to compare.<br>The third field is an operator<br>The fourth field is the field to be compared against or alternately specify a static value in the sixth field to be compared against.<br><br>Click on the small rectangular icon on the right hand side of the block to add a condition. You can Add a condition with the following operators: OR, AND, OR NOT, AND NOT, XOR. Multiple condition list by icon on right side<br><br>Right click on a condition to get a pop-up menu with the following options: Edit condition, Delete condition, Add sub-condition, Move condition to sub-condition (-->), Move condition to parent (<--) and more copy options. These options can be used to create Nested conditions. |
| 9 | Button: Get Fields | In the Conditions Block above only previous fields are available. Click Get Fields to make Additional fields also available in the conditions block. |

## *Criteria

The table below contains the description of criteria.

| No. | Field Name | Description |
|---|---|---|
| 1 | Xpath | Xpath defines the hierarchy of the tags through which you can reach to a particular element. For example /html/body/div[1]/div/div[3]/div[2]/div/div[2]/div/div/div[2]/div[2]/div[1]/div[1]/div[1]/span/span |
| 2 | Id | Id attribute of a web element can be best way to identify an element if it is present and is going to constant and unique. There are cases however when it is either absent or its value changes every time you access the application in the browser |
| 3 | Name | Name is another attribute of a web element which can be used to identify an element. |

| | | |
|---|---|---|
| 4 | Link Text | If a HTML page contains a hyperlink element <a href=www.google.com>Search On Google</a>, you can identify this href element by providing complete text 'Search On Google' as the Link value. |
| 5 | Partial Link text | This criteria is same as Link, just that, in this case, you can give partial string for finding a match |
| 6 | Tag Name | Every HTML element has a tag associated with it. The element can be located using that tag name. For example, textbox has a tag <input>. |
| 7 | Class Name | Class name is the css class associated with an HTML element. If there is a div element MyDiv , you can locate it by giving class name 'someclass'. |
| 8 | Css Selector | This criteria is useful when an HTML element has multiple associated css classes. Consider an HTML element . To identify this div element you can use Css selector as ".btn.primary-btn.submit" |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.21 Web Page Scroll

### 1.21.1 Description

The Web Page Scroll step is used to scroll a web page in Web GUI automation. It can be used to locate or access an element on the web page which requires page scrolling. Criteria and value are specified to locate the element to scroll to and the step will automatically scroll where ever needed.

### 1.21.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |
| 2 | Criteria* | Locate element on web page. |
| 3 | Criteria value | Relevant criteria value has to be specified. |

| | | |
|---|---|---|
| 4 | Multiple elements with same criteria | Check this option if there are multiple web elements in the web page having same criteria value you have specified. By default it will be unchecked. |
| 5 | Element index | If you have checked Multiple elements with same criteria specify index of the element you want to locate.<br>Select option FIRST if you want first element from the located element and select LAST if you want last element. |

## *Criteria

The table below contains the description of criteria.

| No. | Field Name | Description |
|---|---|---|
| 1 | Xpath | Xpath defines the hierarchy of the tags through which you can reach to a particular element. For example /html/body/div[1]/div/div[3]/div[2]/div/div[2]/div/div/div[2]/div[2]/div[1]/div[1]/div[1]/span/span |
| 2 | Id | Id attribute of a web element can be best way to identify an element if it is present and is going to constant and unique. There are cases however when it is either absent or its value changes every time you access the application in the browser |
| 3 | Name | Name is another attribute of a web element which can be used to identify an element. |
| 4 | Link Text | If a HTML page contains a hyperlink element <a href=www.google.com>Search On Google</a>, you can identify this href element by providing complete text ‘Search On Google’ as the Link value. |
| 5 | Partial Link text | This criteria is same as Link, just that, in this case, you can give partial string for finding a match |
| 6 | Tag Name | Every HTML element has a tag associated with it. The element can be located using that tag name. For example, textbox has a tag <input>. |
| 7 | Class Name | Class name is the css class associated with an HTML element. If there is a div element MyDiv , you can locate it by giving class name ‘someclass’. |
| 8 | Css Selector | This criteria is useful when an HTML element has multiple associated css classes. Consider an HTML element . To identify this div element you can use Css selector as “.btn.primary-btn.submit” |

| No. | Common Buttons | |
|---|---|---|

| | | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.22 Web Set Delay

### 1.22.1 Description

The Web Set Delay step sets the global delay value for further steps in Web GUI automation.

### 1.22.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |
| 2 | Delay (in milliseconds) | Global delay value to be set for the specified driver. |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.23 Web Set Retry Count

### 1.23.1 Description

Web Set Retry Count step is used to modify the Retry Count specified in Start Browser of Web GUI automation. Steps after Web Set Retry Count will get newly set Retry Count.

### 1.23.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |
| 2 | Retry Count | A retry count number can be set that is globally used in all steps of GUI Automation for retrying on error. Initially a retry count number can be set in Start Browser step. Whenever the execution of Web GUI step ends up with an error it will retry execution the number of time specified in start browser.<br><br>If one does'nt want to retry for some specific steps, retry count can be set to 0 by using 'Set Retry Count' Step. Use of 'Web Set Retry Count' is to modify the count specified in Start Browser. Steps after Web Set Retry Count will get newly set Retry Count.<br><br>Following steps are excluded from the functionality of Retry Count:<br><br>1. Begin<br>2. End<br>3. CloseWindow<br>4. Robot Handling<br>5. SetDelay<br>6. SetRetryCount<br>7. URL navigation |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.24 Web Set Value

### 1.24.1 Description

The Web Set Value step is used to set the text in the web element, example, text box, column in a table in Web GUI automation.

### 1.24.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |
| 2 | Criteria* | Locate element on web page. |
| 3 | Criteria value | Relevant criteria value has to be specified. |
| 4 | Text | Here you can specify the text you want to enter in to the element you have located. |
| 5 | Clear text before set? | Select **Yes** if you want to remove previously set value in the element you have located and **No** if you want to append the text in the element. |

*Criteria

The table below contains the description of criteria.

| No. | Field Name | Description |
|---|---|---|
| 1 | Xpath | Xpath defines the hierarchy of the tags through which you can reach to a particular element. For example /html/body/div[1]/div/div[3]/div[2]/div/div[2]/div/div/div[2]/div[2]/div[1]/div[1]/div[1]/span/span |
| 2 | Id | Id attribute of a web element can be best way to identify an element if it is present and is going to constant and unique. There are cases however when it is either absent or its value changes every time you access the application in the browser |
| 3 | Name | Name is another attribute of a web element which can be used to identify an element. |
| 4 | Link Text | If a HTML page contains a hyperlink element <a href=www.google.com>Search On Google</a>, you can identify this href element by providing complete text ‘Search On Google’ as the Link value. |
| 5 | Partial Link text | This criteria is same as Link, just that, in this case, you can give partial string for finding a match |

| | | |
|---|---|---|
| 6 | Tag Name | Every HTML element has a tag associated with it. The element can be located using that tag name. For example, textbox has a tag <input>. |
| 7 | Class Name | Class name is the css class associated with an HTML element. If there is a div element MyDiv , you can locate it by giving class name 'someclass'. |
| 8 | Css Selector | This criteria is useful when an HTML element has multiple associated css classes. Consider an HTML element . To identify this div element you can use Css selector as ".btn.primary-btn.submit" |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.25 Web Switch Frame

### 1.25.1 Description

The Web Switch Frame step switches between HTML frames in the web page in Web GUI automation.

### 1.25.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |
| 2 | Locate By* | Switching frame options |
| 3 | Criteria* | Locate element on web page |
| 4 | Criteria Value | Relevant criteria value has to be specified. |
| 5 | Multiple elements with same criteria | Check this option if there are multiple web elements in the web page having same criteria value you have specified. By default it will be unchecked. |
| 6 | Element index | If you have checked Multiple **elements with same criteria** specify index of the element you want to locate. |

| | | |
|---|---|---|
| | | Select option **FIRST** if you want first element from the located element and select **LAST** if you want last element. |

## *Locate By

The table below contains the description of each Locate By option**.**

| No. | Field Name | Description |
|---|---|---|
| 1 | Default | Navigate to default context of the page i.e. outer most region of web page |
| 2 | Parent | Navigate to parent frame |
| 3 | Name | Specify name of the frame to switch to |
| 4 | Id | Specify id of the frame to switch to |
| 5 | Element | Switch to frame where element of specific criteria is present |
| 6 | Index | Index of the frame you want to switch to |

## *Criteria

The table below contains the description of criteria.

| No. | Field Name | Description |
|---|---|---|
| 1 | Xpath | Xpath defines the hierarchy of the tags through which you can reach to a particular element. For example /html/body/div[1]/div/div[3]/div[2]/div/div[2]/div/div/div[2]/div[2]/div[1]/div[1]/div[1]/span/span |
| 2 | Id | Id attribute of a web element can be best way to identify an element if it is present and is going to constant and unique. There are cases however when it is either absent or its value changes every time you access the application in the browser |
| 3 | Name | Name is another attribute of a web element which can be used to identify an element. |
| 4 | Link Text | If a HTML page contains a hyperlink element <a href=www.google.com>Search On Google</a>, you can identify this href element by providing complete text ‘Search On Google’ as the Link value. |
| 5 | Partial Link text | This criteria is same as Link, just that, in this case, you can give partial string for finding a match |

| | | |
|---|---|---|
| 6 | Tag Name | Every HTML element has a tag associated with it. The element can be located using that tag name. For example, textbox has a tag <input>. |
| 7 | Class Name | Class name is the css class associated with an HTML element. If there is a div element MyDiv , you can locate it by giving class name 'someclass'. |
| 8 | Css Selector | This criteria is useful when an HTML element has multiple associated css classes. Consider an HTML element . To identify this div element you can use Css selector as ".btn.primary-btn.submit" |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.26 Web Switch Window

### 1.26.1 Description

The Web Switch Window step is useful when you want to switch between multiple browser windows in Web GUI automation. This basically gives the handle of desired window to perform further activity on that window.

### 1.26.2 Configuration

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |
| 2 | Switching options* | Options to switch window |
| 3 | Criteria* | Locate element on web page. By default this field will be disabled and will be enabled only when **Element** is selected as a switching option. |
| 4 | Value | Relevant value for the specified Switching option. |

*Switching Options

The table below contains the description of each Switching option.

| No. | Field Name | Description |
|---|---|---|
| 1 | Window Title | Option can be used when window title is known. |
| 2 | Element | Tries to find the specified element in all open browser windows and stops on the window where this element is found.<br>Criteria to locate element can be specified using criteria and value field. |
| 3 | URL | If there are multiple windows or tabs open in the browser with different URLs and you need to navigate to the window having specific URL. Specify URL in the value field. |
| 4 | Index | Every window open in the browser has the index starting from zero you can switch to any window using this index. |

## *Criteria

The table below contains the description of criteria.

| No. | Field Name | Description |
|---|---|---|
| 1 | Xpath | Xpath defines the hierarchy of the tags through which you can reach to a particular element. For example /html/body/div[1]/div/div[3]/div[2]/div/div[2]/div/div/div[2]/div[2]/div[1]/div[1]/div[1]/span/span |
| 2 | Id | Id attribute of a web element can be best way to identify an element if it is present and is going to constant and unique. There are cases however when it is either absent or its value changes every time you access the application in the browser |
| 3 | Name | Name is another attribute of a web element which can be used to identify an element. |
| 4 | Link Text | If a HTML page contains a hyperlink element <a href=www.google.com>Search On Google</a>, you can identify this href element by providing complete text ‘Search On Google’ as the Link value. |
| 5 | Partial Link text | This criteria is same as Link, just that, in this case, you can give partial string for finding a match |
| 6 | Tag Name | Every HTML element has a tag associated with it. The element can be located using that tag name. For example, textbox has a tag <input>. |

| | | |
|---|---|---|
| 7 | Class Name | Class name is the css class associated with an HTML element. If there is a div element MyDiv , you can locate it by giving class name 'someclass'. |
| 8 | Css Selector | This criteria is useful when an HTML element has multiple associated css classes. Consider an HTML element . To identify this div element you can use Css selector as ".btn.primary-btn.submit" |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.27 Web Table

### 1.27.1 Description

The Web Table step is used to traverse through HTML table in Web GUI automation. Either entire table or specific columns can be the output.

### 1.27.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |
| 2 | Selection Options* | There are three options to iterate through entire table or specific row with specific columns. |
| 3 | Criteria** | Locate element on web page. |
| 4 | Criteria value | Relevant criteria value to be specified. |
| 5 | Multiple elements with same criteria | Check this option if there are multiple web elements in the web page having same criteria value you have specified. By default it will be unchecked. |
| 6 | Element index | If you have checked Multiple elements with same criteria specify index of the element you want to locate. |

| | | |
|---|---|---|
| | | Select option **FIRST** if you want first element from the located element and select **LAST** if you want last element. |
| 7 | Table Column Index | Specify column index to be retrieved. For multiple columns, mention column index on different rows. |
| 8 | Output Field Name | Field to store the retrieved value of the respective column. |

## *Selection Option

The table below contains the description of each Selection Option.

| No. | Field Name | Description |
|---|---|---|
| 1 | Entire Table With Specific Columns | Traverse entire table and retrieve values from specific column(s). |
| 2 | One Row With Specific Columns | Retrieve one row with specific column(s). |
| 3 | One Cell | Retrieve value of one column of any specific row. |

## **Criteria

The table below contains the description of criteria.

| No. | Field Name | Description |
|---|---|---|
| 1 | Xpath | Xpath defines the hierarchy of the tags through which you can reach to a particular element. For example /html/body/div[1]/div/div[3]/div[2]/div/div[2]/div/div/div[2]/div[2]/div[1]/div[1]/div[1]/span/span |
| 2 | Id | Id attribute of a web element can be best way to identify an element if it is present and is going to constant and unique. There are cases however when it is either absent or its value changes every time you access the application in the browser |
| 3 | Name | Name is another attribute of a web element which can be used to identify an element. |
| 4 | Link Text | If a HTML page contains a hyperlink element <a href=www.google.com>Search On Google</a>, you can identify this href element by providing complete text ‘Search On Google’ as the Link value. |
| 5 | Partial Link text | This criteria is same as Link, just that, in this case, you can give partial string for finding a match |

| | | |
|---|---|---|
| 6 | Tag Name | Every HTML element has a tag associated with it. The element can be located using that tag name. For example, textbox has a tag <input>. |
| 7 | Class Name | Class name is the css class associated with an HTML element. If there is a div element MyDiv , you can locate it by giving class name 'someclass'. |
| 8 | Css Selector | This criteria is useful when an HTML element has multiple associated css classes. Consider an HTML element . To identify this div element you can use Css selector as ".btn.primary-btn.submit" |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.28 Web URL Navigation

### 1.28.1 Description

The Web URL Navigation step is used for navigating to another URL on the same browser window in Web GUI automation.

### 1.28.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |
| 2 | Web URL | URL to navigate to. |
| 3 | Delay (in milliseconds) | Page load timeout. |

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |

| | | |
|---|---|---|
| 3 | Cancel | Click CANCEL button to discard configurations. |

## 1.29 Web Wait Until

### 1.29.1 Description

The Web Wait Until step waits for an element to be visible, enabled or present on the web page in Web GUI automation. The maximum wait time is until the delay specified.

### 1.29.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Web Browser Name | You can select any one Web Browser Name from the given list, Basically you must select the same Web Browser Name which was selected in Start Browser Step. |
| 2 | Expected Condition* | Specify the wait condition for the element |
| 3 | Delay (in milliseconds) | Maximum time in milliseconds to wait for the expected condition. |
| 4 | Criteria* | Locate element on web page |
| 5 | Criteria Value | Relevant criteria value has to be specified. |

*Expected Condition

The table below contains the description of each condition.

| No. | Field Name | Description |
|---|---|---|
| 1 | Alert present | Check if alert present. |
| 2 | Element visible | Check if element with specific criteria is visible. |
| 3 | Element present | Check if element with specific criteria is present. |

*Criteria

The table below contains the description of criteria.

| No. | Field Name | Description |
|---|---|---|
| 1 | Xpath | Xpath defines the hierarchy of the tags through which you can reach to a particular element. For example /html/body/div[1]/div/div[3]/div[2]/div/div[2]/div/div/div[2]/div[2]/div[1]/div[1]/div[1]/span/span |
| 2 | Id | Id attribute of a web element can be best way to identify an element if it is present and is going to constant and unique. There are cases however when it is either absent or its value changes every time you access the application in the browser |
| 3 | Name | Name is another attribute of a web element which can be used to identify an element. |
| 4 | Link Text | If a HTML page contains a hyperlink element <a href=www.google.com>Search On Google</a>, you can identify this href element by providing complete text ‘Search On Google’ as the Link value. |
| 5 | Partial Link text | This criteria is same as Link, just that, in this case, you can give partial string for finding a match |
| 6 | Tag Name | Every HTML element has a tag associated with it. The element can be located using that tag name. For example, textbox has a tag <input>. |
| 7 | Class Name | Class name is the css class associated with an HTML element. If there is a div element MyDiv , you can locate it by giving class name ‘someclass’. |
| 8 | Css Selector | This criteria is useful when an HTML element has multiple associated css classes. Consider an HTML element . To identify this div element you can use Css selector as “.btn.primary-btn.submit” |

Notes: If Expected condition is ‘Alert present’, the Criteria and Criteria value fields are disabled.

| No. | Common Buttons | |
|---|---|---|
| 1 | Help | Click to see the Plugin description and configuration details. |
| 2 | OK | Click OK button to accept configurations. |
| 3 | Cancel | Click CANCEL button to discard configurations. |

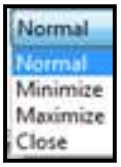
## 1.30 Windows Action

Windows Application GUI: Windows Action plugin step does window related actions such as Maximize, Minimize, Normal and Close.

Note:

- For Windows Action step you need to hover over the edges of the window to detect the window.
- Windows Action cannot be used, on pop-up windows, instead Mouse Action- Click, or Surface plugin –Surface Click or Robot Handing steps could be used.

### 1.30.1 Configurations

| Input Tab: | | |
|---|---|---|
| No. | Field Name | Description |
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |
| 2 | ActionType | Select an ActionType from one of the following,<br>  |
| 3 | Windows Path | Windows Path is the relative path of the desktop application on which element exists. It is captured by Windows Spy in the Generate Step dialog. It is then populated in the step configuration upon Step Generation. You may also enter or modify Windows path manually on the 'Step Generation' window or step configuration.<br><br>Windows path may be provided as a String or regex.<br>Sample of Windows Path as string:<br>w[Desktop]/w[Application Name]<br>or<br>Sample of Windows Path as regex:<br>w[Desktop]/w[/Application N/], w[Desktop]/w[/Application .*/],<br>If you are putting partial application name then put ii as a regex between / and /. Also if the Application name changes dynamically it is a good idea to provide it as a regex. |
| 4 | Timeout(In Seconds) | Specify a Timeout to find the element on the Desktop application. |

| Common Buttons: | | |
|---|---|---|
| No. | Field Name | Description |
| | Buttons: | |

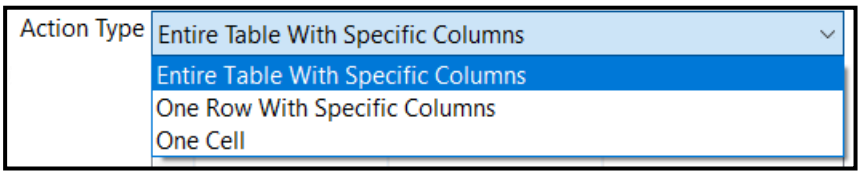
| | | |
|---|---|---|
| 1 | OK | On click of this button field values are checked. If any required field values are missing then validation error message is displayed.<br><br>If all the required field values are provided then it will save the field values. |
| 2 | Cancel | Click this button to cancel the window without saving any values. |
| 3 | Button: Edit Element | Click Edit Element to start spying.<br>Opening spy from step will be restricted to the step i.e. Only relevant elements will be detected from active spy. We can call it as a 'Step Restricted Spy' just for understanding purposes. Use Hotkeys ('Shift' or '~') to detect elements. E.g. Suppose spy is started from 'Window Action' step. Now spy is restricted to elements of type 'Window' only. All element of type 'Window' will be highlighted in blue color. If we try to detect element other than of type 'Window' it will be highlighted in 'Red' color upon hover. Notice Step dropdown is disabled and value selected is 'Window Action' |

## 1.31 Get Table Data [Beta]

### 1.31.1 Description

Windows: Get Table Data plugin step gets table from elements of type - List, Grid, Table or Data grid, filtered based on criteria provided.

### 1.31.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |
| 2 | Action Type | Select an Action Type from the drop down list.<br>  |

| | | |
|---|---|---|
| 3 | Windows Path | Windows Path is the relative path of the desktop application on which element exists. It is captured by Windows Spy in the Generate Step dialog. It is then populated in the step configuration upon Step Generation. You may also enter or modify Windows path manually on the 'Step Generation' window or step configuration.<br><br>Windows path may be provided as a String or regex.<br>Sample of Windows Path as string:<br>w[Desktop]/w[Application Name]<br>or<br>Sample of Windows Path as regex:<br>w[Desktop]/w[/Application N/], w[Desktop]/w[/Application .*/],<br>If you are putting partial application name then put ii as a regex between / and /. Also if the Application name changes dynamically it is a good idea to provide it as a regex. |
| 4 | Criteria | 'Criteria' is the criterion to locate element on Windows Desktop. It is captured by Windows Spy in the Generate Step dialog. Upon Step Generation it is populated in the step configuration. You may change criteria by choosing from the drop down list in 'Step Generation' window or step configuration.<br><br>Note: It is similar as in GUI Automation plugin steps. |
| 5 | Criteria Value | Specify relevant criteria value for the criteria provided above. |
| 6 | Element index(Starts with 0) | For Multiple elements with same criteria, specify index of the element you want to locate.<br>Provide index 0 if you want first element from the located elements, increment index by 1 for the next elements. |
| 7 | Row Index | Specify the Row Index to fetch data.<br>Row index is enabled only if one of the following is chosen in Action Type above,<br>• One Row With Specific Columns<br>• One Cell |
| 8 | Get Values (Tabular): | |
| i | Column Index | Specify the Column Indices to fetch data. |
| ii | Output Variable | Specify an output field to hold the value of Step execution. |
| 9 | Timeout(In seconds) | Specify a Timeout to find the element on the Desktop application. |

| Common Buttons: | | |
|---|---|---|
| No. | Field Name | Description |
| | Buttons: | |
| 1 | OK | On click of this button field values are checked. If any required field values are missing, then validation error message is displayed.<br><br>If all the required field values are provided then it will save the field values. |
| 2 | Cancel | Click this button to cancel the window without saving any values. |
| 3 | Button: Edit Element | Click Edit Element to start spying.<br>Opening spy from step is restricted to the step, i.e. Only relevant elements are detected from active spy. Use Hotkeys (‘Shift’ or ’~’) to detect the element. We may call it as a ‘Step Restricted Spy’ for understanding purposes. E.g. Suppose spy is started from ‘Get Table Data’ step. Now spy is restricted to elements of type ‘Table’ only. All element of type ‘Table’ will be highlighted in blue color. If we try to detect element other than of type ‘Table’ it will be highlighted in ‘Red’ color upon hover. Step dropdown is disabled and value selected is ‘Get Table Data’. |

# 2 Desktop

Windows Desktop plugin steps are used for Windows Desktop applications automation.

**Prerequisites:**

- Windows Operating System
- Install jdk and set JAVA_HOME, JRE_HOME environment variables.
- Windows .NET framework 4.6 and above
- Process studio distribution full path name should not have any spaces.
- Chrome browser is required to run DesktopSpy for AE releases prior to 5.4.0
- License is required to run DesktopSpy and Desktop plugins. License can be acquired and applied by contacting AutomationEdge Inc. A description on how to Apply License is in the section below.
- Desktop Plugin in Release 1.3 is compatible with AutomationEdge Release 5.4.0 onwards. For applet support GUI Automation Plugin1.3 is also required. Applet automation technology only works with the IE browser.
- Plugins1.2 and 1.3 with AutomationEdge R5.3.0 needs an extra external jar file. Please contact AutomationEdge Support for the same.
- Plugins1.2 and 1.3 with AutomationEdge below R5.3.0 needs multiple extra external jar files, hence not recommended.

**Desktop: Apply Licence**

**Apply Desktop Plugins License to Process Studio**

1. Once you get the license file copy or replace it in the following locations.
   - Process-Studio-distribution/psplugins/DesktopPlugins/**desktop-engine**/userdata/config(For Plugins)
   - Process-Studio-distribution/psplugins/DesktopPlugins/**desktop-spy**/userdata/config (for Spy) on your local environment.
2. Restart Process-studio and you are good to go.

| | |
|---|---|
| **Note:** | If the folders above are not present then follow the below steps:<br>1. If Process-Studio-distribution/psplugins/DesktopPlugins/**desktop-engine**/userdata/config" is not present, run sample desktop workflow and the folder is created.<br>2. If "Process-Studio-distribution/psplugins/DesktopPlugins/**desktop-spy**/userdata/config(for Spy)" is not present,<br>Navigate to tools -> Spy -> Desktop Spy. This opens Desktop Spy and the folder is created. |

### Apply Desktop Plugins License to Agents

1. Once you get the license file copy or replace it in the following location,
   - Agent Home/psplugins/DesktopPlugins/**desktop-engine**/userdata/config

| | |
|---|---|
| **Note:** | If the folder above is not present then follow the below step:<br>1. If Agent Home/psplugins/DesktopPlugins/**desktop-engine**/userdata/config" is not present, first import any desktop workflow and assign the workflow to the agent |

### References:

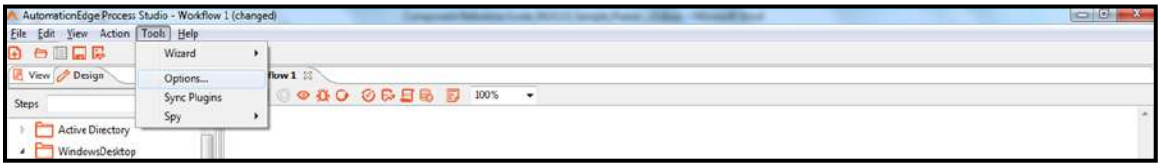
For a description on how to use Desktop Spy refer to Appendix 3: Desktop Spy Tutorial.

For sample workflows refer to, Project 8: Desktop Spy
in the AutomationEdge_R7.0.0_ProcesStudio_Activity_Guide.

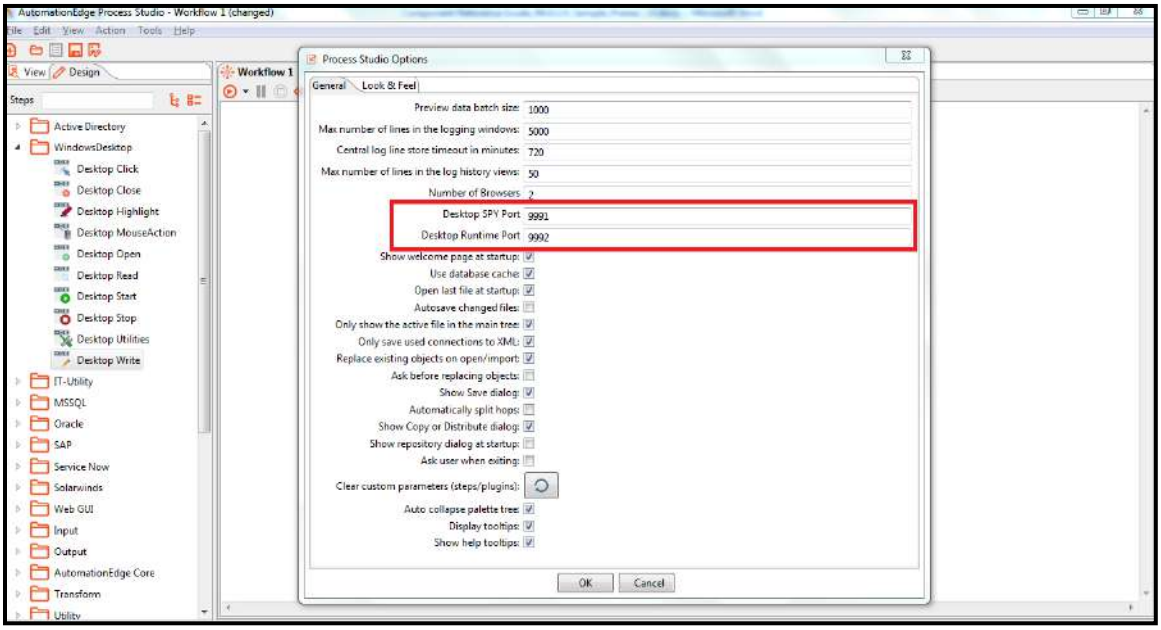
**Setup:**

Process Studio DesktopSpy is used to get many of the configuration parameters of Desktop plugin steps. To be able to use DesktopSpy we need to configure Process Studio Options Desktop SPY Port and Desktop Runtime Port. This is shown in the snapshots below.

1. In Process Studio go to Tools menu and Options sub-menu.




2. Set or change Desktop Spy Port and Desktop Runtime Port if required or leave it to the default values.




3. You are now ready to start using DesktopSpy with Desktop Plugins.

## 2.1 Desktop Click

### 2.1.1 Description

The step is used to perform click action on a Windows Desktop application element.

### 2.1.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |
| 2 | Prefix | Prefix is populated when working with popup windows and multiple domains. Prefix is meant to get an identifier for an instance of an application.<br>This field must be provided if it is returned by Desktop Spy. If Prefix is not returned by Desktop Spy it should be left blank.<br>If prefix is blank it assigns a default prefix. |
| 3 | Accessor/Alternative | Accessor is a Windows Desktop application element identifier for a selected prefix above. This field is populated with Accessor or alternative Accessor.<br>In case the Prefix field is blank it uses the default prefix. |
| 4 | Click Type | Click Type has three options:<br>1. Single Click<br>2. Double Click and<br>3. Right Click |
| 5 | Mode | Select 'WINDOWS' or 'JAVA_APPLET' from the drop down list. Generally WINDOWS mode is used for .Net Windows-based applications while JAVA_APPLET is used for applet based applications(e.g. Oracle E-business suite application) |

## 2.2 Desktop Close

### 2.2.1 Description

The step is used to close the current Windows Desktop application element.

### 2.2.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |

| | | |
|---|---|---|
| 2 | Prefix | Prefix is populated when working with popup windows and multiple domains. Prefix is meant to get an identifier for an instance of a windows application.<br>This field must be provided if it is returned by Desktop Spy. If Prefix is not returned by Desktop Spy it should be left blank.<br>If prefix is blank it assigns a default prefix. |
| 3 | Mode | Select 'WINDOWS' or 'JAVA_APPLET' from the drop down list. Generally WINDOWS mode is used for .Net Windows-based applications while JAVA_APPLET is used for applet based applications(e.g. Oracle E-business suite application) |

## 2.3 Desktop Highlight

### 2.3.1 Description

The step is used to highlight an element on a Windows Desktop Application. This step is also useful during Process Studio Workflow development and debugging.

### 2.3.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |
| 2 | Prefix | Prefix is populated when working with popup windows and multiple domains. Prefix is meant to get an identifier for an instance of an application.<br>This field must be provided if it is returned by Desktop Spy. If Prefix is not returned by Desktop Spy it should be left blank.<br>If prefix is blank it assigns a default prefix. |
| 3 | Accessor/Alternative | Accessor is a Windows Desktop application element identifier for a selected prefix above. This field is populated with Accessor or alternative Accessor.<br>In case the Prefix field is blank it uses the default prefix. |
| 4 | Mode | Select 'WINDOWS' or 'JAVA_APPLET' from the drop down list. Generally WINDOWS mode is used for .Net Windows-based applications while JAVA_APPLET is used for applet based applications(e.g. Oracle E-business suite application) |

## 2.4 Desktop MouseAction

### 2.4.1 Description

The step is used to perform Mouse Actions on Windows Desktop application elements.

### 2.4.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |
| 2 | Prefix | Prefix is populated when working with popup windows and multiple domains. Prefix is meant to get an identifier for an instance of an application.<br>This field must be provided if it is returned by Desktop Spy. If Prefix is not returned by Desktop Spy it should be left blank.<br>If prefix is blank it assigns a default prefix. |
| 3 | Accessor/Alternative | Accessor is a Windows Desktop application element identifier for a selected prefix above. This field is populated with Accessor or alternative Accessor.<br>In case the Prefix field is blank it uses the default prefix. |
| 4 | MouseAction Type | Choose a value from the drop down list from the following values.<br>1. Drag & Drop<br>2. Mouse Hover |
| 5 | Destination prefix | If Drag & Drop is chosen above Destination prefix becomes active. Provide a value for the destination elements prefix. |
| 6 | Destination Accessor/Alternative | If Drag & Drop is chosen above Destination Accessor/Alternative also becomes active. Provide a value for the destination elements Accessor/Alternative. |
| 7 | Mode | Select 'WINDOWS' or 'JAVA_APPLET' from the drop down list. Generally WINDOWS mode is used for .Net Windows-based applications while JAVA_APPLET is used for applet based applications(e.g. Oracle E-business suite application) |

## 2.5 Desktop Open

### 2.5.1 Description

The step is used to open a Windows Application.

### 2.5.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |
| | Prefix | In this plugin Prefix is usually the title for the application window.<br><br>If we need to specify a wait time for an application window (with a specified prefix) to open then provide values in prefix as well as Windows load timeout field below. Both are mandatory.<br><br>Note: If the application window with the specified prefix is already open or not found in the provided time then the step as well as workflow fails. |
| 2 | Path of Application | Provide the full path of the Windows Desktop Application executable. |
| 3 | Windows load timeout | This is maximum time provided after opening the application to load application window elements. |
| 4 | Mode | Select 'WINDOWS' or 'JAVA_APPLET' from the drop down list. Generally WINDOWS mode is used for .Net Windows-based applications while JAVA_APPLET is used for applet based applications(e.g. Oracle E-business suite application) |

## 2.6 Desktop Read

### 2.6.1 Description

The step is used to read value of any element on the Windows Application.

### 2.6.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |
| 2 | Prefix | Prefix is populated when working with popup windows and multiple domains. Prefix is meant to get an identifier for an instance of an application.<br>This field must be provided if it is returned by Desktop Spy. If Prefix is not returned by Desktop Spy it should be left blank.<br>If prefix is blank it assigns a default prefix. |

| Read Parameter Tab: | | |
|---|---|---|
| No. | Field Name | Description |
| | Input Fields: | |
| 1 | Accessor/Alternative | Accessor is a Windows Desktop application element identifier for a selected prefix above. This field is populated with Accessor or alternative Accessor.<br>In case the Prefix field is blank it uses the default prefix. |
| 2 | Read Type | Read Type can be Text or Value depending on the element type. (For e.g. for Button Label Read Type is text and for field Read Type is value). |
| 3 | Output | Provide a name of a field to store the value read. |
| 4 | Mode | Select 'WINDOWS' or 'JAVA_APPLET' from the drop down list. Generally WINDOWS mode is used for .Net Windows-based applications while JAVA_APPLET is used for applet based applications(e.g. Oracle E-business suite application). |

## 2.7 Desktop Start

### 2.7.1 Description

This step is a start node to be able to use any succeeding Desktop plugin steps in a workflow.

### 2.7.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |
| 2 | Start node | Enable the checkbox if this step is the first step in the workflow. |
| 3 | Applet Support | Enable checkbox for Applet support in the workflow. |
| 4 | Open Application | If this checkbox is enabled the three fields below are also enabled else the three fields below are disabled.<br><br>If this checkbox is disabled this step is used to simply starts Desktop plugin driver.<br>However, If this checkbox is enabled it starts Desktop plugin driver. Additionally, the step can be used to open any Windows Application. |
| 5 | Prefix | Prefix is populated when working with popup windows and multiple domains. Prefix is meant to get an identifier for an |

| | | |
|---|---|---|
| | | instance of an application window. Prefix is usually the title for the application window.<br>This field must be provided if it is returned by Desktop Spy. If Prefix is not returned by Desktop Spy it should be left blank.<br>If prefix is blank it assigns a default prefix.<br><br>If we need to specify a wait time for an application window (with a specified prefix) to open then provide values in prefix as well as Windows load timeout field below. Both are mandatory.<br><br>Note: If the application window with the specified prefix is already open or not found in the provided time then the step as well as workflow fail. |
| 6 | Path of Application | Provide the full path of the Windows Desktop Application executable |
| 7 | Windows load timeout(in millisecond) | This is maximum time provided after opening the application to load application window elements. |
| 8 | Mode | Select 'WINDOWS' or 'JAVA_APPLET' from the drop down list. Generally WINDOWS mode is used for .Net Windows-based applications while JAVA_APPLET is used for applet based applications(e.g. Oracle E-business suite application) |

## 2.8 Desktop Stop

### 2.8.1 Description

The step is used to stop Desktop plugin engine.

### 2.8.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |

## 2.9 Desktop Utilities

### 2.9.1 Description

The step is used to check the status of Desktop utilities.

### 2.9.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |
| 2 | Prefix | Prefix is populated when working with popup windows and multiple domains. Prefix is meant to get an identifier for an instance of an application.<br>This field must be provided if it is returned by Desktop Spy. If Prefix is not returned by Desktop Spy it should be left blank.<br>If prefix is blank it assigns a default prefix. |
| 3 | Accessor/Alternative | Accessor is a Windows Desktop application element identifier for a selected prefix above. This field is populated with Accessor or alternative Accessor.<br>In case the Prefix field is blank it uses the default prefix. |
| 4 | Utilities Type | Select a value from the drop down list.<br>• Exists<br>• Is Checked<br>• Is Enabled<br>• Is Visible<br>• Application Exists |
| 5 | Output | Provide a field name to store the value of output. This is a Boolean value. |
| 6 | Mode | Select 'WINDOWS' or 'JAVA_APPLET' from the drop down list. Generally WINDOWS mode is used for .Net Windows-based applications while JAVA_APPLET is used for applet based applications(e.g. Oracle E-business suite application) |

## 2.10 Desktop WaitUntil

### 2.10.1 Description

The Desktop WaitUntil step waits until a desktop application window is open or exists or element is present, visible, checked or enabled on the desktop application window in Desktop automation. The maximum wait time is until the timeout specified. The Desktop WaitUntil step can also wait for reverse of the above conditions.

### 2.10.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |
| 2 | Prefix | Prefix is populated when working with popup windows and multiple domains. Prefix is meant to get an identifier for an instance of an application.<br>This field must be provided if it is returned by Desktop Spy. If Prefix is not returned by Desktop Spy it should be left blank.<br>If prefix is blank it assigns a default prefix. |
| 3 | Accessor/Alternative | Accessor is a Windows Desktop application element identifier for a selected prefix above. This field is populated with Accessor or alternative Accessor.<br>In case the Prefix field is blank it uses the default prefix. |
| 4 | Wait until condition | Specify the wait condition for the element.<br><br>Application Exists: Check if an Application with specified Prefix exists.<br>Exists: Check if element with specified Prefix and Accessor/Alternative exists.<br>Is Visible: Check if element with specified Prefix and Accessor/Alternative is visible.<br>Is Checked: Check if element with specified Prefix and Accessor/Alternative is checked.<br>Is Enabled: Check if element with specified Prefix and Accessor/Alternative is enabled. |
| 5 | reverse of above condition | Enable checkbox to check the reverse of the above conditions. In other words it checks the following,<br><br>Application does not Exists: Check if an Application with specified Prefix does not exist.<br>Not Exists: Check if element with specified Prefix and Accessor/Alternative does not exist.<br>Is Not Visible: Check if element with specified Prefix and Accessor/Alternative is not visible.<br>Is Not Checked: Check if element with specified Prefix and Accessor/Alternative is not checked.<br>Is Not Enabled: Check if element with specified Prefix and Accessor/Alternative is not enabled. |

| | | |
|---|---|---|
| 6 | Timeout (in seconds) | Maximum time in seconds to wait for the expected wait until condition else to timeout and fail the step. |
| 7 | Mode | Select 'WINDOWS' or 'JAVA_APPLET' from the drop down list. Generally WINDOWS mode is used for .Net Windows-based applications while JAVA_APPLET is used for applet based applications(e.g. Oracle E-business suite application) |

## 2.11 Desktop Write

### 2.11.1 Description

The step is used to write text in a windows desktop application.

### 2.11.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |
| 2 | Prefix | Prefix is populated when working with popup windows and multiple domains. Prefix is meant to get an identifier for an instance of an application.<br>This field must be provided if it is returned by Desktop Spy. If Prefix is not returned by Desktop Spy it should be left blank.<br>If prefix is blank it assigns a default prefix. |

| Write Parameter Tab: | | |
|---|---|---|
| No. | Field Name | Description |
| | Fields: | |
| 1 | Accessor/Alternative | Accessor is a Windows Desktop application element identifier for a selected prefix above. This field is populated with Accessor or alternative Accessor.<br>In case the Prefix field is blank it uses the default prefix. |
| 2 | Write Type | Write Type can be Set New Text or Append Text depending on the activity required. |
| 3 | Input Text | Provide the Text to be set as New Text or to Append. |
| 4 | Mode | Select 'WINDOWS' or 'JAVA_APPLET' from the drop down list. |

| | | |
|---|---|---|
| | | Generally WINDOWS mode is used for .Net Windows-based applications while JAVA_APPLET is used for applet based applications(e.g. Oracle E-business suite application) |

# 3 Surface

Surface plugin has been designed with image recognition capabilities to capture images/regions on a screen. The captured image can be used for image search, mouse Actions on the image, image labeling and optical character recognition (OCR).

Surface Automation is particularly useful on remote machines where we do not have any handle to the content and everything is an image or pattern. It can be useful to work on remote environments such as Citrix.

**Prerequisite:**

- For surface plugin steps, Microsoft Visual C++ version 2015-2019 Redistributable (x64) and Microsoft Visual C++ version 2015-2019 Redistributable (x86) should be installed.
    - This is available at the following URL,
    - https://support.microsoft.com/en-in/help/2977003/the-latest-supported-visual-c-downloads
    - Download the following two executable,
        - x86: vc_redist.x86.exe
        - x64: vc_redist.x64.exe

- Surface spy is required to build workflows with Surface plugin steps. For a description on how to use Surface Spy refer the following,

**Additional Prerequisite for Windows Server OS:**

Surface Plugins use some of Windows Media Service DLL's (ex. mfplat.dll) for image recognition. You need to enable “Desktop Experience” windows feature to get the required DLL's. Following are the steps to install the required Windows Media Service- DLL's

1. On Windows Start menu search for ‘Turn windows features on or off’
2. Skip the roles screen and directly go to Feature screen
3. Select "Desktop Experience" under "User Interfaces and Infrastructure"
4. Restart the system.

**Limitation:** Any type of UI Automation with Surface plugin steps is not supported in Linux based OS.

**References:**

For a description on how to use Surface Spy refer to Appendix 4: Surface Spy Tutorial

For a sample workflow refer to Project 8: Surface Spy
in the AutomationEdge_R7.0.0_ProcesStudio_Activity_Guide.

## 3.1 Surface Click

### 3.1.1 Description

Surface Click step is used to perform a click action on an image on the screen.

### 3.1.2 Configurations

<table>
<tr><th>No.</th><th>Field Name</th><th>Description</th></tr>
<tr><td>1</td><td>Step Name</td><td>Name of the step. This name has to be unique in a single workflow.</td></tr>
<tr><td>2</td><td>Match Pattern</td><td>• Retrieve Single closest match: It searches for and retrieves a single closest match. If the image is not found or multiple matches are found then it throws an error.<br>• Retrieve Multiple close matches: It searches for matching patterns and retrieves the closest matches.<br>• Retrieve All matches: It searches for and retrieves all matches (not below 50% or above 150% of the original size).</td></tr>
<tr><td>3</td><td>Click Type</td><td>Click Type has five options as below:
<table>
<tr><th>No</th><th>Click Types</th><th>Description</th></tr>
<tr><td>1</td><td>Click</td><td>It performs left click action.</td></tr>
<tr><td>2</td><td>Right Click</td><td>It performs right click action.</td></tr>
<tr><td>3</td><td>Double Click</td><td>It performs double-click action.</td></tr>
<tr><td>4</td><td>Middle Click</td><td>It performs left Middle click action.</td></tr>
<tr><td>5</td><td>Mouse Move</td><td>It performs Mouse Move action.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><td>4</td><td>Position</td><td>Choose position on the selected image to perform the click action.
<table>
<tr><th>No</th><th>Positions</th><th>Description</th></tr>
<tr><td>1</td><td>Left</td><td>Center Left position of the Image.</td></tr>
<tr><td>2</td><td>Right</td><td>Center Right position of the Image.</td></tr>
<tr><td>3</td><td>Center</td><td>Center of the Image.</td></tr>
<tr><td>4</td><td>Custom</td><td>Relative position from the entire Image. It could be anywhere on the screen. It is preferable to have the relative position of the relative element to click is closer to the current position.</td></tr>
</table></td></tr>
<tr><td>5</td><td>Image Index</td><td>Image index is generated by the image locator of surface spy.</td></tr>
<tr><td>9</td><td>Timeout(in seconds)</td><td>Specify a timeout in seconds to locate the image.</td></tr>
<tr><td>10</td><td>Image</td><td>View the selected image.</td></tr>
</table>

| No. | Common Buttons | |
|---|---|---|
| 1 | OK | Click OK button to accept configurations. |
| 2 | Cancel | Click CANCEL button to discard configurations. |
| 3 | Edit Image(s) | Click Edit Image(s) button to show the Surface Spy window. You can perform capture and locate functions and you can also change editable step properties. |

## 3.2 Surface Find Image

### 3.2.1 Description

Surface Find Image step is used to find an image with on the screen. It returns a Boolean value true if image is found else it returns false.

### 3.2.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |

| | | |
|---|---|---|
| 2 | Match Pattern | • Retrieve Single closest match: It searches for and retrieves a single closest match. If the image is not found or multiple matches are found then it throws an error.<br>• Retrieve Multiple close matches: It searches for matching patterns and retrieves the closest matches.<br>• Retrieve All matches: It searches for and retrieves all matches (not below 50% or above 150% of the original size). |
| 3 | Output field | Specify a field name to hold the output Boolean True or false |
| 4 | Timeout(in seconds) | Specify a timeout in seconds to locate the image. |
| 5 | Image | View the selected image. |

| No. | Common Buttons | |
|---|---|---|
| 1 | OK | Click OK button to accept configurations. |
| 2 | Cancel | Click CANCEL button to discard configurations. |
| 3 | Edit Image(s) | Click Edit Image(s) button to show the Surface Spy window. You can perform capture and locate functions and you can also change editable step properties. |

## 3.3 Surface Get Value

### 3.3.1 Description

Surface Get value step is used to get value on an image on the screen.

### 3.3.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |
| 2 | Match Pattern | • Retrieve Single closest match: It searches for and retrieves a single closest match. If the image is not found or multiple matches are found then it throws an error.<br>• Retrieve Multiple close matches: It searches for matching patterns and retrieves the closest matches.<br>• Retrieve All matches: It searches for and retrieves all matches (not below 50% or above 150% of the original size). |

| | | |
|---|---|---|
| 3 | Position | Position has a default value Custom. Custom position is based on a relative position to a position on an image. |
| 4 | Image Index | Image index is generated by the image locator on surface spy. It is identified by the image locator on surface spy.<br><br>locating the image on the screen<br>Image index automatically populated the Image Index and is the preferred way.<br><br>You may set this manually, however locating the image on the screen automatically populates the Image Index and is the preferred way. |
| 5 | Output field | Specify a field to hold the extracted text. |
| 6 | Timeout(in seconds) | Specify a timeout in seconds to get value from the image. |
| 7 | Convert To grey scale | In case the step is not able to read characters in an image accurately you may enable Convert to grey scale checkbox to fetch better results. |
| 8 | Image | Depicts the image with respect to which we need to Get Value. |

| No. | Common Buttons | |
|---|---|---|
| 1 | OK | Click OK button to accept configurations. |
| 2 | Cancel | Click CANCEL button to discard configurations. |
| 3 | Edit Image(s) | Click Edit Image(s) button to show the Surface Spy window. You can perform capture and locate functions and you can also change editable step properties. |

## 3.4 Surface Set Value

### 3.4.1 Description

Surface Set Value step is used to set value on an image on the screen.

### 3.4.2 Configurations

| No. | Field Name | Description |
|---|---|---|
| 1 | Step Name | Name of the step. This name has to be unique in a single workflow. |

<table>
<tr><td>2</td><td>Match Pattern</td><td><ul><li>Retrieve Single closest match: It searches for and retrieves a single closest match. If the image is not found or multiple matches are found then it throws an error.</li><li>Retrieve Multiple close matches: It searches for matching patterns and retrieves the closest matches.</li><li>Retrieve All matches: It searches for and retrieves all matches (not below 50% or above 150% of the original size).</li></ul></td></tr>
<tr><td>3</td><td>Position</td><td>Choose position on the selected image to Set Value action.<table><tr><th>No</th><th>Positions</th><th>Description</th></tr><tr><td>1</td><td>Left</td><td>Center Left position of the Image.</td></tr><tr><td>2</td><td>Right</td><td>Center Right position of the Image.</td></tr><tr><td>3</td><td>Center</td><td>Center of the Image.</td></tr><tr><td>4</td><td>Custom</td><td>Relative position from the entire Image. It could be anywhere on the screen. If the relative position of the relative element to click is closer it is preferable.</td></tr></table></td></tr>
<tr><td>4</td><td>Image Index</td><td>Image index is generated by the image locator of surface spy.</td></tr>
<tr><td>5</td><td>Input Type</td><td>Input Type has two options as below:<table><tr><th>No</th><th>Click Types</th><th>Description</th></tr><tr><td>1</td><td>Type</td><td>It types the characters one by one.</td></tr><tr><td>2</td><td>Set Value</td><td>It sets the entire text in one go. (similar to pasting text)</td></tr></table></td></tr>
<tr><td>6</td><td>Value</td><td></td></tr>
<tr><td>7</td><td>Timeout(in seconds)</td><td>Specify a timeout in seconds to locate the image.</td></tr>
<tr><td>8</td><td>Image</td><td>Depicts the image with respect to which we need to Set Value.</td></tr>
</table>

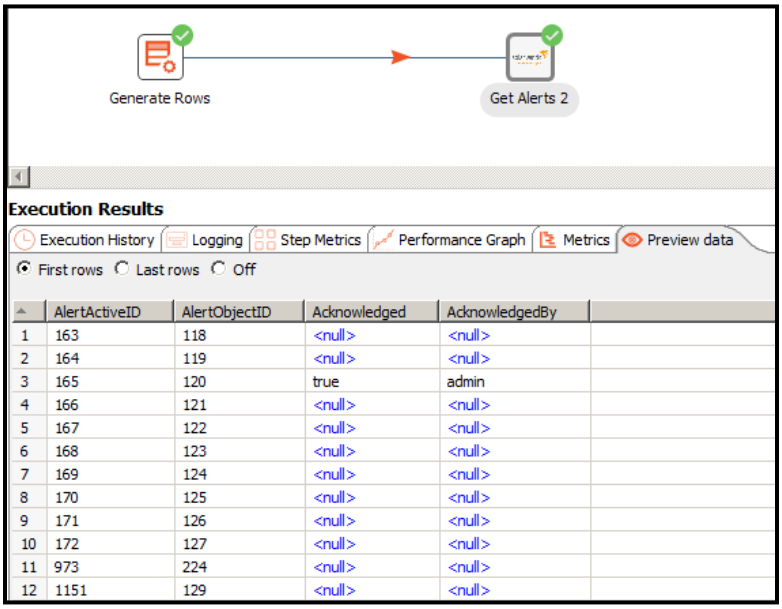
| No. | Common Buttons | |
|---|---|---|
| 1 | OK | Click OK button to accept configurations. |
| 2 | Cancel | Click CANCEL button to discard configurations. |
| 3 | Edit Image(s) | Click Edit Image(s) button to show the Surface Spy window. You can perform capture and locate functions and you can also change editable step properties. |

# IV. Appendices

# 1 Appendix 1: Error Handling in IT Plugins

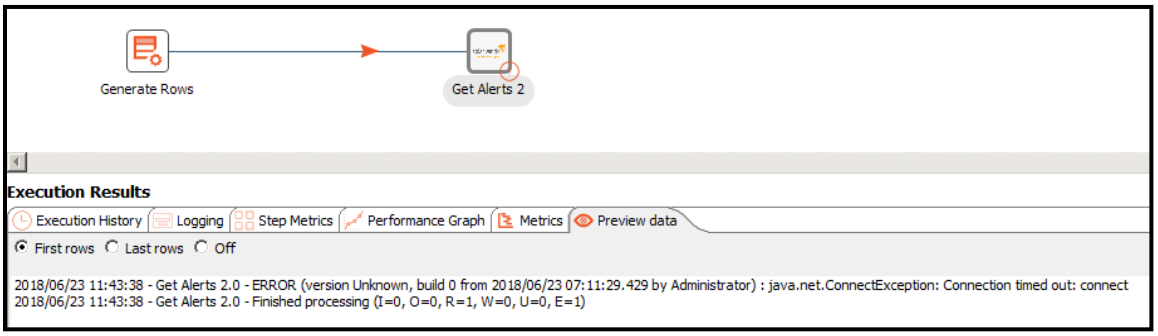
This appendix describes Error Handling for IT plugins.

1. Upon Successful execution of steps in Process Studio a workflow completes successfully and any output fields data is displayed.

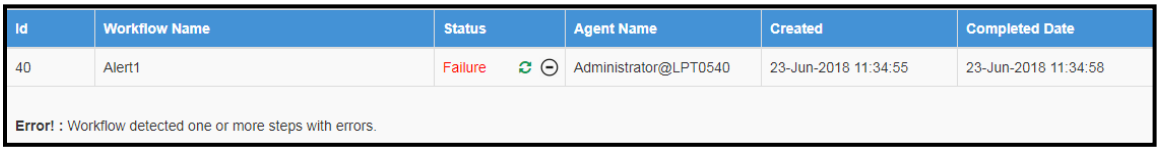



| | AlertActiveID | AlertObjectID | Acknowledged | AcknowledgedBy |
|---|---|---|---|---|
| 1 | 163 | 118 | <null> | <null> |
| 2 | 164 | 119 | <null> | <null> |
| 3 | 165 | 120 | true | admin |
| 4 | 166 | 121 | <null> | <null> |
| 5 | 167 | 122 | <null> | <null> |
| 6 | 168 | 123 | <null> | <null> |
| 7 | 169 | 124 | <null> | <null> |
| 8 | 170 | 125 | <null> | <null> |
| 9 | 171 | 126 | <null> | <null> |
| 10 | 172 | 127 | <null> | <null> |
| 11 | 973 | 224 | <null> | <null> |
| 12 | 1151 | 129 | <null> | <null> |

2. In case any step fails the workflow aborts.




3. On AutomationEdge the Error Message displayed is- Workflow detected one or more steps with errors.

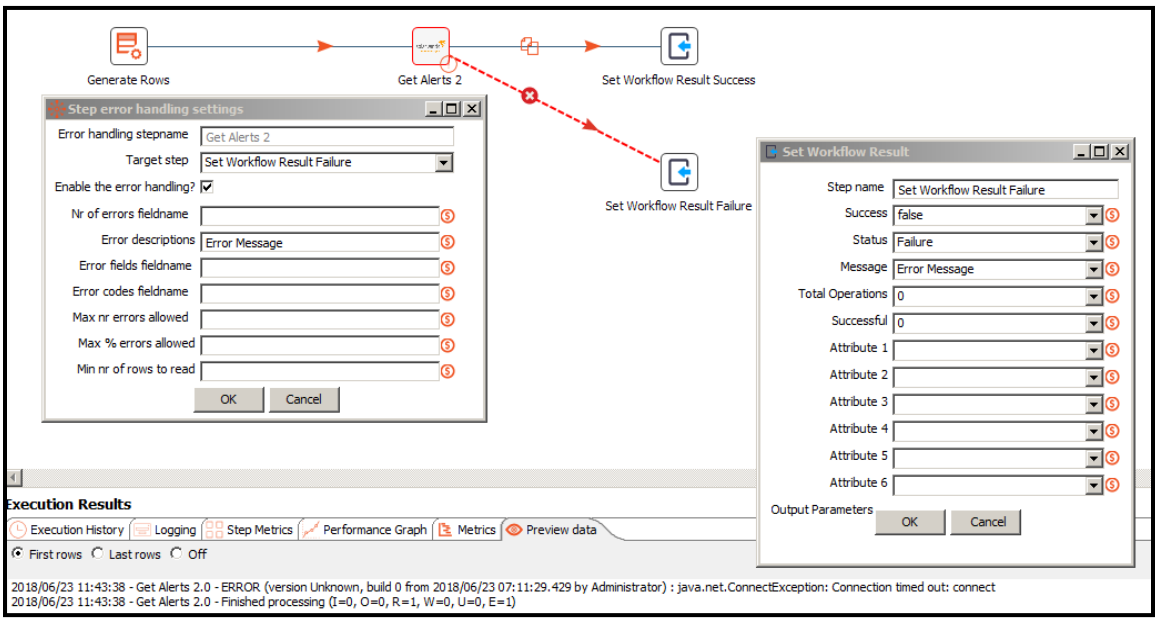


| Id | Workflow Name | Status | Agent Name | Created | Completed Date |
|---|---|---|---|---|---|
| 40 | Alert1 | Failure | Administrator@LPT0540 | 23-Jun-2018 11:34:55 | 23-Jun-2018 11:34:58 |

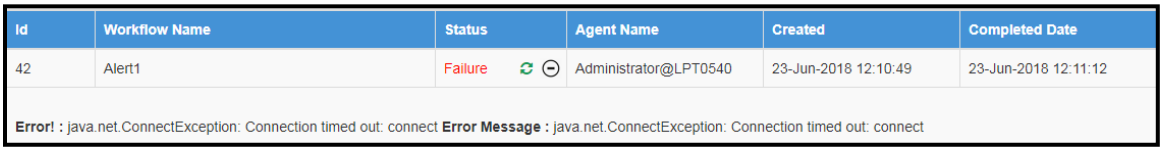
**Error! :** Workflow detected one or more steps with errors.

4. To get the exact error message as displayed in Process Studio error handling needs to be done in the process studio workflow as shown below.

- Add two Set Workflow Result steps as seen below, one for workflow success and the other one for error handling.
- Now right click the IT Plugin step (Alert) and provide a name for Error Descriptions field (In this case 'Error Message).
- In the Set Workflow Result Failure select this field, 'Error Message' from the Message drop down combo box.




5. If the workflow is now published on AutomationEdge and run, it displays the complete error message as shown below.6 h



# 2 Appendix 2: GUI Spy Tutorial

## 2.1 Introduction

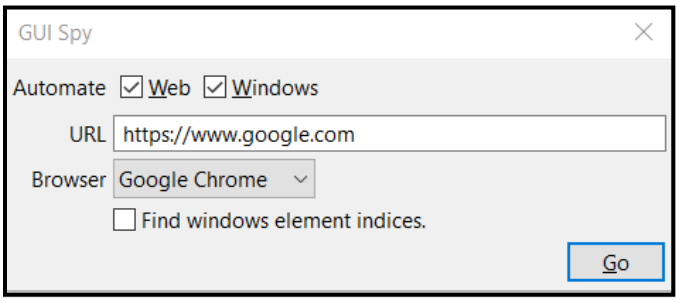
Process Studio GUI Spy is a tool for automating GUI RPA (robotic process automation). GUI Spy can extract both web and windows element attributes, giving users a seamless GUI automation experience.

GUI Spy can generate individual Steps on the workflow. Alternatively, use the GUI Spy Recorder to record several steps and place them on the workflow in one go.

## 2.2 GUI Spy

The following is a snapshot of the GUI Spy dialog on Window OS.

- Enable checkbox for Web, Windows or Web + Windows.
- Set the URL to spy for Web elements and choose a browser from the list. Clicking on Go opens the GUI Spy toolbar.
- 'Find windows element indices checkbox' is applicable only for windows applications. Enable the checkbox for more robust automation; however, note that element detection will be slower in this scenario.




The following is a snapshot of the GUI Spy pop-up on Linux OS. Linux support


The table below summarizes the Automate mode behaviour for Linux and Windows operating systems.

**Table: Windows and Linux GUI Automation Behaviour**

| OS | Automate Mode | Behaviour |
|---|---|---|
| Windows/ Linux | Web | • Detects all Web elements<br><br>**Notes:**<br>• If GUI Spy opens multiple browser windows, it is advisable to switch windows from the GUI Spy toolbar list as it will automatically display dialogue for confirmation to add a Web Switch Window step.<br>• On the other hand, if you switch to another window manually, it does not show any Switch step creation dialogue, and you will have to add the step manually in the workflow.<br>• Linux supports only Web mode. |
| Windows | Windows | • Detects all Windows elements<br>• Here the elements are detected and automated using windows automation. This mode is not advisable for web automation. |
| | Web + Windows | • Detects all Web and Windows elements<br>• Web mode gets priority over Windows mode while detecting web elements.<br><br>**Notes:**<br>• If GUI Spy opens multiple browser windows, switching windows through the GUI Spy toolbar or manually displays a dialogue for confirmation to add a Web Switch Window step. |

## 2.3 GUI Spy Toolbar

GUI Spy toolbar is compact and occupies a minimum screen area to spy on a web page with ease. Upon Spying, it opens the Generate Step dialog with two sections- Attributes and Step Generation for the web; make configuration changes if required and generates steps.

GUI Spy operates in three modes,

i. Web
(Enable only Web checkbox)
The GUI Spy toolbar for the web is as follows,


ii. Windows
(Enable only Windows checkbox)
The GUI Spy toolbar for windows is as follows,

iii. Web + Windows
(Enable Web + Windows checkbox)
The GUI Spy toolbar for the web and windows is the same as the GUI Spy toolbar for the web.


## 2.4 Shortcut/Hot Keys

All web and windows elements can be detected using GUI Spy. Following are the shortcut keys available for GUI Spy.

1. Ctrl + Alt + W – To open Windows Spy. Windows Spy Interface opens, as seen below. Notice that the background of the lens is orange, which signifies that the GUI Spy is not in active mode. Click on the lens icon.

or


2. The background of the magnifying lens changes to white which signifies that Spy is in active mode, and you are ready to spy.

or


3. Mouse hover over the elements to Spy. Press keyboard key ‘right shift’ or back tick (`) to open the Generate Step dialog with element information; change configurations if required.
4. Close any active GUI Spy by clicking ( )or ( with Ctrl+Q).

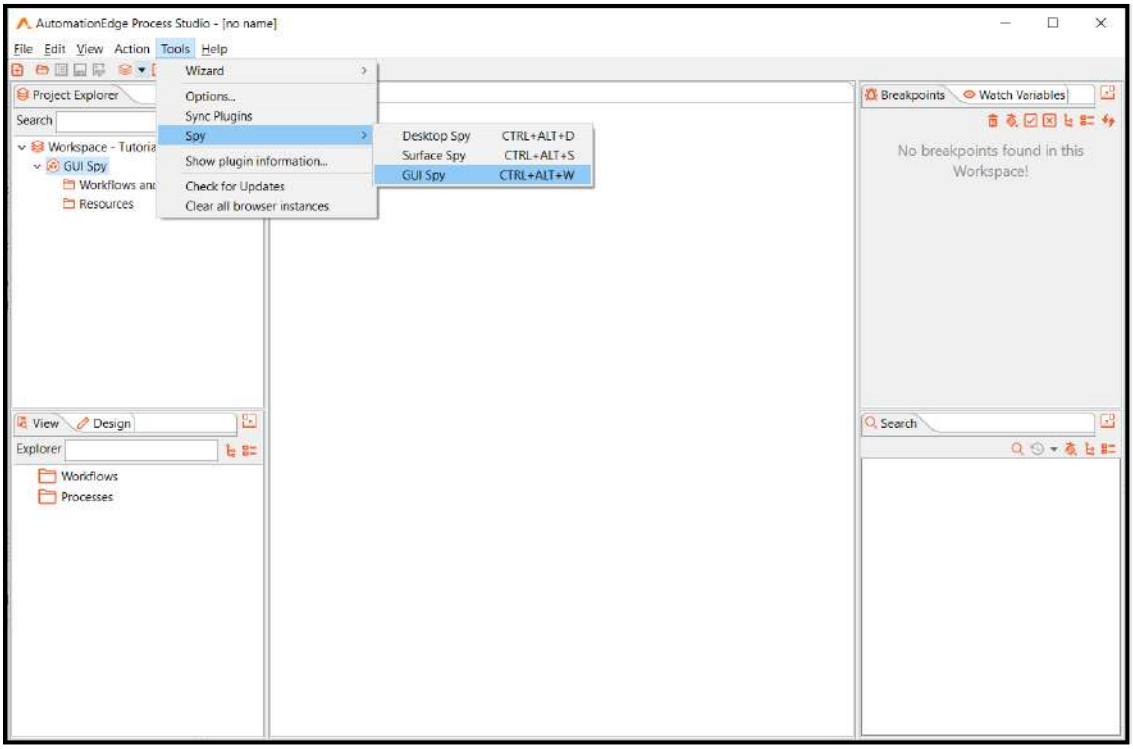
The following sections provide a step-by-step demo of GUI Spy for Web and Windows.

## 2.5 GUI Spy for Web

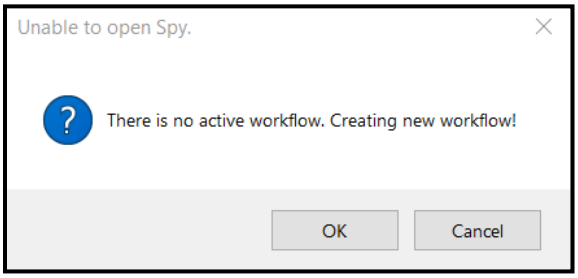
### 2.5.1 Start using GUI Spy

Follow the steps below to launch and use GUI Spy to create a workflow.

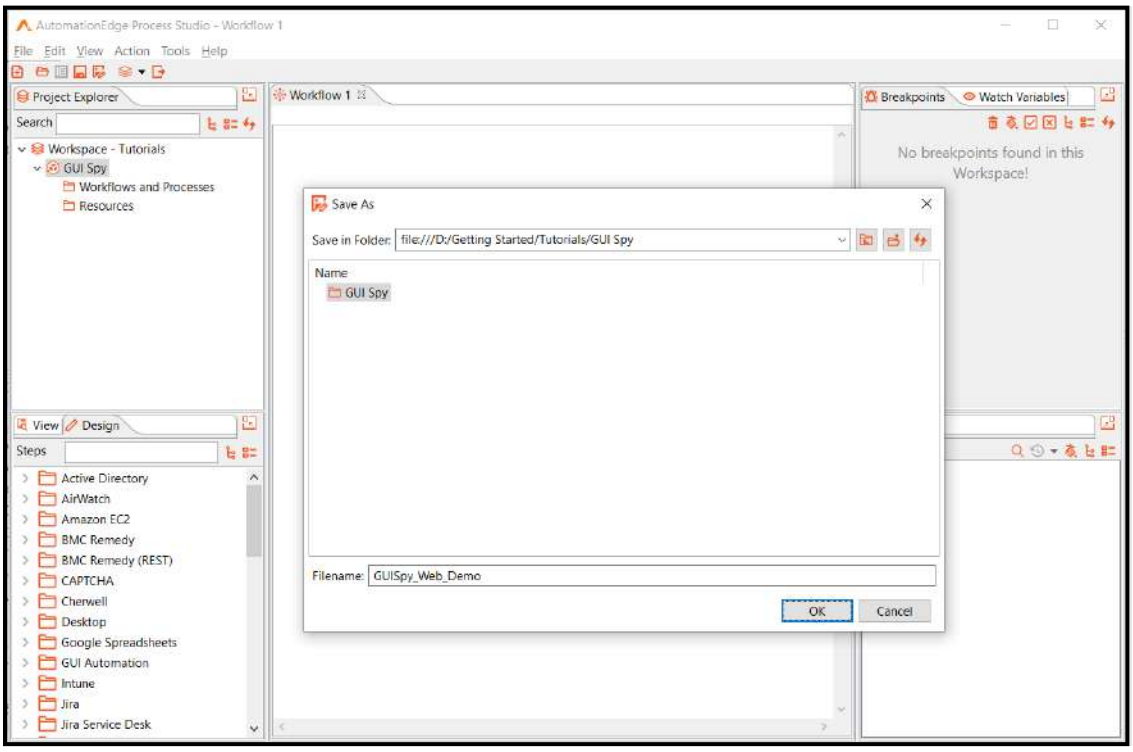
1. Right-click on Tutorials Workspace and select New Project. Create a Project named GUI Spy.
2. To launch GUI Spy, go to the Tools menu and click Spy→GUI Spy sub-menu. Click GUI Spy. Alternatively, use the shortcut `Ctrl+alt+w`



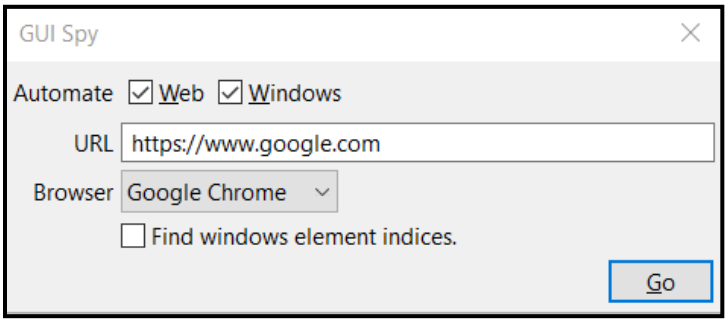

3. A dialog window with a message – 'There is no Active Workflow. Creating new workflow!' appears as seen below. Click OK to create a new workflow.



4. Provide a name for the new workflow, say GUISpy_Web_Demo, in this example. Click OK to create the new workflow.




5. The GUI Spy dialog appears, as seen in the snapshot below. Click Go to open the GUI Spy toolbar.



The If IE is selected as the browser, an 'Ignore IE Protected Mode Settings' checkbox is visible. Starting version 7, IE introduced the notion of 4 zones; namely Intranet, Internet, Trusted Sites and Restricted Sites. The following two scenarios are to be noted in GUI automation using IE.

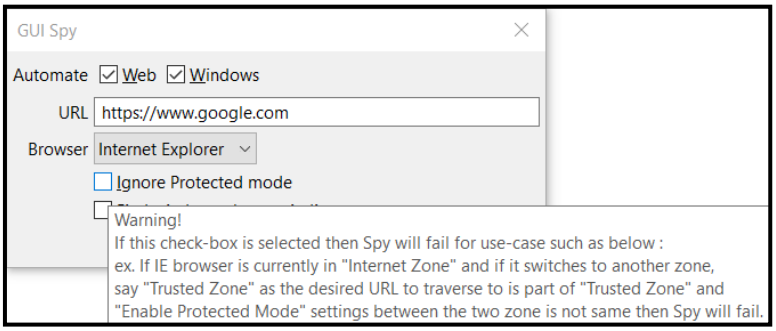
The default behaviour in GUI automation using IE is to check "Protected Mode" settings for all the zones and throw an error when this setting is different for any two zones.

'Ignore IE Protected Mode Settings' checkbox in GUI-Spy can handle the scenario.

When the "Ignore Protected Mode" setting is CHECKED in GUI-Spy, the "Protected Mode" setting check for different zones is ignored, and it starts without error no matter what "Protected Mode" settings for different zones are.

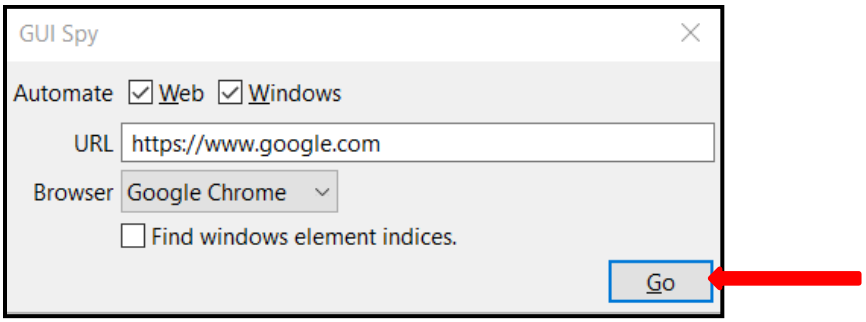
**Note**

**Note:** GUI automation workflow will fail when it traverses to another zone with a different 'Enable Protected Mode' setting.

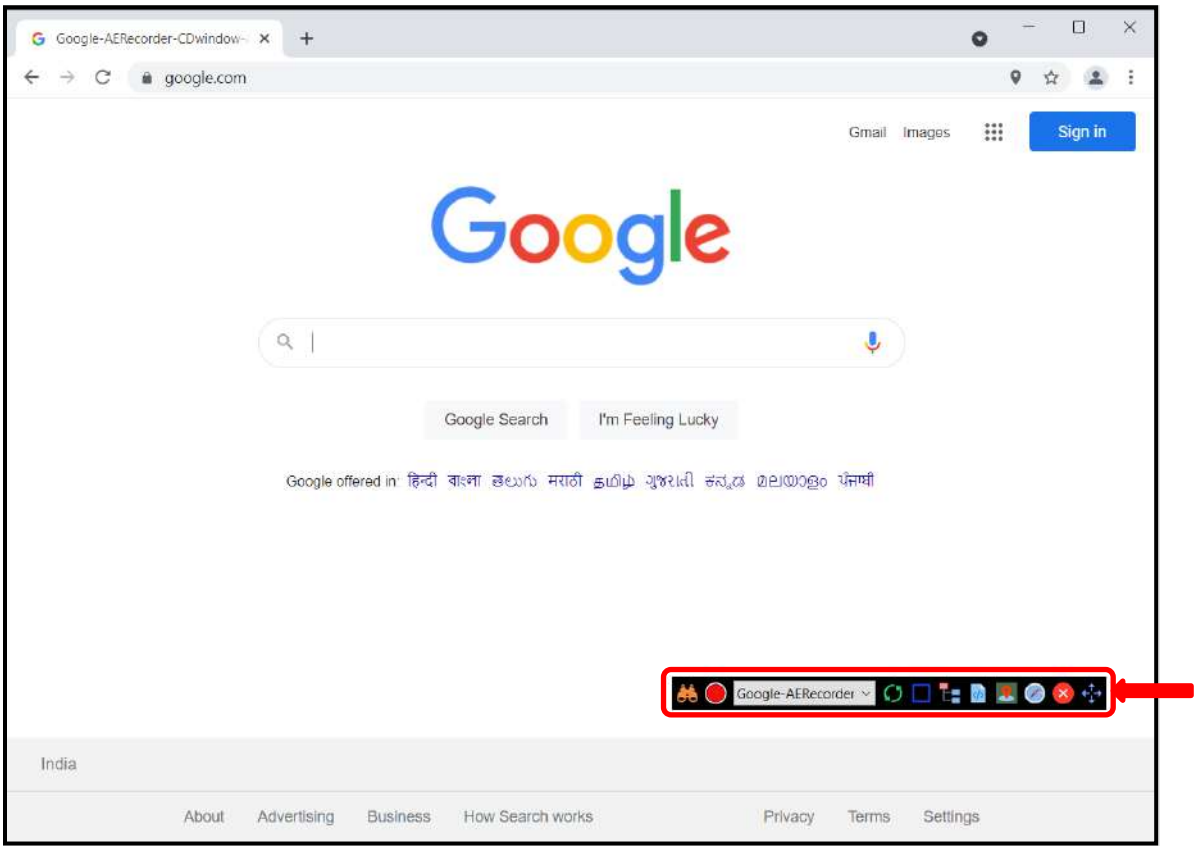



6. Enter the web page's address you to inspect in the text field corresponding to the URL label. URL should be a complete URL starting with http/https. Select the browser name from the dropdown list – Firefox, Chrome or IE (Internet Explorer). In this case, the browser selected is Google Chrome and the URL entered is https://www.google.com.

7. Click on the “Go” button and wait for the browser to launch. A blue line surrounds the Go button until the browser completely initialises.




8. GUI Spy toolbar opens in the foreground to the web page.
In the screenshot below GUI Spy toolbar is highlighted in a red box.



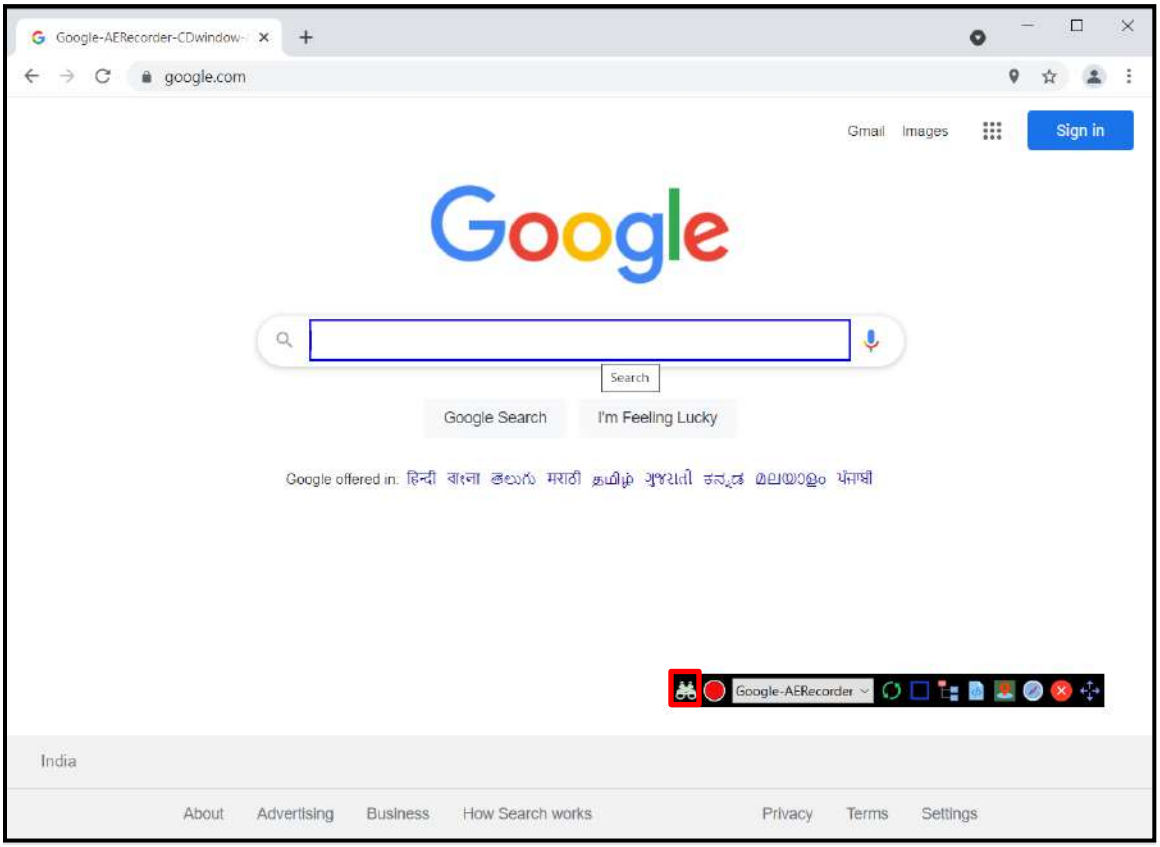
### 2.5.2 GUI Spy toolbar components

The following sections discuss the ten icons and the windows dropdown on GUI Spy on Web + Windows toolbar ( 

) and how to use them:

#### 2.5.2.1 GUI Spy: Spy (  )

1. Click the first icon Spy on the GUI Spy toolbar. It turns the icon from orange to white (  ). Hover over any element on the web page, and it shows a blue box over the element. In this case, the blue box is over the Search text box.
2. Hover over elements in the window to spy. If it does not highlight web elements on hover, check if you have clicked on the Spy icon (Spy button turns from orange to white) and then hover the mouse pointer to highlight elements.




3. Press the backtick (‘) icon on your keyboard.
4. If everything is as expected and you hit the **backtick (`),** GUI Spy captures element information and opens a Step generation dialog. This dialog contains the element’s attribute table and step generation form to generate any step from GUI Spy. If you hit ‘`’

again without closing Generate step dialog, then the attributes table gets populated with information of the current element captured by GUI Spy.

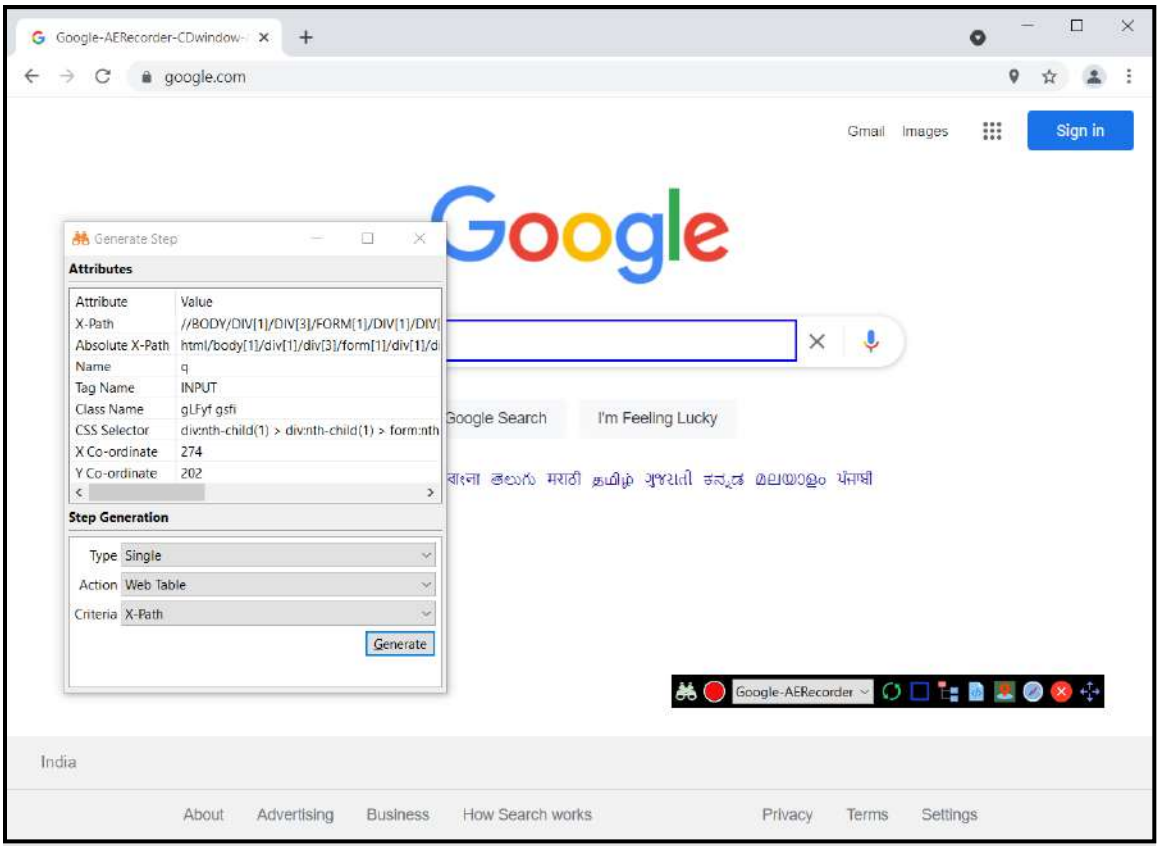
5. Select the type of step from the Type drop-down list. The values in the list are Single or Composite.
6. Select a value from the Action list.
7. The element Attributes window is displayed. Select a desired Attribute from the table or choose the desired criteria from the Criteria drop-down list.

---

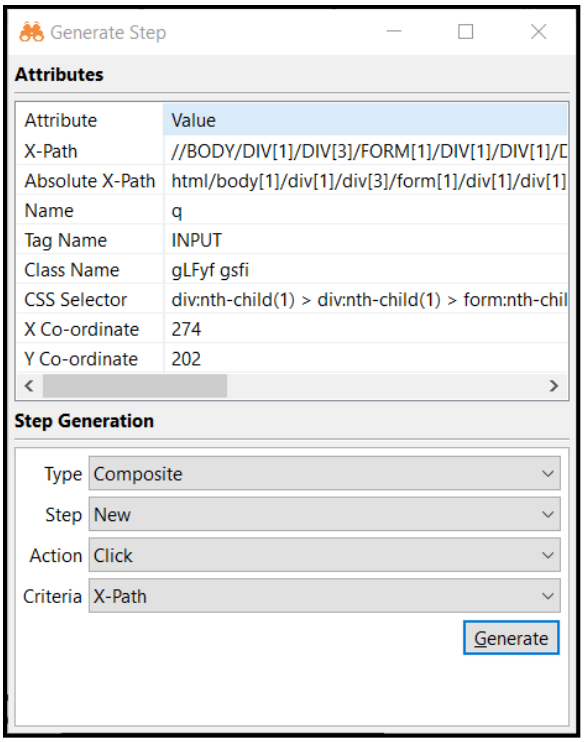
**Note:** Absolute X-Path cannot be used to Generate a step in Process Studio as It is not available in the Criteria drop-down list. GUI Spyt Generates the step with X-Path. If you need to Generate a step with Absolute X-Path, you need to first Generate it with X-Path. Copy the Absolute X-Path in the Attributes section, with a right-click and select Copy. Now replace the value of the Generated X-Path with the Absolute X-Path in the step in Process Studio.

---

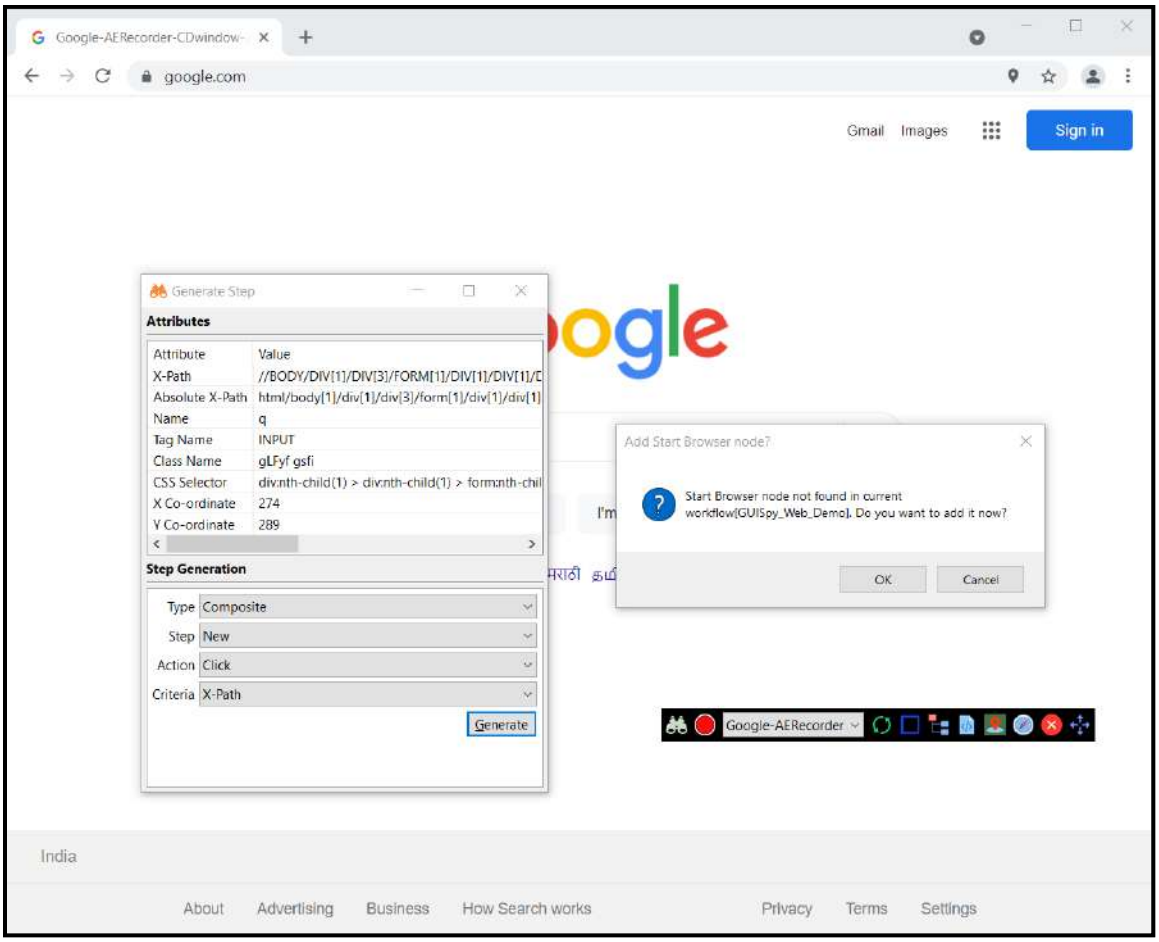
8. Click Generate button to generate a Single or Composite step with Action and Criteria chosen.



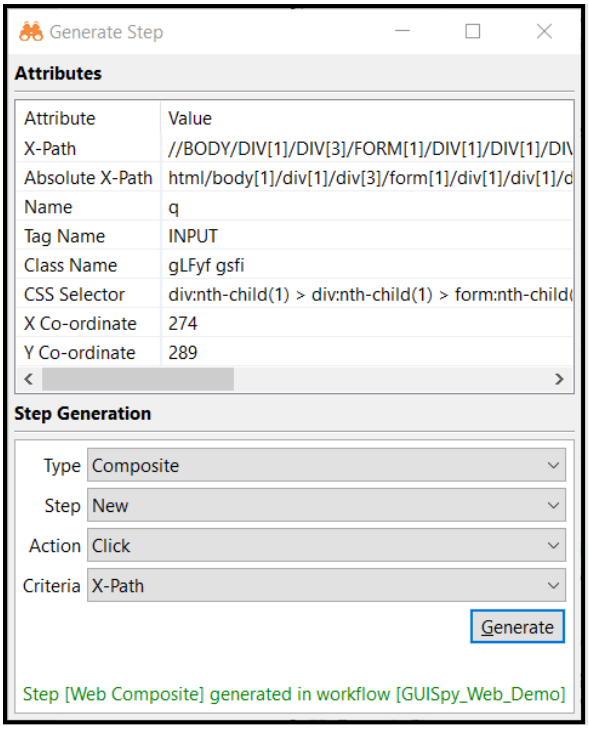
9. The screenshot below shows the Type is Composite is selected to Generate a step. Click Generate.



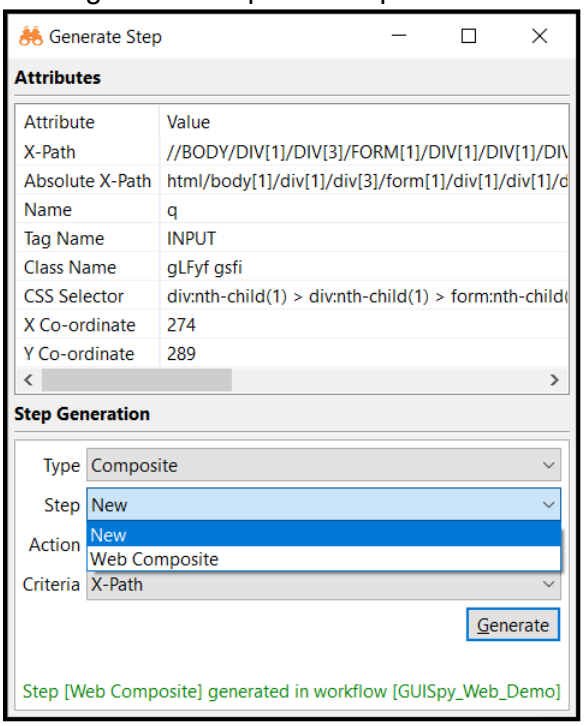
10. In case the workflow does not have a Start Browser step, a warning dialog appears. Click OK to add a Start Browser step to the workflow.



11. The message ‘Web Composite step generated in the workflow’ displays at the bottom.




12. After a Composite step Generation for the Click Action, create a New step or add to the existing ‘Web Composite’ step as seen in the Step drop-down list in the snapshot below.



| | |
|---|---|
| **Note** | If multiple windows open up with the current spy URL, spy on an element on the windows or elements contained in frames; GUI Spy automatically handles the switching of windows or frames. |

### 2.5.2.2 GUI Spy: Record ( )

GUI Spy: Record is a toggle button to start and stop recording activity.

1. Click the Start Recording button to start recording.


2. Click the Stop Recording button to stop recording.


3. Once you start the recording, it understands the mouse and the key actions.
4. It considers every element as windows by default until there is a preference for detecting web application while recording.

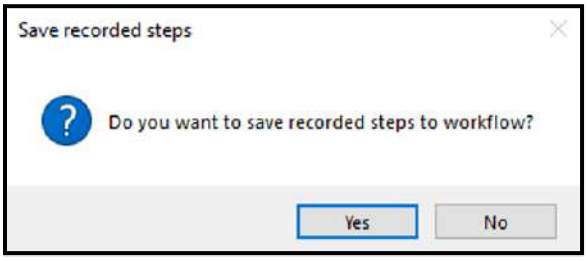
5. The GUI Recorder records the following actions.

| No. | Record Actions | Description |
|---|---|---|
| i | Click | The recorder automatically detects click type on an actionable elements. |
| ii | Set/Write a value | Recorder generates the set value step if a user,<br>• Types in a text field or<br>• Performs keyboard action ctrl +v on the text field |
| iii | Read/Get value | To read the value of a particular element, hover the mouse on that element and press Ctrl + C. |

6. Like GUI spy, the recorder handles complex window/frame tree switching activities internally and generates appropriate steps during workflow creation.

| Note | In some cases, if GUI Recorder fails to highlight/record a few elements, use GUI Spy to generate steps for such element actions. |
|---|---|

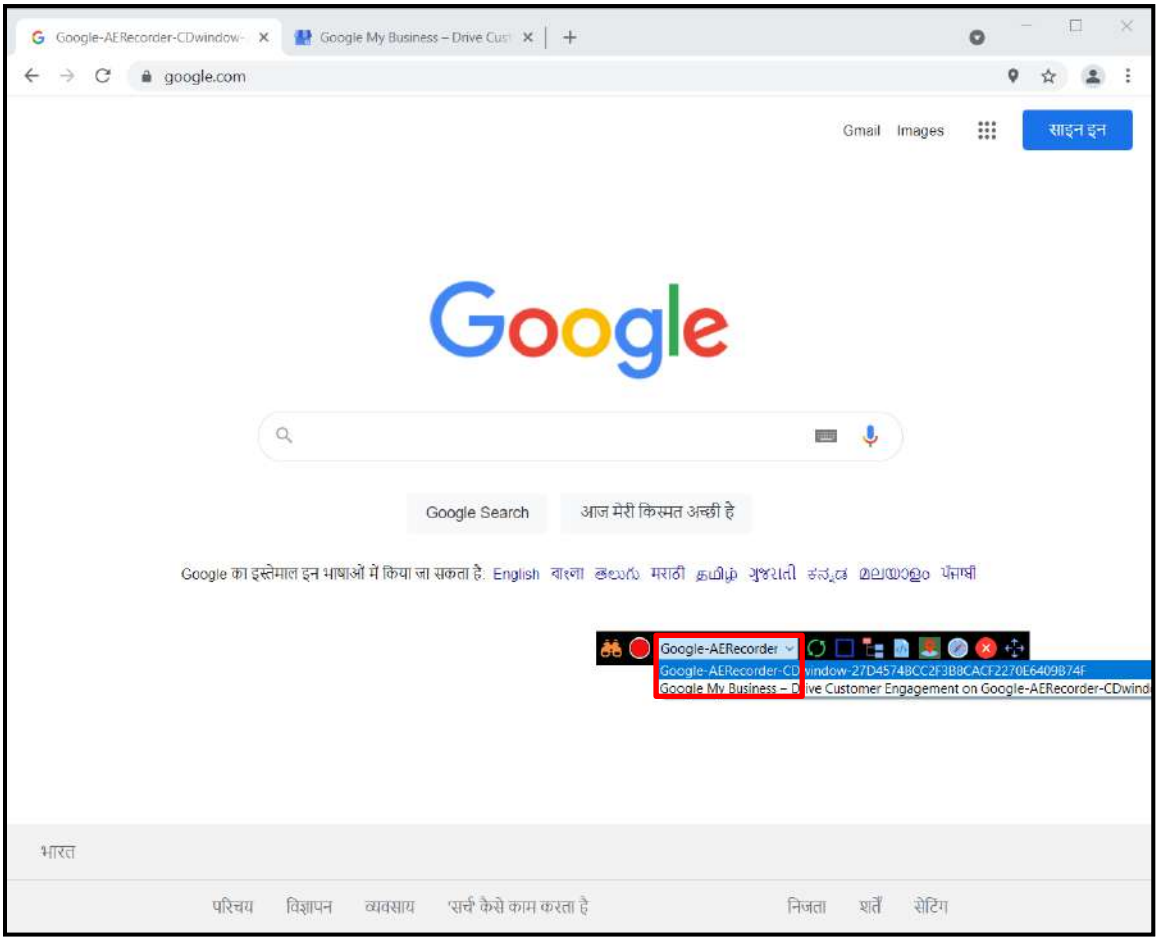
7. Once you click on stop recording, the recorder prompts the create workflow dialog. Confirm by clicking Yes to create a workflow with recorded steps; else, click No.



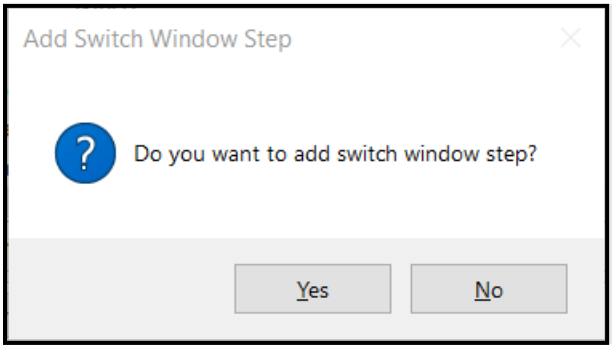

### 2.5.2.3 GUI Spy: Web page Windows (Google-AERecorder)

1. If multiple windows open up with the current URL, you may select the desired window using the window combo box in the GUI Spy toolbar, as seen in the screenshot below. For example, in the snapshot below, we can see two windows in the drop-down list next GUI Spy icon. Select any window to highlight it and bring it to the forefront.

2. We can now spy web elements on that window.




3. If you switch to another window from the list,
   - The Web + Windows mode asks if you want to add a Web Switch Window step to the workflow, whether you switch windows through the GUI Spy list or otherwise.
   - The Web mode asks if you want to add a Web Switch window only if you select a window from the GUI Spy list; else, manually add the step to the workflow.



#### 2.5.2.4 GUI Spy: Refresh ( )

The Refresh icon reloads the web page.

13. Click the Refresh icon to reload the page. Reload also refreshes the list of windows in the drop-down.

#### 2.5.2.5 GUI Spy: Start Highlighting ( )

If, for some reason, GUI Spy does not highlight elements, click on the Start Highlighting icon to resume.

1. On a browser (Specifically IE) when recording/spying is in progress, if the page is changed by some actions the highlighting stops.
2. Click highlight icon on toolbar to start highlighting the elements again and continue spying/recording.

#### 2.5.2.6 GUI Spy Frames ( )

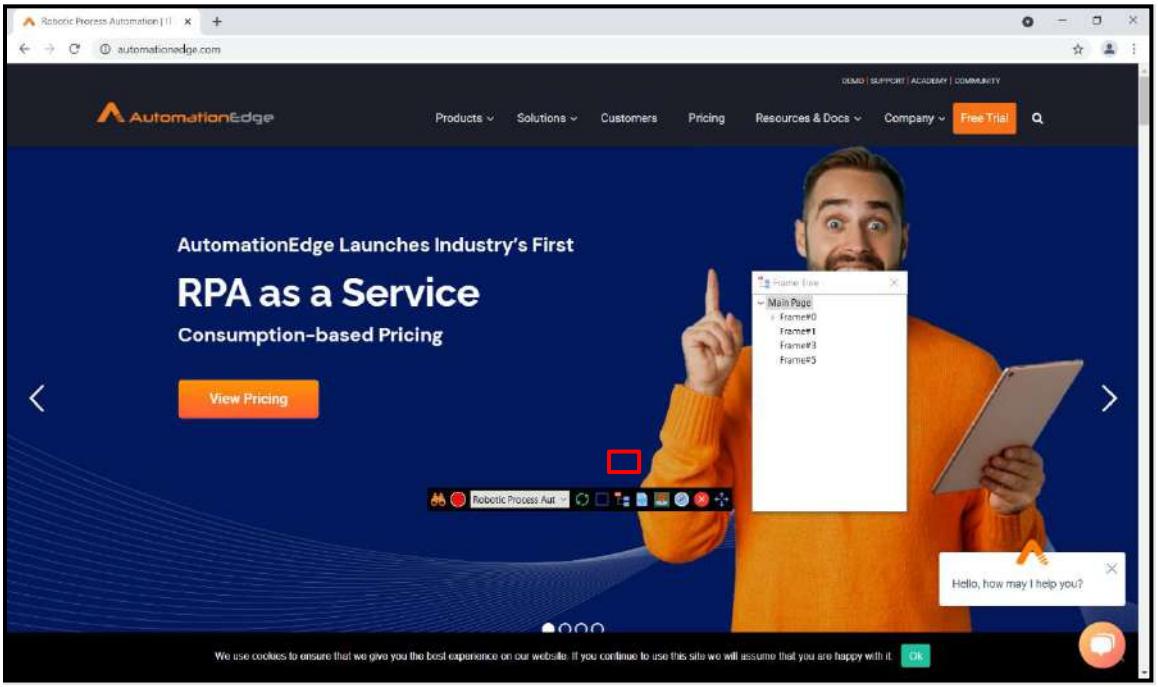
**Important:**

Frame switching is automatic. Frame tree button should only be used in case of spy is not able to extract/highlight elements from particular frame.

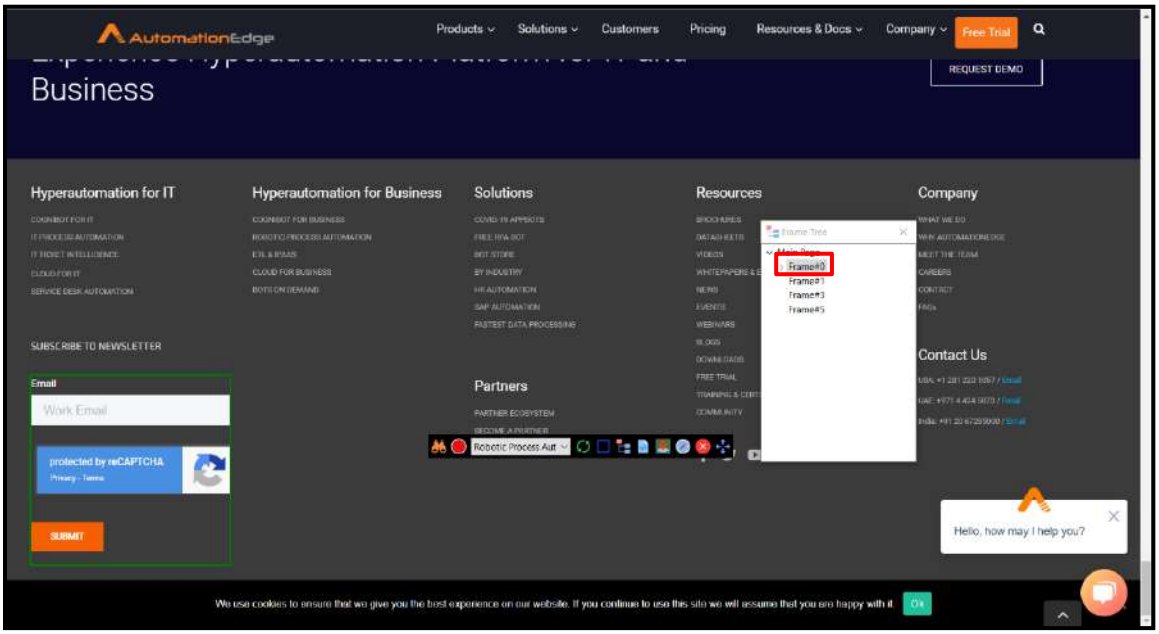
In case you need to use Frames follow the steps below,

1. If you want to spy on element which is contained in a frame, then you need to switch to that frame. To do that you need to click on the frame icon on GUI Spy toolbar. It opens a dialog with frame tree where you can select any frame to which you want to switch.
2. The Frame icon is highlighted in the screen below. Click the Frame icon.
3. The Frame Tree for the current window appears as shown below.
4. Let us now GUI Spy the following URL - https://automationedge.com
5. When the page is initialized the first time and all components are loaded.
6. Click on the Frames icon ( ). The frame tree for the Current window is populated.

7. GUI Spy can detect any number of frames. If the page contains frames a Frame Tree window appears. Select any frame to which you want to switch. Click on “Main Page” to expand the tree. Click on the nodes to expand the hierarchy.



8. Select a parent frame and a child frame underneath, and then hover over any frame element. In case web elements are not highlighted on hover click Spy (Spy button turns from orange to white) and then hover the mouse pointer to highlight elements.
9. The following snapshot displays the expanded frame hierarchy. In the example below we are on Frame#0 under Main Page. The frame is highlighted in a green box as seen in the snapshot below.

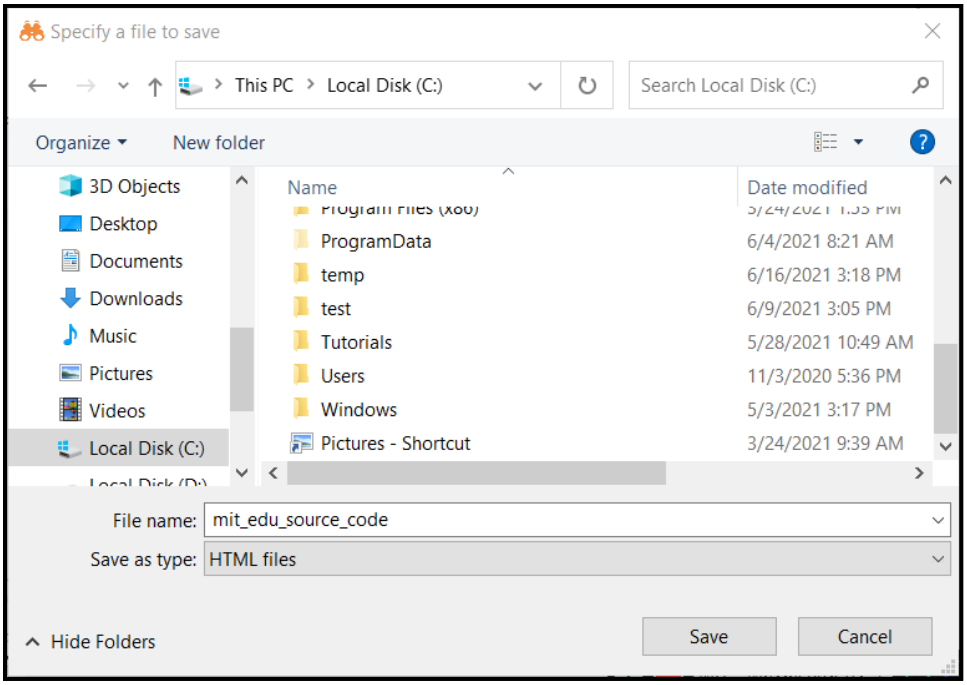



**Note** If you switch to frame, spying on that will start automatically. To be able to spy on rest of the document select **Main page** from tree; you additionally need to manually add the Web Switch Frame step with the Locate By Default option. As soon as you select another frame spying from first frame stops.

### 2.5.2.7 GUI Spy: Get Source ( )

Download the source code of the current page using the Source icon on the GUI Spy toolbar.

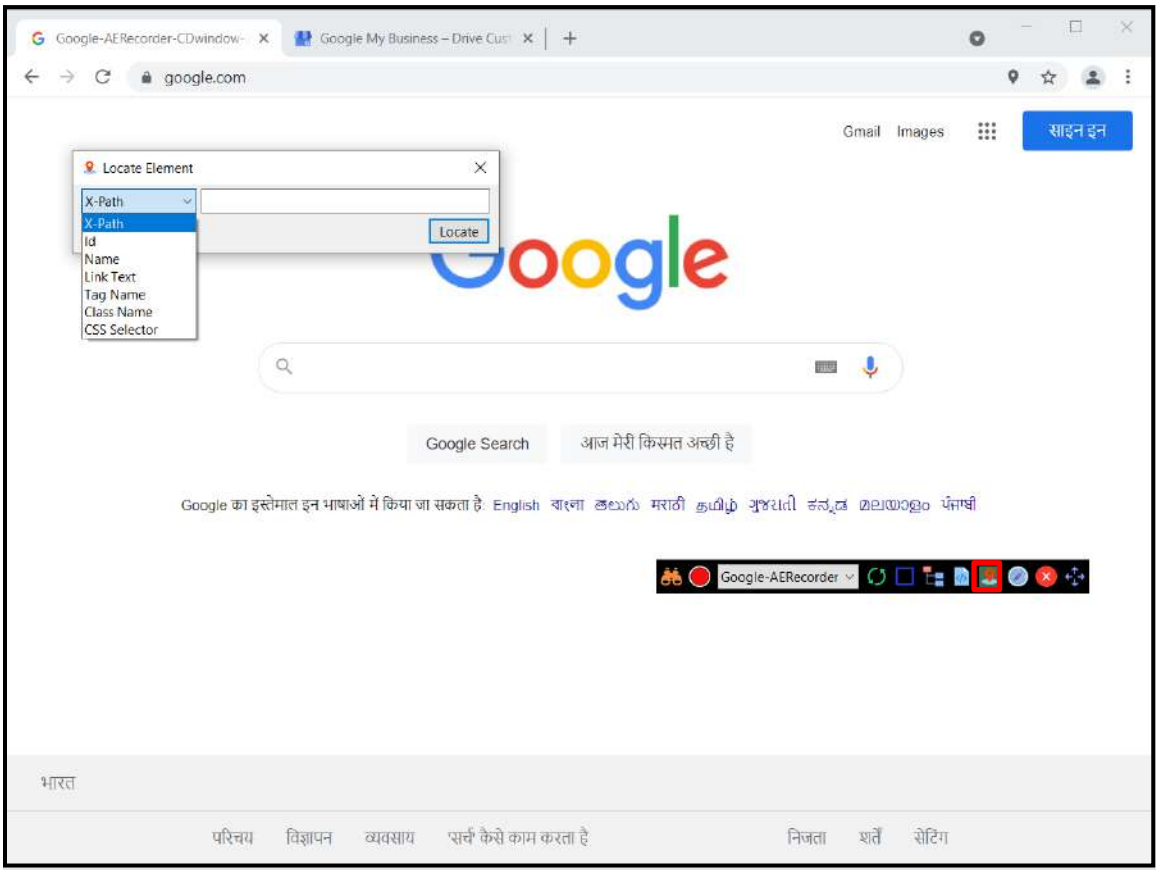
1. Click the Get Source icon on GUI Spy.
2. A Windows Explorer dialogue box to save the Source Code opens. Navigate to an appropriate directory path to save the Source Code. Specify a name for the file. Click Save.



### 2.5.2.8 GUI Spy: Locate Element ( )

Locate Element locates an element on the web page based on an element attribute value.

1. In the screenshot below, click the highlighted Locate Element on the GUI Spy toolbar.
2. The Locate Element window pops up. Select a criterion from the drop-down list, provide a known value for criteria in the text box on the right and click Locate to locate the element on the web page.




3. Clicking the highlighted element deselects it and erases the field value from the Locate Element dialog.

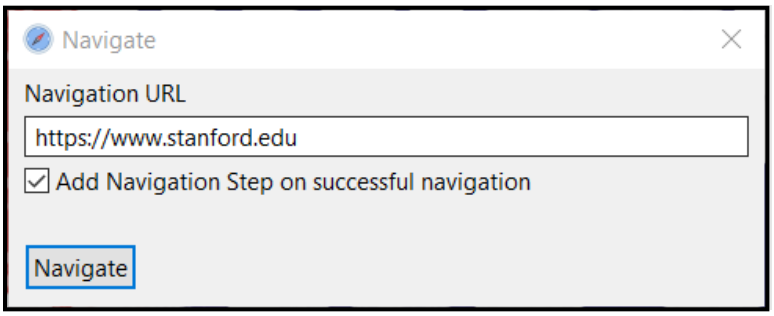
**Known Limitation:**

- If a Link Text element locator contains the “&” sign, it shows it as &amp, so correct it before locating elements.

### 2.5.2.9 GUI Spy: Navigate to URL ( )

While spying element on a web page, if there is a need to navigate to another URL, use the Navigate to URL icon.

1. Click Navigate icon to navigate to another URL and optionally create a Navigate to URL step.
   Upon clicking Navigate to URL, a Navigate dialog appears as below. Provide the navigation URL and optionally enable the 'Add Navigation Step on successful navigation' check box. Enabling the checkbox creates a Navigate to URL step in the workflow.




2. The drop-down list now contains the window name of the new URL.


### 2.5.2.10 GUI Spy: Close GUI Spy ( )

Closed GUI Spy when Spying is complete or as required,

1. Click on the Close GUI Spy icon to close GUI Spy.
2. Alternatively, click on the Move Toolbar icon. Click Ctrl +Q to close GUI Spy

### 2.5.2.11 GUI Spy: Move Toolbar ( )

Move the tollbar as required,

1. Click on the Move Toolbar icon, and while clicked, move Mouse to move the GUI Spy toolbar to an appropriate location.

| | |
|---|---|
| **Note:** | Refer to Project 8: GUI Spy for Web in AutomationEdge_ProcessStudio_Activity_Guide_R7.0.0 for a sample workflow with steps to identify elements with GUI Spy. |

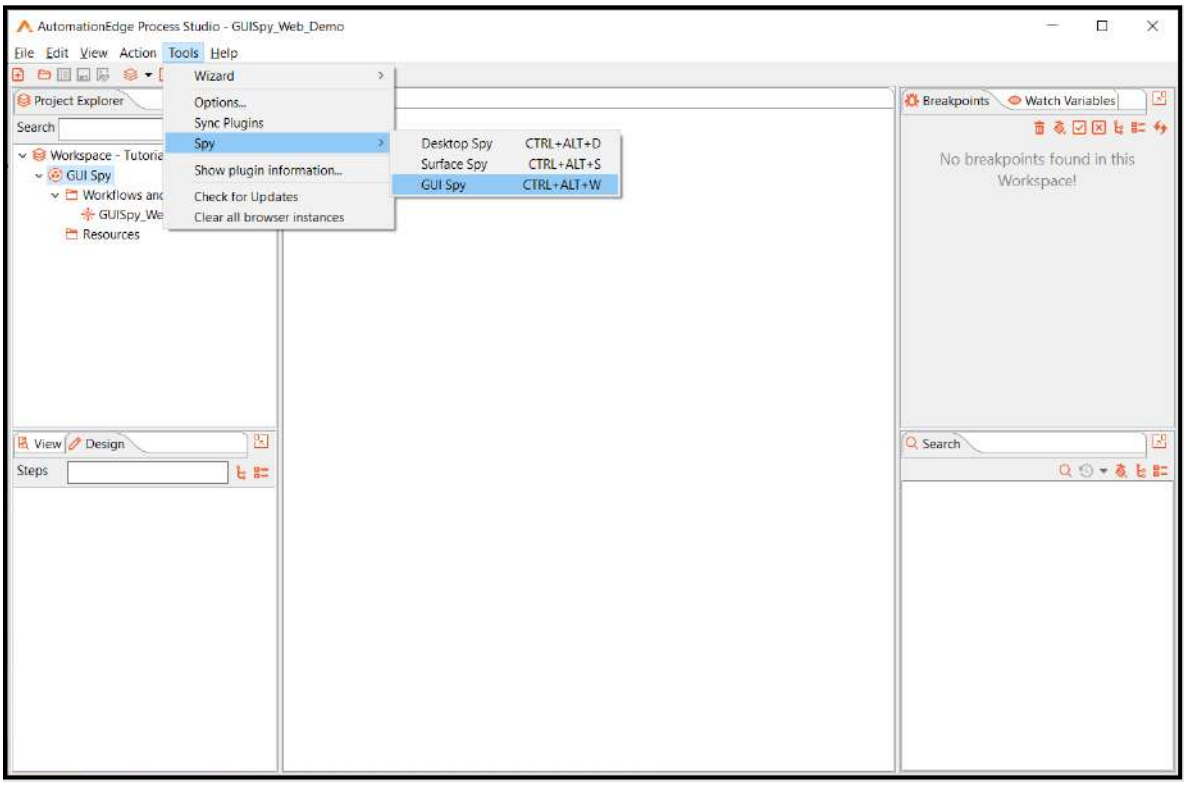
## 2.6 GUI Spy for Windows

Process Studio GUI Spy for Windows extracts Windows applications element attributes for GUI Plugins steps. GUI Spy is a compact toolbar and occupies a minimum screen area to spy on a Windows application with ease. Upon spying, it opens a Generate Step Window with two sections – Attributes table and Step Generation; change configurations if required and generate GUI plugin steps for Windows elements.

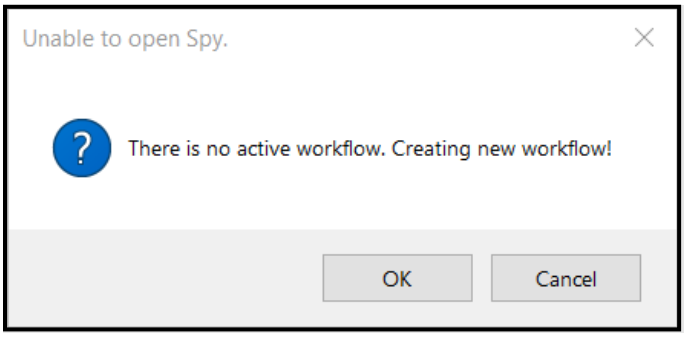
### 2.6.1 Start Using Spy for Windows

Start Windows Spy from the Process Studio toolbar or using Shortcut/Hotkeys to open GUI Spy for windows. Following are the steps,

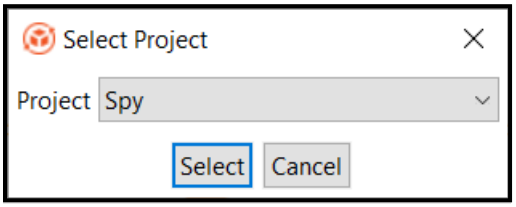
1. Add a new folder to the Spy project. Name the folder Windows Spy. Windows Spy folder is now available.
2. Navigate to Tools→Spy→Windows Spy to open Spy. Alternatively, use the shortcut `Ctrl+alt+w`



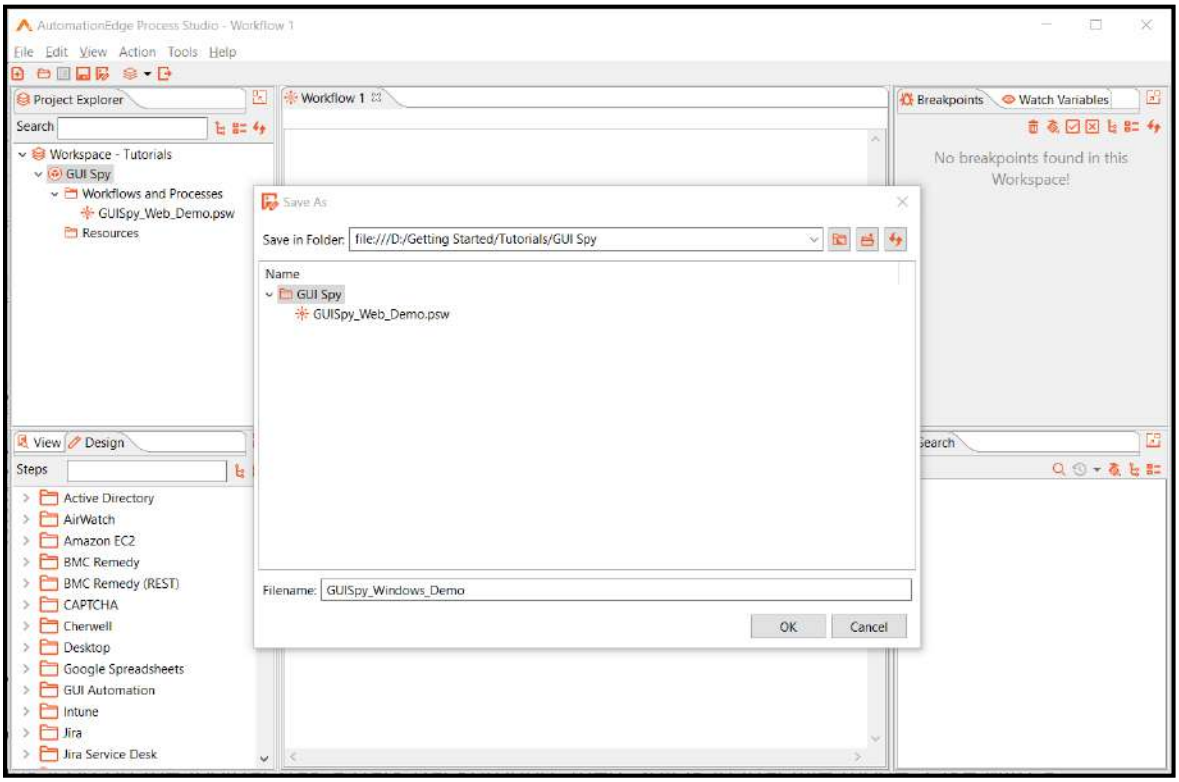
3. If there is no active workflow, respond to the warning message to create a new workflow.



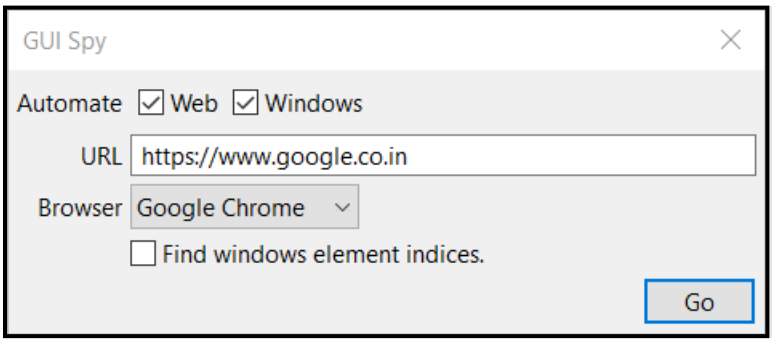

4. If there are multiple projects, select the project in which to create the workflow.



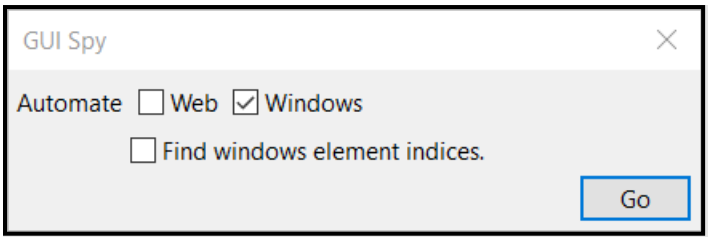

5. Provide a name for the workflow, say GUISpy_Windows_Demo.



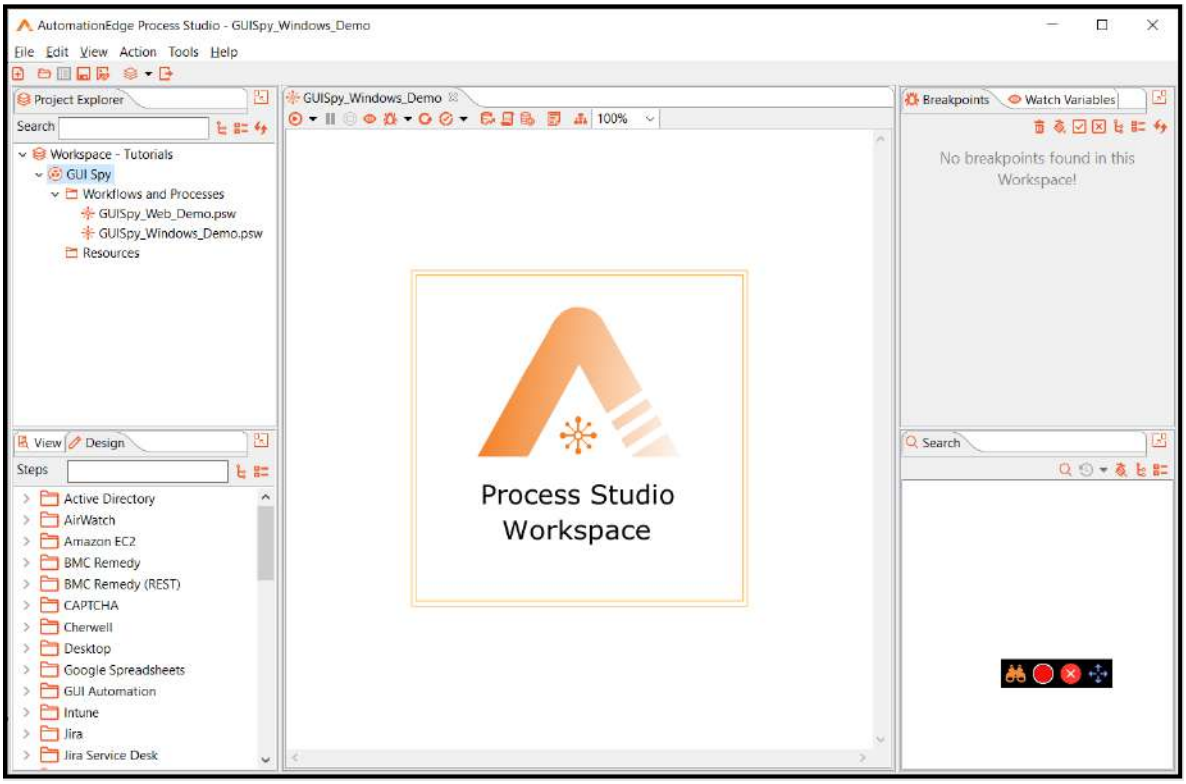
6. The GUI Spy window opens. For GUI automation for Windows, uncheck the Web checkbox.




7. The GUI Spy window displays as below.



8. GUI Spy for Windows toolbar opens, as seen at the bottom right in the screenshot below.
9. If the Spy icon on the toolbar is orange, then Spy is in inactive mode. Else with a white background, it is in the active mode.




## 2.6.2 GUI Spy for Windows toolbar components

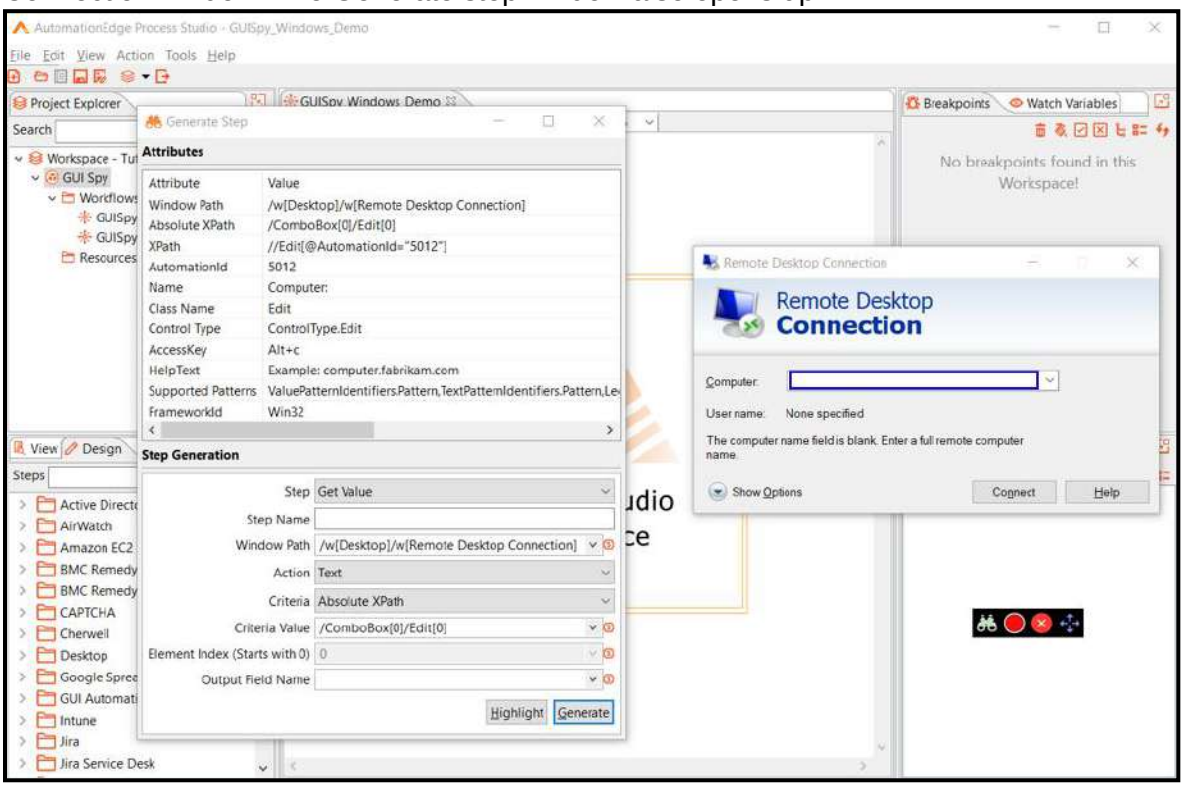
### 2.6.2.1 GUI Spy: Spy ( )

1. Windows Spy Interface opens with its icons seen below.

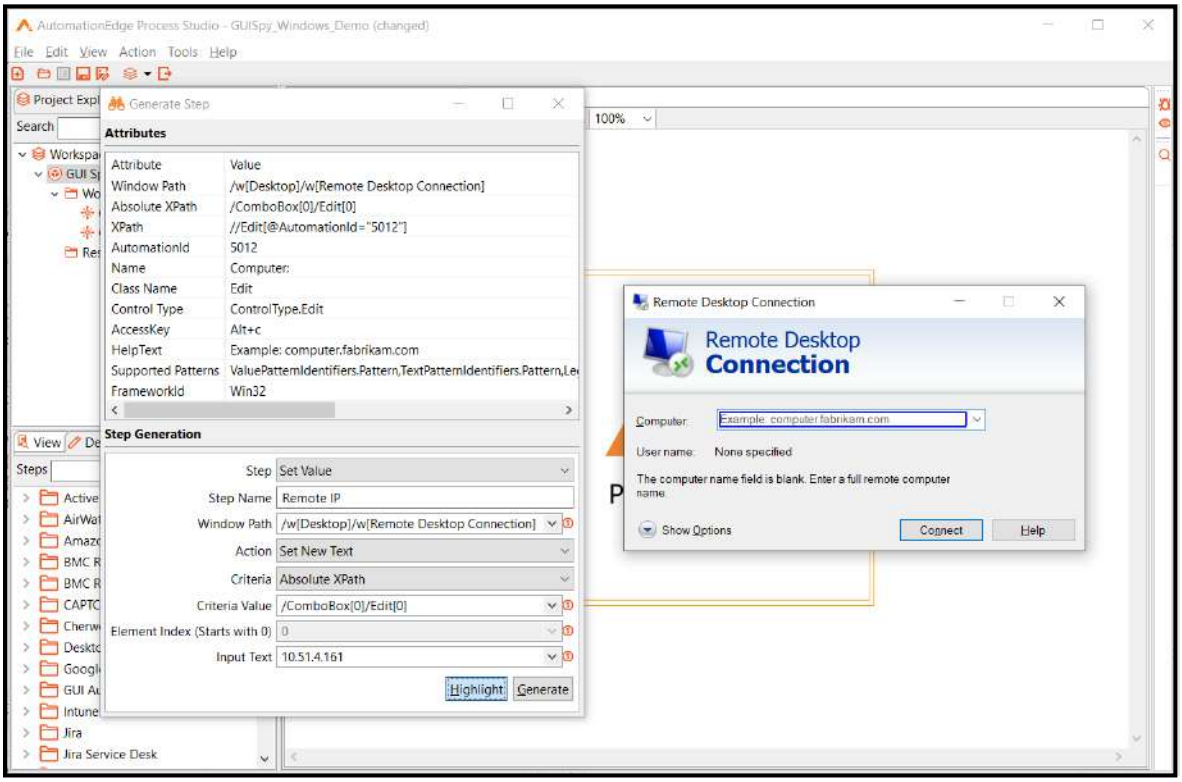
2. All possible Windows elements can be detected using Windows Spy.
3. Click the lens. The background of the lens turns white.

4. When the background of the lens is white, it is in Spy mode.
5. Now hover over the element to Spy. Press the keyboard key ‘right Shift’ or back tick (‘) to highlight it in blue color.

6. In this example, a blue box surrounds the Computer Value field on the ‘Remote Desktop Connection window. The Generate step window also opens up.



7. Change the Step and Action from the drop-down list depending on the operation to be performed. Provide a Step Name. Windows Title Regex for the application is also populated. Click the Highlight button to highlight the element as seen in blue below, and click Generate button to generate the step.




8. Pressing the ‘Generate’ Button adds a new step to workflow with the provided configurations.
9. The Step list in the window “Generate Step” above changes as per the element detected.
   It depends on the type of action it supports. e.g. let say the element detected is of type textbox. So it has a step list – Element Action, Set Value, Get Value, Mouse Action. While for the element of type checkbox step list is Element Action, Mouse Action.
10. The actions in the Action list also change for element types. Click Generate.

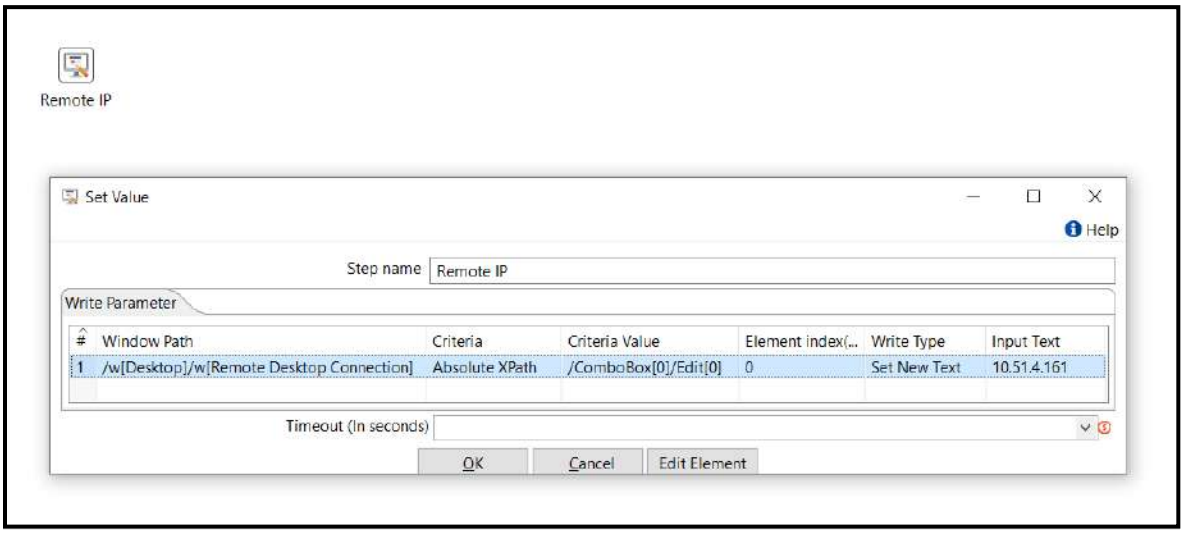
### 2.6.2.2 Windows Spy from Step

You can start GUI Spy from the step dialog in most of the GUI plugin windows steps. In this section, we will start Windows Spy from the plugin step.

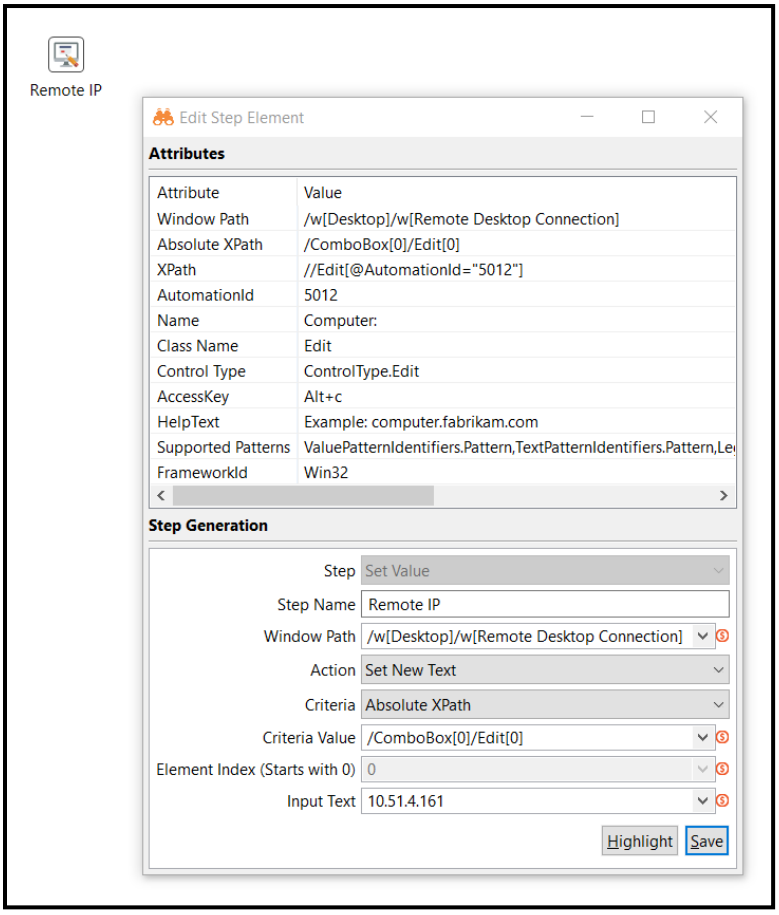
**Note** We recommend using GUI Spy from Step to edit existing windows steps rather than creating a new and similar step from the applications.

Start Spy from a Step dialog. Following are the steps to start Windows Spy from a plugin step.

1. Let us continue with the Set Value step we generated in the previous section. It sets the IP address on the 'Remote Desktop Connection' windows application.
2. Double click the step to open the configuration window, as seen below.
3. Click on any configuration item to enable the Edit Element button and click the Edit Element to start Windows Spy from step.



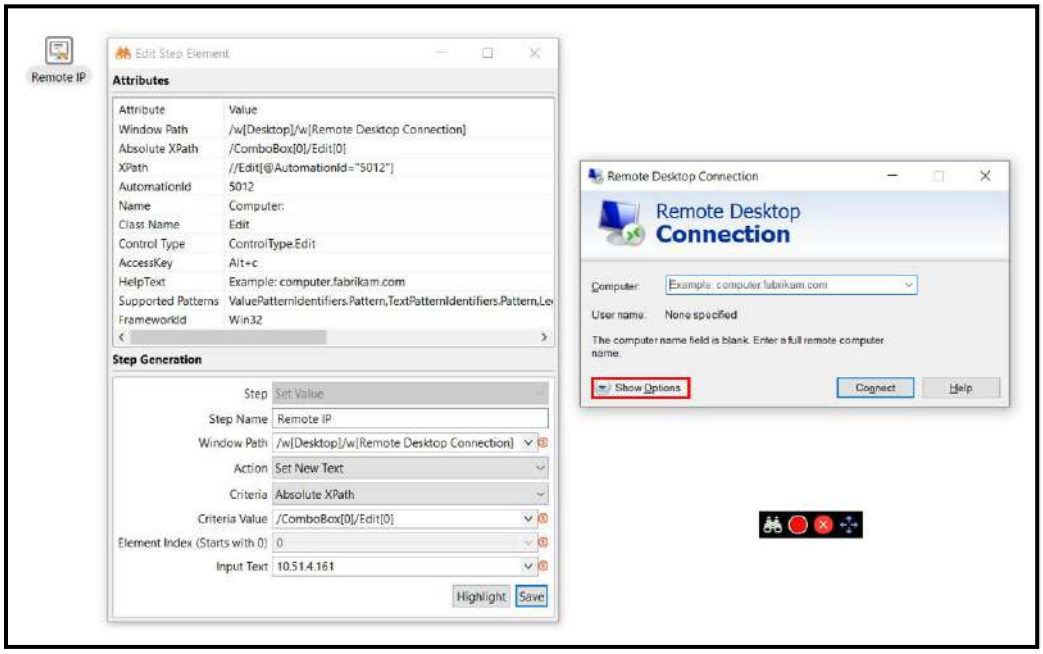
4. The Action field disabled.



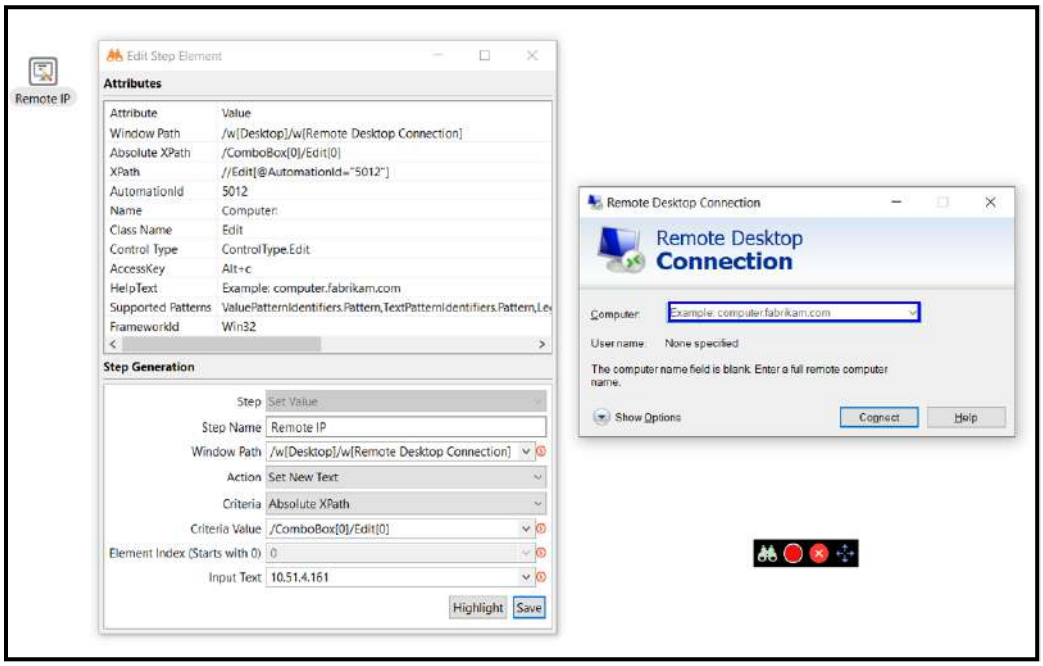

5. Opening spy with the Edit Element button from a step restricts it to the step. As seen above, the Step field is not editable, and Spy detects only relevant elements to the step. We can call it a ‘Step Restricted Spy’. For example,
    - Suppose spy starts from the ‘Set Value’ step. Now spy is restricted to elements that support Set Value in the same or other applications.
    - Suppose spy starts from the ‘window action’ step. Now spy is restricted to elements of type ‘Window’ only. It will highlight all the elements of type ‘window’ in blue color. If we try to detect elements other than type ‘Window’, it highlights it in ‘Red’ color.
6. In summary, we recommend the use of Edit Element to edit the existing step configuration it highlights all elements that support a step in blue color. If we try to detect non supported elements, it highlights them in ‘Red’ color.
7. You may also use Hotkeys to detect elements. Pressing ‘right Shift’ or back tick (`), will help you know if the element is supported for the step.

The following is a demonstration,

- Hover the Options drop-down and Press right Shift or back tick (`). The following screen shows that the drop-down is not a supported element step since it has a red outline.




- Hover over the Computer field value and press right Shift or back tick (`). The field gets a blue border immediately, and the Edit Step Element window also opens; since the Set Value step is valid for the input step.




- Notice Step dropdown is disabled and value selected is 'Set Value' since we are spying from a step.

### 2.6.2.3 GUI Spy: Record ( )

It's a toggle button to start and stop recording activity.

### 2.6.2.4 GUI Spy: Close GUI Spy ( )

Click on the Close Spy icon to close GUI Spy.

### 2.6.2.5 GUI Spy: Move Toolbar ( )

Left Click on the Move Toolbar icon and simultaneously move Mouse to move the GUI Spy toolbar to an appropriate location.

| | |
|---|---|
| **Note:** | Refer to Project 8: GUI Spy for Windows in AutomationEdge_ProcessStudio_Activity_Guide_R7.0.0 for a sample workflow with steps to identify elements with Desktop Spy. |

## 2.7 GUI Spy Limitations and Known Issues

**Limitations:**

1. GUI Spy does not yet support alerts.
2. Web Recording does not support Cross-domain applications fully.
3. Windows elements recording does not support Form fill activity using tab traversal.
4. During recording sessions, manually refreshing a page is not allowed. You can continue to record without refreshing/restarting the spy. But if you want to start from beginning then you need to restart spy.
5. Web recording can be performed only for highlighted elements.
6. Cross-domain applications and iframes are not fully supported yet in case of web recording. You can try every application and let us know if not working in some cases.
7. Form fill activity using tab traversal is not yet supported in windows elements recording.
8. During web recording sessions, an element you are acting upon should be highlighted. In other words, whatever action you are going to perform it will be performed on highlighted elements.
9. On Linux you need to manually add switch window step as there's no desktop support.
10. Some auto complete input boxes are not supported in recording but those inputs can be automated using spy.
11. If a Link Text element locator contains the "&" sign, it shows it as &amp, so correct it before locating elements.
12. Scrolling action in Windows applications is not captured by GUI Recorder.

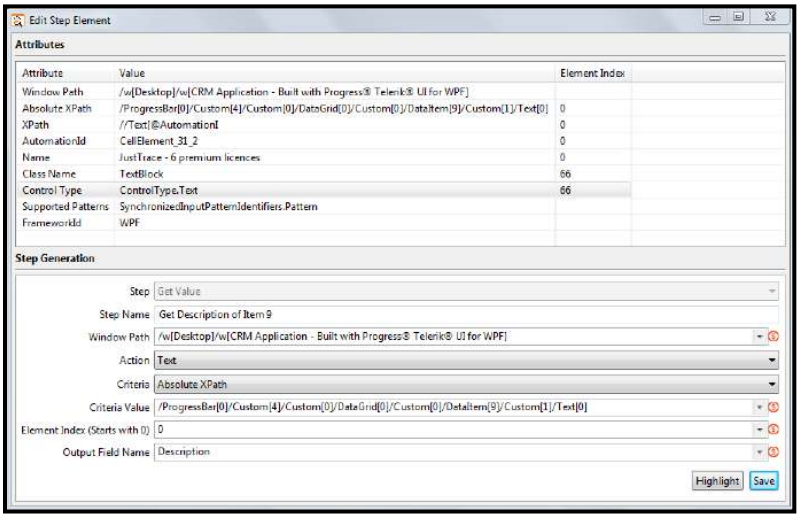
**Known Issues:**

1. Sometimes submitting a form in GUI recording for Web, using a button click creates two similar steps, and submitting a form using enter key does not create the submit click step.
2. GUI recording of Windows applications password fields generates the step but does not capture the password value.
3. In GUI Automation for Windows applications, for some elements,
    - Get Value step does not capture its value.
    - The Recorder does not record the values written or set by a user.
4. When switching to a child frame, a Switch to Frame (to Default), Switch to Frame (to parent frame) and Switch to Frame (to child frame) steps are added in the workflow. You can manually remove the Switch to Frame (to Default) and Switch to Frame (to parent frame) steps if desirable to reduce the length of the workflow.
5. Inside a frame, on a page, if we click a link that takes us to another page; on the second page, if GUI Spy does not highlight elements, toggle GUI Spy and highlight them.

## 2.8 GUI Spy for Windows Best practices

The following are best practices to select Criteria for element detection,

- Preferably detect elements with Criteria having element index 0 (i.e. unique elements). Avoid using element index greater than 0 (e.g. class name in the snapshot below). Instead, select the element having the element index as 0.

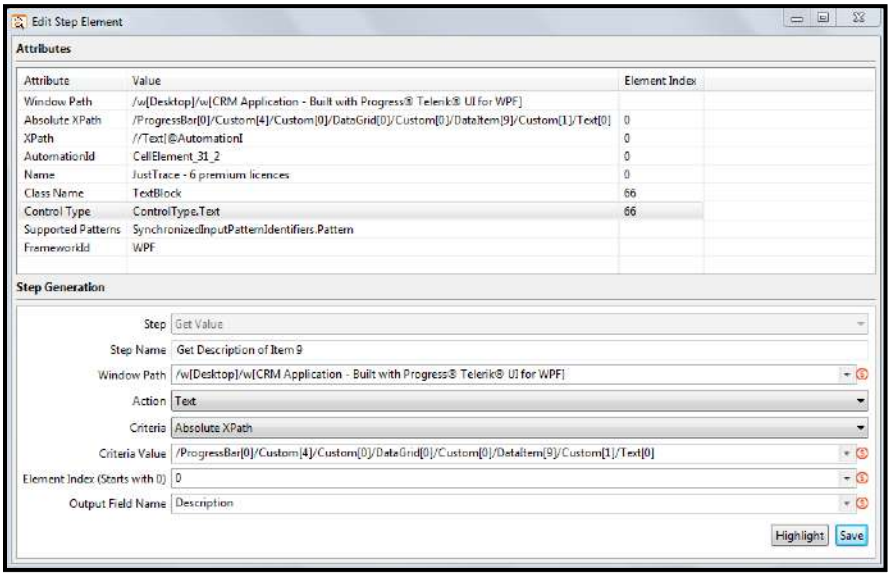



- If some Criteria are dynamic, then use Criteria for which Criteria value is constant (such as, If Absolute XPath or AutomationID are dynamic use Criteria - Name if its value remains constant). For example, absolute XPath is best for static application structure with dynamic element names.
- Use All criteria every time you have an application with no window name or if a spy doesn't provide any absolute XPath or element index.
- While generating step or editing step, with criteria All, replace any non-consistent criteria value with a blank value.

## 2.9 GUI Spy Workarounds

The following are GUI Spy workarounds,

- If the element index in the snapshot below is null/not working, then in Criteria, select All.




- While trying to detect an element, the user can take an alternative approach if it gives a red border; copy the windows path and criteria values from a different configuration of the same step, or else from another step where it detects the element with a blue border.
- Suppose there are limited Actions in the drop-down in mouse-click, then copy the windows path and criteria value and manually create the step on the element with desired Action.
- In Mouse action, if drag-drop is not working, use the Set Value step.
- We cannot use Windows Action on dialog windows. Instead, we use Mouse Action-Click, or Surface plugin– surface Click or robot handling.
- If Get Table Data is not working, use the Get Value step in a loop to fetch the table data row by row.
- Any Windows element encountered during Web + Windows mode that you cannot record or spy; then records/spy them in the Windows only mode.

# 3 Appendix 3: Desktop Spy Tutorial

## 3.1 Introduction

Desktop Plugin is helpful for .Net based Windows applications and Applet based applications. Process Studio Desktop Spy is used to extract windows element attributes.

## 3.2 Prerequisites

Following are the prerequisites for Desktop Plugin,

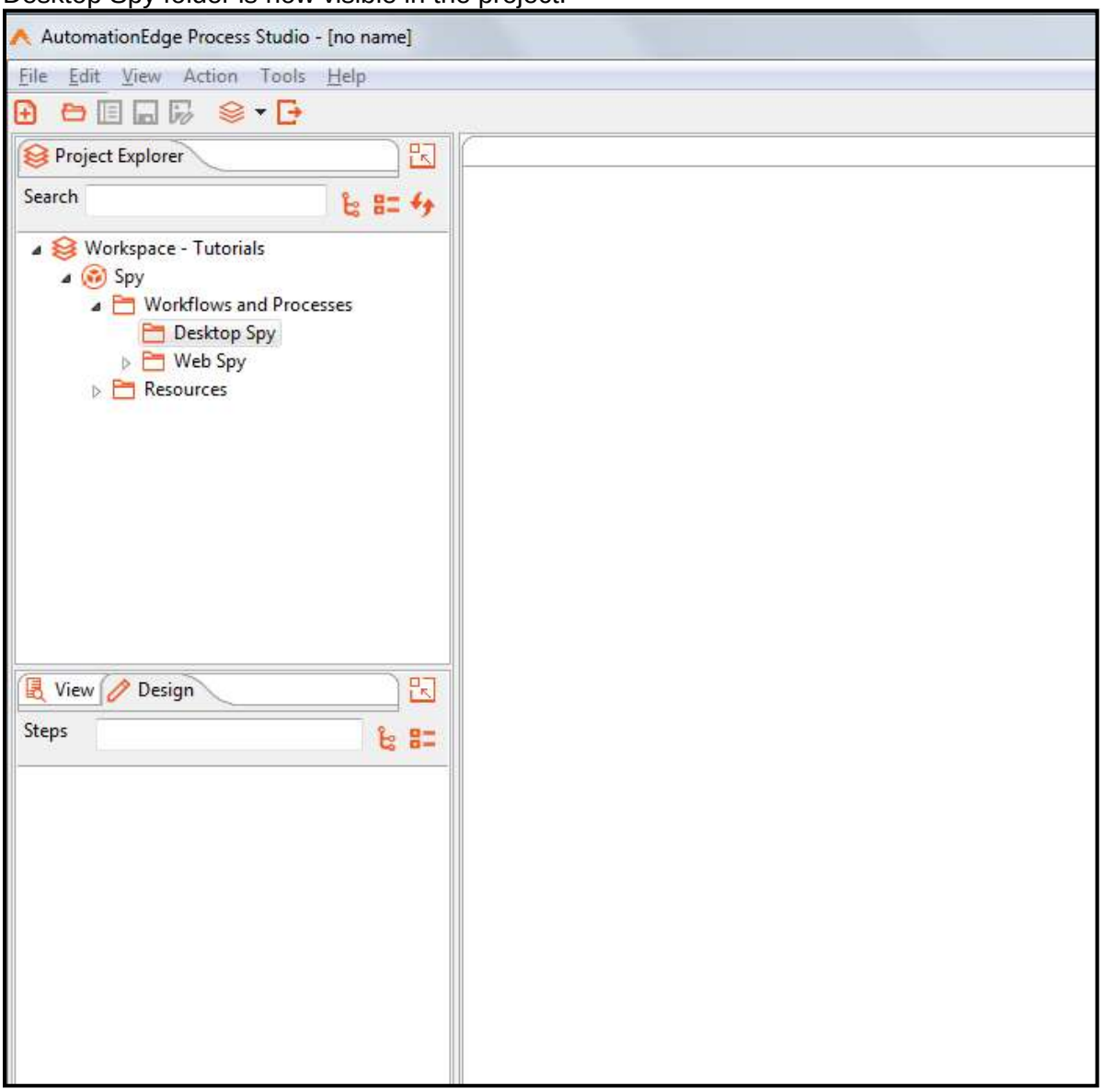
- Windows Operating System
- Install jdk and set JAVA_HOME, JRE_HOME environment variables.
- Windows .NET framework 4.6 and above
- Process Studio full pathname should not have any spaces.
- Chrome browser is required to run Desktop Spy for AE releases prior to R5.3.0.
- License is required to run Desktop Spy and Desktop plugins. The license can be acquired and applied by contacting AutomationEdge Inc. (Refer to the Desktop plugin documentation above for how to apply the License).

## 3.3 Setup

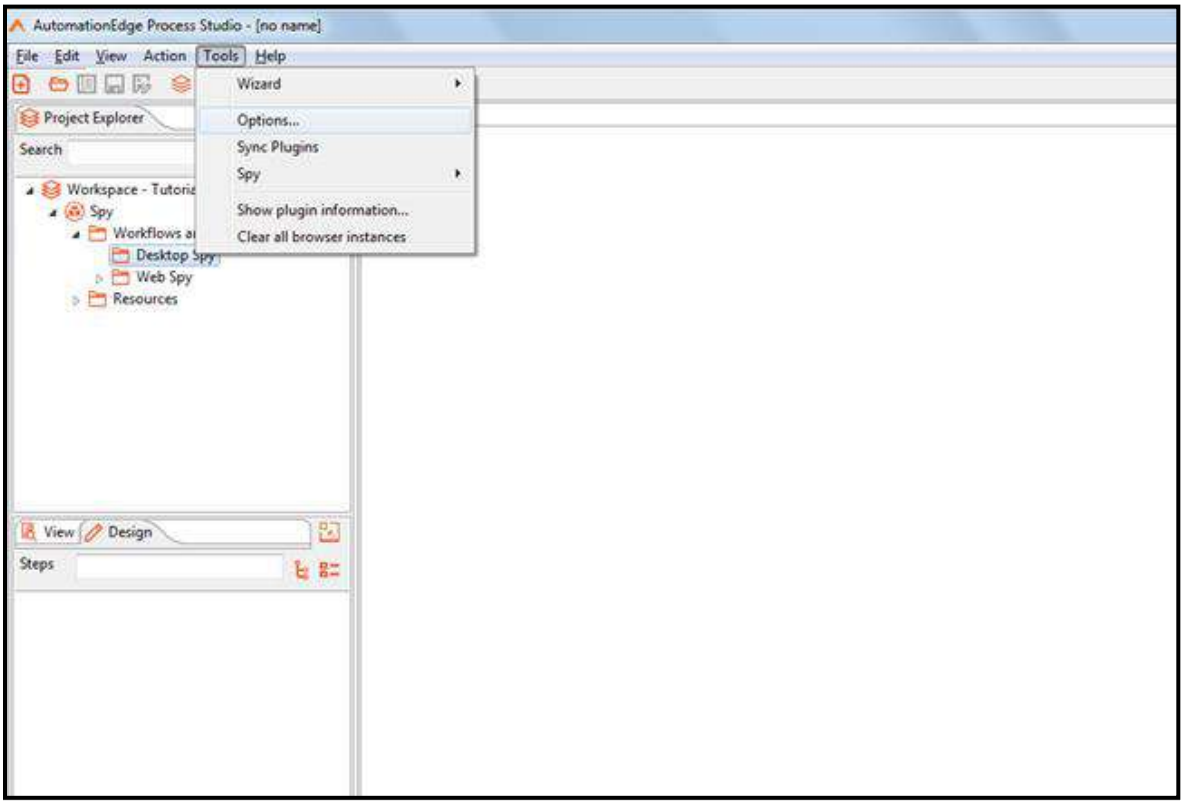
Desktop Spy fetches the configuration parameters required in Desktop plugin steps.

Follow the steps below to set up and use Desktop Spy.

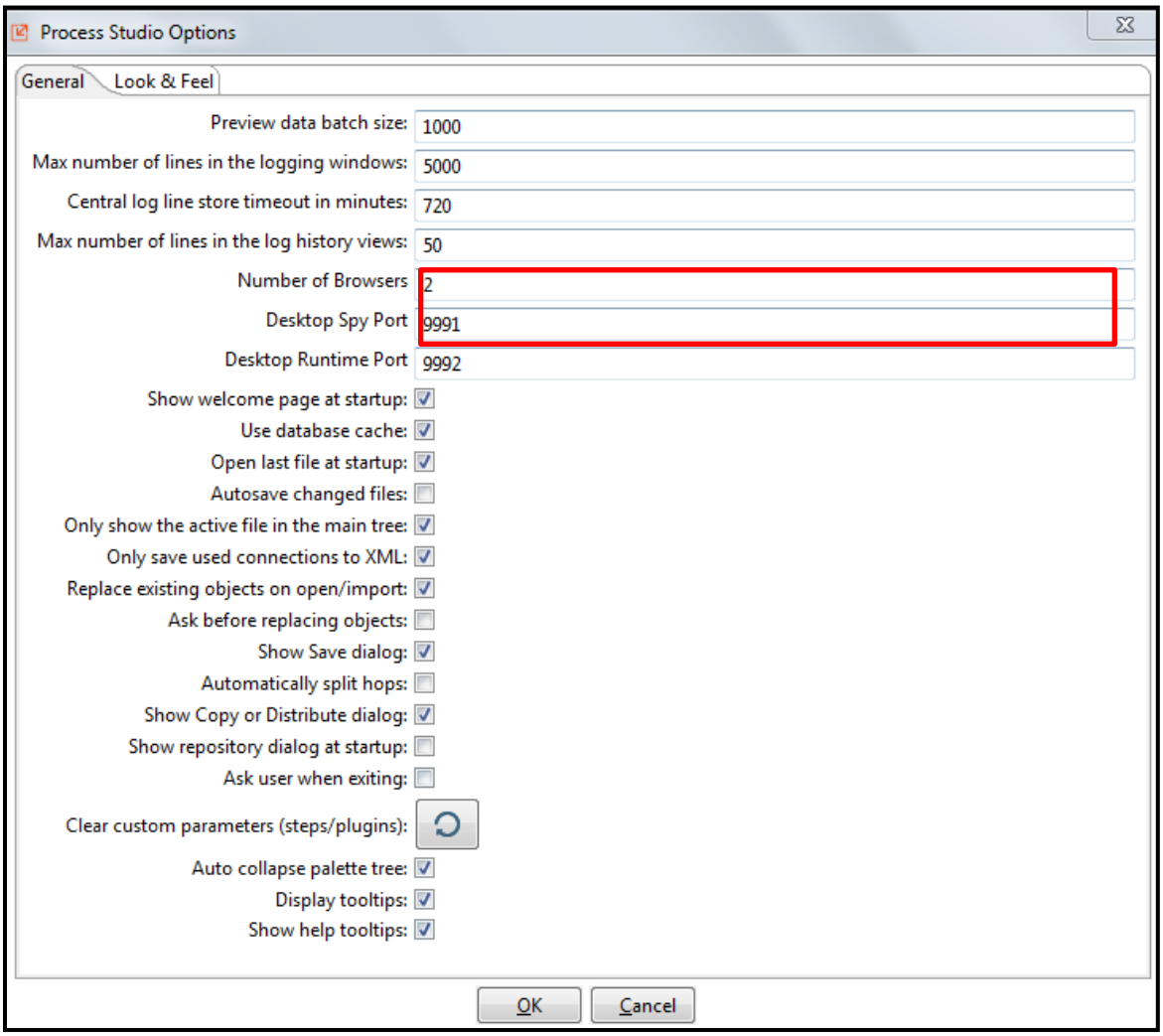
1. Create a new folder in Spy Project. Name the folder Desktop Spy.
2. Desktop Spy folder is now visible in the project.



1. To use Desktop Spy, configure Process Studio Options Desktop Spy Port and Desktop Runtime Port as seen in the snapshots below.
2. In Process Studio, go to the Tools menu and click on Options...



3. Set or change Desktop Spy Port and Desktop Runtime Port if required or leave it to the default values.

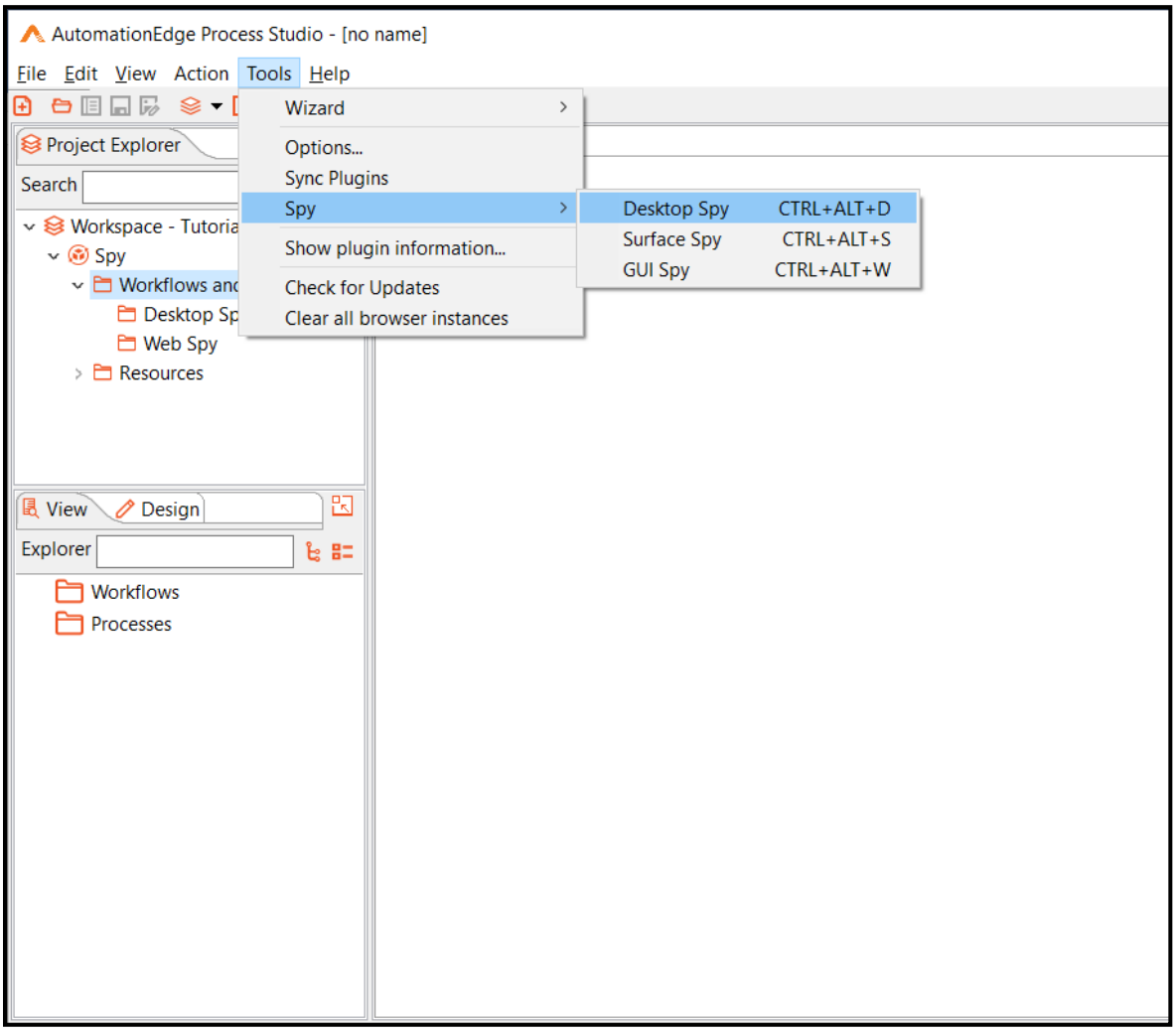



4. You are now ready to start using Desktop Spy.

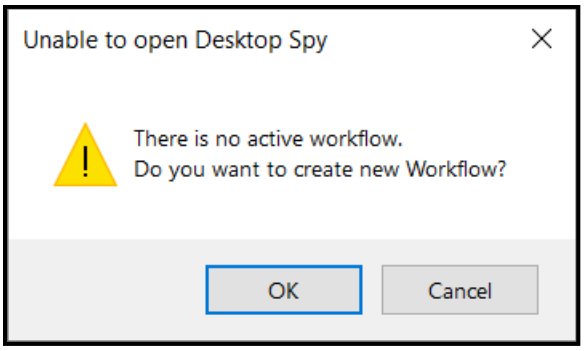
## 3.4 Start using Desktop Spy

Follow the steps below to get started with Desktop Spy.

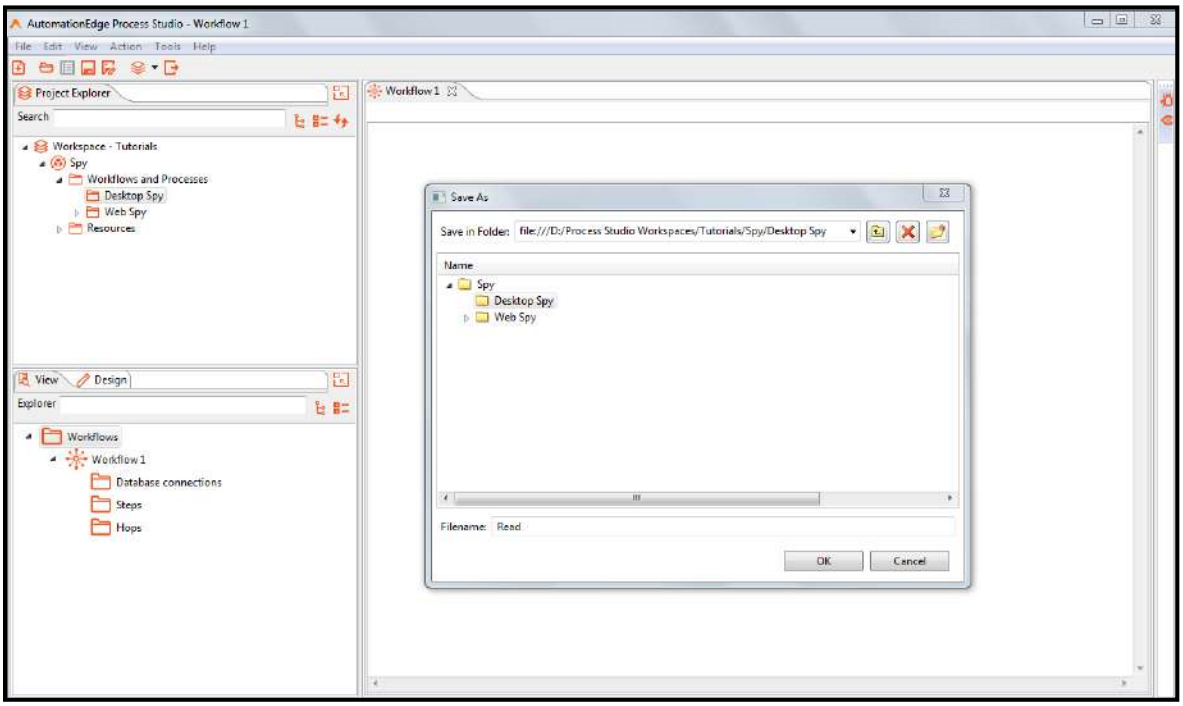
1. To launch GUI Spy, go to the Tools menu and click Spy→Desktop Spy sub-menu.



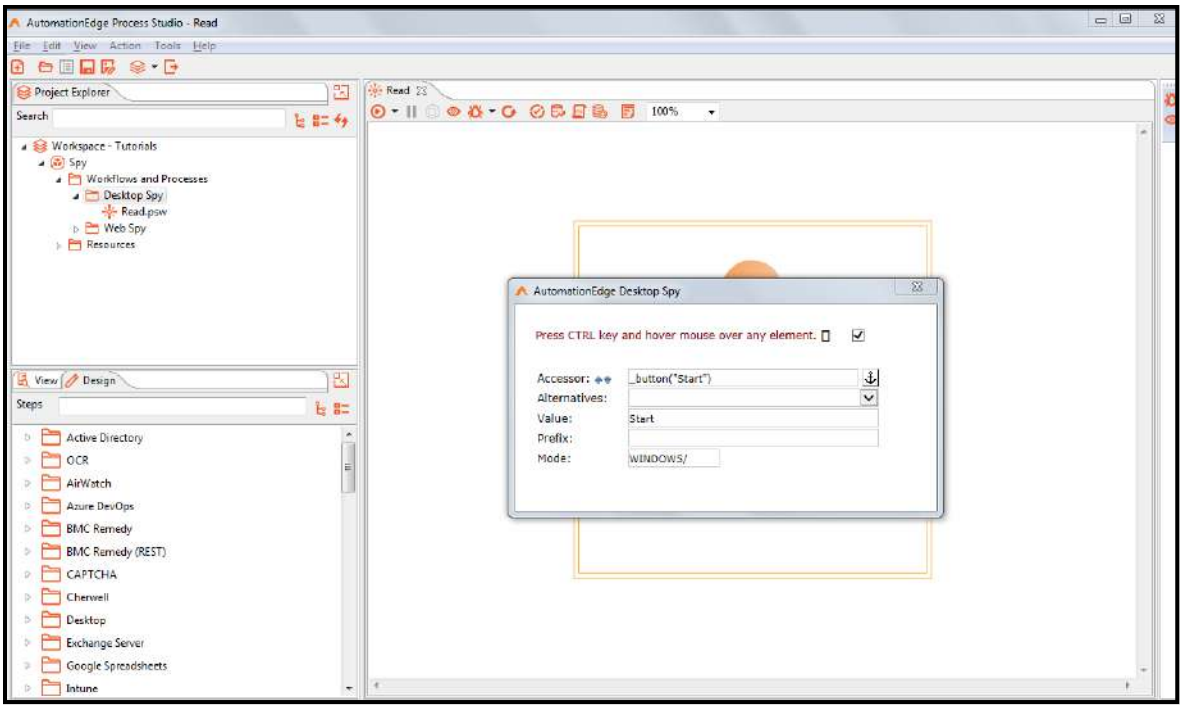
2. If there is no active workflow, a warning dialog is shown. Click OK to create a new Workflow.




3. Name the workflow Read.

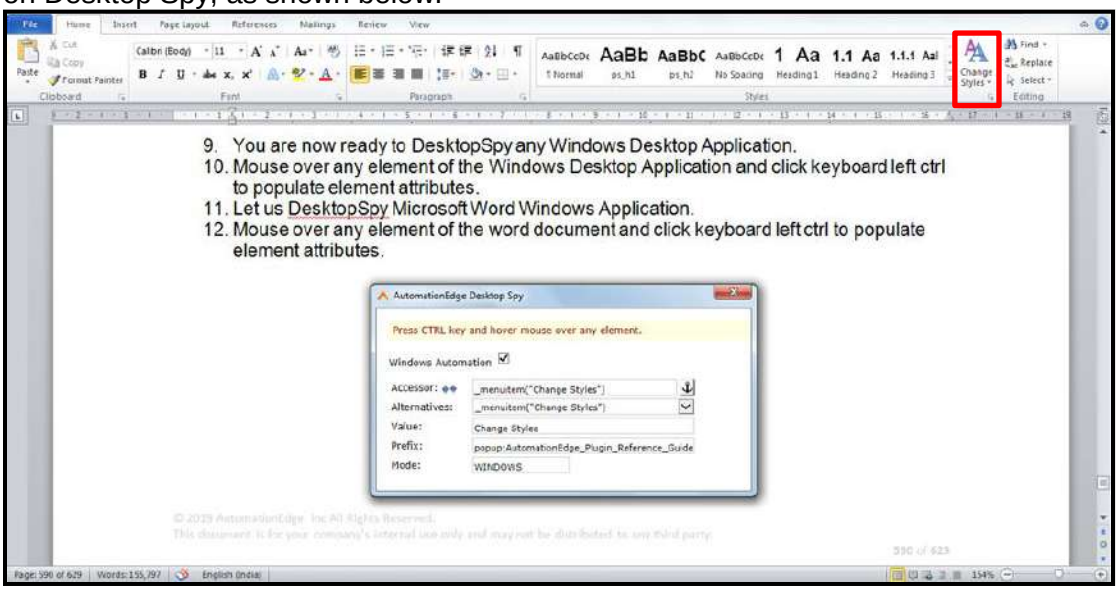


4. Click Desktop Spy. Desk Spy opens in the foreground to Process Studio, as seen in the snapshot below.

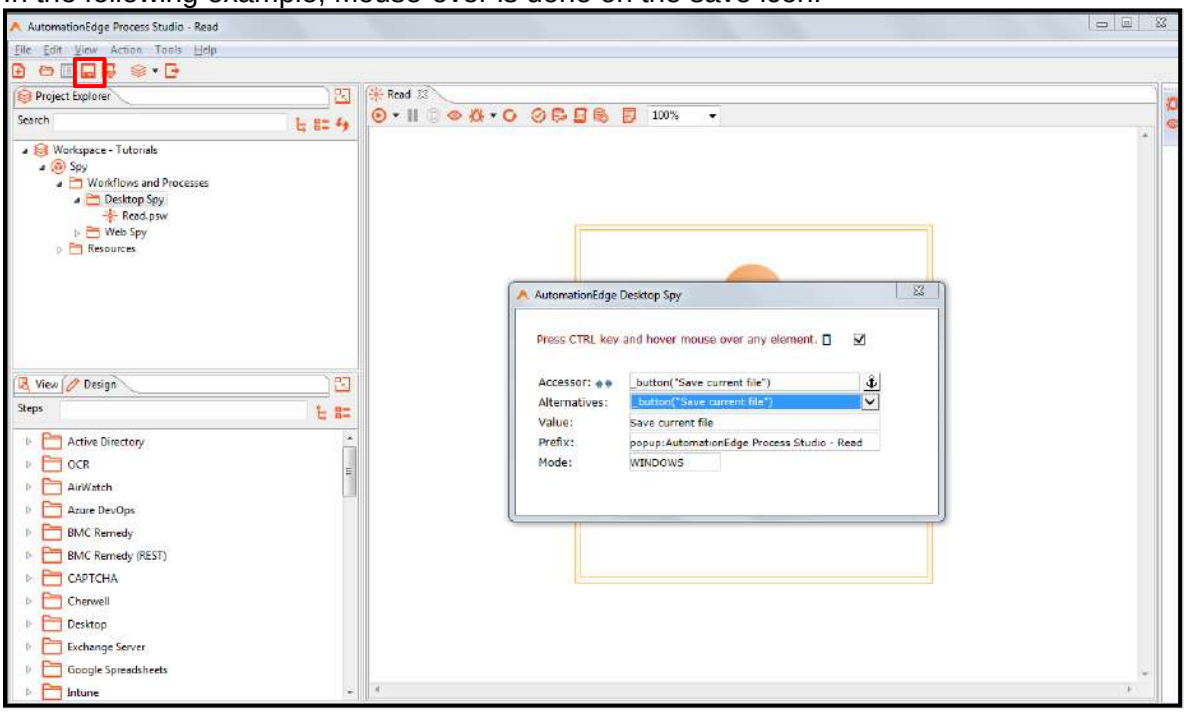



5. You are now ready to Desktop Spy on any Windows Desktop Application.
6. Mouse over any element of the Windows Desktop Application and click keyboard left ctrl to populate element attributes.
7. Let us Desktop Spy Microsoft Word Windows Application.
8. Mouse over any element of the word document and click keyboard left ctrl to populate element attributes.

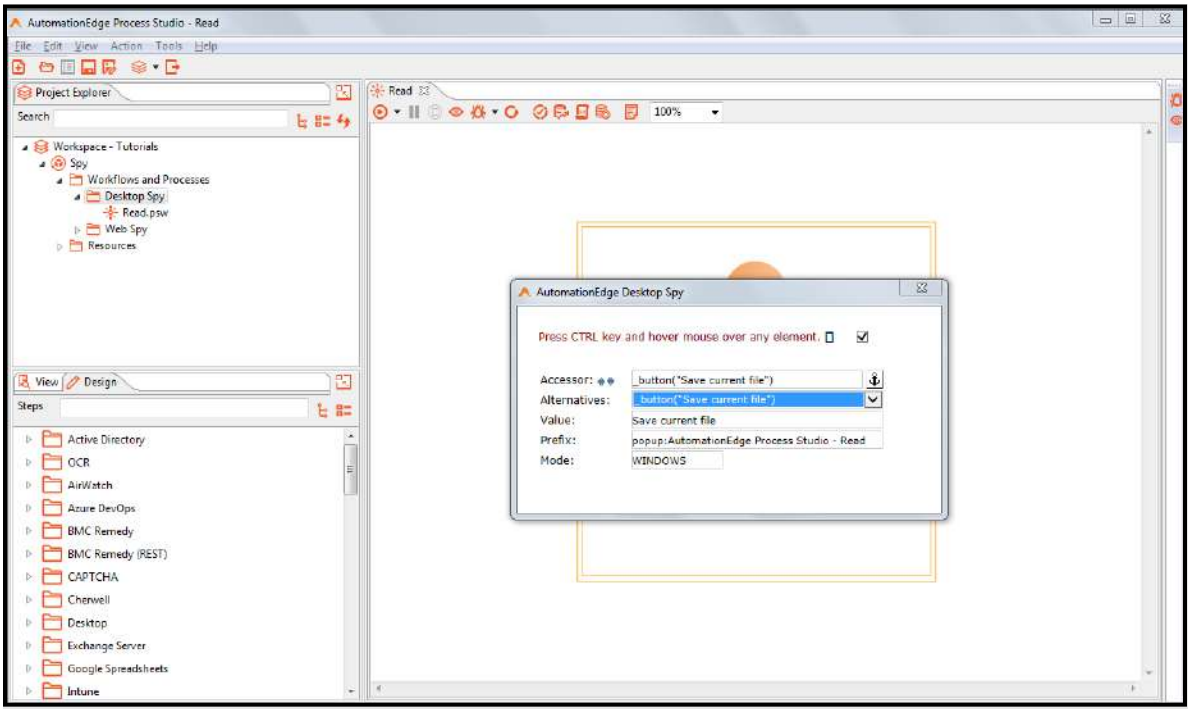
9. For Example, mouse over Change Styles and click keyboard left ctrl to populate values on Desktop Spy, as shown below.



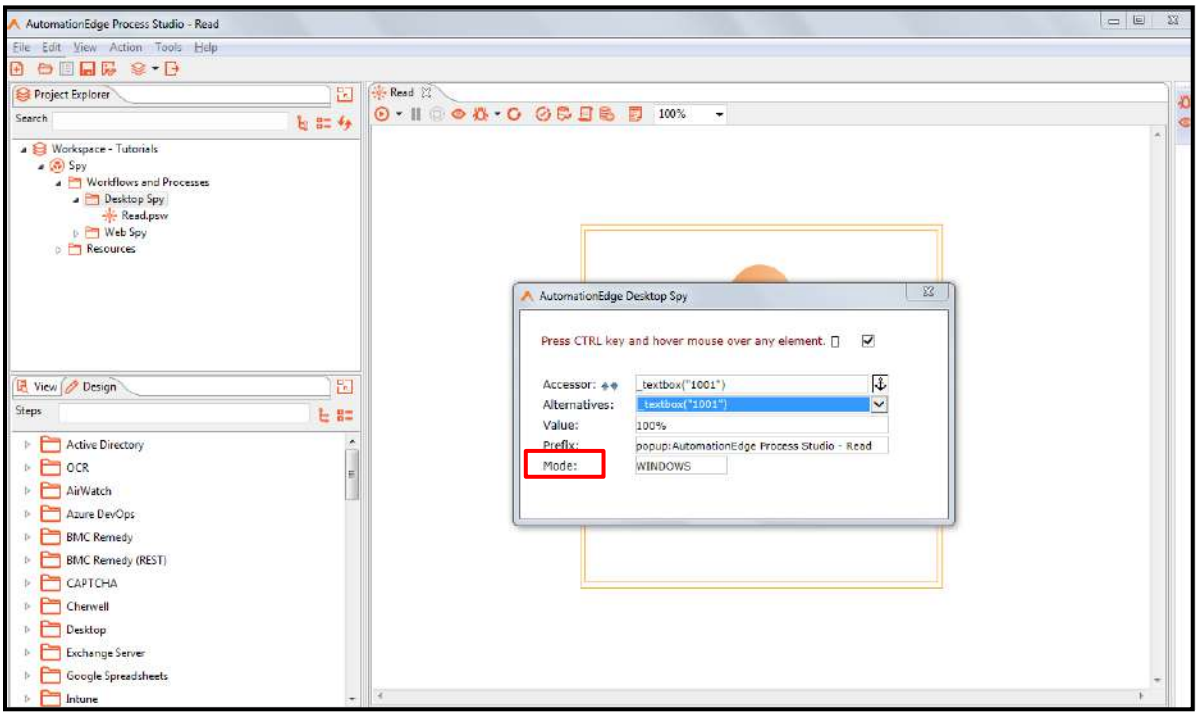

10. In the following example, Mouse-over is done on the save icon.



11. The values of Accessor, Alternatives or Prefix can be used in plugin steps as required.
12. We can also select Accessor and Alternatives from the drop-down as shown below.



13. The Value field in Desktop Spy shows the shows predicted output Value/Text of the Desktop Read plugin. The value of the text box highlighted in red is seen in the image below.




14. Similarly, you may continue Desktop Spy on other Windows applications.

| | |
|---|---|
| **Note:** | Refer to Project 8: Desktop Spy in AutomationEdge_ProcessStudio_Activity_Guide_R7.0.0 for a sample workflow 'RemoteDesktop' - with steps to identify elements with Desktop Spy. |

# 4 Appendix 4: Surface Spy Tutorial

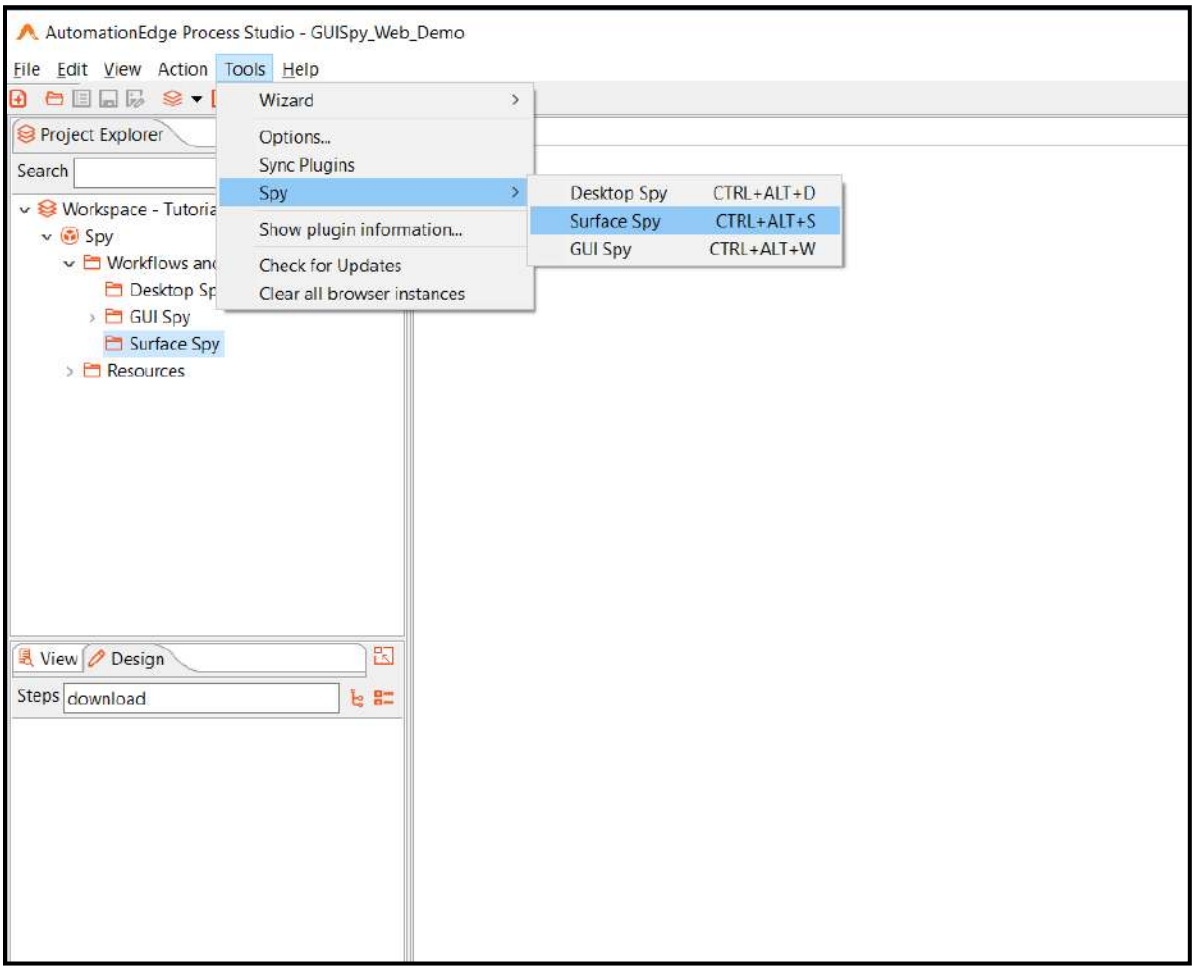
## 4.1 Introduction

Process Studio Surface Spy is used to capture images/regions on the screen. Locate this image based on a matching pattern specified. It is advantageous on remote machines with no handle to the content, and everything is an image or pattern.

## 4.2 Start Using Surface Spy

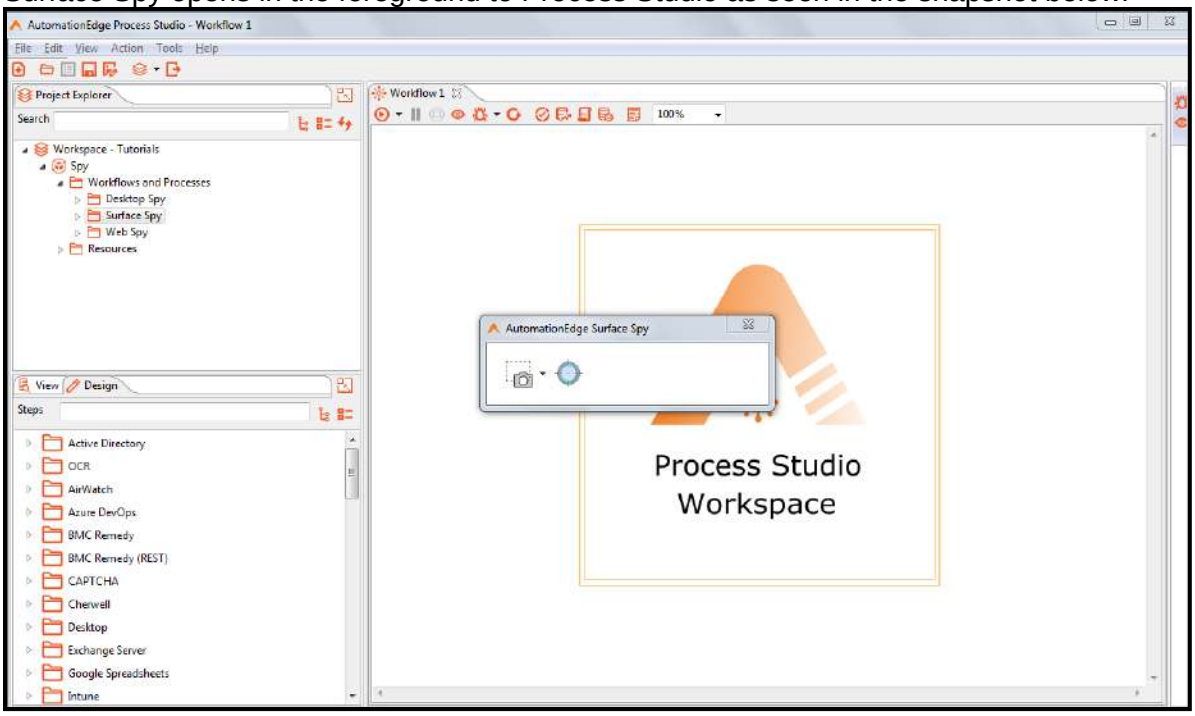
Follow the steps below to set up and use Surface Spy.

1. Open Process Studio.
2. In this demo, In Spy Project, create a folder Surface Spy. The Surface Spy folder is visible in the workspace as below.
3. To launch Surface Spy, click Tools→Spy →Surface Spy menu option.




4. A warning message to create a workflow displays. The default workflow name is Workflow_1. Click OK.

5. Surface Spy opens in the foreground to Process Studio as seen in the snapshot below.

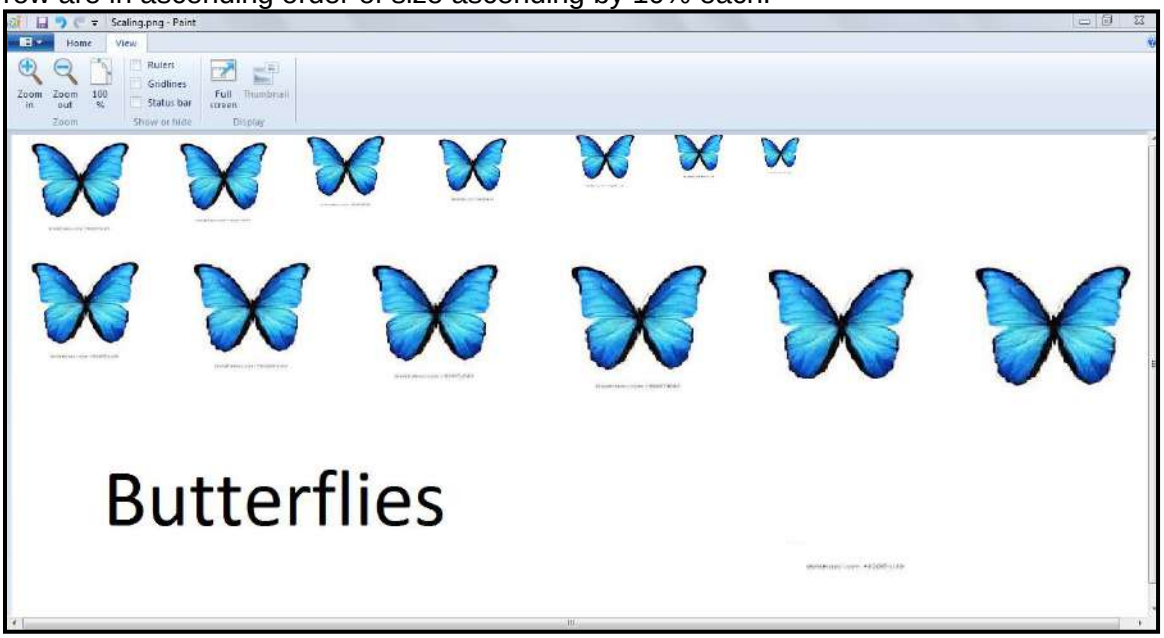



6. The two icons on Surface Spy are explained below and implemented in the steps that follow.

| Icon on Surface Spy | Description |
|---|---|
|  | On hovering over the icon it shows the name of the icon - Capture Image<br><br>If you click on the drop down arrow it shows three options,<br>• Capture now<br>If you click on the Capture Image icon or select Capture now from the drop down list, you see a + mouse pointer that can be used to outline any area on the screen.<br>• Capture in 3 Seconds<br>If you select Capture in 3 Seconds from the drop down list you see a + mouse pointer after 3 seconds, that can be used to outline any area on the screen..<br>• Capture in 10 Seconds<br>If you select Capture in 10 Seconds from the drop down list you see a + mouse pointer after 10 seconds, that can be used to outline any area on the screen. |

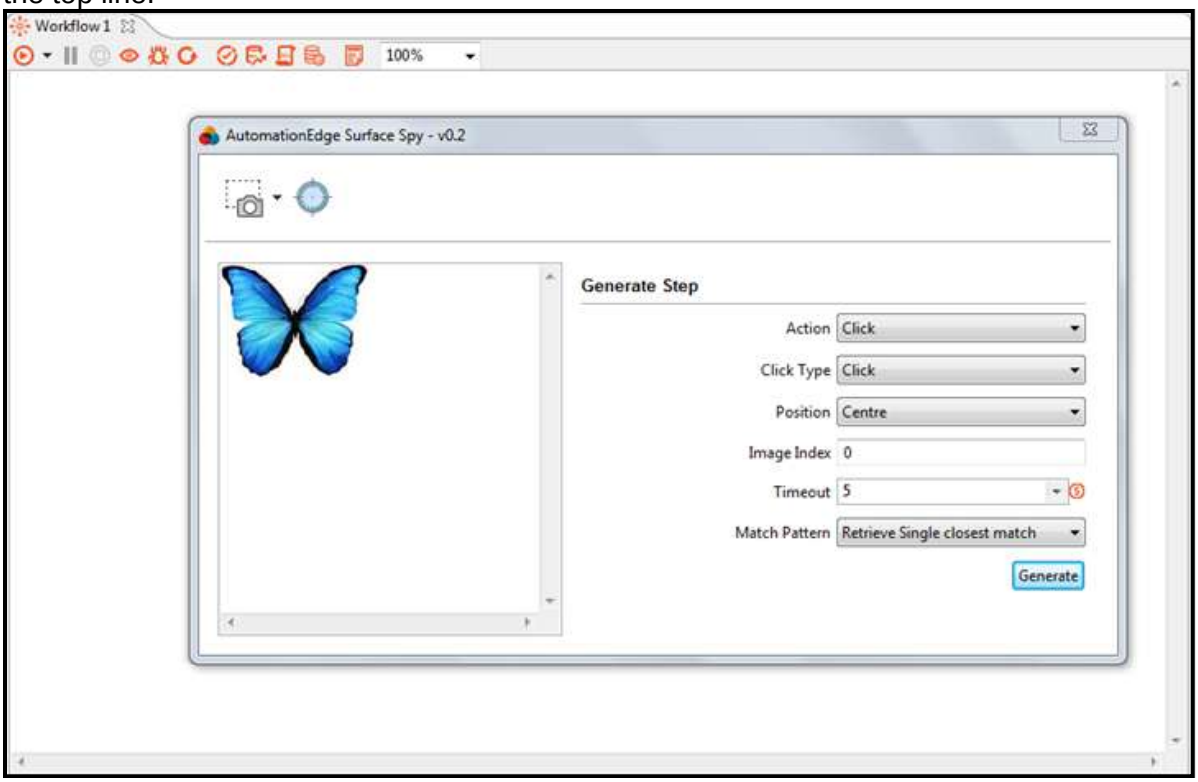
| Icon on Surface Spy | Description |
|---|---|
| | This is useful if the area to outline is not on the screen immediately but you need less than 3 or 10 seconds as mentioned in the two points above, to navigate to a desired screen. |
|  | On hovering over the icon, it shows the name of the icon - Locate Images<br>Once you outline an area using Capture Image, the next step is to Locate Images. Clicking on Locate images icon searches for matching patterns(/images) based on the Matching Pattern selection from:<br>• Retrieve Single closest match<br>• Retrieve Multiple close matches<br>• Retrieve All matches |

7. Capture Image on the paint document as shown in the screenshot below. Let us assume the image on the top left corner to be the base image with a size 100%. The images on the right are smaller images with 10% reduction in size each. The images in the bottom row are in ascending order of size ascending by 10% each.

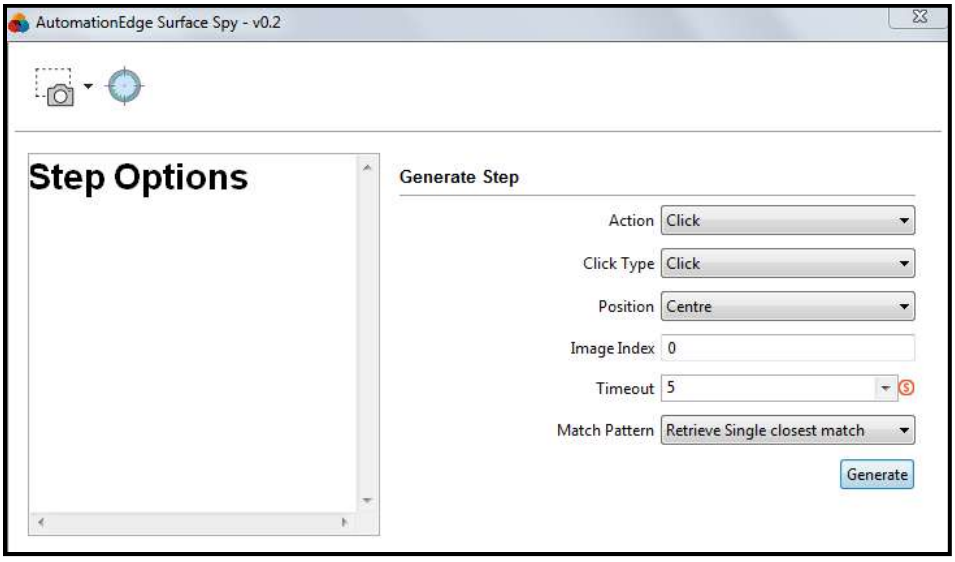



8. Once you capture an image or pattern, the AutomationEdge Surface Spy window expands, as shown below. In this case, we have captured the third image from the left in

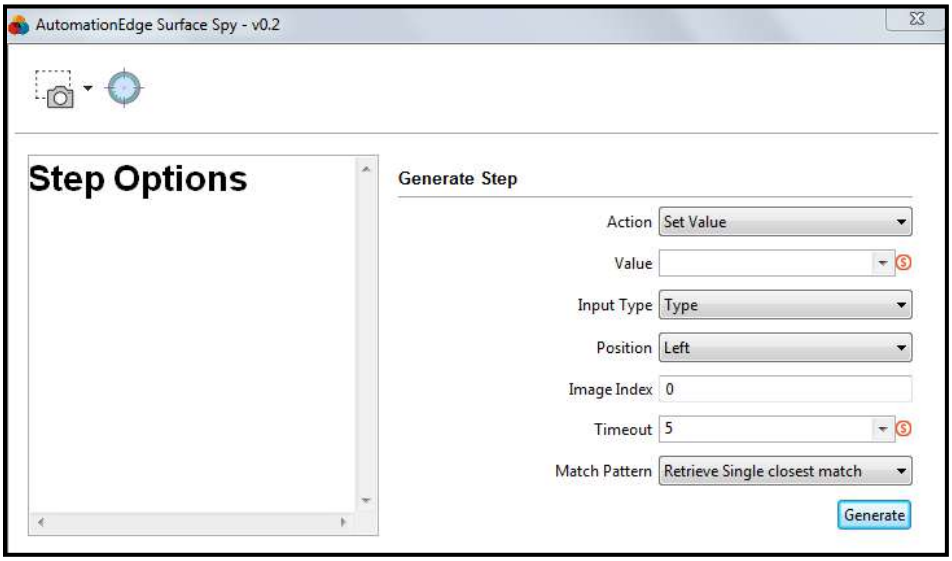
the top line.



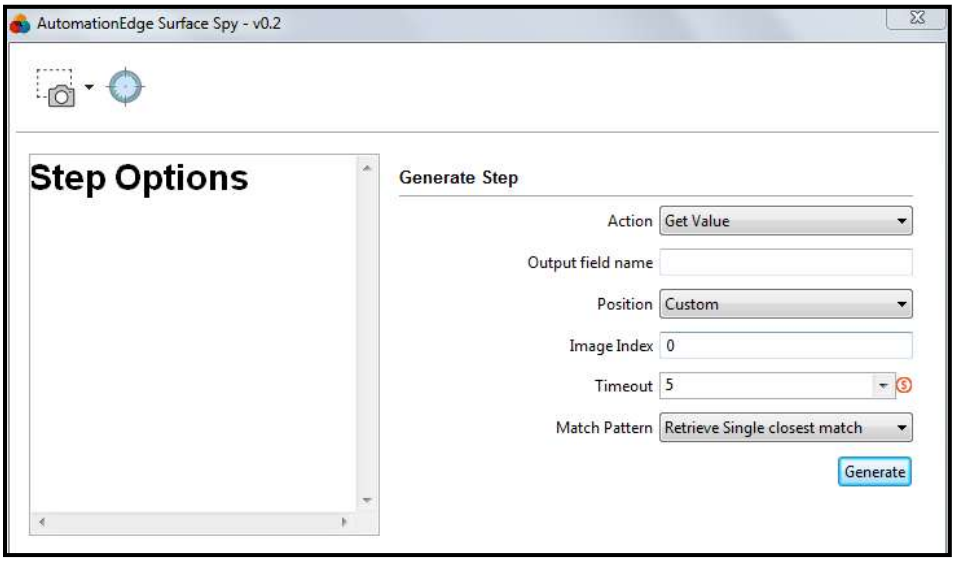

9. Generate Step configuration screen options differ depending on the Action chosen.
10. The following screen shows the Generate Step configuration options when Action chosen is Click as seen above also.



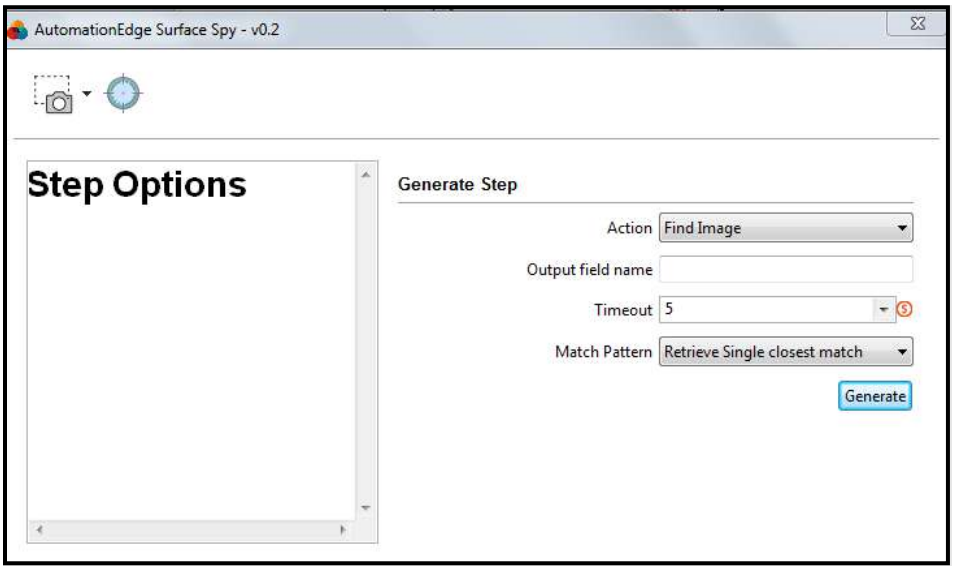
11. The following screen shows the Generate Step configuration options when Action chosen is Action Set Value




12. The following screen shows the Generate Step configuration options when Action chosen is Action Get Value




13. The following screen shows the Generate Step configuration options when Action chosen is Action Find Image.



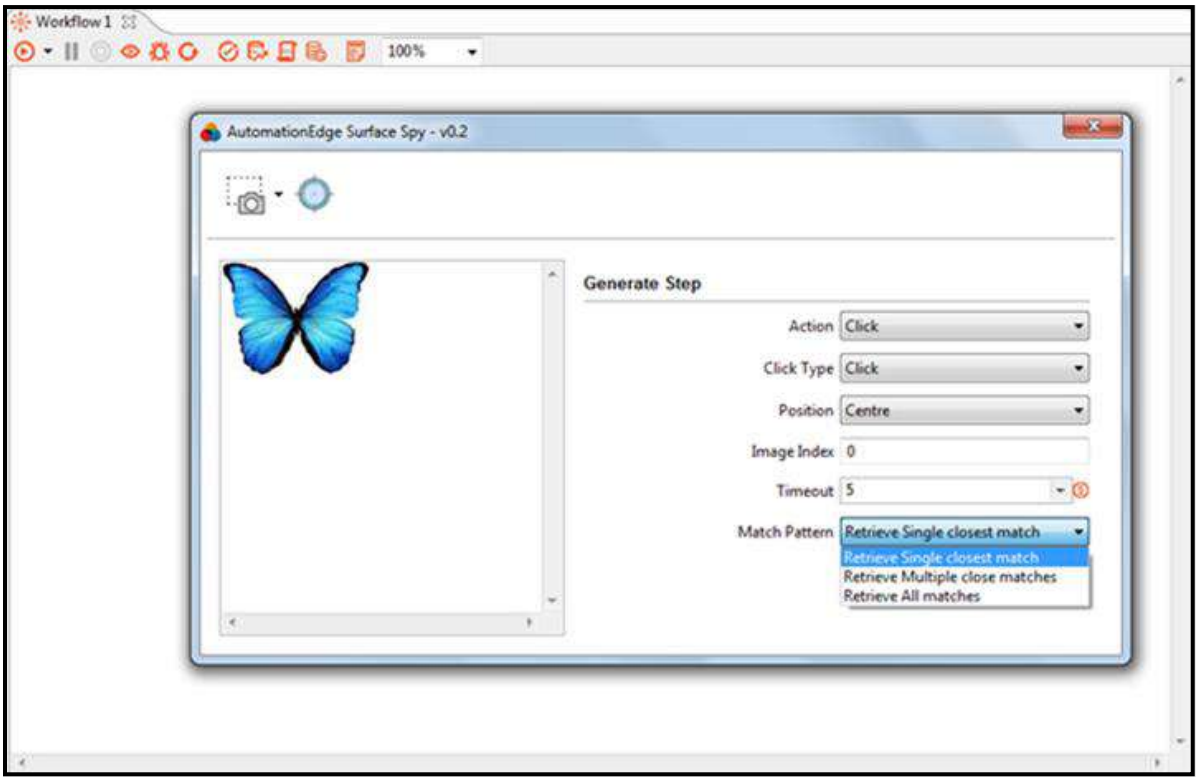

14. The following table summarizes all the Generate step options for all Actions types.

Table: Surface Spy Generate Step Options

<table>
<tr><th>No.</th><th>Option</th><th>Description</th></tr>
<tr><td>1</td><td>Action</td><td>The Action Drop Down has the following options available.<br><br><table><tr><td>No</td><td>Click Types</td><td>Description</td></tr><tr><td>1</td><td>Click</td><td>It performs a click action.</td></tr><tr><td>2</td><td>Set Value</td><td>It performs Set Value on a region/image.</td></tr><tr><td>3</td><td>Get Value</td><td>It Gets Value from a region/image.</td></tr><tr><td>4</td><td>Find Image</td><td>It checks for an image and returns true or false.</td></tr></table></td></tr>
<tr><td>2</td><td>Click Type</td><td>The Click Type Drop Down has the following options available. This option is available if Action chosen in Click.<br><br><table><tr><td>No</td><td>Click Types</td><td>Description</td></tr><tr><td>1</td><td>Click</td><td>It performs left click action.</td></tr><tr><td>2</td><td>Right Click</td><td>It performs right click action.</td></tr><tr><td>3</td><td>Double Click</td><td>It performs double-click action.</td></tr><tr><td>4</td><td>Middle Click</td><td>It performs left Middle click action.</td></tr><tr><td>5</td><td>Mouse Move</td><td>It performs Mouse Move action.</td></tr></table></td></tr>
<tr><td>3</td><td>Value</td><td>Specify a Value to be set on a region/image.</td></tr>
<tr><td>4</td><td>Output field name (for Action Get Value)</td><td>This field is active when the Action chosen is Get Value. Provide a field name to hold the value retrieved from a region.</td></tr>
<tr><td>5</td><td>Output field name (for Action Find Image)</td><td>This field is active when the Action chosen is Find Image. Provide a field name to hold a Boolean value(Y/N) indicating whether the image was found or not.</td></tr>
</table>

<table>
<tr><th>No.</th><th>Option</th><th>Description</th></tr>
<tr><td>6</td><td>Position</td><td>The Position Drop Down has the following options. Perform the Action chosen above at the position selected here.
<table>
<tr><th>No</th><th>Positions</th><th>Description</th></tr>
<tr><td>1</td><td>Left</td><td>Center Left position of the Image.</td></tr>
<tr><td>2</td><td>Right</td><td>Center Right position of the Image.</td></tr>
<tr><td>3</td><td>Center</td><td>Center of the Image.</td></tr>
<tr><td>4</td><td>Custom</td><td>The relative position from the entire Image, it could be anywhere on the screen. If the relative position of the relative element to click is closer, it is preferable.</td></tr>
</table></td></tr>
<tr><td>7</td><td>Image Index</td><td>Image Index is retrieved once an image is clicked from the selected images.</td></tr>
<tr><td>8</td><td>Timeout</td><td>Specify this is the Timeout for step execution. By default, timeout is 5 seconds</td></tr>
<tr><td>9</td><td>Match Pattern</td><td>The Match Pattern Drop Down has the following options available.<ul><li>Retrieve Single closest match</li><li>Retrieve Multiple close matches</li><li>Retrieve All close match</li></ul></td></tr>
<tr><td>10</td><td>Button: Generate</td><td>Click Generate button to generate the step with the configurations above.</td></tr>
</table>

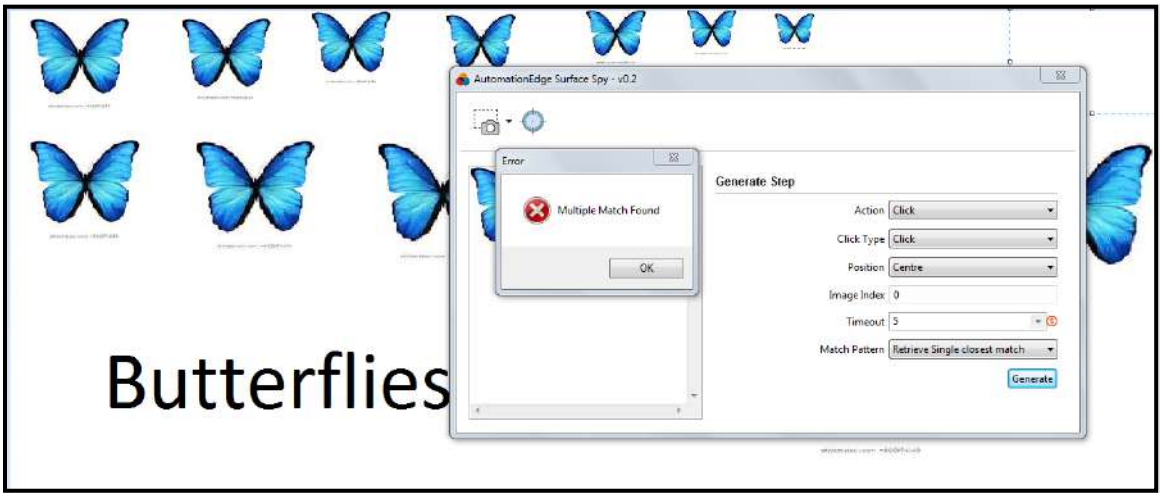
15. The Match Pattern drop-down field has the following options as shown below,
    - Retrieve Single closest match
    - Retrieve Multiple close matches
    - Retrieve All close match



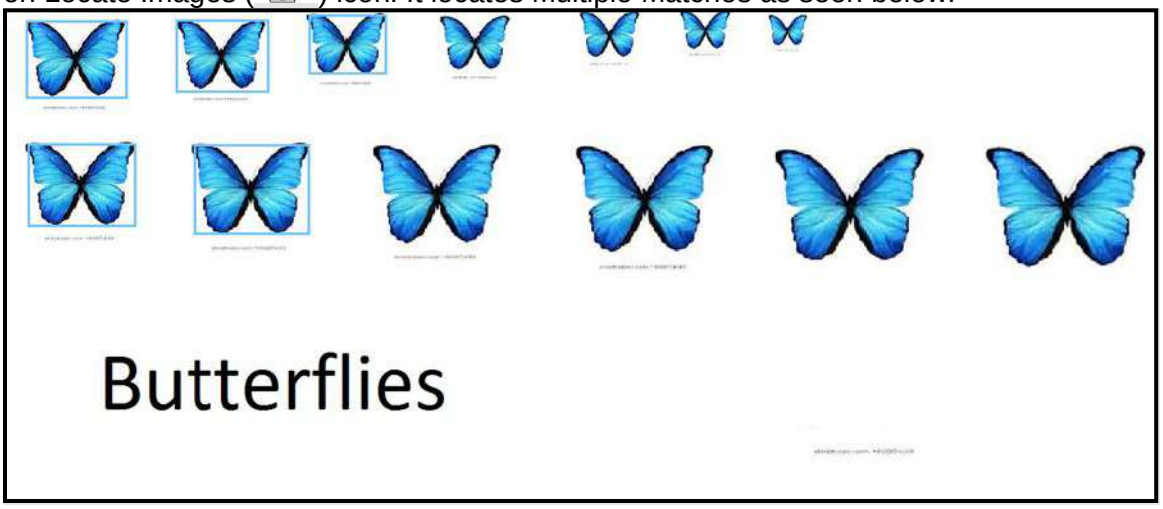

16. Once you outline an area using Capture Image the next step is to Locate Images. Clicking on Locate images icon searches for matching images (/patterns) depending on the Match Pattern chosen as mentioned above.

17. Let us select Retrieve Single closest match in the match Pattern drop down. Now click on Locate Images ( ) icon. It gives an error Multiple Match Found as it has found more than one close matches.

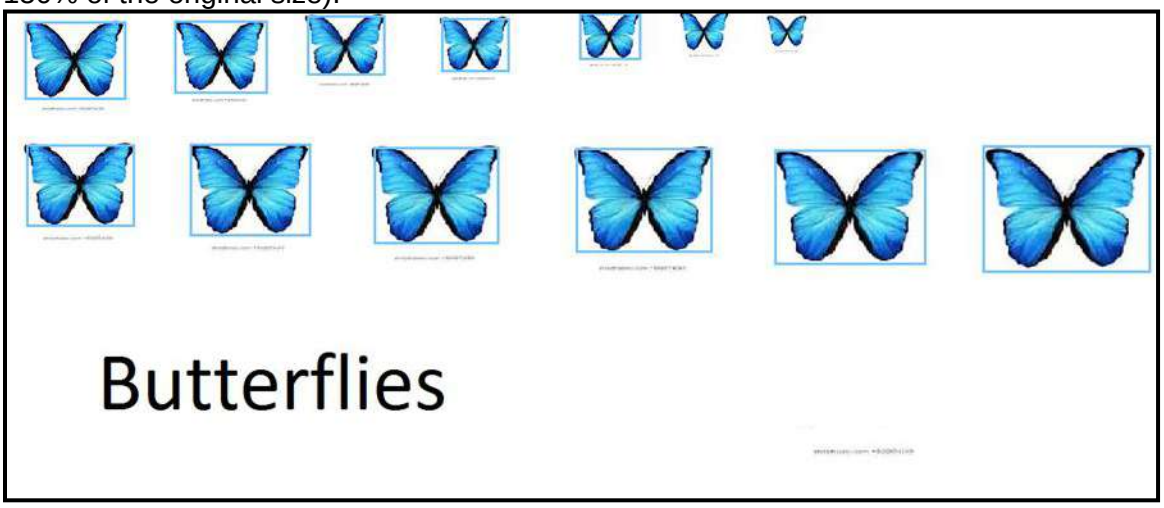
| Note | We Click on an image to select and press Esc key to exit. |
|---|---|




18. Let us select Retrieve Multiple close matches in the match Pattern drop down. Now click on Locate Images ( ) icon. It locates Multiple Matches as seen below.



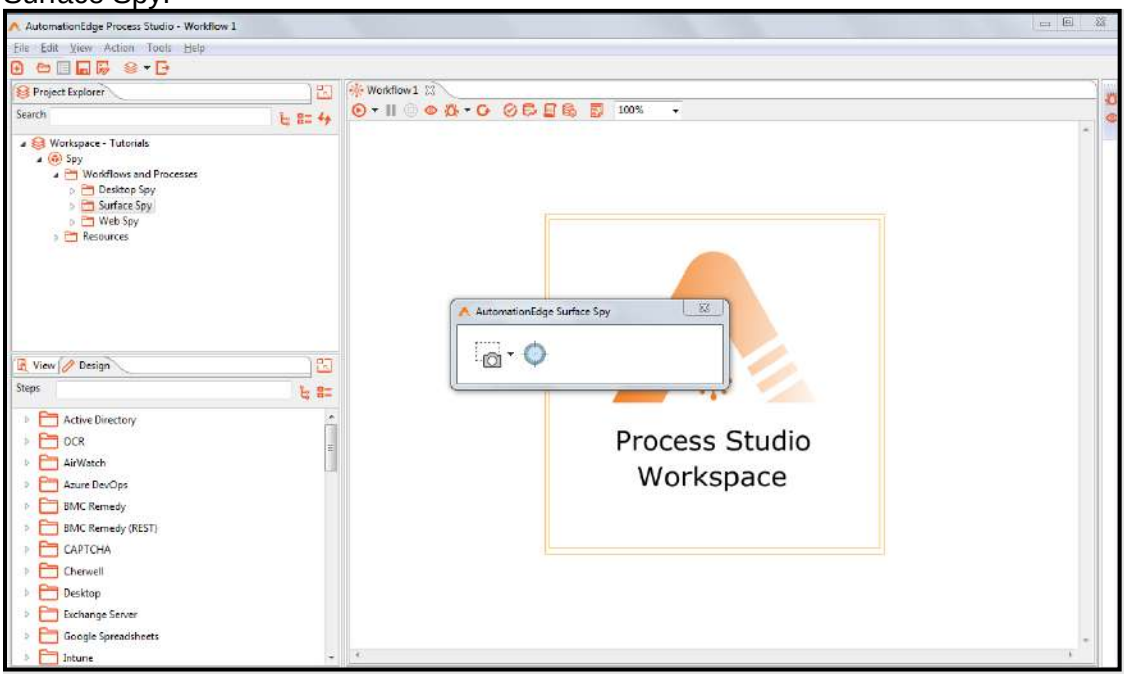
19. Let us select Retrieve All matches in the match Pattern drop down. Now click on Locate Images ( ) icon. It locates All Matches as seen below (not below 50% or above 150% of the original size).



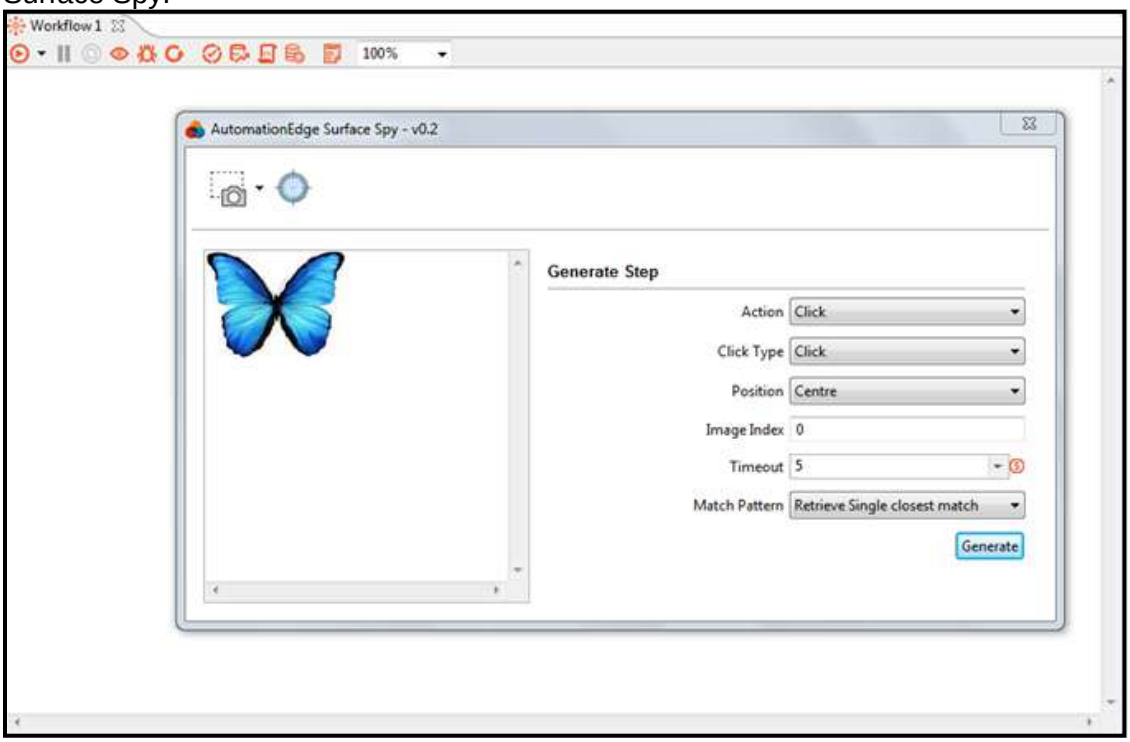

20. In this section we saw, how Surface Spy is used Capture and Locate Images and the various options available.

## 4.3 Close Surface Spy

1. You may click the X button on the AutomationEdge Surface Spy window to close Surface Spy.



2. You may also click X on the expanded AutomationEdge Surface Spy window to close Surface Spy.




3. Surface Spy also closes when you close Process Studio.

| | |
|---|---|
| **Note:** | Refer Project 8: Surface Spy in AutomationEdge_ProcessStudio_Activity_Guide_R7.0.0 for a sample workflow using Surface Spy. |

# Contact

## AutomationEdge Technologies Inc

General: info@automationedge.com

Support: support@automationedge.com

Trainings: training.support@automationedge.com

Website: https://automationedge.com

India: +91-20-6632-1000

USA: +1-281-220-1057