AFSG

(Assurant Framework Step Generator)

User Manual

Prepared for

AEB, Kansas

Table of Contents

Section 1: About This Document

The Framework Step Generator User Manual is designed with a purview of answering the most frequently asked question – "*How does Step Generator –Work?*"

The document highlights on step generator utility usage and its advantages. The document also speaks about to usage of the Step Generator to create; update and deletion of test steps in automation script along with object management and script execution.

Section2: Step Generator Artifacts

Overview

The framework step generator uses different types of test artifacts to create; update and deletion of objects and test steps for new test scripts. They are Admin login, Step Creation, Add Object and Delete/Update Object. All these artifacts are ***stored*** in i-ASTG folder.

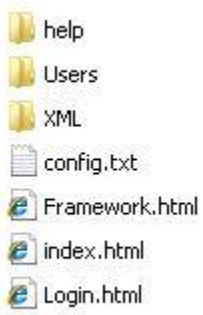
Types of Artifacts

Test Artifacts	Description	Location	Can be modified?
Login	The main page where user uses his/her LAN ID and password to login to the application	*T:\Test Services\Offshore Folder\Final_VI\HTML PRoj/Login.html*	Yes. Only user having access to QC Project: AEB_AUTOMATION and Domain: Web_Automation can login.
Index Page • Select Application • Add Application • Update Application • Delete Application	This page contains "Select Applications, object Repository and Test Script Path browse button" i.e. Available Applications for automation along with Application Object Repository path and Test Script Path (XML).	*E.g.:* *Application Name: SAS* *Application OR Path: T:\Test Services\Offshore Folder\Final_VI\HTML PRoj\XML\Application OR\SAS\SAS.xml* *Test Script Path: T:\Test Services\Offshore Folder\Final_VI\HTML PRoj\lr1385\temp\temp.xml*	Yes. Select Application and Test Script path can be modified.
Framework Page • Add Step • Object Repository • Action Management • Function Library Management • Execute Script	**Add Step** tab contains Test Case details i.e. Test Case Id, objects details and its respective action list and data parameter.		Yes. Add Step for automation can be added and it can be modified.
	Object Repository tab contains the list of the properties to be added in to the object.		Yes. Adding the objects with different properties can be achieved.
	Action Management tab is to manage actions i.e. new keywords to add or to change the name of the existing actions.		Yes. Adding the actions can be achieved.
	Function Library Management tab is to manage the paths of function library.		Yes. Adding the function libraries along with paths can be achieved.
	Execute Script tab is to execute scripts.		Yes. User can execute script from this tab by passing the Test Case ID.
Config.txt	This Properties file can be accessed using notepad and configure the settings for the paths of the different files.	//config.txt	Yes. We need to make sure that all the settings in this file are valid before executing a test case. Values can be changed by opening file from the login Page.

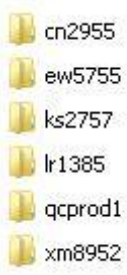
Test Artifacts	Description	Location	Can be modified?
Temp.xml	This XML contains the temporary test case of the test cases to be executed.	\\Users\ <User Name>\ Temp\ temp.xml e.g. For ***lr1385*** application: *\\Users\lr1385\Temp\temp.xml*	No.
Actions.XML	This XML contains the set of actions.	XML\Actions.xml	Yes. From application itself.
ApplicationsList.XML	This XML contains the set of Applications.	XML\ ApplicationsList.xml	Yes. From application itself.
Class.XML	This XML contains the set of Classes and classID.	XML\ Class.xml	Yes. From application itself.
FunctionLibraries.XML	This XML contains the set of Function Libraries and path of Function Libraries.	XML\ FunctionLibraries.xml	Yes. From application itself.

Folder Structure

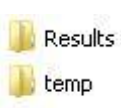
Main Folder \\StepGenerator



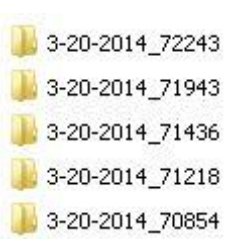
User Folder \\ StepGenerator \Users



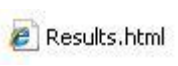
User Folder \\ StepGenerator \Users\lr1385



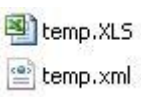
Results Folder \\ StepGenerator \Users\lr1385\Results



Result File \\ StepGenerator \Users\lr1385\Results\



Temp Folder \\ StepGenerator \Users\lr1385\temp\



Folder	Description
Users	• Individual user level folder. • Used to manage files for user level.
Results	• Used to store execution results. Date and Time stamp is used to differentiate. • Ex: 3-20-2014_72243
Temp	• Temporary folder to save the current xml and xls. • Contains temp.xml and teml.xls

Section3. Software

Prerequisites

3.1.1 Windows XP/ windows 7

3.1.2 HP QTP

3.1.3 HP QC Credentials

3.1.4 Internet Explorer Version 8

3.1.5 Access to "\\T Drive"

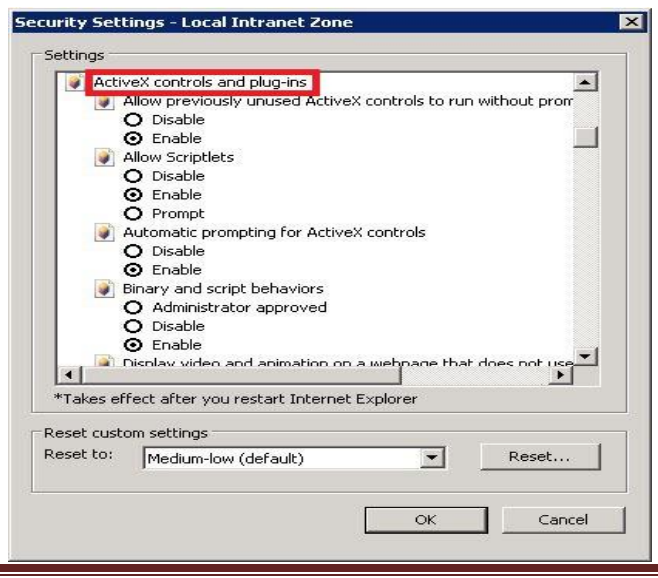
Browser Settings

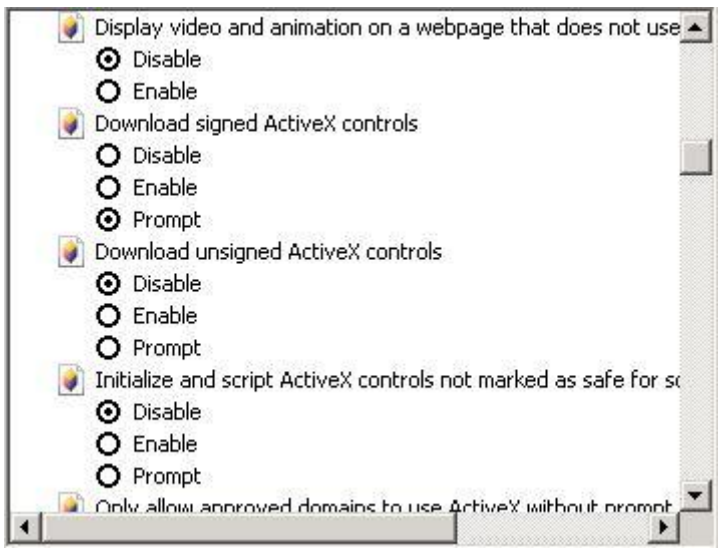
Note: **Supported Browsers**: IE8. Following settings are related to IE8

Step 1: Go to Tools ->Internet Options.

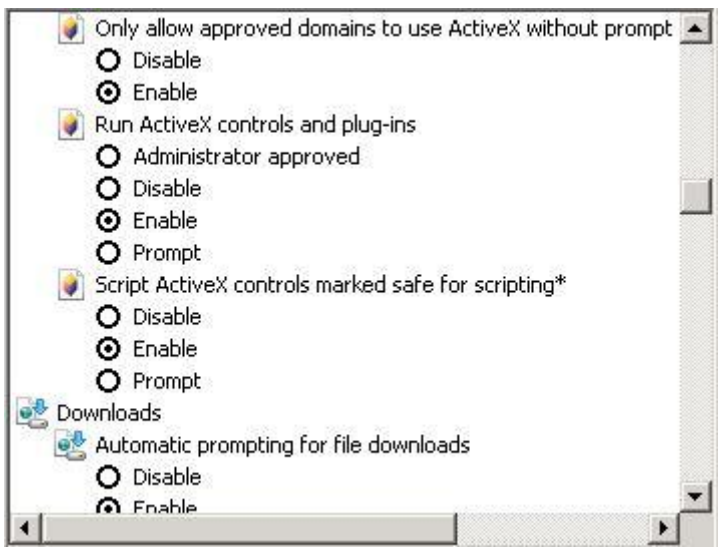
Step2: Click on the **Local internet** symbol and click on Custom Level button

Step 3: In the **ActiveX controls and plug-ins** options mark everything "**enabled**".

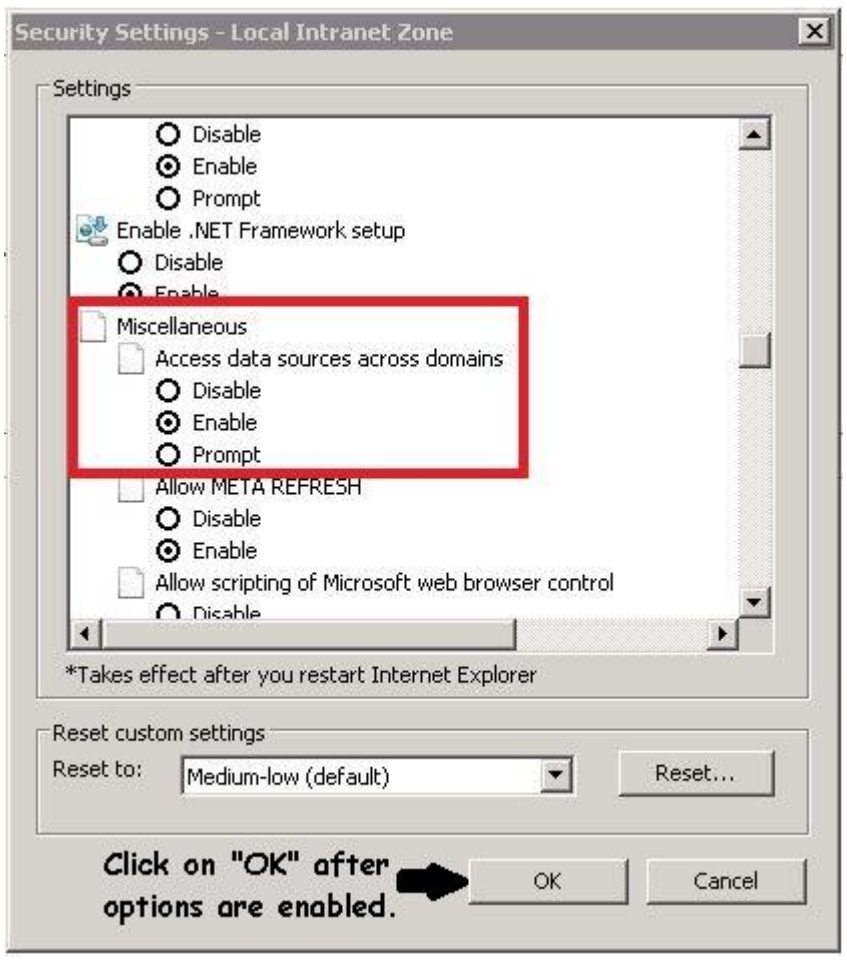


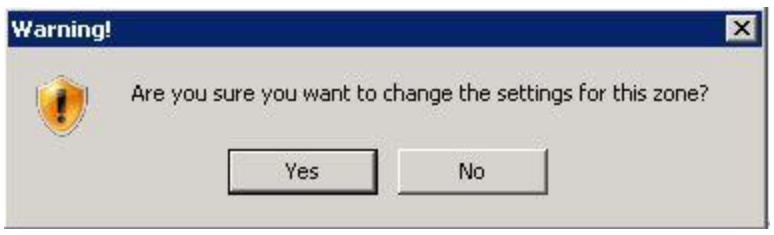


Select **Enable** up to Script ActiveX controls marked safe for scripting*

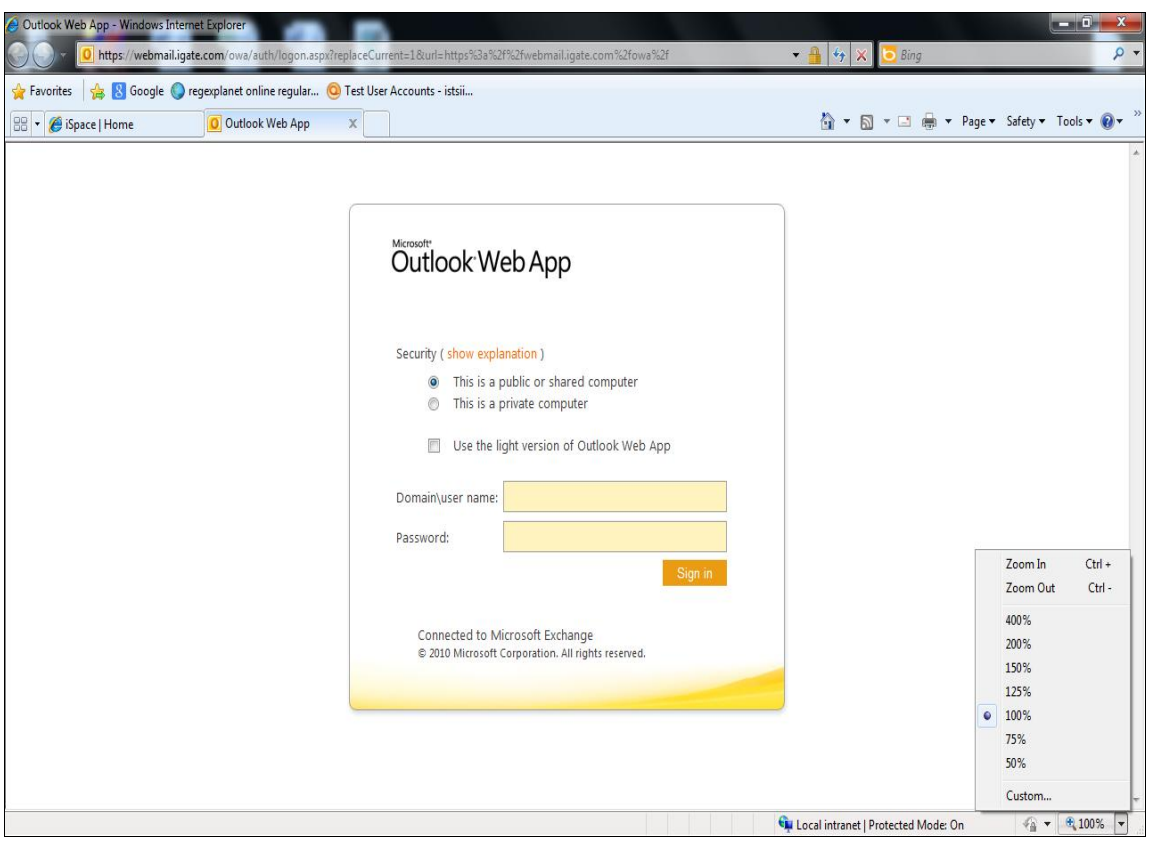


Step 4: Scroll down to Miscellaneous Options, Select **Enable** for "Access data sources across domains". Step 4: Click on "OK" button and then if it throws a warning message, click on Yes





Step 6: Make sure that Zoom is set to 100% in IE.



Step 7: Install IE.reg from the home folder. This is required to avoid unnecessary script prompts.

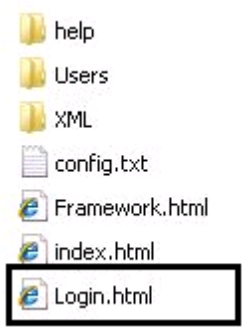
Section4.Using Step Generator…

4.1Prerequisites

- **HP Quality Center credentials** should be available.
- Browser setting should set prior to launch the login file.

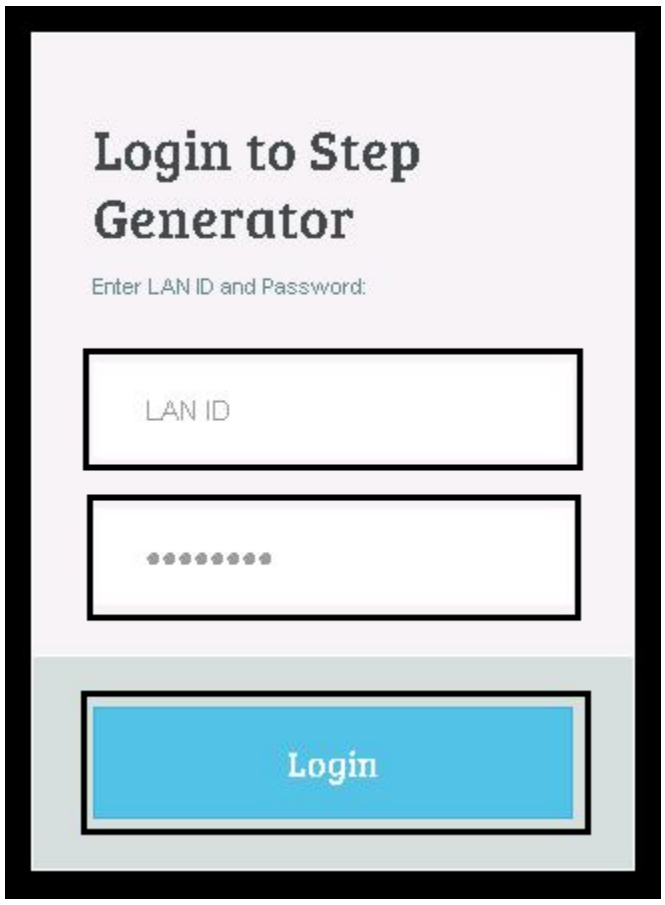
4.2 Login to Step Generator

Step 1: Open 'Login.html' placed in the 'Step Generator' folder.



Step 2: Change the **config** settings if required from the link available on the Login Page (Right Bottom of the page).

Step 3: Enter credentials (Quality Center) and click on login button.

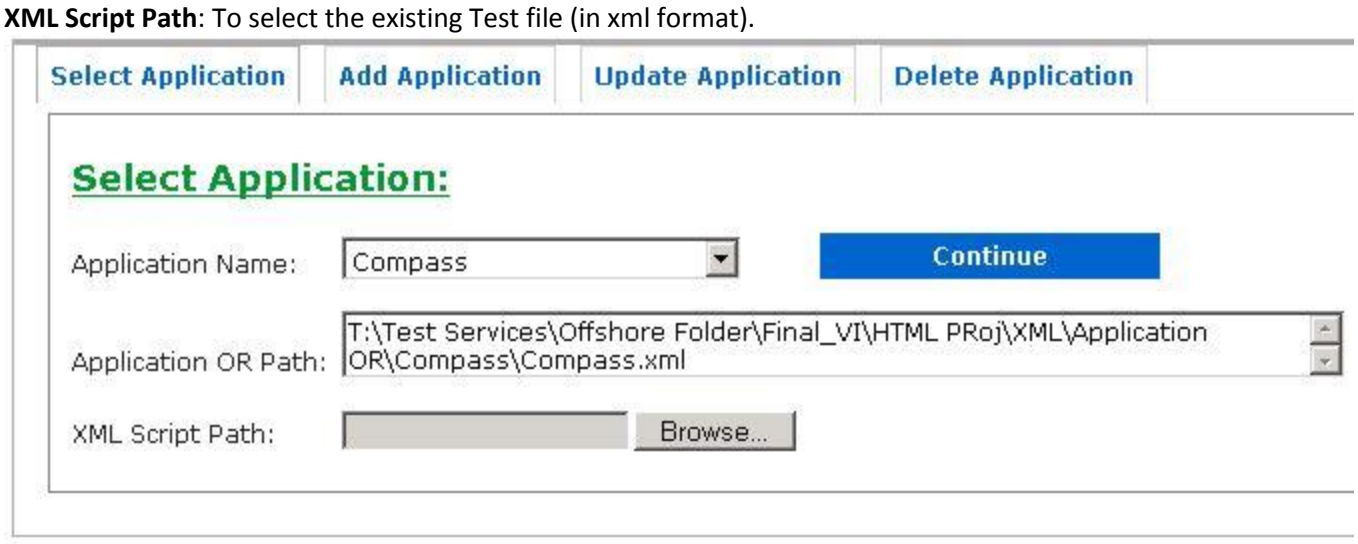


4.3 Application Maintenance

4.3.1 Select Application

This tab is used to select the application available from the list.

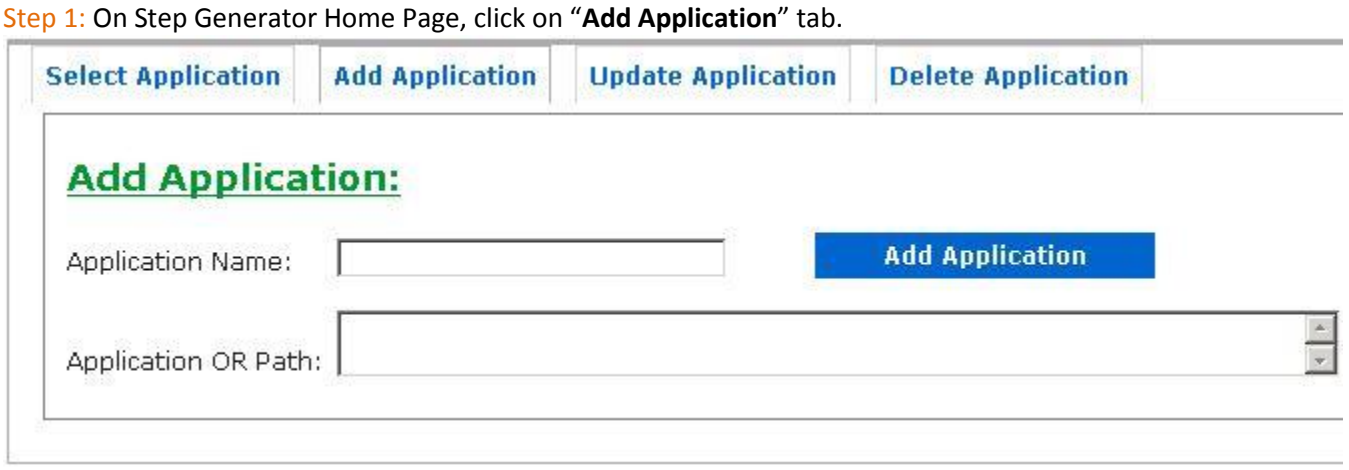
Application Name: Drop Down with all available applications.
Application OR Path: Not editable. This box shows the path of Application Object Repository Path.
XML Script Path: To select the existing Test file (in xml format).



4.3.2 Add Application

This tab is used to add new application.

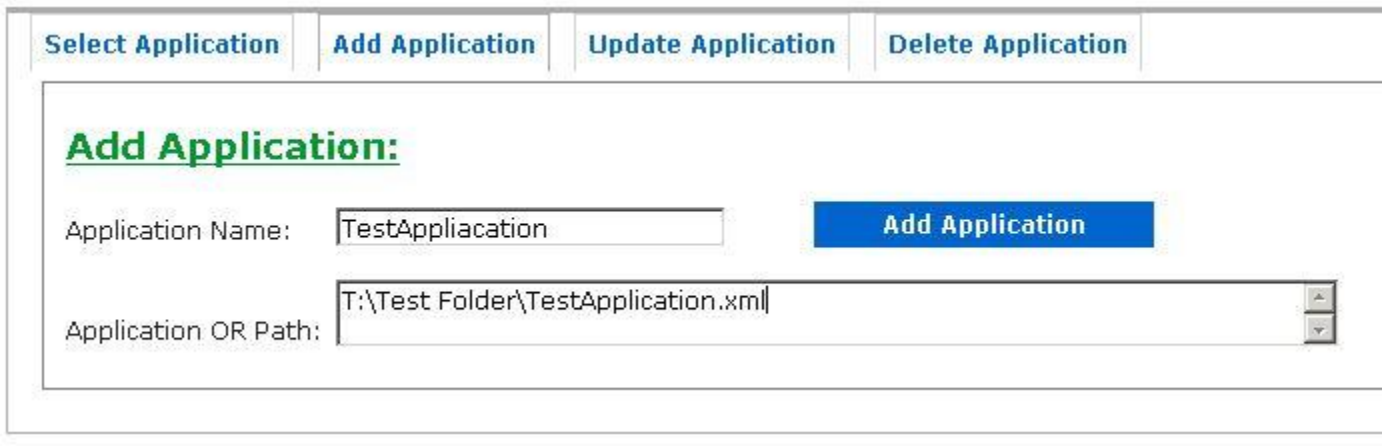
Step 1: On Step Generator Home Page, click on "**Add Application**" tab.



Step 2: Enter Name of the application and path of the Object Repository in XML format of the new application.

Example:

Application Name: "TestApplication"
Application OR Path: "T:\Test Folder\TestApplication.xml"

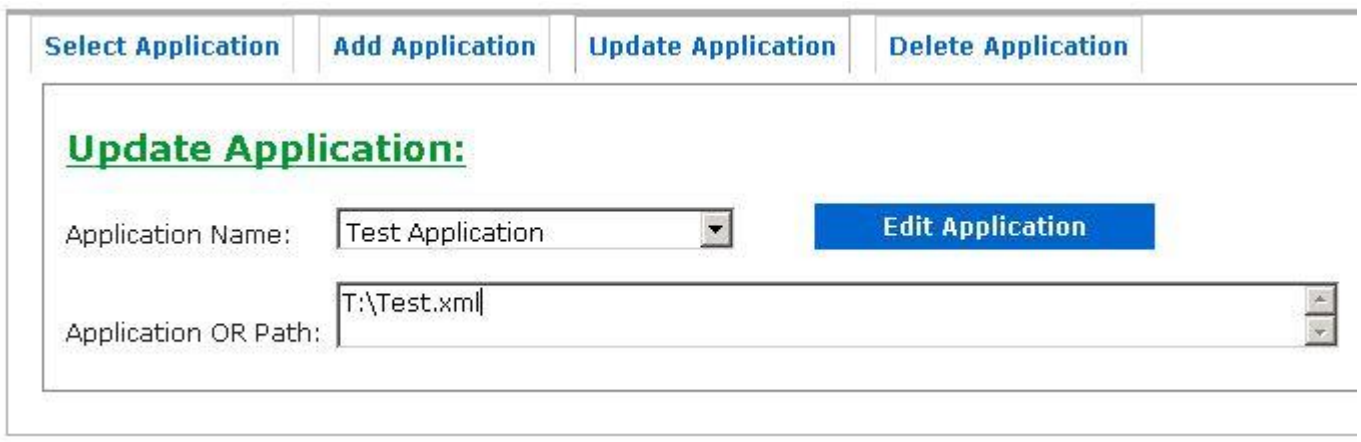


Step 3: Click on **Add Application** button. Click "**OK**" on the message box.

4.3.3 Edit Application

This tab is used to update the **Application OR Path**.

Step 1: On Step Generator Home Page, click on "**Update Application**" tab.

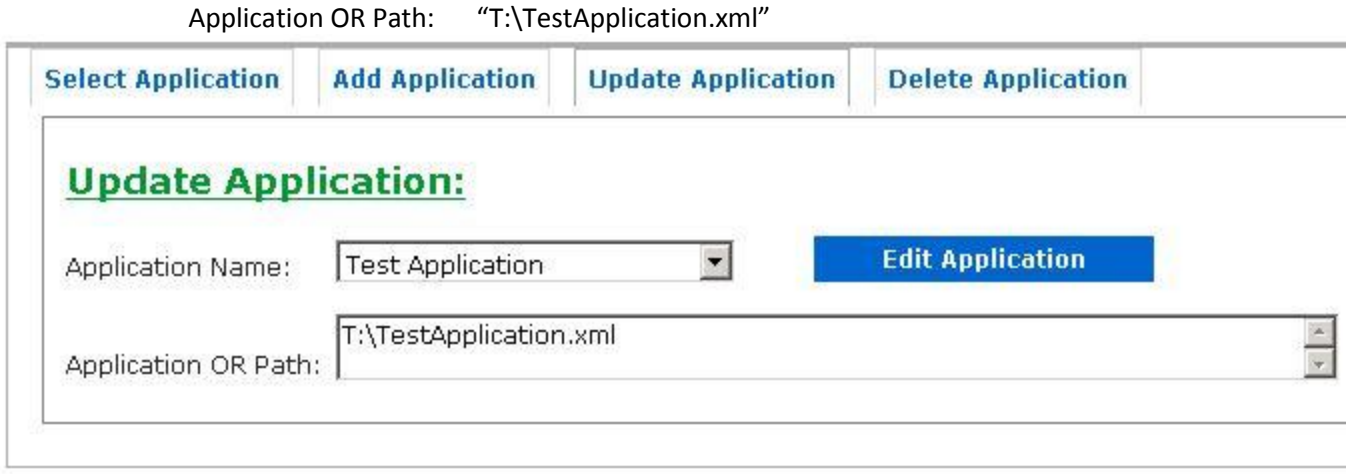


Step 2: Enter Name of the application and path of the Object Repository in XML format of the new application.

Example:

Application Name: "TestApplication"

Application OR Path: "T:\TestApplication.xml"

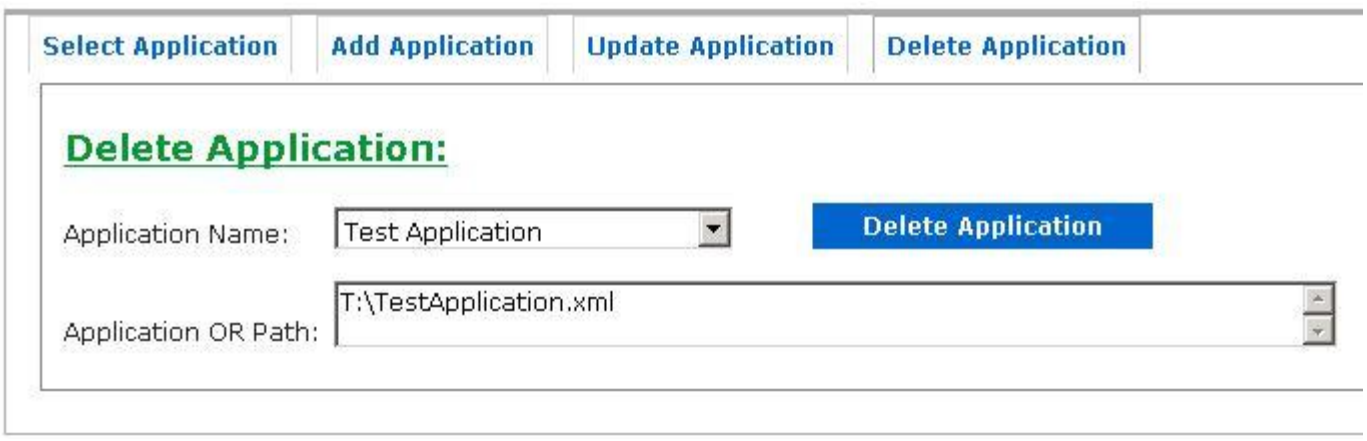


Step 3: Click on **Edit Application** button. Click "**OK**" on the message box.

4.3.4 Delete Application

This tab is used to delete the **Application**.

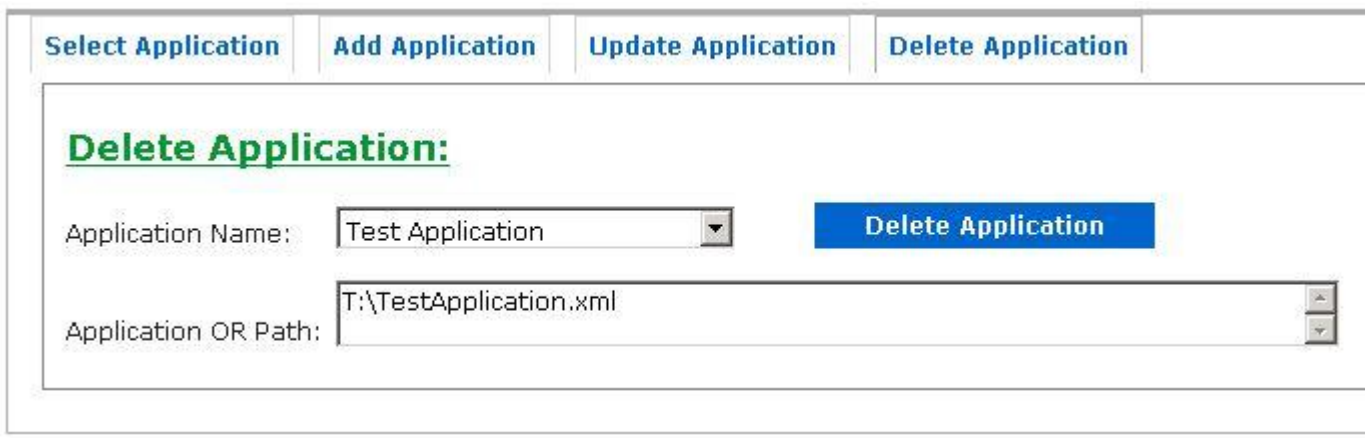
Step 1: On Step Generator Home Page, click on "**Update Application**" tab.



Step 2: Select Application from the list.

Example:

Application Name: “TestApplication”



Step 3: Click on **Delete Application** button. Click “**OK**” on the message box.

4.4 Object Repository Management

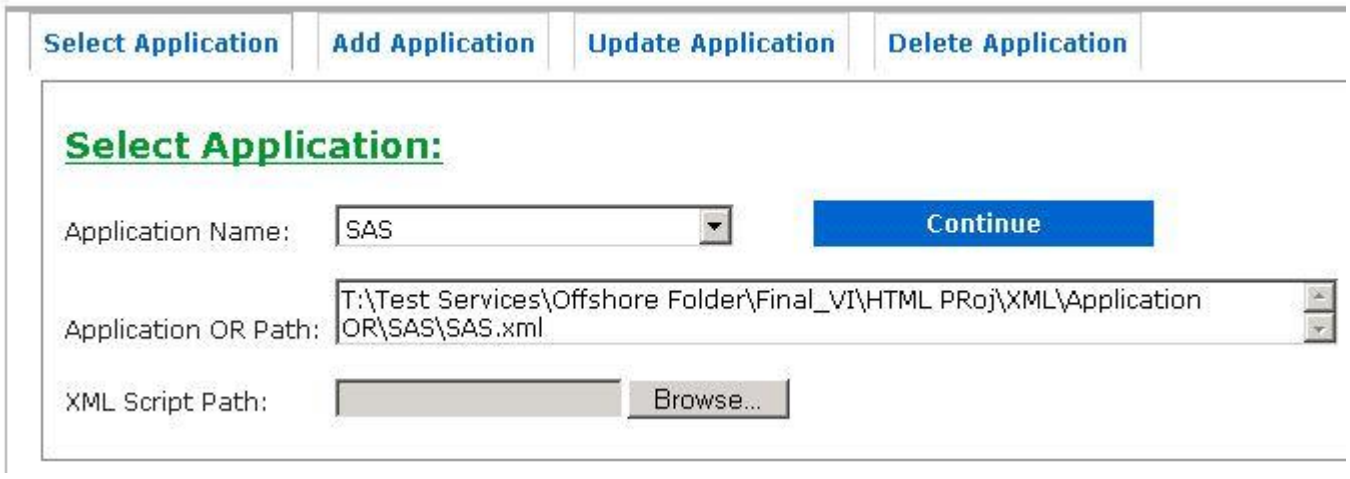
4.4.1 Adding Objects in Object Repository

Adding TestObject in object Repository
In this example, TestObject is WebEdit and its parent object is WebPage which is AEB_Page which in turn is child of Browser named AEB_Browser.
i.e. Hierarchy is like:

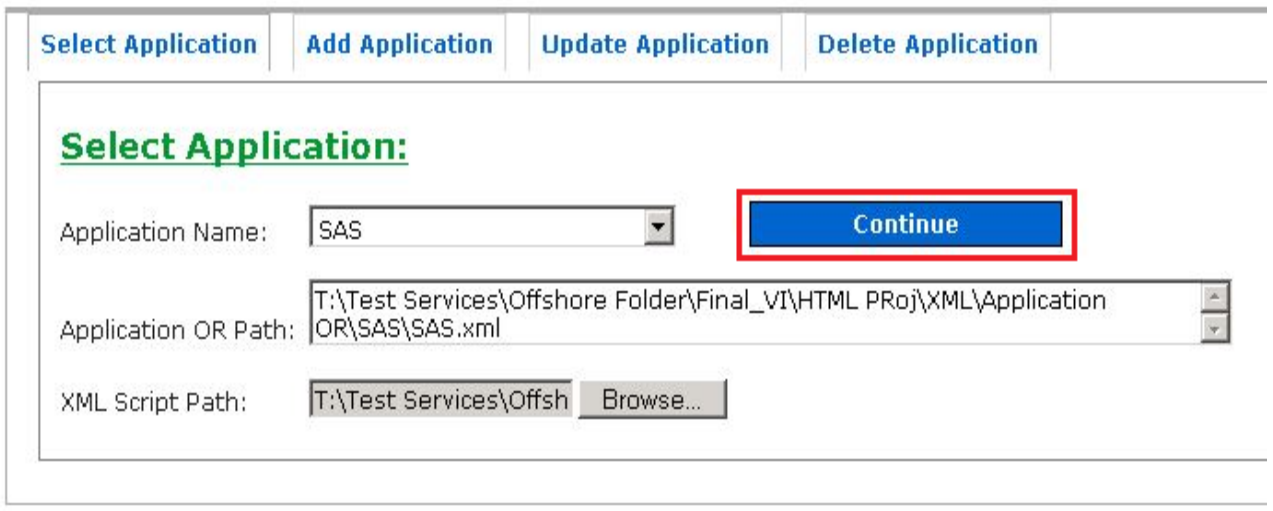
AEB_Browser().AEB_Page().TestObject()

Step 1: Select Application from the list.

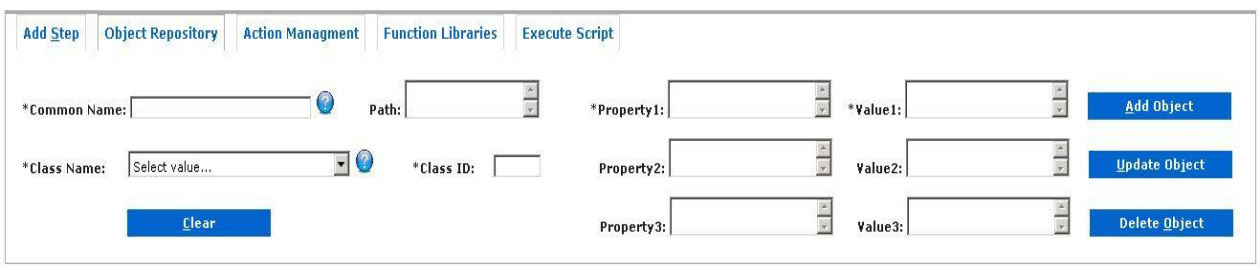
Example: SAS



Step 2: Click on "**Continue**" button.



Step 3: Click on "**Object Repository**" Tab.



Step 4: Enter Common Name; it should be unique and not used before.

Example: TestObject (WebEdit)

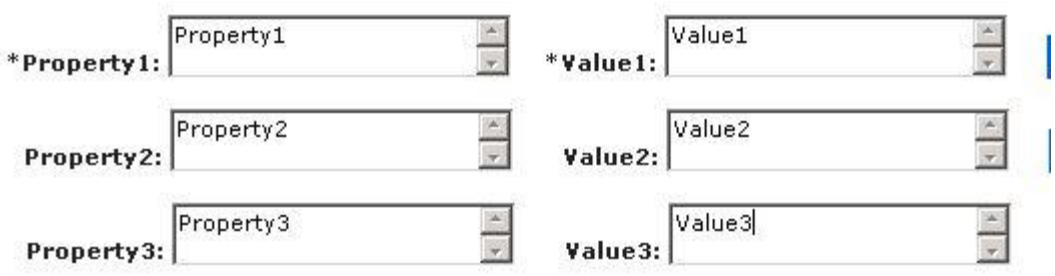
Enter Path. Hierarchy to parent objects.

Example: AEB_Browser.AEB_Page

Select Class Name, Class ID will be populated automatically.

Example: WedEdit

Step 5: Using QTP "**Object Spy**", get the object properties and add them.



Step 6: Click on "**Add Object**". Click on Ok Button on message box.

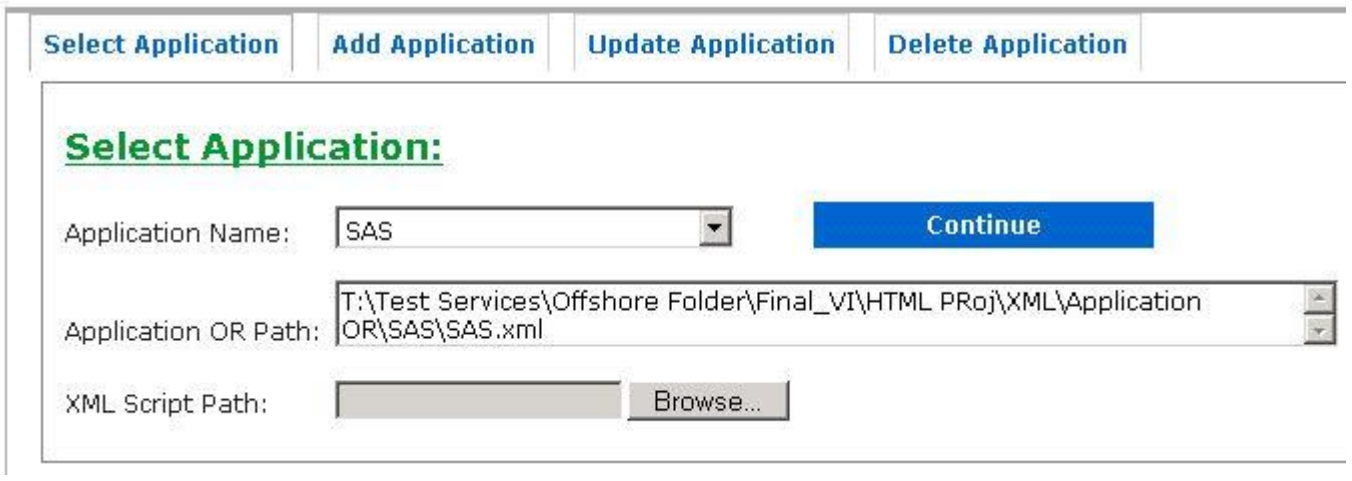
4.4.2 Editing Objects in Object Repository

Editing TestObject in object Repository
In this example, TestObject is WebEdit and its parent object is WebPage which is AEB_Page which in turn is child of Browser named AEB_Browser.
i.e. Hierarchy is like:

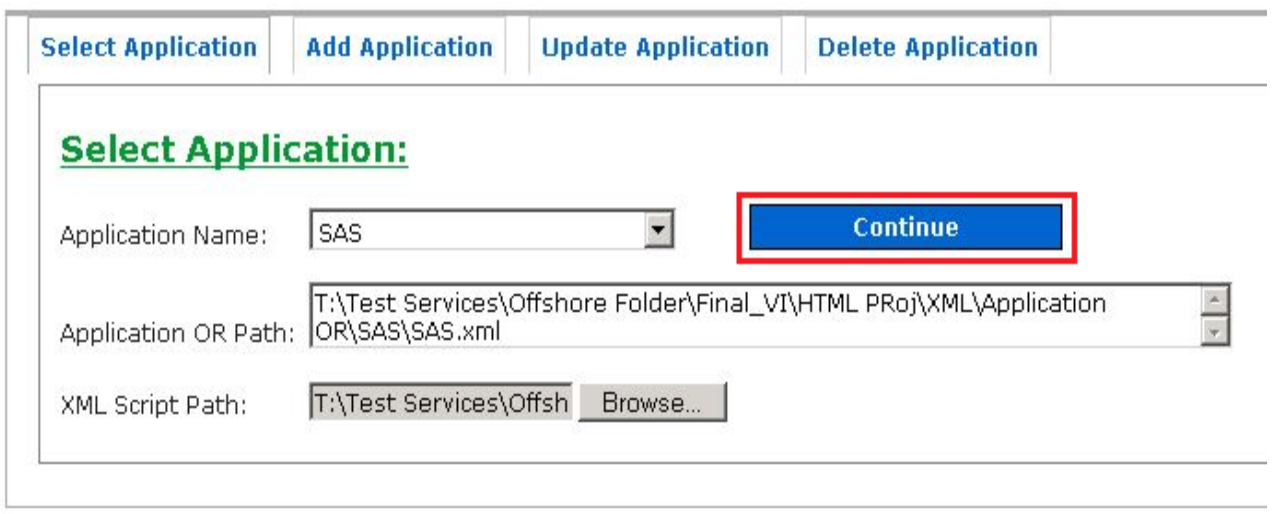
AEB_Browser().AEB_Page().TestObject()

Step 1: Select Application from the list.

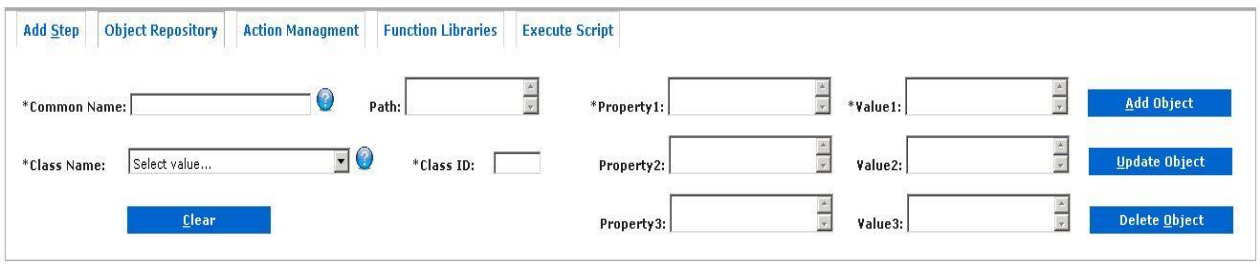
Example: SAS



Step 2: Click on "**Continue**" button.

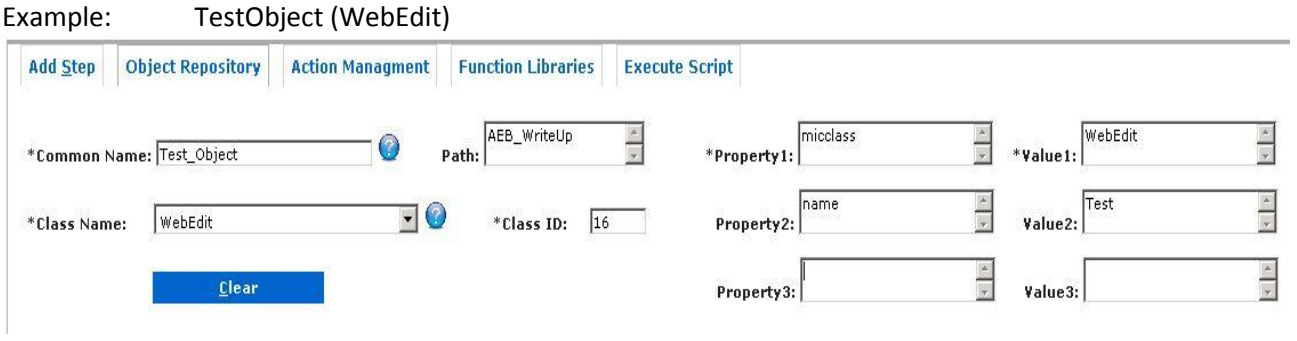


Step 3: Click on "**Object Repository**" Tab.

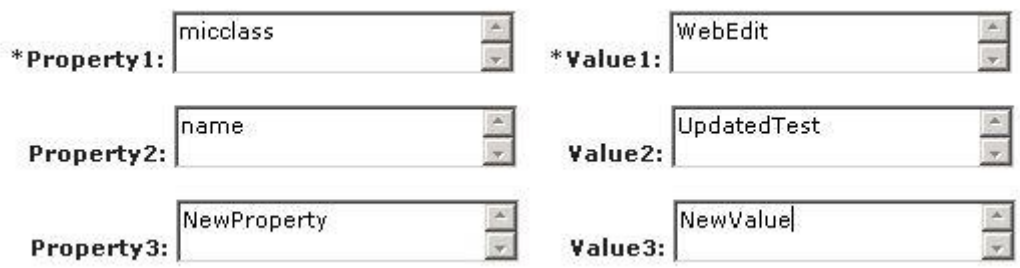


Step 4: Enter Common Name and press "**Tab**", this will load the properties

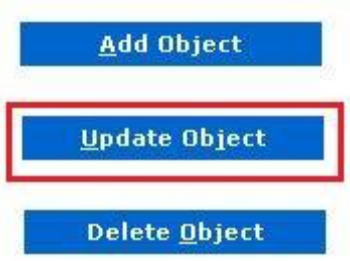
Example: TestObject (WebEdit)



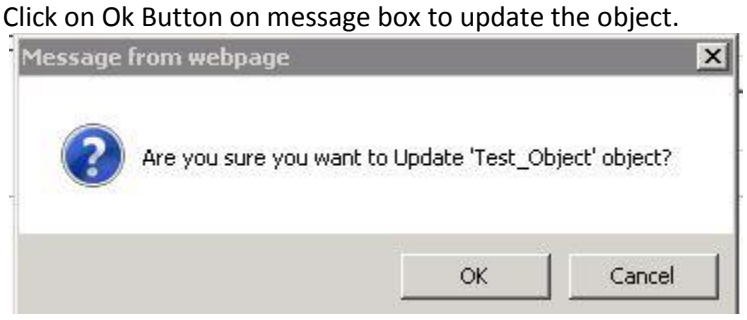
Step 5: Using QTP "**Object Spy**", get the object properties and update the required ones or add new property if available. User can also update Path and Class.



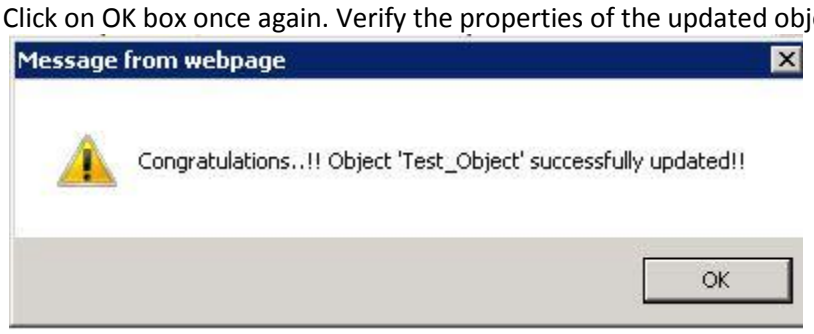
Step 6: Click on "**Update Object**".



Step 7: Click on Ok Button on message box to update the object.



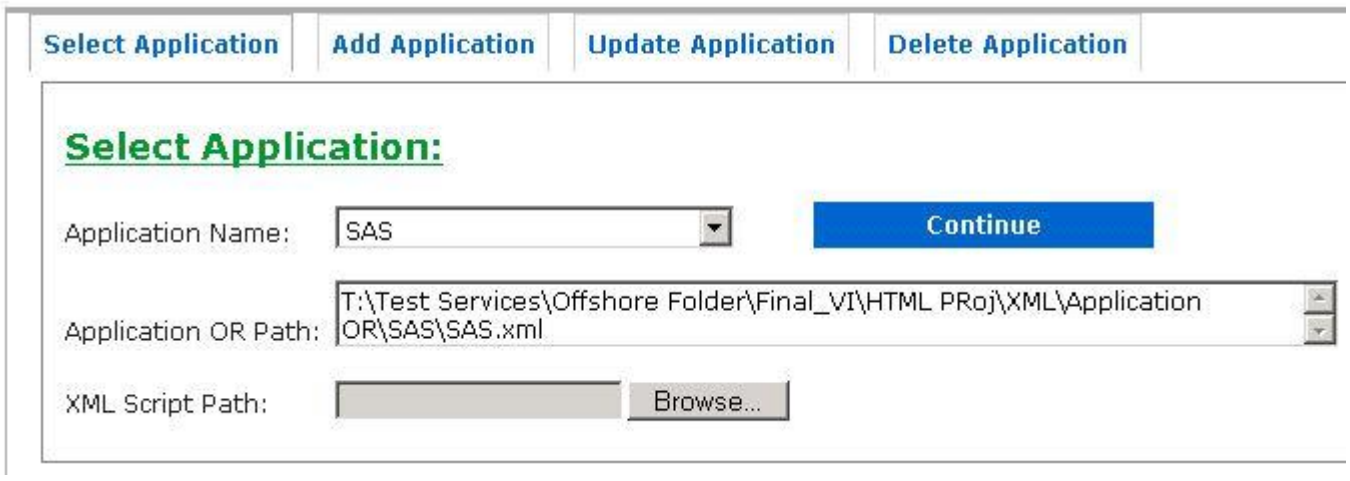
Step 8: Click on OK box once again. Verify the properties of the updated object.



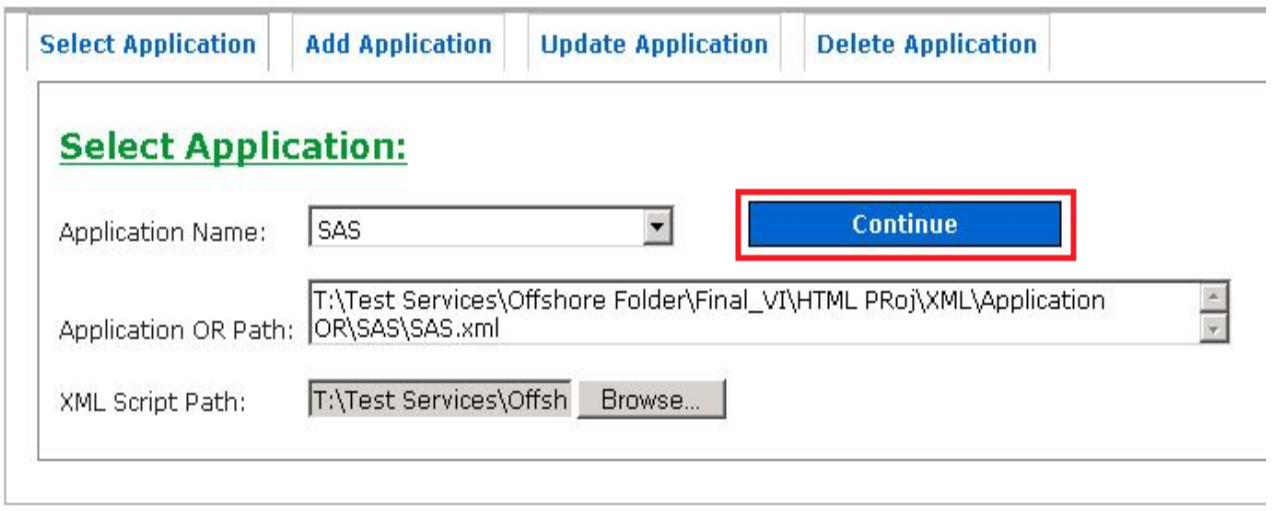
4.4.3 Deleting Objects in Object Repository

Deleting TestObject in object Repository

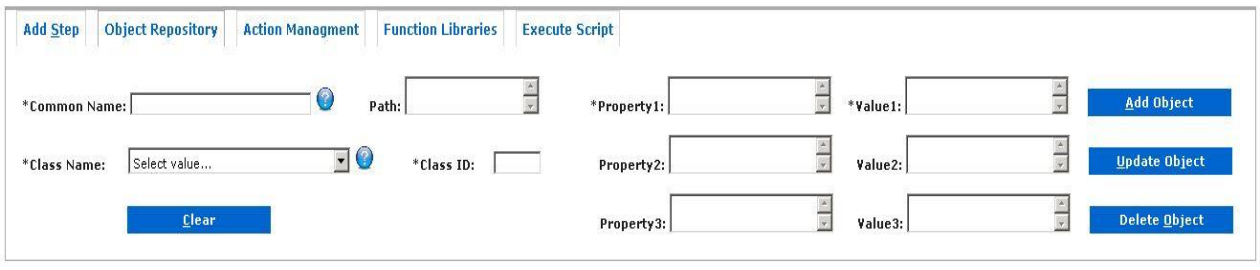
Step 1: Select Application from the list.
Example: SAS



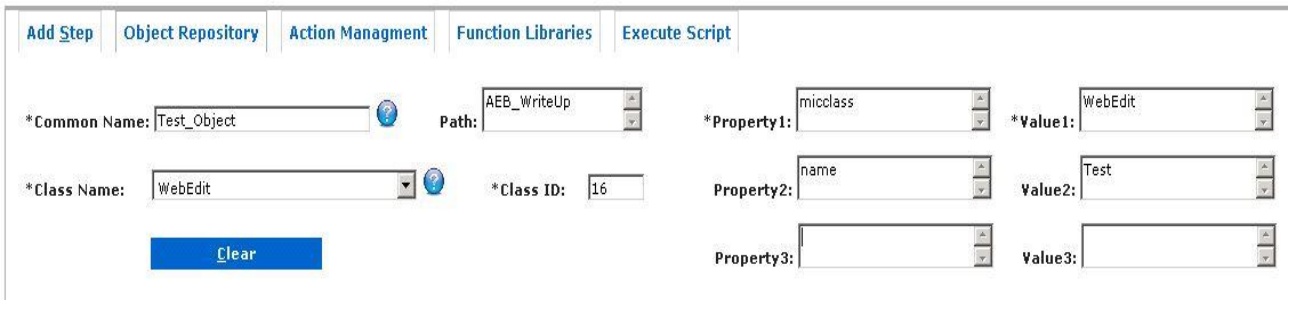
Step 2: Click on "**Continue**" button.



Step 3: Click on "**Object Repository**" Tab.

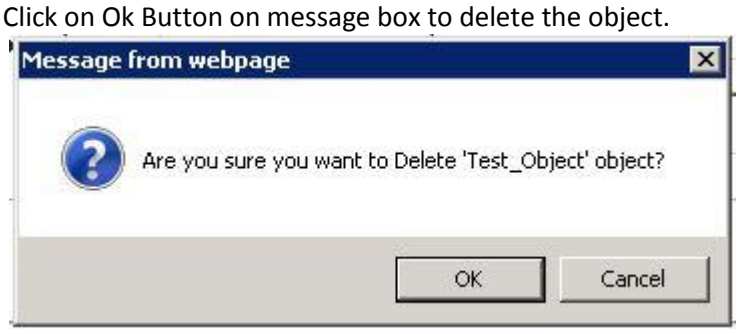


Step 4: Enter Common Name and press "**Tab**", this will load the properties
Example: TestObject (WebEdit)

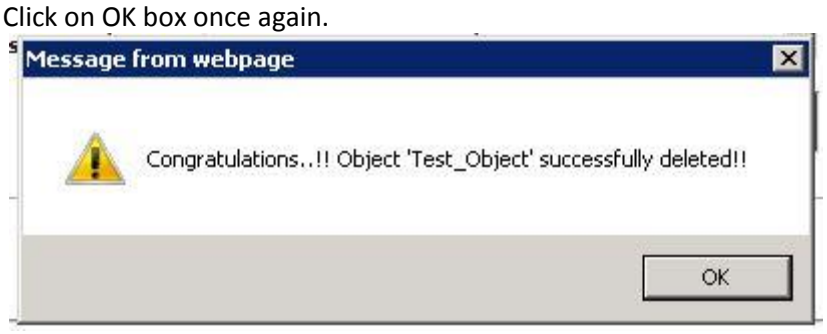


Step 5: Click on “**Delete Object**”.

Step 6: Click on Ok Button on message box to delete the object.



Step 7: Click on OK box once again.



4.5 Action Management

This tab is used to manage actions.

Whenever new keyword is added in the function library (in QTP) we need to add the corresponding action name also. This task is achieved from Action Management tab.

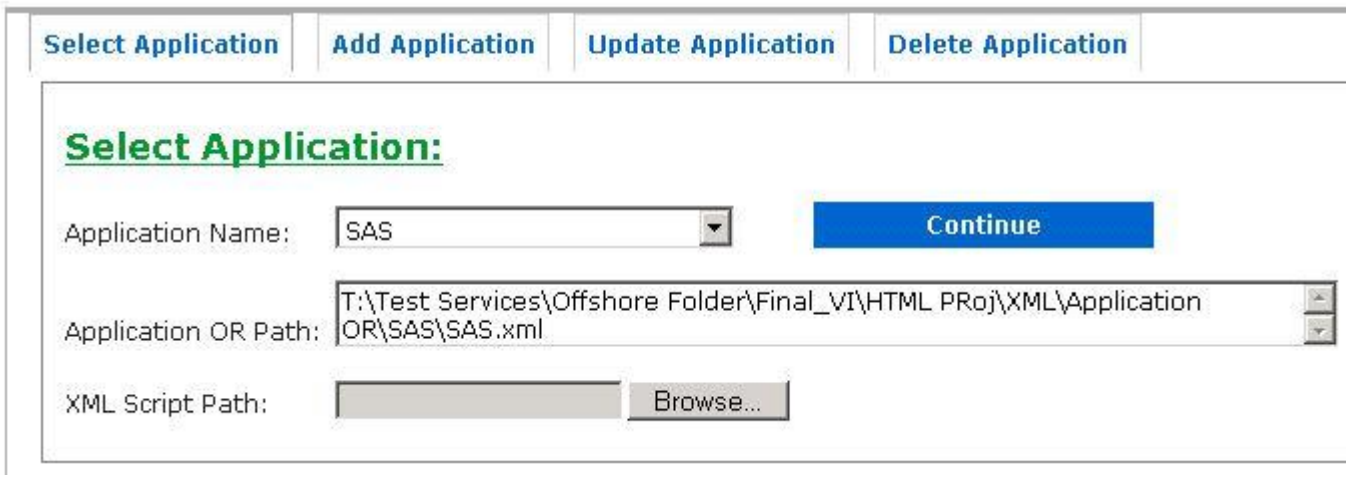
Also , sometimes we also need to update or delete the action, same thing can also be done from Action Management tab.

4.5.1 Adding Actions in Step Generator

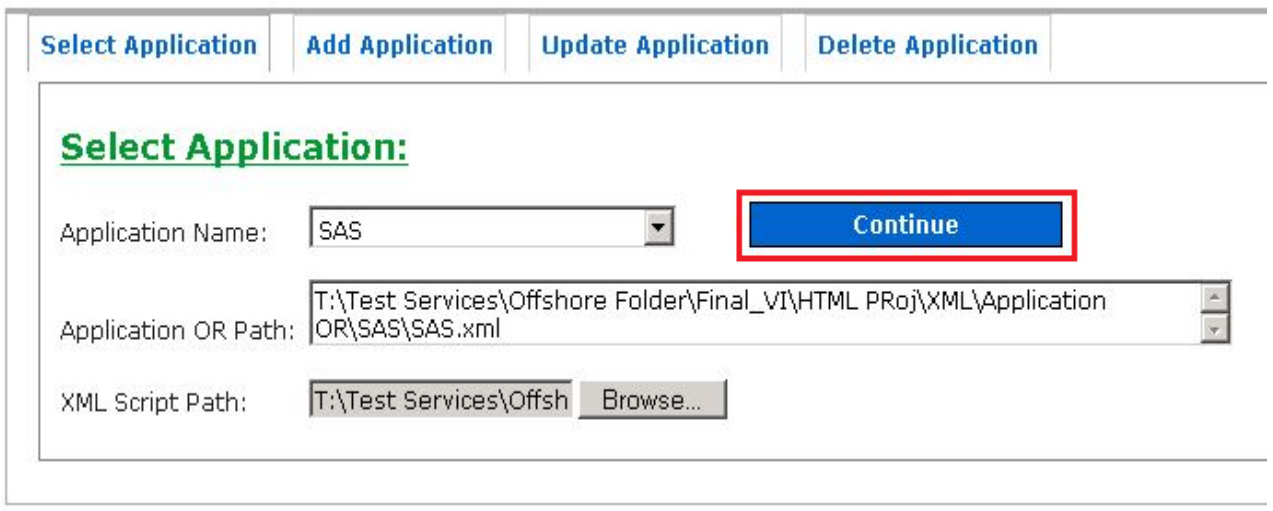
Below are the steps for adding action in the utility.

Step 1: Select Application from the list.

Example: SAS



Step 2: Click on "**Continue**" button.



Step 3: Click on "**Action Management**" Tab.

Step 4: Enter the name of the action.

Example: "Test_Action"

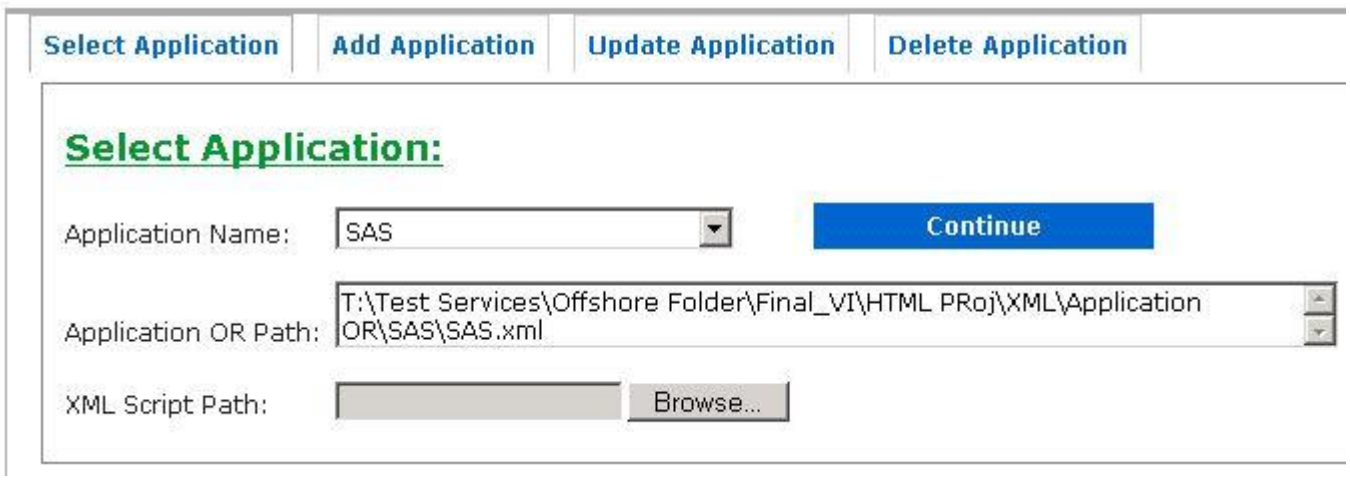
Step 5: Click on "**Add Action**" Button. Click on "**OK**" button on the alert box.

4.5.2 Editing Actions in Step Generator

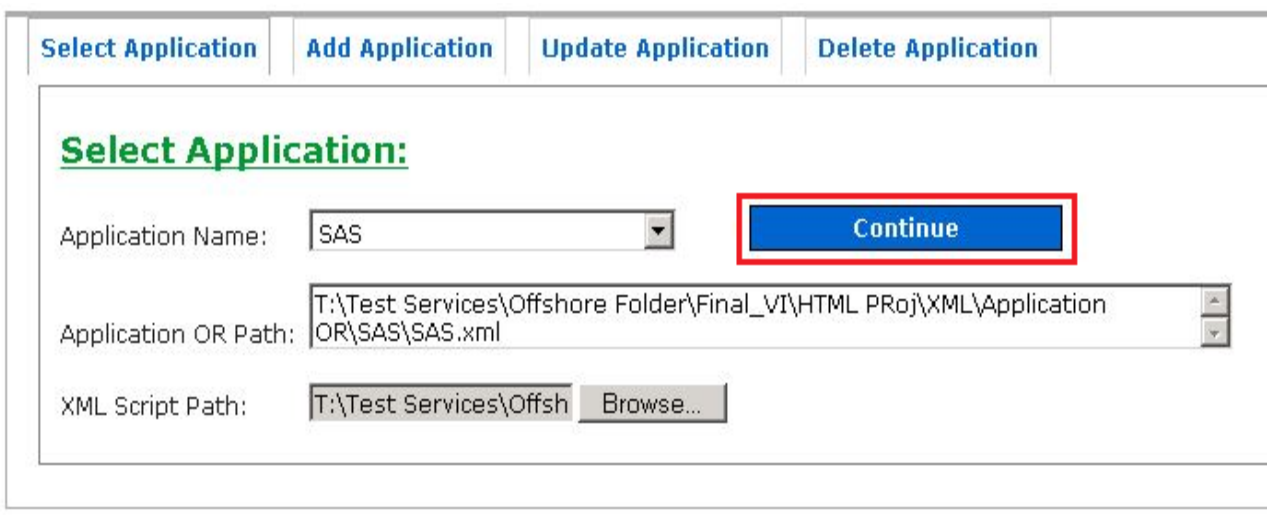
Below are the steps for editing action in the utility.

Step 1: Select Application from the list.

Example: SAS



Step 2: Click on "**Continue**" button.

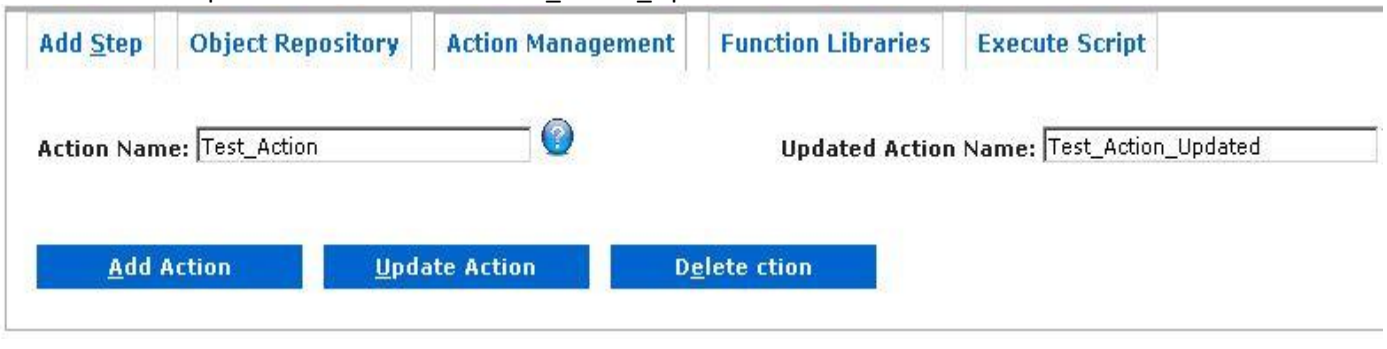


Step 3: Click on "**Action Management**" Tab.

Step 4: Enter the name of the action and **Updated Action Name** to change the name.

Example: Action Name: "Test_Action"

Updated Action Name: "Test_Action_Updated"



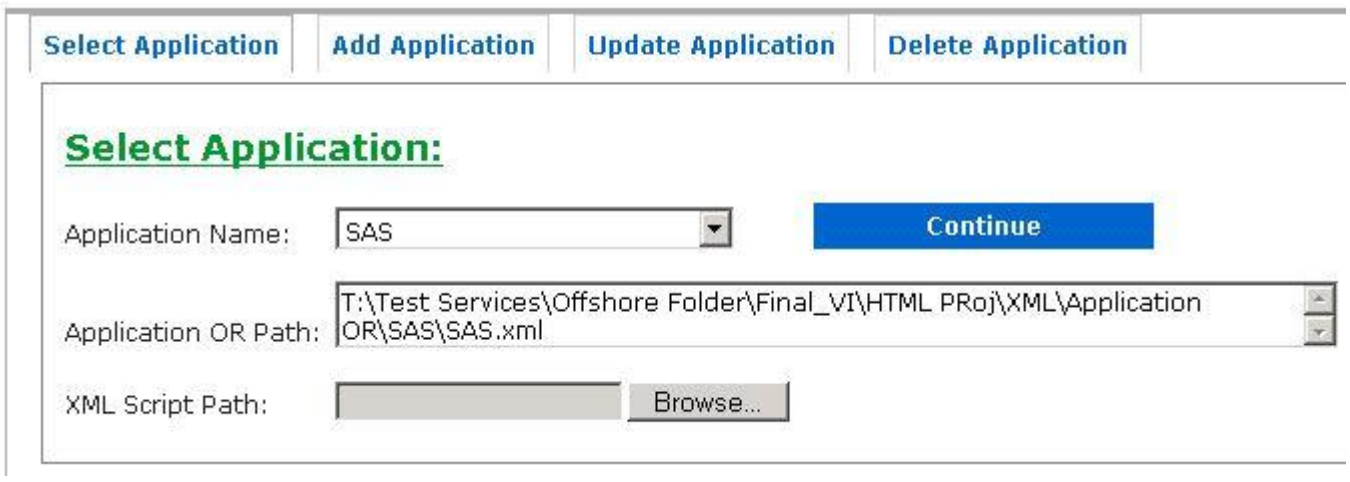
Step 5: Click on “**Update Action**” Button. Click on “**OK**” button on the alert box.

4.5.3 Deleting Actions in Step Generator

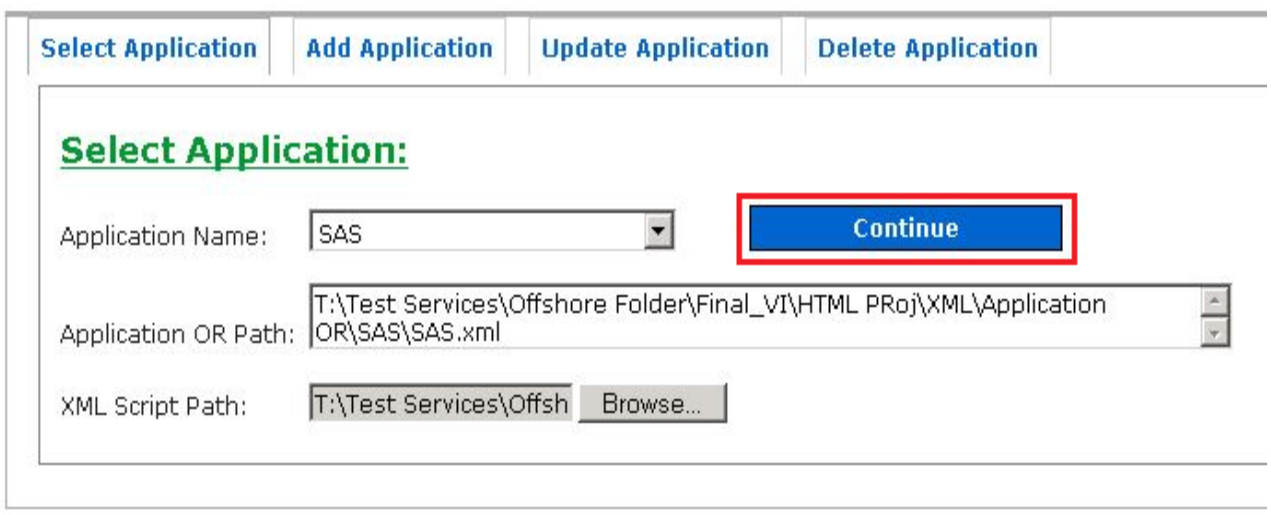
Below are the steps for deleting action in the utility.

Step 1: Select Application from the list.

Example: SAS



Step 2: Click on "**Continue**" button.



Step 3: Click on "**Action Management**" Tab.

Step 4: Enter the name of the action to delete.

Example: Action Name: "Test_Action_Updated"

Step 5: Click on “**Delete Action**” Button. Click on “**OK**” button on the alert box.

4.6 Function Library Management

This tab is used to manage Function Library.

Whenever new Function Library is added (in QTP) we need to add the corresponding Function Library name also. This task is achieved from Function Libraries tab.

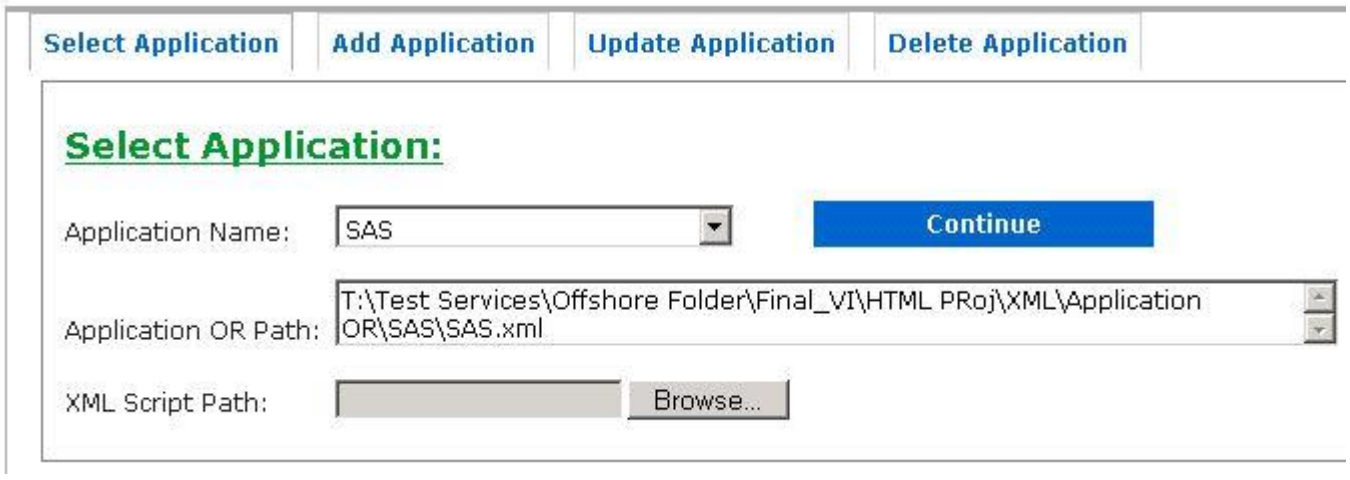
Also , sometimes we also need to update or delete the Function Library, same thing can also be done from Function Libraries tab.

4.6.1 Adding Function Library in Step Generator

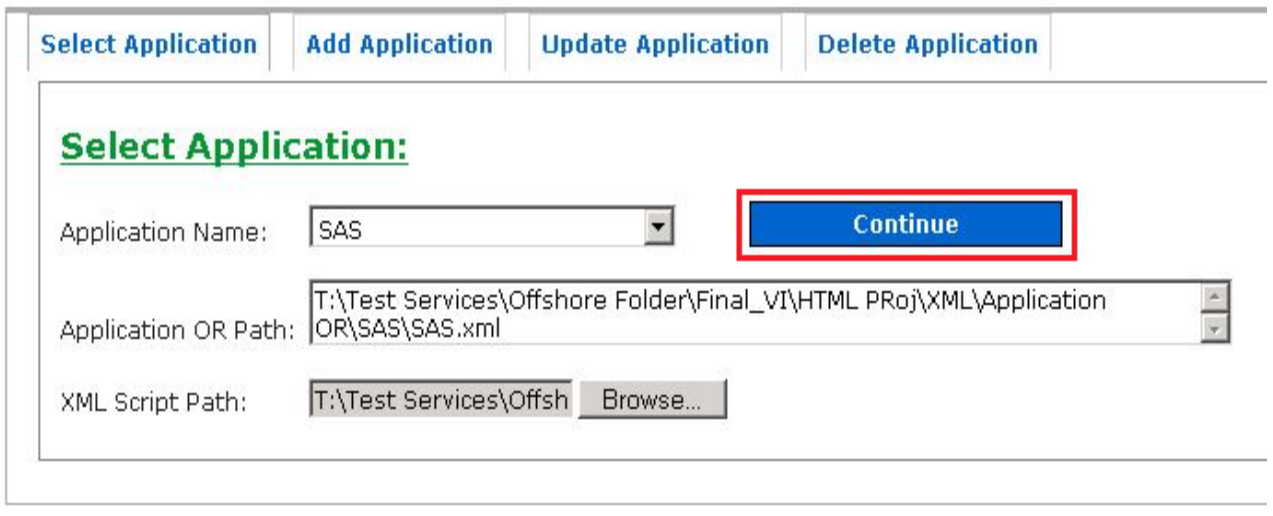
Below are the steps for adding Function Library in the utility.

Step 1: Select Application from the list.

Example: SAS



Step 2: Click on "**Continue**" button.



Step 3: Click on "**Function Libraries**" Tab.

Step 4: Enter the name of the Function Library.

Example: Function Library: “Test_Func_Lib”

Function Library Path: “T:\Test.qfl”

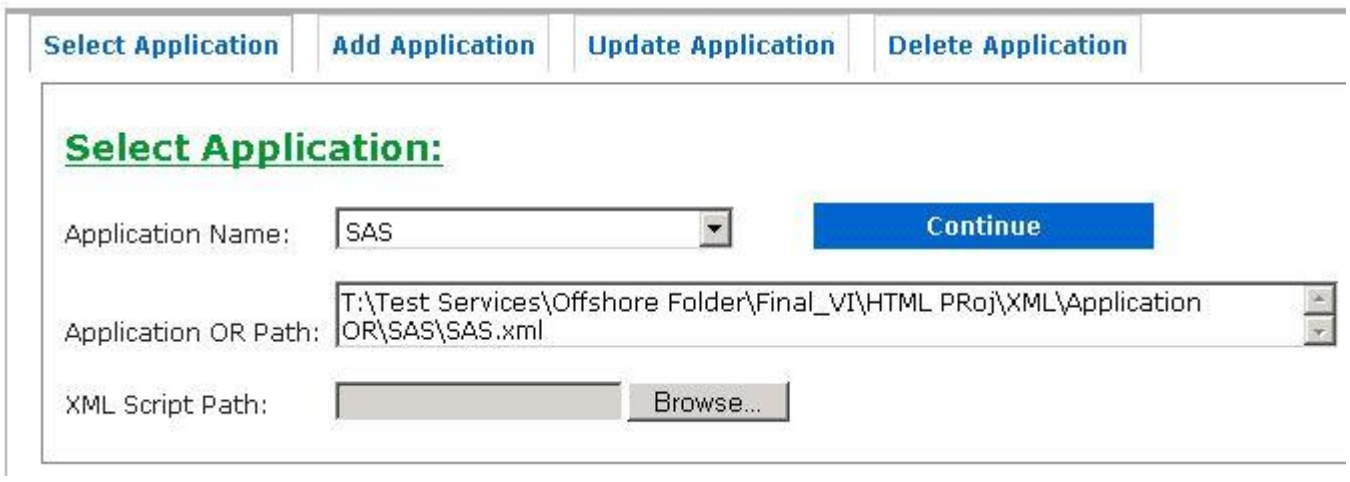
Step 5: Click on “**Add Function Library**” Button. Click on “**OK**” button on the alert box.

4.6.2 Editing Function Library path in Step Generator

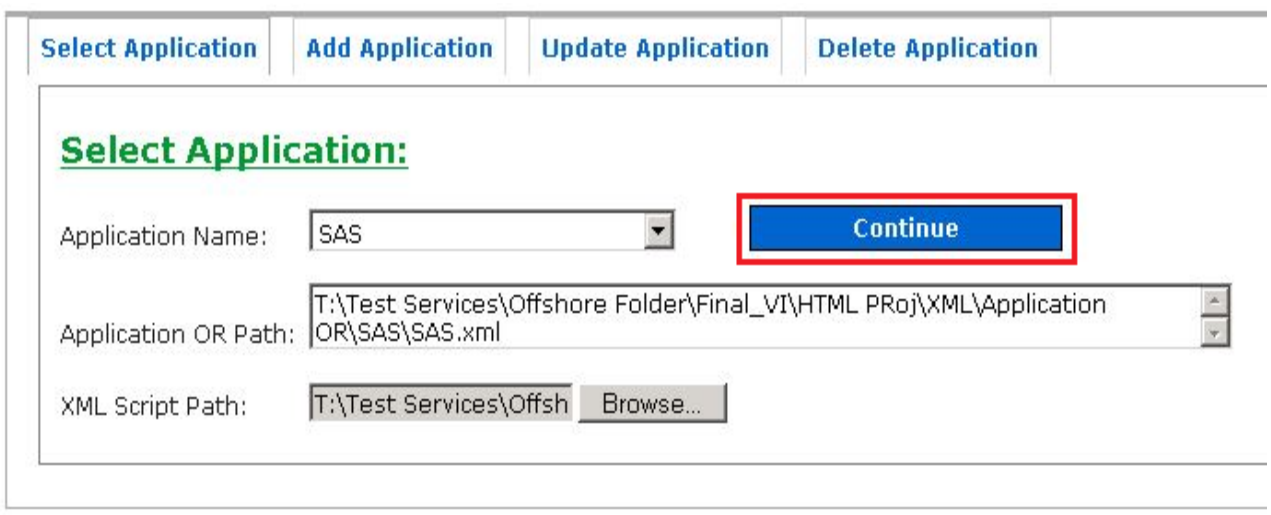
Below are the steps for editing action in the utility.

Step 1: Select Application from the list.

Example: SAS



Step 2: Click on "**Continue**" button.



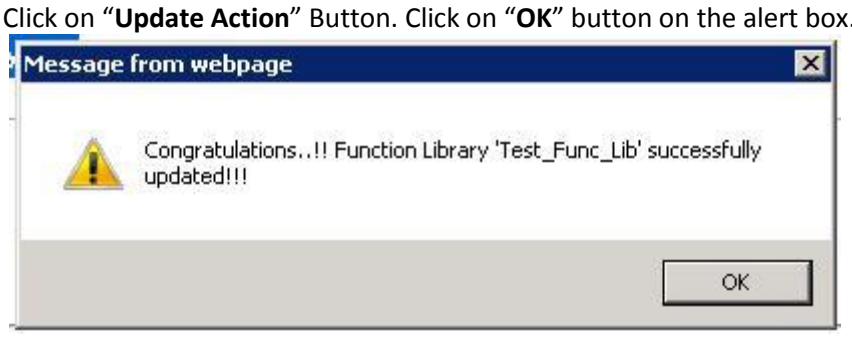
Step 3: Click on "**Function Libraries**" Tab.

Step 4: Select Function Name and Enter **Updated** Action Name to change the path.

Example: Function Library: "Test_Action"

Function Library Path: "T:\Test_Updated_Path.qfl"

Step 5: Click on “**Update Action**” Button. Click on “**OK**” button on the alert box.

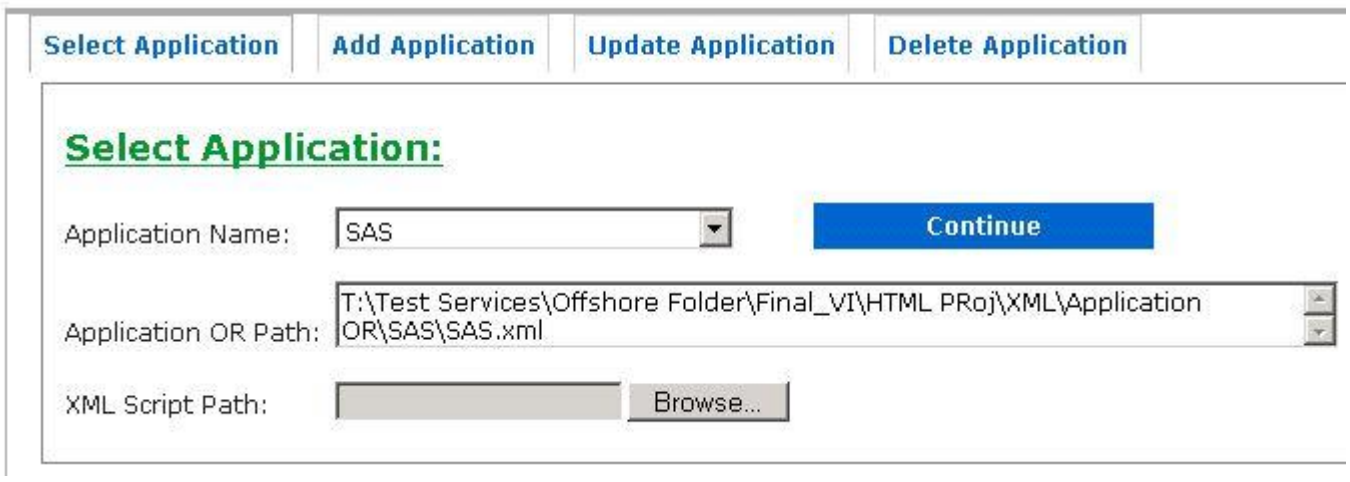


4.6.3 Deleting Function Library in Step Generator

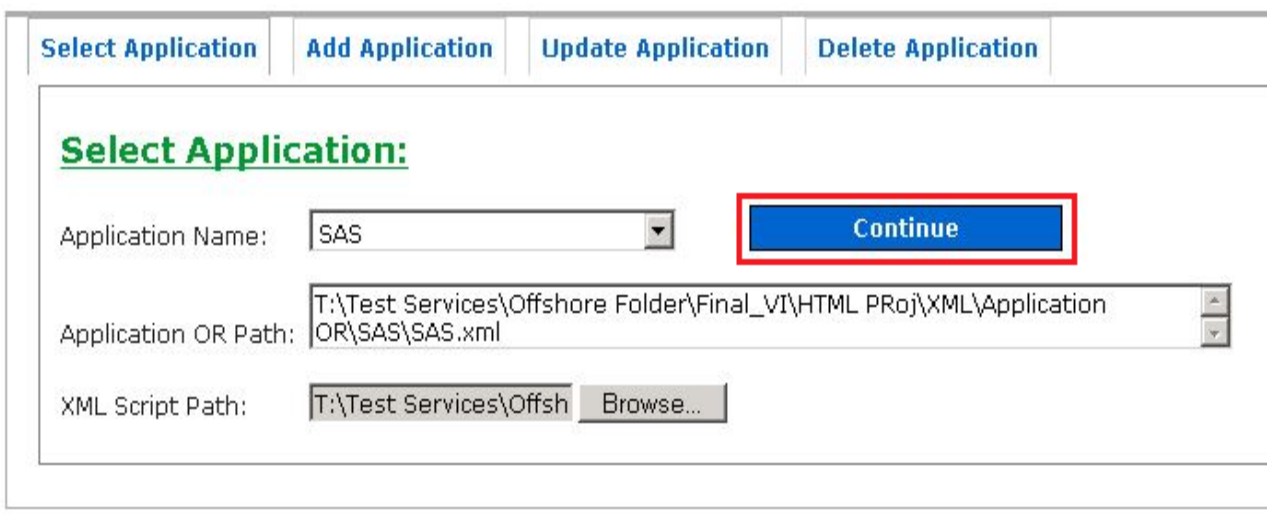
Below are the steps for deleting Function Library in the utility.

Step 1: Select Application from the list.

Example: SAS



Step 2: Click on "**Continue**" button.

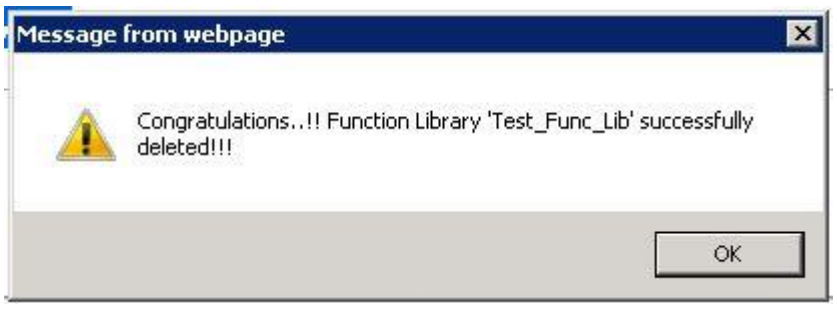


Step 3: Click on "**Function Libraries**" Tab.

Step 4: Enter the name of the Function Library to delete.

Example: Function Library: "Test_Func_Lib"

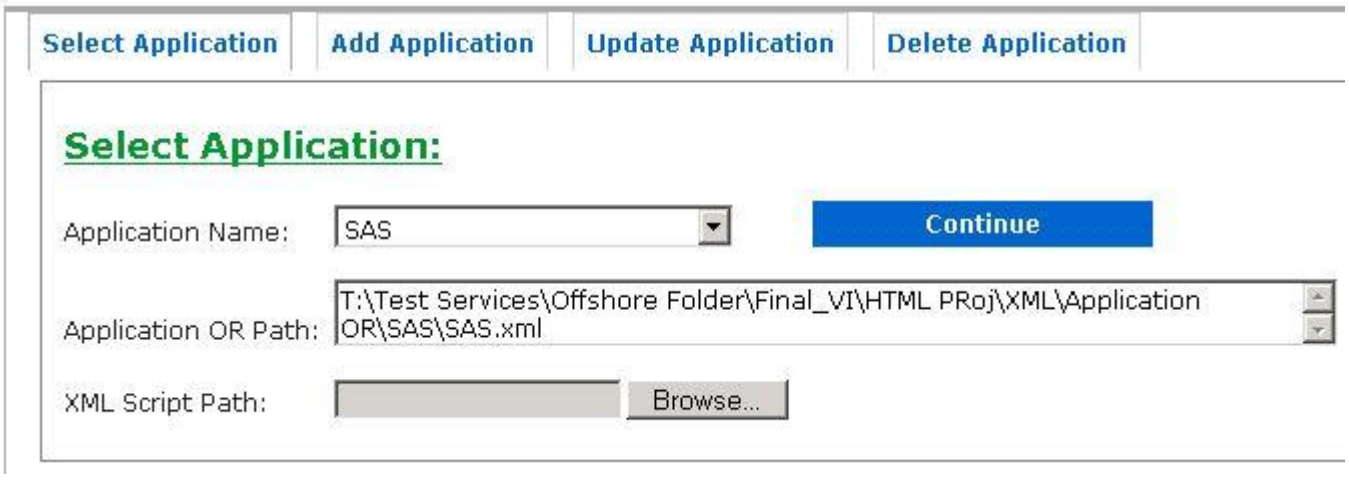
Step 5: Click on “**Delete Action**” Button. Click on “**OK**” button on the alert box.



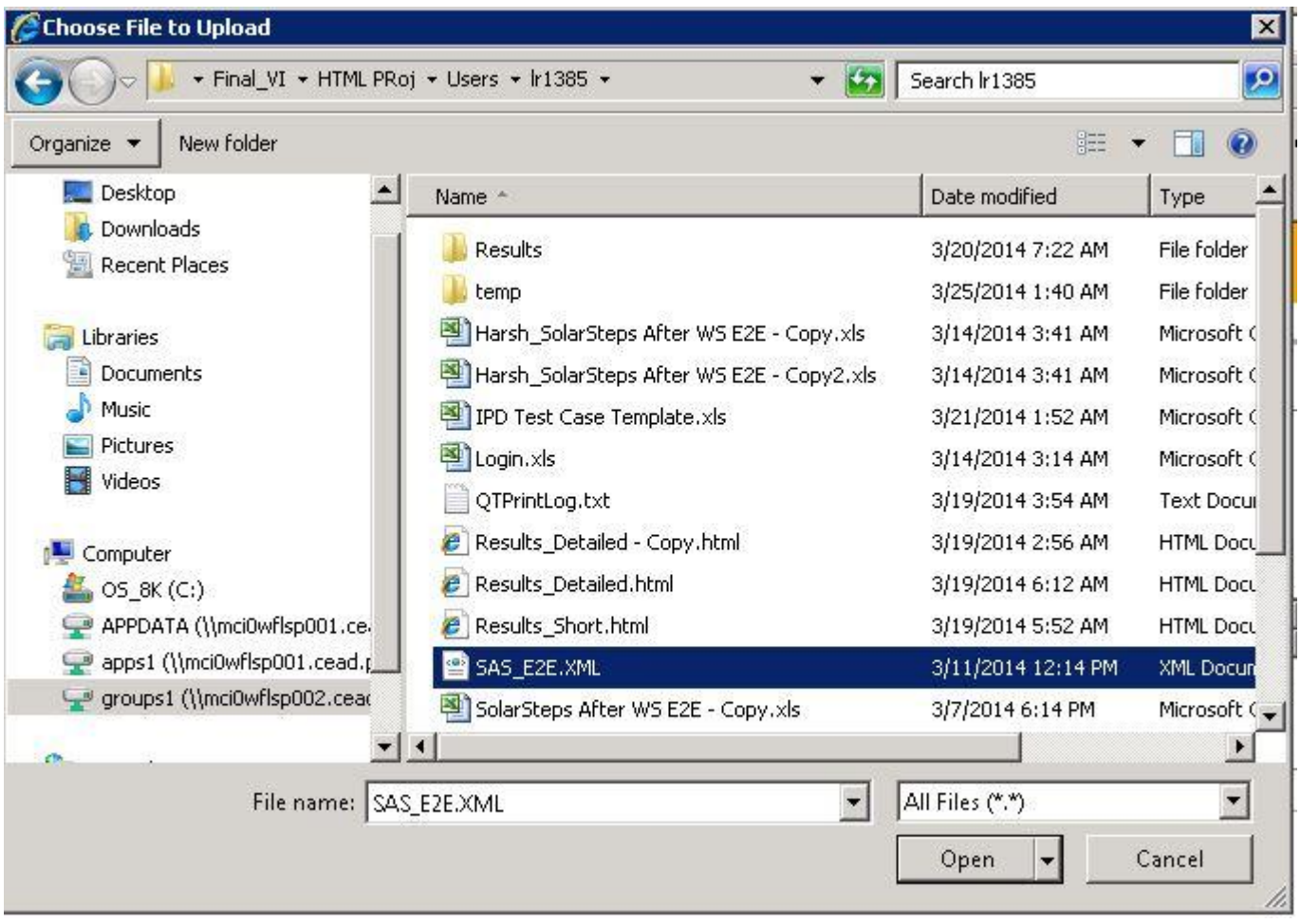
4.7 Selecting existing Test Case (.xml)

Step 1: Select Application from the list.

Example: SAS



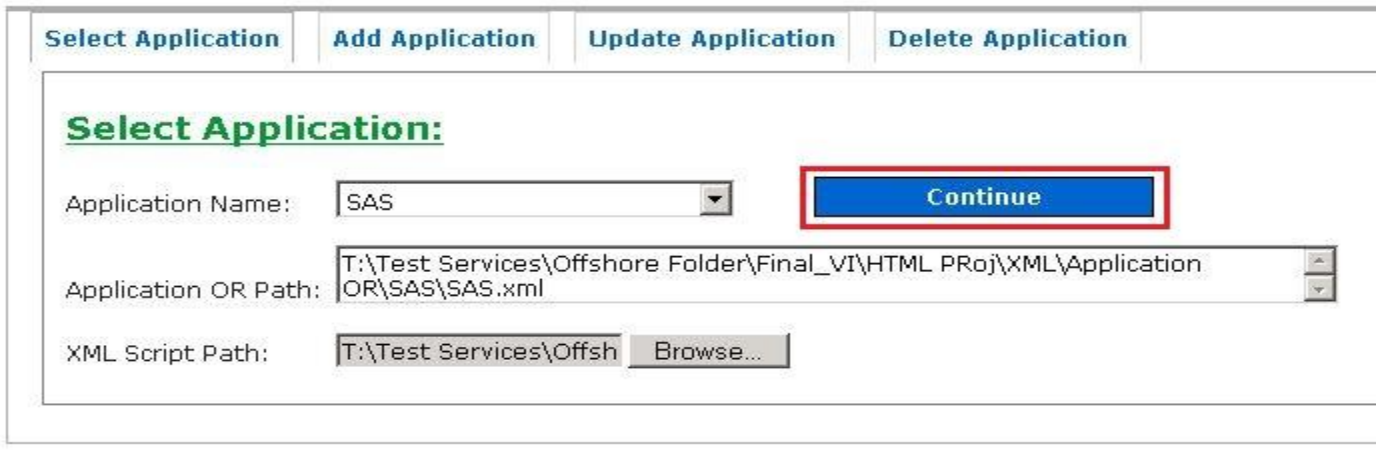
Step 2: Click on Browse and select the xml file.



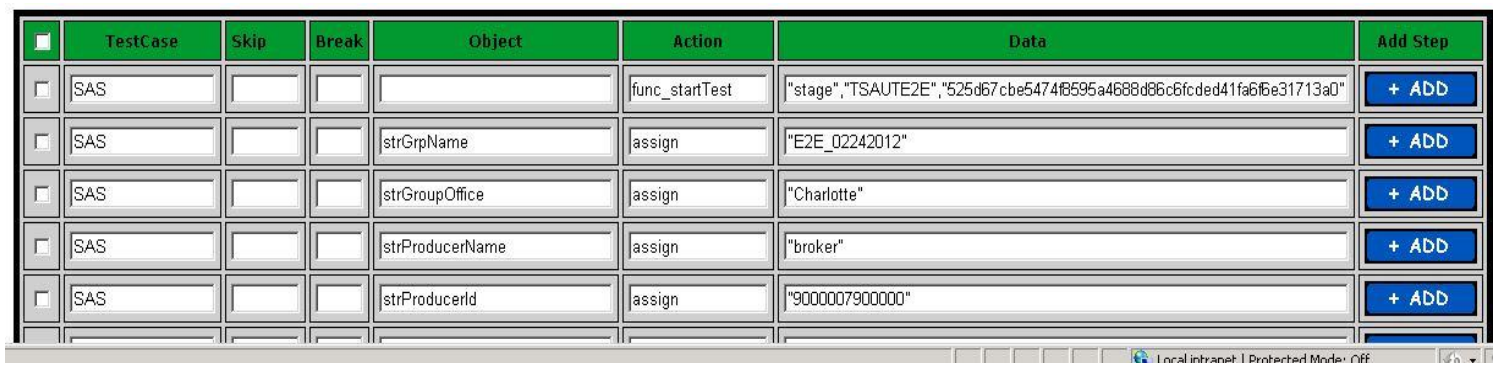
Example: SAS_E2E.XML

NOTE: Only *.xml files are allowed to select.

Step 3: Click on "**Continue**" button.



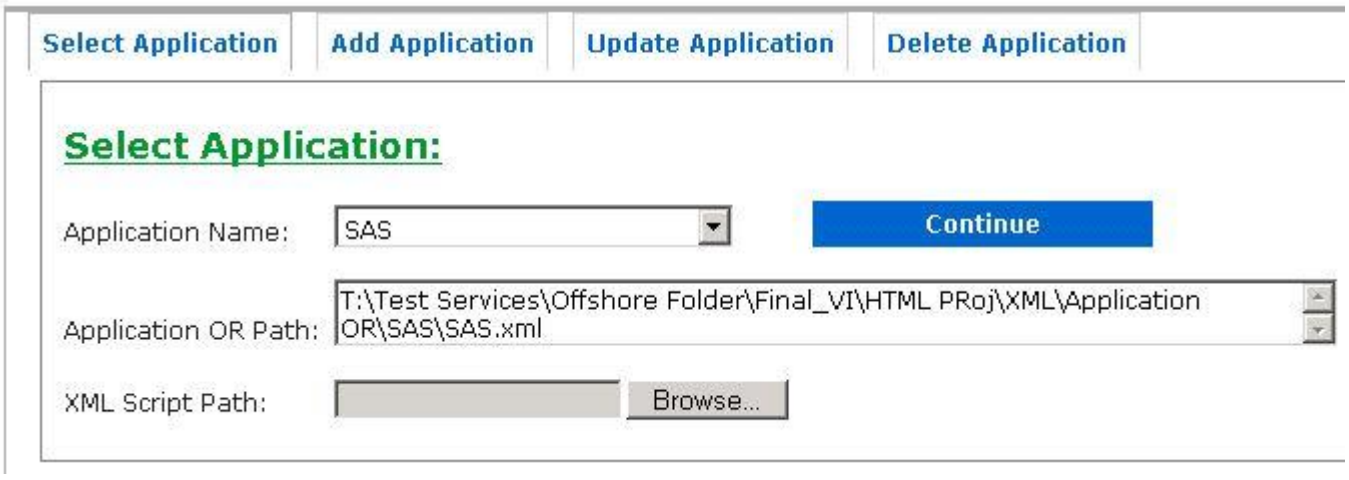
Step 4: Verify Table in the bottom section of the **Framework Utility** Page.



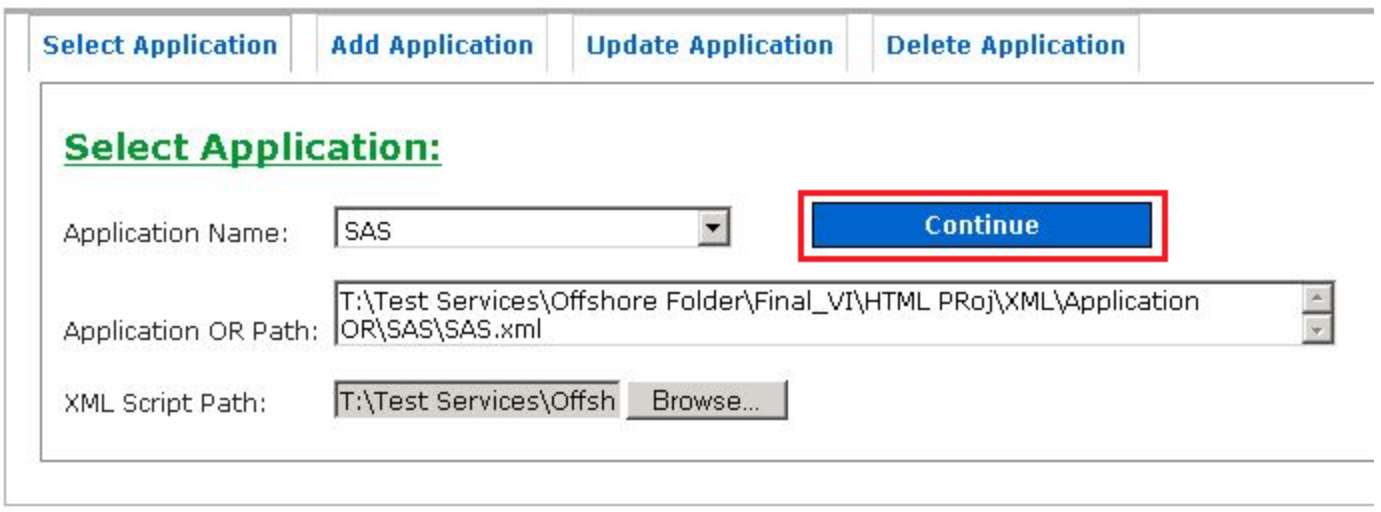
4.8 Starting with new Test Case

Step 1: Select Application from the list.

Example: SAS



Step 2: Click on "**Continue**" button.



4.9 Add Step Tab

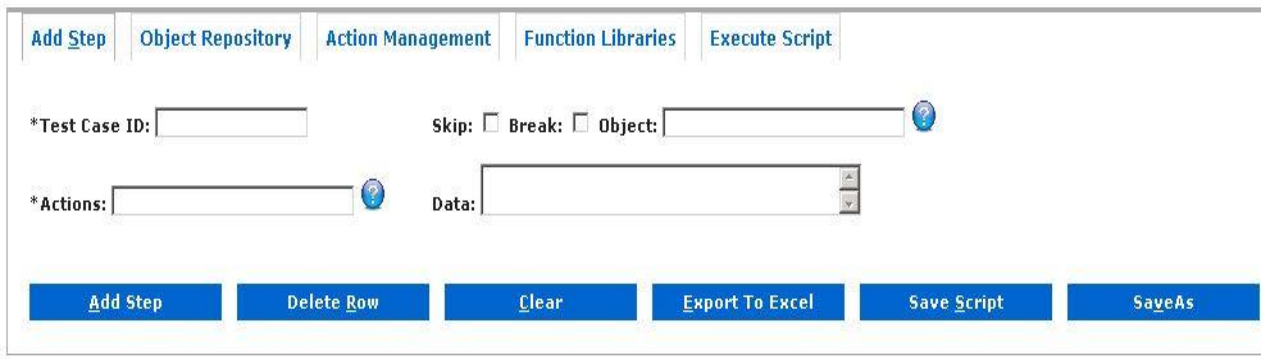
4.9.1 Writing Test Case

There are two ways of writing test cases using Step Generator

1. Using the GUI available in the top section.
2. Directly writing in the web-table.

4.9.1.1 Using the GUI

Understanding GUI:

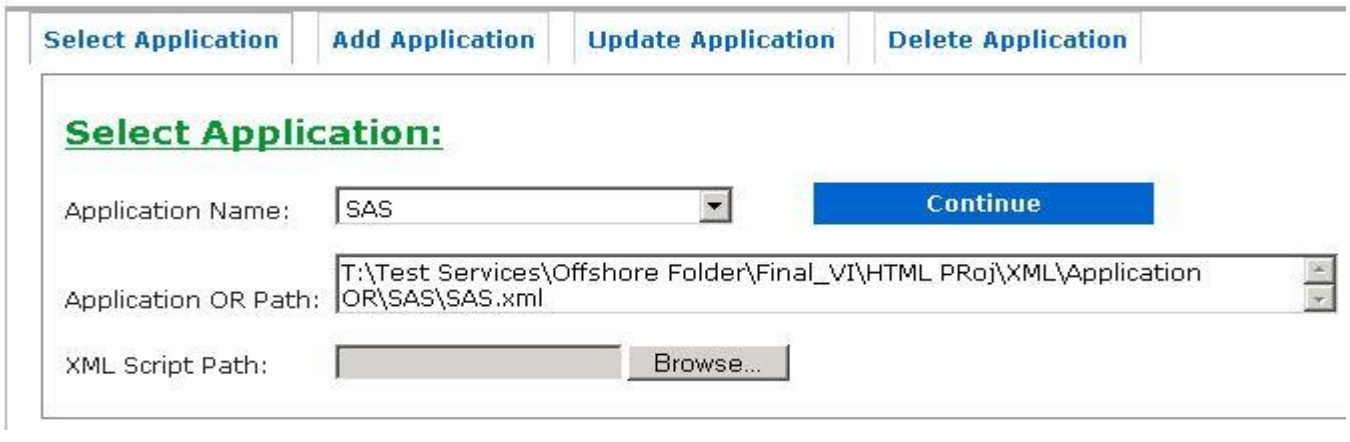


Field Name	Description
Test Case ID	Mandatory field. Test Case ID of the test Script Example "**TC01**"
Skip	Check Box. Check to skip the step.
Break	Check Box. Check to have a break during execution on the step (For debugging).
Object	Textbox. Object Name. Fetched from object repository.
Actions	Mandatory field. Action Name or "**keyword**".
Data	Textbox. For entering data of the step.

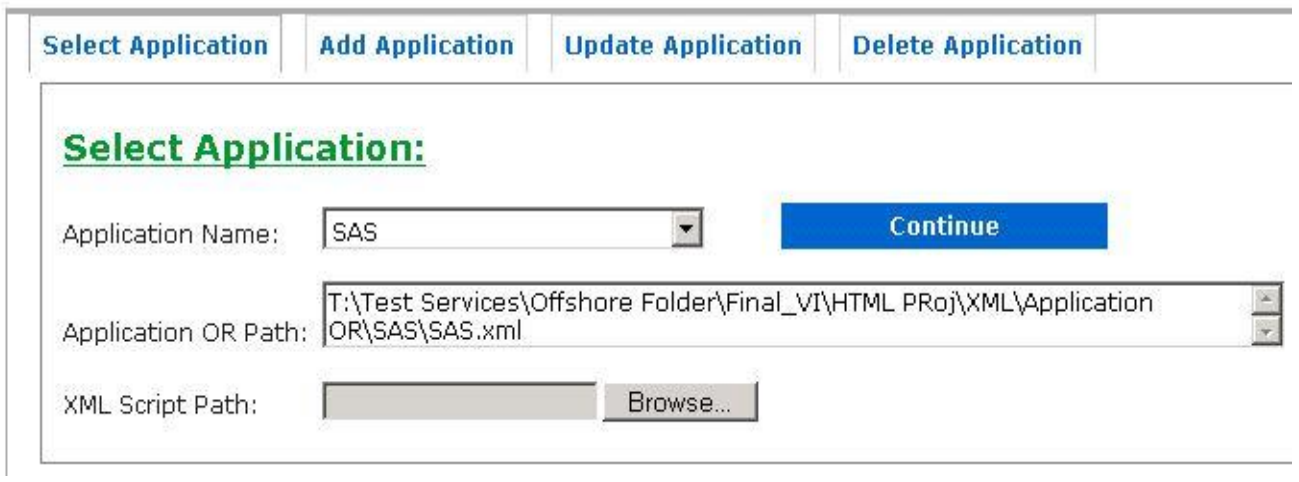
Below are the steps for writing test case.

Step 1: Select Application from the list.

Example: SAS



Step 2: Click on "**Continue**" button.



Step 3: Enter details in Add Step Tab for

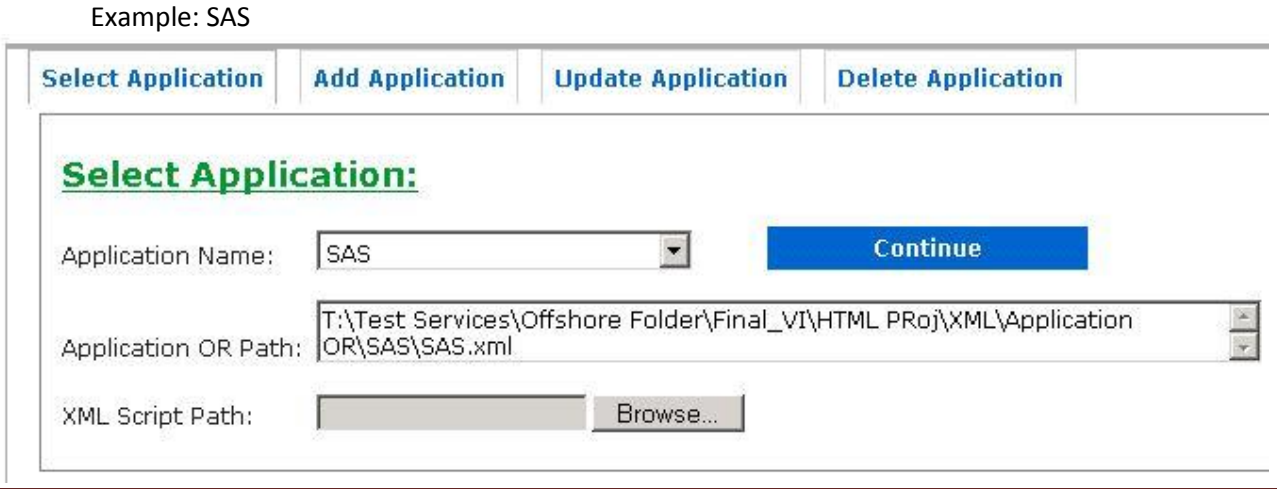
- Test Case ID : "**TC01**"
- Skip: "**Uncheck**"
- Break: "**Check**"
- Object: "**AEB_WriteUp**"
- Action: "**msgbox**"
- Data: "**This is a test step**"

Step 4: Click on "**Add Step**" button. Verify Step in the Table below the tab.

4.9.1.2 Using the Web-Table

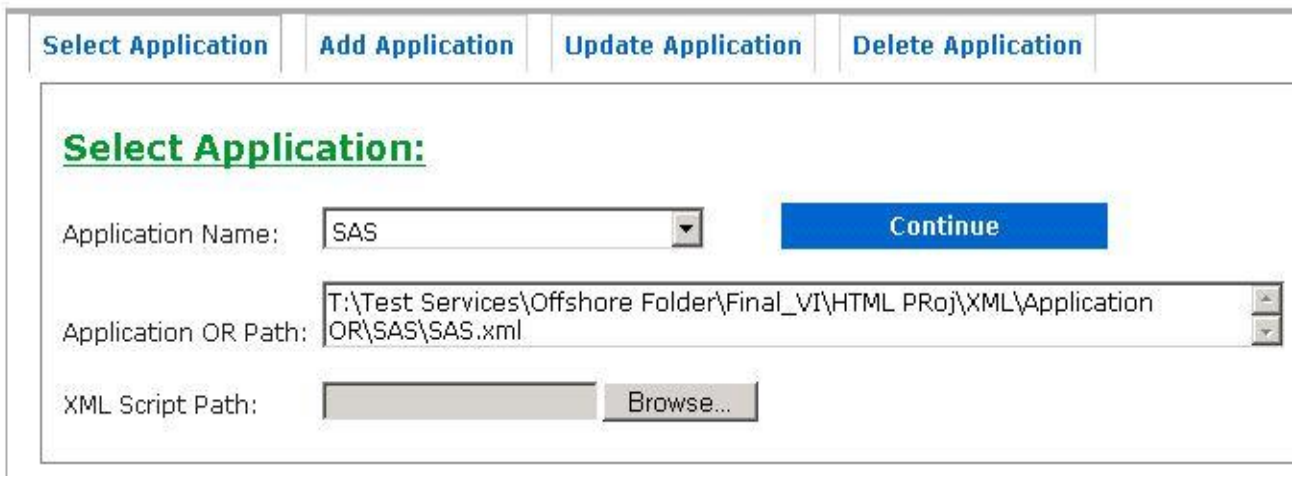
We can directly use the web table (Only after first step is added from the GUI).
Below are the steps for writing test case.

Step 1: Select Application from the list.
Example: SAS



Back to Top…

Step 2: Click on "**Continue**" button.



Step 3: Enter details in Add Step Tab for

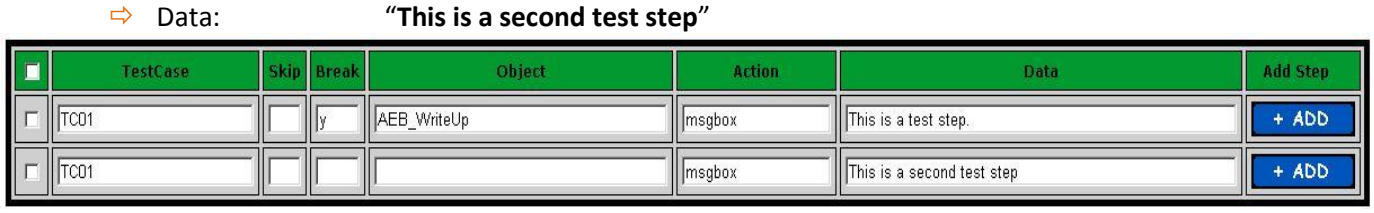
- Test Case ID : "**TC01**"
- Skip: "**Uncheck**"
- Break: "**Check**"
- Object: "**AEB_WriteUp**"
- Action: "**msgbox**"
- Data: "**This is a test step**"

Step 4: Click on "**Add Step**" button. Verify Step in the Table below the tab.

Step 5: Click on "**+ Add**" button in the last column of the Web-Table. Test Case will automatically be populated.

Step 6: Enter the details same as required.

- Test Case ID : "**TC01**"
- Skip: "**Uncheck**"
- Break: "**Uncheck**"
- Object: "**AEB_WriteUp**"
- Action: "**msgbox**"
- Data: "**This is a second test step**"

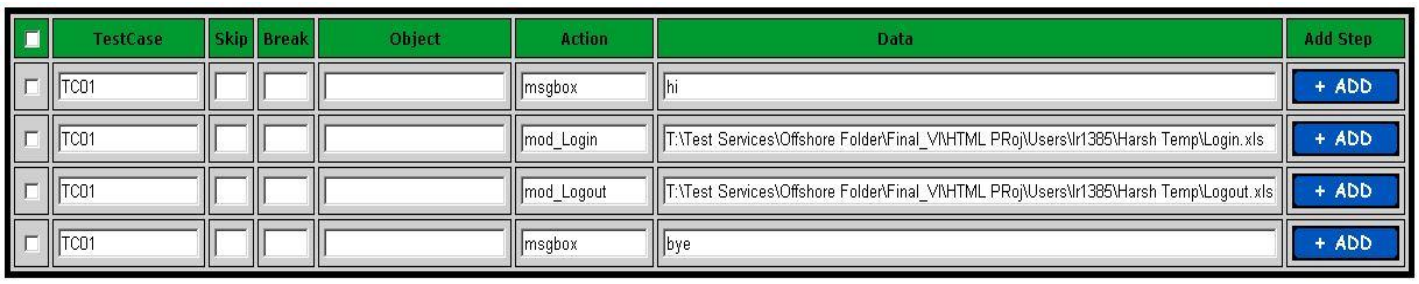


4.9.1.3 Re-Usable Scripts

Step Generator can also handle Re-usable scripts.
Syntax: **mod_**
Use mod_ in action to along with some prefix (for eg: mod_Test1, mod_Test2 etc.)
In Data column enter the path of the xls file you want to use.

For example: If Login is the file (**Location**: T:\Test Services\Offshore Folder\Final_VI\HTML PRoj\Users\lr1385\Harsh Temp\Login.xls) which we want to use in our Test Script then we will write:
Action: mod_Login
Data: T:\Test Services\Offshore Folder\Final_VI\HTML PRoj\Users\lr1385\Harsh Temp\Login.xls

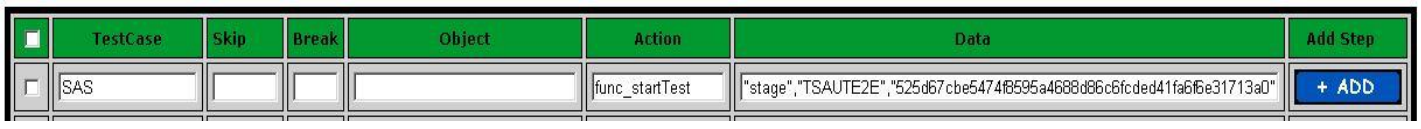
Similarly if we want to use Logout File (**Location**: T:\Test Services\Offshore Folder\Final_VI\HTML PRoj\Users\lr1385\Harsh Temp\Logout.xls) than we will write:
Action: mod_Logout
Data: T:\Test Services\Offshore Folder\Final_VI\HTML PRoj\Users\lr1385\Harsh Temp\Logout.xls



4.9.1.4 User Defined Functions

For using function we have a keyword/ prefix called **func_**
Similar to mod functionality, for function also we will add "**func_**" as the prefix before the function name. Like: func_FunctionName.
In Data we will write the input parameters for the function
"Parameter1", "Paramter2"

Suppose we want to use a custom function named "FormatDateinMMDDYYYFormat" in our test case then we will write:
Action: func_FormatDateinMMDDYYYFormat



4.9.2 Deleting Test Steps

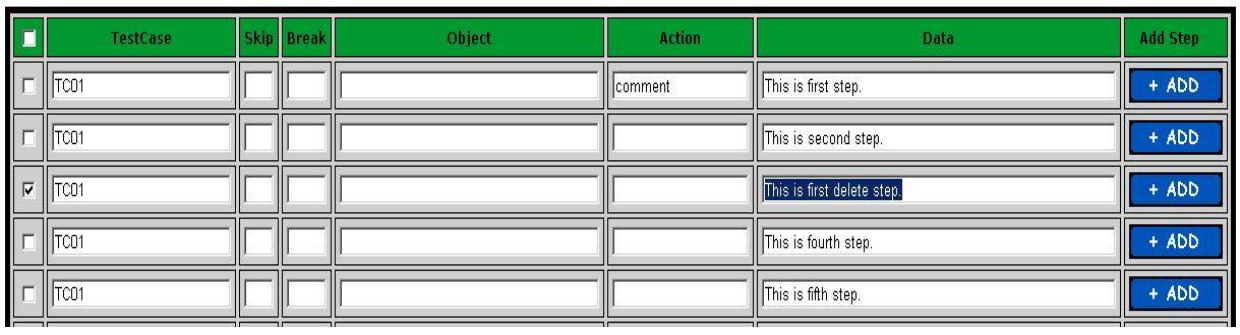
Deleting test steps can be performed on:

1. Individual Step
2. Multiple random steps
3. All the test steps.

User need to click "**Delete**" button.

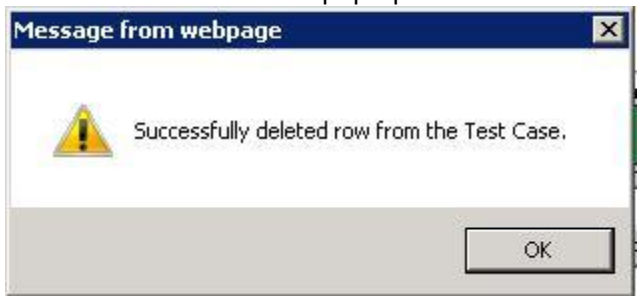
4.9.2.1 Deleting Single Test Step

Step 1: Mark/Check the checkbox in first column

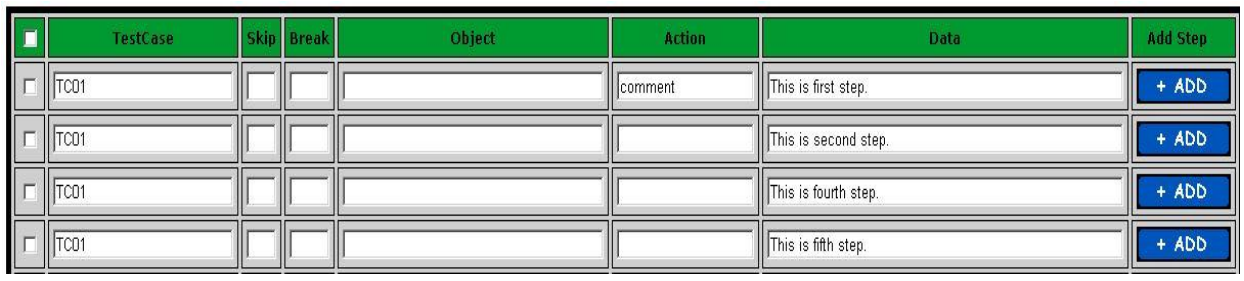


Step2: Click on "**Delete Row**" button.

Step3: Click on "**OK**" button in the pop-up box.

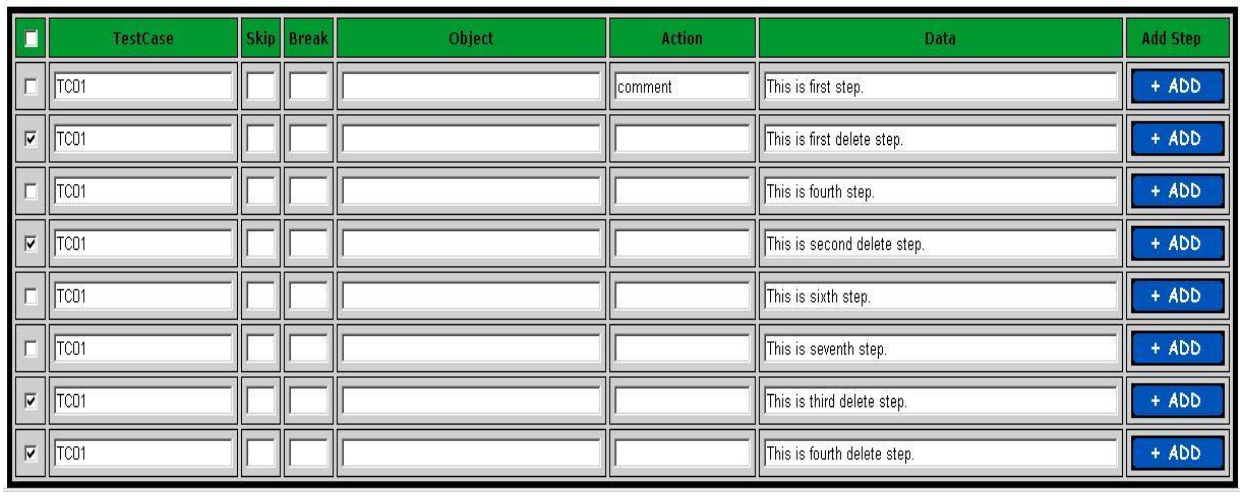


Step4: Step is deleted:



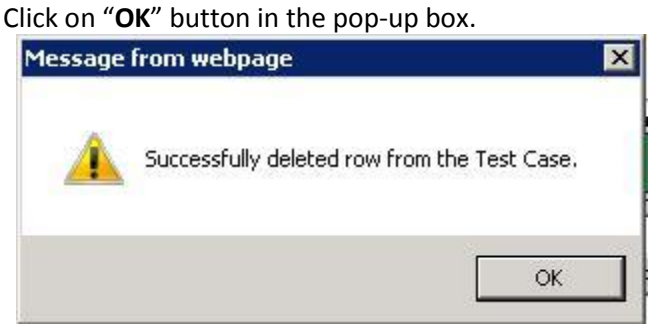
4.9.2.2 Deleting multiple steps from Test Case

Step 1: Mark/Check the checkbox in first column against the steps to delete

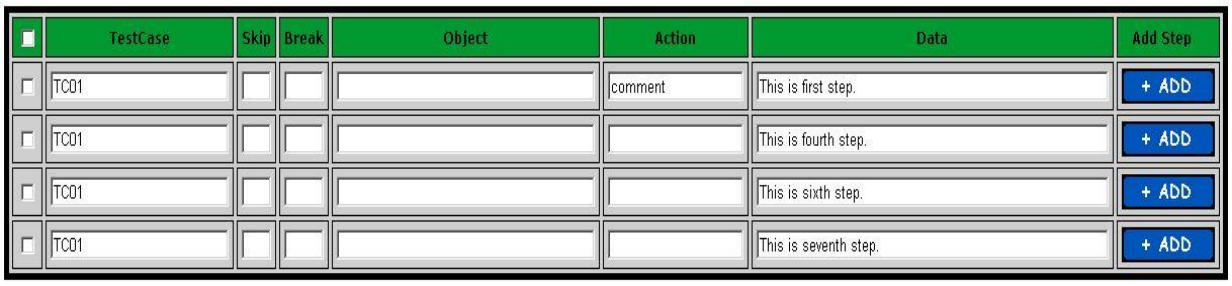


Step2: Click on "**Delete Row**" button.

Step3: Click on “**OK**” button in the pop-up box.

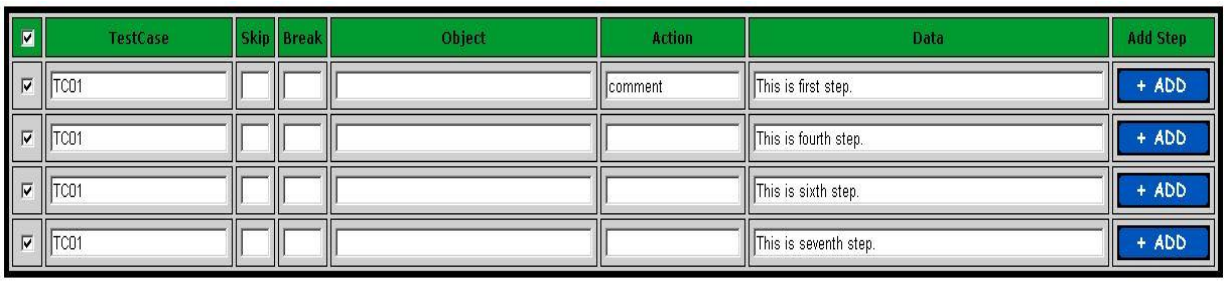


Step4: Mark steps are deleted:



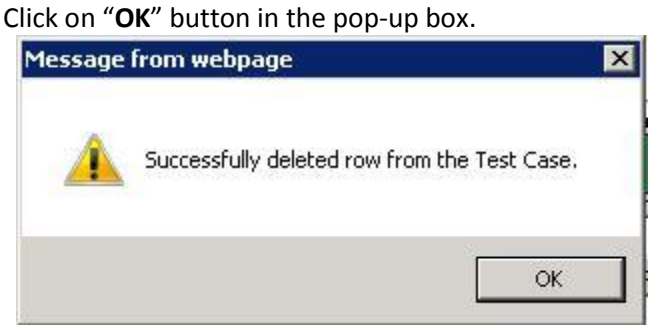
4.9.2.3 Deleting all steps from Test Case

Step 1: Mark/Check the checkbox in header row.

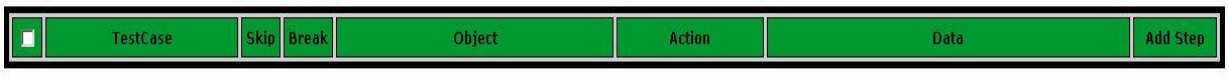


Step2: Click on “**Delete Row**” button.

Step3: Click on “**OK**” button in the pop-up box.



Step4: Mark steps are deleted:



4.9.3 Saving Test Case

Saving xml can be done in two ways:

4. Saving to temp File (Save Script)
5. Saving to custom folder (Save As Script)

4.9.3.1 Saving to temp File

Click on "**Save Script**" button to save directly to the temp file which is saved in the temp folder of the logged in user.

4.9.3.2 Saving to custom folder

Steps to save the script to custom folder.

Step 1: Click on "**SaveAs**" button.

Step2: Enter the path to save file. Click on "**OK**" button.

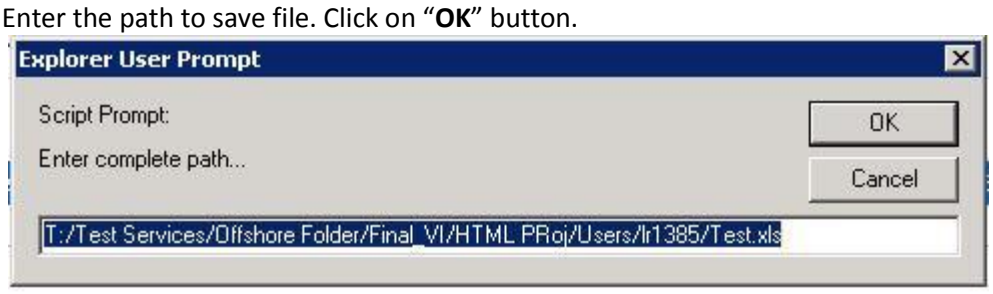
Step3: Click on "**OK**" button in the pop-up box.

4.9.4 Exporting Test Case (In Excel)

Steps to export test script in to excel.

Step 1: Click on "**Export To Excel**" button.

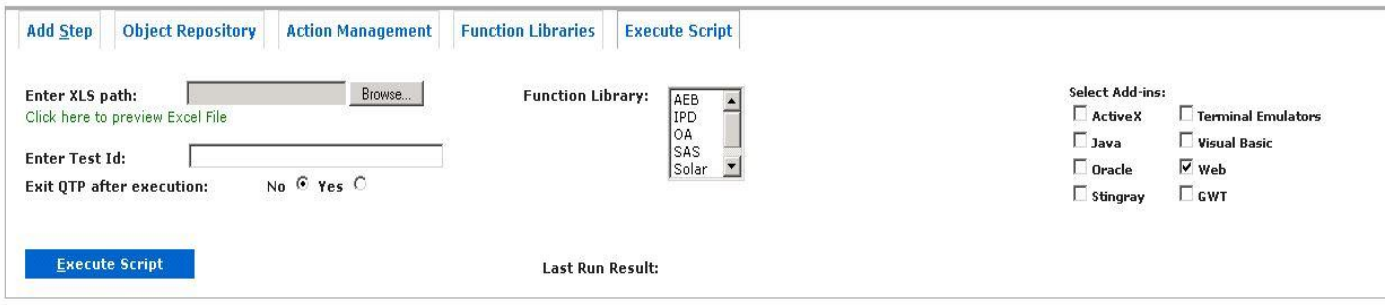
Step 2: Enter the path to save file. Click on "**OK**" button.



Step3: Wait for pop-up box and click on "**OK**" button in the pop-up box.

4.10 Execute Script Tab

4.10.1 Understanding Execute Script Tab



Field Name	Description
Enter XLS Path	This field is optional. If user wants to execute any excel test case he can execute by selecting excel file.
Click here to preview Excel File	Link which gives preview of the selected excel file in “Enter XLS Path” field.
Enter Test ID	Mandatory Field. It will execute the rows having the same value as mentioned in the box. Example: **TC01**
Exit QTP after execution	Radio Button. Determines whether to exit QTP or not after execution.
Exit QTP after execution	Mandatory field. Action Name or “**keyword**”.
Function Library	Web List. Select the function libraries that are applicable for the test script execution.
Last Run Result	Shows the result for the last run. Possible values: Fail, Pass, Stopped, Done.
Select Add-ins	Checkboxes. Allows user to select the QTP Add-ins which are applicable for the particular test script.

4.10.2 Executing Script

Scripts can be executed from utility in two ways:

1. Selecting the Excel script.
2. Executing the script written from the step generator.

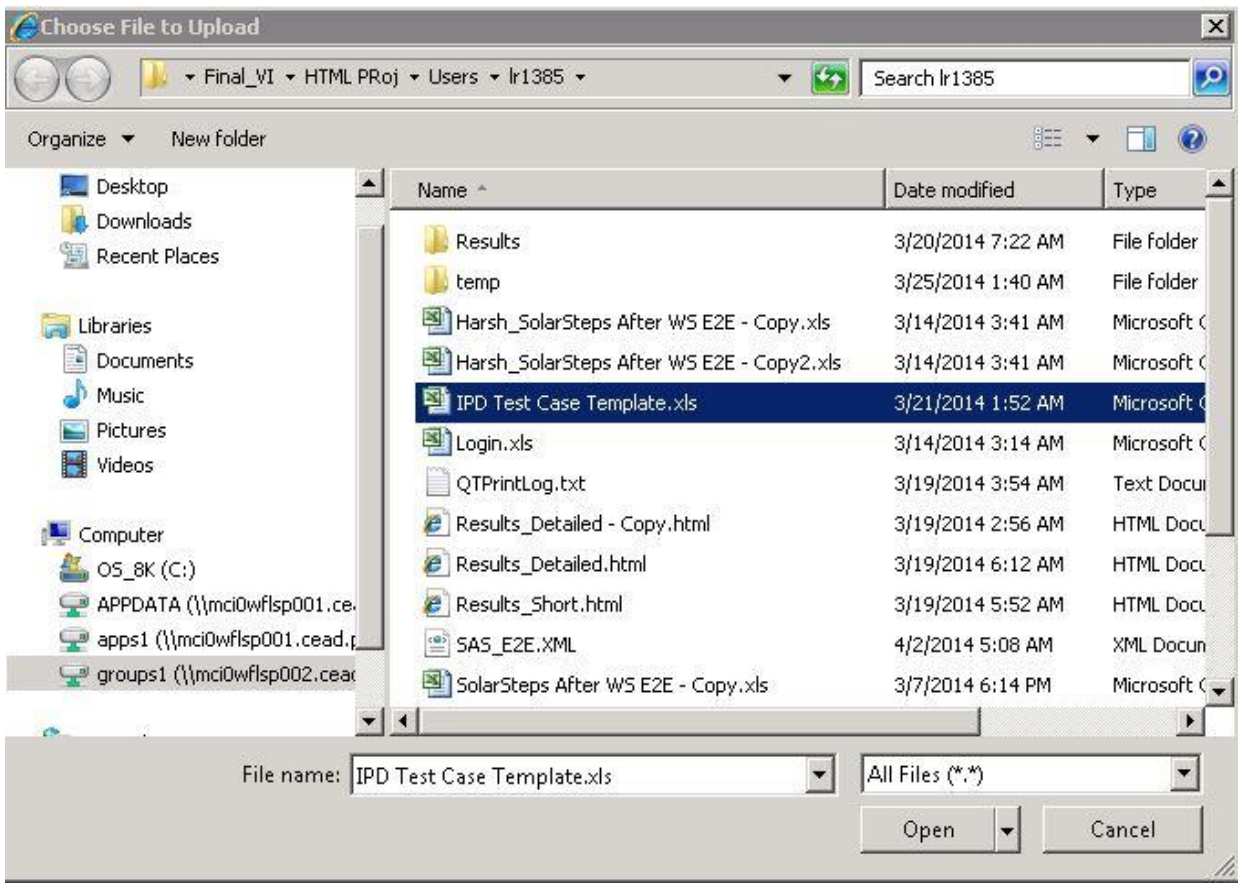
Mandatory Fields:

- ⇨ Test Case ID
- ⇨ QTP Add-ins
- ⇨ Function Library if applicable.

4.10.2.1 Selecting the Excel file

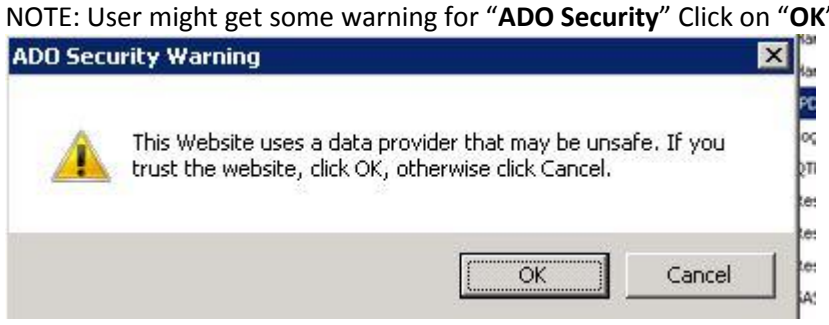
User can select Excel file and can execute the script.

Step 1: Click on "**Browse**" button and select the excel file.

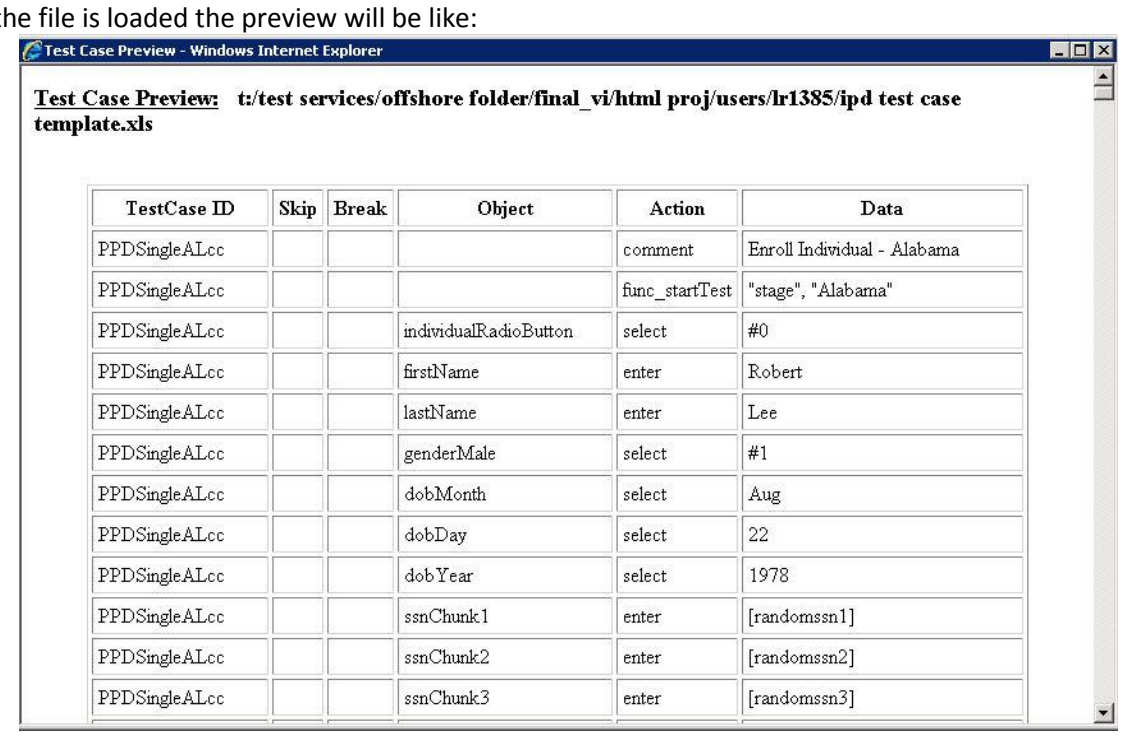


Step 2: Click on "**Click here to preview Excel File**" link to cross verify if the selected file is correct or not.

NOTE: User might get some warning for "**ADO Security**" Click on "**OK**" button.



Once the file is loaded the preview will be like:



Step 3: Close the preview window and

Enter Test Case ID:

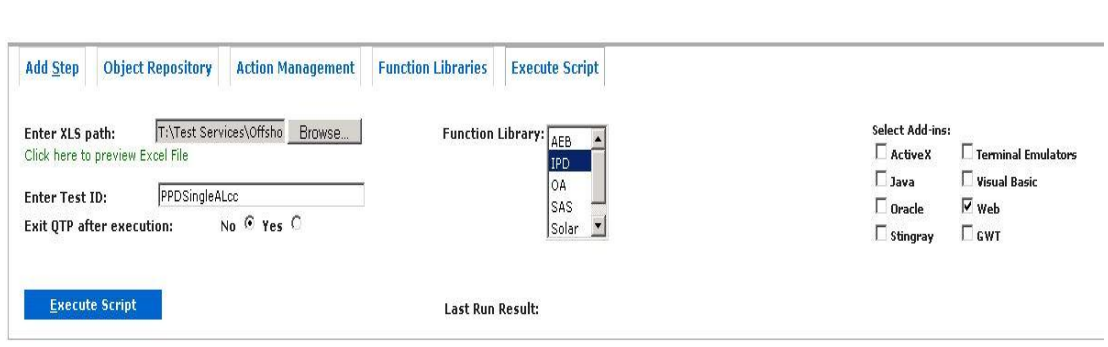
Example: **PPDSingleALcc**

Select Function Libraries:

Example: **IPD**

Select Add-ins:

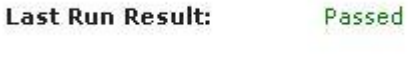
Example: **Web**



Step 4: Click on "**Execute Script**" button.

Wait for QTP to complete the execution.

Step 5: Get the **Last Run Result**.



4.10.2.2 Executing after writing script in Utility

User can execute the script which he/she has written in the utility

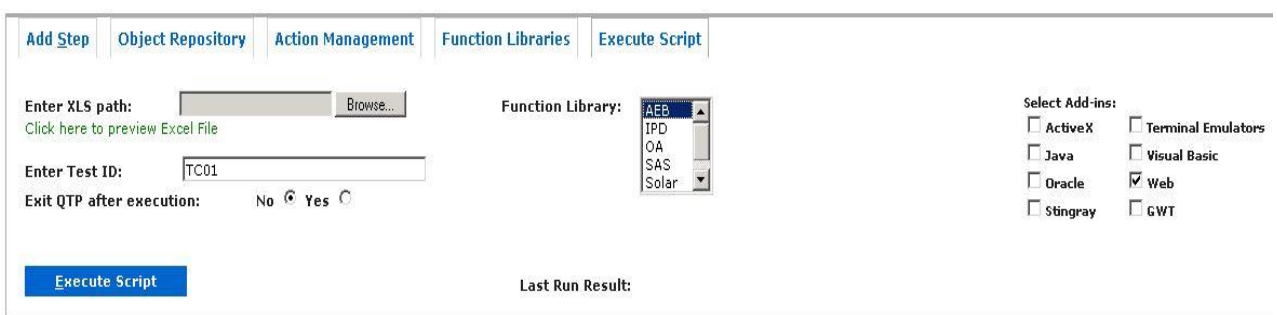
Step 1: Write test steps using Utility.

Step 2: Enter details for:

Enter Test Case ID:
Example: **TC01**

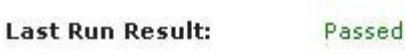
Select Function Libraries:
Example: **IPD**

Select Add-ins:
Example: **Web**



Step 3: Click on "**Execute Script**" button.
Wait for QTP to complete the execution.

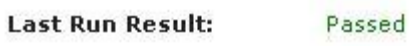
Step 4: Get the **Last Run Result**.



4.10.3 Analyzing results

User can analyze the results by clicking on the status of the result i.e. “Fail”, “Passed”, etc. Result will be opened in a new window.

Click on result status:



New Window with detailed result:

Execution Report

Step	Status
Setting up the object dicitonary.	Pass
Objects Added.	Pass
3/20/2014 6:27:00 AM ExecuteLine , comment, Enroll Individual - Alabama	Pass
3/20/2014 6:27:00 AM ExecuteLine , func_startTest, "stage", "Alabama"	Pass
3/20/2014 6:27:00 AM Calling custom function: startTest ""stage", "Alabama""	Pass
3/20/2014 6:27:34 AM ExecuteLine Browser("creationtime:=0").Page("title:=Individual PPD").WebRadioGroup("html id:=gwt-debug-individualRadioButton-input","index:=0"), select, #0	Pass
3/20/2014 6:27:34 AM ExecuteLine Browser("creationtime:=0").Page("title:=Individual PPD").WebEdit("html id:=gwt-debug-firstName"), enter, Robert	Pass
3/20/2014 6:27:35 AM ExecuteLine Browser("creationtime:=0").Page("title:=Individual PPD").WebEdit("html id:=gwt-debug-lastName"), enter, Lee	Pass
3/20/2014 6:27:35 AM ExecuteLine Browser("creationtime:=0").Page("title:=Individual PPD").WebRadioGroup("html id:=gwt-debug-genderMale-input","index:=0"), select, #1	Pass
3/20/2014 6:27:36 AM ExecuteLine Browser("creationtime:=0").Page("title:=Individual PPD").WebList("html id:=gwt-debug-dobMonth"), select, Aug	Pass
3/20/2014 6:27:37 AM ExecuteLine Browser("creationtime:=0").Page("title:=Individual PPD").WebList("html id:=gwt-debug-dobDay"), select, 22	Pass
3/20/2014 6:27:37 AM ExecuteLine Browser("creationtime:=0").Page("title:=Individual PPD").WebList("html id:=gwt-debug-dobYear"), select, 1978	Pass
3/20/2014 6:27:38 AM ExecuteLine Browser("creationtime:=0").Page("title:=Individual PPD").WebEdit("html id:=gwt-debug-ssnChunk1"), enter, [randomssn1]	Pass
3/20/2014 6:27:39 AM ExecuteLine Browser("creationtime:=0").Page("title:=Individual PPD").WebEdit("html id:=gwt-debug-ssnChunk2"), enter, [randomssn2]	Pass
3/20/2014 6:27:39 AM ExecuteLine Browser("creationtime:=0").Page("title:=Individual PPD").WebEdit("html id:=gwt-debug-ssnChunk3"), enter, [randomssn3]	Pass
3/20/2014 6:27:40 AM ExecuteLine Browser("creationtime:=0").Page("title:=Individual PPD").WebEdit("html id:=gwt-debug-address"), enter, 2010 East Cedar Avenue	Pass
3/20/2014 6:27:41 AM ExecuteLine Browser("creationtime:=0").Page("title:=Individual PPD").WebEdit("html id:=gwt-debug-city"), enter, Mobile	Pass