Open2Test Test Automation Web Framework for OpenScript – Keyword Naming Conventions for Developers

Version 1.0

January 2010

TABLE OF CONTENTS

1. Purpose of the Document

This document provides the naming conventions followed in the Open2Test Test Automation Framework for any Web-based application.

2.   Keyword Naming Conventions

2.1.   Naming Convention Used for Keywords

1. **Launch**

   'Launch' is used to launch the application under test (AUT). This keyword triggers the driver script to launch the application.

2. **Perform**

   'Perform' is used to perform an operation on a particular object (e.g., clicking on a button, closing an open window, or typing some text in a textbox). This keyword should be entered in the corresponding row in the second column if any such operations are to be performed.

3. **Check**

   'Check' is used to check if the required property of a particular object is attained at runtime. This is a type of validation step (expected result).

4. **Condition**

   'Condition' is used to compare two variables, checking properties, existence of windows, etc.

5. **CallFunction**

   'CallFunction' is used to call an external user-defined function. The function can have one or more arguments.

6. **Storevalue**

   'Storevalue' is used to store the property values of different objects in different environment variables. These environment variables can later be used as input parameters in various functions and also in scripts.

7. **Message**

   'Message' is used for debugging to display the contents of a variable in Results View as comments.

8. **Strreplace**

   'Strreplace' is used for replacing a 'sub string' inside a 'main string' with a new 'sub string'.

9. **Strconcat**

   'Strconcat' is used to concatenate any number of strings with each other.

10. **Wait**

   'Wait' is used to place static waits in the keyword script.

11. **Arith**

   'Arith' is used to perform the arithmetic operations on the variables.

12. **Assignvalue**

   'Assignvalue' is used to assign dynamically generated values from the application to environment variables.

13. Callaction

'Callaction' is used to call another linear script or Keyword Scripts. Boolean Values of the parameters <doInitialize>, <doRun>, and <doFinish> denote whether to call the corresponding modules in the linear script or not.

14. Loop

'Loop' is used to loop a set of actions given in the data table.

15. Endcondition

'Endcondition' denotes the end of a condition construct.

16. Endloop

'Endloop' denotes the end of a loop.

17. JumpTo

'Jumpto' is used to transfer flow to the required label.

18. Label

'Label' is an identifier used to denote particular lines of code with which execution flow can be transferred directly to the denoted code.

19. Strlen

'Strlen' is used to determine length of a string.

20. GetNextDb

Navigates to the next record (row) of configured databank while executing.

2.2.   Naming Conventions Used for Objects

The table below provides the naming conventions followed for different Web objects in the Open2Test Test Automation Framework.

Sl.No	Web Object Class	Naming Conventions Used in the Open2Test Test Automation Framework
1.	Browser	Browser
2.	WebButton	Button
3	WebImage	Image
4.	WebCheckBox	Checkbox
5.	WebEditBox	Textbox
6.	WebRadioButton	Radiobutton
7.	WebLabel	Label
8.	Weblink	Link
9.	WebSelectbox	SelectBox
11.	Webalertdialog	AlertDialog
12.	Weblogindialog	Logindialog

Table 1: Naming Conventions Used for Objects

2.3.   Naming Conventions Used for Actions

1. **Click**

    'Click' is used to perform the click operation on Web objects. It is used with the perform keyword in keyword scripts (e.g., clicking a button).

2. **Cleartext**

    'Cleartext' is used to clear the specified text from the textbox control. It is used with the perform keyword in keyword scripts.

3. **Close**

    'Close' is used to perform the close operation on any open browser. It is used with the perform keyword in keyword scripts.

4. **Maximize**

    'Maximize' is used to perform the maximize operation on any open browsers. It is used with the perform keyword in keyword scripts.

5. **Minimize**

    'Minimize' is used to perform the minimize operation on any open browsers. It is used with the perform keyword in keyword scripts.

6. **Selecttext;<name/Item>**

    'Selecttext' is used to select an item from Selectbox based on the text of the items. It is used with the perform keyword in keyword scripts.

7. **Selectindex;<index>**

    'Selectindex' is used to select an item from a Selectbox based on the index of the items. It is used with the perform keyword in keyword scripts.

8. **Settext;<Text>**

    'Settext' is used to assign a value to a text box. It is used with the perform keyword.

9. **Setpasswordtext;<Text>**

    'Setpasswordtext' is used to assign a value to a text box in an encrypted format. It is used with the perform keyword.

10. **Setlocation;<a;b;c;d>**

    'Setlocation' is used to assign a location to a browser object. It is used with the perform keyword.

11. **Setcurrentdate;**

    This sets the current system date.

    It is used with the perform keyword in keyword scripts.

12. **Setcurrenttime;**

    This sets the current system time. It is used with the perform keyword in keyword scripts.

13. **Checked;<True/False>**

This is a check operation that is used to verify whether a checkbox is selected or not.

14. **Name;<Varname>**

   This is a Check operation that is used to verify whether a Web element with the name <varname> exists.

15. **Disabled;<True/False>**

   This is a Check operation that is used to verify whether the given Web element is disabled or not.

16. **ReadOnly; <True/False>**

   This is a check operation that is used to verify whether the particular textbox is editable or not.

17. **RadioOption; <True/False>**

   This is a check operation that is used to verify whether the particular radio option exists for a group of radio buttons.

18. **Navigate; <URL of Browser>**

   This is a perform operation and is used on a browser object to navigate to a particular URL.

19. **Back**

   This is a perform operation and is used on a browser object to navigate to previous page.

20. **Forward**

   This is a perform operation and is used on a browser object to navigate to next page.

21. **WaitforPage;<tsec>**

   This is a perform operation and is used on a browser object to make the browser page wait for specified time in seconds.

22. **Refresh**

   This is a perform operation and is used on a browser object to refresh current browser page.

23. **ReadOnly; <Varname>**

   This is a store value operation that is used to store the read only property value of the object in the specified variable.

24. **Value; <True/False>**

   This is a check operation that is used to verify whether the particular textbox has a particular value.

25. **Value; <Varname>**

   This is a check operation that is used to verify whether the particular textbox with the value having name as <varname> exists.

26. **Statusbartext;<text>**

   This is a check operation that is used to verify whether the browser's status bar displays the text.

27. **innertext;<text>;<true/false>**

   This is a check operation that is used to verify whether the particular text is present or not in a specified dom path.

28. **innertext;<varname>**

This is a store value operation that is used to store the text of the provided dom path in a variable.

29. Check

This is used to select/unselect the checkbox depending on the 'Action value' column value. It is used with the perform keyword.

30. Doubleclick

'Doubleclick' is used to perform the doubleclick operation on Web objects. It is used with the perform keyword in keyword scripts.

31. Rightclick

'Rightclick' is used to perform the rightclick operation on Web objects. It is used with the perform keyword in keyword scripts.

32. ShowContextMenu

'Showcontextmenu' is used to perform the 'Showcontextmenu' operation on Web objects. Show Context Menu command shows the right click menu on the specified object. It is used with the perform keyword in keyword scripts.

33. Mouseover

'Mouseover' is used to perform the mouse over operation on Web objects. Mouse Over command simulates the action of hovering the mouse over the specified object. It is used with the perform keyword in keyword scripts.

34. Keypress;<key combination>

This is a perform operation that simulates the same expected result as the key combination.

35. Multiselecttext;<text1>;<text2>or <ind1>;<ind2>

'Multiselecttext' is used to select many items from the selectbox based on the index values or item names. It is used with the perform keyword in keyword scripts.

36. Alertok

Clicks on the 'OK' button of the alert box.

37. lgndialogok;<uname>;<pwd>

'**lgndialogok**' is a perform operation that clicks the 'OK' button in the Login Dialog box to initiate login action.

38. lgndialogcancel

'**lgndialogcancel**' is a perform operation that clicks the 'Cancel' button in the Login Dialog box to abort login action.

39. Settext;<Value>

Sets the required text box with the provided text.

40. Setd;<Value to be added/subtracted>

Subtracts/adds the specified date value with the current date and stores the result in the text box.

41. Setm;<Value to be added/subtracted>

Subtracts/adds the specified month value with the current month and stores the result in the text box.

42. **Sety;<Value to be added/subtracted>**

Subtracts/adds the specified year value with the current year and stores the result in the text box.

43. **Ddownlen;<len>**

It is a check operation that calculates the number of elements present in the dropdown and compares it with the provided <len> value. It either denotes True or False, depending on the comparison result.

44. **Ddownlen;<varname>**

Calculates the number of elements present in the dropdown and stores the result in a variable.

45. **GetCellCnt**

Finds the number of cells that a Web table contains; stores the result in a variable or compares it with the provided value; and returns the result depending on either the 'StoreValue' or 'Check' operation, respectively.

46. **GetColCnt**

Finds the number of columns that a Web table contains; either stores the result in a variable or compares it with the provided value; and returns the result depending on either the 'StoreValue' or 'Check' operation, respectively.

47. **GetRowCnt**

Finds the number of rows that a Web table contains; either stores the result in a variable or compares it with the provided value; and returns the result depending on either the 'StoreValue' or 'Check' operation, respectively.

48. **RadioOption**

Identifies whether the specified radio button is selected or not; stores the value in the provided variable or compares the current status with the provided status of the radio button; and then returns the result – depending on the operation.

49. SelectedText

Identifies the selected text of the select box; stores it in a variable or compares it with the provided text; and then returns the result – depending on the operation.

50. **Selectedmultitext;Itm1:Itm2..**

It is a check operation that identifies whether the provided texts of the select box have been selected or not. It also stores the result as either True or False, depending on the comparison.

51. **Selectedmultitext;<Var1>**

It is a store value operation that captures the items selected in the list box, concatenates them with the delimiter ':' and stores it in a required variable.

52. **Folder;Create|<Folder Path/Name>;<bOverwrite>**

This is used to create a folder in the specified path with a flag for overwrite.

53. **Folder;Delete|<Folder Path/Name>/<File Path/Name>**

This is used to delete a folder in the specified path.

54. **Folder;Copy|<Source Path/Name>;<DestinationFolder Path/Name>;bOverwrite**

This is used to copy a folder from the source to the destination path specified with a flag for overwrite.

55. **File;Create|<Folder Path/Name>;<bOverwrite>**

This is used to create a file in the specified path with a flag for overwrite.

56. **File;Delete|<Folder Path/Name>/<File Path/Name>**

This is used to delete a file in the specified path.

57. **File;Copy|<Source Path/Name>;<DestinationFolder Path/Name>;bOverwrite**

This is used to copy a file from source to destination path specified with a flag for overwrite.

58. **File;append|<File Path/Name>;<The value to be entered>**

This is used to write the file with the data mentioned in the specified path.

59. **File;Read|<File Path/Name>;<Variable to store data from file>**

This is used to read the contents of a mentioned file and store the values in the specified variable.

60. **File;Read|<File Path/Name>;<Variable to store data from file>;<lineIndex>**

This is used to read the contents of the specified line from the mentioned file and store the values in the specified variable.

61. **File;linecount|<File Path/Name>;<Variable to store the value>**

This is used to find the number of lines of data in the specified file.

62. **Sqldefine;datasourcename|<Uname>;<Pwd>;<AliasName>**

Defines the SQL connection to the specified Data Source Name with the provided Username, password, and alias name.

63. **SqlConnect;<aliasname>**

Establishes the connection with the provided data source alias name.

64. **SqlDisconnect;<aliasname>**

Disconnects the connection from the provided data source alias name.

65. **Sqlexecute;aliasname|<query>;<varname>**

'sqlexecute' is used for executing the specified SQL query in the provided alias name of the database. It stores the value of the first query result in the provided variable.

66. **Sqlexecute;aliasname|<query>;<varname>;<rowindex>**

'sqlexecute' is used for executing the specified SQL query in the provided alias name of the database. It stores the value of the specified row index in the provided variable.

2.4. Naming Conventions used for Variables

- To store the property value of an object, an environment variable is used.

  Example;

Storevalue	Textbox; <textbox name>	Prop_name; <varName>

  Here, the value in the textbox is stored to a variable 'varName'.

- To assign a value to a field from a variable, the variable should be preceded by '**#**'.

Perform	Textbox;<textbox name>	Settext;#varName

  Here, the value stored in varName is assigned to the textbox.

  Note: Variable naming standards to be followed are shown below.

2.5. Naming Convention Used for Databank

To input a value to a field from a data table, the naming convention used is **db**.

Example;

Perform	Textbox;<textbox name>	Set; dt_<Parameter>

Here the 'dt' signifies that the value is to be taken from the data table from the column "Parameter".