



Open2Test Test Automation Framework for OpenScript - Scripting Standards for Web

Version 1.0

January 2010

DISCLAIMER

Verbatim copying and distribution of this entire article is permitted worldwide, without royalty, in any medium, provided this notice is preserved.

TABLE OF CONTENTS

1. INTRODUCTION	5
1.1. Purpose	5
2. STANDARDS FOR KEYWORD SCRIPTING	6
2.1. Before Starting	6
2.2. Column Description	7
2.2.1. Automate (Column 'A')	7
2.2.2. Keyword (Column 'B')	7
2.2.3. Object (Column 'C')	9
2.2.4. Actionvalue1 (Column 'D')	10
2.2.5. Manual Step Description (Column 'E')	13
2.2.6. Delimiters.....	13
2.2.7. Variables.....	14
3. SEQUENCE OF KEYWORDS	16
3.1. Use of 'Conditional statements'	16
3.2. Use of 'Loop' Constructs	16

LIST OF TABLES

Table 1: Objects used in the Open2Test Test Automation Framework..... 10

TABLE OF FIGURES

Figure 1: Calling the Framework and Keyword Script.....	6
Figure 2: Column 'Automate'	7
Figure 3: Column 'Action'	7
Figure 4: Column 'Object'	9
Figure 5: Column 'Actionvalue1'	11
Figure 6: Column 'Manual Step Description'	13

1. Introduction

1.1. Purpose

This document provides details about the various columns used, the keywords and their descriptions, along with some methodologies that need to be followed while scripting using keywords.

2. Standards for Keyword Scripting

2.1. Before Starting

Before going into the details about the columns used for keyword scripting, the user should be familiar with what is known as the 'keyword script' and how to create an instance of Web_Framework class in the test script for execution.

Keyword test script is a spreadsheet in .csv file format, which corresponds to the manual test case. Keyword Test script details the user actions to be performed on AUT.

To be Autom	Keyword	Object	Action Value1	Ma
r	perform	browser;*about blank*	navigate;http://s2b21140/O2T/	Open
r	perform	browser;*Login Page*	waitforpage;15	Wait until Airline ticket reservation system launches
r	perform	textbox;Uname	settext;Aadmin	Enter Username
r	perform	textbox;Pwd	settext;Ppwd	Enter Password
r	perform	button;Lgn	click	Click on the Login button
r	perform	browser;*Home Page*	waitforpage;10	Wait until system logs into Airline Home Page

For execution, instance of Web_Framework class needs to be created in Test Script, which can be created by:

Web_Framework F = new Web_Framework ("CSV file Path", "Object Library Path", "DataBank Alias Name").

```

// * File: script.java
import com.Open2Test.Oscript.Web_Framework;

public class script extends IteratingVUserScript {
    @ScriptService oracle.oats.scripting.modules.utilities.api.UtilitiesService utilities;
    @ScriptService oracle.oats.scripting.modules.browser.api.BrowserService browser;
    @ScriptService oracle.oats.scripting.modules.functionalTest.api.FunctionalTestService ft;
    @ScriptService oracle.oats.scripting.modules.webdom.api.WebDomService web;

    public void initialize() throws Exception {
    }

    /**
     * Add code to be executed each iteration for this virtual user.
     */
    public void run() throws Exception {
        browser.launch();
        InternalWebDOMService Ione = web.getInternalService();
        PlaybackHandler Ph = Ione.getHandler();
        Connection conn1 = Ph.getConnection();

        //To use Databank, configure the csv file with the script and provide the a
        //Use '/' instead of '\\' while mentioning the drive path
        Web_Framework F = new Web_Framework("C:\\KTC1.CSV", "C:\\Portal.properties",
        IScriptVUser iVuser = this.getScriptVUser();

        F.Set(iVuser, conn1);

        F.run();
    }

    public void finish() throws Exception {
    }
}

```

Figure 1: Calling the Framework and Keyword Script

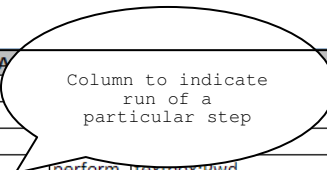
Data Bank name is an optional parameter while creating the instance of the Web_Framework class. This can be made NULL when there is no data bank usage in the test script.

2.2. Column Description

This section gives a description of the columns used for keyword scripting.

2.2.1. Automate (Column 'A')

The data in the 'Automate' column decides whether the current step in the test case is to be run (automated) or not. This column has the letter 'r', which denotes that the current step should be run. If any step in the test case is not being run then the corresponding row in the first column is to be left blank. The steps will run based only on the data in this column.

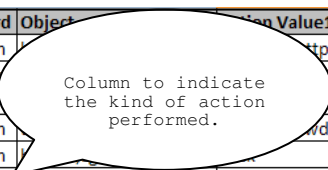


To be Automated	Action Value1	Manual Test Description
r	navigate;http://s2b21140/O2T/	Open the Airline Ticket Reservation System
r	waitforpage;15	Wait until Airline Ticket Reservation system launches
r	settext;Aadmin	Enter Username
r	perform;textbox;Pwd	settext;Ppwd
r	perform;button;Lgn	click
r	perform;brower;*Home Page*	waitforpage;10
		Wait until system logs into Airline Home Page

Figure 2: Column 'Automate'

2.2.2. Keyword (Column 'B')

The second column of the active data sheet is used to indicate the generic type of action being performed on the application under test (AUT). The action column is dedicated to different kind of actions that are to be performed on a particular object.



To be Automated	Keyword	Object	Action Value1	Manual Test Description
r	perform		navigate;http://s2b21140/O2T/	Open the Airline Ticket Reservation System
r	perform		waitforpage;15	Wait until Airline Ticket Reservation system launches
r	perform		settext;Aadmin	Enter Username
r	perform		perform;textbox;Pwd	settext;Ppwd
r	perform		perform;button;Lgn	click
r	perform		perform;brower;*Home Page*	waitforpage;10
				Wait until system logs into Airline Home Page

Figure 3: Column 'Action'

The keywords that can be used in this column are:

1. Launch

'Launch' is used to launch the AUT. This keyword triggers the driver script to launch the browser

2. Perform

'Perform' is used to perform an operation on a particular object, such as clicking on a button, closing an open browser, typing some text in a textbox, etc. 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' provides a feature for comparing two variables, checking properties, checking for the existence of Web pages, etc.

5. CallFunction

'CallFunction' is used to call any declared function that is used in a particular script. These functions should be declared in a different class file.

6. Storevalue

'Storevalue' is used to store the property values of different objects in different dictionary variables. These dictionary 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. Strsearch

'Strsearch' is used to search for a 'sub string' inside a 'main string'.

9. Strreplace

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

10. Strconcat

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

11. Wait

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

12. Arith

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

13. Assignvalue

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

14. Callaction

'Callaction' is used to call reusable actions that are declared in the script.

15. Loop

'Loop' is used to loop a set of actions given in the test script.

16. convert

'Convert' is used to typecast from one data type to another.

17. Function

'Function' is used to perform FSO (File system Object) operations, such as creating a folder in a specified path, creating a file in a specified path, etc.



A detailed description of the keywords is provided in the Keyword Reference Document.

2.2.3. Object (Column 'C')

The third column of the active data sheet is used to indicate the object on which a particular type of action is to be performed. When the object is present in the object library, the object type and object path (alias name of the corresponding element in Object Library) are specified in column C (as shown in example 1).

The object column or column 'C' contains all the required details for an object (viz. Class to which the objects belong and the object name) on which various operations and validations are to be performed separated by a delimiter ';'.

Example 1:

Action	Object
Perform	Textbox; Lastname

In the above example, some operation has to be performed on an object of class 'Textbox', the path of which is stored with the name 'Lastname' in the Object Library.

The object and its name are usually separated by a delimiter ';' as shown in the above example. (Delimiters will be covered in a later topic.)



Column to indicate the Object

To be Autom	Keyword	Object	Action value1	Manual Test Description
r	perform	browser;*about:blank*	navigate;http://s2b21140/O2T/	Open the Airline Ticket Reservation System
r	perform	browser;*Login Page*	waitforpage;15	Wait until Airline Ticket Reservation system launches
r	perform	textbox;Uname	settext;Aadmin	Enter Username
r	perform	textbox;Pwd	settext;Ppwd	Enter Password
r	perform	button;Lgn	click	Click on the Login button
r	perform	browser;*Home Page*	waitforpage;10	Wait until system logs into Airline Home Page

Figure 4: Column 'Object'

The objects that are commonly used are:

Sl.No	Objects Used in the Open2Test Test Automation Framework	Web Object Class
1.	Window	Window
2.	Button	WebButton
3.	Checkbox	WebCheckBox
4.	Dropdown	WebSelectBox
5.	Textbox	WebEditBox
6.	Radiobutton	WebRadioButton
7.	Link	WebLink
8.	Image	WebImage
9.	Login Dialog	Login Dialog box

Table 1: Objects Used in the Open2Test Test Automation Framework



A detailed description of the keywords is given in the Keyword Reference Document.

2.2.4. Actionvalue1 (Column 'D')

The fourth column of the active data sheet indicates the specific action being performed on the object present in the AUT. It contains the details of all the operations or verifications that have to be performed on the objects listed in the 'Objects' column.

Consider the example of the object 'Web Button' with the name OK.

One of the actions that can be performed on a WebButton would be Click, so in column 4 the above operation is put in the keyword form as "CLICK".

Example 2: The keyword CLICK on an OK button is as follows:

Action	Object	Operation
Perform	Button;LgnButton	Click

ACTION

If the user wants to **check** if the button is enabled before clicking, the syntax would be:

Action	Object	Operation
Check	Button;CheckBox1	disabled;False

CHECKING

It would be the same if the user wants to check whether the object is disabled. The syntax would be:

Action	Object	Operation
Check	Button;CheckBox2	Disabled:True

CHECKING

To store the required attribute value in a variable, the syntax would be:

Action	Object	Operation
StoreValue	TextBox;Uname	value;Var1

Store Value

To be Auton	Keyword	Object	Action Value1	
r	perform	browser;*about blank*	navigate;http://s2b2114	System
r	perform	browser;*Login Page*	waitforpage;15	on system launches
r	perform	textbox;Uname	settext;Aadmin	
r	perform	textbox;Pwd	settext;Ppwd	Enter Password
r	perform	button;Lgn	click	Click on the Login button
r	perform	browser;*Home Page*	waitforpage;10	Wait until system logs into Airline Home Page

Column to indicate specific action to be performed

Figure 5: Column 'Actionvalue1'

The most commonly used keywords for specific actions that can be used with the generic keyword 'Perform' written in Column 'Action' are:

1. **Click**

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

2. **Close**

'Close' is used to perform the close operation on any open objects. It is used with the perform keyword in keyword scripts (e.g., closing a window)

3. **Maximize**

'Maximize' is used to perform the maximize operation on the browser object. It is used with the perform keyword in keyword scripts.

4. **Minimize**

'Minimize' is used to perform the minimize operation on the browser object. It is used with the perform keyword in keyword scripts.

5. **Restore**

'Restore' is used to perform the restore operation on any open objects. It is used with the perform keyword in keyword scripts.

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

'Select' is used to select a required text item from Dropdown, Combo box and List box. It is used with the perform keyword in keyword scripts.

7. **Selectindex; <index>**

'Selectindex' is used to select a required item index from Dropdown, Combo box and List box. It is used with the perform keyword in keyword scripts.

8. **Settext;<Text>**

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

9. **setcurrenttime**

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

10. **setcurrentdate**

This sets the current system date to the text box. It is used with the perform keyword in keyword scripts.

11. **Doubleclick**

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

12. **<conversiontype>;<variable name>;<format type>**

This is used to convert a variable from one data type to another.

13. **MultiSelectText;<item name1>:<item name2>**

This is used to select the range of items in a list box.

14. **<Folder Path/Name>/<File Path/Name>;bOverwrite**

This is used to create a folder/file in the specified path.

15. **Delete|<Folder Path/Name>/<File Path/Name>**

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

16. **<Source Path/Name>;<DestinationFolder Path/Name>;bOverwrite**

This is used to copy a folder from the source to the destination path specified.

17. **<Source File Path/Name>;<Destination Folder Path>;bOverwrite**

This is used to copy a file from the source to the destination path specified.

18. **Append|<File Path/Name>;<The value to be entered>**

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

19. **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.

20. **SQLExecute|<query>;<variable to store data>**

This is used for executing the query and to store the value of the first result in a variable.

The most commonly used keywords for specific actions that can be used with the generic keyword 'Check' written in Column 'Action' are:

1. **Checked:<True/False>**

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

2. **Enabled:<True/False>**

This is a check operation that is used to verify whether the given window or object is enabled or not.

3. **Exist:<True/False>**

This is a check operation that is used to verify whether the specified window or object exists or not.

4. **ItemCount:<Item>**

This is a check operation that is used to verify the number of items present or not in a specified object.

5. **Text:<text/#Variable_Name>**

This is a check operation that is used to verify whether the required text is present or not in the object.

6. **ItemExist:<Item name>**

This is a check operation that is used to verify whether an item is present or not in the window object.



A detailed description of the keywords is provided in the Keyword Reference Document.

2.2.5. **Manual Step Description (Column ‘E’)**

The fifth column of the keyword test sheet may be used to store the test description of the corresponding step, which would be populated in Results View providing detailed test reporting.

It provides a better understanding of the steps being performed in the particular test script and also helps to map the test script to the manual test case.

To be Auto	Keyword	Object	Action Value1	Manual Test Description
r	perform	browser;*about blank*	navigate;http://s2b21140/O2T/	Open the Airline Ticket Reservation System
r	perform	browser;*Login Page*	waitforpage;15	Wait until Airline Ticket Reservation System launches
r	perform	textbox;Uname	settext;Aadmin	Enter Username
r	perform	textbox;Pwd	settext;Ppwd	Enter Password
r	perform	button;Lgn	click	Click on the Login button
r	perform	browser;*Home Page*	waitforpage;10	Wait until system logs into Airline Home Page

Figure 6: Column ‘Manual Step Description’

2.2.6. **Delimiters**

Delimiters are any string characters used to identify the sub-string limits. Delimiters are generally used with the Split function, which is used to split the input into different sub strings.

When a delimiter is omitted, the space character (“ ”) is assumed to be a delimiter.

Purpose of using delimiters:

The main purpose of using delimiters in this framework is to break down the input values to different strings and take them as keywords to perform any operation concerned with that object.

Delimiters Used in this framework:

The most important point to keep in mind while scripting using the keyword-driven approach is to place separators or delimiters between two keywords. The delimiters that are used in the framework are:

- ; (semi colon)
- : (colon)

Understanding the usage of delimiters:

There are four columns involved in the keyword-driven approach. The role of delimiters comes in the 'Objects' column (column 3) and 'Actionvalue1' column (column 4).

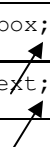
'Objects' column (column 3) And 'Action Value' column (column 4):

In the 'Objects' column, delimiters are used to define the class and the path of the object. In the 'Action Value' column, delimiters are used to define 'Action Type' and 'Action Value'.

The delimiter used in this column separates the class of the object and the path of the object with a semicolon ';'.
;

Example:

Textbox; <textboxname>
settext;Testing



2.2.7. Variables

- To store a value in a variable, a dictionary variable is used.

Example:

Assignvalue	strName;Smith	
-------------	---------------	--

Here in the variable 'strName', the value 'Smith' is stored.

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

Example:

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

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

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

Example:

Perform	Textbox;<textbox name>	Settext:#varName
---------	------------------------	------------------

Here, the value stored in varName is typed into the textbox.

To define a variable certain standards need to be followed. For example, for a variable to store a string value it should be appended with “Str” ex.StrVarName. Similarly, for integer, it should be appended with “int”. For Boolean, it should be appended with “bln”.

Note: Variable names are ***case sensitive*** in Open2Test Test Automation Framework.

- To use the databank variable value in the test execution databank, the variable name should precede with ‘dt_’ as shown below:

Example:

Perform	Textbox;<textbox name>	Set:dt_Coll
---------	------------------------	-------------

Where ‘Coll’ corresponds to the column name in databank file

3. Sequence of Keywords

While scripting using keywords, some keywords have to be written in combination with other keywords. This section deals with those methodologies.

3.1. Use of 'Conditional statements'

If the user is implementing an If - Else conditional statement, the following keyword syntax needs to be followed:

Example:

Keyword	Object	Action Value
Condition	<var1>;comparator	<var2>
Keywords...		
Keywords...		
endCondition		

If the condition mentioned is True, execution starts from the preceding and would end at the row where the corresponding 'endCondition' has been specified. If the condition specified is False, then the execution would continue from the preceding row of corresponding 'endCondition' keyword.

Note: Every 'condition' keyword should contain the corresponding 'endCondition' keyword.

Note: Nested if condition can also be used. Please refer to Open2Test Test Automation Framework for OpenScript - Tips & Tricks for more details.

3.2. Use of 'Loop' Constructs

Users can loop over certain test steps for the required number of times using the 'Loop' keyword as mentioned below:

Keyword	Object	Action Value
Loop	4	
Keywords...		
Keywords...		
endLoop		

Note: Every 'Loop' keyword should contain the corresponding 'endLoop' keyword.

Note: Nested Loop constructs can also be used. Please refer to Open2Test Test Automation Framework for OpenScript - Tips & Tricks for more details.



This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.