



# Open2Test Test Automation Framework Keyword Naming Conventions for Developers (Oracle Applications) - QTP

Version 1.0

November 2009

## 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**

<b>1. PURPOSE OF THE DOCUMENT .....</b>	<b>3</b>
<b>2. KEYWORD NAMING CONVENTIONS.....</b>	<b>4</b>
2.1. Naming Conventions Used for Keywords .....	4
2.2. Naming Conventions Used for Objects .....	6
2.3. Naming Conventions Used for Variables .....	10
2.4. Naming Convention Used for Data Table .....	11
2.5. Naming Convention Used for Parameter .....	11
2.6. Naming Convention Used for Environment .....	11

## 1. Purpose of the Document

This document provides the naming conventions followed in Open Source Test Automation Framework for the Oracle technology.

## 2. Keyword Naming Conventions

### 2.1. Naming Conventions Used for Keywords

#### 1. **LaunchApp**

'LaunchApp' is used to launch the application under test (AUT). This keyword triggers the driver script to launch the application, either from a specified folder (the location of which is specified in the third column). Or if the application is already synchronized with QTP, then this automatically launches the application from the location specified in QTP.

#### 2. **Context**

'Context' is used only on a window or a session or a dialog. This keyword brings a particular window to the current context, so that any operation or checking can be performed on it.

#### 3. **Perform**

'Perform' is used to perform an operation on a particular object such as clicking on a button, selecting an option from combobox, or typing 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.

#### 4. **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).

#### 5. **Condition**

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

#### 6. **CallFunction**

'Call Function' is used to call any declared function, which is used in a particular script. These functions should be declared in a different .vbs file.

#### 7. **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 as well as in scripts.

#### 8. **PressKey**

'Press Key' is used to press the hot keys such as Enter, F3, F10, Ctrl-S, etc.

#### 9. **Msgbox**

'Msgbox' is used for debugging to display the contents of a variable.

#### 10. Report

'Report' is used for customized reporter events. It is displayed in the result sheet. The report can be of four types: i) Pass, ii) Fail, iii) Done, iv) Warning.

#### 11. Strsearch

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

#### 12. Strreplace

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

#### 13. Strconcat

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

#### 14. Wait

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

#### 15. Arith

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

#### 16. Assignvalue

'Assignvalue' is used to assign dynamically generated value from the application or values stored in a particular variable to environment variables.

#### 17. CallAction

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

#### 18. Loop

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

#### 19. Convert

'Convert' is used for typecasting from one data type to another.

#### 20. Split

'Split' is used to split a string (statement) based on the delimiter.

#### 21. Strreverse

'Strreverse' is used to reverse a string.

#### 22. ImportData

Importdata' is used to import the external test data sheet into the Action1 sheet of the QTP.

#### 23. 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.

## 2.2. Naming Conventions Used for Objects

The table below provides the naming conventions followed for different Oracle objects in the Open Source Test Automation Framework.

Sl.No	Objects used in the Open Source Test Automation Framework	Window Object Class
1.	Window	OracleFormWindow
2.	FlexWindow	OracleFlexWindow
3.	Dialog	Dialog
4.	Button	OracleButton
5.	Checkbox	OracleCheckbox
6.	Textbox	OracleTextField
7.	Application	OracleApplications
8.	ListOfValues	OracleListOfValues
9.	Radiobutton	OracleRadioGroup
10.	Table	OracleTable
11.	Tree	OracleTree
12.	Page	Page
13.	Tab	OracleTabbedRegion
14.	Combobox	OracleList
15.	Calendar	OracleCalendar
16.	Navigator	OracleNavigator
17.	Notification	OracleNotification
18.	StatusLine	OracleStatusLine

Table 1: Naming Conventions Used for Objects

Naming conventions used for commonly used actions are as given below:

### 1. Click

'Click' is used to perform the click operation on objects (e.g., clicking an OracleButton).

### 2. Close

'Close' is used to perform the close operation on any open objects (e.g., closing an OracleFormWindow or Dialog box, etc.).

### 3. Select:<name>

'Select' is used to select an item from combobox, radiobutton, treeView, etc.

4. **Selectindex:<index>**

'Selectindex' is used to select an item from a list, combobox object.

5. **Check**

This is used to set a checkbox to ON state.

6. **Uncheck**

This is used to set the checkbox to OFF state.

7. **Exist:<variable>**

'Exist' is used to store the status of the object, whether the object exists or not, into a variable.

8. **Collapse:<name>**

'Collapse' is used to collapse all items under the specified node of the navigationbar or the treeview object.

9. **Set:<Text>**

'Set' is used to assign a value to the edit field.

10. **Set:d\_currenttime**

This sets the current system time to the edit field.

11. **Set:d\_currentdate**

This sets the current system date to the edit field.

12. **Set:d\_d; <value to be added/subtracted>**

This adds or subtracts the value specified to the current system date and sets the edit field to a given value.

13. **Set:d\_m; <value to be added/subtracted>**

This adds or subtracts the value specified to the current system month and sets the edit field to a given value.

14. **Set:d\_y; <value to be added/subtracted>**

This adds or subtracts the value specified to the current system year and sets the edit field to a given value.

15. **Set: dt\_<value>**

This is used to set a database parameter value to an edit field.

16. **Activate**

'Activate' is used to activate the window.

17. **Selection:<item>**

This is a check operation used to verify whether the desired item is selected or not in the combobox.

**18. Checked:<true/False>**

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

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

This is a check operation used to verify whether the specified object is enabled or not.

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

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

**21. Focused:<True/False>**

This is a check operation used to verify whether the specified object is focused or not.

**22. Text:<text/#Variable\_Name>**

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

**23. TableSearch:<colname1>; <rowval1>:: <colname2>; <rowval2>**

If it is used with the perform keyword, this is used to get column and row number based on the column names (e.g., <colname1>) and row values (e.g., <rowval1>) specified in the keyword script. If it is used with the Check keyword, it is used to verify whether the specified column name and row value are present or not in the table. If it is used with the Storevalue keyword, it is used to store the status of the search performed in an environment variable.

**24. prop\_name:<variable\_name>**

This is used to store the property value of an object in a specified variable. It is used with the storevalue keyword.

**25. set:env\_<Environment Name>**

This is used to assign the value, which is stored in environment variable to the edit field.

**26. set:dt\_Parameter**

This is used to assign the value, which is given in the external test data sheet, to the edit field.

**27. set:#<variable>**

This is used to assign the value, which is stored in the variable to the edit field.

**28. set:p\_<parameter>**

This is used to assign the value, which is stored in the input parameter to the edit field.

**29. text:p\_<parameter name>**

This is a check operation used to verify whether a string specified in a parameter variable is present or not in the table.

**30. text:env\_<environment name>**

This is a check operation used to verify whether a string specified in an environment variable is present or not in the table.

**31. text:dt\_<parameter name>**

This is a check operation used to verify whether a string specified in the action sheet variable is present or not in the table.

**32. <conversiontype>:<variable name>:<format type>**

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

**33. Expand:<node\_name>**

This is used to expand the specified tree node to display all of its child nodes.

**34. ItemExist:<item>**

This is a check operation used to verify whether the specified item is present in the object list.

**35. Toolbarselect: <item>**

This is a perform operation used to select a specified toolbar from the parent navigator.

**36. Find: <item>**

This is a perform operation used to find a particular item in the specified listofvalues.

**37. Save**

This is used to save the changes made in the navigator.

**38. Selectfunction: <name>**

This is used to select the specified function from the navigator.

**39. Columncount: <value>**

This is used to check for the count of columns present in the specified table.

**40. Approve**

This is used to approve the changes displayed in the flexwindow.

**41. Cancel**

This is used to cancel the changes displayed in the flexwindow.

**42. Opendialog**

This is used to open a dialog present in the specified textbox

**43. Restore**

This is used to restore a window or a dialog box.

**44. Create;<Folder Path/Name>/<File Path/Name>**

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

**45. Delete;<Folder Path/Name>/<File Path/Name>**

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

**46. Copy;<Source Path/Name>;<DestinationFolder Path/Name>/<Source File Path/Name>;<Destination Folder Path>**

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

**47. Move;<Source Path/Name>;<DestinationFolder Path/Name>/<Source File Path/Name>;<Destination Folder Path>**

This is used to move a folder/file from the source to the destination path.

**48. Write;<File Path/Name>;<The value to be entered>**

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

**49. 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.

**50. Append;<File Path/Name>;<text to be appended to file>**

This is used to append the data specified with the data contained in the file.

**51. OutputCheckpointName**

This is used for capturing multiple values from the database.

OutputCheckpointName is the name of the output checkpoint, inside which many output values are captured and stored in variables specified in the output checkpoint.

**52. Commit: <yes/no>**

This is used to commit the changes to the database if 'yes' is specified and is used with SQL execute keyword.

**53. CheckPointName**

This is used to check if the required data is present in the database and is used with SQLCheckpoint keyword.

### 2.3. 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>	Set:#varName
---------	------------------------	--------------

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

**Note:** Variable Naming Standards to be followed as below.

**StrVarName -**

Example - To store a username in a variable which is a string value:

Storevalue	Textbox; Username	Text: StrUserName
------------	-------------------	-------------------

**intVarName -**

Example - To store an order number, which is an integer value in a variable:

Storevalue	Textbox; OrderNo.	Text: intOrderNo
------------	-------------------	------------------

**blnVarName -**

Example - To store an exit property value, which is a boolean value in a variable:

Storevalue	Textbox; OrderNo.	Exist: blnOrderProValue
------------	-------------------	----------------------------

## 2.4. Naming Convention Used for Data Table

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

Example:

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

Here, the 'dt' signifies taking the value from the data table from the column "Parameter".

## 2.5. Naming Convention Used for Parameter

To input a value to a field from a parameter, the naming convention used is **P**.

Example:

Perform	Textbox;<textbox name>	Set: P_<Parameter>
---------	------------------------	--------------------

Here the 'P' signifies taking the value from the parameter.

## 2.6. Naming Convention Used for Environment

To input a value to a field from an environment variable, the naming convention used is **env**.

Example:

---

Perform	Textbox;<textbox name>	Set: env_<VarName>
---------	------------------------	--------------------

Here the 'env' signifies taking the value from the environment variable 'VarName'.

---

**COPYRIGHT**

*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.*