



# Open2Test Mainframe Test Automation Framework for QTP – Keyword Naming Conventions for Developers

Version 1.0

June 2011

## DISCLAIMER

*Verbatim copying and distribution of this entire article are 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 .....	5
2.3. Naming Conventions Used for Actions .....	5
2.4. Naming Conventions Used for Variables .....	6
2.5. Naming Convention Used for Datatable .....	6
2.6. Naming Convention Used for Environment .....	7

## 1. Purpose of the Document

This document provides the naming conventions followed in Open2Test Test Automation Framework for the Mainframe 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 screen object. This keyword brings a particular Window or screen to the current context, so that any operation or checking can be performed on that particular mainframe window.

#### 3. Perform

'Perform' is used to perform an operation on a particular object such as send a key operation to mainframes window, closing an open window, or typing text in a particular field. 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. MsgBox

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

#### 9. Strsearch

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

**10. Strreplace**

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

**11. Strconcat**

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

**12. Wait**

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

**13. Arith**

‘Arith’ is used to perform the arithmetic operations on the 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 datatable.

**2.2. Naming Conventions Used for Objects**

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

	Mainframe Object Class	Naming Conventions
1.	Window	TeWindow
2.	Screen	TeScreen
3.	Field	TeField

Table 1: Naming Conventions Used for Objects

**2.3. Naming Conventions Used for Actions**

**1. Enter**

‘Enter’ is used to perform the submit operation in mainframes.  
(ex. Press Enter key to submit data etc.)

**2. PF1 to PF18**

Function keys “PF1” to “PF18” are used to perform the specific operations in mainframes.  
(ex. Navigating screens, Update actions etc.)

**3. Set:<Text>**

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

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

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

**5. set:env<variable>**

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

**6. set:dt\_Parameter**

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

**2.4. Naming Conventions Used for Variables**

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

Example:

Storevalue	Field; <Field name>	Prop_name: <varName>
------------	---------------------	----------------------

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

**StrVarName -**

Ex. To store a username, which is a string value in a variable.

Storevalue	Tefield;TeUsername	Text: StrUserName
------------	--------------------	-------------------

**intVarName -**

Ex. To store an order number, which is a integer value in a variable.

Storevalue	Tefield;TeUsername	Text: intOrderNo
------------	--------------------	------------------

**blnVarName -**

Ex. To store an exit property value., which is a boolean value in a variable.

Storevalue	Tefield;TeUsername	Exist: blnOrderProValue
------------	--------------------	----------------------------

**2.5. Naming Convention Used for Datatable**

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

Example:

Perform	Field;<field_name>	Set: dt_<Parameter>
---------	--------------------	---------------------

Here, the 'dt' signifies to take the value from the datatable from the column "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 to take 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.*