

QTP - WORKING WITH GUI OBJECTS

http://www.tutorialspoint.com/qtp/qtp_gui_objects.htm

Copyright © tutorialspoint.com

Working with GUI Objects:

There are various GUI objects with which QTP interacts during the script execution. Hence it is important to know the basic methods for the key GUI objects using which we will be able to work on it effectively.

Working with Text Box

Below are the methods using which we access text box during Run Time.

- Set - Helps the tester to Set Values into the Text Box
- Click - Clicks on the Text Box
- SetSecure - Used to set the text in the password boxes securely.
- WaitProperty - Waits Till the Property value becomes true.
- Exist - Checks for the existence of the Text Box
- GetROProperty " text " - Gets the Value of the Text Box
- GetROProperty " Visible " - Returns a Boolean value if visible.

Example:

```
Browser("Math Calculator").Sync
Set Obj = Browser("Math Calculator").Page("SQR Calc").WebEdit("")

'Clicks on the Text Box
Obj.Click

'Verify if the Object Exist - Returns Boolean value
a= obj.Exist
print a

'Set the value
obj.Set "10000" : wait(2)

'Get the Runtime Object Property - Value of the Text Box
val = obj.GetROProperty("value")
print val

'Get the Run Time Object Property - Visibility - Returns Boolean Value
x= Obj.GetROProperty("visible")
print x
```

Working with Check Box

Following are some of the key methods with which one can work with Check Box.

- Set - Helps the tester to Set the checkbox value "ON" or "OFF"
- Click - Clicks on the check Box. Even checks ON or OFF but user won't be sure about the status.
- WaitProperty - Waits Till the Property value becomes true.
- Exist - Checks for the existence of the Check Box
- GetROProperty " name " - Gets the Name of the check Box

- `GetROProperty " Visible "` - Returns a Boolean value if visible

Example:

```
'To Check the Check Box
Set Obj = Browser("Calculator").Page("Gmail").WebCheckBox("PersistentCookie")
Obj.Set "ON"

'To UnCheck the Check Box
Obj.Set "OFF"

'Verifies the Existence of the Check box and returns Boolean Value
val = Obj.Exist
print val

'Fetches the Name of the CheckBox
a= Obj.GetROProperty("name")
print a

'Verifies the visible property and returns the boolean value.
x = Obj.GetROProperty("visible")
print x
```

Working with Radio Button

Following are some of the key methods with which one can work with Radio Button.

- `SelectRadioButtonName` - Helps the tester to Set the Radio Box "ON"
- Click - Clicks on the Radio Button. Even Radio Button ON or OFF but tester can't get the status.
- `WaitProperty` - Waits Till the Property value becomes true.
- `Exist` - Checks for the existence of the Radio Button
- `GetROProperty " name "` - Gets the Name of the Radio Button
- `GetROProperty " Visible "` - Returns a Boolean value if visible

Example:

```
'Select the Radio Button by name "YES"
Set Obj = Browser("Calculator").Page("Forms").WebRadioGroup("group1")
Obj.Select("Yes")

'Verifies the Existence of the Radio Button and returns Boolean Value
val = Obj.Exist
print val

'Returns the Outerhtml of the Radio Button
txt = Obj.GetROProperty("outerhtml")
print txt

'Returns the boolean value if Radio button is Visible.
vis = Obj.GetROProperty("visible")
print vis
```

Working with Combo Box

Following are some of the key methods with which one can work with Combo Box.

- `SelectValue` - Helps the tester to Select the value from the ComboBox
- Click - Clicks on the object.

- WaitProperty - Waits Till the Property value becomes true.
- Exist - Checks for the existence of the Combo Box
- GetROProperty " *Text* " - Gets the Selected Value of the Combo Box
- GetROProperty " *allitems* " - Returns all the items in the combo Box
- GetROProperty " *itemscount* " - Returns the number of items in the combo Box

Example:

```
'Get the List of all the Items from the ComboBox
Set ObjList = Browser("Math Calculator").Page("Statistics").WebList("class")
x = ObjList.GetROProperty("all items")
print x

'Get the Number of Items from the Combo Box
y = ObjList.GetROProperty("items count")
print y

'Get the text value of the Selected Item
z = ObjList.GetROProperty("text")
print z
```

Working with Buttons

Following are some of the key methods with which one can work with Buttons.

- Click - Clicks on the Button.
- WaitProperty - Waits Till the Property value becomes true.
- Exist - Checks for the existence of the Button
- GetROProperty " *Name* " - Gets the Name of the Button
- GetROProperty " *Disabled* " - Returns a boolean value if enabled/disabled

Example:

```
'To Perform a Click on the Button
Set obj_Button = Browser("Math Calculator").Page("SQR").WebButton("Calc")
obj_Button.Click

'To Perform a Middle Click on the Button
obj_Button.MiddleClick

'To check if the button is enabled or disabled.Returns Boolean Value
x = obj_Button.GetROProperty("disabled")
print x

'To fetch the Name of the Button
y = obj_Button.GetROProperty("name")
print y
```

Working with webTables

In Today's web based application, webtables have become very common and testers need to understand how Web Tables work and how to perform an action on web Tables. This Topic will help you to work with the web Tables Effectively.

Statement	Description
if statement	An if statement consists of a boolean expression followed by one or more statements.

if..else statement	An if else statement consists of a boolean expression followed by one or more statements. If the condition is True, the statements under If statements are executed. If the condition is false, Else part of the script is Executed
if...elseif..else statement	An if statement followed by one or more Elseif Statements, that consists of boolean expressions and then followed by an optional else statement , which executes when all the condition becomes false.
nested if statements	An if or elseif statement inside another if or elseif statement(s).
switch statement	A switch statement allows a variable to be tested for equality against a list of values.

- **html id** - If the table has an id tag then it is best to make use of this property.
- **innerText** - Heading of the Table.
- **sourceIndex** - Fetches the Source Index of the Table
- **ChildItemCount** - Gets the number of ChildItems present in specified Row
- **RowCount** - Gets the number of Rows in the Table
- **ColumnCount** - Gets the number of Columns in the Table
- **GetcellData** - Gets the Value of the Cell based on the column and Row Index

Example:

```

Browser("Tutorials Point").Sync
' WebTable
Obj = Browser("Tutorials Point").Page("VBScript Decisions").WebTable("Statement")
' Fetch RowCount
x = Obj.RowCount
print x

' Fetch ColumnCount
y = Obj.ColumnCount(1)
print y

' Print the Cell Data of the Table
For i = 1 To x Step 1
    For j = 1 To y Step 1
        z = Obj.GetCellData(i,j)
        print "Row ID : " & i & " Column ID : " & j & " Value : " & z
    Next
Next

'Fetch the Child Item count of Type Link in a particular Cell
z = Obj.ChildItemCount(2,1,"Link")
print z

```

Loading [MathJax]/jax/output/HTML-CSS/jax.js