1. Regular Expresssion:

Set regexp = new RegExp

Regexp.pattern =”[a-z, A-Z,0-9]”

Regexp.global = true

Regexp.ignorecase =true

Set matches = regexp.execute(str)

For each match in matches

Msgbox match.value

Next

1. **Factorial of a given number:**

F = 1

X=10

For I =1 to X

F= F\*I

Next

Msgbox F

1. **Get All characters of a string:**

A = “visweswara”

L = len(A)

For I=1 to L

B= mid(A,I,1)

Msgbox B

Next

1. **Frequency of Characters:**

S = “Visweswara”

For I= 1 to Len(S)

Character = mid (S,I, 1)

Check = split(S, Character, -1)

Arrlen = ubound(Check)

Msgbox “Character ”&Character&”appears”&Arrlen&” no of times”

Next

1. **Close the latest opened browser:**

Set obj = Description.create

Obj(“micclass”).value = “Browser”

Set browserObjList = Desktop.childobject(Obj)

Index = browserObjList.Count -1

Browser(“creationtime:=”&Index).Close

Set obj = Nothing

Set browserObjList = Nothing

1. **Length of a String without using len function:**

Str = “Visweswara”

I =1

Do

Char = mid (Str,I,1)

If Char =”” Then

Exit Do

Else

I= I+1

End If

Loop Until Char=””

Msgbox I-1

Or

Set reg = new RegExp

Str = “Visweswara”

Reg.pattern = “a-z,A-Z”

Reg.global =true

Reg.ignorecase =true

Set matches =reg.execute(Str)

For each match in matches

C = C+1

Next

Msgbox C-1

1. StringReverse w/o using function:

inputString = InputBox("Enter the string")

For i = Len(inputString) to 1  Step-1

    var= Mid(inputString,  i  , 1)

    reverseString = reverseString & var

Next

Msgbox reverseString

1. **Swap without using 3rd variable:**

A=10

B=20

A=A+B

B=A-B

A=A-B

Or

Str1 =”Infosys” Str2 =”Tech”

Len1 = len(Str1)

Len2 =len(Str2)

Str1 = Str1&Str2

Str2 = Left(Str1, Len1)

Str1 = Right(Str1,Len2)

1. **Find .txt files in the folder:**

Set Fso = CreateObject(“Scripting.FileSystemObject”)

folderPath = “path..”

Set obj = Fso.GetFolder(folderPath)

Set objFiles = obj.Files

For each objFile in objFiles

A= inst(1, objFile.Name,”txt”,1)

If A>0 Then

Msgbox objFile.Name

End If

Next

1. **How many alpha characters present in a string?**

Str =”Viswes1234”

Len=len(Str)

I=0

For J=1 to L

If Not isNumeric(mid(str,J,1))) Then

I =I +1

End If

Next

1. **Sort an array**
2. for a = UBound(ArrayOfTerms) - 1 To 0 Step -1
3. for j= 0 to a
4. if ArrayOfTerms(j)>ArrayOfTerms(j+1) then
5. temp=ArrayOfTerms(j+1)
6. ArrayOfTerms(j+1)=ArrayOfTerms(j)
7. ArrayOfTerms(j)=temp
8. end if
9. next
10. next

12. UFT 14 Features

UFT is now offered in 3 new flavors. UFT Ultimate, UFT Enterprise, and UFT Pro.

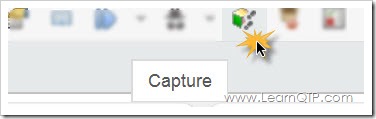
* If you previously had a UFT License, it will now be renamed to UFT Enterprise.
* If you previously had a LeanFT License, it will now be renamed to UFT Pro (LeanFT).
* While UFT Enterprise and UFT Pro are available as seat license as well as concurrent license, UFT Ultimate is available only as a concurrent license.
* Starting with UFT 14, UFT will now support device id based concurrent licenses in addition to concurrent license based on license server IP address.
* UFT RunTime Engine licensing remains the same.

Data Generator or Test Combinations Generator for GUI tests

Getting relevant test data has always been one of the challenges of automated testing. Till now, one has to rely on [3rd party tools for test data](https://www.learnqtp.com/generate-test-data-for-software-testing/) needs.

Starting with UFT 14, UFT has introduced a new tool inside the UFT IDE called Test Combinations Generator.

New “Capture” Mode

Using the newly introduced **Capture** button on the toolbar, you can capture all the objects in a selected area of your application.

I would want to know and explore the use cases for this feature and how is it better than **Navigate and Learn** which is already available in the object repository manager.

 UFT Pro (LeanFT) goes cross platform

LeanFT is now supported across Windows, Mac OS and Linux for your web and mobile testing needs.

For web testing, you would get all capabilities you had in Windows version.

You should, however, note in case of mobile testing you still need to create scripts on Windows OS. Only **execution** is supported on Mac OS and Linux.

8. LeanFT for Selenium

Selenium has seen a high rate of adoption in the past few years. HPE has developed a solution **LeanFT for Selenium**which is bundled with HPE UFT Pro (LeanFT). It enables Selenium users to create and maintain Selenium scripts easily. **LeanFT4Selenium** is available for Windows, Linux, and Mac, and supports the most common browsers.

13. **Random number and random string generation:**

Dim max,min  
max=100  
min=1  
Randomize ‘To avoid getting the same number every time  
response.write(Int((max-min+1)\*Rnd+min))

Function RandomString( ByVal strLen )

Dim str, min, max

Const LETTERS = "abcdefghijklmnopqrstuvwxyz0123456789"

min = 1

max = Len(LETTERS)

Randomize

For i = 1 to strLen

str = str & Mid( LETTERS, Int((max-min+1)\*Rnd+min), 1 )

Next

RandomString = str

End Function

14. ByVal and ByRef:

Function GetValue(ByVal var)

var = var + 1

End Function

Dim x: x = 5

'Pass the variable x to the GetValue function ByVal

Call GetValue(x)

Call Rhino.Print("x = " & CStr(x))

o/p;

x = 5

In other words, when we passed the variable x (ByVal) to the function GetValue, we were simply passing a copy of the variable x. When GetValue executes, var stores a copy of the variable x and increments itself by 1. Therefore, because what we are passing to GetValue is a copy of x, it cannot be modified.

unction GetReference(ByRef var)

var = var + 1

End Function

Dim x: x = 5

'Pass the variable x to the GetReference function ByRef

Call GetReference(x)

o/p;

x = 6

When the function GetReference executes, var becomes a reference of x, and therefore, any changes made to var would impact x. So if var increments itself by 1, so would x. If var becomes 0 (zero), so would x.

15. Checkpoints:

<https://www.guru99.com/quick-test-professional-qtp-tutorial-13.html>

16. How to click on weblinks using descriptive programming?

Function To\_Click(strLink)  
Set MyDescription = Description.Create()   
MyDescription("Class Name").Value = "Link"   
MyDescription("html tag").Value = "A"   
Set CheckLinks = Browser("Itinerary").Page("Itinerary").ChildObject s(MyDescription)   
NoOfChildObjs = Checkboxes.Count   
For Counter=0 to NoOfChildObjs-1   
strgetROProp = CheckLinks(Counter).GetROproperty("text")  
If strgetROProp = strLink Then  
CheckLinks(Counter).Click  
To\_Click = True  
Exit For  
Else  
To\_Click = False  
End If  
Next   
End Function

**Ex1**: If there are multiple signin links on the page, using child objects method we can click on any signin link.

'\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Dim oLinkDescription

Dim oLinks

Dim lnkCount

Set oLinkDescription=Description.Create

oLinkDescription("name").value="signin"

set oLinks=browser("micclass:=Browser").page("micclass:=Page").ChildObjects(oLinkDescription)

if oLinks.count <> 0 then

oLinks(0).click

End If

'\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

17. **Click on a link inside a webtable**

Browser(<BROWSER NAME>).Page(<PAGE NAME>).WebTable(<WEBTABLE NAME>).ChildItem(<ROW>, <COL>, "WebLink", 0).click

18. Descriptive programming:

**Types of Programmatic Descriptions**

1. **Static**: - You list the set of properties and values that describe the object directly in a VBScript statement.

a. Direct specification of description in script

Browser(“micclass:=Browser”).Page(micclass:=page”).Link(“name:=Login”).Click  
b. Assigning description to the variables and use that variables in script  
g\_MainBrowser = “micclass:=Browser”  
g\_MainPage = “micclass:=Page”  
g\_Lnk\_Login = “name:=Login”  
Browser(g\_MainBrowser).Page(g\_MainPage).Link(g\_Lnk\_Login).Click

2. **Dynamic**: - You add a collection of properties and values to a Description object, and then enter the Description object name in the statement.

Set oBrowser = Description.create

oBrowser (“micclass”).value=”Browser”  
oBrowser (“name”).value= “Google”  
Set oPage = Description.create  
oPage (“micclass”).value=”Page”  
oPage (“name”).value= “Google”  
Set oLink = Description.create  
oLink (“name”).value= “Login”  
oLink (“index”).value= 1  
Browser(oBrowser).Page(oPage).Link(oLink).click

Using the Static type to enter programmatic descriptions directly into your statements may be easier for basic object description needs. However, in most cases, using the Dynamic type provides more power, efficiency, and flexibility.

**Using Regular Expressions**

You can directly use regular expressions in the properties.

Ex1:  
Public const LoginPage\_Edit\_UserName = “name:=UserName.\*”

Ex2:  
Browser(“micclass:=Browser”).Page(micclass:=page”).Link(“name:=Login.\*”).Click

Ex3:  
Set oLink = Description.create  
oLink (“name”).value= “Login.\*”  
oLink (“index”).value= 1  
Browser(“micclass:=Browser”).Page(micclass:=page”).Link(oLink).click  
By default regular expressions for description object is enable. To disable Regular expressions for description object  
Set oLink = Description.create  
oLink (“name”).value= “Login.\*”  
oLink (“name”).RegularExpression=False  
oLink (“index”).value= 1

<https://www.qtpsudhakar.com/2009/02/descriptive-programming-in-qtp.html>