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

' Purpose: Determines which function to call based on keyword

'

' Inputs: strObject -

' strParent -

' Returns:

'

' Author: Ritesh Kawadkar

' Cretesd At: 6/4/2018

' Modified By:

' Reason For Change:

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

Public Function Keyword\_verifyText(strParam1, strParam2, strParam3, strParam4, strScreenShot, TS\_Name, strFunctionName, strTCStepID, stepDescription, TC\_Name)

If CStr(strParam1) = CStr(strParam2) Then

Keyword\_verifyText = True

GlobalReporting(stepDescription)="Text verification passed: " & strParam1 & " = " & strParam2

Else

Keyword\_verifyText = False

GlobalReporting(stepDescription)="Text verification failed: " & strParam1 & " NOT= " & strParam2

'Print "Keyword\_verifyText Step Failed"

WriteLog "STEP FAILED","Keyword\_verifyText Step Failed"

End If

End Function

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

' Purpose: Determines which function to call based on keyword

'

' Inputs: strObject -

' strParent -

' Returns:

'

' Author: Ritesh Kawadkar

' Cretesd At: 6/4/2018

' Modified By:

' Reason For Change:

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

Public Function Keyword\_assertText(strParam1, strParam2, strParam3, strParam4, strScreenShot, TS\_Name, strFunctionName, strTCStepID, stepDescription, TC\_Name)

If Ucase(CStr(Trim(strParam1))) = UCase(CStr(Trim(strParam2))) Then

Keyword\_assertText = True

GlobalReporting(stepDescription)="Text verification passed: " & strParam1 & " = " & strParam2

Else

Keyword\_assertText = False

'Print "Keyword\_assertText Step Failed param1 = " & strParam1 & " param2 = " & strParam2

WriteLog "STEP FAILED","Keyword\_assertText Step Failed param1 = " & strParam1 & " param2 = " & strParam2

GlobalReporting(stepDescription)="Text verification failed: " & strParam1 & " NOT= " & strParam2

End If

End Function

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

' Purpose: Determines which function to call based on keyword

'

' Inputs: strObject -

' strParent -

' Returns:

'

' Author: Ritesh Kawadkar

' Cretesd At: 6/4/2018

' Modified By:

' Reason For Change:

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

Public Function Keyword\_assertNull(strParam1, strParam2, strParam3, strParam4, strScreenShot, TS\_Name, strFunctionName, strTCStepID, stepDescription, TC\_Name)

If IsNUll(strParam1) and IsNUll(strParam2) Then

Keyword\_assertNull = True

GlobalReporting(stepDescription)="Null verification passed"

Else

Keyword\_assertNull = False

'Print "Keyword\_assertNull Step Failed param1 = " & strParam1 & " param2 = " & strParam2

WriteLog "STEP FAILED","Keyword\_assertNull Step Failed param1 = " & strParam1 & " param2 = " & strParam2

GlobalReporting(stepDescription)="Null verification failed: " & strParam1 & " NOT= " & strParam2

End If

End Function

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

' Purpose: Determines which function to call based on keyword

'

' Inputs: strObject -

' strParent -

' Returns:

'

' Author: Ritesh Kawadkar

' Cretesd At: 6/4/2018

' Modified By:

' Reason For Change:

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

Public Function Keyword\_verifyNull(strParam1, strParam2, strParam3, strParam4, strScreenShot, TS\_Name, strFunctionName, strTCStepID, stepDescription, TC\_Name)

If IsNUll(strParam1) and IsNUll(strParam2) Then

Keyword\_assertNull = True

GlobalReporting(stepDescription)="Null verification passed"

Else

Keyword\_assertNull = False

'Print "Keyword\_assertNull Step Failed param1 = " & strParam1 & " param2 = " & strParam2

WriteLog "STEP FAILED","Keyword\_assertNull Step Failed param1 = " & strParam1 & " param2 = " & strParam2

GlobalReporting(stepDescription)="Null verification failed: " & strParam1 & " NOT= " & strParam2

End If

End Function

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

' Purpose: Determines which function to call based on keyword

'

' Inputs: strObject -

' strParent -

' Returns:

'

' Author: Ritesh Kawadkar

' Cretesd At: 6/4/2018

' Modified By:

' Reason For Change:

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

Public Function Keyword\_executeQTPCommand(strParam1, strParam2, strParam3, strParam4, strScreenShot, TS\_Name, strFunctionName, strTCStepID, stepDescription, TC\_Name)

If Not IsNull(strParam1) and IsNull(strParam2) and IsNull(strParam3) and IsNull(strParam4) Then

Execute(" """ & strParam1 & " """)

ElseIf Not IsNull(strParam1) and Not IsNull(strParam2) and IsNull(strParam3) and IsNull(strParam4) Then

Execute(strParam1 & " """ & strParam2 & """")

ElseIf IsNull(strParam1) and Not IsNull(strParam2) and Not IsNull(strParam3) and IsNull(strParam4) Then

Execute(strParam1 & " """ & strParam2 & """, """ & strParam3 & """")

ElseIf Not IsNull(strParam1) and Not IsNull(strParam2) and IsNull(strParam3) and Not IsNull(strParam4) Then

Execute(strParam1 & " """ & strParam2 & """, """ & strParam3 & """, """ & strParam4 & """")

End If

Keyword\_executeQTPCommand = True

GlobalReporting(stepDescription)="Executed QTP Command Successfully"

End Function

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

' Purpose: Determines which function to call based on keyword

'

' Inputs: strObject -

' strParent -

' Returns:

'

' Author: Ritesh Kawadkar

' Cretesd At: 6/4/2018

' Modified By:

' Reason For Change:

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

Public Function Keyword\_run(strParam1, strParam2, strParam3, strParam4, strScreenShot, TS\_Name, strFunctionName, strTCStepID, stepDescription, TC\_Name)

Set oShell = CreateObject("Wscript.ShelL")

If Not IsNull(strParam1) and IsNull(strParam2) and IsNull(strParam3) and IsNull(strParam4) Then

oShell.Run """" & strParam1 & """"

'oShell.Run strParam1

ElseIf Not IsNull(strParam1) and Not IsNull(strParam2) and IsNull(strParam3) and IsNull(strParam4) Then

oShell.Run """" & strParam1 & """, """ & strParam2 & """"

ElseIf IsNull(strParam1) and Not IsNull(strParam2) and Not IsNull(strParam3) and IsNull(strParam4) Then

oShell.Run """" & strParam1 & """, """ & strParam2 & """, """ & strParam3 & """"

ElseIf Not IsNull(strParam1) and Not IsNull(strParam2) and IsNull(strParam3) and Not IsNull(strParam4) Then

oShell.Run """" & strParam1 & """, """ & strParam2 & """, """ & strParam3 & """, """ & strParam4 & """"

End If

Set oShell = Nothing

Keyword\_run = True

GlobalReporting(stepDescription)="Run command passed successfully"

End Function

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

' Purpose: Determines which function to call based on keyword

'

' Inputs: strObject -

' strParent -

' Returns:

'

' Author: Ritesh Kawadkar

' Cretesd At: 6/4/2018

' Modified By:

' Reason For Change:

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

Public Function Keyword\_wait(strParam1, strParam2, strParam3, strParam4, strScreenShot, TS\_Name, strFunctionName, strTCStepID, stepDescription, TC\_Name)

If Not IsNull(strParam1) Then

Wait CInt(strParam1)

GlobalReporting(stepDescription)="Wait statement executed for " & strParam1 & " sec."

Else

Wait 2

GlobalReporting(stepDescription)="Wait statement executed for 2 sec."

End If

Keyword\_wait = True

End Function

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

' Purpose: Determines which function to call based on keyword

'

' Inputs: strObject -

' strParent -

' Returns:

'

' Author: Ritesh Kawadkar

' Cretesd At: 6/4/2018

' Modified By:

' Reason For Change:

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

Public Function Keyword\_dateDiff(strParam1, strParam2, strParam3, strParam4, strScreenShot, TS\_Name, strFunctionName, strTCStepID, stepDescription, TC\_Name)

If IsNull(strParam1) Then

Keyword\_dateDiff = False

GlobalReporting(stepDescription)="DateDiff inavlid parameter count"

WriteLog "STEP FAILED","DateDiff inavlid parameter count"

Exit Function

End If

If Not IsNull(strParam2) and IsNull(strParam3) Then

diff = Abs(DateDiff(strParam1, now, strParam3))

GlobalReporting(stepDescription)="DateDiff between dates " & now & " -- " & strParam3 & "is = " & diff & strParam1

End If

If Not IsNull(strParam2) and Not IsNull(strParam3) Then

diff = Abs(DateDiff(strParam1, strParam2, strParam3))

GlobalReporting(stepDescription)="DateDiff between dates " & strParam2 & " -- " & strParam3 & "is = " & diff & strParam1

End If

If Not IsNull(strParam2) and IsNull(strParam3) Then

diff = Abs(DateDiff(strParam1, strParam2, now))

GlobalReporting(stepDescription)="DateDiff between dates " & strParam2 & " -- " & now & "is = " & diff & strParam1

End If

If Not IsNull(strParam4) Then

If Left(strParam4, 5)="OUT>>" Then

strParam4 = Replace(strParam4, "OUT>>", "")

End If

Set objKeyword = new Connections

'check if the variable nam ealready exists

objKeyword.fnConnectToIOXL GlobalDictionary("TestDataPath"), TS\_Name

strQuery = "Select Count(\*) from [Output$] WHERE [TestCases] = '" & TC\_Name & "' and [Variables] = '" & strParam4 & "'"

objKeyword.fnExecuteQuery(strQuery)

count = objKeyword.objRecordSet.Fields(0).Value

Set objKeyword = Nothing

Set objKeyword = new Connections

objKeyword.fnConnectToXLForWriting GlobalDictionary("TestDataPath"), TS\_Name

If count>0 Then

'UPDATE the variable's value

strQuery = "UPDATE [Output$] SET [Values] = '" & diff & "' WHERE [TestCases] = '" & TC\_Name & "' and [Variables] = '" & strParam4 & "'"

Else

'INSERT new variable

strQuery = "INSERT INTO [Output$] VALUES('" & TC\_Name & "','" & strParam4 & "','" & diff & "')"

End If

objKeyword.fnWriteToExcel(strQuery)

Set objKeyword = Nothing

End If

Keyword\_dateDiff = True

End Function

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

' Purpose: Determines which function to call based on keyword

'

' Inputs: strObject -

' strParent -

' Returns:

'

' Author: Ritesh Kawadkar

' Cretesd At: 6/4/2018

' Modified By:

' Reason For Change:

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

Public Function Keyword\_removeTimeFromApprovals(strParam1, strParam2, strParam3, strParam4, strScreenShot, TS\_Name, strFunctionName, strTCStepID, stepDescription, TC\_Name)

If IsNull(strParam1) Then

Keyword\_removeTimeFromApprovals = False

GlobalReporting(stepDescription)="RemoveTimeFromApprovals inavlid parameter count"

WriteLog "STEP FAILED","RemoveTimeFromApprovals inavlid parameter count"

Exit Function

Else

strParam1 = Left(strParam1, 175)

Set objRegEx = New RegExp

objRegEx.Global = True

objRegEx.IgnoreCase = True

objRegEx.Pattern = "[\d|:]\* by "

newStr = objRegEx.Replace(strParam1, "")

End If

If Not IsNull(strParam2) Then

If Left(strParam2, 5)="OUT>>" Then

strParam2 = Replace(strParam2, "OUT>>", "")

Set objKeyword = new Connections

'check if the variable nam ealready exists

objKeyword.fnConnectToIOXL GlobalDictionary("TestDataPath"), TS\_Name

strQuery = "Select Count(\*) from [Output$] WHERE [TestCases] = '" & TC\_Name & "' and [Variables] = '" & strParam2 & "'"

objKeyword.fnExecuteQuery(strQuery)

count = objKeyword.objRecordSet.Fields(0).Value

Set objKeyword = Nothing

Set objKeyword = new Connections

objKeyword.fnConnectToXLForWriting GlobalDictionary("TestDataPath"), TS\_Name

If count>0 Then

'UPDATE the variable's value

strQuery = "UPDATE [Output$] SET [Values] = '" & newStr & "' WHERE [TestCases] = '" & TC\_Name & "' and [Variables] = '" & strParam2 & "'"

Else

'INSERT new variable

strQuery = "INSERT INTO [Output$] VALUES('" & TC\_Name & "','" & strParam2 & "','" & newStr & "')"

End If

objKeyword.fnWriteToExcel(strQuery)

Set objKeyword = Nothing

ElseIf Left(strParam2, 4)="GD>>" Then

strParam2 = Replace(strParam2, "GD>>", "")

If GlobalDictionary.Exists(strParam2) Then

GlobalDictionary(strParam2) = newStr

Else

GlobalDictionary.Add strParam2, newStr

End If

End If

End If

Keyword\_removeTimeFromApprovals = True

GlobalReporting(stepDescription)="RemoveTimeFromApprovals statement executed"

End Function

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

' Purpose: Determines which function to call based on keyword

'

' Inputs: strObject -

' strParent -

' Returns:

'

' Author: Ritesh Kawadkar

' Cretesd At: 6/4/2018

' Modified By:

' Reason For Change:

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

Public Function Keyword\_buildApprovalFormat(strParam1, strParam2, strParam3, strParam4, strScreenShot, TS\_Name, strFunctionName, strTCStepID, stepDescription, TC\_Name)

If Not IsNull(strParam1) Then

If Left(strParam1, 5)="OUT>>" Then

strParam1 = Replace(strParam1, "OUT>>", "")

Set objKeyword = new Connections

'check if the variable nam ealready exists

objKeyword.fnConnectToIOXL GlobalDictionary("TestDataPath"), TS\_Name

strQuery = "Select Count(\*) from [Output$] WHERE [TestCases] = '" & TC\_Name & "' and [Variables] = '" & strParam1 & "'"

objKeyword.fnExecuteQuery(strQuery)

count = objKeyword.objRecordSet.Fields(0).Value

strQuery = "Select Values from [Input$] WHERE [TestCases] = '" & TC\_Name & "' and [Variables] = 'NAME'"

objKeyword.fnExecuteQuery(strQuery)

strName = objKeyword.objRecordSet.Fields(0).Value

newStr = "Final Approval: " & Split(now(), " ")(0) & strName & " +++ Prod Approval: " & Split(now(), " ")(0) & strName & " +++ Pres Approval: " & Split(now(), " ")(0) & strName & " +++"

Set objKeyword = Nothing

Set objKeyword = new Connections

objKeyword.fnConnectToXLForWriting GlobalDictionary("TestDataPath"), TS\_Name

If count>0 Then

'UPDATE the variable's value

strQuery = "UPDATE [Output$] SET [Values] = '" & newStr & "' WHERE [TestCases] = '" & TC\_Name & "' and [Variables] = '" & strParam1 & "'"

Else

'INSERT new variable

strQuery = "INSERT INTO [Output$] VALUES('" & TC\_Name & "','" & strParam1 & "','" & newStr & "')"

End If

objKeyword.fnWriteToExcel(strQuery)

Set objKeyword = Nothing

ElseIf Left(strParam1, 4)="GD>>" Then

strParam1 = Replace(strParam1, "GD>>", "")

Set objKeyword = new Connections

objKeyword.fnConnectToIOXL GlobalDictionary("TestDataPath"), TS\_Name

strQuery = "Select [Values] from [Input$] WHERE [TestCases] = '" & TC\_Name & "' and [Variables] = 'NAME'"

objKeyword.fnExecuteQuery(strQuery)

strName = " " & objKeyword.objRecordSet.Fields(0).Value

newStr = "Final Approval: " & Split(now(), " ")(0) & strName & " +++ Prod Approval: " & Split(now(), " ")(0) & strName & " +++ Pres Approval: " & Split(now(), " ")(0) & strName & " +++"

Set objKeyword = Nothing

If GlobalDictionary.Exists(strParam1) Then

GlobalDictionary(strParam1) = newStr

Else

GlobalDictionary.Add strParam1, newStr

End If

End If

End If

Keyword\_buildApprovalFormat = True

GlobalReporting(stepDescription)="BuildApprovalFormat statement executed"

End Function

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

' Purpose: Determines which function to call based on keyword

'

' Inputs: strObject -

' strParent -

' Returns:

'

' Author: Ritesh Kawadkar

' Cretesd At: 6/4/2018

' Modified By:

' Reason For Change:

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

Public Function Keyword\_inStr(strParam1, strParam2, strParam3, strParam4, strScreenShot, TS\_Name, strFunctionName, strTCStepID, stepDescription, TC\_Name)

If IsNull(strParam1) and IsNull(strParam2) Then

Keyword\_inStr = False

GlobalReporting(stepDescription)="BuildApprovalFormat inavlid parameter count"

WriteLog "STEP FAILED","BuildApprovalFormat inavlid parameter count"

Exit Function

Else

If InStr(1, strParam1, strParam2)>0 Then

'Passed

Else

Keyword\_inStr = False

GlobalReporting(stepDescription)="InStr comparision failed: " & strParam1 & " NOT = " & strParam2

WriteLog "STEP FAILED","InStr comparision failed: " & strParam1 & " NOT = " & strParam2

Exit Function

End If

End If

Keyword\_inStr = True

GlobalReporting(stepDescription)="InStr statement executed sucessfully: " & strParam1 & " = " & strParam2

End Function

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

' Purpose: Determines which function to call based on keyword

'

' Inputs: strObject -

' strParent -

' Returns:

'

' Author: Ritesh Kawadkar

' Cretesd At: 6/4/2018

' Modified By:

' Reason For Change:

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

Public Function Keyword\_maths(strParam1, strParam2, strParam3, strParam4, strScreenShot, TS\_Name, strFunctionName, strTCStepID, stepDescription, TC\_Name)

If Not IsNull(CInt(strParam2)) and Not IsNull(CInt(strParam3)) Then

Select case UCase(strParam1)

case "ADD"

result = CInt(strParam2) + CInt(strParam3)

case "SUBSTRACT"

result = CInt(strParam2) - CInt(strParam3)

case "MULTIPLY"

result = CInt(strParam2) \* CInt(strParam3)

case "DIVIDE"

result = CInt(strParam2) / CInt(strParam3)

End select

End If

If strParam4 <> "" Then

If Left(strParam4, 5)="OUT>>" Then

strParam4 = Replace(strParam4, "OUT>>", "")

End If

Set objKeyword = new Connections

objKeyword.fnConnectToIOXL GlobalDictionary("TestDataPath"), TS\_Name

strQuery = "Select Count(\*) from [Output$] WHERE [TestCases] = '" & TC\_Name & "' and [Variables] = '" & strParam4 & "'"

count = 0

objKeyword.fnExecuteQuery(strQuery)

count = objKeyword.objRecordSet.Fields(0).Value

Set objKeyword = Nothing

Set objKeyword = new Connections

objKeyword.fnConnectToXLForWriting GlobalDictionary("TestDataPath"), TS\_Name

If count>0 Then

'UPDATE the variable's value

strQuery = "UPDATE [Output$] SET [Values] = '" & result & "' WHERE [TestCases] = '" & TC\_Name & "' and [Variables] = '" & strParam4 & "'"

Else

'INSERT new variable

strQuery = "INSERT INTO [Output$] VALUES('" & TC\_Name & "','" & strParam4 & "','" & result & "')"

End If

objKeyword.fnWriteToExcel(strQuery)

Set objKeyword = Nothing

End If

Keyword\_maths = True

GlobalReporting(stepDescription)="Maths Fn" & strParam1 & " passed"

End Function