

QTP - ERROR HANDLING

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

Copyright © tutorialspoint.com

What is Error Handling?

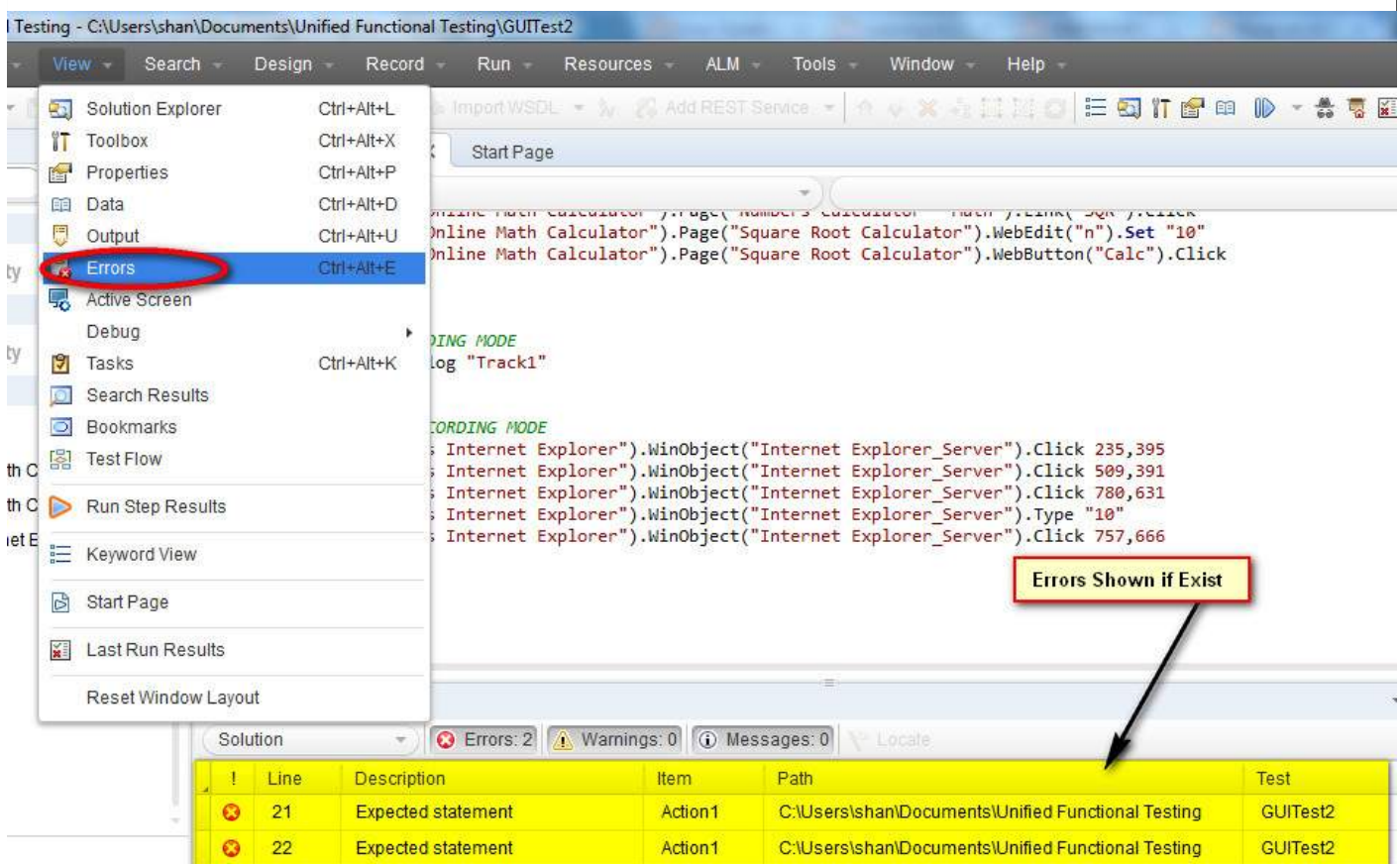
There are various ways on handling errors in QTP. There are three possible kinds of error type one would encounter while working with QTP.

- Syntax Errors
- Logical Errors
- Run Time Errors

Error Types:

Syntax Errors:

Syntax errors are the typos or a piece of the code that does not confirm with the VBScripting language grammar. Syntax errors occur at the time of compilation of code and cannot be executed until the errors are fixed. To verify the syntax one use the keyboard shortcut as Ctrl+F7 and the result is displayed as shown below. If the window is NOT displayed one can navigate to "View" → "Errors".



Logical Errors:

If the script is syntactically correct but it produces unexpected results. Logical error usually does not interrupt the execution but produces incorrect results. Logical errors could occur due to variety of reasons, viz- wrong assumptions or misunderstanding of the requirement and sometimes incorrect program logics using *do – while* instead of *do – Until* or Infinite Loops.

One of the ways to detect a logical error is to perform peer reviews and also verifying the QTP output file/result file to ensure the tool has performed what it has intended to do.

RunTime Errors:

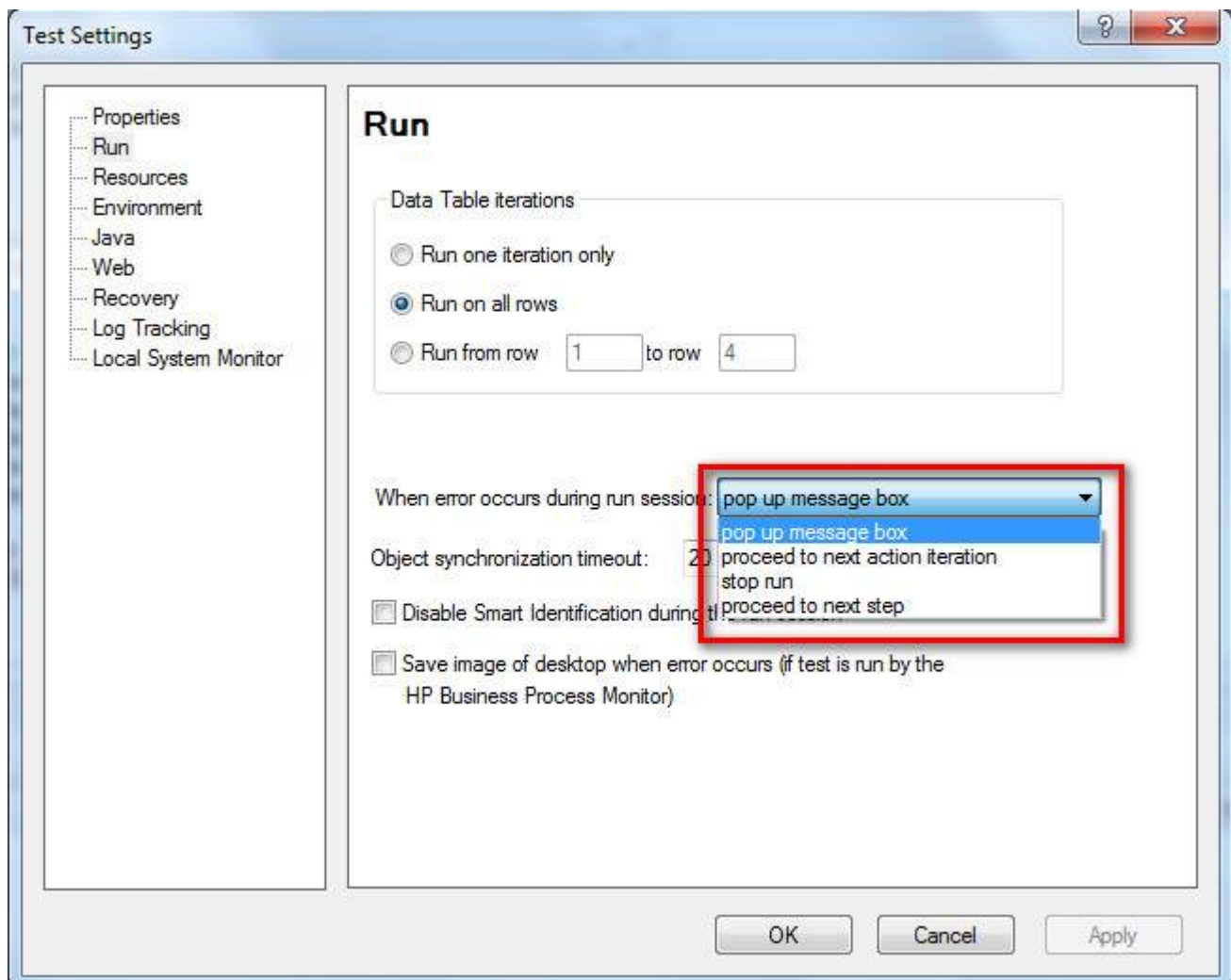
As The name states, this kind of Error happens during Run Time. The reason for such kind of errors is that the script trying to perform something but it is unable to do so and the script usually stops as it is unable to continue with the execution. Classic Examples for Run Time Errors are,

- File NOT found but the script trying to read the file.
- Object NOT found but script is trying to act on that particular object.
- Dividing a number by Zero.
- Array Index out of bounds while accessing array elements.

Handling Run-Time Errors:

There are various ways to handle errors in the code.

1. Using Test Settings - Error handling can be defined the Test Settings by Navigating to "File" >> "Settings" >> "Run" Tab as shown below. We can select any of the specified settings and click "OK".



2. Using On Error Statement - On Error statement is used to notify the VBScript engine of intentions to handle the run-time errors by tester, rather than allowing the VBScript engine to display error messages that are not user friendly.

- On Error Resume Next - On Error Resume Next informs the VBScript engine to process executing the next line of code when an error is encountered.
- On error Goto 0 - This helps the testers to turn off the error handling.

3. Using Err Object - Error object is an inbuilt object within VBScript that captures the run time error number and error description with which we will be able to debug the code easily.

- Err.Number - The Number property returns or Sets a numeric value specifying an error. If Err.Number value is 0 then No error had occurred.
- Err.Description - The Description property returns or sets a brief description about an error.
- Err.Clear - The Clear method resets the Err object and clears all the previous values associated with it.

Example:

```
'Call the function to Add two Numbers
Call Addition(num1,num2)

Function Addition(a,b)

    On error resume next
    If NOT IsNumeric(a) or IsNumeric(b) Then
        Print "Error number is " & err.number & " and description is : " &
err.description
        Err.Clear
        Exit Function
    End If

    Addition = a+b

'disables error handling
On Error Goto 0

End function
```

Using Exit Statement - Exit Statements can be used along with Err object to exit from a test or action or iteration based on the Err.Number value. Let us see each one of those Exit statements in detail.

- ExitTest - Exits from the entire QTP test no matter what the run-time iteration settings are.
- ExitAction - Exits the current action.
- ExitActionIteration - Exits the current iteration of the action.
- ExitTestIteration - Exits the current iteration of the QTP test and proceeds to the next iteration.

5. Recovery Scenarios - Upon encountering an error, recovery scenarios are triggered based on certain conditions and it is dealt in detail in a separate chapter.

6. Reporter Object - Reporter Object helps us to report an event to the run results. It helps us to identify if the concerned action/step is pass/fail.

```
'Syntax: Reporter.ReportEventEventStatus, ReportStepName, Details, [ImageFilePath]
```

```
'Example
```

```
Reporter.ReportEvent micFail, "Login", "User is unable to Login."
```

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