**What is Framework?**

A framework is considered to be a combination of set protocols, rules, standards and guidelines that can be incorporated or followed as a whole so as to leverage the benefits of the scaffolding provided by the Framework.

### ****Test Automation Framework****

A “Test Automation Framework” is scaffolding that is laid to provide an execution environment for the automation test scripts. The framework provides the user with various benefits that helps them to develop, execute and report the automation test scripts efficiently. It is more like a system that has created specifically to automate our tests.

**Advantage of Test Automation framework**

1. Reusability of code
2. Maximum coverage
3. Recovery scenario
4. Low cost maintenance
5. Minimal manual intervention
6. Easy Reporting

### ****Types of Test Automation Framework****

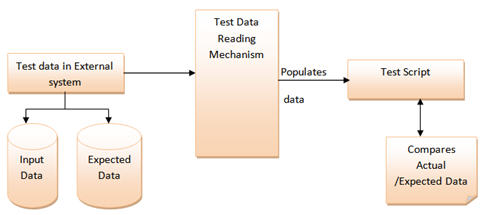
1. Data Driven Testing Framework
2. Keyword Driven Testing Framework
3. Hybrid Testing Framework

**Data Driven Testing Framework**

While automating or testing any application, at times it may be required to test the same functionality multiple times with the different set of input data. Thus, in such cases, we can’t let the test data embedded in the test script. Hence it is advised to retain test data into some external data base outside the test scripts.

Data Driven Testing Framework helps the user segregate the test script logic and the test data from each other. It lets the user store the test data into an external database. The external databases can be property files, xml files, excel files, text files, CSV files, ODBC repositories etc. The data is conventionally stored in “Key-Value” pairs. Thus, the key can be used to access and populate the data within the test scripts.

***Note***: The test data stored in an external file can belong to the matrix of expected value as well as matrix of input values.

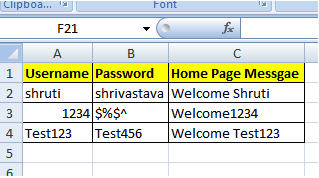
[](http://cdn2.softwaretestinghelp.com/wp-content/qa/uploads/2014/11/Test-Automation-Frameworks-5.jpg)

**Example:**

Let us understand the above mechanism with the help of an example.

Let us consider the “Gmail – Login” Functionality.

**Step 1:** First and the foremost step are to create an external file that stores the test data (Input data and Expected Data). Let us consider an excel sheet for instance.

[](http://cdn2.softwaretestinghelp.com/wp-content/qa/uploads/2014/11/Test-Automation-Frameworks-6.jpg)

------------

**Step 2:** The next step is to populate the test data into Automation test Script. For this purpose several API’s can be used to read the test data.

|  |  |
| --- | --- |
| 1 | public void readTD(String TestData, String testcase)throws Exception { |
| 2 | TestData=readConfigData(configFileName,"TestData",driver); | |

|  |  |
| --- | --- |
| 3 | testcase=readConfigData(configFileName,"testcase",driver); |
| 4 | FileInputStreamtd\_filepath = new FileInputStream(TestData); | |

|  |  |  |
| --- | --- | --- |
| 5 | Workbook td\_work =Workbook.getWorkbook(td\_filepath); | |
| 6 | Sheet td\_sheet = td\_work.getSheet(0); |

|  |  |
| --- | --- |
| 7 | if(counter==0) |
| 8 | { |

|  |  |  |
| --- | --- | --- |
| 9 | for (int i = 1,j = 1; i <= td\_sheet.getRows()-1; i++){ | |
| 10 | | if(td\_sheet.getCell(0,i).getContents().equalsIgnoreCase(testcase)){ | |

|  |  |  |
| --- | --- | --- |
| 11 | | startrow = i; |
| 12 | arrayList.add(td\_sheet.getCell(j,i).getContents()); | | |

|  |  |  |
| --- | --- | --- |
| 13 | testdata\_value.add(td\_sheet.getCell(j+1,i).getContents());}} | |
| 14 | | for (int j = 0, k = startrow +1; k <= td\_sheet.getRows()-1; k++){ |

|  |  |  |
| --- | --- | --- |
| 15 | | if(td\_sheet.getCell(j,k).getContents()==""){ |
| 16 | arrayList.add(td\_sheet.getCell(j+1,k).getContents()); | |

|  |  |  |  |
| --- | --- | --- | --- |
| 17 | testdata\_value.add(td\_sheet.getCell(j+2,k).getContents());}} | | |
| 18 | | } |

|  |  |  |
| --- | --- | --- |
| 19 | counter++; | |
| 20 | } |

### ****Keyword Driven Testing Framework****

The Keyword driven testing framework is an extension to Data driven Testing Framework in a sense that it not only segregates the test data from the scripts, it also keeps the certain set of code belonging to the test script into an external data file.

These set of code are known as Keywords and hence the framework is so named. Key words are self-guiding as to what actions needs to be performed on the application.

The keywords and the test data are stored in a tabular like structure and thus it is also popularly regarded as Table driven Framework. Take a notice that keywords and test data are entities independent of the automation tool being used.

**Example Test case of Keyword Driven Test Framework**

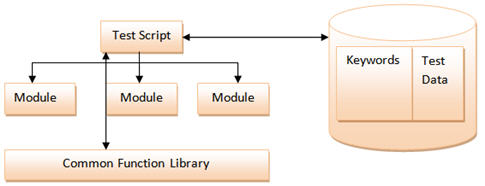
[](http://cdn.softwaretestinghelp.com/wp-content/qa/uploads/2014/11/Test-Automation-Frameworks-8.jpg)

In the above example keywords like login, clickLink and verifyLink are defined within the code.  
Depending upon the nature of application keywords can be derived. And all the keywords can be reused multiple times in a single test case. Locator column contains the locator value that is used to identify the web elements on the screen or the test data that needs to be supplied.

All the required keywords are designed and placed in base code of the framework.

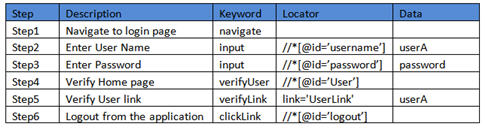
### ****Hybrid Testing Framework****

As the name suggests, the Hybrid Testing Framework is a combination of more than one above mentioned frameworks. The best thing about such a setup is that it leverages the benefits of all kinds of associated frameworks.

[](http://cdn2.softwaretestinghelp.com/wp-content/qa/uploads/2014/11/Test-Automation-Frameworks-9.jpg)

**Example of Hybrid Framework**

Test sheet would contain both the keywords and the Data.

[](http://cdn2.softwaretestinghelp.com/wp-content/qa/uploads/2014/11/Test-Automation-Frameworks-10.jpg)

In the above example, keyword column contains all the required keywords used in the particular test case and data column drives all the data required in the test scenario. If any step does not need any input then it can be left empty.