USEFUL LINKS:

1.<http://www.testingtools.com/>

2.<http://www.seleniumhq.org/docs/>

SELENIUM SOFTWARES USED FOR SELENIUM APPLICATIONS:

3.<http://www.softwaretestinghelp.com/most-popular-web-application-testing-tools/>

WEBTESTING GUIDE:

<http://www.softwaretestinghelp.com/web-application-testing/>

SELENIUM GUIDE:

http://www.softwaretestinghelp.com/sitemap/

<http://www.softwaretestinghelp.com/>

This link includes topics like:

## 

* [Home](http://www.softwaretestinghelp.com/)
* [Get FREE Updates](http://www.softwaretestinghelp.com/subscribe/)
* [Testing RESOURCES](http://www.softwaretestinghelp.com/resources/)
* [QA Testing TRAINING](http://www.softwaretestinghelp.org/?utm_campaign=sthmenu)
* [Premium eBook](http://www.softwaretestinghelp.org/software-testing-help-ebook-new/?utm_campaign=sthmenu)
* [FREE eBooks](http://www.softwaretestinghelp.com/practical-software-testing-new-free-ebook-download/)
* [ALL Articles](http://www.softwaretestinghelp.com/sitemap/)

### 1- Automation feasibility analysis

In this section you have to think from different perspective. The main objective of this phase will be to check feasibility of automation.

So your main focus will be on below points.

1- Which test case can be automated and how we can automate them

2- Which module of your application can be tested and which can not be automated

3- Which tools we can use for our application (like Selenium,QTP,Sahi,OATS, Telrik etc) and which tools will be best of our application

4- Take following factors into consideration like Team size,Effort and cost involved for tools which we will use.

2- Test Plan/Test Design

This phase plays very important role in Automation test life cycle. In this phase you have to create a Test plan by considering below point into considerations.

1- Fetch all the manual test case from test management tool that which TC has to automate.

2- Which framework to use and what will be advantage and disadvantage of the framework which we will use.

3- Create a test suite for Automation test case in Test Management tool.

4- In test plan you can mention background, limitation, risk and dependency between application and tools.

5- Approval from client/ Stack holders.

### 3- Environment Setup/Test lab setup

By name itself you can understand that we need to setup machine or remote machine where our test case will execute.

1- In this section you can mention how many machine you want.

2- What should be the configuration in terms of hardware and software.

### 4-Test Script development/ Automation test case development

In this phase you have to start develop automation script and make sure all test script is running fine and should be stable enough.

1- Start creating test script based on your requirement

2- Create some common method or function that you can reuse throughout your script

3- Make your script easy, reusable,well structured and well documented so if third person check your script then he/she can understand your scripts easily.

4- Use better reporting so in case of failing you can trace your code

5- Finally review your script and your script should be ready before consumption.

### 5-Test script execution

Now its time for execution of test scripts, in this phas you have to execute all your test script.

Some points to remember while execution

1- Your script should cover all the functional requirement as per testcase.

2- Your script should be stable so it should run in multiple environment and multiple browsers (depends on your requirement)

3- You can do batch execution also if possible so it will save time and effort.

4- In case of failure your script should take screen shots.

5- If test case is failing due to functionality, you have to raise a bug/defect.

### 6- Generate test result / Analyses of result

This is the last phase of Automation test life cycle in which we will gather test result and will share with team/client/stack holders.

1- Analyze the output and calculate how much time it take to complete the testcase.

2- You should have good report generation like XSLT report, TestNG report, ReporterNG etc

1.

1.WHAT IS SELENIUM? HOW IT IS USED

2.TYPES OF SELENIUM SOFTWARES

3.LANGUAGES USED IN SELENIUM AND INSTALLATIONS OF JDK(JAVA),ECLIPSE,FIREFOX(FIREBUG,FIREPATH).

4.GENRAL DEFINITION OF API,IDE,DIFFERENCES BETWEEN SE AND SE RC.

1:

Q1..What is software Testing and how is it used ?

It is used to identify the bugs ,errors.

Also used to test client requirements like whether designing is accurate or not.

And alos used to check whether client requirements are satisfied or not..

Q2.What are the types of software used to test in selenium?

<http://www.testingtools.com/test-automation/>

# Software Test Automation Tools

Automated software testing is becoming more and more important for many software projects in order to automatically verify key functionality, test for regressions and help teams run a large number of tests in a short period of time. Many teams (especially larger projects) still require a significant amount of manual functional testing in addition to automated testing, either because of the lack of sufficient resources or skills to automate all tests.

There are various tools that help software teams build and execute automated tests. Many teams are actively using unit tests as part of their development efforts to verify critical parts of their projects such as libraries, models and methods. Historically, testing user interfaces of desktop-based applications via automated tests have been more challenging, and currently available tools for this are usually commercial and quite expensive.

With the growing number of web-based applications this is changing, however, as verifying and testing web-based interfaces is easier and there are various tools that help with this, including free open source projects. Please see below for a list of popular and useful tools, projects, books and resources to get started with automated software testing.

## Automated Web Testing Tools7 Tools

Automatically testing your web application is a good way to ensure that new versions of your application don't introduce bugs and regressions. Automation of your web application testing also allows your development team to make changes and refactor code with more confident, as they can quickly verify the functionality of the application after every change.

However, actually building automated tests for web applications can be challenging because the user interface of your application might change regularly, because of incompatibilities between browsers and because you usually need to support various server or client platforms. The following tools make it easier to build and execute automated tests for your web application.

Open Source

### [Selenium](http://docs.seleniumhq.org/)

Selenium is a popular automated web testing tool and helps you automate web browsers across different platforms. Selenium has the support of some of the largest browser vendors who have taken steps to make Selenium a native part of their browser.

Open Source

### [Watir](http://watir.com/)

Watir is a set of Ruby libraries for automating web browsers and allows you to write tests that are easy to read and maintain. Watir drives browsers the same way people do (it clicks links, fills in forms, presses buttons etc.) and also checks results such as whether expected text appears on the page.

Open Source

### [Windmill](http://www.getwindmill.com/)

Windmill is a web testing tool designed to help testers automate and debug web applications. It comes with a cross-browser test recorder, JavaScript integration and an interactive shell to automate web browsers.

Commercial

### [Ranorex](http://www.ranorex.com/)

Ranorex allows you to automate your web application testing (among other things) and both record user interactions and play them back to execute your tests. Ranorex is one of the more popular commercial tools to build and run automated web and GUI tests.

Open Source

### [SoapUI](http://www.soapui.org/)

SoapUI is a cross-platform functional testing tool. It has been specifically designed to help automatically test APIs such as SOAP and REST interfaces to ensure the interoperability of different applications.

Open Source

### [Sahi](http://sahi.co.in/)

Sahi is a tool for automation of web application testing. Sahi is available as a free open source edition as well as a commercial Pro edition.

Commercial

### [Tellurium](http://www.te52.com/)

Tellurium is a web automation tool that allows you to design and write your automated tests using plain English without any scripting or programming experience.

## Automated GUI Testing Tools5 Tools

Building robust automated GUI tests for desktop applications (e.g. on Windows or Mac systems) is quite difficult, as small changes to the user interface can often result in broken tests. The following tools help you build and execute robust GUI tests for various platforms and operating systems.

Commercial

### [Squish](http://www.froglogic.com/squish/gui-testing/)

Squish is a GUI testing tool for various platforms, including QT, native Windows and Mac applications. Squish allows testers and developers to build automated tests using familiar scripting languages such as JavaScript, Perl, Python and Ruby.

Commercial

### [Ranorex](http://www.ranorex.com/)

Ranorex allows you to automate your desktop applications (among other things) and both record user interactions and play them back to execute your tests. Ranorex is one of the more popular commercial tools to build and run automated GUI and web tests.

Commercial

### [TestComplete](http://smartbear.com/products/qa-tools/automated-testing-tools)

TestComplete is an automated testing tool for the Windows platform. It allows you to record, script and run GUI tests for applications built using different frameworks and languages, such as .NET or C++.

Commercial

### [Test Studio](http://www.telerik.com/automated-testing-tools)

Test Studio is an automated functional and load testing tool that helps you test applications on various platforms built using different frameworks and tools.

Commercial

### [eggPlant](http://www.testplant.com/products/eggplant/)

eggPlant is a GUI test automation tool for professional software applications and enterprise teams. It can be used to automate different application types, such as .NET, Java and Flash applications.

## Unit Testing Frameworks9 Tools

See below for a list of popular unit testing frameworks and tools for major platforms and programming languages. These frameworks can be used by programmers to test specific functionality in libraries and applications. Unit tests can then be used to automatically test new versions and builds as part of an automated build system or deployment process.

.NET, Open Source

### [NUnit](http://www.nunit.org/)

NUnit is a unit-testing framework for all .Net languages. It was initially ported from JUnut to .NET and has been redesigned to take advantage of many .NET language features.

.NET, Open Source

### [xUnit.net](http://xunit.codeplex.com/)

xUnit.net is a community-focused unit testing tool for the .NET Framework written by the original inventor of NUnit. xUnit.net is the latest technology for unit testing C#, F#, VB.NET and other .NET languages.

Python, Open Source

### [PyUnit / unittest](http://docs.python.org/2/library/unittest.html)

The Python unit testing framework, sometimes referred to as "PyUnit", is a Python language version of JUnit. It's part of the Python framework and supports test automation, sharing of setup and shutdown code for tests and various other features.

Java, Open Source

### [JUnit](http://junit.org/)

JUnit is a simple unit testing framework to write repeatable tests in Java. JUnit has been important in the development of test-driven development and is one of the standard testing frameworks for Java developers.

Java, Open Source

### [TestNG](http://testng.org/doc/)

TestNG is a Java testing framework inspired by JUnit and NUnit and introduces some new functionalities that make it more powerful and easier to use. TestNG is designed to cover multiple categories of tests, including unit, functional, end-to-end and integration tests.

PHP, Open Source

### [PHPUnit](http://www.phpunit.de/)

PHPUnit is a popular framework for unit testing in PHP projects. It provides both a framework that makes it easier to write tests as well as the functionality to easily run, execute and analyze tests and results.

PHP, Open Source

### [Symfony Lime](http://symfony.com/)

Lime is a unit and functional testing framework built for the popular Symfony PHP web application framework. The framework is designed to have readable output from tests, including color formatting, by following the Test Anything Protocol.

Ruby, Open Source

### [Test::Unit](http://www.ruby-doc.org/stdlib-2.0/libdoc/test/unit/rdoc/Test/Unit.html)

Ruby comes with its own standard unit testing framework as part of the Test::Unit namespace and can be used to define basic pass/fail tests and group tests. The framework also comes with tools to run single or whole groups of tests.

Ruby, Open Source

### [RSpec](http://rspec.info/)

RSpec is a testing tool for the Ruby programming language. Born under the banner of behaviour-driven development, it is designed to make test-driven development more productive and enjoyable.

## Automated Testing Cloud Services

Cloud testing services allow you to run your automated web, mobile and unit tests in different environments and on multiple machines without having to build your own testing infrastructure. The below list of services provides a good overview of popular online Selenium and continuous integration services that are easy and affordable to use.

|  |  |  |  |
| --- | --- | --- | --- |
| Service | Type | Description | Pricing |
| [Sauce Labs](https://saucelabs.com/) | Selenium & Mobile | Cross-browser Selenium & mobile testing | Starting @ $149/month |
| [TestingBot](http://testingbot.com/) | Selenium | Cross-browser Selenium testing | Starting @ $40/month |
| [Gridlastic](https://gridlastic.com/) | Selenium | Cross-browser Selenium testing | Starting @ Free |
| [CircleCI](https://circleci.com/) | Continuous Integration | Continuous integration service & unit testing | Starting @ $19/month |
| [Tddium](https://www.tddium.com/) | Continuous Integration | Continuous integration service & unit testing | Starting @ $15/month |
| [CloudBees](http://www.cloudbees.com/) | Continuous Integration | Continuous integration service & unit testing | Starting @ Free |
| [Mailosaur](https://mailosaur.com/) | Email Testing | Service to integrate emails in automated tests | Starting @ Free |

What are the software used in Selenium and how to install

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

1) Install Java

a) we are maintain the jdk version

b) Right application run administartor

c) Set java path

i) open c drive - program files - java - jdk folder - open bin - copy path - then that path right click on my computer - select properties - select advanced system setting - environment variables - select path - click on edit and then paste the copy folder path

ii) open c drive - program files - java - jre folder - open bin - copy path - then that path right click on my computer - select properties - select advanced system setting - environment variables - select path - click on edit and then paste the copy folder path

d) open command promat - type javac then press enter - java press enter

2) IDE ( Intergrated developement environment )

a) in our java we are maintain the Eclipse

a) Extract Eclipse

b) That extract folder copy and paste in c driver

c) open folder select eclipse icon application file - right on that one send to desktop short cut

3) selenium java folder

1) Open url : <http://www.seleniumhq.org/download/>

2) select java language click on download

3) extract folder - copy folder paste in c driver

3) mozilla fire fox download

4) Adding Add on features to mozilla

1) fire Bug

i) open mozilla - type firebug for firefox - click on add

2) Firepath

i) Open Mozilla - type firepath for firefox

3) Selenium IDE

i) Open Mozilla - type the selenium ide for firefox

Difference between SE web driver and SE RC.

Selenium-WebDriver makes direct calls to the browser using each browser’s native support for automation. How these direct calls are made, and the features they support depends on the browser you are using. Information on each ‘browser driver’ is provided later in this chapter.

For those familiar with Selenium-RC, this is quite different from what you are used to. Selenium-RC worked the same way for each supported browser. It ‘injected’ javascript functions into the browser when the browser was loaded and then used its javascript to drive the AUT within the browser. WebDriver does not use this technique. Again, it drives the browser directly using the browser’s built in support for automation.

You may, or may not, need the Selenium Server, depending on how you intend to use Selenium-WebDriver. If you will be only using the WebDriver API you do not need the Selenium-Server. If your browser and tests will all run on the same machine, and your tests only use the WebDriver API, then you do not need to run the Selenium-Server; WebDriver will run the browser directly.

There are some reasons though to use the Selenium-Server with Selenium-WebDriver.

* You are using Selenium-Grid to distribute your tests over multiple machines or virtual machines (VMs).
* You want to connect to a remote machine that has a particular browser version that is not on your current machine.
* You are not using the Java bindings (i.e. Python, C#, or Ruby) and would like to use HtmlUnit Driver.

API:

API (application program interface) is a set of routines, protocols, and tools for building software applications. The API specifies how software components should interact and APIs are used when programming graphical user interface (GUI) components

IDE:

An integrated development environment (IDE) is a programming environment that has been packaged as an application program, typically consisting of a code editor, a[compiler](http://whatis.techtarget.com/definition/compiler), a debugger, and a graphical user interface ([GUI](http://searchwindevelopment.techtarget.com/definition/GUI)) builder. The IDE may be a standalone application or may be included as part of one or more existing and compatible applications. The [BASIC](http://whatis.techtarget.com/definition/BASIC-Beginners-All-purpose-Symbolic-Instruction-Code) programming language, for example, can be used within Microsoft Office applications, which makes it possible to write a WordBasic program within the Microsoft Word application. IDEs provide a user-friendly framework for many modern programming languages, such as [Visual Basic](http://searchwindevelopment.techtarget.com/definition/Visual-Basic), [Java](http://searchsoa.techtarget.com/definition/Java), and [PowerBuilder](http://searchsoa.techtarget.com/definition/PowerBuilder).

Languages used for se automation scripting:

Java, c#(Dot net),perl,ruby,php.

SELENIUM SOFTWARE USED FOR SELENIUM APPLICATIONS:

It’s a testing methodology focused on web applications i.e. applications hosted on the web.

With web application testing, issues such as website functionality, security, accessibility, usability, compatibility and performance are uncovered before the web application is released in public.

1) Websites are testing - selenium , uft

2) GUI application are testing - uft (unfied functional testing - QTP )

3) Webservices are tesing - SoapUI , Soapui Pro

### Load, Stress and Performance Testing Tools:

Here are some important tools used to test the performance, load and stress of the web application. These tools ensure that your web application will run under high performance and less load & stress.

1. Apache JMeter
2. NeoLoad
3. LoadRunner
4. LoadUI
5. Loadster
6. Load impact
7. Wapt
8. Qtest
9. Telerik test studio

### #1. Apache JMeter

It is an Apache Open source load testing tool, written in Java 6+ and supports all platforms. Recently, Apache released the stable version of JMeter “v2.11” that supports all platforms.

Basically, JMeter is used for load testing and to analysing and measuring the performance of system/application.



Also, this tool is helpful in testing JDBC database connections (FTP, LDAP, Web services, JMS, HTTP, HTTPS, TCP connections) and OS Native processes. It analyses overall performance under different load on a server, group of servers & network place.

Capability to check the performance of the SOAP, LDAP, Message-oriented middleware (MOM) via JMS, Mail (SMTP(S), POP3(S) and IMAP(S)), MongoDB (NoSQL), and Native commands or shell scripts. Its strong GUI design helps in fast building of Test Plan and debugging process.

Official Website: http://jmeter.apache.org/

Download link: [JMeter download](http://jmeter.apache.org/download_jmeter.cgi)

### # 2. NeoLoad

It is a Neotys load and stress testing tool for Windows, Linux, and Solaris, available in English and French with the latest version 4.2, designed to measure, analyze and improve the performance of the website.Although, load increases on the website due to simultaneous access of many users, this tool will be helpful in checking the performance of the website under heavy load.



Testing process by using this tool will be very fast, efficient, and frequent. By using this tool we can get very fast result. Give surety that your website will be accurate and reliable to fulfill the business and user expectations

Official Website: http://www.neotys.com/product/overview-neoload.html

Download Link: [Neoload download](http://www.neotys.com/support/download-neoload.html)

### #3. LoadRunner

It is a load testing tool for Windows and Linux, given by HP to test the web and other applications very efficiently before it goes to end user hands. It is available in stable version 12.0 in many languages.



LoadRunner is very much helpful in determining the performance and result of the web application under heavy load (load due to thousands of users accessing the web application at a same time).

It handles various protocols for load testing: .NET Record/Replay, Database, DCOM, GUI Virtual Users, Java Record/Replay, Network, Oracle E-Business, Remote Access, Remote Desktop, Rich Internet Applications, SAP, SOA, Web 2.0, Web and Multimedia and Wireless.

Official Website: http://www8.hp.com/in/en/software-solutions/loadrunner-load-testing/

Download Like: [LoadRunner Download](http://h30499.www3.hp.com/t5/HP-LoadRunner-and-Performance/Download-HP-LoadRunner-and-enhance-your-experience-with-the-tips/ba-p/5969843#.VBaUp2OTJ1M)

### #4. LoadUI

A load testing tool “LoadUI” latest version is 2.7, written in Java, JaveFX, and Groovy language, and supports all platforms. Mostly,support web services and SopaUI 5.0 (functional testing tool).



This latest version corrected some issues that were there in the previous version (v2.6) of LoadUI, issue: “File “.bat, .sh and .command” line ending issue” and the About LoadUI dialog box does not display always.

Now, it is available in Pro version. LoadUI Pro drag-and-drop powerful interface facilitates you to test the overall load efficiently in real-time environment.

Official Website: http://www.loadui.org/

Download Link: [LoadUI download](http://smartbear.com/products/qa-tools/web-service-load-testing/creating-web-service-load-tests/)

### #5. Loadster

A commercial load testing tool to test websites, web applications/services (HTTP web services) and to simulate and guess how web applications will perform under heavy load, given by Loadster, supports Linux, Mac, and Windows. This full-features tool works on actual web applications/services cookies, sessions, custom header, dynamic form of data, etc…



Loadster is used to test the performance, stability, and scalability of web applications/services and websites. It simulates numbers of users, networking with the site and collects stats for each virtual user distinctly. Load testing assists to determine performance bottlenecks, expect and avoid crashes, and confirm your application can handle high traffic events.

Official Website: http://www.loadsterperformance.com/

Download Link: [Loadster download](http://www.loadsterperformance.com/pricing/)

### #6. Load impact

An online load testing tool used to test websites, web applications, mobile applications and APIs under heavy load for all platforms.

Load impact uses to test all applications/services and websites online, instead of setting-up offline to test the application. Testing process is very fast and simple. You can also repeat the test very easily in very less time.



Official Website: http://loadimpact.com/

Download Link: [LoadImpact download](http://loadimpact.com/pricing)

### #7. Wapt

A load and stress testing tool works on all Windows, provides an easy and cheapest way to test websites, like; business applications websites, mobile websites, web portals, etc.Testing requires 500MB of free disc space, browsers (Microsoft IE 6 or higher, Firefox 3.0 or higher, and Google Chrome), and OS of 64 bit version. It works on secure HTTPS websites, dynamic content and RIA applications under data-driven mode



Official Website: http://www.loadtestingtool.com/index.shtml

Download Link: [Wapt Download](http://www.loadtestingtool.com/download.shtml)

### #8. Qtest

A web load testing tool analyses the application completely and accurately, given by Quotium Technologies SA, supports all Windows platforms. Its original User interface (UI) is easy to use and understand, and used as an OnDemand hosted solution or an OnPremise application.



Official Website: http://www.quotium.com/qtest/qtest\_overview/

Download Link: [Qtest Download](http://web.quotium.com/acton/form/5332/000b:d-0001/0/index.htm)

### #9. Telerik Test Studio

A software testing tool is to test web and desktop applications of all Windows OS, developed by Telerik. It tests the functionality, performance, and load of web, desktop, and mobile applications. The tool offers a plugin for Visual Studio and a standalone app that used to test the cross-browsing issue and have the same file-format.

Test Studio supports, languages; HTML, AJAX, Silverlight, ASP.NET MVC, JavaScript and WPF andrun test on browsers; Internet Explorer, Firefox, Safari and Chrome.



Official Website: http://www.telerik.com/teststudio

Download Link: [Telerik Test Studio Download](http://www.telerik.com/download/teststudio)

———————————————————————————–

### Web Functional/Regression testing Tools:

Many websites/web applications suffer from bugs due to its improper functionality, behavior, usability, security, etc. For those websites/web applications, web testing tools do magic (these Testing tools take out all bugs from websites/web applications in limited time period and limited cost). Determining and fixing a bug, drafting a bug sheet, and determining priority and severity is hard during development phase of websites. Most of the bugs come due to functional issue of the website/web application. So, to come out of functional issues, we have tried to provide some web functional/regression testing tools.

1. Selenium
2. TestCafe
3. CasperJS
4. Screenster
5. SoapUI
6. Sahi
7. Watin
8. Watir
9. QTP (UFT)
10. Ranorex

### #1. Selenium

Selenium is one of the most popular automated testing suite. Selenium is designed in a way to support and encourage automation testing of functional aspects of web based applications and a wide range of browsers and platforms.

Selenium suite is comprised of 4 basic components; Selenium IDE, Selenium RC, WebDriver, Selenium Grid. Selenium IDE is Firefox add-on for record-and-playback web application tests. WebDriver directly communicates with the web browser and uses its native compatibility to automate.

Official website: http://www.seleniumhq.org/

Download Link: [Selenium Download](http://www.seleniumhq.org/download/)

### #2. TestCafe v14.1.3

It is aDevExpress web testing framework to test all web applications of Windows, MacOS, and Linux OS.



It is always in top in terms of web application testing, generates very clear and simply tests as possible, uses all browsers that support HTML 5 to record and run functional tests across operating systems (Windows, Mac and Linux, and on remote computers) and mobile devices (iPhone, iPad, Android and Windows Phone). It has capabilities to simultaneously run tests on multiple browsers and multiple machines.

Official Website: http://testcafe.devexpress.com/

Download Link: [TestCafe Download](http://testcafe.devexpress.com/Download)

### #3. CasperJS v1.1-beta3

It is an easy to use open source navigation tool for Windows, MacOS, and Linux OS, used for scripting & testing the PhantomJS and SlimerJS (Gecko) with the help of its useful and valuable functions, methods and syntactic sugar, written in Java script language. Basically, it works for forms, links, page screenshot, remote DOM, events sign-in process, etc. Also, downloads binary and other resources, writes functional test suites and save it in a JUnit XML format.



Official Website: http://casperjs.org/

Download Link: [CasperJS download](http://casperjs.org/)

### #4. Screenster

It is an UI based functional and regression testing tool for web applications, works on all Windows OS that requires all major browsers and Java6 or high. A very good screen validator uses to give 10x productivity without a single line of code. During operation, it takes screenshot on each and every step and compares them to baseline, permits verification of changes or lack of changes to UI, provides full access to Selenium API when needed.



Official website: http://www.creamtec.com/products/screenster/index.html

Download link: [Screenster Download](http://www.creamtec.com/products/screenster/index.html)

### #5. SoapUI

A cross-platform free open source functional testing tool for service-oriented architectures (SOA) and representational state transfers (REST), written in Java language. Its user-friendly interface acquires various features, mainly used for API testing facilitates to easily and quickly perform functional/regression and load testing, supports all protocols and technologies related to the particular application. Its commercial version (SoapUI Pro) is mainly used to test functionality of the application.



Official Website: http://www.soapui.org/

Download Link: [SOAPUI Download](http://www.soapui.org/Downloads/download-soapui-pro-trial.html)

### #6. Sahi

It is an open-source cross-platform web application testing tool, written in Java and JavaScript, released under an Open Source Apache License 2.0. It is available in Pro v5.1.2 is used to test multiple browsers applications (holds many AJAX and dynamic content) under web 2.0.



Sahi Pro identifies elements very easily and also works on application includes dynamic ids (\_near and \_in) and applications develop using ExtJS, ZK, Dojo, YUI, etc.

Official Website: http://sahipro.com/

Download Link: [Sahi Download](http://sahipro.com/sahi-open-source/)

### #7. Watin (Web Application Testing in .Net)

A cross-platform web application testing tool, latest version is 2.1, and developed in C# language. Due to the influence of Watir, Watin tested a web application written in .Net in December 2005.



Now, it has become a user-friendly and stable framework with much functionality, supports all main HTML elements and dialogs (modal and modeless). Also, supports native Page and Control model, testing of AJAX website, screenshots of webpages creation process, and frames (cross domain) and iframes. It take care of popular popup dialog box: alert, confirm, login etc… and works on Internet Explorer 6, 7, 8, 9 and FireFox 2 and 3. You can download it more than 120.000 times

Official Website: http://watin.org/

Download Link: [WatiN Download](http://sourceforge.net/projects/watin/files/WatiN%202.x/)

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

### #8. Watir (Web Application Testing in Ruby)

An open-source (BSD) cross-platform web application testing tool, pronounced as water*. It* is an easy and flexible automation tool of Ruby libraries for web browsers automation. Although, it is of Ruby libraries, supports applications written in any language and just like other languages, Ruby enables you to link to databases, export XML files, read files and spreadsheets, and organize your code as reusable libraries.



Watir drives Internet Explorer, Firefox, Chrome, Opera and Safari browsers in the same ways how others do.

Official Website: http://watir.com/

Download Link: [Watir Download](http://watir.com/installation/)

### #9. QTP (UFT)

QTP (Quick Test Professional) a Windows based software testing tool used to test the applications on the web or desktop, best for “Functional” and “Regression” testing, given by Hewlett Packard (HP). Its UI is called as an Integrated Development environment (IDE) comes with various functionalities which motivates tester towards testing. QTP uses VB Script language to run the script and supports Web, Java (Core and Advanced), .Net, Flex, Web Services, WPF, Delphi, Power Builder, Stingray 1, Terminal Emulator, SAP, Oracle, Siebel, PeopleSoft, Windows Mobile, VisualAge Smalltalk, Silverlight and mainframe terminal emulators.



QTP latest version is UFT 12.0. UFT 12.0 provides easier and smaller package to download the product, that’s why; UFT 12.0 installation process has become easier than previous one. QTP was only working on Windows operating system, but UFT 12.0 supports Safari on a remote Mac, all useful browsers(Internet Explorer6, 7, 8, 9, 10, 11, Firefox 3.0.X, 3.5, 3.6 to v24 and Google Chrome), desktop technologies (Delphi XE2, Stingray 12, Flex, and Web Dynpro ABAP for Netweaver 7.31),and Visual Studio (2010 and 2012).

Official Website: http://www8.hp.com/in/en/software-solutions/unified-functional-testing-automation/

Download Link: [QTP/UFT Download](http://ssl.www8.hp.com/in/en/ssl/dlc/secure_software.html?prodNumber=T6508DAE&siebelid=3-3HIDW9E&lang=en&cc=in&resouce_type=Trial%20Software&cpt_resource_type=hp.resource_type.software_trial.label&sectionid=software&simpletitle=hp%20unified%20functional%20testing%20software&subbu=tsg.software&parentPageName=3.0&analytics_page_name=3.0&parentUrl=http%3A//www8.hp.com/in/en/software-solutions/unified-functional-testing-automation/&compURI=tcm%3A188-1641349&fv=FLEX2%20SW3&metrics_asset_value=eb&bu=tsg&st=/in/en/software-solutions/unified-functional-testing-automation&as=software&wsi=r11374&cu=false)

### #10. Ranorex

A Windows based GUI testing tool to test the desktop, web and mobile based applications and to handle any kind of modern and dynamic GUI application, given by Ranorex GmbH. Checks the functionality of the application that involves tester and developer both. Involvement of both; tester and developer become reason of reliable testing and finding bugs very quickly. It supports technologies; .NET, Winforms, WPF, Java, SAP, HTML5, Flash, Flex, Silverlight, iOS, Android, Windows Apps (Native/Hybrid), etc. Although Ranorex is user-friendly and affordable, it can be used by any organization and small testing team.



Official Website: http://www.ranorex.com/

Download Link: [Ranorex Download](http://www.ranorex.com/free-trial.html)

———————————————————————————–

### Link Manager Testing Tools:

All websites are made up of many links or we can say all websites are based on links. Any broken and wrong link can take you to some wrong page and you will get wrong information about the website. So, it is very much important to correct the link using link manager testing tool,

1. SpringTrax
2. LinkTiger
3. LinkScan

### #1. SpringTrax

It is a cross-platform link checker provided by SpringTrax Inc., expert in finding broken links, and also expert in discovering, fixing, and stopping 404 errors. It evaluates that why the traffic has been lost. It monitors each and every visitors using JavaScript tracker code, and analyses every 404 error instantaneously.



Official Website: https://springtrax.com/home.htm

Download link: [SpringTrax Download](http://springtrax.com/features-pricing/default.htm)

### #2. LinkTiger

It is also a link checker, works on e-mail alerts, dashboard, and rich custom reports, supports Linux, Mac, Windows, and Windows Phone platforms. Its scan features can scan PDF, CSS, Flash and MS Office files, flash-animation.



LinkTiger displays status of all links present on the website with an intuitive dashboard. For each website the dashboard displays three pie-charts with the pages status, the link status and the error types of the dead link.

Official Website: http://www.linktiger.com/

Download link: [LinkTiger Download](http://www.linktiger.com/link-checker-pricing.php)

### #3. LinkScan v12.4

It is a strong link checker and website mapping tool for Windows, Unix, Linux and Mac OS X, provides very strong test automation skills for the web based applications, available for Unix and Microsoft Windows operating systems, highly customizable and fast. It is also a good HTML validator.

web testing21

Official Website: http://www.elsop.com/

Download link: [LinkScan Download](http://www.elsop.com/download/)

———————————————————————————–

### Web Site Security Testing Tools:

Most of the virus comes to the system through application which is uploaded on internet that can corrupt the system and the application which is on system. We cannot stop using application which is on internet, so, the best thing is; use those applications securely by using website security testing tool.

1. NTOSpider
2. Brakeman
3. SiteDigger
4. Netsparker
5. NMap
6. OWASP

### #1. NTOSpider

It is a Windows based web security tool, provides full security to the web applications/services, mobile, and rich internet applications (RIA’s). The most important thing is, it scans your application fully in less time, provides full security to the system at very less cost.



Official website: http://www.ntobjectives.com/security-software/ntospider-application-security-scanner/

Download link: [NTOSpider Download](http://www.ntobjectives.com/security-software/ntospider-application-security-scanner/ntospider-trial-download-request/)

### #2. Brakeman v2.6.1

It is an open source scanner, checks Ruby on Rails apps for security vulnerabilities, available for Jenkins/Hudson and works on Rails 2.x, 3.x, and 4.x. It statically examines Rails application code to discovery security problems at any stage of development.

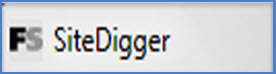


Official Website: http://brakemanscanner.org/

Installation path: [Brakeman Download](http://brakemanscanner.org/docs/install/)

### #3. SiteDigger v3.0

It is expert in examining Google’s cache, errors, configuration problems, proprietary information, and remarkable security nuggets on websites.



Having very good user interface and provides real time result in result page with ability to save signature selection and result set, Google API License Key is not required to access this tool. It runs on All Windows OS required Microsoft .NET Framework v3.5.

Official Website: http://www.mcafee.com/us/downloads/free-tools/sitedigger.aspx

Download link: [SiteDigger Download](http://www.mcafee.com/us/downloads/free-tools/termsofuse.aspx?url=http://b2b-download.mcafee.com/products/tools/foundstone/sitedigger3.msi)

### #4. Netsparker

A cross-platform web application security scanner is helpful in detecting and reporting website/web application vulnerabilities (SQL Injection and Cross-site Scripting (XSS)) and security issues, no matter on which platform and technology the website/web application has built-in.



Official Website: https://www.netsparker.com/

Download Link: [Netsparker Download](http://www.netsparker.com/web-vulnerability-scanner/download/)

### #5. NMap (Network Mapper)

A cross-platform web security scanner, written by Gordon Lyon (Fyodor) founder of hosts and services on a computer network.

This scanner delivers correct packets to the target machine and examines the response comes from there. Earlier, Nmap was only for Linux, but it is for Microsoft Windows, BSD variants – Mac OS X, AmigaOS, Solaris, HP-UX, , andSGI IRIX.



Official Website: http://nmap.org/

Download Link: [NMap Download](http://nmap.org/download.html)

### #6. OWASP (Open Web Application Security Project)

A cross-platform online security project/community, involves worldwide organizations, educational organizations and individuals to focus on security on web applications by creating free articles, documentations, techniques, and tools.



Official Website: https://www.owasp.org/index.php/Main\_Page

———————————————————————————–

### Cross Browser Testing Tools:

Cross-browsing issue comes due to comparison of two or more browsers output values. So, to come out of that issue, use cross-browser testing tool,

1. Browsera
2. IE NetRenderer
3. IETester

### #1. Browsera

An online cross-browser testing tool works on Windows and Mac OS X platform, given by Browsera, finds cross-browser layout issues automatically by comparing each browser’s output. After completion of test, each and every Java Script errors are reported and collected. The site crawling feature of the tool is very much helpful in testing all the pages of the site.



Official Website: http://www.browsera.com/

Download link: [Browsera Download](http://www.browsera.com/plans)

### #2. IE NetRenderer

A free HTML tool for Apple iMac and Linux, given by GEOTEK Netzwerkservice Berlin, facilitates users to see how their website displays in IE browsers 5.5 to 11. Additionally, it also displays a large number of capturing jobs in parallel and in real-time. These all services are very fast.



Official Website: http://netrenderer.com/

### #3. IETester v0.5.2

A Windows web browser developed by Core Services, provides a user-friendly interface facilitates users to avail all functionality easily and quickly. A very easy configuration without customization,mainly focus on speed and efficiency. It’s Microsoft ”ribbon” toolbar facilitates users to switch between Internet Explorer versions (5.5 to 9) from the same interface with a single click. Its in-built feature “Favorites” list includes many websites that you can use for compatibility tests.



Official Website: http://www.my-debugbar.com/wiki/IETester/HomePage

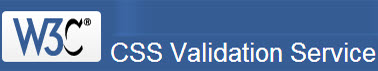
Download Link: [IETester Download](http://www.my-debugbar.com/wiki/IETester/HomePage)

———————————————————————————–

### W3C CSS validator

The W3C CSS Validator is free software given by the W3C, provides services to designers and developers of a web application to check Cascading Style Sheets (CSS).

At present, most of the websites are developed using HTML language, this language is basically used to display text on the page, giving links, and placing multiple objects. To give style (color and layout) to the text of the page, HTML uses CSS (Cascading Style Sheets). The W3C CSS validator used to check CSS and X(HTML) documents with the style sheets.



Official website: http://jigsaw.w3.org/css-validator/

———————————————————————————–

### W3C Link Checker

A link checking service provided by W3C, used to check links and anchors presents on a page of website or whole website using languages; HTML, XHTML or CSS.



Official website: <http://validator.w3.org/checklink>

SELENIUM DAY 2

1.SELENIUM TEST PROCESS LIFE CYCLE:

1. Automation feasibility analysis

2- Test Plan/Test Design

3- Environment Setup/Test lab setup

4-Test Script development/ Automation testcase development

5-Test script execution

6- Generate test result / Analyses of result

### 1- Automation feasibility analysis

In this section you have to think from different perspective. The main objective of this phase will be to check feasibility of automation.

So your main focus will be on below points.

1- Which test case can be automated and how we can automate them

2- Which module of your application can be tested and which can not be automated

3- Which tools we can use for our application (like Selenium,QTP,Sahi,OATS, Telrik etc) and which tools will be best of our application

4- Take following factors into consideration like Team size,Effort and cost involved for tools which we will use.

2- Test Plan/Test Design

This phase plays very important role in Automation test life cycle. In this phase you have to create a Test plan by considering below point into considerations.

1- Fetch all the manual test case from test management tool that which TC has to automate.

2- Which framework to use and what will be advantage and disadvantage of the framework which we will use.

3- Create a test suite for Automation test case in Test Management tool.

4- In test plan you can mention background, limitation, risk and dependency between application and tools.

5- Approval from client/ Stack holders.

### 3- Environment Setup/Test lab setup

By name itself you can understand that we need to setup machine or remote machine where our test case will execute.

1- In this section you can mention how many machine you want.

2- What should be the configuration in terms of hardware and software.

### 4-Test Script development/ Automation test case development

In this phase you have to start develop automation script and make sure all test script is running fine and should be stable enough.

1- Start creating test script based on your requirement

2- Create some common method or function that you can reuse throughout your script

3- Make your script easy, reusable,well structured and well documented so if third person check your script then he/she can understand your scripts easily.

4- Use better reporting so in case of failing you can trace your code

5- Finally review your script and your script should be ready before consumption.

### 5-Test script execution

Now its time for execution of test scripts, in this phas you have to execute all your test script.

Some points to remember while execution

1- Your script should cover all the functional requirement as per testcase.

2- Your script should be stable so it should run in multiple environment and multiple browsers (depends on your requirement)

3- You can do batch execution also if possible so it will save time and effort.

4- In case of failure your script should take screen shots.

5- If test case is failing due to functionality, you have to raise a bug/defect.

### 6- Generate test result / Analyses of result

This is the last phase of Automation test life cycle in which we will gather test result and will share with team/client/stack holders.

1- Analyze the output and calculate how much time it take to complete the testcase.

2- You should have good report generation like XSLT report, TestNG report, ReporterNG etc

WHAT ARE WEBELEMENTS IN SELENIUM AND HOW TO LOCATE.

In order to write UI tests with Selenium WebDriver you need to be able to identify web page elements fast and in an accurate way. You don’t want to revisit these selectors very often so you must choose the right selector from the beginning.

There are some browser tools that you can use in order to identify web elements in the DOM easier. These are:

* Firebug for Firefox
* Google Developer Tools for Chrome
* Web Inspector for Safari

You can also create automated website tests for free with [InspectFocus.com](http://inspectfocus.com/)

Selenium WebDriver API supports different possibilities to identify elements: by ID, by CLASS, by NAME, by CSS selector, by XPath, by TAG name. Also you define your custom selector in order to interact with the elements.

It’s always a good practice to assign unique IDs to elements, also names and classes in order to be more usable for automatic UI tests. If that is not possible you’ll need to use advanced or XPath selector to interact with those elements. The most popular selectors are the CSS selectors due to performance and simplicity reasons.

To inspect an element you just have to open the desired web page, right-click the desired element and click on Inspect Element. A new panel will open showing the desired element. Also you can inspect other elements by clicking on the cursor in the top left side of the Developer Tools or Firebug panels and hovering page elements.

Locating Elements with Selenium WebDriver, findElement() method returns and WebElement and findElements() returns a list of WebElements.

1. By ID:

|  |  |
| --- | --- |
| 1 | in Java: driver.findElement(By.id("element id")) |

2. By CLASS:

|  |  |
| --- | --- |
| 1 | in Java: driver.findElement(By.className("element class")) |

3. By NAME:

|  |  |
| --- | --- |
| 1 | in Java: driver.findElement(By.name("element name")) |

4. By TAGNAME:

|  |  |
| --- | --- |
| 1 | in Java: driver.findElement(By.tagName("element html tag name")) |

5. By CSS Selector:

|  |  |
| --- | --- |
| 1 | in Java: driver.findElement(By.cssSelector("css selector")) |

6. By Link:

|  |  |
| --- | --- |
| 1 | in Java: driver.findElement(By.link("link text")) |

7. By XPath:

|  |  |
| --- | --- |
| 1 | in Java: driver.findElement(By.xpath("xpath expression")) |

a HTML snippet code and see how can we use these Selenium WebDriver selectors in order to identify the desired elements:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | </pre>  <div class="thumbnail center well well-small text-center">  <h2>Newsletter</h2>  Subscribe to our weekly Newsletter and stay tuned.  <form action="" method="post" name="subscribe"><label for="name">Name: </label>  <input class="name" id="name" type="text" placeholder="Enter name..." />  <label for="email">Email: </label> <input class="email" id="email" type="text" placeholder="your@email.com" />  <input class="btn btn-large" type="submit" value="Subscribe" /></form>  <a title="first link" href="#link1">First Link</a>  <a title="second link" href="#link2">Second Link</a></div>  <pre> |

some WebElements from the above HTML code snippet:

|  |  |
| --- | --- |
| 1 | WebElement nameInputField = driver.findElement(By.id("name")); |

or

|  |  |
| --- | --- |
| 1 | WebElement nameInputField = driver.findElement(By.className("name")); |

or

|  |  |
| --- | --- |
| 1 | WebElement emailInputField = driver.findElement(By.id("email")); |

Locate link elements with findElements() method from the HTML snipped

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | @Test  public void findLinksTest(){  //Get all the links displayed  List links = driver.findElements(By.tagName("a"));  assertEquals(2, links.size());  for(WebElement link : links)  System.out.print(link.getAttribute("href"));  } |

Locate link elements from the HTML snipped:

How to find a link with Selenium WebDriver API by full text:

|  |  |
| --- | --- |
| 1  2 | WebElement firstLink = driver.findElement(By.linkText("First Link"));  assertEquals("#link1", firstLink.getAttribute("href")) |

How to find a link with Selenium WebDriver API by partial text:

|  |  |
| --- | --- |
| 1  2 | WebElement firstLink = driver.findElement(By.partialLinkText("First"));  assertEquals("#link1", firstLink.getAttribute("href")) |

Locate elements by HTML Tag Name:

|  |  |
| --- | --- |
| 1 | WebElement subscribeButton = driver.findElement(By.tagName("button")); |

Locate elements by CSS Selectors:

|  |  |
| --- | --- |
| 1  2 | WebElement emailInputField =  driver.findElement(By.cssSelector("form input#email")); |

or

|  |  |
| --- | --- |
| 1  2 | WebElement emailInputField =  driver.findElement(By.cssSelector("input.email")); |

or

|  |  |
| --- | --- |
| 1  2 | WebElement emailInputField =  driver.findElement(By.cssSelector("input[type='submit'][value='Subscribe']")); |

Performing partial match on attribute values

|  |  |
| --- | --- |
| 1 | ^= as in input[id^='email'] means Starting with. |

|  |  |
| --- | --- |
| 1 | $= as in input[id$='\_name'] means Ending with. |

|  |  |
| --- | --- |
| 1 | \*= as in Input[id\*='userName'] means Containing. |

Finding elements with XPATH

* Absolute path

|  |  |
| --- | --- |
| 1  2 | WebElement userName =  driver.findElement(By.xpath("html/body/div/form/input")); |

* Relative path

|  |  |
| --- | --- |
| 1 | WebElement email = driver.findElement(By.xpath("//input")); |

Finding elements using index

|  |  |
| --- | --- |
| 1  2 | WebElement email =  driver.findElement(By.xpath("//input[3]")); |

Finding elements using attributes values with XPath

|  |  |
| --- | --- |
| 1  2 | WebElement logo =  driver.findElement(By.xpath("img[@alt='logo']")); |

XPATH

|  |  |
| --- | --- |
| 1  2  3 | starts-with()    input[starts-with(@id,'input')] |

Starting with:

|  |  |
| --- | --- |
| 1  2  3 | ends-with()    input[ends-with(@id,'\_field')] |

Ending with:

|  |  |
| --- | --- |
| 1  2  3 | contains()    input[contains(@id,'field')] |

Containing

Locating table rows and cells

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 | @Test  public void testTable() {  WebElement simpleTable = driver.findElement(By.id("items"));    //Get all rows  List rows = simpleTable.findElements(By.tagName("tr"));  assertEquals(3, rows.size());    //Print data from each row  for (WebElement row : rows) {  List cols = row.findElements(By.tagName("td"));  for (WebElement col : cols) {  System.out.print(col.getText() + "\t");  }  System.out.print();  }  } |

Using jQuery selectors

Locate all the Checkbox which are checked by calling jQuery find() method.

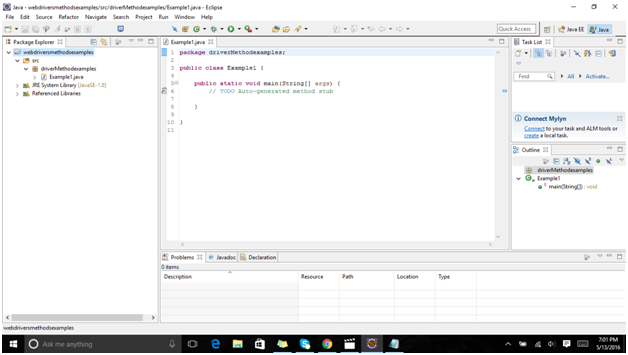
find() method returns elements in array

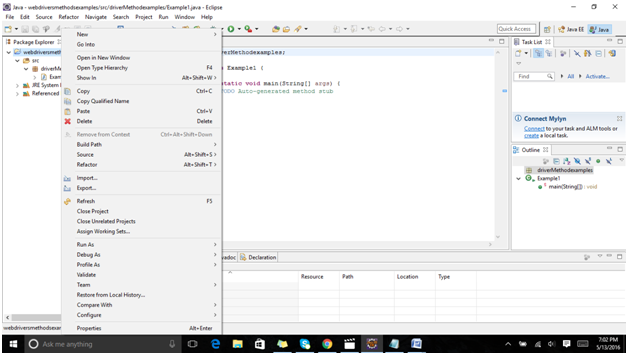
|  |  |
| --- | --- |
| 1  2 | List elements =  (List) js.executeScript("return jQuery.find(':checked')");  How to find web element locator properties  BY knowning id,link,name,class name,link test,css locators,xpath.  TEST CASES FOR AUTOMATIONS.  i) Santity Test : First time u can test project that type of testing is know as sanity test  ii) Regrassion Test : After Modify the project u re testing project that type is know as regrassion test  iii) Data Driven Test : U can test the project using multiple testing data  TEST NG:  TestNG is a testing framework that overcomes the limitations of another popular testing framework called JUnit. The "NG" means "Next Generation". Most Selenium users use this more than JUnit because of its advantages. There are so many features of TestNG, but we will only focus on the most important ones that we can use in Selenium.  Advantages of testng over junit & installing testng go through this link: <http://www.guru99.com/all-about-testng-and-selenium.html>. |

How To Add Selenium jar files to your project

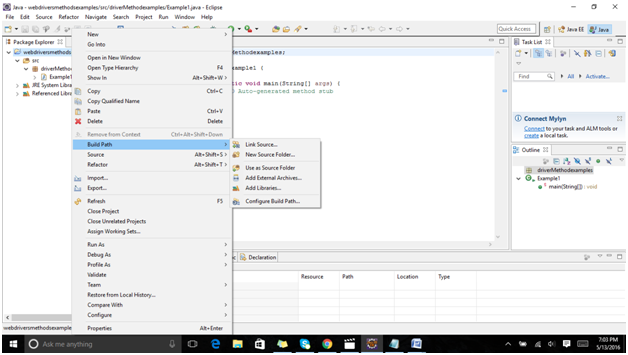
--------------------------------------

1) Right click on your project

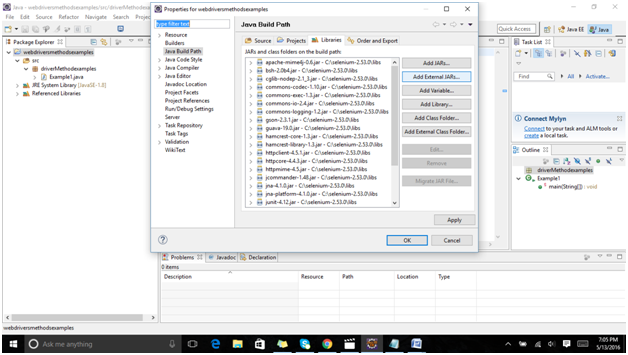




2) Select build path - configure build path -



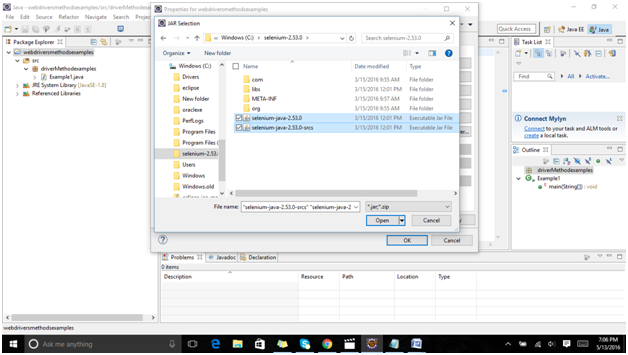
3) select libraries - select add external jar - select your selenium java folder jar files



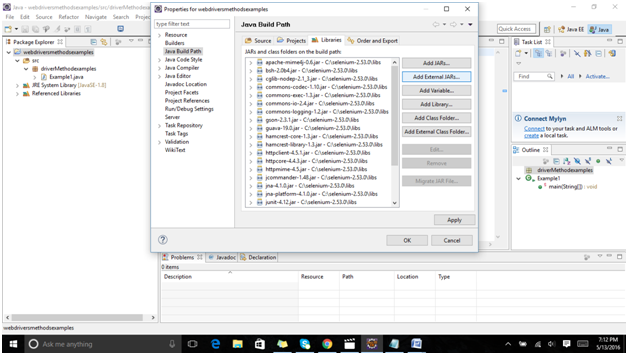
Select add external files.

Then you will get a window like this:

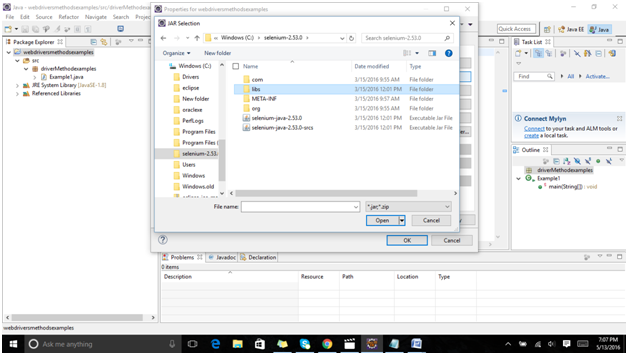
You need to select from the selenium jar folder where u have kept in the location...I kept in windows c drive so i selected that path here is a screen shot of it:



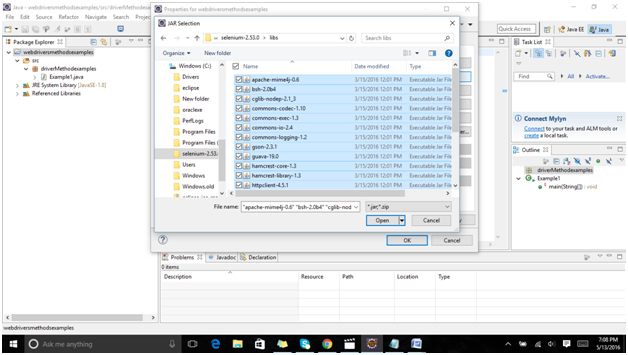
Then add library files :



Then go to the window c drive where te selenium files are located and then select lib file like the screen shot below.



Then select all the folders in lib folder



Then click open and you are done.

WEB DRIVER METHODS:

a) Webdriver Object :

This used to open or test your project based on which browser

Syntax : WebDriver d=new FirefoxDriver();

b) Get() :

This is used to open the your project url

Syntax : d.get("<http://www.google.co.in>");

c) GetTitle() :This is used to get the open project title

Syntax : String t = d.getTitle();

System.out.println("Title : "+t);

d) Get Current Url () :

This is used get the current url on ur browser

Syntax :

String url = d.getCurrentUrl();

System.out.println("Current Url : "+url);

e) Navigate.To():

This is used to open project url next tab on browser

Syntax : d.navigate().to("http://www.facebook.com");

f) Navigate.Back() :

This is used open back previous url on browser

Syntax : d.navigate().back();

g) Navigate.Forward() :

This is used to open forward previous url on browser

syntax : d.navigate().forward();

h) FindElement() :

This is used to find the elements location on webpage using their properites , that properites we are identification using firebug or firepath or seleniumide

Syntax : d.findElement(By.id("Email")).sendKeys("maheshprojectdeveloper@gmail.com");

g) Sendkeys();

This is used to send values to ur webelements like textbox , ..etc

Syntax : d.findElement(By.id("Email")).sendKeys("[maheshprojectdeveloper@gmail.com](mailto:maheshprojectdeveloper@gmail.com)");

h) Click() :

This is used to apply the click action on ur webelements

Syntax :

d.findElement(By.xpath(".//\*[@id='next']")).click();

GENRATING BASIC SELENIUM SCRIPT:

Example :

=========

package driverMethodsExamples;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver

public class Example1 {

public static void main(String[] args)

{

// create the browser object

WebDriver d=new FirefoxDriver();

// open url

d.get("<http://www.google.co.in>");

// get the title

String t = d.getTitle();

System.out.println("Title : "+t);

// get the current url

String url = d.getCurrentUrl();

System.out.println("Current Url : "+url);

// navigate to

d.navigate().to("http://www.facebook.com");

// back to url

d.navigate().back()

// forward url

d.navigate().forward();

// open gmail

d.get("http://www.gmail.com");

d.findElement(By.id("Email")).sendKeys("[maheshprojectdeveloper@gmail.com](mailto:maheshprojectdeveloper@gmail.com)");

d.findElement(By.xpath(".//\*[@id='next']")).click();

}

}

4

1.HANDLING IMAGES,DROPDOWNLIST,RADIO BUTTONS

IMAGES:

Handling images in selenium webdriver basically holds three types whic are as follows:

1.Genral Images.

2.Image Hyper Links.

3.Image buttons (Radio buttons)

Genral images: how to maintain in html tag : ?? <img></img>

Images have to be in link.

Hyperlinks also have images.

Input tag also contains images.i.e they contain images

1.sample code for genral images:

package imagesHandling;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

public class GeneralImage {

public static void main(String[] args)

{

// Create the fire fox driver

WebDriver d=new FirefoxDriver();

//open google

d.get("https://www.google.co.in/?gws\_rd=ssl");

// find general image

String logoname=d.findElement(By.xpath(".//\*[@id='hplogo']")).getAttribute("title");

System.out.println("Image title : "+logoname);

}

}

2. Genral Script for image hyperlinks :

package imagesHandling;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

public class HyperLinkImgae {

public static void main(String[] args)

{

// create the webdriver

WebDriver d=new FirefoxDriver();

// open site

d.get("http://www.seleniumhq.org/");

// handling hyperlink images

d.findElement(By.xpath(".//\*[@id='header']/h1/a")).click();

}

}

3.Handling Image buttons:

package imagesHandling;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

public class InputImages {

public static void main(String[] args)

{

// create the webdriver

WebDriver d=new FirefoxDriver();

// open site

d.get("<http://www.newtours.demoaut.com/>")

// handling input images

d.findElement(By.name("login")).click();

}

}

4.Handling Radio Buttons :

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

public class RadioButtonClass {

public static void main(String[] args) throws InterruptedException

{

// Create the WebDrivedr

WebDriver d=new FirefoxDriver();

// open website

d.get("http://www.newtours.demoaut.com/");

// send username

Thread.sleep(3000);

d.findElement(By.name("userName")).sendKeys("mahesh");

// send password

Thread.sleep(3000);

d.findElement(By.name("password")).sendKeys("mahesh");

// click on signin button

Thread.sleep(3000);

d.findElement(By.name("login")).click();

// select one way radio button

Thread.sleep(3000);

d.findElement(By.xpath("html/body/div[1]/table/tbody/tr/td[2]/table/tbody/tr[4]/td/table/tbody/tr/td[2]/table/tbody/tr[5]/td/form/table/tbody/tr[2]/td[2]/b/font/input[2]")).click();

boolean b1=d.findElement(By.xpath("html/body/div[1]/table/tbody/tr/td[2]/table/tbody/tr[4]/td/table/tbody/tr/td[2]/table/tbody/tr[5]/td/form/table/tbody/tr[2]/td[2]/b/font/input[2]")).isSelected();

if(b1==true)

{

System.out.println("One way is selected");

}

// select roundtip way radio button

Thread.sleep(3000);

d.findElement(By.xpath("html/body/div[1]/table/tbody/tr/td[2]/table/tbody/tr[4]/td/table/tbody/tr/td[2]/table/tbody/tr[5]/td/form/table/tbody/tr[2]/td[2]/b/font/input[1]")).click();

boolean b2=d.findElement(By.xpath("html/body/div[1]/table/tbody/tr/td[2]/table/tbody/tr[4]/td/table/tbody/tr/td[2]/table/tbody/tr[5]/td/form/table/tbody/tr[2]/td[2]/b/font/input[1]")).isSelected();

if(b2==true)

{

System.out.println("round tip is selected");

}

}

}

# Usage of Selenium Select Class for Handling Dropdown Elements on a Web Page :

Moving ahead with the few upcoming tutorials in the Selenium series, we would concentrate on handling the various types of web elements available on the web pages. Therefore, in this tutorial, we would consider “dropdowns” and exercise their handling strategies.

Before moving towards problem statement and its resolution, let us take a moment to introduce and create an understanding regarding the application under test. As a sample, we have created a dummy HTML page consisting of multiple and assorted web elements.

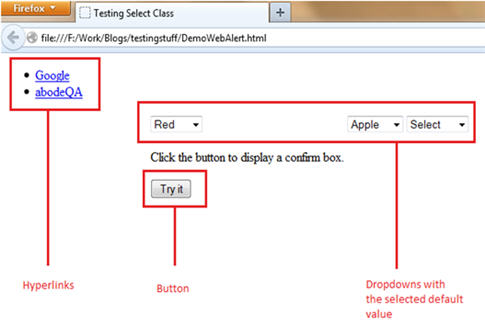
The elementary web elements those constitute the web page are:

Dropdown.

Hyperlink.

Button.

Please take a reference of the following webpage aforementioned above:



### EXAMPLE:

### Explanation of Application under Test

We have designed the web page in a way to include a few fundamental types of web elements.

* Hyperlink: The two hyperlinks namely “Google” and “abodeQA” have been provided that re-directs the user to “https://www.google.co.in/” and “http://www.abodeqa.com/” respectively on the click event.
* Dropdown: The three dropdowns have been created for selecting colors, fruits and animals with a value already set to default.
* Button: A “try it” button has been created to show up the pop up box having Ok and Cancel button upon click event.

How to handling dropdownlist

package dropDownlist;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

import org.openqa.selenium.support.ui.Select;

public class SampleExample {

public static void main(String[] args) throws InterruptedException

{

// create the fire fox driver

WebDriver d=new FirefoxDriver();

// open url

d.get("<http://newtours.demoaut.com/mercurywelcome.php>");

// send username

Thread.sleep(5000);

d.findElement(By.name("userName")).sendKeys("mahesh");

// send password

Thread.sleep(5000);

d.findElement(By.name("password")).sendKeys("mahesh");

// click on sing in

Thread.sleep(5000);

d.findElement(By.name("login")).click()

// dropdownlist

Thread.sleep(5000);

Select drp=new Select(d.findElement(By.name("fromPort")));

Thread.sleep(5000);

drp.selectByValue("London");

Thread.sleep(5000);

drp.selectByValue("Paris")

Thread.sleep(5000);

drp.selectByValue("San Francisco");

Thread.sleep(5000);

drp.selectByValue("Sydney");

Thread.sleep(5000);

d.findElement(By.xpath("html/body/div[1]/table/tbody/tr/td[2]/table/tbody/tr[2]/td/table/tbody/tr/td[1]/a")).click();

}

}

TESTNG FRAMEWORK:

TestNG is an advance framework designed in a way to leverage the benefits by both the developers and testers. For people already using JUnit, TestNG would seem no different with some advance features. With the commencement of the frameworks, JUnit gained an enormous popularity across the Java applications, Java developers and Java testers, with remarkably increasing the code quality.

Despite being an easy to use and straightforward framework, JUnit has its own limitations which give rise to the need of bringing [TestNG](http://testng.org/doc/index.html)into the picture. TestNG was created by an acclaimed programmer named as “Cedric Beust”. TestNG is an open source framework which is distributed under the Apache software License and is readily available for download.

Talking about our requirement to introduce TestNG with WebDriver is that it provides an efficient and effective test result format that can in turn be shared with the stake holders to have a glimpse on the product’s/application’s health thereby eliminating the drawback of WebDriver’s incapability to generate test reports. TestNG has an inbuilt exception handling mechanism which lets the program to run without terminating unexpectedly.

Both TestNG and JUnit belong to the same family of Unit Frameworks where TestNG is an extended version to JUnit and is more extensively used in the current testing era.

### Features of TestNG

* Support for annotations
* Support for parameterization
* Advance execution methodology that do not require test suites to be created
* Support for Data Driven Testing using Dataproviders
* Enables user to set execution priorities for the test methods
* Supports threat safe environment when executing multiple threads
* Readily supports integration with various tools and plug-ins like build tools (Ant, Maven etc.), Integrated Development Environment (Eclipse).
* Facilitates user with effective means of Report Generation using ReportNG

TestNG versus JUnit

There are various advantages that make TestNG superior to JUnit. Some of them are:

* Advance and easy annotations
* Execution patterns can be set
* Concurrent execution of test scripts
* Test case dependencies can be set

Annotations are preceded by a “@” symbol in both TestNG and JUnit.

So now let us get started with the installation and implementation part.

### TestNG Installation in Eclipse

Follow the below steps to TestNG Download and installation on eclipse:

### Creation of Sample TestNG project

### Creating TestNG class

### estNG Annotations:

Following is the list of the most useful and favorable annotations in TestNG:

|  |  |
| --- | --- |
| Annotation | Description |
| @Test | The annotation notifies the system that the method annotated as @Test is a test method |
| @BeforeSuite | The annotation notifies the system that the method annotated as @BeforeSuite must be executed before executing the tests in the entire suite |
| @AfterSuite | The annotation notifies the system that the method annotated as @AfterSuite must be executed after executing the tests in the entire suite |
| @BeforeTest | The annotation notifies the system that the method annotated as @BeforeTest must be executed before executing any test method within the same test class |
| @AfterTest | The annotation notifies the system that the method annotated as @AfterTest must be executed after executing any test method within the same test class |
| @BeforeClass | The annotation notifies the system that the method annotated as @BeforeClass must be executed before executing the first test method within the same test class |
| @AfterClass | The annotation notifies the system that the method annotated as @AfterClass must be executed after executing the last test method within the same test class |
| @BeforeMethod | The annotation notifies the system that the method annotated as @BeforeMethod must be executed before executing any and every test method within the same test class |
| @AfterMethod | The annotation notifies the system that the method annotated as @AfterMethod must be executed after executing any and every test method within the same test class |
| @BeforeGroups | The annotation notifies the system that the method annotated as @BeforeGroups is a configuration method that enlists a group and that must be executed before executing the first test method of the group |
| @AfterGroups | The annotation notifies the system that the method annotated as @AfterGroups is a configuration method that enlists a group and that must be executed after executing the last test method of the group |

*Note*: Many of the aforementioned annotations can be exercised in JUnit 3 and JUnit 4 framework also.

### Conclusion

Through this tutorial, we tried to make you acquainted with a java based testing framework named as TestNG. We started off the session with the installation of the framework and moved with the script creation and advance topics. We discussed all the annotations provided by TestNG. We implemented and executed our first TestNG test script using annotations and assert statements.

Article summary:

* TestNG is an advance framework designed in a way to leverage the benefits by both the developers and testers.
* TestNG is an open source framework which is distributed under the Apache software License and is readily available for download.
* TestNG is considered to be superior to JUnit because of its advance features.
* Features of TestNG
  + Support for Annotations
  + Advance execution methodology that do not require test suites to be created
  + Support for parameterization
  + Support for Data Driven Testing using Dataproviders
  + Setting execution priorities for the test methods
  + Supports threat safe environment when executing multiple threads
  + Readily supports integration with various tools and plug-ins like build tools (Ant, Maven etc.), Integrated Development Environment (Eclipse).
  + Facilitates user with effective means of Report Generation using ReportNG
* Advantages of TestNG over JUnit
  + Added advance and easy annotations
  + Execution patterns can be set
  + Concurrent execution of test scripts
  + Test case dependencies can be set
* TestNG is freely available and can be easily installed in the Eclipse IDE using Eclipse Market.
* Upon installation, TestNG would be available as a library within the Eclipse environment.
* Create a new Java Project and configure the build path using TestNG library.
* Create a new TestNG class by expanding the created TestNG project and traverse to its “src” folder. Right click on the “src” package and navigate to New -> Other. Select TestNG class option.
* @Test is one of the annotations provided by TestNG. This annotation lets the program execution to know that method annotated as @Test is a test method. To be able to use different TestNG annotations, we need to import the package “import org.testng.annotations.\*”.
* There is no need of main() method while creating test scripts using TestNG.
* We use Assert class while comparing expected and the actual value. Assert class is used to perform various verifications. To be able to use different assertions, we are required to import “import org.testng.Assert”.
* If a test script is composed of more than one test methods, the execution priority and sequence can be set using TestNG annotation “@Test” and by setting a value for the “priority” parameter.
* TestNG has a capability of generating human readable test execution reports automatically. These reports can be viewed in any of the browser and it can also be viewed using Eclipse’s built – in browser support.

TestNg Automation Framework

===========================

i) Using TestNg we are create the testcases of the project

In our selenium there are two types of automation frameworks

1) Junit

2) TestNg( Next Generation)

Note : Compare between junit and testng , the testng is very power full , It open source automation framework

Benfits

1) It Genrate the html Test Reports

2) Easy to create the test cases

3) Easy to running number testcases can be grouped very easily

4) Parallel testing is possiable

5) Parameterization is also possiable

There is No Main Method we are implementing in testng so with out main method we are runing script

How to add TestNg Features to Eclipse

=====================================

1) Goto Help - Select Install New Software

2) Click on Add

3) Enter Name Field : TestNg

4) Enter Location : http://www.beust.com/eclipse/

5) click on ok

6) Select all TestNg Points

7) Click on Next

8) Click on Nest

9) Tick mark on I accept Message

How to Write the TestNg Programs

In TestNg script containg the number of functions

@Test : TestNg Automation

@Test(priority=1) : Test Method is execute using priority number

@Test(enabled=false) : To stop the execute of the function or testcases

Example

-------

package samplePackage;

import org.testng.annotations.Test;

public class SampleTestNgClass

{

@Test(priority=1)

public void Login()

{

System.out.println("Login Sucessfull..");

}

@Test(priority=2

public void BillPayMent()

{

System.out.println("Bill PayMent Sucessfull...");

}

@Test(enabled=false)

public void Trasaction()

{

System.out.println("Money Transfer Sucessfull...");

}

@Test(priority=4)

public void Logout()

{

System.out.println("Logout Sucessfull...");

}

}

@BeforeMethod - > Execute the testcase every testcase before execute this testcase

@AfterMethod -> Exete the test case every testcase after execute this tesecase

package samplePackage;

import org.testng.annotations.AfterMethod;

import org.testng.annotations.BeforeMethod;

import org.testng.annotations.Test;

public class SampleTestNgClass

{

@BeforeMethod

public void Login()

{

System.out.println("Login Sucessfull..");

}

@Test(priority=1)

public void BillPayMent()

{

System.out.println("Bill PayMent Sucessfull...");

}

@Test(priority=2)

public void Trasaction()

{

System.out.println("Money Transfer Sucessfull...");

}

@AfterMethod

public void Logout()

{

System.out.println("Logout Sucessfull...");

}

}

Open Urls Using TestNG

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

package samplePackage;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

import org.testng.annotations.BeforeMethod;

import org.testng.annotations.Test;

public class OpenUrlsUsingTestNg

{

WebDriver d=new FirefoxDriver();

@BeforeMethod

public void DelayExecution() throws InterruptedException

{

Thread.sleep(5000);

System.out.println("Please Wait...");

}

@Test(priority=1)

public void OpenGoogle()

{

d.get("http://www.google.co.in");

System.out.println("Google Site Open Sucessfully...");

}

@Test(priority=2)

public void OpenGmail()

{

d.get("http://www.gmail.com");

System.out.println("Gmail Site Open Sucessfully...");

}

@Test(priority=3)

public void OpenFacebook()

{

d.get("<http://www.facebook.com>");

System.out.println("Facebook Site Open Sucessfully...");

}

}

Frameworks in selenium:

What is Frame work in selenium ?

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.

In a very simple language, we can say that a framework is a constructive blend of various guidelines, coding standards, concepts, processes, practices, project hierarchies, modularity, reporting mechanism, test data injections etc. to pillar automation testing. Thus, user can follow these guidelines while automating application to take advantages of various productive results.

The advantages can be in different forms like ease of scripting, scalability, modularity, understandability, process definition, re-usability, cost, maintenance etc. Thus, to be able to grab these benefits, developers are advised to use one or more of the Test Automation Framework.

Moreover, the need of a single and standard Test Automation Framework arises when you have a bunch of developers working on the different modules of the same application and when we want to avoid situations where each of the developer implements his/her approach towards automation.

*Note*: Take a note that a testing framework is always application independent that is it can be used with any application irrespective of the complications (like Technology stack, architecture etc.) of application under test. The framework should be scalable and maintainable.

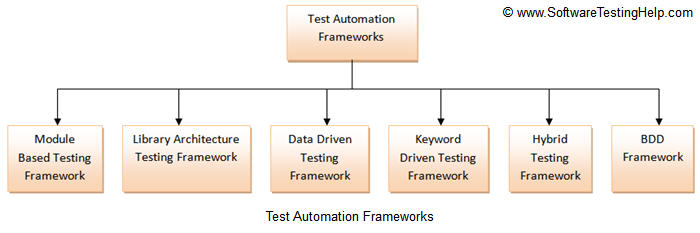
Advantage of Test Automation framework

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

Types of Automation framework:

There is a divergent range of Automation Frameworks available now days. These frameworks may differ from each other based on their support to different key factors to do automation like reusability, ease of maintenance etc.

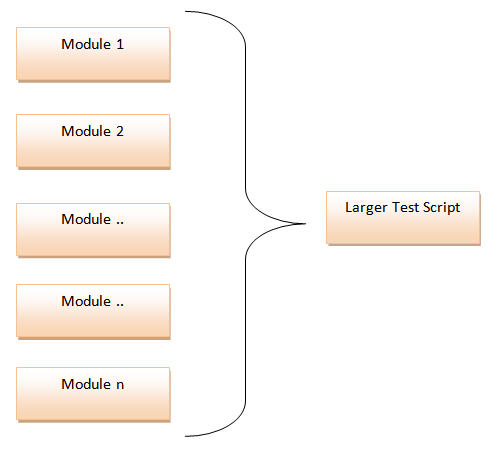
1. Module Based Testing Framework.
2. Library Architecture Testing Framework.
3. Data Driven Testing Framework.
4. Keyword Driven Testing Framework.
5. Hybrid Testing Framework.
6. Behavior Driven Development Framework.



### #1) Module Based Testing Framework

Module based Testing Framework is based on one of the popularly known OOPs concept – Abstraction. The framework divides the entire “Application Under Test” into number of logical and isolated modules. For each module, we create a separate and independent test script. Thus, when these test scripts taken together builds a larger test script representing more than one modules.

These modules are separated by an abstraction layer in such a way that the changes made in the sections of the application doesn’t yields affects on this module.



Advantages:

1. The framework introduces high level of modularization which leads to easier and cost efficient maintenance.
2. The framework is pretty much scalable
3. If the changes are implemented in one part of the application, only the test script representing that part of the application needs to be fixed leaving all the other parts untouched.

Disadvantages:

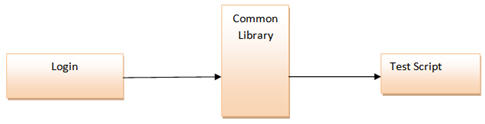
While implementing test scripts for each module separately, we embed the test data (Data with which we are supposed to perform testing) into the test scripts. Thus, whenever we are supposed to test with a different set of test data, it requires the manipulations to be made in the test scripts.

### 2) Library Architecture Testing Framework

The Library Architecture Testing Framework is fundamentally and foundationally built on Module Based Testing Framework with some additional advantages. Instead of dividing the application under test into test scripts, we segregate the application into functions or rather common functions can be used by the other parts of the application as well. Thus we create a common library constituting of common functions for the application under test. Therefore, these libraries can be called within the test scripts whenever required.

The basic fundamental behind the framework is to determine the common steps and group them into functions under a library and call those functions in the test scripts whenever required.

Example: The login steps can be combined into a function and kept into a library. Thus all the test scripts those require to login the application can call that function instead of writing the code all over again.



Advantages:

1. Like Module Based Framework, this framework also introduces high level of modularization which leads to easier and cost efficient maintenance and scalability too.
2. As we create common functions that can be efficiently used by the various test scripts across the Framework. Thus, the framework introduces a great degree of re-usability.

Disadvantages:

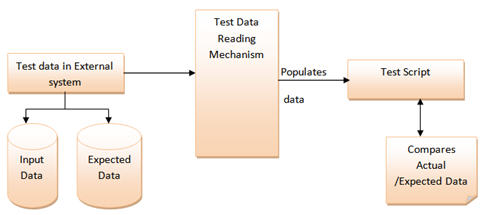
Like Module Based Framework, the test data is lodged into the test scripts, thus any change in the test data would require changes in the test script as well.

1. With the introduction of libraries, the framework becomes a little complicated.
2. With the introduction of libraries, the framework becomes a little complicated.

### 3) 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.



**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.

public void readTD(String TestData, String testcase) throws Exception {

TestData=readConfigData(configFileName,"TestData",driver);

testcase=readConfigData(configFileName,"testcase",driver);

FileInputStream td\_filepath = new FileInputStream(TestData);

Workbook td\_work==Workbook.getWorkbook(td\_filepath);

Sheet td\_sheet = td\_work.getSheet(0);

if(counter==0)

{

for (int i = 1,j = 1; i <= td\_sheet.getRows()-1; i++)

{

if(td\_sheet.getCell(0,i).getContents().equalsIgnoreCase(testcase))

{

startrow = i;

arrayList.add(td\_sheet.getCell(j,i).getContents());

testdata\_value.add(td\_sheet.getCell(j+1,i).getContents());

for (int j = 0, k = startrow +1; k <= td\_sheet.getRows()-1; k++)

{

if (td\_sheet.getCell(j,k).getContents()=="")

{

arrayList.add(td\_sheet.getCell(j+1,k).getContents());

testdata\_value.add(td\_sheet.getCell(j+2,k).getContents());

}

}

}

Counter++;

}

**Advantages:**

1. The most important feature of this framework is that it considerably reduces the total number of scripts required to cover all the possible combinations of test scenarios. Thus lesser amount of code is required to test a complete set of scenarios.
2. Any change in the test data matrix would not hamper the test script code.
3. Increases flexibility and maintainability
4. A single test scenario can be executed altering the test data values.

**Disadvantages:**

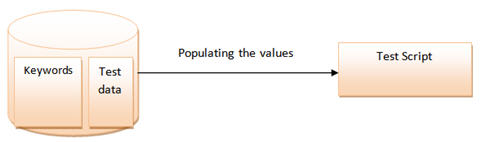
1. The process is complex and requires an extra effort to come up with the test data sources and reading mechanisms.
2. Requires proficiency in a programming language that is being used to develop test scripts.

### **4) 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**



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.

**Advantages:**

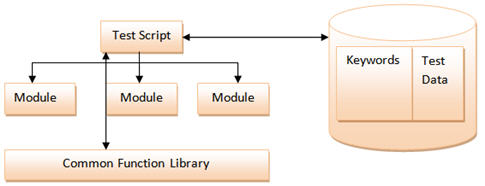
1. In addition to advantages provided by Data Driven testing, Keyword driven framework doesn’t require the user to possess scripting knowledge unlike Data Driven Testing.
2. A single keyword can be used across multiple test scripts.

**Disadvantages:**

1. The user should be well versed with the Keyword creation mechanism to be able to efficiently leverage the benefits provided by the framework.
2. The framework becomes complicated gradually as it grows and a number of new keywords are introduced.

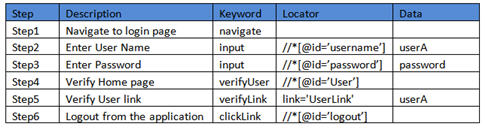
### **5) Hybrid Testing Framework**

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.



**Example of Hybrid Framework**

Test sheet would contain both the keywords and the Data.



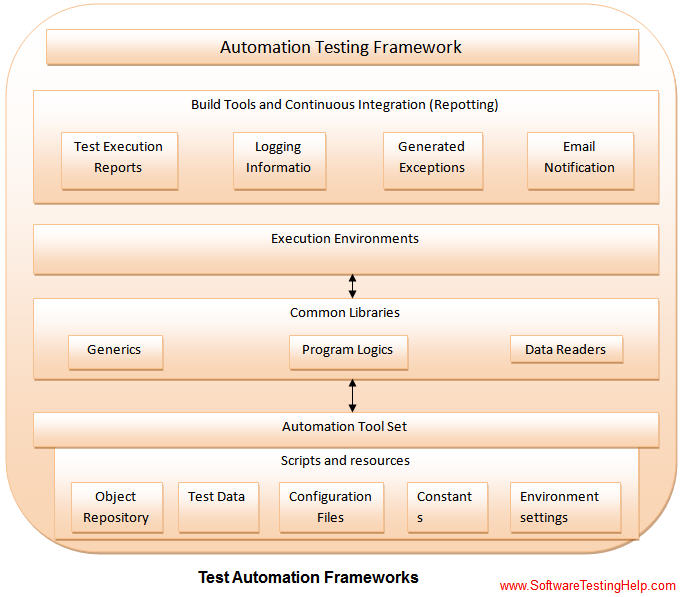
In the above example, keyword column contains all the required keywords used in the particular test case and data column drives all the data required in the test scenario. If any step does not need any input then it can be left empty.

### **6) Behavior Driven Development Framework**

Behavior Driven Development framework allows automation of functional validations in easily readable and understandable format to Business Analysts, Developers, Testers, etc. Such frameworks do not necessarily require the user to be acquainted with programming language. There are different tools available for BDD like cucumber, Jbehave etc. Details of BDD framework are discussed later in Cucumber tutorial. We have also discussed details on Gherkin language to write test cases in Cucumber.

**Components of Automation Testing Framework**

***(click on image to view enlarged)***



Though the above pictorial representation of a framework is self-explanatory but we would still highlight a few points.

1. **Object Repository**: Object Repository acronym as OR is constituted of the set of locators types associated with web elements.
2. **Test Data:** The input data with which the scenario would be tested and it can be the expected values with which the actual results would be compared.
3. **Configuration File/Constants/ Environment Settings**: The file stores the information regarding the application URL, browser specific information etc. It is generally the information that remains static throughout the framework.
4. **Generics/ Program logics/ Readers**: These are the classes that store the functions which can be commonly used across the entire framework.
5. **Build tools and Continuous IIntegration**: These are the tools that aids to the frameworks capabilities to generate test reports, email notifications and logging information.

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |