Pre-Requisites for using the selenium:

- Java should be installed and configured.
- Download the Eclipse.

Steps to install and configure Java.

Step 1: Navigate to following website

http://www.oracle.com/technetwork/java/javase/downloads/index.html

Step 2: Double click on the downloaded exe file. Follow the instruction by clicking on the next button.

Step 3: After the installation of Java, we need to update the PATH Environment Variable

Step 4: Copy the path "C:\Program Files\Java\jdk1.8.0_151\bin"

- Right click on my computer
- Click on properties
- Click on advanced system settings
- Click on environment variable button on system properties button
- Click on new button in system variables
- Paste the path copied and click on the ok button.

Step 5: How to check java is installed or not:-

Open command prompt

- Type **java –version** and click on enter
- It will show the current installed version of java

Why we need to set the path in java?

- We need to set the path to tell the OS where our java commands are present and to recognize.

The path is required to be set for using tools such as javac, java etc. If you are saving the java source file inside the jdk/bin directory, path is not required to be set because all the tools will be available in the current directory. But if you are having your java file outside the jdk/bin folder, it is necessary to set path of JDK.

Step 6: After installing Java. Install the eclipse.

Step 7: Create workspace in any of the directory.

Example: D: //Automation

Step 8: Launch the eclipse and navigate to the location which is created in the above location.

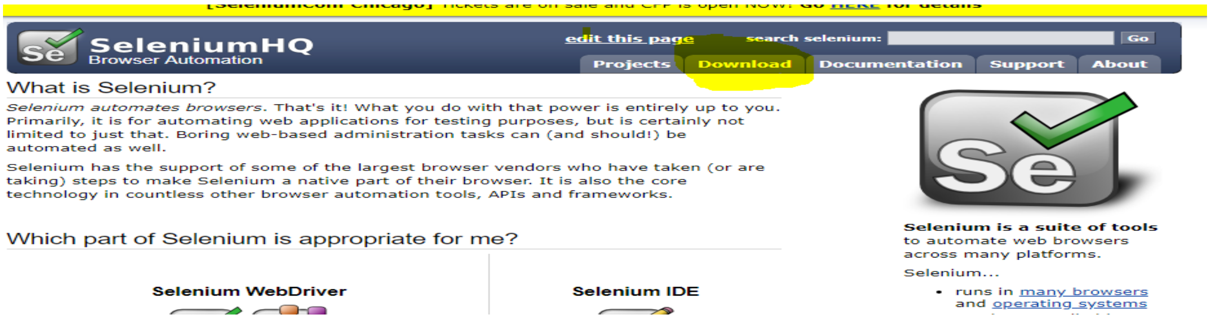
Step 8: Create the project in eclipse. Example: Selenium

Download the selenium:

Step 1: Navigate to the following website

https://docs.seleniumhq.org/

Step 2: Click on the download tab



What is Selenium?

Selenium automates browsers. That's it! What you do with that power is entirely up to you. Primarily, it is for automating web applications for testing purposes, but is certainly not limited to just that. Boring web-based administration tasks can (and should!) be automated as well.

Selenium has the support of some of the largest browser vendors who have taken (or are taking) steps to make Selenium a native part of their browser. It is also the core technology in countless other browser automation tools, APIs and frameworks.

Which part of Selenium is appropriate for me?

Selenium WebDriver

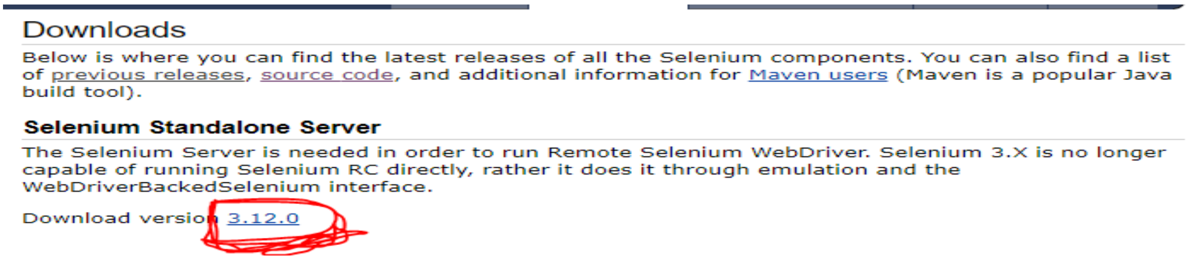
Selenium IDE

Selenium is a suite of tools to automate web browsers across many platforms.

Selenium...

- runs in many browsers and operating systems

Step 3: Click on the link called '3.12.0'



Downloads

Below is where you can find the latest releases of all the Selenium components. You can also find a list of previous releases, source code, and additional information for Maven users (Maven is a popular Java build tool).

Selenium Standalone Server

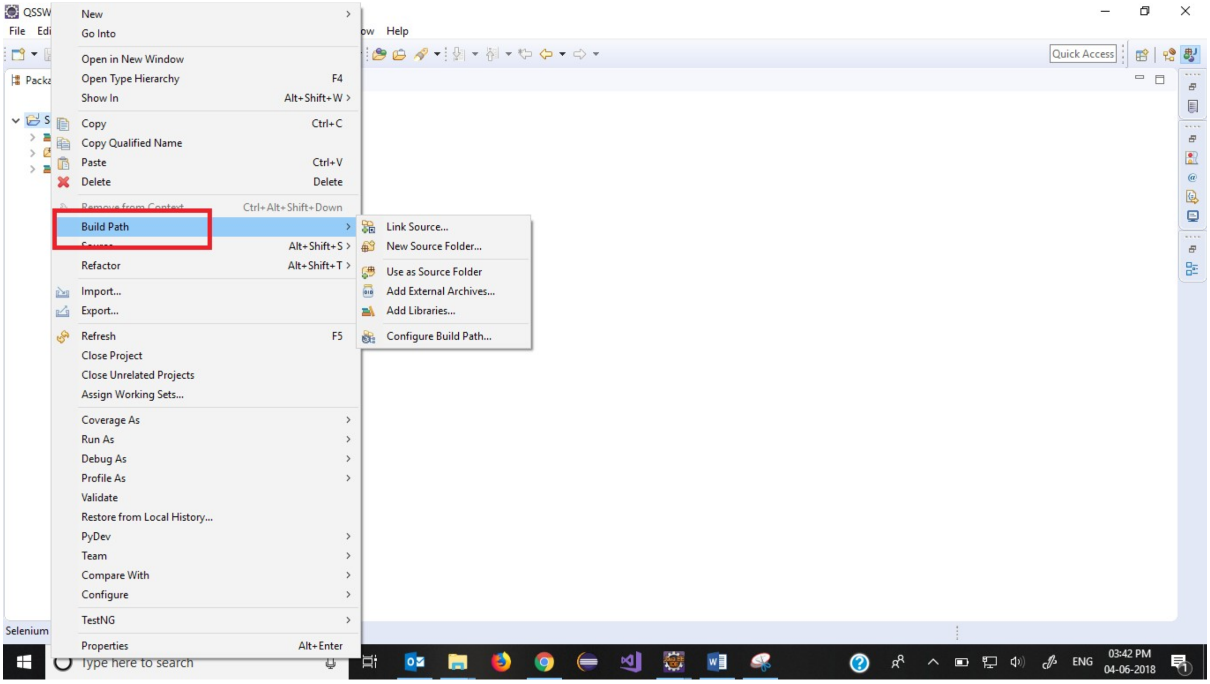
The Selenium Server is needed in order to run Remote Selenium WebDriver. Selenium 3.X is no longer capable of running Selenium RC directly, rather it does it through emulation and the WebDriverBackedSelenium interface.

Download version 3.12.0

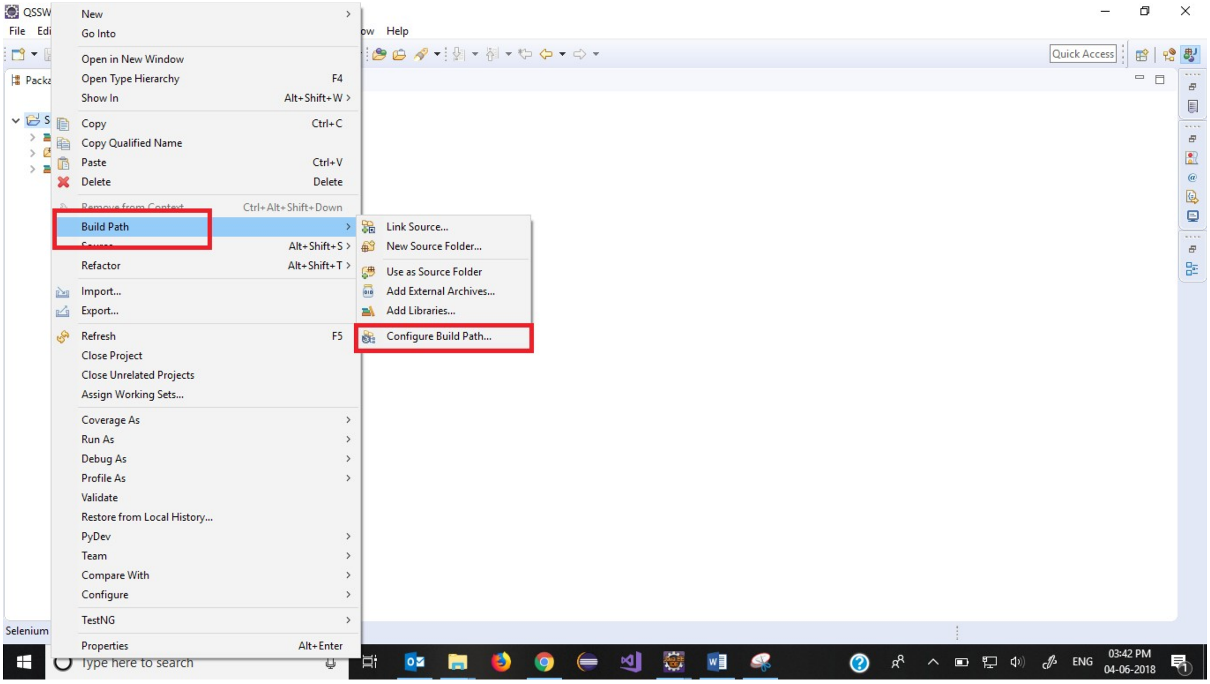
Step 4: "selenium-server-standalone-3.12.0.jar" file will be downloaded.

Step 5: Open the eclipse

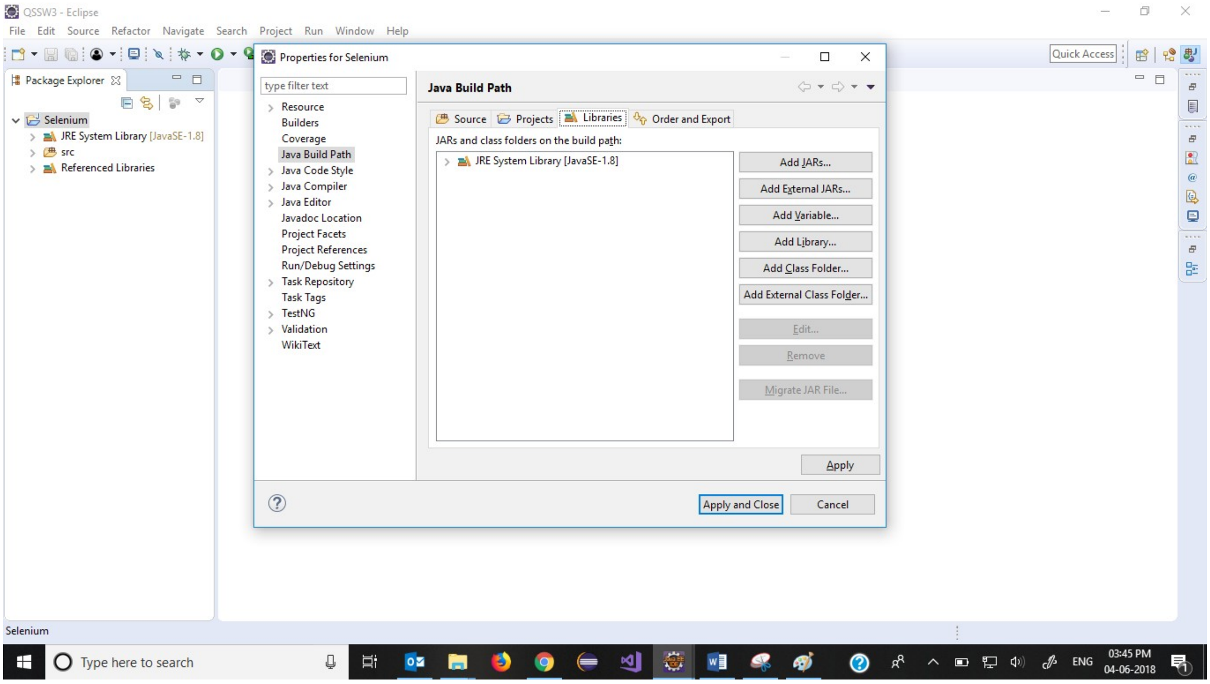
Step 6: Right click on the project which is created and move the cursor to build path.



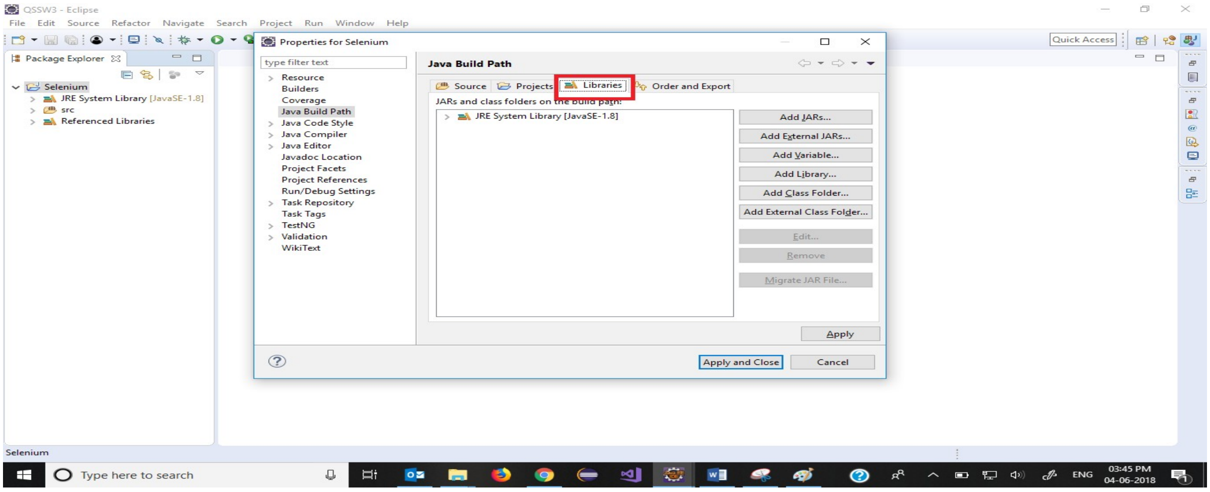
Step 7: Select configure build path



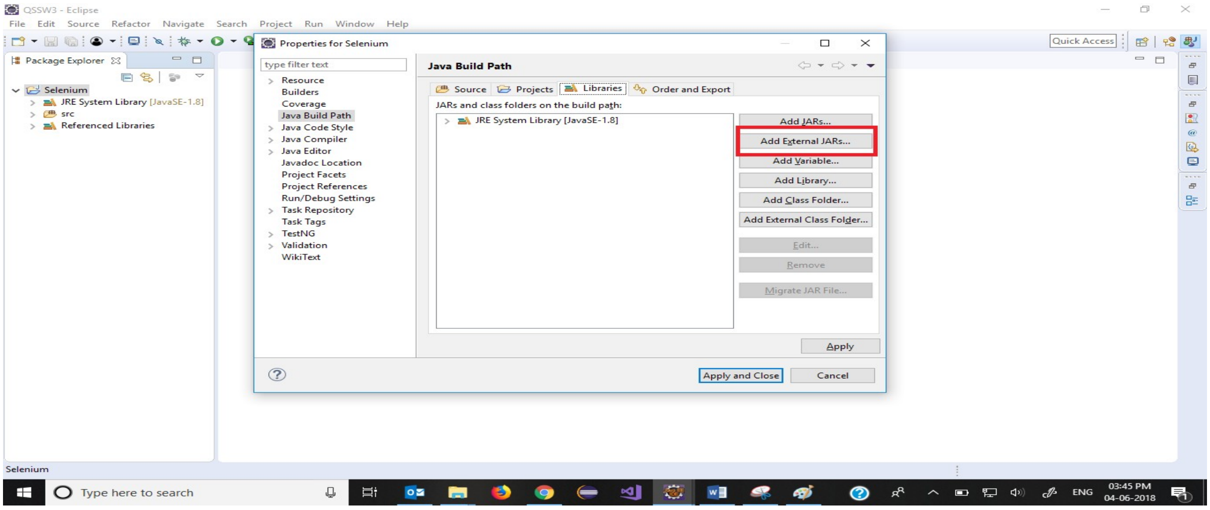
Step 8: Java build path window should display



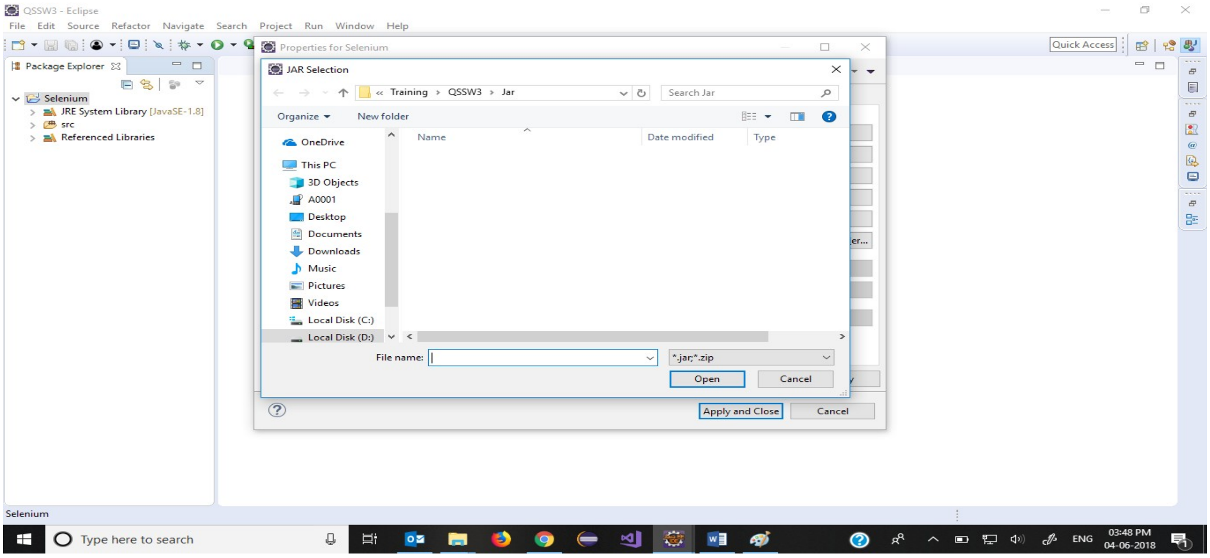
Step 9: Click on the Libraries tab



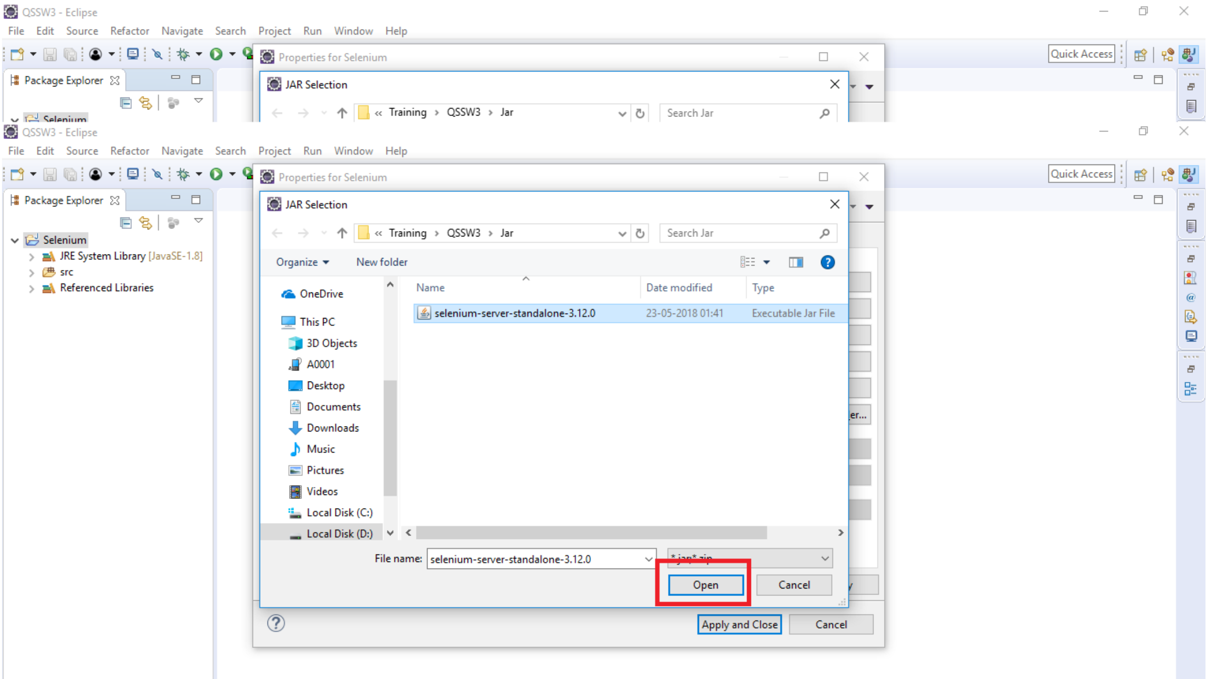
Step 10: Click on the “Add External JARs” button



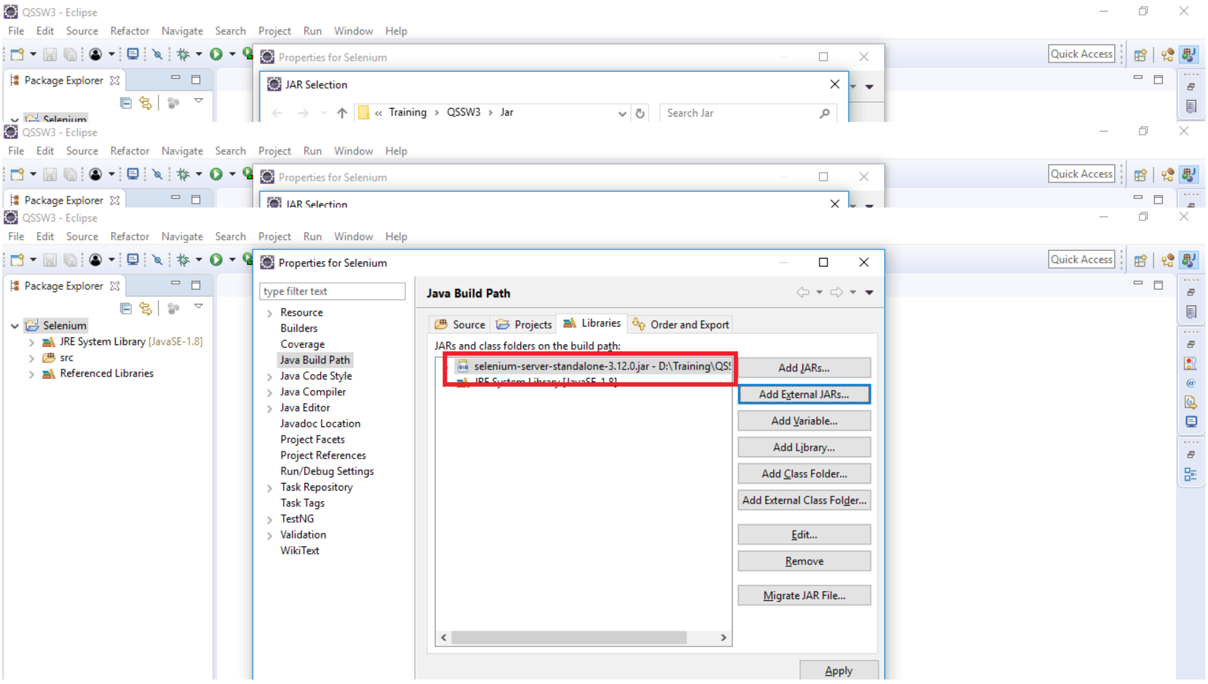
Step 11: Jar selection window should display



Step 12: Navigate the folder where you have kept the jar file. Select the jar file and click on Open button



Step 13: Make sure that the jar file is loaded into the libraries.



Step 14: If it is present, then click on apply and click on apply and close button

Note: Selenium integration is completed.

Develop the code for launching the browser.

- ➔ **Program to launch the chrome browser.**

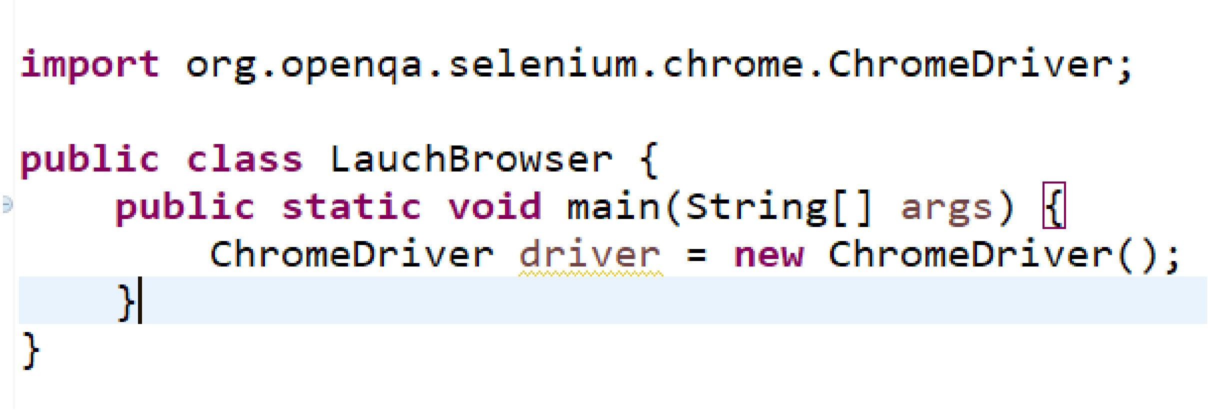
In order to launch the browser, we need to create the object of browser classes,

Example: In order launch the chrome browser, create the object of chromedriver class, same way for Firefox browser create the object of Firefoxdriver class.

Below is the program to launch the chrome browser.

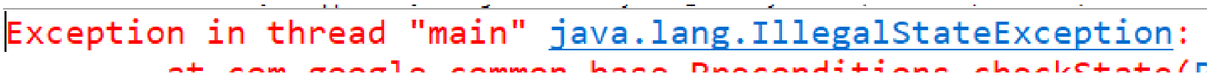
First, import the chromedriver class from the below package.

“org.openqa.selenium.chrome.ChromeDriver”



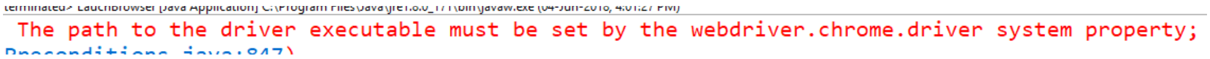
```
import org.openqa.selenium.chrome.ChromeDriver;

public class LauchBrowser {
    public static void main(String[] args) {
        ChromeDriver driver = new ChromeDriver();
    }
}
```

If you execute the above program you will get the below exception.



```
Exception in thread "main" java.lang.IllegalStateException:
```

With below message.



```
 The path to the driver executable must be set by the webdriver.chrome.driver system property;
```

Reason for the above exception: Selenium cannot control the chrome browser directly.

Selenium supports to run webdriver in browsers by just adding an .exe path of the driver server for the individual browsers.

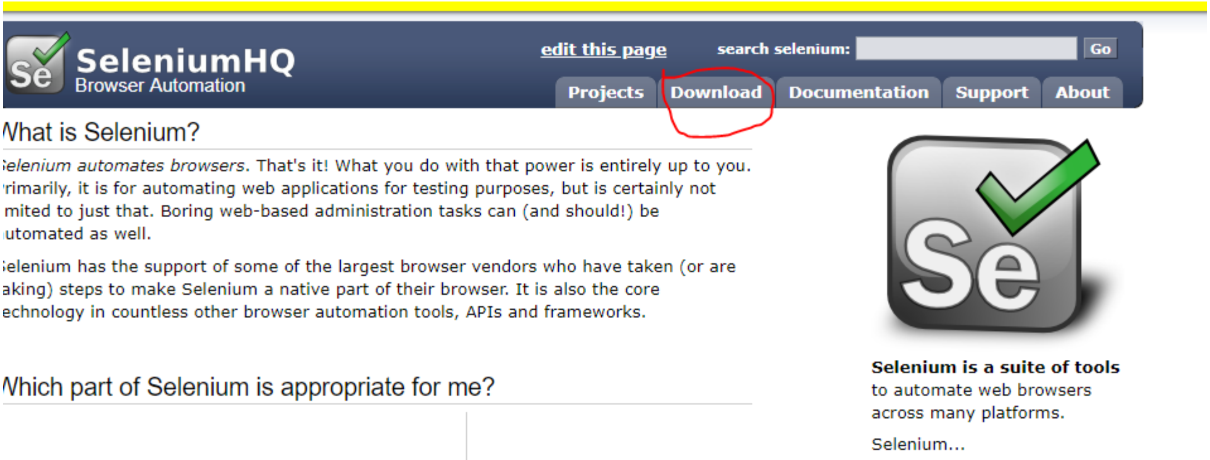
Now to run selenium webdriver in Chrome browser, we need to take the help of ChromeDriver which is a separate executable that selenium webdriver uses to control chrome.

Note: ChromeDriver is supported by the Chromium team, ChromeDriver is a standalone server which implements WebDriver's wire protocol for Chromium.

So, In order to work with the chrome or any other browser first we need to download the driver executable files from the below website.

“https://docs.seleniumhq.org/”

Click on the download tab in the above website



Vhat is Selenium?

ielenium automates browsers. That's it! What you do with that power is entirely up to you. 'rimarily, it is for automating web applications for testing purposes, but is certainly not mited to just that. Boring web-based administration tasks can (and should!) be utomated as well.

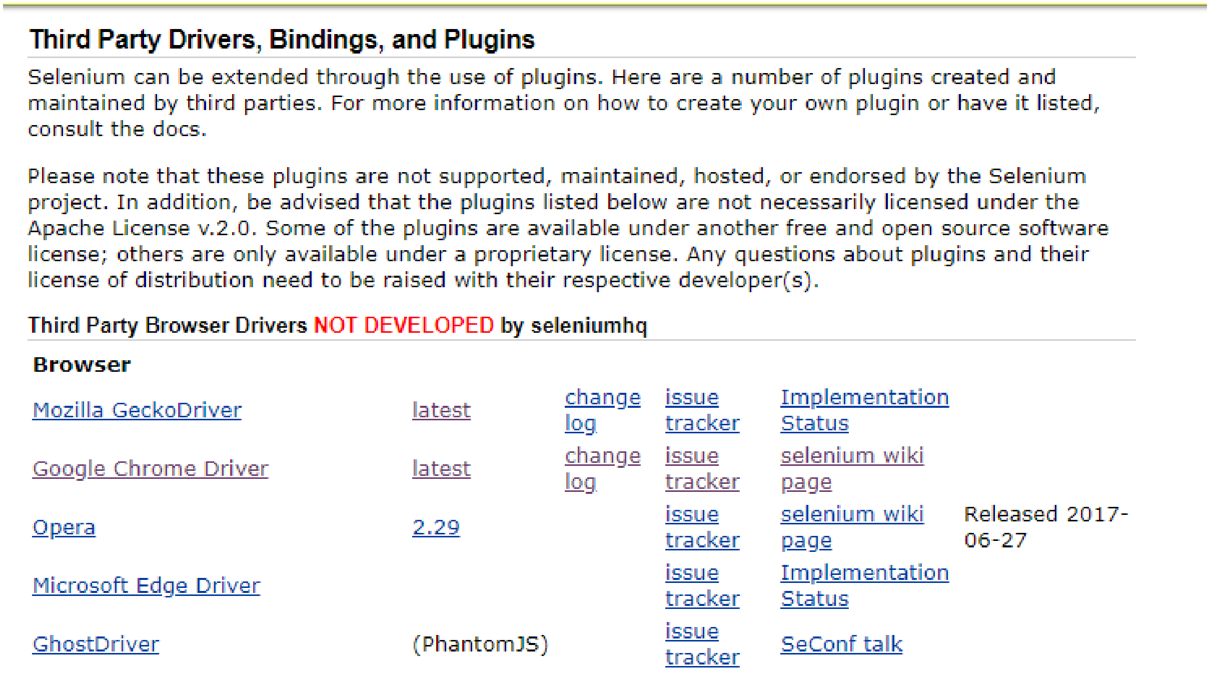
ielenium has the support of some of the largest browser vendors who have taken (or are aking) steps to make Selenium a native part of their browser. It is also the core echnology in countless other browser automation tools, APIs and frameworks.

Vhich part of Selenium is appropriate for me?

Selenium is a suite of tools to automate web browsers across many platforms.

Selenium...

Scroll down to the third party drivers, bindings, and plugins



Third Party Drivers, Bindings, and Plugins

Selenium can be extended through the use of plugins. Here are a number of plugins created and maintained by third parties. For more information on how to create your own plugin or have it listed, consult the docs.

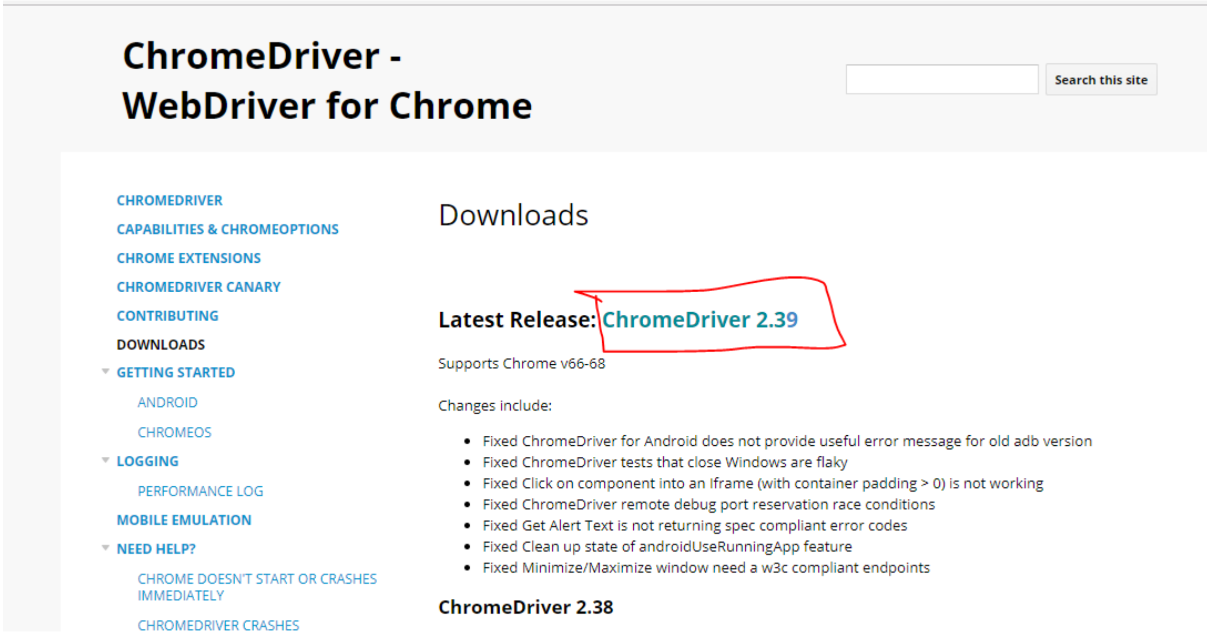
Please note that these plugins are not supported, maintained, hosted, or endorsed by the Selenium project. In addition, be advised that the plugins listed below are not necessarily licensed under the Apache License v.2.0. Some of the plugins are available under another free and open source software license; others are only available under a proprietary license. Any questions about plugins and their license of distribution need to be raised with their respective developer(s).

Third Party Browser Drivers NOT DEVELOPED by seleniumhq

Browser					
Mozilla GeckoDriver	latest	change log	issue tracker	Implementation Status	
Google Chrome Driver	latest	change log	issue tracker	selenium wiki page	
Opera	2.29		issue tracker	selenium wiki page	Released 2017-06-27
Microsoft Edge Driver			issue tracker	Implementation Status	
GhostDriver	(PhantomJS)		issue tracker	SeConf talk	

Click on the "latest" link of "Google Chrome Driver"

It navigates to the chromium web site. Click on the **ChromeDriver 2.39** link



ChromeDriver -
WebDriver for Chrome

Downloads

Latest Release: ChromeDriver 2.39

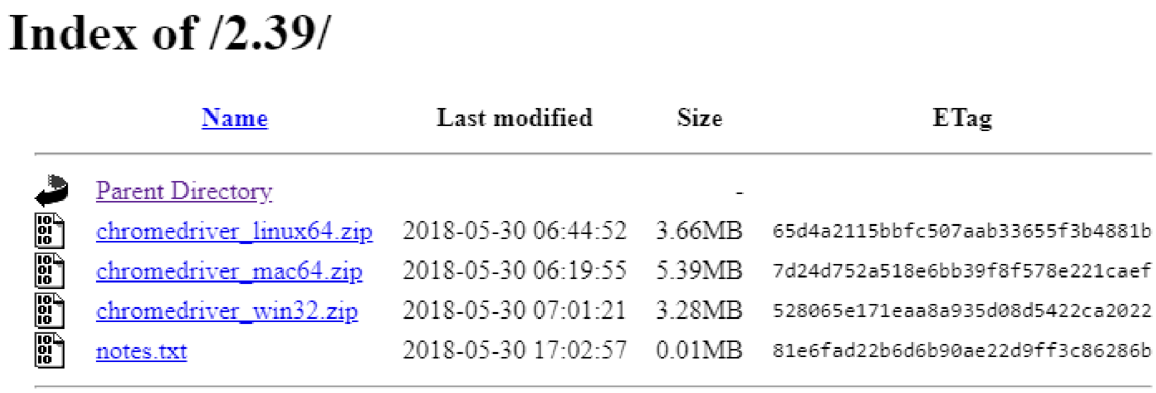
Supports Chrome v66-68

Changes include:

- Fixed ChromeDriver for Android does not provide useful error message for old adb version
- Fixed ChromeDriver tests that close Windows are flaky
- Fixed Click on component into an Iframe (with container padding > 0) is not working
- Fixed ChromeDriver remote debug port reservation race conditions
- Fixed Get Alert Text is not returning spec compliant error codes
- Fixed Clean up state of androidUseRunningApp feature
- Fixed Minimize/Maximize window need a w3c compliant endpoints

ChromeDriver 2.38

It navigates to the download page.



Index of /2.39/

Name	Last modified	Size	ETag
Parent Directory		-	
chromedriver_linux64.zip	2018-05-30 06:44:52	3.66MB	65d4a2115bbfc507aab33655f3b4881b
chromedriver_mac64.zip	2018-05-30 06:19:55	5.39MB	7d24d752a518e6bb39f8f578e221caef
chromedriver_win32.zip	2018-05-30 07:01:21	3.28MB	528065e171eaa8a935d08d5422ca2022
notes.txt	2018-05-30 17:02:57	0.01MB	81e6fad22b6d6b90ae22d9ff3c86286b

Download the chromedriver zip file based on your OS. If your OS is windows click on the chromedriver_win32.zip

Chromedriver_win32.zip file will be downloaded to your machine. Extract the zip file. You will get the chromedriver.exe file.

Now, you need to set the property for chrome driver exe file by using the method **"System.setProperty"**

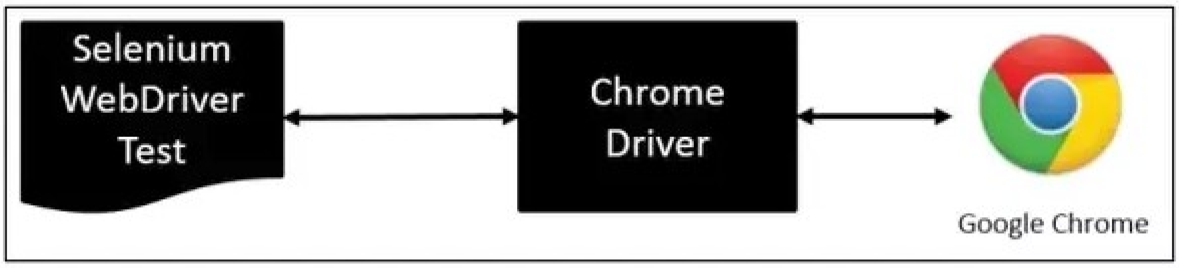
Syntax:

System.setProperty (String Key, String Value)

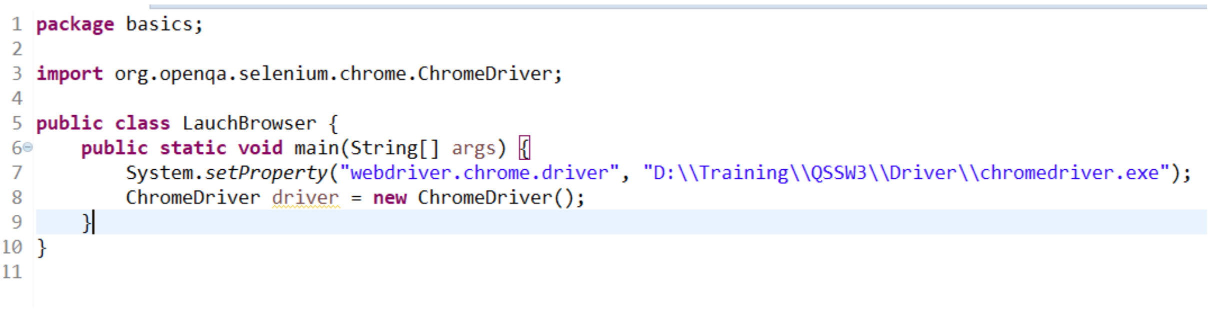
- System.setProperty method takes two arguments, key and value. For launching the chrome browser, we need to mention the below value as key **"webdriver.chrome.driver"** and path of chromedriver as a value **"D://driverExe/chromedriver.exe"**

Below is the key and value

Browser Name	Key	Value
Chrome	webdriver.chrome.driver	D://drivers/chromedriver.exe
Firefox	webdriver.gecko.driver	D://drivers/geckodriver.exe



Write a program to launch the chrome browser.



```java
package basics;

import org.openqa.selenium.chrome.ChromeDriver;

public class LauchBrowser {
    public static void main(String[] args) {
        System.setProperty("webdriver.chrome.driver", "D:\\Training\\QSSW3\\Driver\\chromedriver.exe");
        ChromeDriver driver = new ChromeDriver();
    }
}
```

Now if you run the above program, it will launch the chrome browser successfully.

Selenium always launch the default browser, i.e. it will always launch an empty browser. i.e.

- ➔ No history
- ➔ No add-ons
- ➔ No auto-completes
- ➔ No cookies