

## Installing and Creating Your First Application in ECLIPSE:

Selenium installation is a 3 step process:

1. Install Java SDK
2. Install Eclipse
3. Install Selenium Webdriver Files

### Step 1 – Install Java on your computer

Download and install the Java Software Development Kit (JDK)

ORACLE

Sign In/Register Help Country Communities I am a... I want to... Search

Products Solutions Downloads Store Support Training

Oracle Technology Network > Java > Java SE > Downloads

Overview Downloads Documentation Community Technologies Training

Java SE Downloads

Click on this button

Java Platform (JDK) 8u121

NetBeans with JDK 8

Java Platform, Standard Edition

Java SE 8u121

Java SE 8u121 includes important security fixes. Oracle strongly recommends that all Java users upgrade to this release.

Learn more

Next –

1 Click this radio button

Java SE Development Kit 8u121

You must accept the Oracle Binary Code License Agreement for Java SE to download this software.

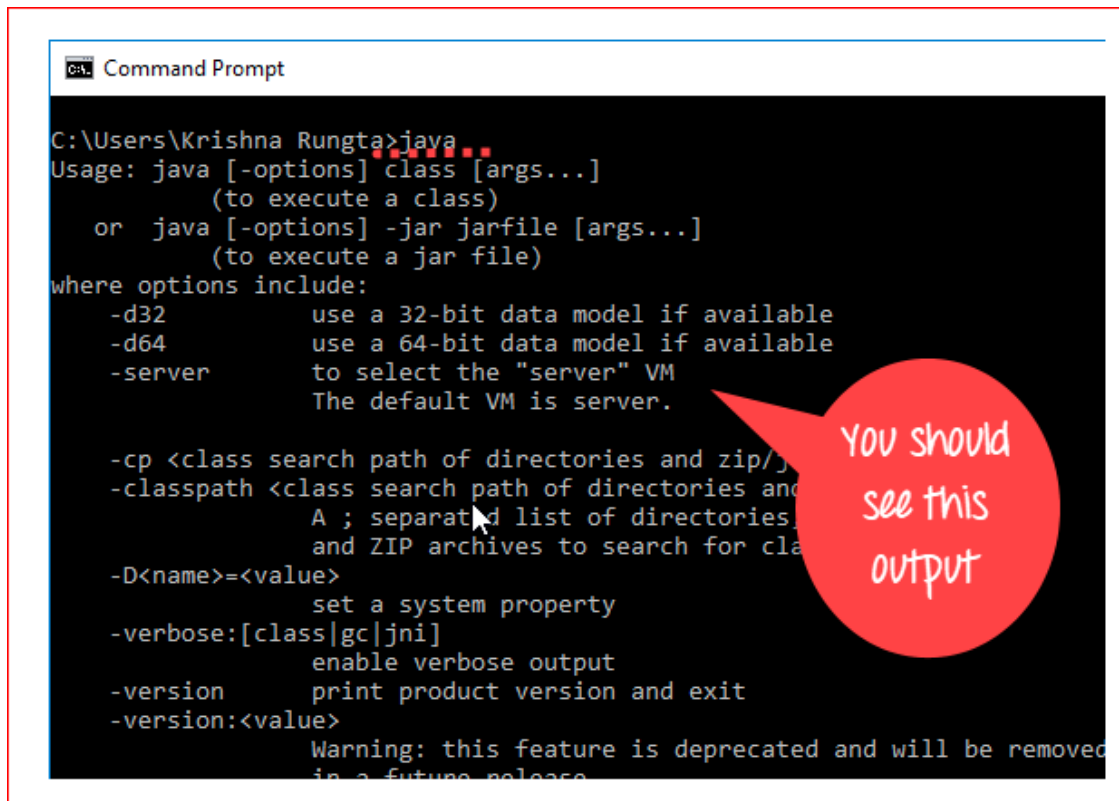
☐ Accept License Agreement ☒ Decline License Agreement

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	77.86 MB	<a href="#">jdk-8u121-linux-arm32-vfp-hflt.tar.gz</a>
Linux ARM 64 Hard Float ABI	74.83 MB	<a href="#">jdk-8u121-linux-arm64-vfp-hflt.tar.gz</a>
Linux x86	162.41 MB	<a href="#">jdk-8u121-linux-i586.rpm</a>
Linux x86	177.13 MB	<a href="#">jdk-8u121-linux-i586.tar.gz</a>
Linux x64	159.96 MB	<a href="#">jdk-8u121-linux-x64.rpm</a>
Linux x64	174.76 MB	<a href="#">jdk-8u121-linux-x64.tar.gz</a>
Mac OS X	223.21 MB	<a href="#">jdk-8u121-macosx-x64.dmg</a>
Solaris SPARC 64-bit	139.64 MB	<a href="#">jdk-8u121-solaris-sparcv9.tar.Z</a>
Solaris SPARC 64-bit	99.07 MB	<a href="#">jdk-8u121-solaris-sparcv9.tar.gz</a>
Solaris x64	140.42 MB	<a href="#">jdk-8u121-solaris-x64.tar.Z</a>
Solaris x64	96.9 MB	<a href="#">jdk-8u121-solaris-x64.tar.gz</a>
Windows x86	189.36 MB	<a href="#">jdk-8u121-windows-i586.exe</a>
Windows x64	195.51 MB	<a href="#">jdk-8u121-windows-x64.exe</a>

2 Choose the JDK that corresponds to your OS

This JDK version comes bundled with Java Runtime Environment (JRE), so you do not need to download and install the JRE separately.

Once installation is complete, open command prompt and type “java”. If you see the following screen you are good to move to the next step

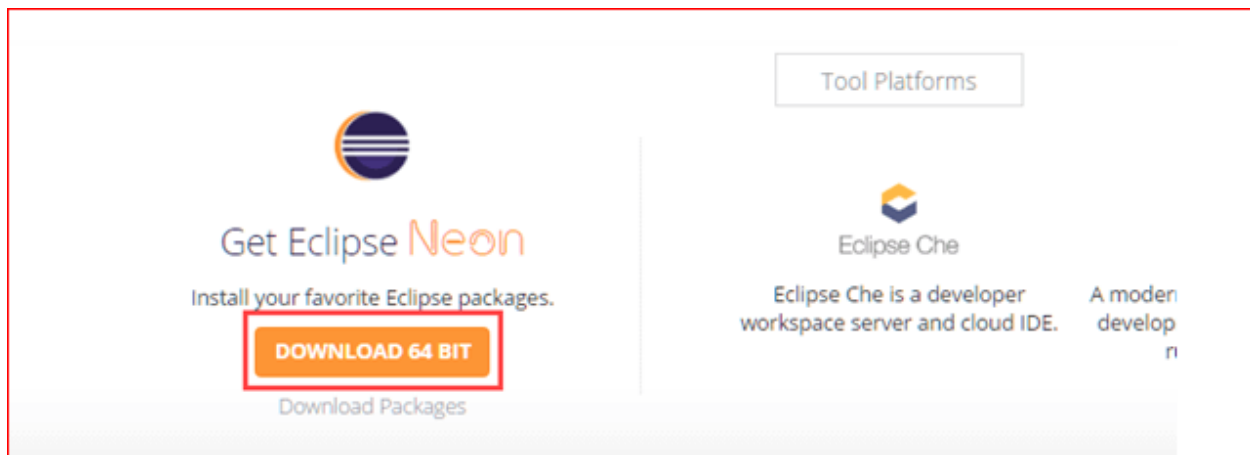


```
Command Prompt
C:\Users\Krishna Rungta>java
Usage: java [-options] class [args...]
           (to execute a class)
 or  java [-options] -jar jarfile [args...]
           (to execute a jar file)
where options include:
  -d32          use a 32-bit data model if available
  -d64          use a 64-bit data model if available
  -server       to select the "server" VM
                 The default VM is server.

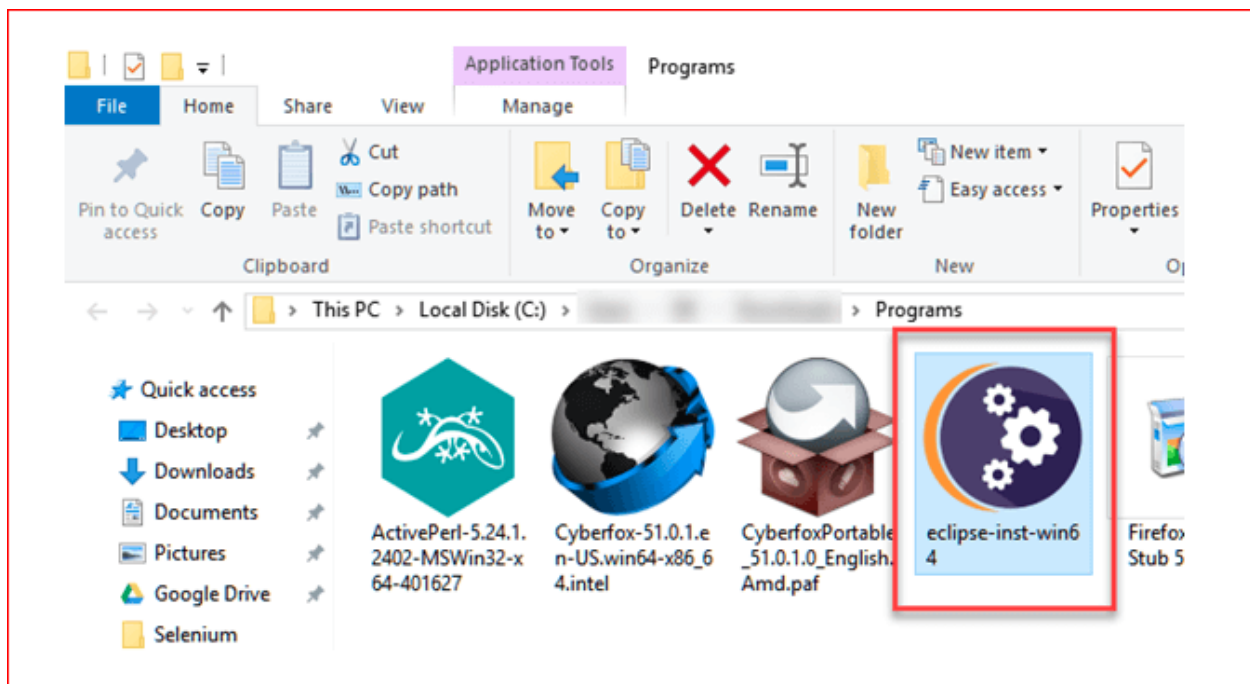
  -cp <class search path of directories and zip/jar files>
  -classpath <class search path of directories and zip/jar files>
                 A ; separated list of directories,
                 and ZIP archives to search for class files.
  -D<name>=<value>
                 set a system property
  -verbose:[class|gc|jni]
                 enable verbose output
  -version      print product version and exit
  -version:<value>
                 Warning: this feature is deprecated and will be removed
                 in a future release.
```

## Step 2 – Install Eclipse IDE

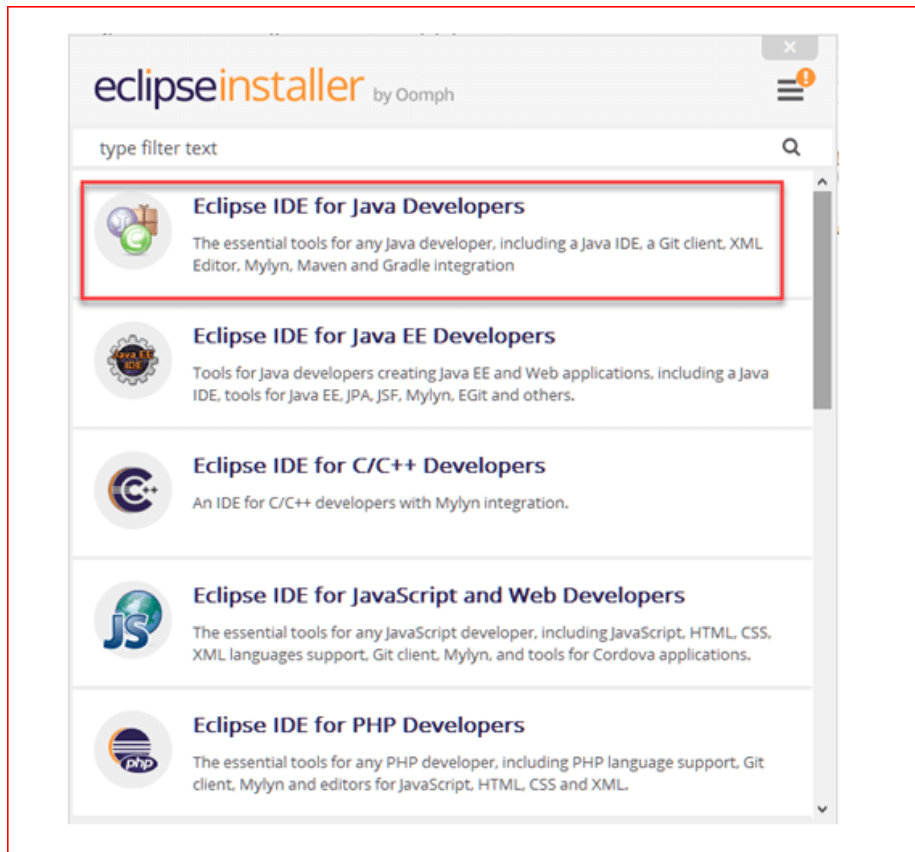
Download latest version of “Eclipse IDE for Java Developers” [here](#). Be sure to choose correctly between Windows 32 Bit and 64 Bit versions.



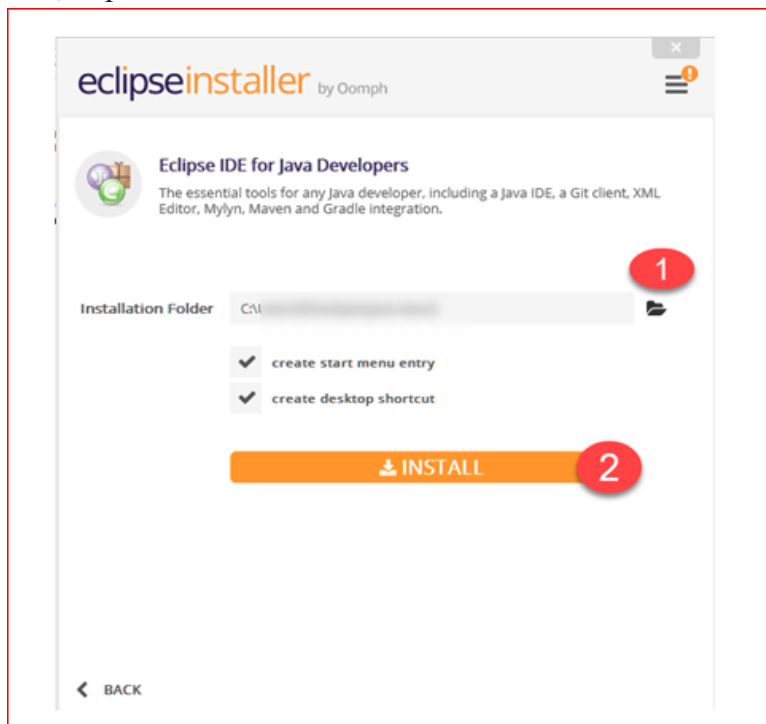
You should be able to download an exe file named “eclipse-inst-win64” for Setup.



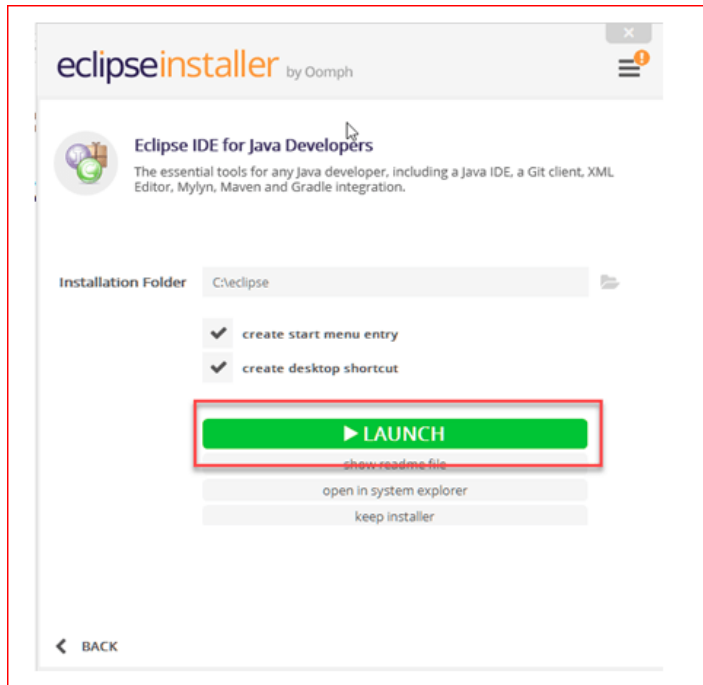
Double-click on file to Install the Eclipse. A new window will open. Click Eclipse IDE for Java Developers.



After that, a new window will open which click button marked 1 and change path to “C:\eclipse”. Post that Click on Install button marked 2



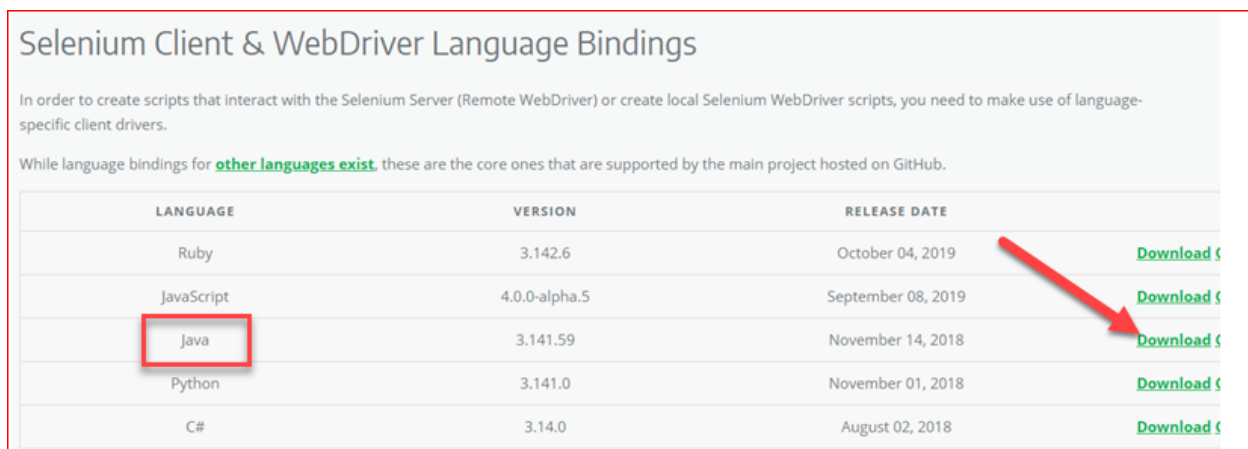
After successful completion of the installation procedure, a window will appear. On that window click on Launch



This will start eclipse neon IDE for you.

### Step 3 – Download the Selenium Java Client Driver

You can download **Selenium Webdriver for Java Client Driver** [here](#). You will find client drivers for other languages there, but only choose the one for Java.



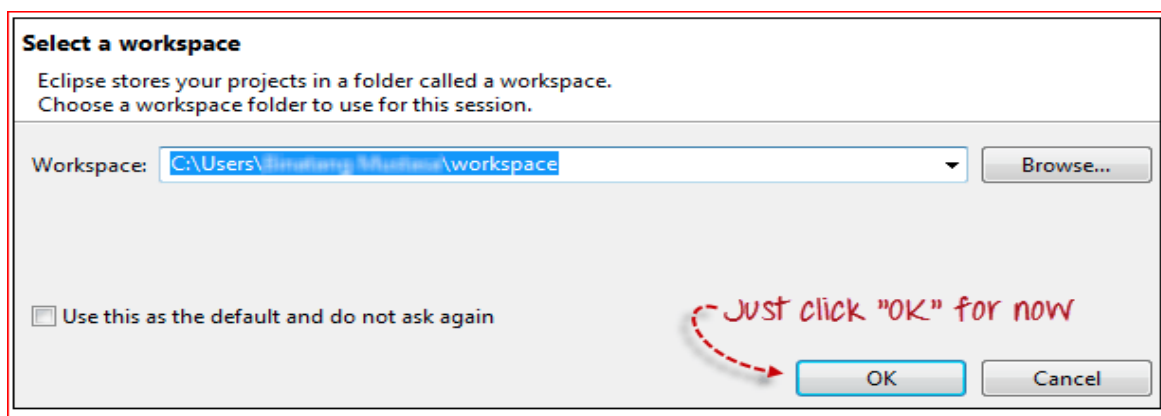
In order to create scripts that interact with the Selenium Server (Remote WebDriver) or create local Selenium WebDriver scripts, you need to make use of language-specific client drivers.			
While language bindings for <a href="#">other languages exist</a> , these are the core ones that are supported by the main project hosted on GitHub.			
LANGUAGE	VERSION	RELEASE DATE	
Ruby	3.142.6	October 04, 2019	<a href="#">Download ↗</a>
JavaScript	4.0.0-alpha.5	September 08, 2019	<a href="#">Download ↗</a>
Java	3.141.59	November 14, 2018	<a href="#">Download ↗</a>
Python	3.141.0	November 01, 2018	<a href="#">Download ↗</a>
C#	3.14.0	August 02, 2018	<a href="#">Download ↗</a>

This download comes as a ZIP file named “selenium-3.14.0.zip”. For simplicity of Selenium

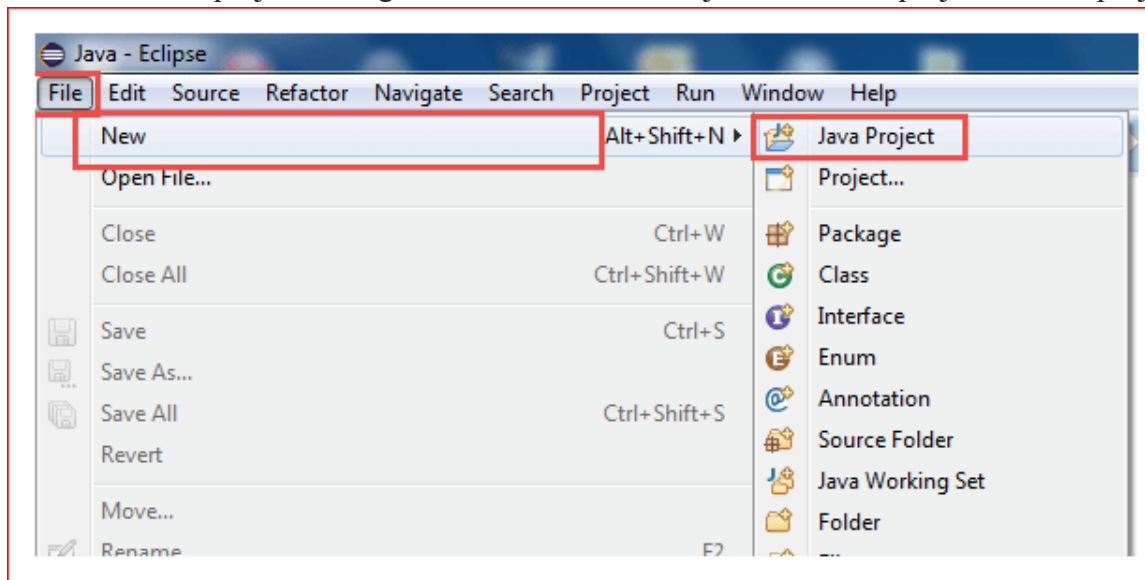
installation on Windows 10, extract the contents of this ZIP file on your C drive so that you would have the directory “C:\selenium-3.14.0”. This directory contains all the JAR files that we would later import on Eclipse for Selenium setup.

## Step 4 – Configure Eclipse IDE with WebDriver

1. Launch the “eclipse.exe” file inside the “eclipse” folder that we extracted in step 2. If you followed step 2 correctly, the executable should be located on C:\eclipse\eclipse.exe.
2. When asked to select for a workspace, just accept the default location.



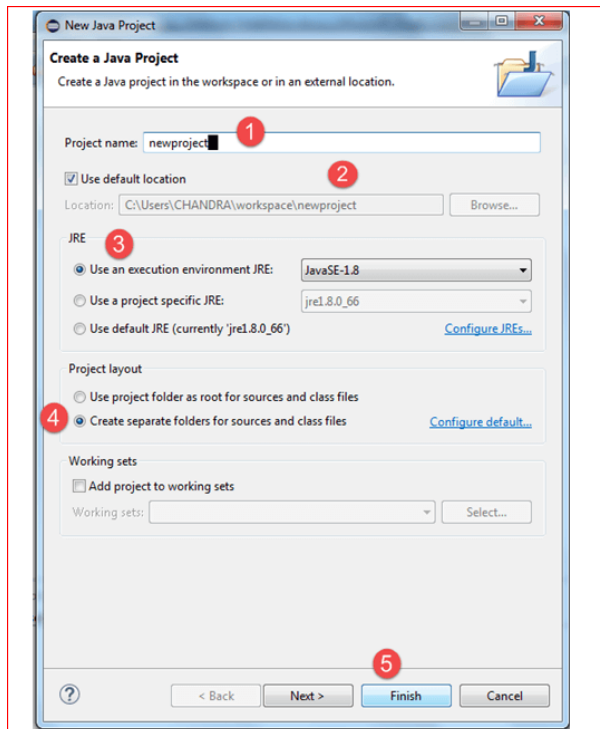
3. Create a new project through File > New > Java Project. Name the project as “newproject”.



A new pop-up window will open enter details as follow

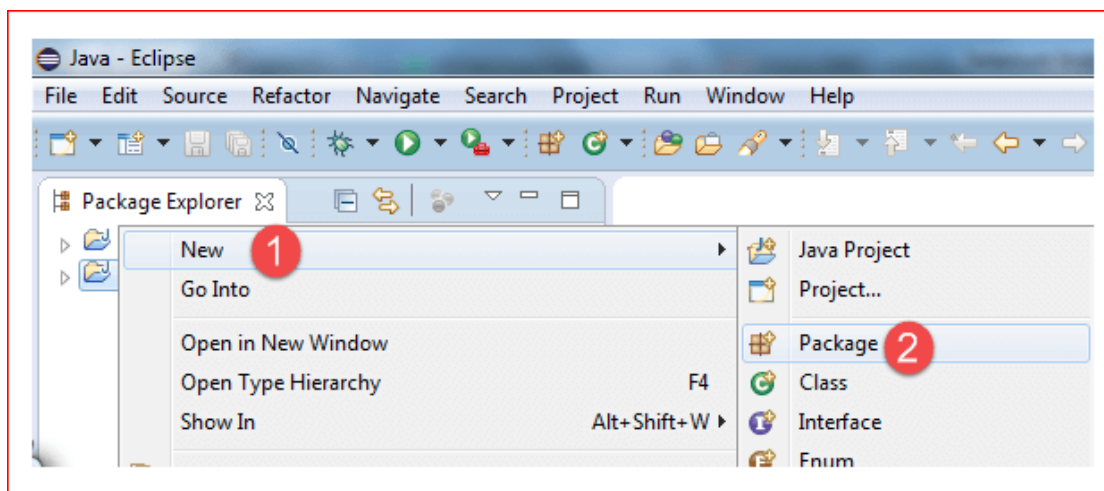
1. Project Name
2. Location to save project

3. Select an execution JRE
4. Select layout project option
5. Click on Finish button



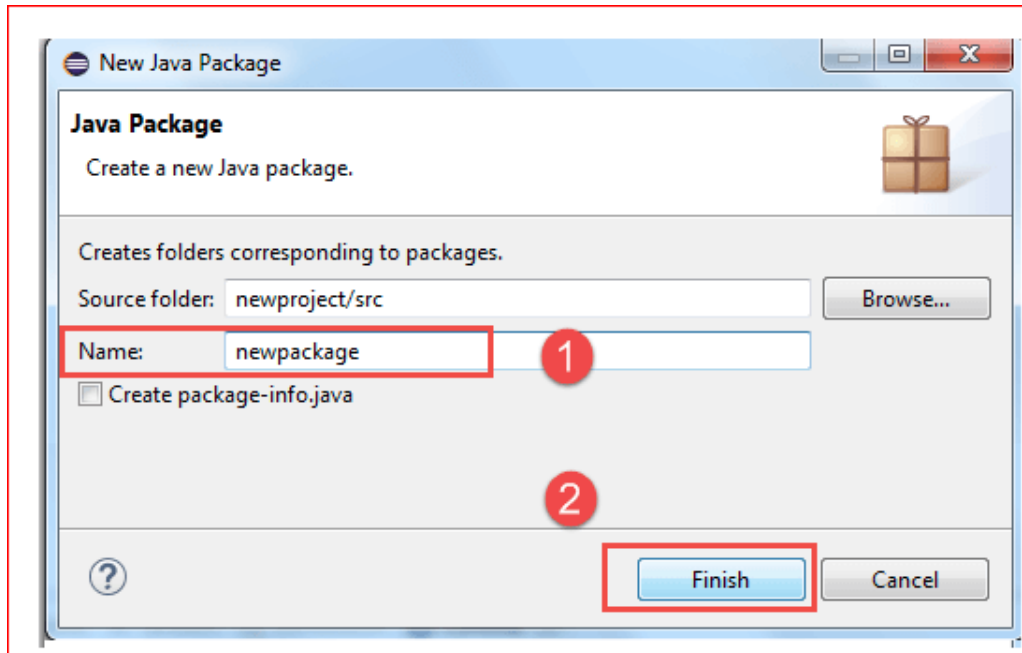
4. In this step,

1. Right-click on the newly created project and
2. Select New > Package, and name that package as “newpackage”.

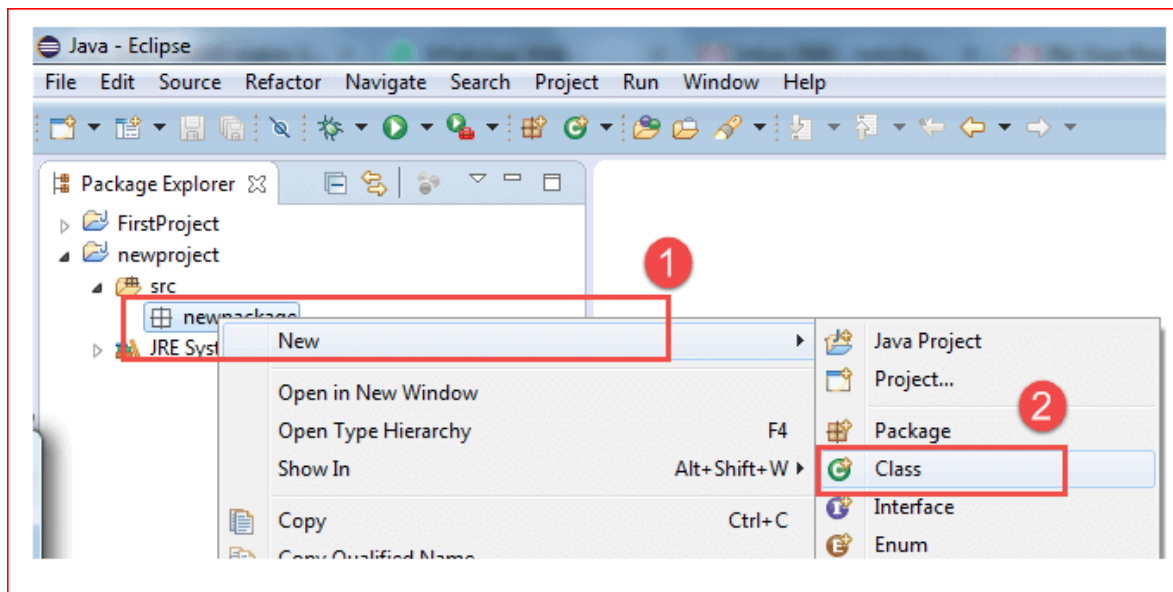


A pop-up window will open to name the package,

1. Enter the name of the package
2. Click on Finish button



5. Create a new Java class under newpackage by right-clicking on it and then selecting- New > Class, and then name it as “MyClass”. Your Eclipse IDE should look like the image below.

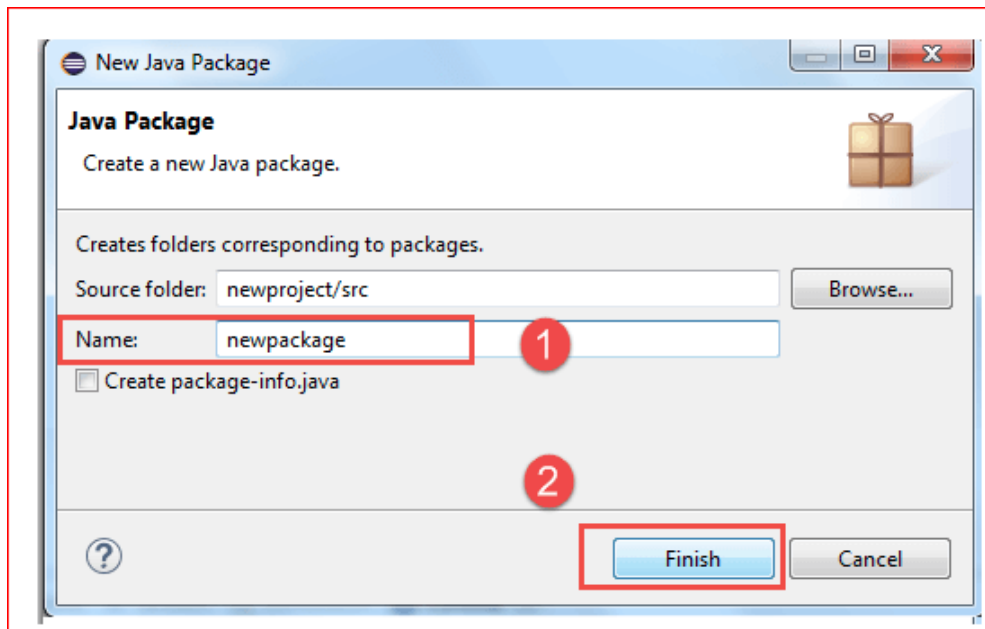


When you click on Class, a pop-up window will open, enter details as

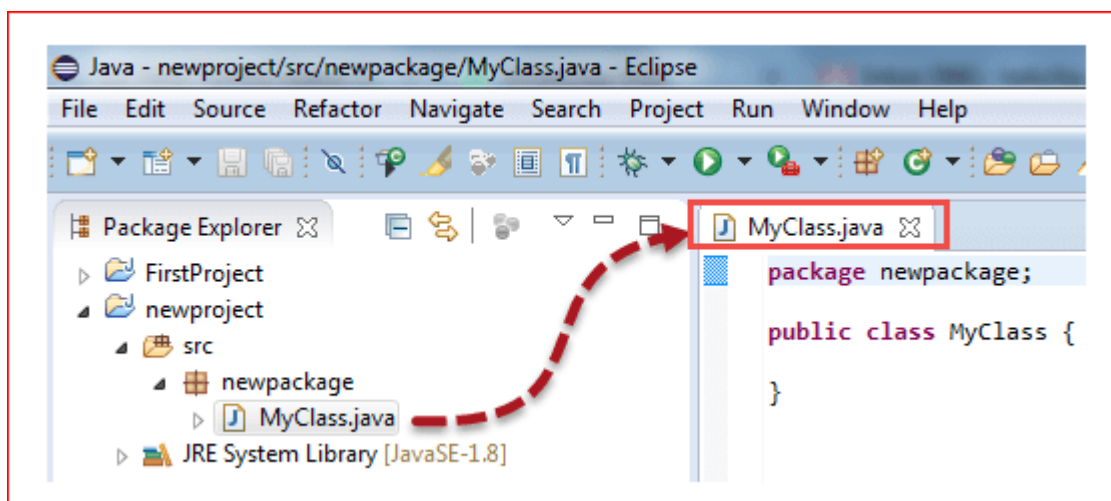
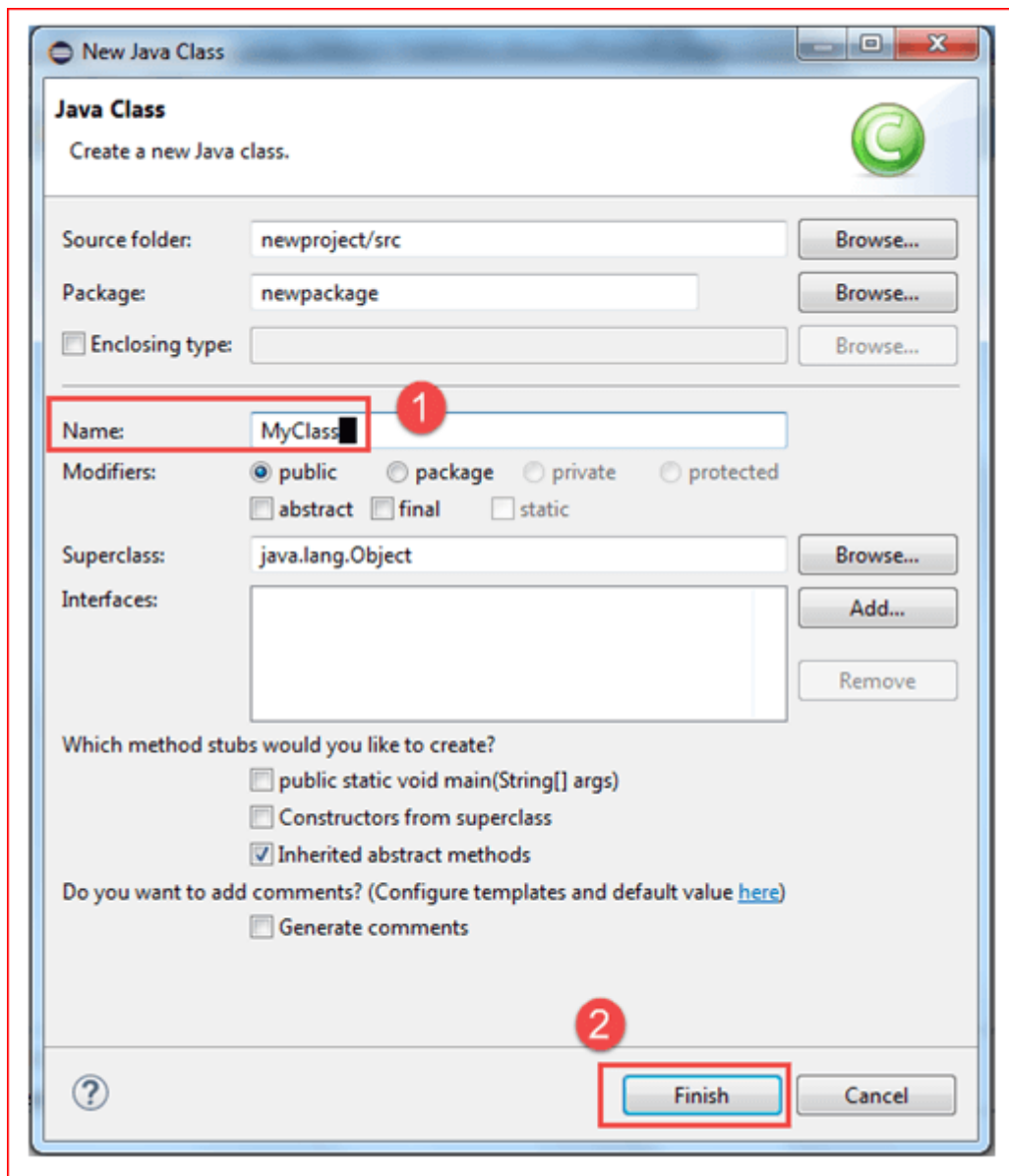
1. Name of the class



2. Click on Finish button

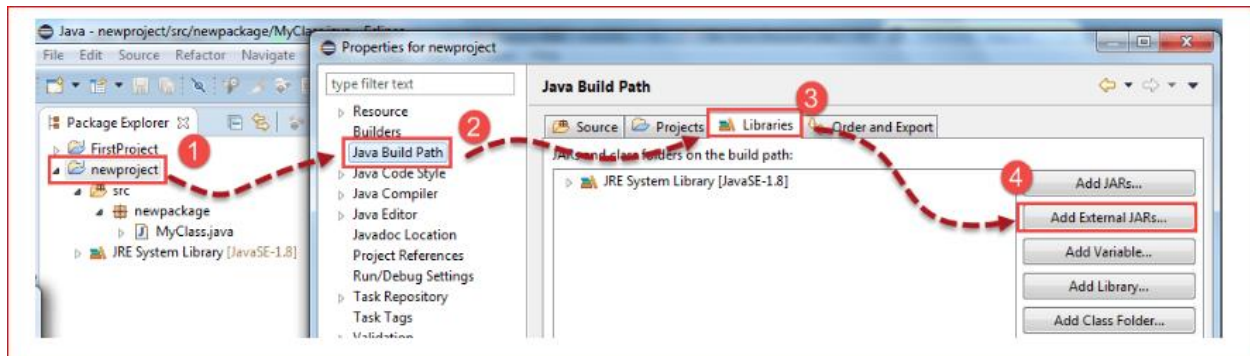


This is how it looks like after creating class.

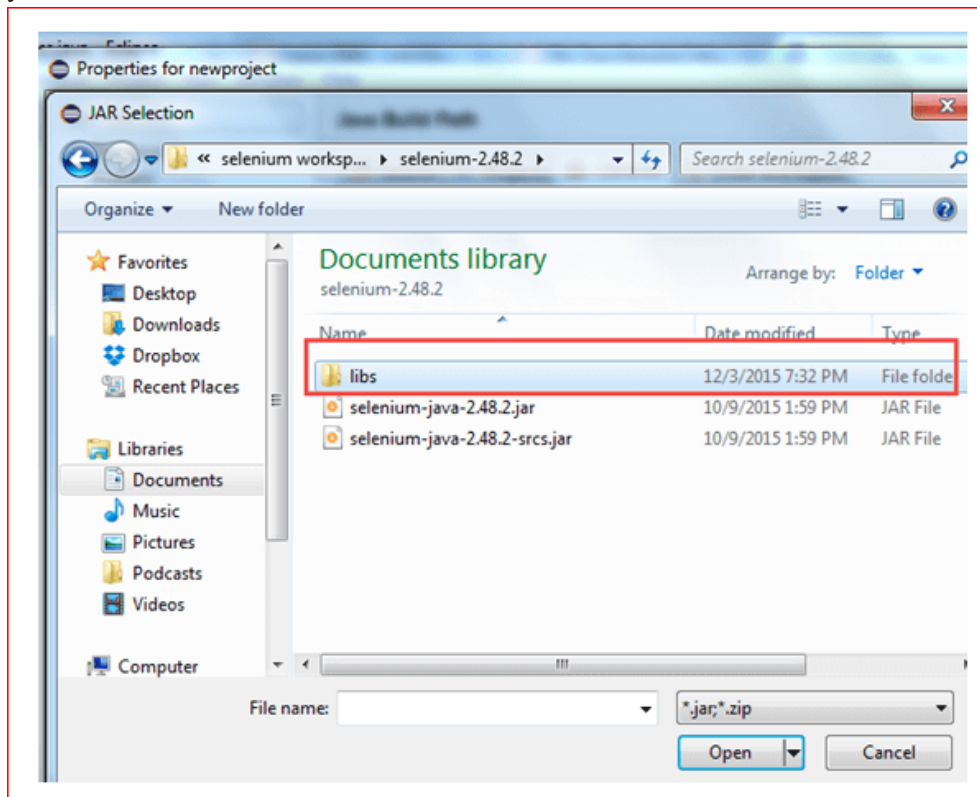


Now selenium WebDriver's into Java Build Path  
In this step,

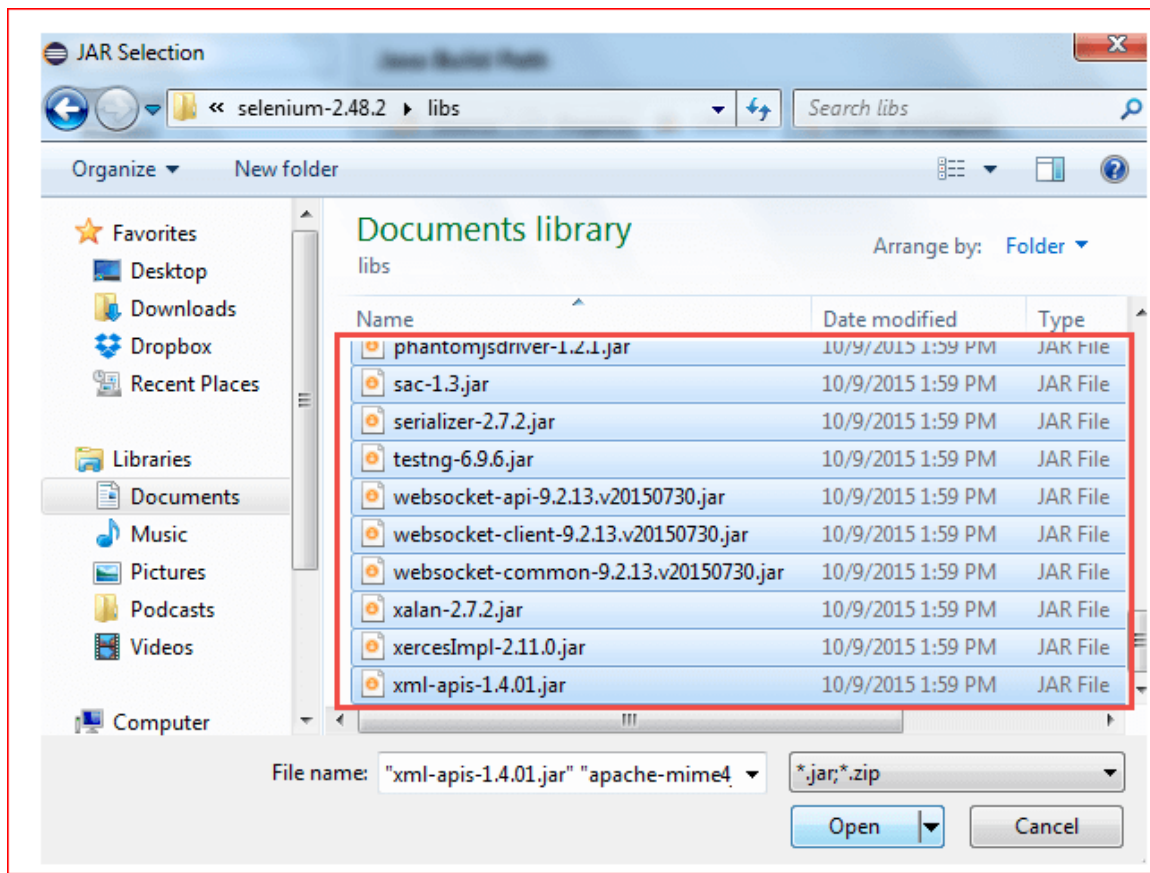
1. Right-click on “newproject” and select **Properties**.
2. On the Properties dialog, click on “Java Build Path”.
3. Click on the **Libraries** tab, and then
4. Click on “Add External JARs..”



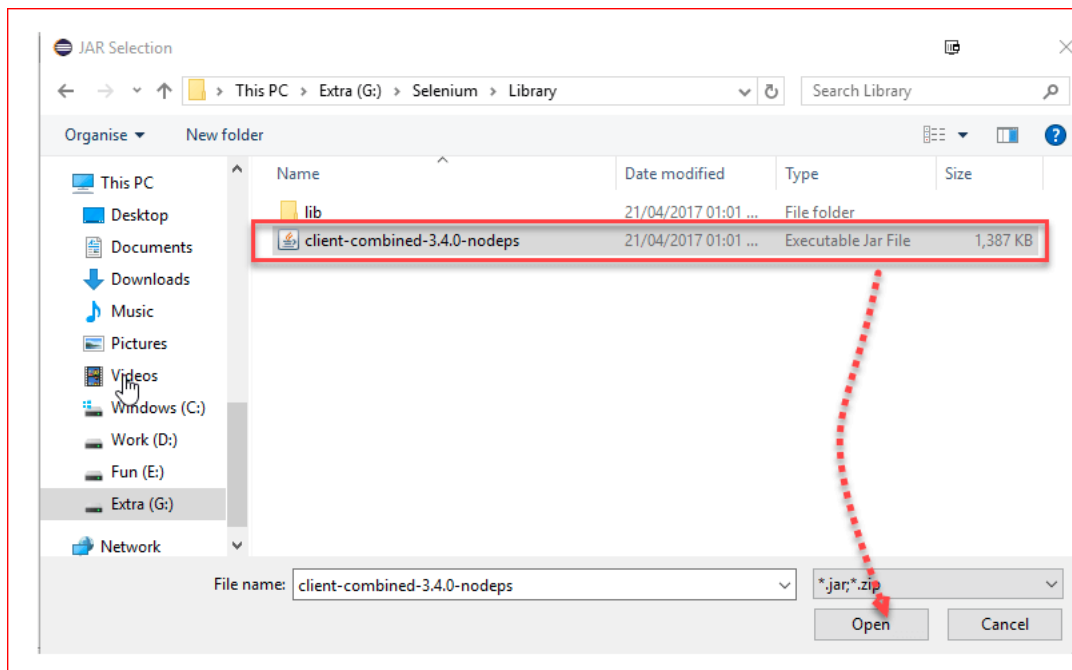
When you click on “Add External JARs..” It will open a pop-up window. Select the JAR files you want to add.



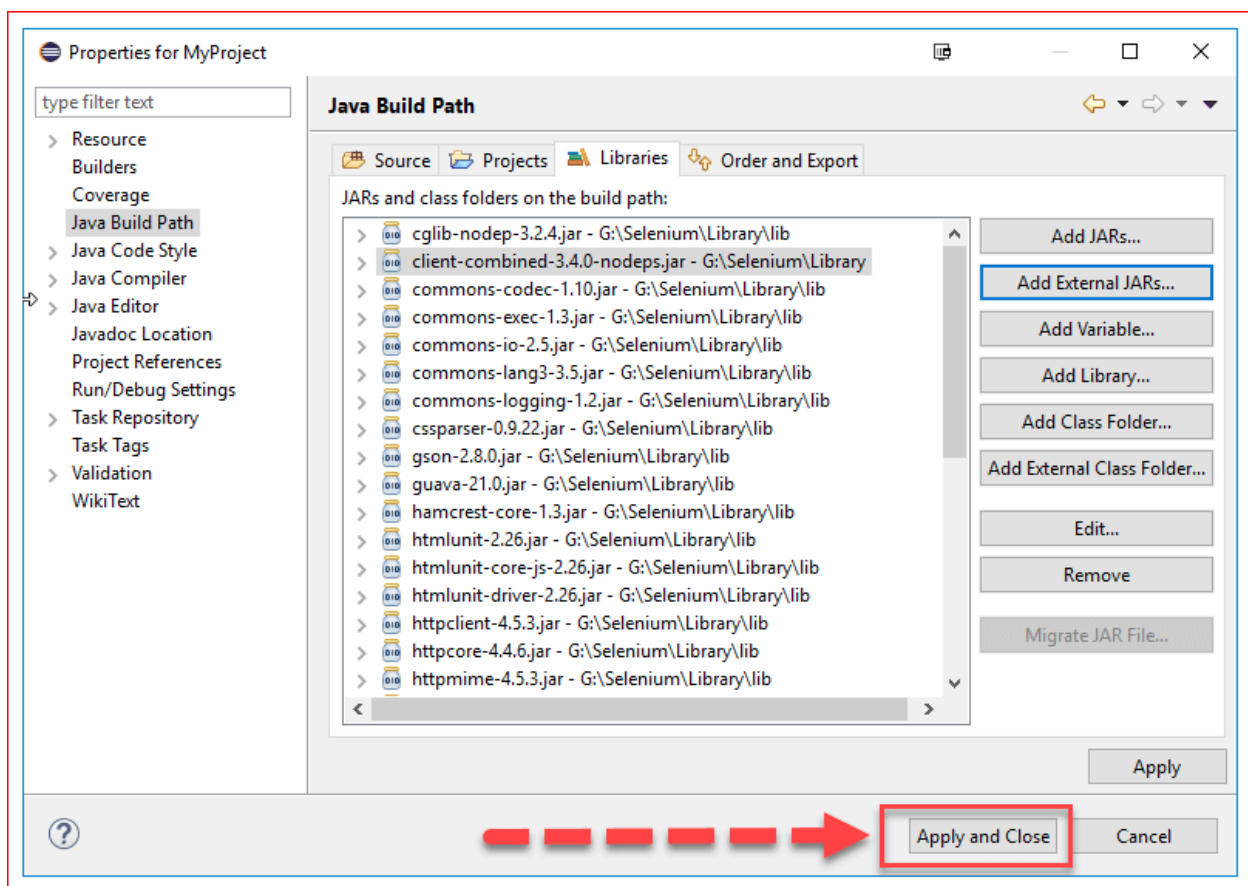
After selecting jar files, click on OK button.  
Select all files inside the lib folder.



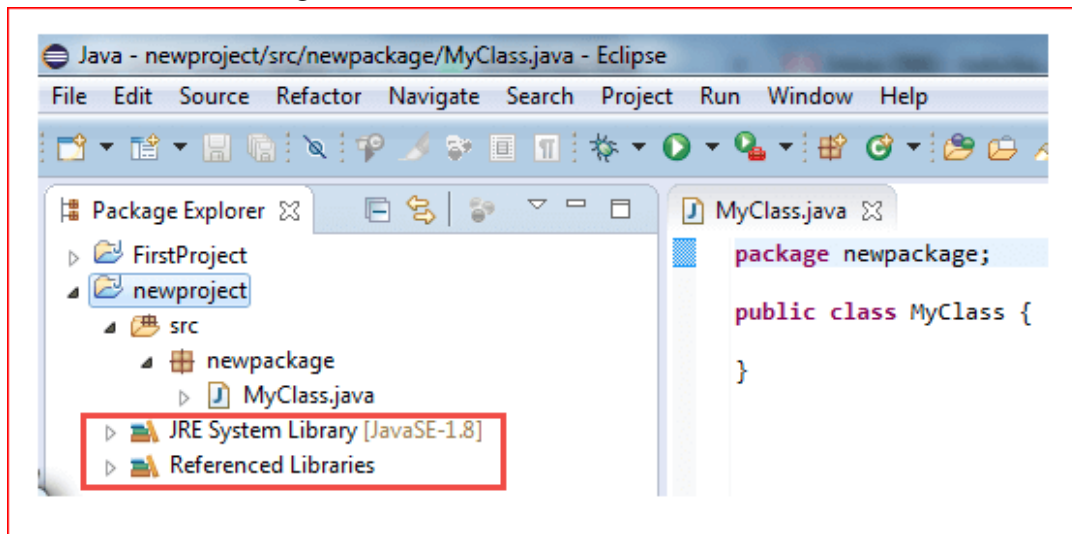
Select files outside lib folder



Once done, click “Apply and Close” button



6. Add all the JAR files inside and outside the “libs” folder. Your Properties dialog should now look similar to the image below.



7. Finally, click OK and we are done importing Selenium libraries into our project.

## WEEK PROGRAMS

### WEEK-1

**AIM: Download and Install JAVA, Associate SWD Jars and Browser drivers.**

**ALGORITHM:**

1. How to install JAVA
2. How to create and associate with SWD jars
3. How to select the browser drivers and make them to install.
4. How to convert browser drivers into .exe format.

**SOURCE CODE:**

### Install Java on Windows 10

After downloading the installation file, proceed with installing Java on your Windows system.

Follow the steps below:

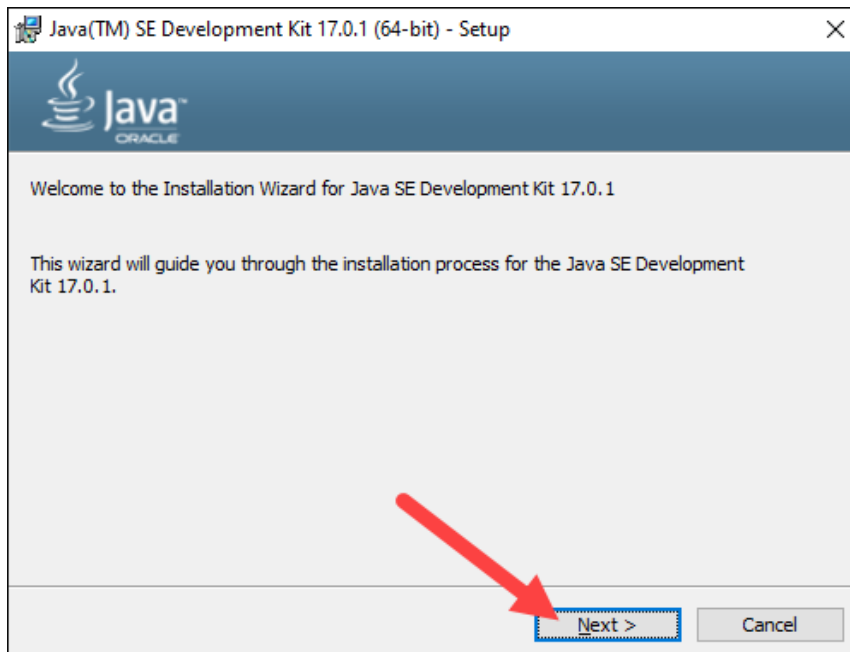
**Step 1: Run the Downloaded File**

Double-click the **downloaded file** to start the installation.

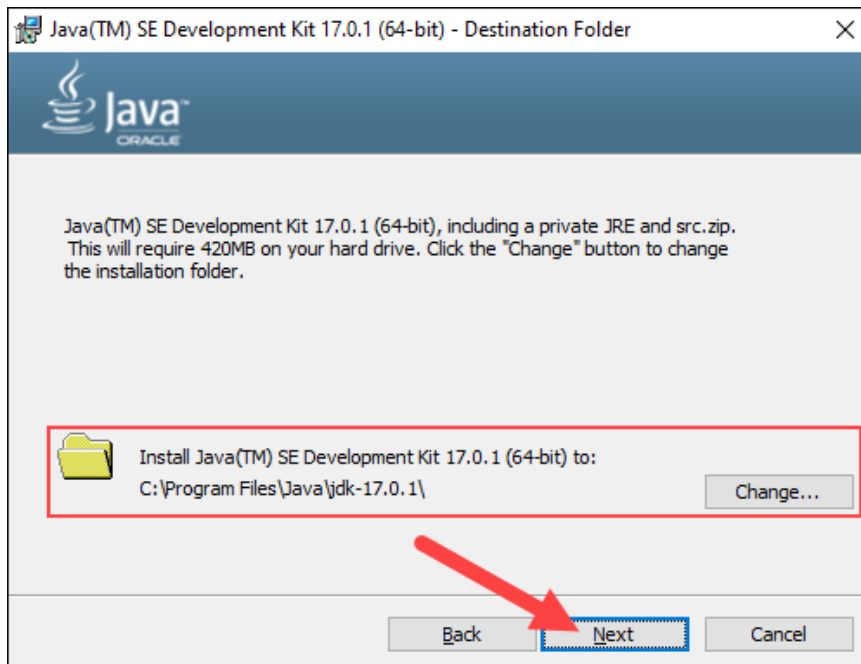
**Step 2: Configure the Installation Wizard**

After running the installation file, the installation wizard welcome screen appears.

1. Click **Next** to proceed to the next step.



3. Choose the destination folder for the Java installation files or stick to the default path. Click **Next** to proceed.



5. Wait for the wizard to finish the installation process until the *Successfully Installed* message appears. Click **Close** to exit the wizard.





## **Associate SWD Jars : [selenium web driver]**

### Selenium Server (Grid)

The Selenium Server is needed in order to run Remote Selenium WebDriver (Grid).

Selenium Grid allows the execution of WebDriver scripts on remote machines (virtual or real) by routing commands sent by the client to remote browser instances. It aims to provide an easy way to run tests in parallel on multiple machines.

Selenium Grid allows us to run tests in parallel on multiple machines, and to manage different browser versions and browser configurations centrally (instead of in each individual test).

Selenium Grid is not a silver bullet. It solves a subset of common delegation and distribution problems, but will for example not manage your infrastructure, and might not suit your specific needs.

### Purposes and main functionalities

- Central entry point for all tests
- Management and control of the nodes / environment where the browsers run
- Scaling
- Running tests in parallel
- Cross-platform testing
- Load balancing

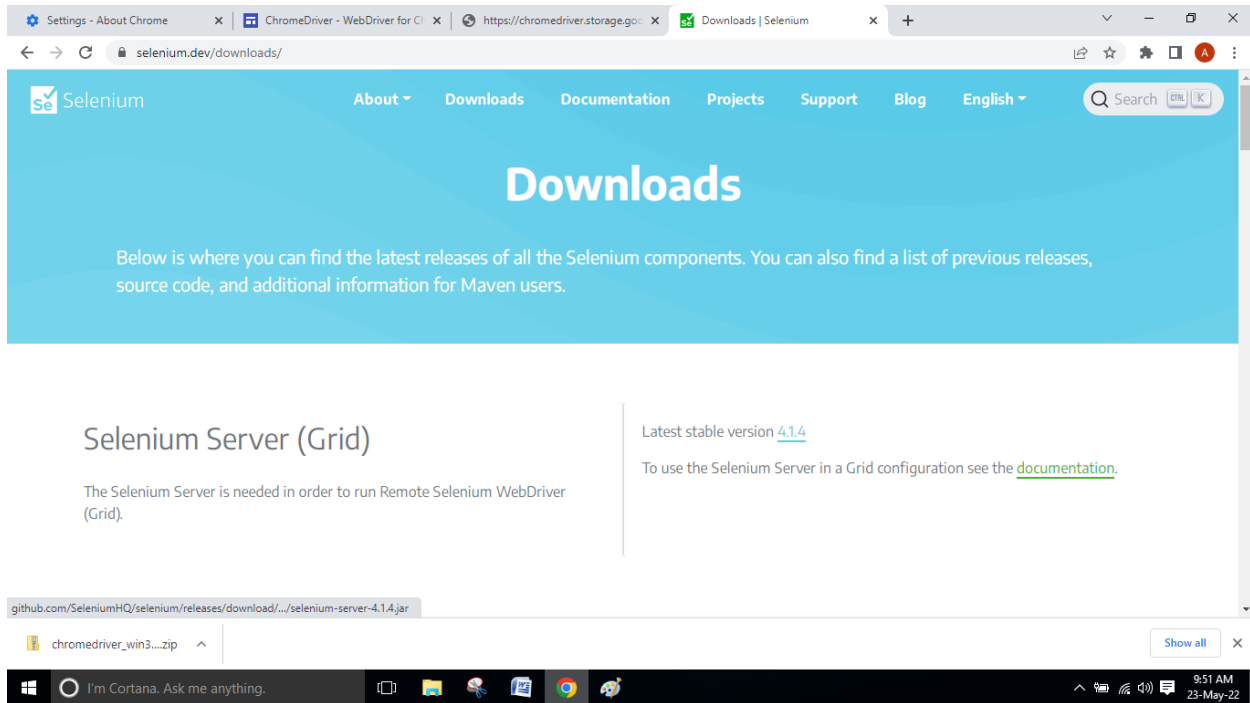
## **Selenium Grid 4**

Grid 4 takes advantage of a number of new technologies in order to facilitate scaling up while allowing local execution.

Selenium Grid 4 is a fresh implementation and does not share the codebase the previous version had.

To get all the details of Grid 4 components, understand how it works, and how to set up you own, please browse thorough the following sections.

// Latest stable version [4.1.4](#)



## **Browser drivers :**

### **Install browser drivers**

Setting up your system to allow a browser to be automated.

Through WebDriver, Selenium supports all major browsers on the market such as Chrome/Chromium, Firefox, Internet Explorer, Edge, Opera, and Safari. Where possible, WebDriver drives the browser using the browser's built-in support for automation.

Since all the driver implementations except for Internet Explorer are provided by the browser vendors themselves, they are not included in the standard Selenium distribution. This section explains the basic requirements for getting you started with the different browsers.

### Three Ways to Use Drivers

#### **1. Driver Management Software**

Most machines automatically update the browser, but the driver does not. To make sure you get the correct driver for your browser, there are many third party libraries to assist you.

1. Import [WebDriver Manager](#)

```
import io.github.bonigarcia.wdm.WebDriverManager;
```

Copy

2. Calling `setup()` automatically puts the correct browser driver where the code will see it:

```
WebDriverManager.chromedriver().setup();
```

Copy

3. Just initialize the driver as you normally would:

```
ChromeDriver driver = new ChromeDriver();
```

## 2. The PATH Environment Variable

This option first requires manually downloading the driver.

This is a flexible option to change location of drivers without having to update your code, and will work on multiple machines without requiring that each machine put the drivers in the same place.

You can either place the drivers in a directory that is already listed in `PATH`, or you can place them in a directory and add it to `PATH`.

To see what directories are already on `PATH`, open a Command Prompt and execute:

```
echo %PATH%
```

Copy

If the location to your driver is not already in a directory listed, you can add a new directory to `PATH`:

```
setx PATH "%PATH%;C:\WebDriver\bin"
```

Copy

You can test if it has been added correctly by starting the driver:

```
chromedriver.exe
```

Copy

If your `PATH` is configured correctly above, you will see some output relating to the startup of the driver:

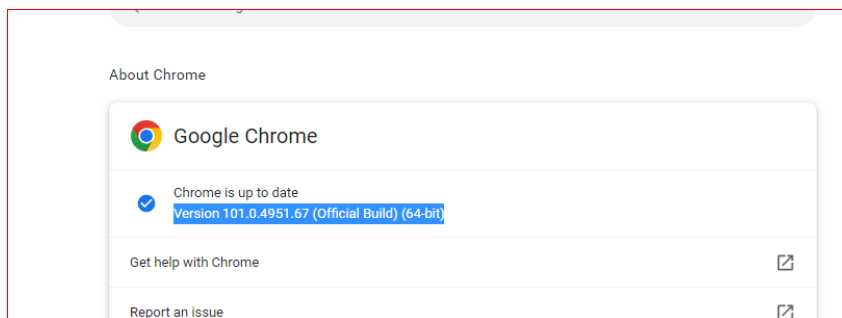
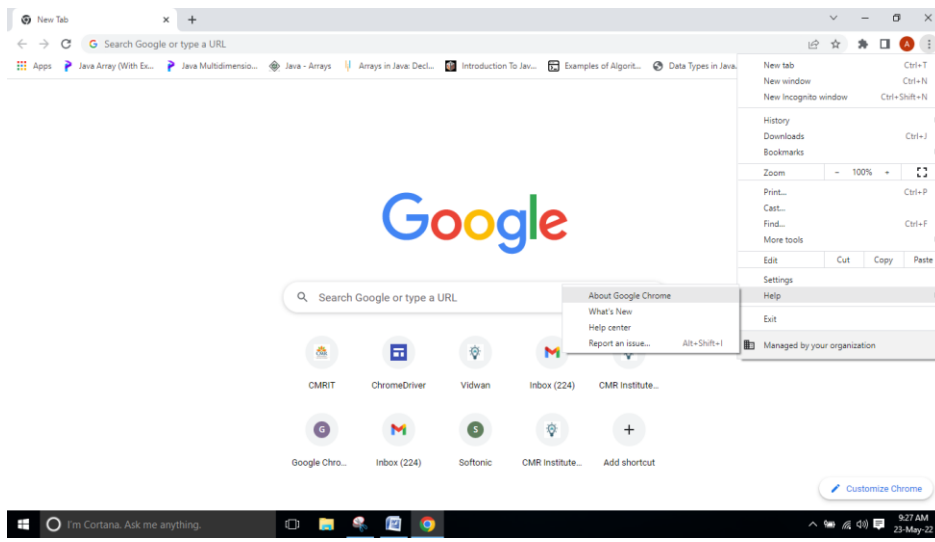
```
Starting ChromeDriver 95.0.4638.54 (d31a821ec901f68d0d34ccdbaea45b4c86ce543e-refs/branch-heads/4638@{#871}) on port 9515
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.
ChromeDriver was started successfully.
```

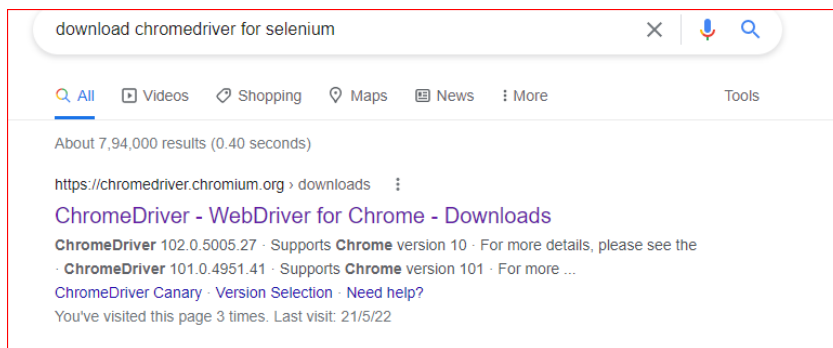
### 3. Hard Coded Location

Similar to Option 2 above, you need to manually download the driver. Specifying the location in the code itself has the advantage of not needing to figure out Environment Variables on your system, but has the drawback of making the code much less flexible.

```
System.setProperty("webdriver.chrome.driver", "/path/to/chromedriver");  
ChromeDriver driver = new ChromeDriver();
```

#### STEPS TO INSTALL CHROME DRIVER :











## Current Releases

- If you are using Chrome version 102, please download [ChromeDriver 102.0.5005.27](#)
- If you are using Chrome version 101, please download [ChromeDriver 101.0.4951.41](#)
- If you are using Chrome version 100, please download [ChromeDriver 100.0.4896.60](#)

## Index of /101.0.4951.41/

	<u>Name</u>	Last modified	Size	ETag
	<a href="#">Parent Directory</a>	-	-	-
	<a href="#">chromedriver_linux64.zip</a>	2022-04-27 07:02:29	5.92MB	57fc88db21f5d009cdf526480378
	<a href="#">chromedriver_mac64.zip</a>	2022-04-27 07:02:31	7.88MB	1589eb6b65c5a6848d44dd43c881
	<a href="#">chromedriver_mac64_m1.zip</a>	2022-04-27 07:02:34	7.19MB	d6d6cfd06ca5139f3663d2e68a8
	<a href="#">chromedriver_win32.zip</a>	2022-04-27 07:02:37	6.05MB	594669544f54e61c3762252d1a85
	<a href="#">notes.txt</a>	2022-04-27 07:02:42	0.00MB	c63873505b72aa1911a152618e25

