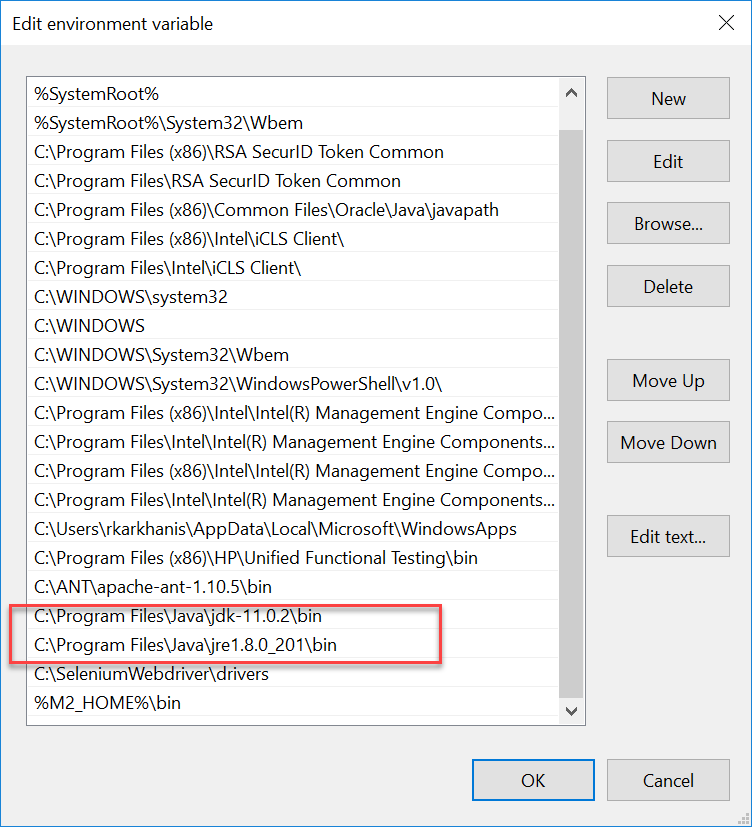
**Prerequisites:**

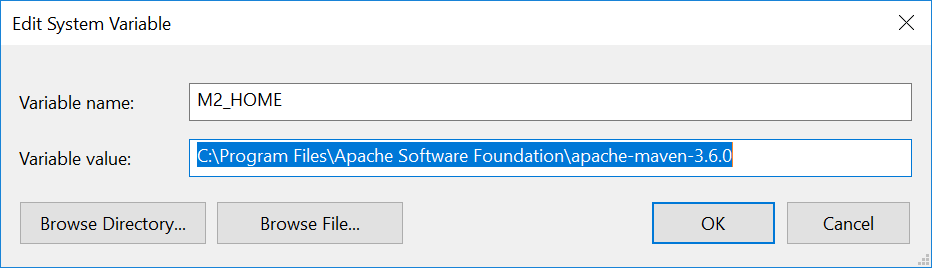
1. JDK and JRE should be present on your machine
2. JAVA\_HOME and path variables for JDK and JRE should be set up in the environment variables



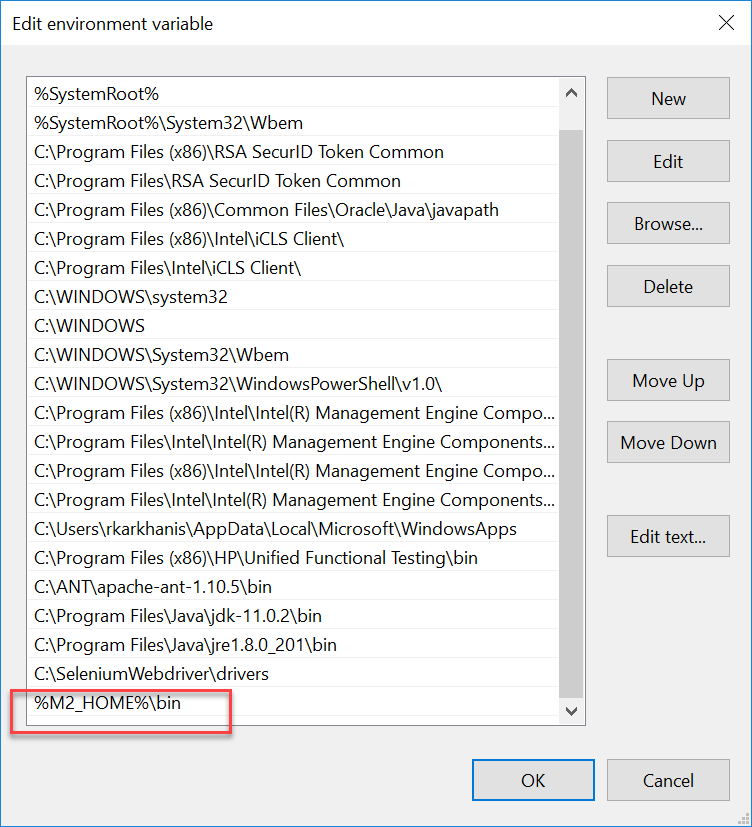
1. Mozilla Firefox should be present. **Please note:** Web Driver supports other browsers too, however for this assignment I have restricted the scope to Firefox only.
2. Apache Maven should be installed on your machine as the project is a Maven Project
3. If Maven is not already installed, ***please follow the below steps to install Maven***
4. Download the following zipped file and extract anywhere on your local machine. I have it on the following path : C:\Program Files\Apache Software Foundation\apache-maven-3.6.0



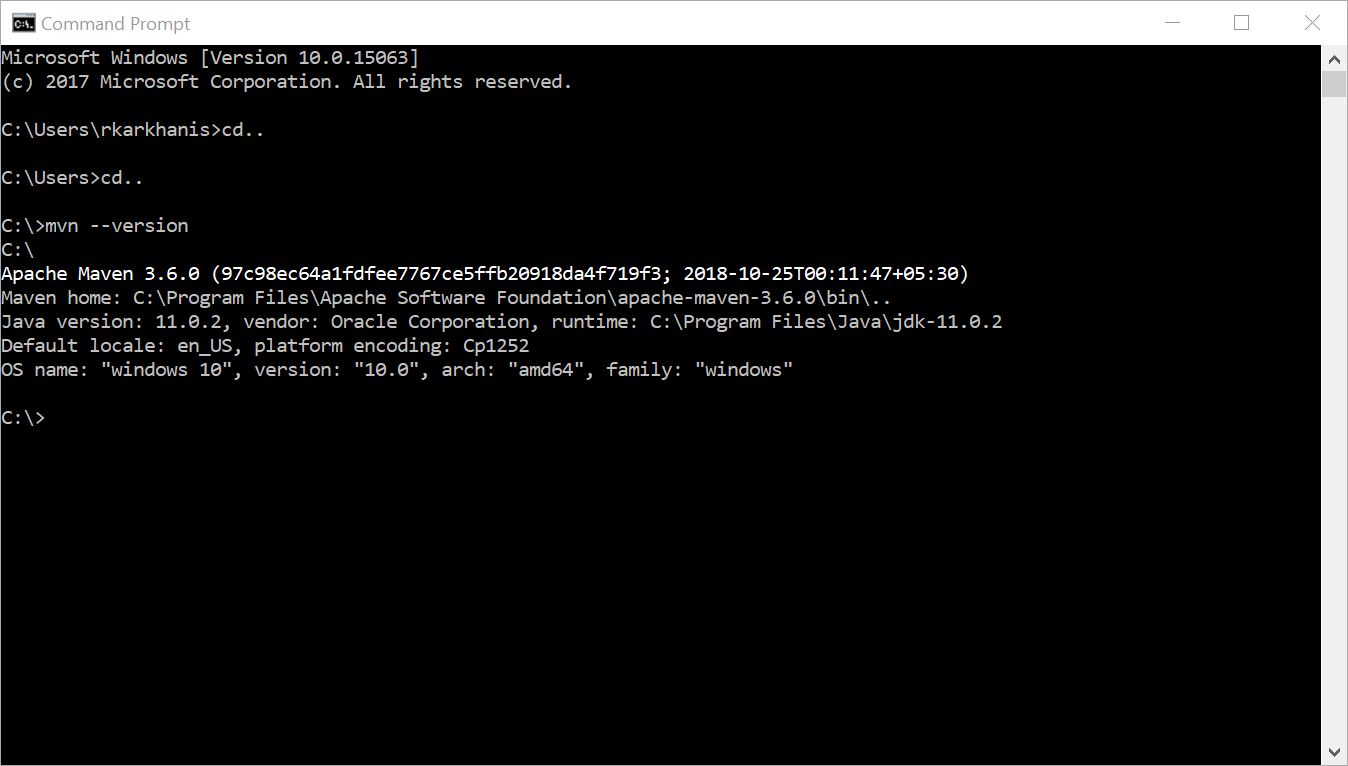
1. Setup M2\_Home variable: For this go to advanced system setting and add a new system variable called M2\_HOME and set the value of the folder in which you have unzipped your Maven. In my case it was the following:



1. The "Edit system variable" dialog opens. Add "<M2\_HOME>\bin" without quotes to Variable value, where <M2\_HOME> is the Maven installation directory.



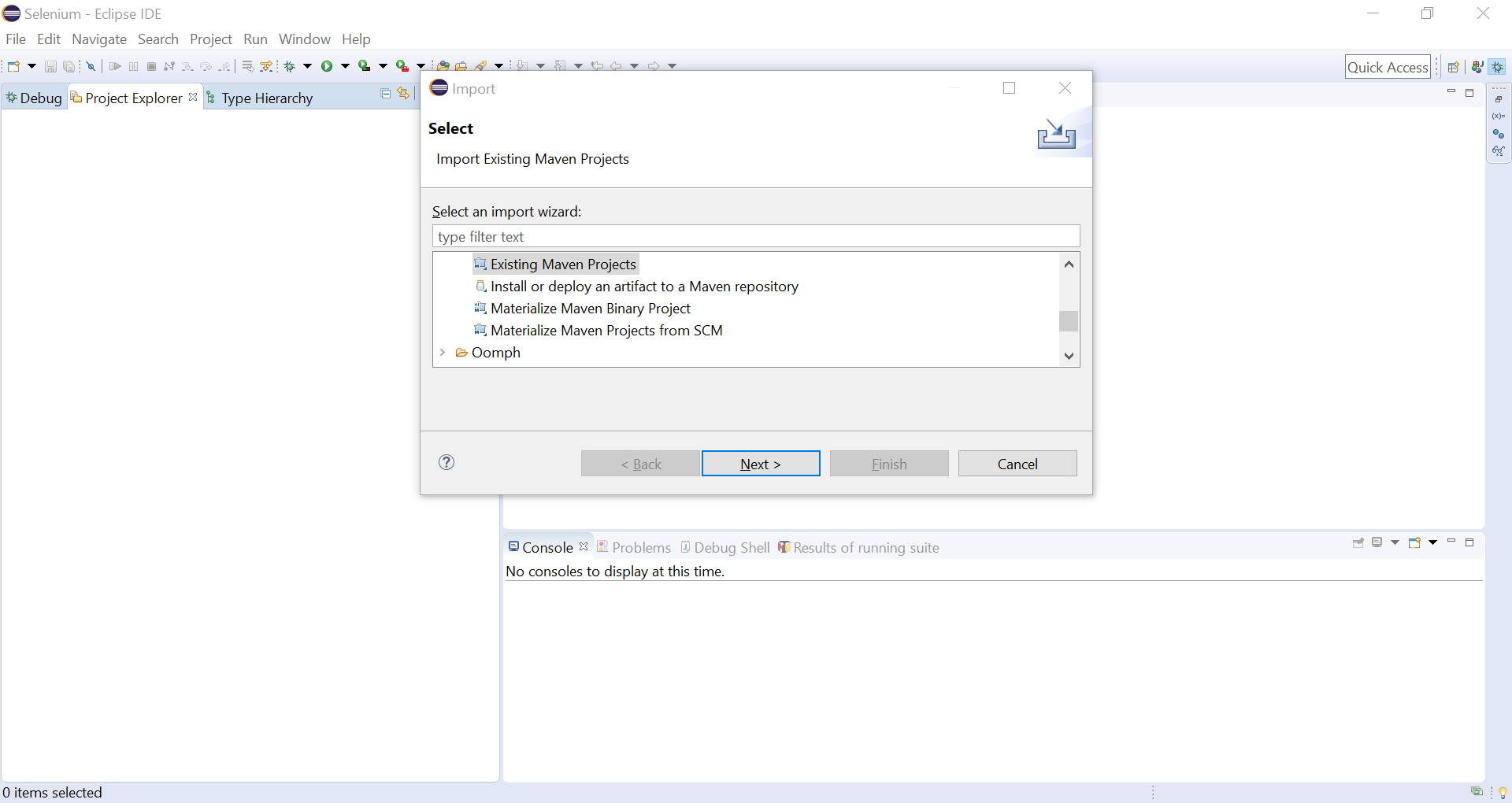
1. Verify installation, in the cmd type the command mvn –version



**Executing the test cases**

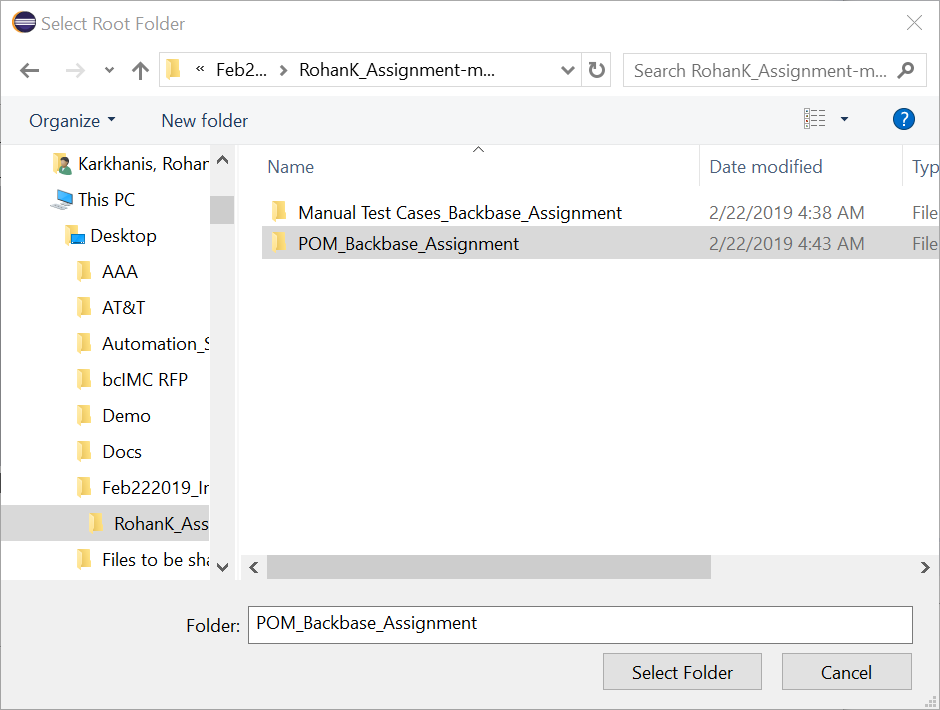
Follow the below mentioned steps to execute the test cases:

1. Download and install any JAVA IDE (Example Eclipse or IntelliJ) . I have used eclipse for my assignment.
2. Open Eclipse and Go to File menu and select Import.

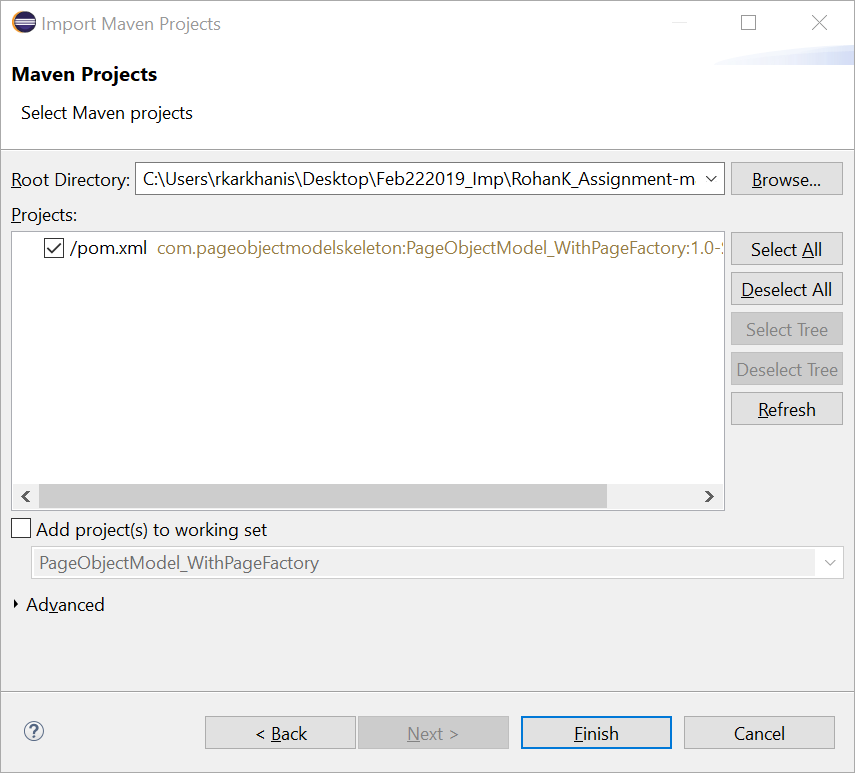


1. Under Maven, select Existing Maven Project. Click on Next to browse to the project. Select the

“**POM\_Backbase\_Assignment**” folder

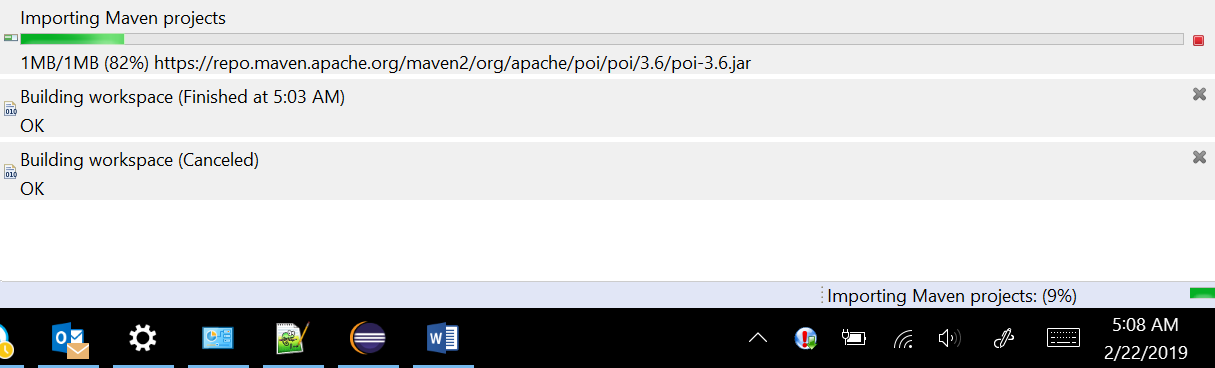


4)On selection of the folder the pom.xml file will be selected as shown below:



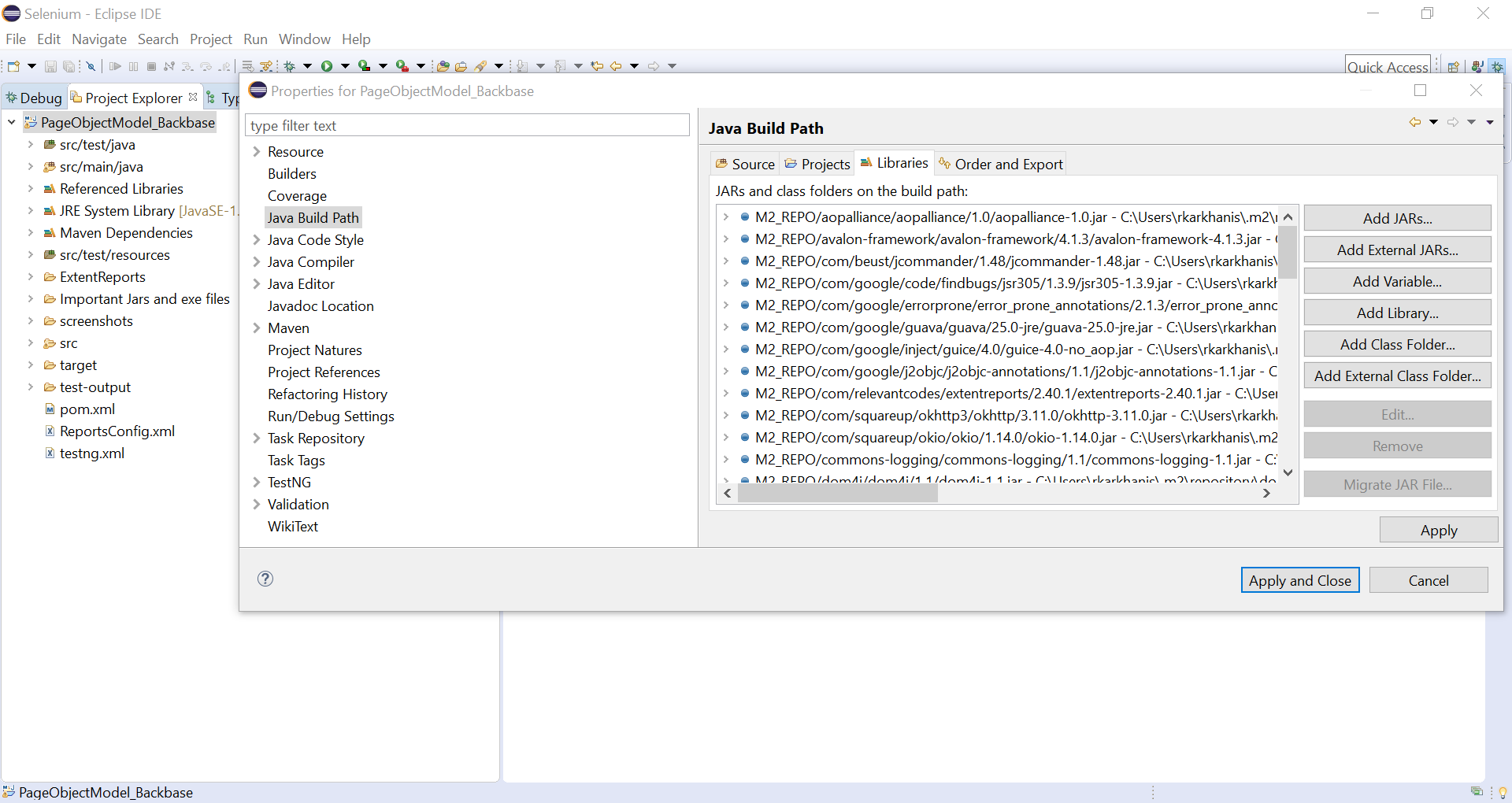
5)Click on Finish. This will import the project to the eclipse and importing status will be show as below.



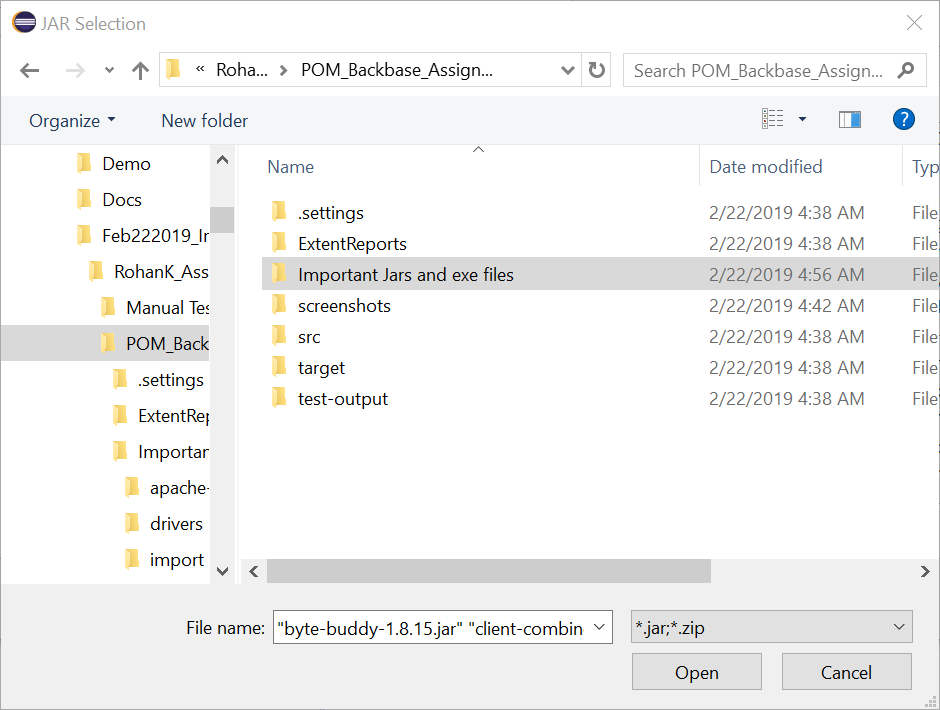


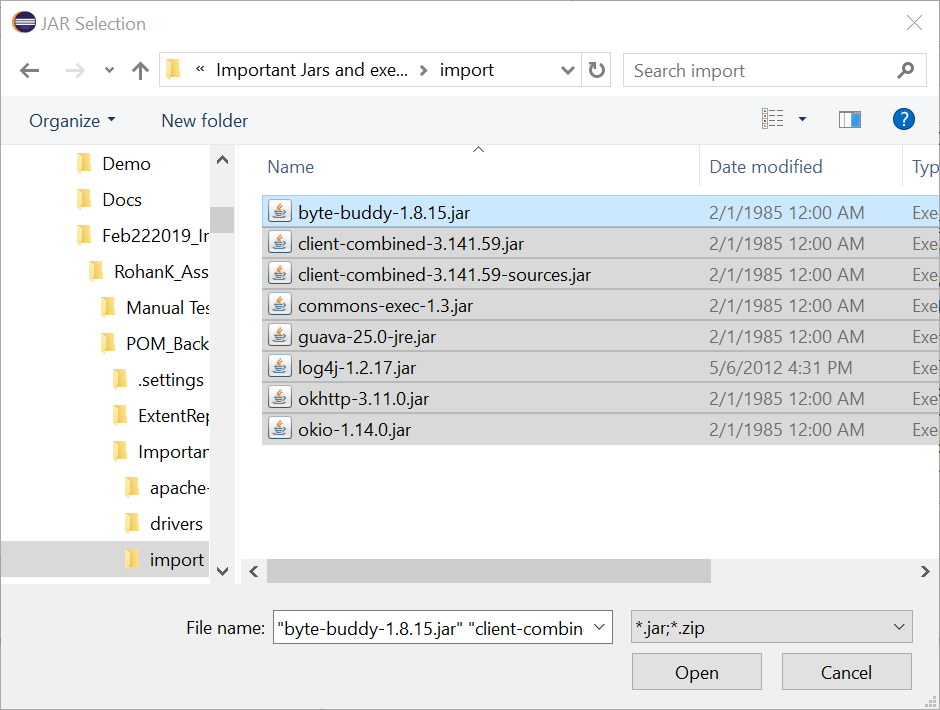
6)Right click on the Project and select Maven🡪Update Project. This will create “.M2” folder in the users folder which will be the local Maven repository. This will contain all the required dependencies.

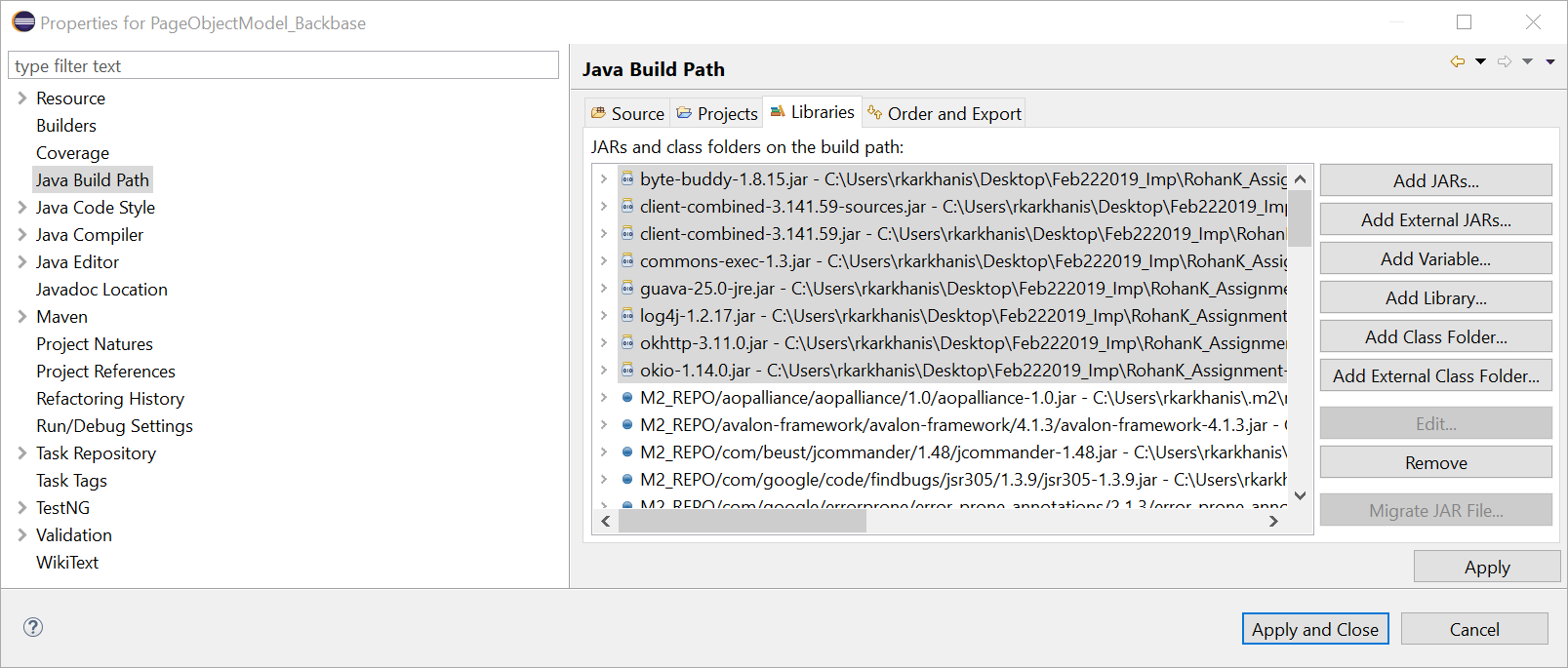
7)Once again Right click on the project and go to Properties. Select Java Build Path and select Add external JARs..



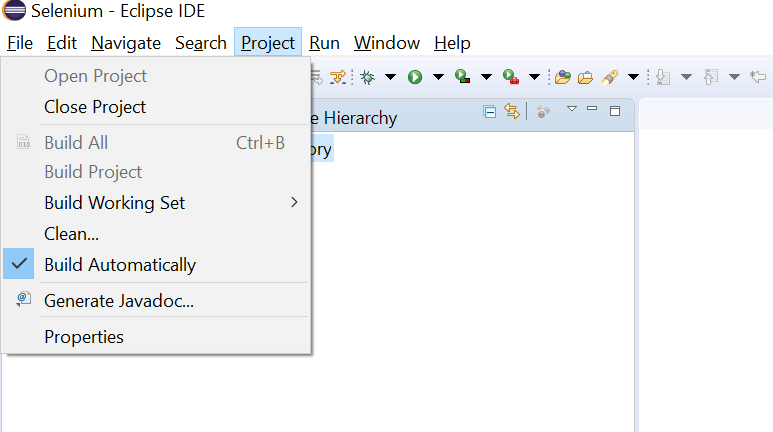
8)There is a folder “Important Jars and Exe files”. Select all the Jar files as shown in the screenshot below to include these in the JAVA build path.







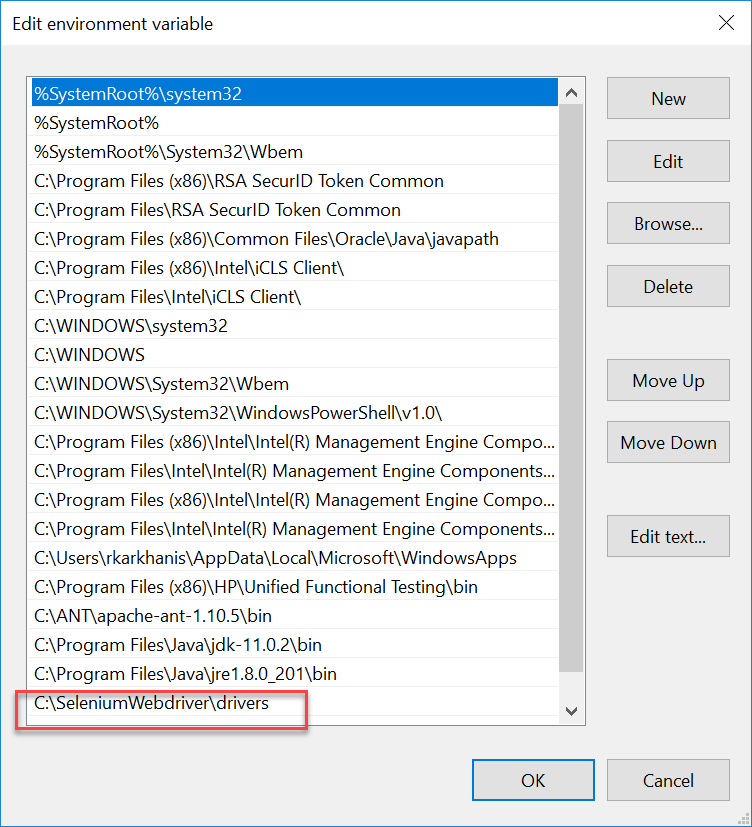
9) From the top Menu bar select Project and ensure that “Build automatically” option is ticked.



10) Since TestNG is also used in my framework, please install the TestNG plugin for eclipse. Go to Help and select “**Install New Software..” .**  In the “work with” text field provide the following URL for TestNG [**http://beust.com/eclipse/**](http://beust.com/eclipse/) **.** Click on next and finish the installation.

11) Once again navigate to Control Panel\System and Security\System 🡪 Advanced system settings🡪Environment variables🡪Path and edit the path to include the geckoDriver.exe path as shown below:

In this case it will be <*Path where the project is cloned>\KarkhanisRohan\_BackbaseAssignment\POM\_Backbase\_Assignment\Important Jars and exe files\drivers*



12) Navigate to the bottom of the project to find “testng.xml” file and right click on the file to select **Run As : TestNG Suite** . This will trigger all the 4 TCs one after the other. Alternatively if you wish to run individual test cases then navigate to the folders src/test/java, go to package “com.pageobjectmodelassignment.testcases” and right click on individual tests to select Run As :TestNG Test.

13) In the project folder navigate to “Extent Reports” folder to find all the execution reports.

I have also uploaded a small video recording of the execution just in case there happens to be any issue in configuring any of the steps mentioned above.

Also, the extent report folder has the latest reports from my execution.