



Maven Setup

Prerequisite: Maven Setup zip file is available with the user. If it is not available then you can download it from <https://maven.apache.org/download.cgi>

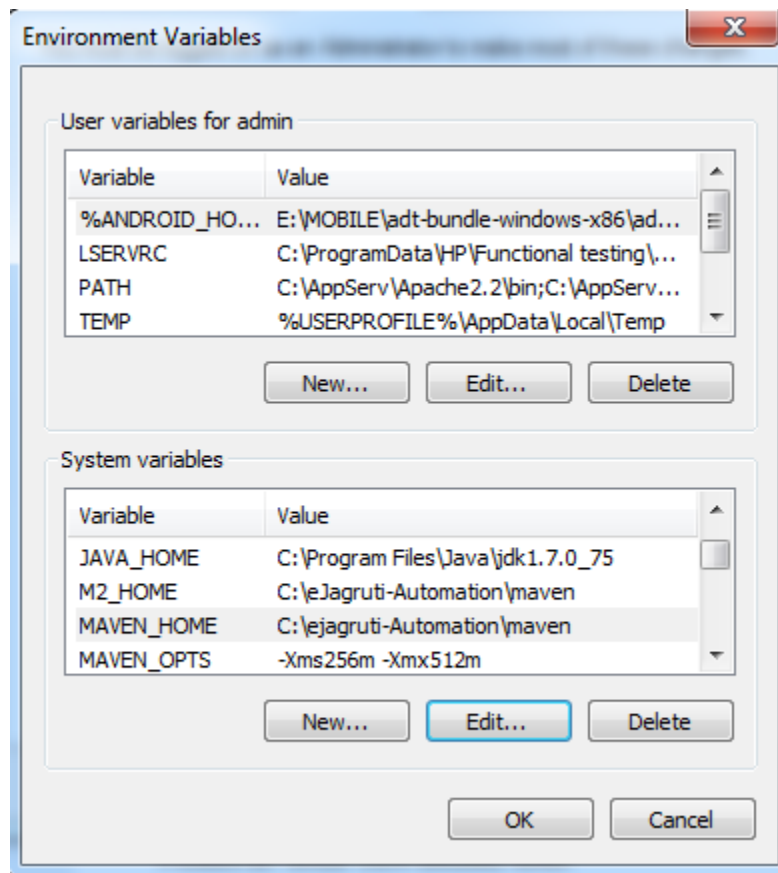
Once the download is over unzip the file and change the name of the unzip folder to maven.

E.g. if the unzip folder name is **apache-maven-3.3.9-bin** then rename it to **maven** it is just for our convenience.

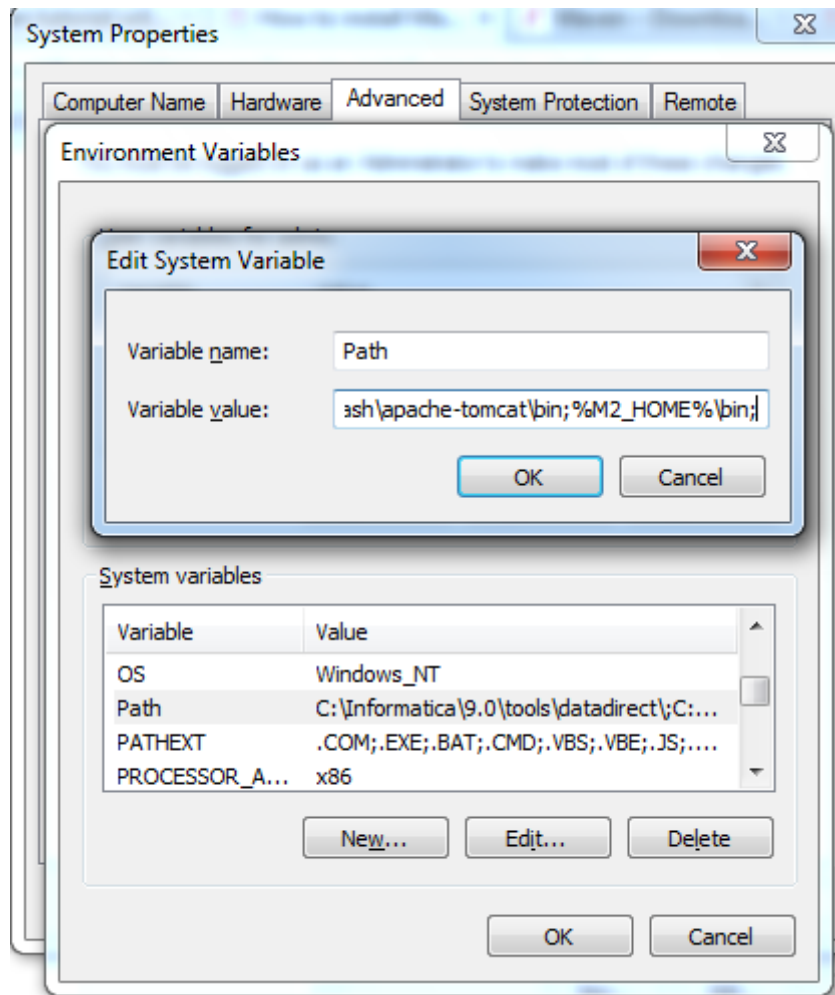
Create a new folder **ejagruti-Automation** in **C:**

Copy given **maven** folder to **C:\ejagruti-Automation**

Make sure **JDK** is installed, and "**JAVA_HOME**", **M2_HOME** and **MAVEN_HOME** variable is added as **Windows environment system variable** if it is not available.



Update System PATH variable, append Maven bin folder – **%M2_HOME%\bin**, so that you can run the Maven's command everywhere.



Do not forget to restart your PC

After restart type given below command in command prompt

mvn -version

If you see the similar output given below then maven installation is successful. **Congrats!!!!**

```
Administrator: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\admin>mvn -version
Apache Maven 3.3.9 (bb52d8502b132ec0a5a3f4c09453c07478323dc5; 2015-11-10T22:11:47+05:30)
Maven home: c:\eJagruti-Automation\maven
Java version: 1.7.0_75, vendor: Oracle Corporation
Java home: C:\Program Files\Java\jdk1.7.0_75\jre
Default locale: en_US, platform encoding: Cp1252
OS name: "windows 7", version: "6.1", arch: "x86", family: "windows"
C:\Users\admin>
```

Now add **maven plugin** into eclipse if it is not available.

To check whether maven plugin is present in the eclipse navigate to **Window>>Preferences>>Maven**. If it is not present then follow given steps

To install Maven plug-in for Eclipse:

Open Eclipse

Go to Help -> Eclipse Marketplace

Search by Maven

Click "Install" button at "Maven Integration for Eclipse" section

Follow the instruction step by step

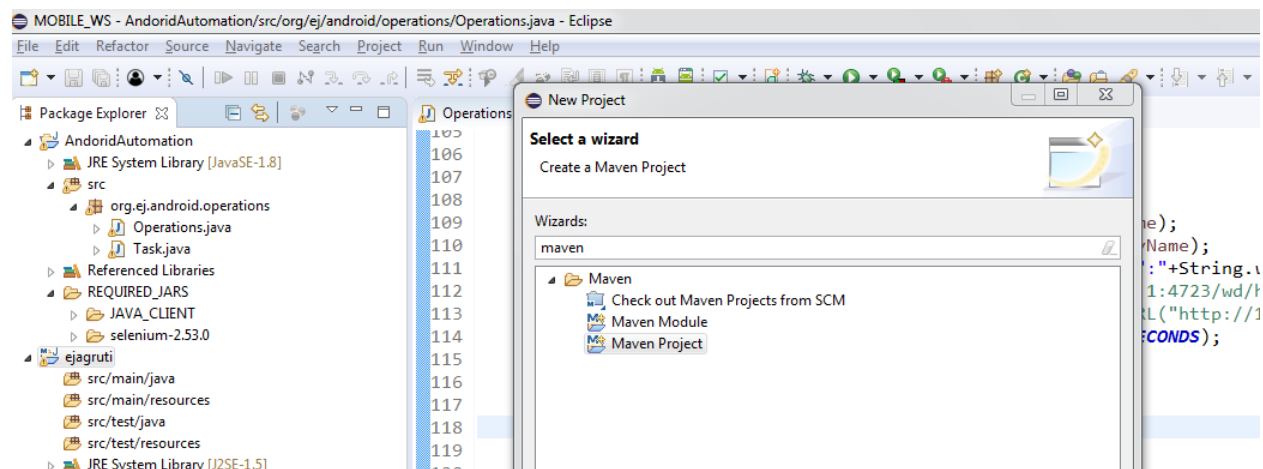
After successful installation do the followings in Eclipse:

Go to Window --> Preferences

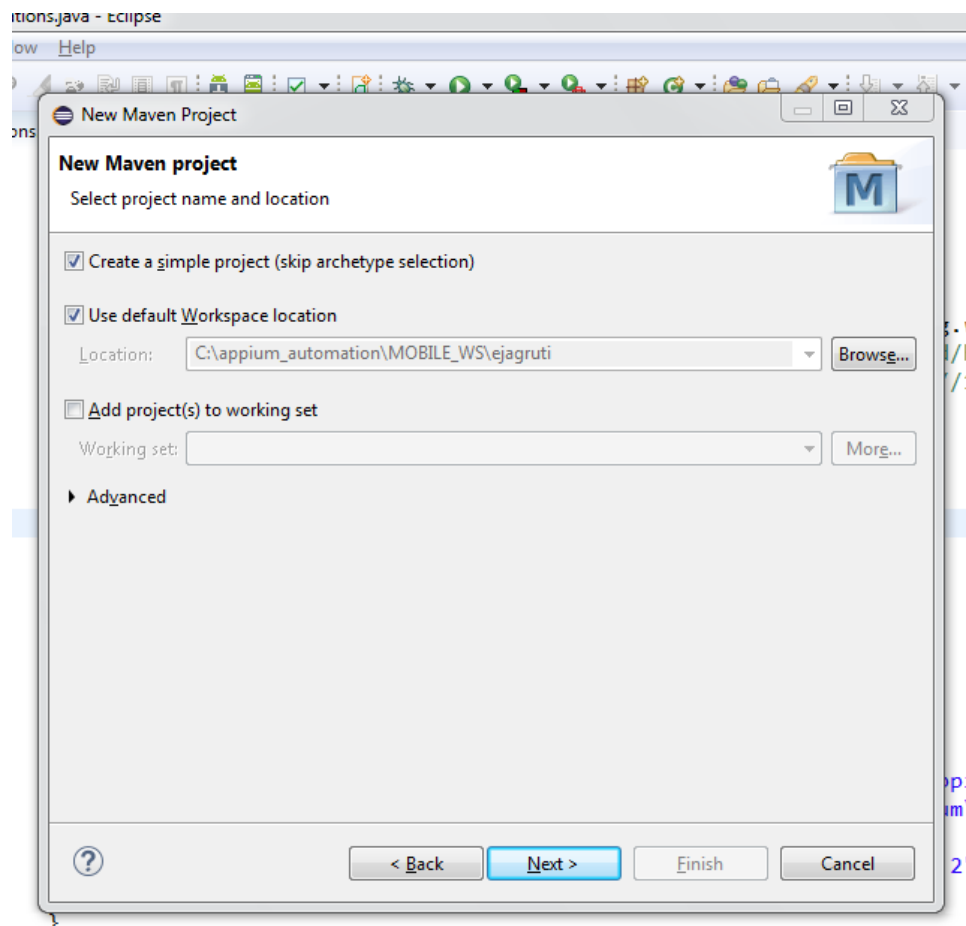
Observe, Maven is enlisted at left panel

-----CREATE MAVEN PROJECT-----

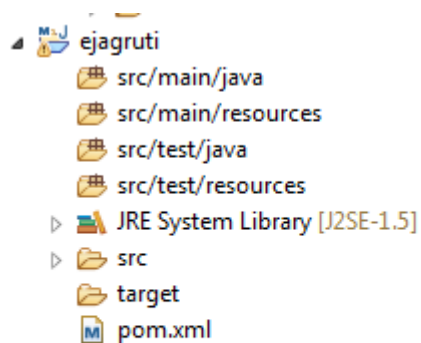
Open eclipse and create new Maven Project. Navigate to File>>New>>Project and in the Wizard search for Maven and click on Maven Project.



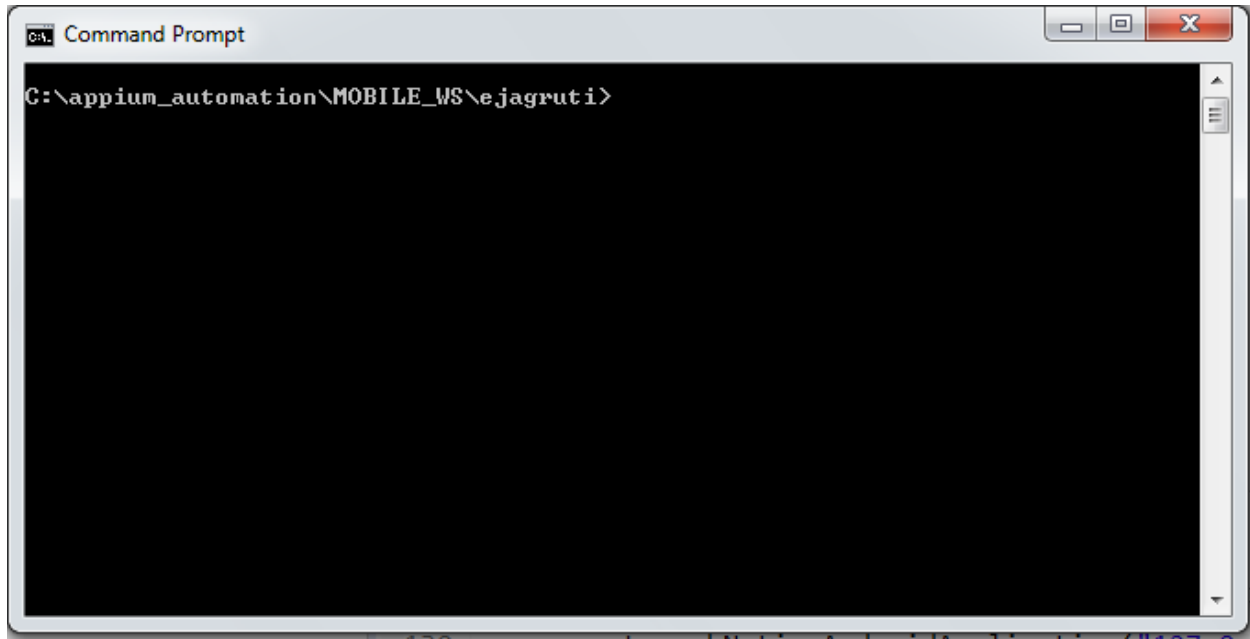
Click on Next Button and check the Option "Create a simple project (skip archetype selection)"



Click again Next Button and in the group id put **org.ej.selenium** and in the artifact id put **ejagruti** and click Finish.

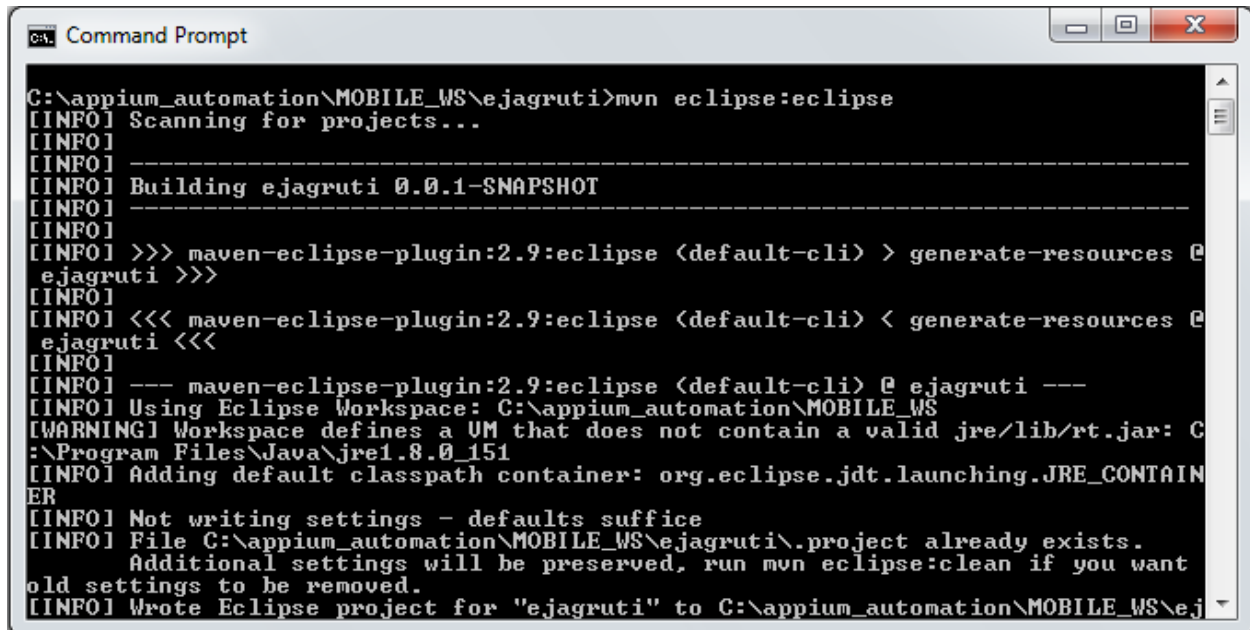


Now open command prompt and go to project folder path which we have recently created. In this example it is ejagruti.

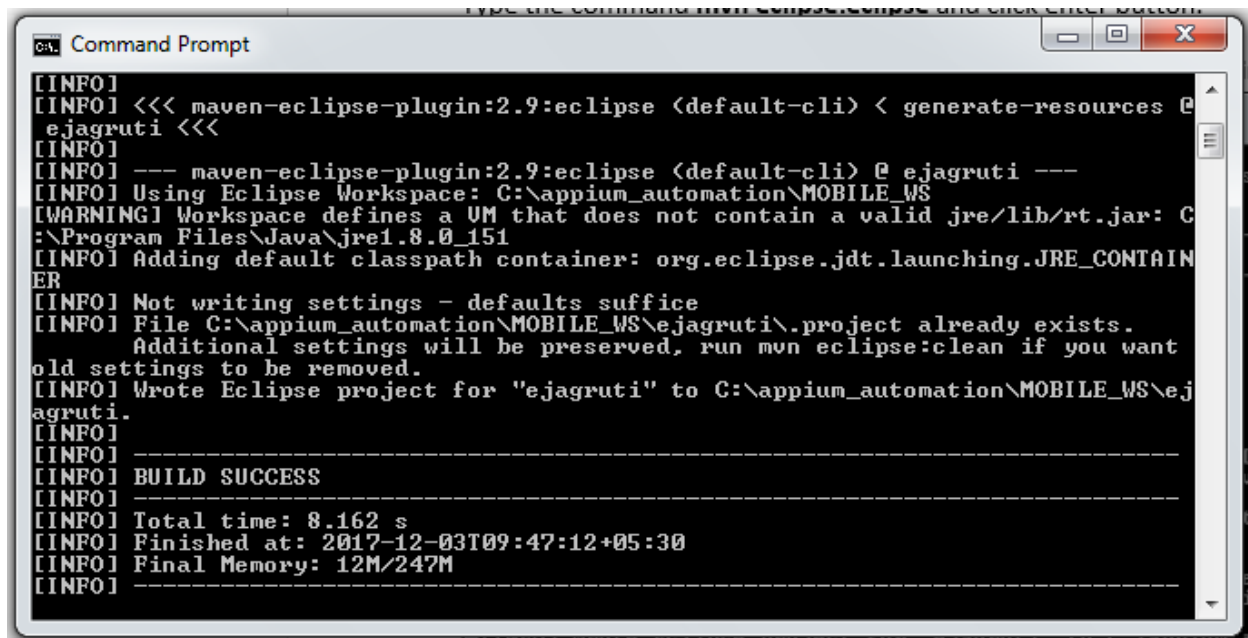


Type the command **mvn eclipse:eclipse** and click enter button.

Maven will scan the projects and will download all the required things from the



Wait till process gets completed. At the end you should see BUILD Successful message given below



```
C:\> Command Prompt

[INFO] <<< maven-eclipse-plugin:2.9:eclipse <default-cli> < generate-resources @
ejagruti <<<
[INFO] --- maven-eclipse-plugin:2.9:eclipse <default-cli> @ ejagruti ---
[INFO] Using Eclipse Workspace: C:\appium_automation\MOBILE_WS
[WARNING] Workspace defines a UM that does not contain a valid jre/lib/rt.jar: C
:\Program Files\Java\jre1.8.0_151
[INFO] Adding default classpath container: org.eclipse.jdt.launching.JRE_CONTAIN
ER
[INFO] Not writing settings - defaults suffice
[INFO] File C:\appium_automation\MOBILE_WS\ejagruti\.project already exists.
Additional settings will be preserved, run mvn eclipse:clean if you want
old settings to be removed.
[INFO] Wrote Eclipse project for "ejagruti" to C:\appium_automation\MOBILE_WS\ej
agruti.
[INFO]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 8.162 s
[INFO] Finished at: 2017-12-03T09:47:12+05:30
[INFO] Final Memory: 12M/247M
[INFO] -----
```

In order to provide necessary dependencies please refer <https://mvnrepository.com/>

Thank you!!!