Cucumber Execution with Maven 

This section will guide you to understand:

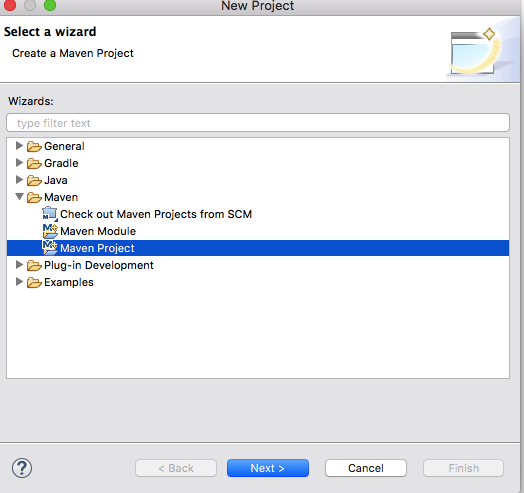
* How to set up Cucumber eclipse plugin
* Maven project setup
* Extent report setup

**Development Environment:**

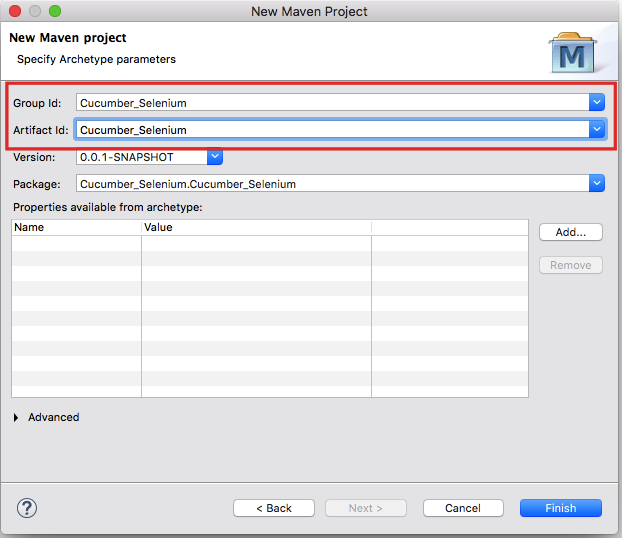
* JRE: OpenJDK Runtime Environment 11.0.2
* Eclipse IDE for Enterprise Java Developers v2019-03 (4.11.0)
* TestNG
* Selenium jars
* Cucumber jars

Executing Cucumber Tests with Maven on Local Machine

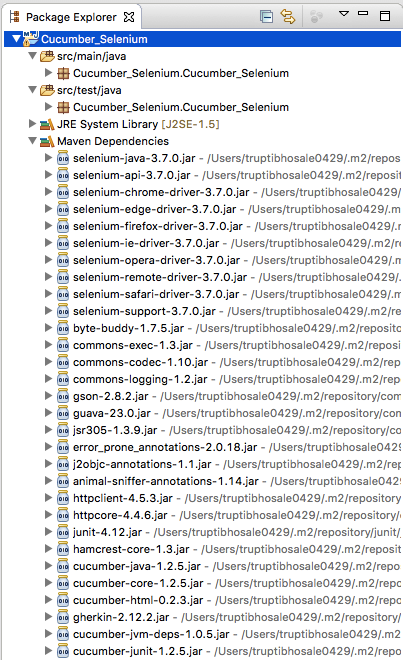
* To create a Maven Project in Eclipse, click on **New** *→* **Project** *→* **In the wizard**, select **Maven Project**.



* On the new Maven Project pop-up, select the checkbox to create your project at the default location OR you can also browse and set a new location of your choice. Click on Next to proceed.
* In the next screen, you will have to mention a Group ID and Artifact ID of your own choice; this is the name of your Maven project. Once you click the Finish button, a Maven project will be created in Eclipse.



* Now, in order to build a Selenium-Cucumber framework, we need to add a dependency for Selenium and Cucumber in pom.xml.
* Copy the dependency tag for the following from the Maven Repository.
  + Selenium Web driver
  + Cucumber-Core jar
  + Cucumber-Java jar
  + Cucumber-TestNG jar
* Make sure to update the project after adding dependencies to pom.xml; you can do that by right clicking *Project → Maven → Update Project*. Once you update the project, you will see that many JAR files are added to the Maven Dependencies folder in your project.



* To proceed with Cucumber implementation, you need to create three packages to store the feature files, step definition code, and test runner code. Let us create three packages: the features, seleniumgluecode, and runner. To create a new package in src/test/java, right Click the folder-> New-> Package.
* Now create the feature file in the Feature package. Right click ->New ->File->Enter name test.feature.
* Create a class test.java to write the gluecode for the features written. Right click seleniumgluecode->New ->Class->enter name as test and save.
* To run the feature files and their respective code, we need to writed a TestNG runner class. Right click **runner** ->**New**-> **Class**->enter name as **testrunner**.