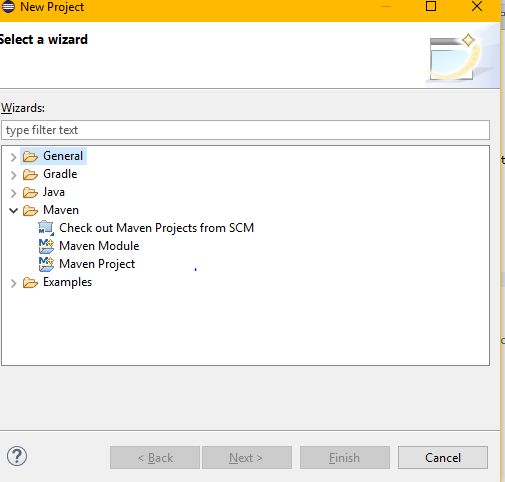
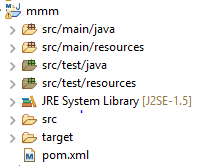
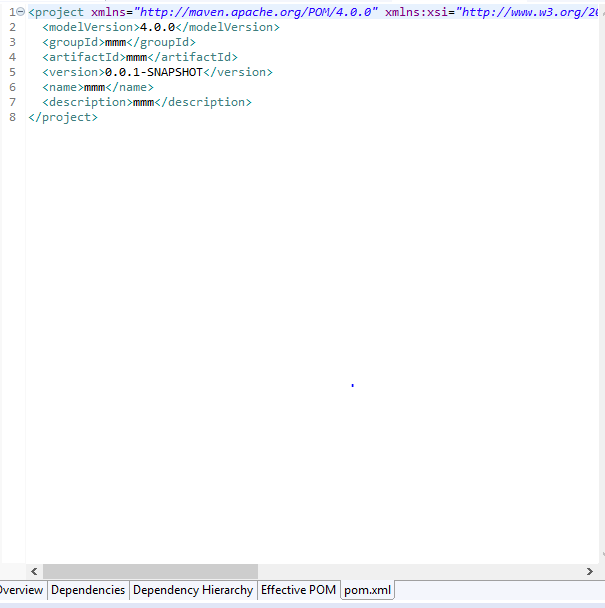
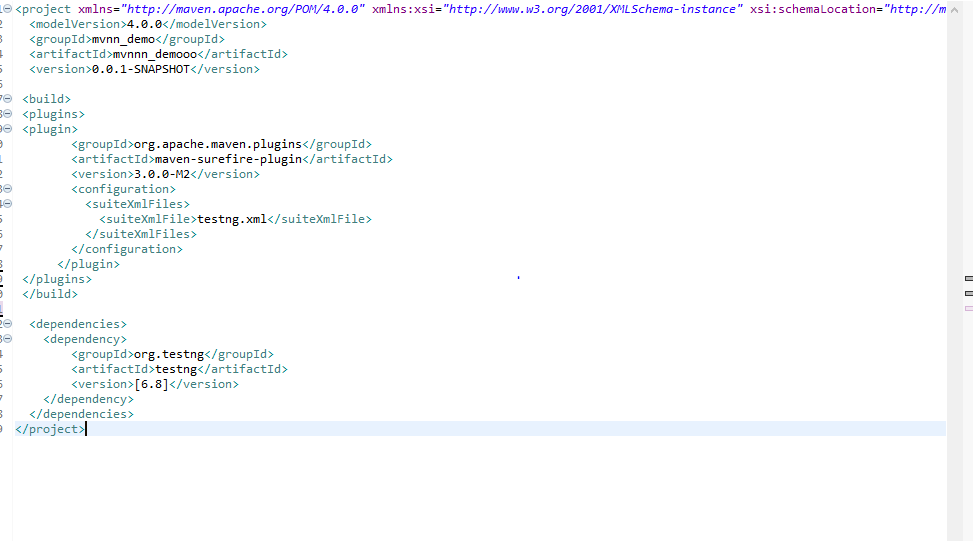
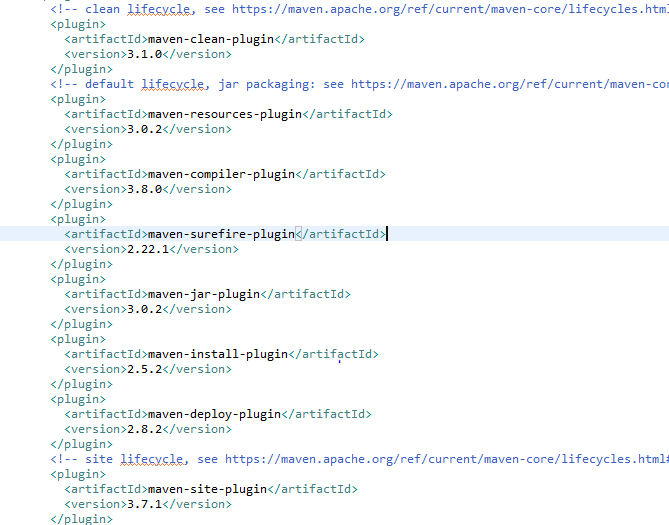
**MAVEN**

* **Maven:**
  + Maven is build management tool or build automation tool.
  + Maven is project management tool that is based on pom(project object model).
  + It simplifies the build process like Ant. It is far advanced than Ant.
  + Maven has its
  + structure.
  + Easy to integrate with continues integration tools like Jenkins.
* **Why maven:**
  + If you are using maven you can easily download jar files with only one dependency.
  + Instead of creating builds manually we can do that process automatically by maven.
* **Setup in eclipse:**
  + Open eclipse click on file then click on new then click on project then go maven section and select maven project.
  + 
  + Follow the steps and click finish. Your maven project will be created.
  + Your project will be created in the below form.
  + 
  + As we see in the above picture src/main/java is for java code.
  + Src/main/resources is to keep resources of java like excel files, json files, pdf files etc…
  + Src/test/java is to keep our test files that may be either from junit or testng.
  + Src/test/resources is to keep resources of test files like excel files, pdf files etc…
  + When you create a maven project by default one pom.xml will be created in the project.
  + Pom stands for project object model.
  + Your pom.xml will look like this.
  + 
* **Maven Life cycle:**
  + There are 3 phases in this as per testing.
    - **Maven Compile:**
      * It means whatever the code is available it will compile that code using **maven-compiler plugin.**
    - **Maven Test:**
      * It means your test cases that is it may be your Junit test cases or Testng test cases. It is done using **Surefile plugin.**
    - **Maven Resources or Generate jar files:**

Once your compile and and exectution is done now if u want to create the jar or war files is is done by **maven resources plugin**.

**Running Maven from eclipse:**

* Go to eclipse open your pom.xml and add surefire plugin to pom.xml.
* If you want to create a jar file of your project you add resources plugin.
* Execute your pom.xml.
* 
* 
* Above are the plugins of maven.
* When you execute your pom.xml it will run as per maven life cyle first it will compile your code(using compiler plugin) then it will run your test cases(using sure fire plugin) and it will generate resources(using resources plugin) that is it will create jar files of your project.

TROUBLESHOOTING

1)What to do if you are getting “No compiler is found in this environment are sure you are using JDK instead JRE”(Exception MojoException)

And:

<build>

...

<plugins>

...

<plugin>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.1</version>

<configuration>

<fork>true</fork>

<executable>C:\Program Files\Java\jdk1.7.0\_45\bin\javac.exe</executable>

</configuration>

</plugin>

</plugins>

</build>

OR

In eclipse goto Window>>Preferences>>Click on Installed JRE’s>> check if it is pointing to JRE or JDK