Dependency Management using Maven HONEY ARORA

Trainee

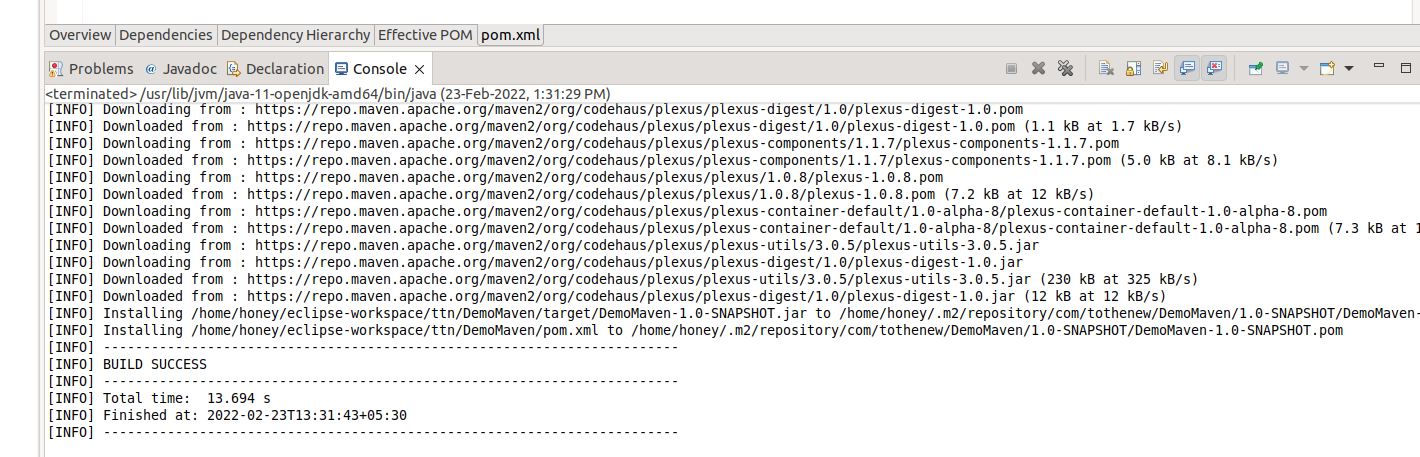
**QUES1**. **Add a maven dependency and its related repository URL.**

****

**QUES2. Add a new repository in the pom.xml and use its dependencies.**

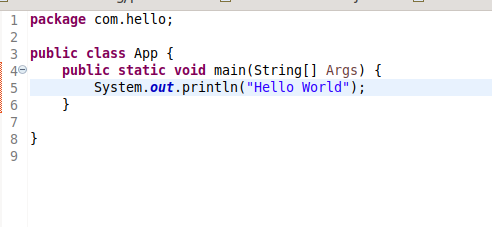
****

****

****

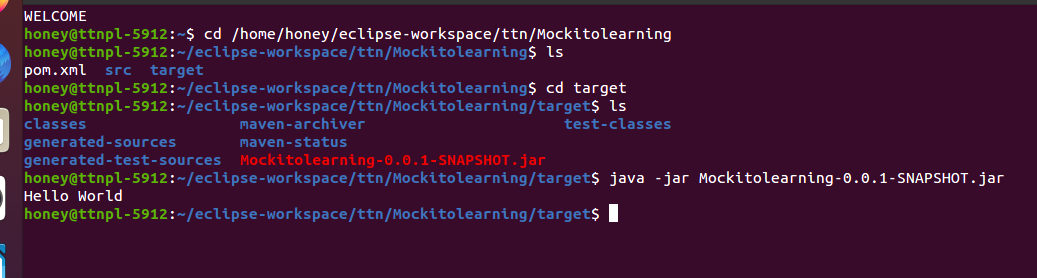
**QUES3. Using the JAR plugin, make changes in the pom.xml to make the jar executable. Using java -jar JAR\_NAME, the output should be printed as "Hello World"**

**App.java**

****

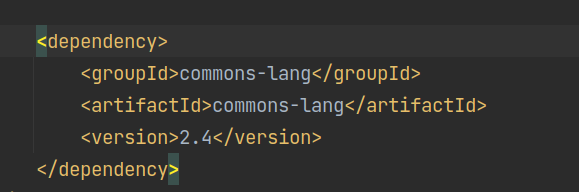
**Pom.xml**

****

****

**QUES4. Differentiate between the different dependency scopes: compile, runtime, test, provided using different dependencies being defined in your pom.xml.**

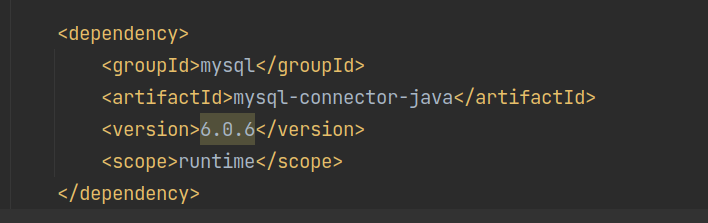
Compile: It is the default scope. It is needed to build,test and run the project.

****

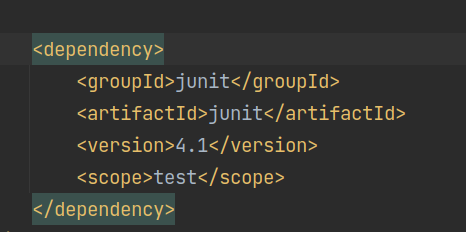
Runtime: They are not needed to build, but are part of the classpath to test and

run the project. Dependencies marked with this scope will be present in the

runtime and test classpath, but they will be missing from the compile classpath

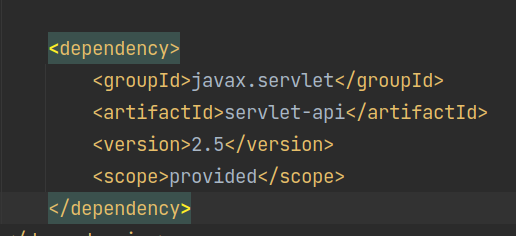
****

Test: They are needed to compile and run the unit tests.

****

Provided: It is used for only build and testing the project. The dependencies are

provided during runtime.

****

**QUES5. Create a multi-module project. Run the package command at the top level to make a jar of every module.**

Let’s create a Maven Project : Mavendemo

Now it will have its pom.xml

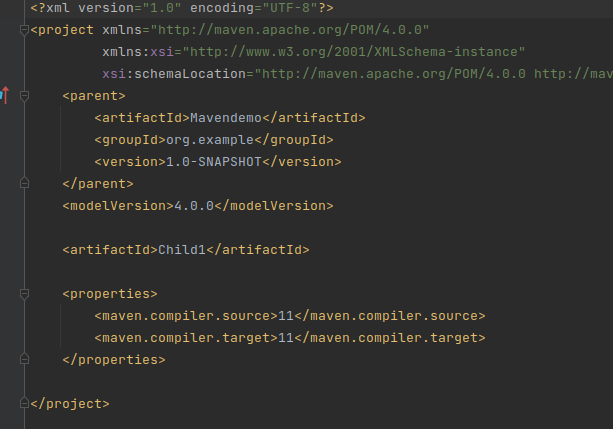
So, if we create multiple modules in this project we’ll use the maven module.

The pom.xml of Mavendemo will be the parent pom for all the modules.

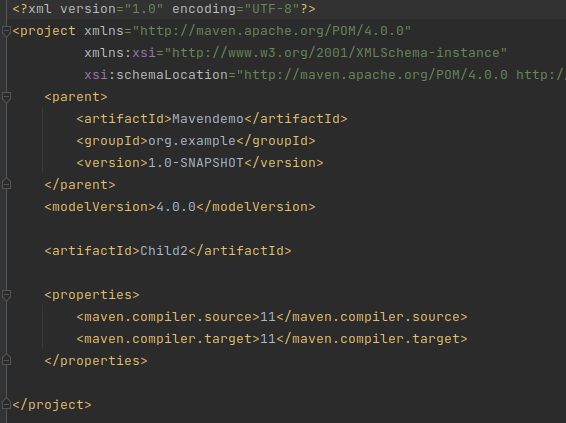
Let’s say we add child1 module and child2 module in the project.

In child1 module the pom.xml will have reference to parent module

pom.xml(Child1)

****

pom.xml(Child2)



The pom.xml of parent module will have child1 and child2 module inside

<modules></modules> section.

pom.xml(Mavendemo)



Now, through terminal

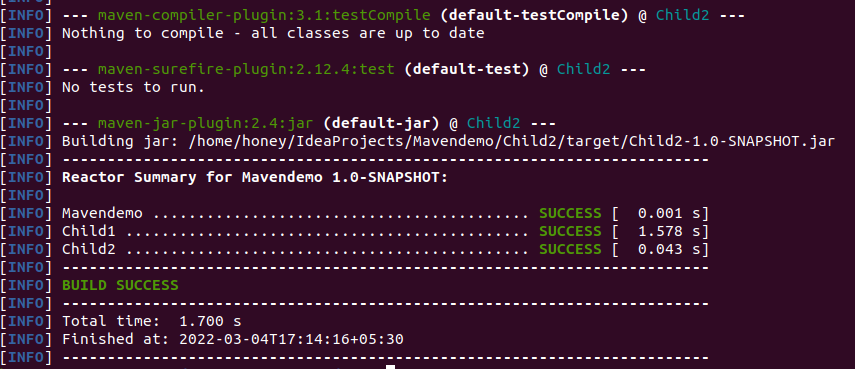
We’ll perform the commands

1.mvn clean

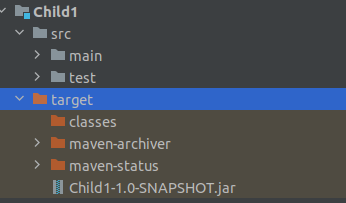
2.mvn compile

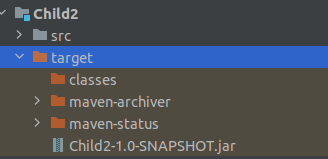
3.mvn package

4.mvn install



Jar of every module will be created. The directories showing the jar of child1 and child2 module are as follows.

****

****