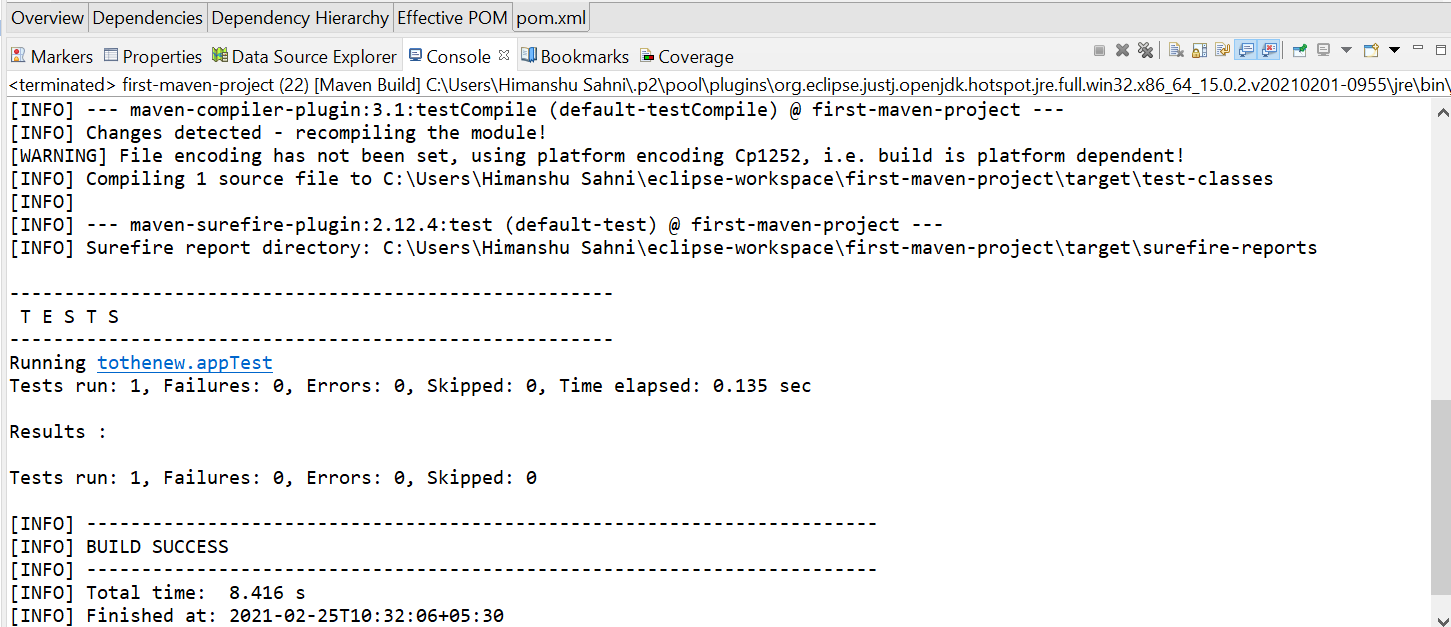
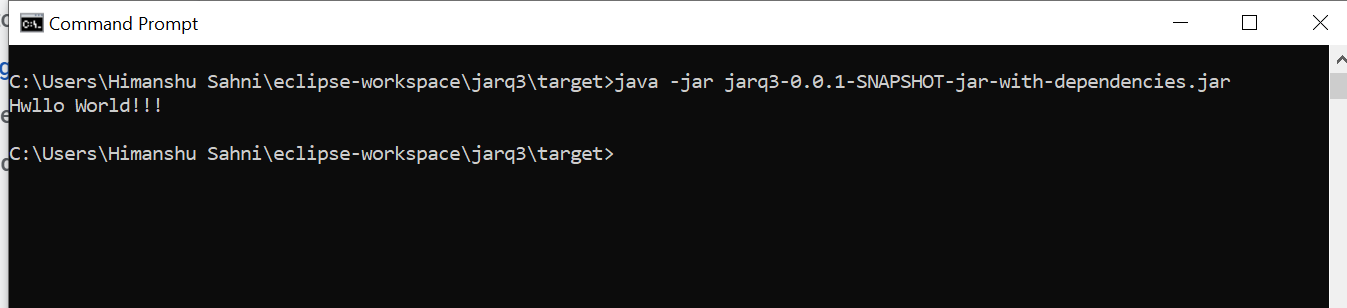
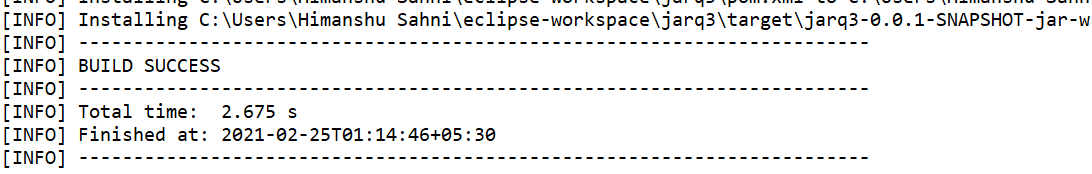
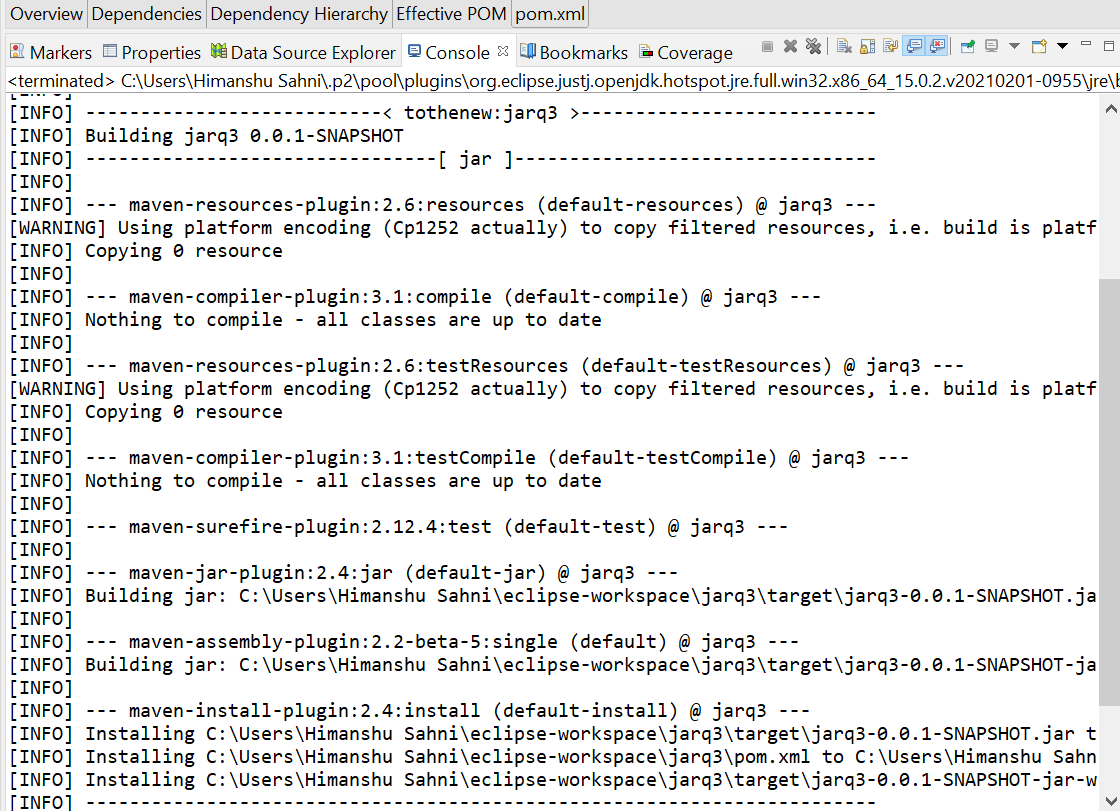
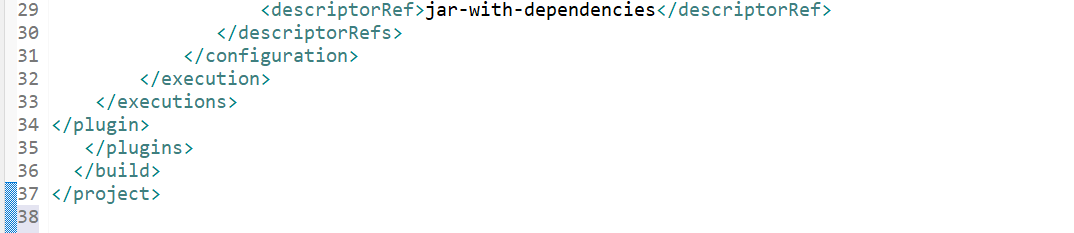
1. **Add a maven dependency and its related repository URL.**
2. **Add a new repository in the pom.xml and use its dependencies.**
3. **Using the JAR plugin, make changes in the pom.xml to make the jar executable. Using java -jar JAR\_NAME, the output should be printed as "Hello World"**

****

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

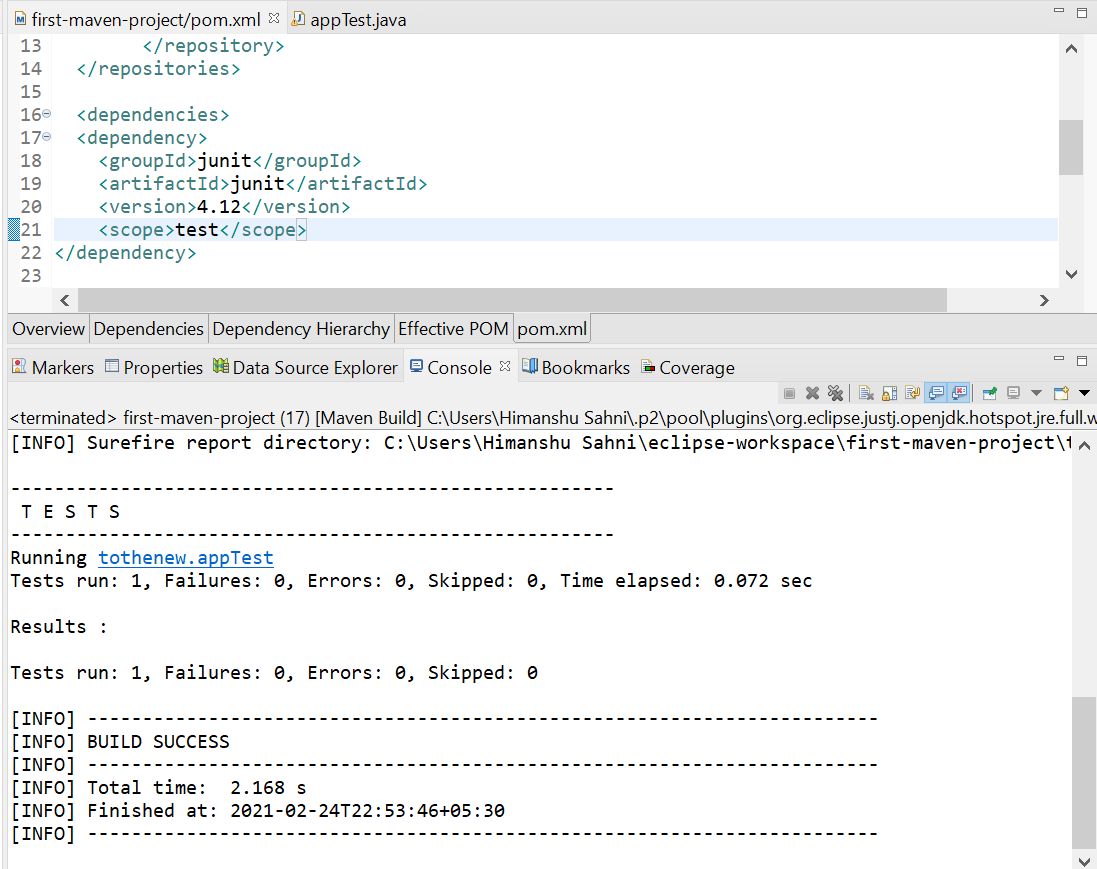
**COMPILE-**

This is maven default scope. Dependencies with compile scope are needed to *build*, *test*, and *run* the project.Scope compile is to be required in most of the cases to resolve the import statements into your java classes source code.

#### 

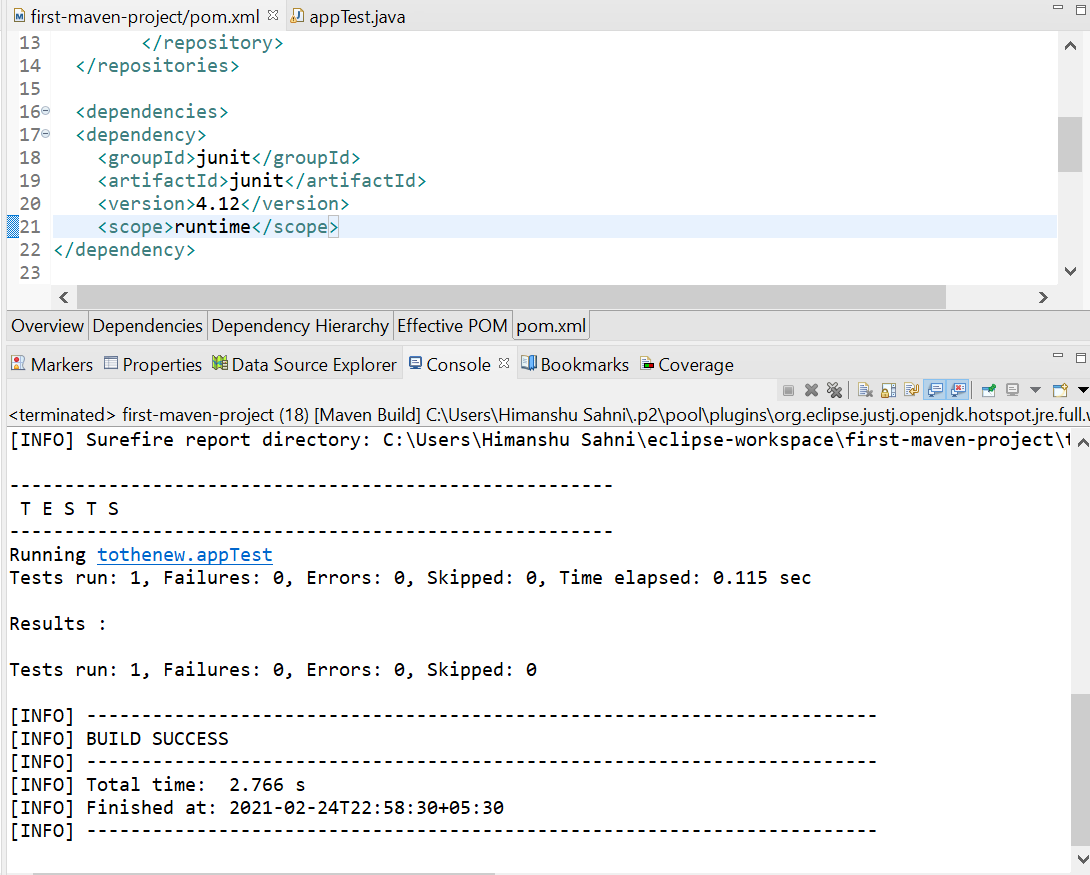
**TEST-**

Dependencies with maven dependency scope tests are not needed to build and run the project. They are needed to compile and run the unit tests.



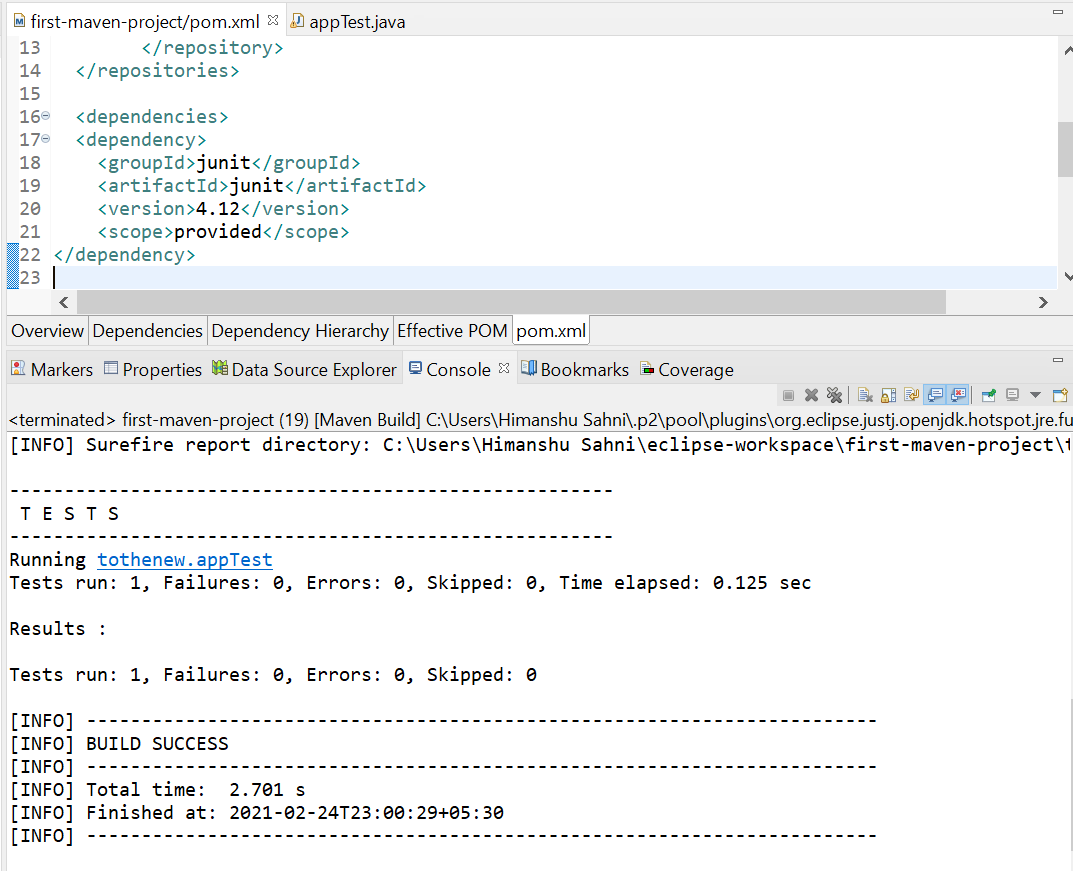
**RUNTIME-**

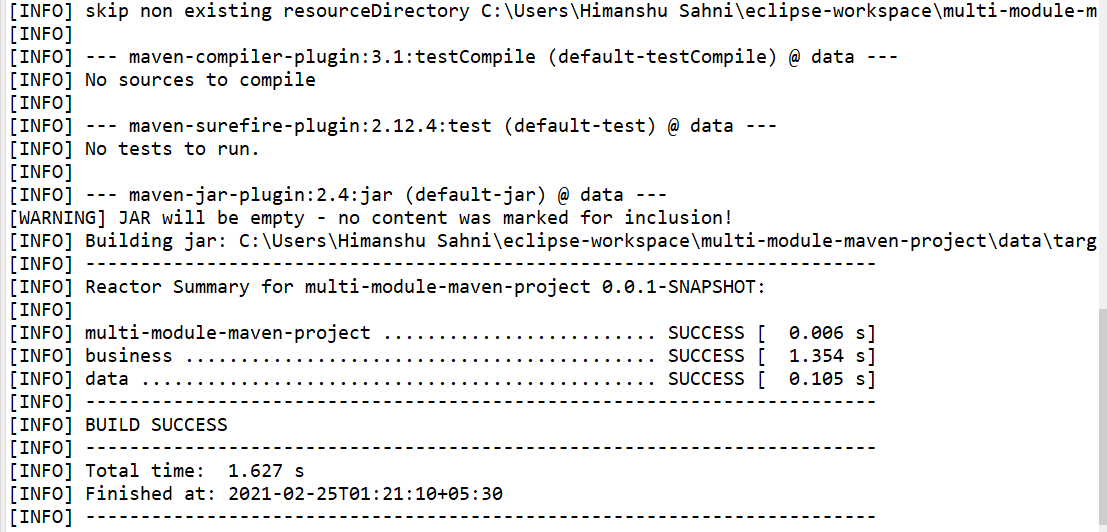
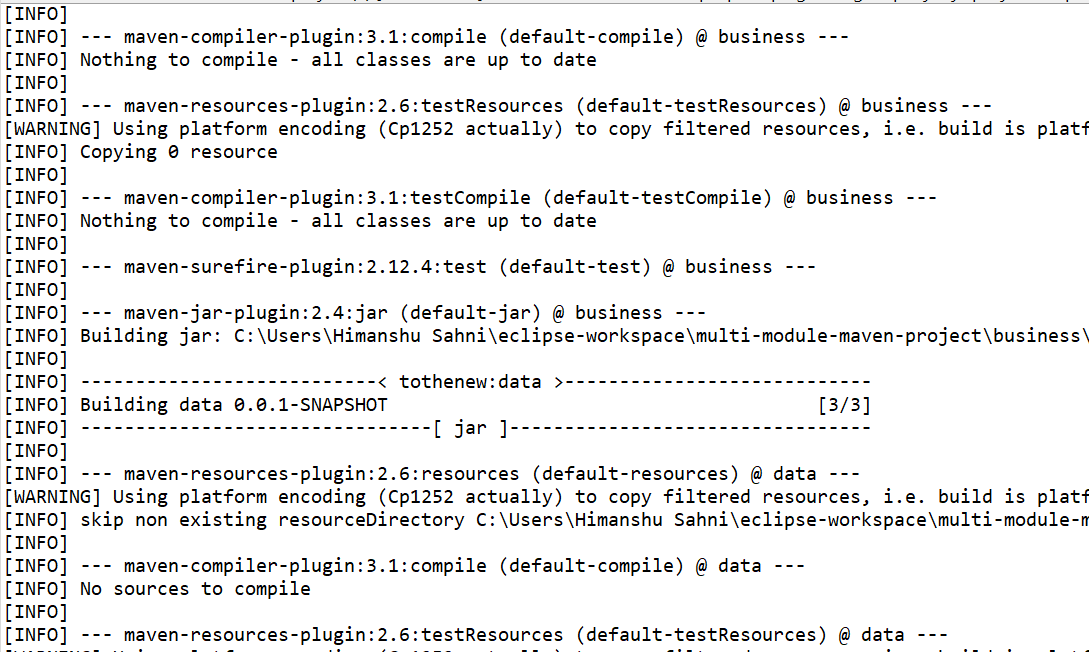
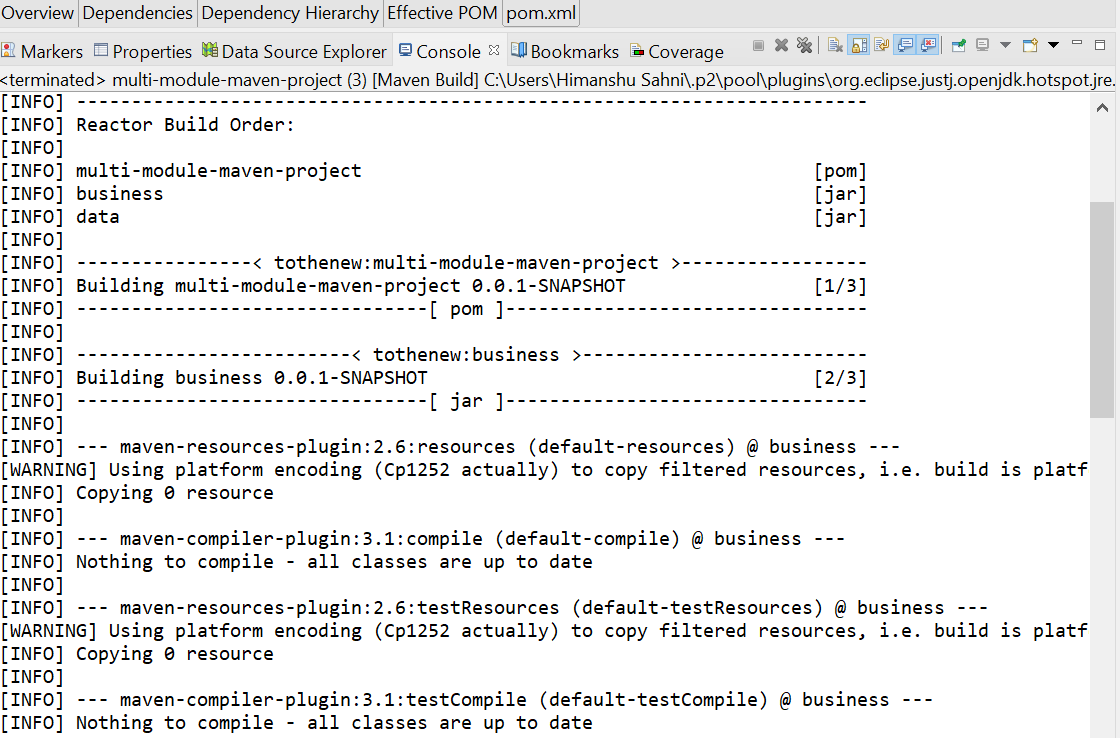
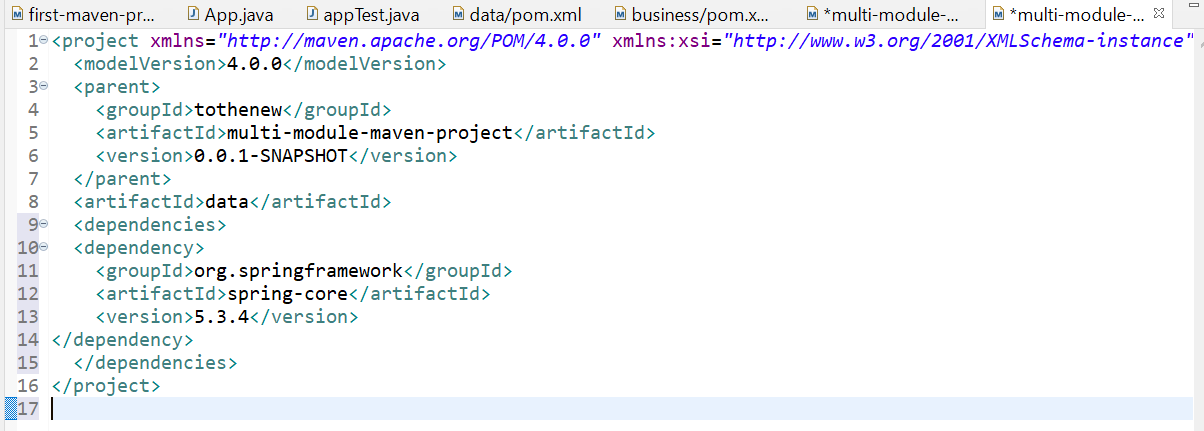
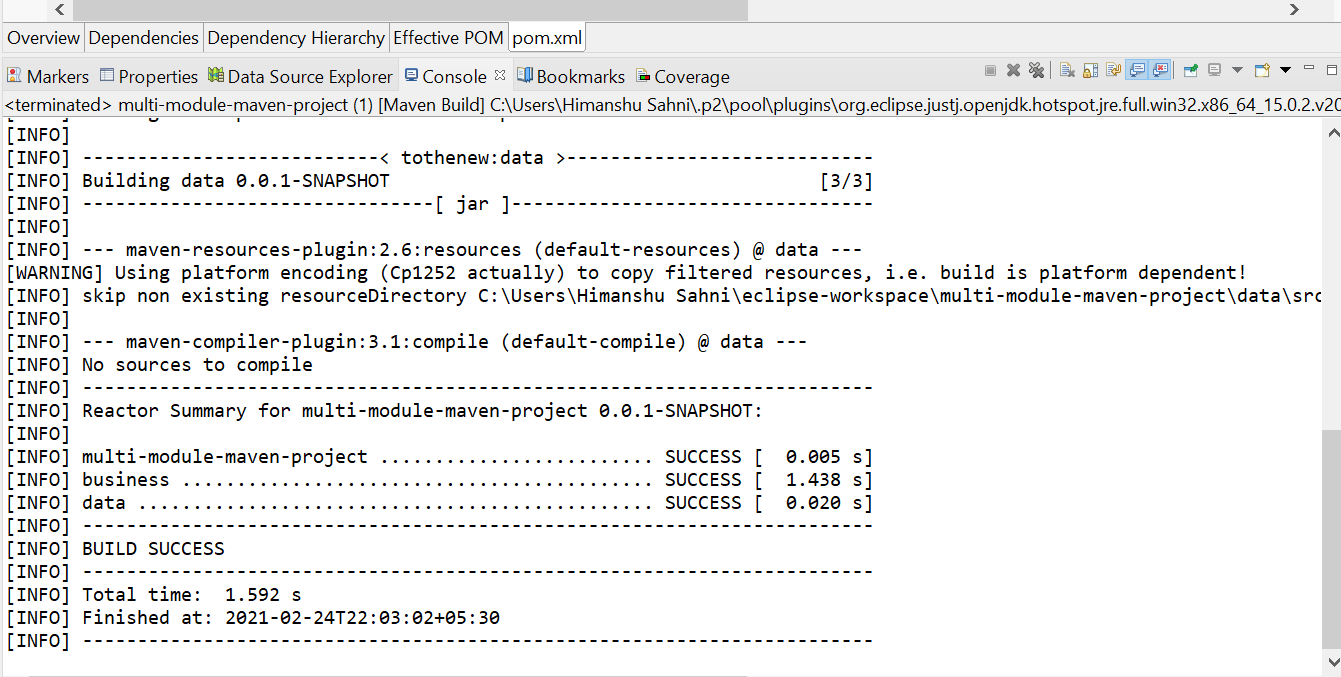
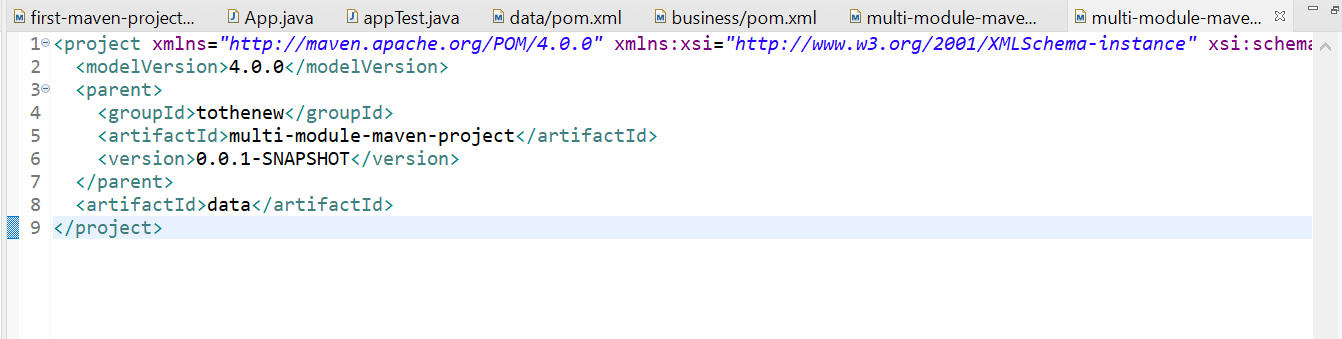
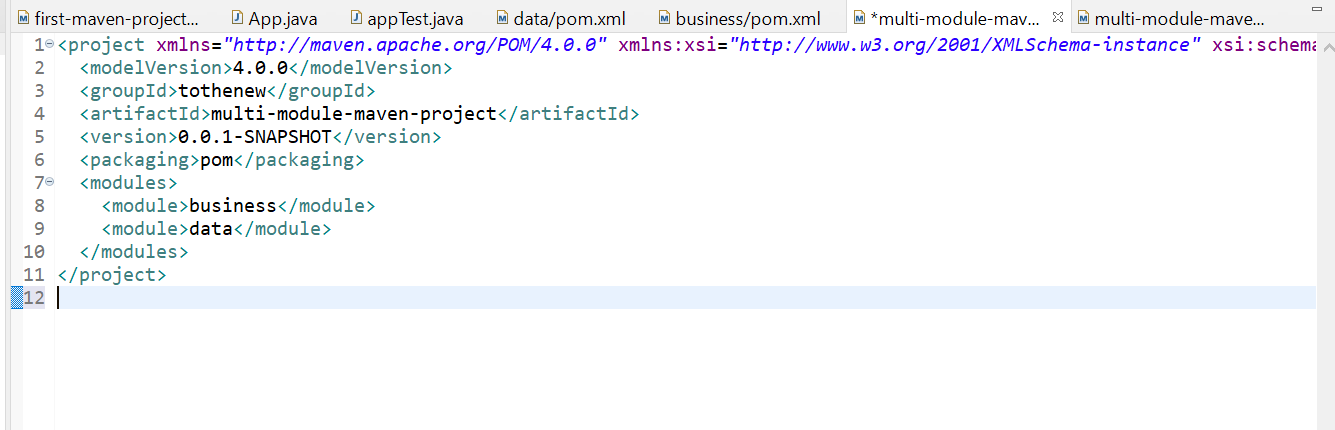
Dependencies with maven dependency scope runtime are not needed to build, but are part of the classpath to test and run the project.



**PROVIDED-**

Maven dependency scope provided is used during build and test of the project. They are also required to run, but should not be exported, because the dependency will be provided by the runtime, for instance, by servlet container or application server.



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