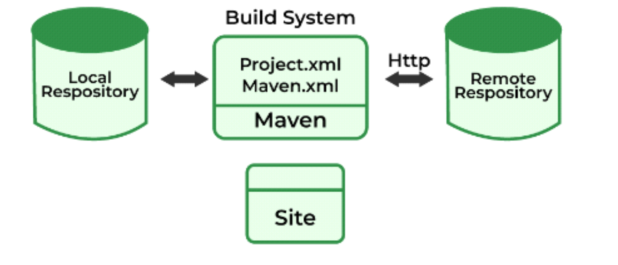
**MAVEN**

* Maven is a build automation tool developed using the Java programming language.
* It is primarily used for Java-based projects to manage the build process, including source code compilation, testing, packaging, and more.
* Maven utilizes the Project Object Model (POM), where the pom.xml file describes the project’s configuration and dependency management.

Maven is a build tool takes care of everything for building a process as follows :

* Generates source code (if auto-generated code is used)
* Generates documentation from source code
* Compiles source code
* Packages compiled code into JAR of ZIP file
* Installs the packaged code in local repository, server repository, or central repository

**Maven Architecture :**



**POM :**

A Project Object Model or POM is the fundamental unit of work in Maven. It is an XML file that contains information about the project and configuration details used by Maven to build the project. It contains default values for most projects. Examples for this is the build directory, which is target; the source directory, which is src/main/java; the test source directory, which is src/test/java; and so on. When executing a task or goal, Maven looks for the POM in the current directory. It reads the POM, gets the needed configuration information, then executes the goal.

Some of the configuration that can be specified in the POM are the project dependencies, the plugins or goals that can be executed, the build profiles, and so on. Other information such as the project version, description, developers, mailing lists and such can also be specified.

The minimum requirement for a POM are the following:

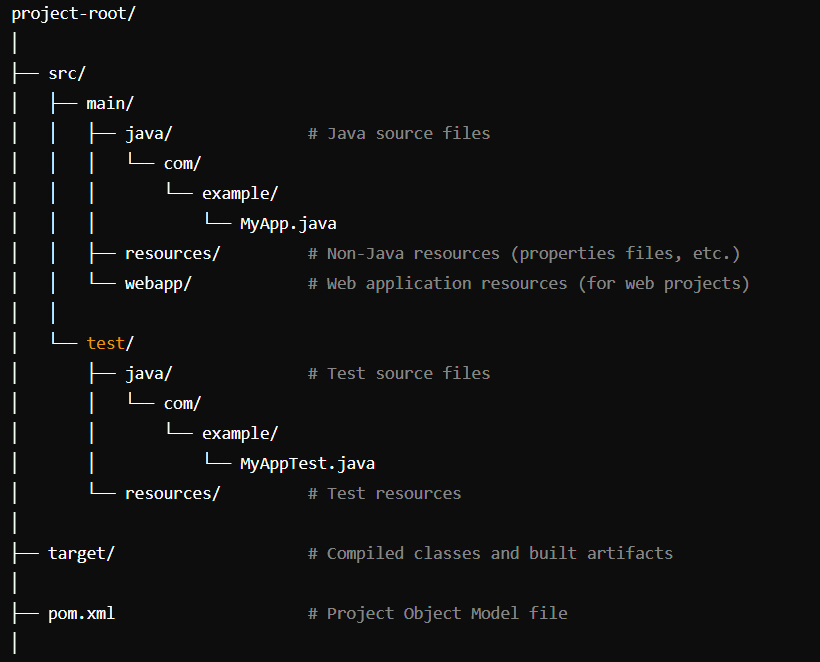
* project root
* modelVersion - should be set to 3.0.0
* groupId - the id of the project's group.
* artifactId - the id of the artifact (project)
* version - the version of the artifact under the specified group

example:

1. <project>
2. <modelVersion>3.0.0</modelVersion>
4. <groupId>com.Neoteric.javademo</groupId>
5. <artifactId>mavenjavademo</artifactId>
6. <version>1.0.0</version>
7. </project>

A POM requires that its groupId, artifactId, and version be configured. These three values form the project's fully qualified artifact name. This is in the form of <groupId>:<artifactId>:<version>. As for the example above, its fully qualified artifact name is " com.Neoteric.javademo: mavenjavademo :1.0.0".

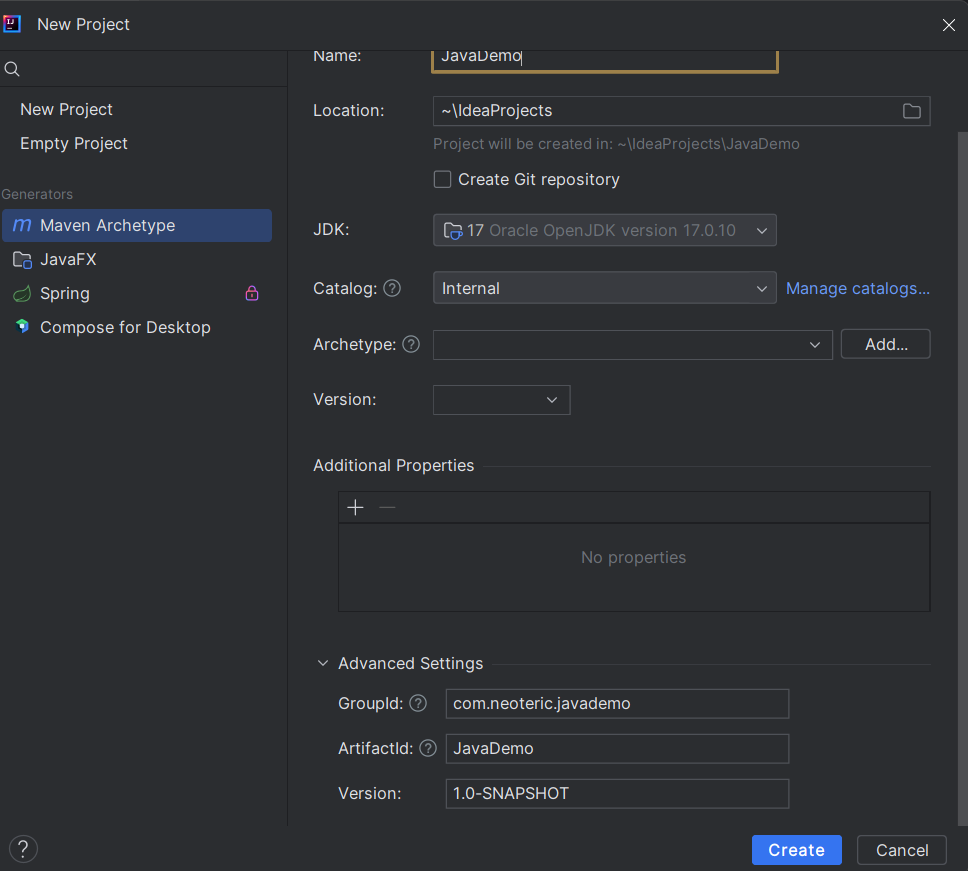
**Maven Project Structure:**



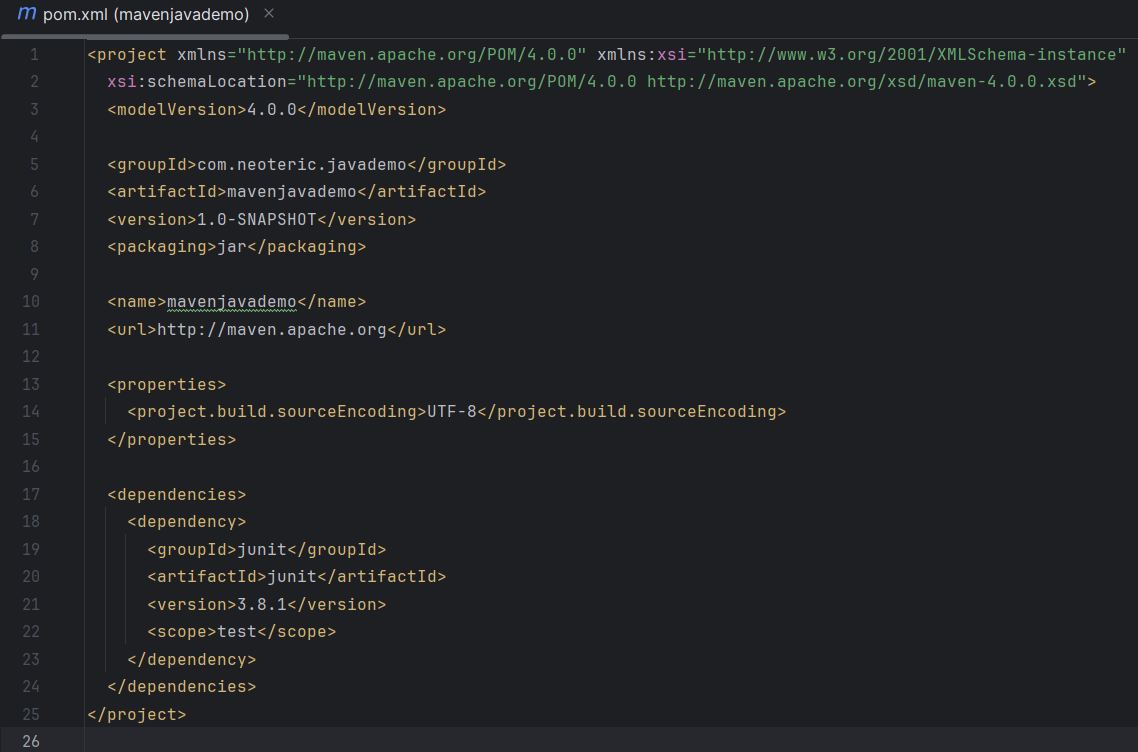
**Create a Simple Maven Project**

* we create a sample maven project with required dependencies.
* Once project successfully created then observe the project folder structure.
* Then locate the pom.xml file that has required configuration for the maven project.

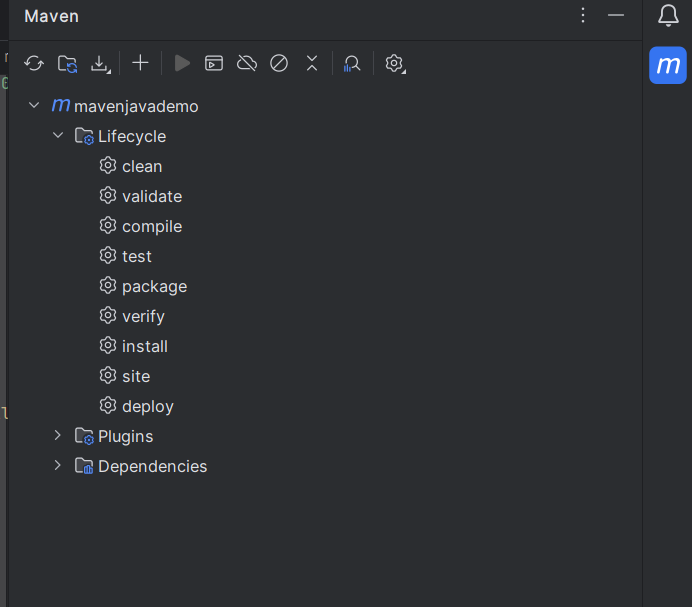
Step 1: Create a Maven Project



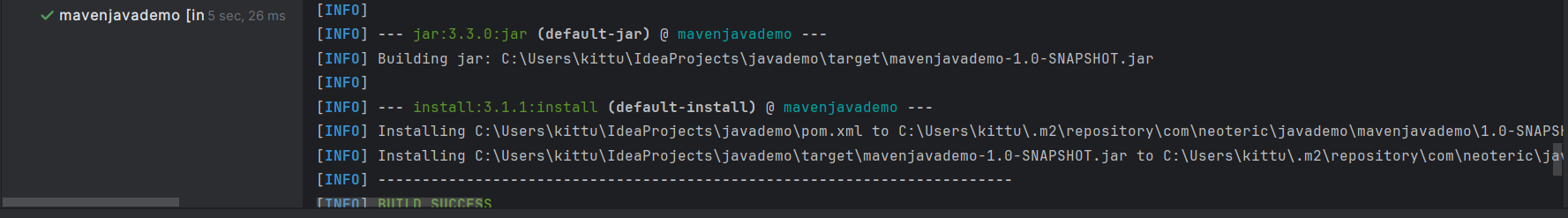
**Required Dependencies:**

****

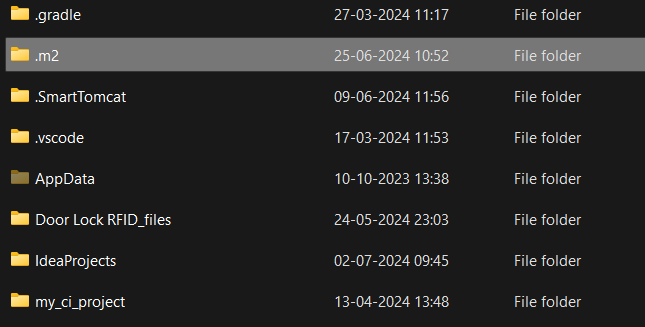
Step 2 : Clean and Install



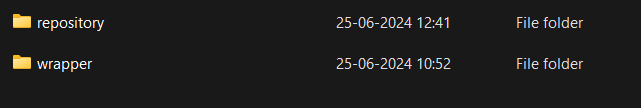
Step 3 : After Install Building a jar



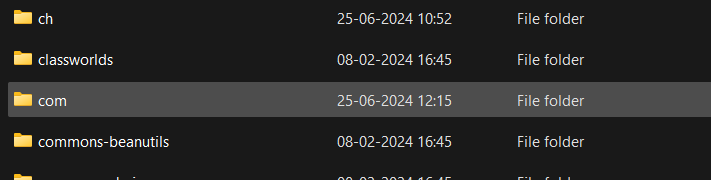
Step 4 : after Installing .m2 folder will be created click on folder



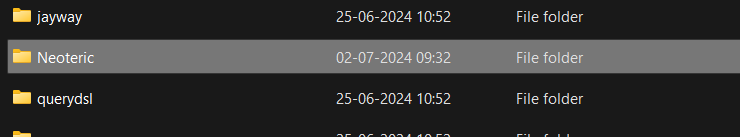
Step 5 : click on Local Repository



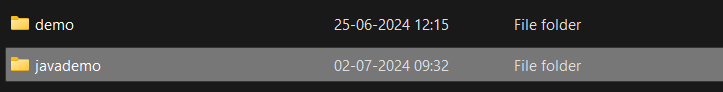
Step 5 : Click on Group Id Com



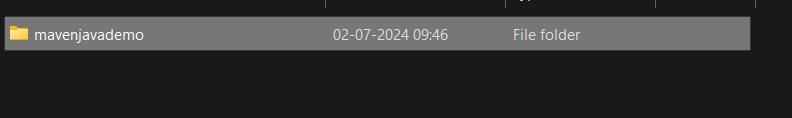
Step 6 : click on Neoteric GroupId



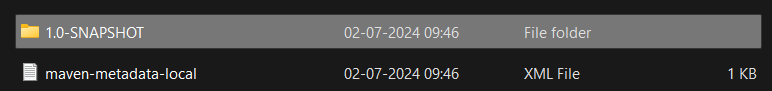
Step 7 : click on javademo GroupId



Step 8 : click on Arctifact id



Step 9 : Click on Version

Step 10 : Jar file will be created

