Maven

1. Maven is a Build tool.
2. Maven will help in the development activity. Starting from project template creation till the project execution in every Maven can be used.
3. The project template is also known as **Archetype**.
4. There are different stage in maven which helps in the development activity
5. Maven can compile you code in the **mvn compile** stage
6. Maven can help us to execute the test cases written inside project using **mvn test** stage
7. Maven can help us to create a package of the project using **mvn package** stage
8. Can execute the project using **mvn deploy** stage.
9. Maven will also help to manage the dependencies (jar files) inside project.

**Maven Setup**

1. Download Maven Zip file. (<https://maven.apache.org/download.cgi>)



1. Extract Zip file into any location (Prefer C drive)
2. Setting the environment variable for maven.
   1. **MAVEN\_HOME** : Create a New Variable and set Path of Maven extracted folder



* 1. **M2\_HOME**: Create a New Variable and set Path of Maven extracted folder



* 1. **Path**: Use Existing path variable and create new value inside Path variable

Value Must be: **%M2\_HOME%\bin**

1. Verify Maven Setup

Open a command prompt and use following command **mvn -version**



**Create Maven Project using IDE (Eclipse)**

1. “File” menu -> “New” Option -> select “Maven Project” option.
2. Keep default option as it is on the first page and click on “Next”
3. Search for the Archetype “org.apache.maven”
   1. **maven-archetype-quickstart**: This option is use to get the code java project template
   2. **maven-archetype-webapp**: This option is used to get the web application java project template
4. select an archetype and click on “Next”
5. Provide the following
   1. Group Id: Project package structure
   2. Artifact Id: In the name of the project
   3. Version: Keep the default version as it is.
   4. Package: keep the group id and package name same.
6. Click on “Finish” button

Maven Project Structure  
  


**Maven Dependency Management**  
  




**Maven Life Cycle (goals/stages)**

1. **Clean**: in this stage you can clean the previous execution of the Maven from the target folder.
2. **Validate**: In this stage the project correctness will be check such as project structure, dependencies etc.
3. **Compile**: The java classes will be compile and generated .class file inside a target folder.
4. **Test**: the unit test cases will be executed here.
5. **Package**: In this stage the project will be bundled into a .jar or .war file.
6. **Verify**: Maven will verify foe the package correctness.
7. **Install**: In this stage the package file will be store inside local repo and also it will be prepare for the execution.
8. **Deploy**: can execute application in this stage.

**Maven Scope**

Scope of the dependency is denotes till what time the jar file will be available.

There are total 5 maven scope

1. Compile: The dependency will be available till the compilation stage of the maven life cycle.
2. Runtime: The dependency will be available in the compilation and execution of the code. This is the default scope if you don not provide it explicitly.
3. Provided: this scope is use to get the dependency from the project instead of local or central repository.
4. Test: the dependency will be available till the test stage of the life cycle.
5. System: in this scope you can specify your own location for the dependency.