**1. Generate a project template in dinteractive mode.**

mvn archetype:generate -DgroupId=com.howtodoinjava -DartifactId=DemoJavaProject -DarchetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false

**2. Advantages of maven**

* Maven can add all the dependencies required for the project automatically by reading pom file.
* One can easily build their project to jar, war etc. as per their requirements using Maven.
* Maven makes easy to start project in different environments and one doesn’t need to handle the dependencies injection, builds, processing, etc.
* Often in a project, it’s required to migrate to new features, and with it, the migration is simplified.

**3. History of maven & what is maven**

Maven is a popular open-source build tool developed by the Apache Group to build, publish, and deploy several projects at once for better project management. The tool allows developers to build and document the lifecycle framework.

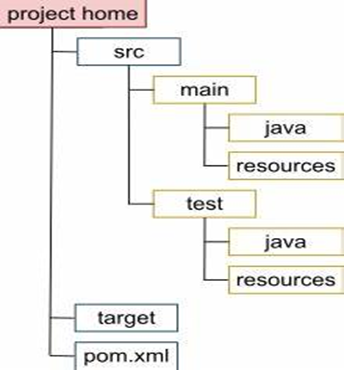
Maven is written in Java. It is based on Project Object Model (POM), which is an XML file that has all the information regarding project and configuration details. The POM has the description of the project, details regarding the versioning, and configuration management of the project.

The XML file is located in the project home directory. When you execute a task, Maven searches for the POM in the current directory.

It provides developers ways to manage the following −

* Builds
* Documentation
* Reporting
* Dependencies
* SCMs
* Releases
* Distribution
* Mailing list

**Maven project structure**



* src folder contains the source code.
* test folder contains the test cases.
* target folder contains the report and other project related configurations get generated here.
* Pom.xml (Project Object Model. Extensible Markup Language) contains all the dependencies and plugins

**4. Dependency and plugins**

Dependency is another archive—JAR, ZIP or WAR file which your current project needs in order to compile, build, test, and/or to run. The dependencies are gathered in the pom. When you run a build or execute a maven goal, these dependencies are resolved, and are then loaded from the local repository.

The maven plugins are central part of maven framework, it is used to perform specific goal.

Here follows the list of few common plugins:

* clean: Clean up target after the build. Deletes the target directory.
* compiler: Compiles Java source files.
* surefile: Run the JUnit unit tests. Creates test reports.
* jar: Builds a JAR file from the current project.
* war: Builds a WAR file from the current project.
* site: generates a site for the current project.