**Maven Properties:**

Maven is a complete project and java management tool.

Before maven, Ant was used to build projects, the problem with maven was that we had to build projects by writing codes manually. But with maven, writing codes/scripts manually is not required.

Maven has a property called archetypes, which can be used to generate web applications, Applications that use Spring MVC, struts and a whole lot of projects.

Maven defines a few things called Defaults. Maven uses convention over configuration. Some of these defaults are:

1. Your source files should be in a folder called src/main/java
2. Your test files should be in a folder called src/test/java
3. Your pom.xml file should be in the root folder because maven finds pom.xml file. (Project Object Model)

Go to the root folder using command prompt.

Check if you have Java installed in it by using command “java -version”.

Check if you have maven installed in it by using command “mvn --version”. If maven is installed, it will give you the location where maven home is installed.

“mvn compile”, this will compile your source file. This creates a new folder called target -> classes -> and there will be a class file corresponding to your source file.

*Maven-compiler-plugin:3.1:compile* helps to compile java.

“mvn test-compile”, this will compile both the source files and test files. This creates a new folder in target called test-classes -> and there will be a class file corresponding to your source file.

*Maven-compiler-plugin:3.1:testCompile* helps to compile java test files.

“mvn clean” command deletes everything from the “target” folder

“mvn test” command is used to run the junit tests. So this command will first compile the sources, then it will compile the test files and then it will run the tests.

**The phases of Maven Build Lifecycle:**

1. Validate
2. Compile
3. Test
4. Package (into Jar or War)
5. Integration Test
6. Verify
7. Install

So if you run any step, it will automatically run the upper steps. For example, Test command will first Validate, then Compile and the Test the application. Maven provides plugins, so you can attach plugins to each phase of the above lifecycle. There are some default plugins defined in maven which are available in parent pom.xml (super pom).