What is Maven?

Maven is a build automation tool that provides developers a complete build lifecycle framework. Development team can automate the project's build infrastructure in almost no time as Maven uses a standard directory layout and a default build lifecycle.

For Maven JDK is a pre requisite.

If you are using Selenium Webdriver.Jar files.

In Maven all the dependent jar files are implemented in the form of dependency. Maven centralized repository is available somewhere like USA.

Suppose u r creating project here…Instead of downloading all jar files u can add those dependency in POM file.

Iam using client for ex Eclipse and my maven project will be in this eclipse. And I need all the technologies (TestNg, Cucumber, SIKULI) for my project, then I will try to interact with Maven repo.In order to establish the connection between Eclipse and the Maven repo,maven plugin should be installed in eclipse .Maven will send request in the form of dependency.

Dependencies will have three attributes :Group Id , Artifact id and Version no

Maven is capable of interacting with global Maven repo as well as local/internal repo.



Maven Life Cycle:

1.Maven Compile.(By using Maven Compiler plugin)

2.Maven Test. (All test cases excuted) (By Surefire plugin)

3. Maven Resource. (To generate resources -JAR, WAR, EARS via Maven resource plugin)

maven-compiler-plugin

maven-surefire-plugin

maven-source-plugin

Maven Setup

1.Maven is a Java based tool, so the very first requirement is to have JDK installed on your machine.

2. \*Download Maven from here: [https://maven.apache.org/download.cgi](https://www.youtube.com/redirect?q=https%3A%2F%2Fmaven.apache.org%2Fdownload.cgi&v=tyLSFciTU-s&event=video_description&redir_token=LT4cDWFJAyC6vESiBQqepeCJx2J8MTUxODEwNTQ1OUAxNTE4MDE5MDU5)

3. <https://www.mkyong.com/maven/how-to-install-maven-in-windows/>

**Maven Set Up:**

1> Create Maven Project.

2>GroupId:MavenDemoTest ; ArtificatID:same.

3>Check for the folder struc created. /src/Java and src/test/java

4>Check for Pom XML

POM: Project Object Model(Heart of Maven)

Dependencies in POM xml:

1>Add TestNg dependency. (TestNg Maven google search)

2>Add selenium dependency.

3>Once you save , all the Jar files are added from internet automatically.

4>No need to externally add Jar files.

5>After adding the above two dependency , we can create a package and then write test class files in /src/Java/test.

6>Run as Maven Test.

7>No Test Ng report will be generated as we have run Maven project.

Maven Commands from Command Prompt:

1> Get the Maven Project location from properties.

2>Go to command prompt. And reach the maven location.

3>mvn clean install

4>It will execute all the four diff life cycles. Once life cycles are completed , test cases will be executed.It will create a build which is nothing but jar file.

The jar file can be given to the QA team to test it.

5>mvn test (execute only test cases ). Build wont be executed.

6>ignore the test cases by command mvn package -DskipTests (Not executing test cases onky build getting generated)

second way is by executing the command mvn package -Dmaven.test.skip=true

Add the below property in pom.xml:

<properties>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

<maven.test.skip>false</maven.test.skip> means all test cases r considered

</properties>

7.Create /MavenDemoTest/src/main/resources package and place a testng.xml file for executing multiple classes.And specify all the classes to be executed.

8>Add plugin to pom xml . (Compiler, Surefire and source plugin under build xml tag)

9>