Maven :   
  
-apache org

-java language

-CLI/IDE to work

-POM-project object model

-java projects only

-create project folder strecture

- download dependencies(jars)

-copile source code

-execute Junit test cases

-package our application (jar/war)

Work with Maven:

-CLI

-IDE

CLI use

Maven setup:

1 java home setup : java –version: upto jdk directory

2. java path setup : upto jdk/bin directory

3.maven home setup : mvn -version : dowland & install : upto mvn directory

3. maven apath setup: mvn/bin : upto

Maven Terminology:

1.architype

quick-start : stand-alnoe applicvation

web-app : web application

2.groupId -> company name

com.tcs

com.arcle

com.ashokIT

3.artifactId-> Project name

Sbi-car-loan-app

4.version ->

Snapshot – under development

Release -> release for customer

Sbi-car-loan-app-2.0

5.packajing : jar(or)war(or)(POM)

JAR :STANDALONE

WAR : WEB

Pom: MULTI MODULE(PARENT CHILED)

6. repositories: Locaton where maven dependencies will be maintained  
 a. central : public repo  
 b. remote : private repo -> Nexis(Jofrog)  
 c. local : in our system (c/user/username/.m2)

7.goals : To perform maven build procees

Evry gol will associate with pluging that performs required task

- clean -> delete byte code ( target )

-complie -> complie source code

-test -> runt joint test caeses  
-package -jar (or) war

8.dependencies : LIBRARIES REQUIORED FOR PROJECT

EX: JUNIT,SPRINGHIBERNATE

9.scope : when to use dependency

Ex: <scope>test</test>

10. Pom.xml : to manage depencies maven configuration filem

GOALS:  
  
  
mvn clean : delete byte code  
mvn compile: generate byte code(target)

Mvn test : runt unit testing

Mvn package :package the applicate

Mvn clean complie test package = mvn package

Repository flow: C:\apache-maven-3.9.9-bin\apache-maven-3.9.9\conf\settings.xml

Repote repo  
(Nexes/jfrog)

Central repo

Pom.xml

LocalRepo

Project

Name :  
<build>  
<finalName> akash\_const</finalName>  
</build>

Mvn install -> upload to Repositroy

**Transitive dependency Management**  
by adding 1 dependancy -> multiple dependency can add  
**Dependency Exclusion:**  
  
<dependency>

<exclusions>

**<exclusion>**

**<groupId></groupId>  
<artifactId></ArtifactId>**

**</exclusion>**

<\exclusions>  
  
</dendency>

Multi module project :   
  
package is POM.

To create 1. Create with paakeging POM  
 2. Create maven Module in the parent package

Dependency Scope :

Default : compile   
  
others : (6) compile , provide ,runtime, test, system , import

Source code repository servers :   
  
SVN

Git Hub : microsoft

Bit Byucket : atlasian

Why :  
1. Code integration

2. code changes tracking(who,whe, why,where)