2018

**Maven Explained**

**Environment: Windows**

**Vibranarayanan**

Table of Contents

[Daily Learning tracker 2](#_Toc513841629)

[What is Maven 3](#_Toc513841630)

[What is POM 4](#_Toc513841631)

[Maven Objectives 4](#_Toc513841632)

[Install Maven 5](#_Toc513841633)

[Prerequisites 5](#_Toc513841634)

[Download Maven 5](#_Toc513841635)

[Setting up Environment Variable 5](#_Toc513841636)

[Verify Maven Installation 6](#_Toc513841637)

[Creating Maven Project Command Line 6](#_Toc513841638)

[Project Initiation 6](#_Toc513841639)

[Folder structure verification after mvn generation 8](#_Toc513841640)

[Java code path 8](#_Toc513841641)

[Junit path 8](#_Toc513841642)

[POM file path 8](#_Toc513841643)

[POM file 9](#_Toc513841644)

[Explore Maven goals/commands (Command Line) 9](#_Toc513841645)

[mvn clean 9](#_Toc513841646)

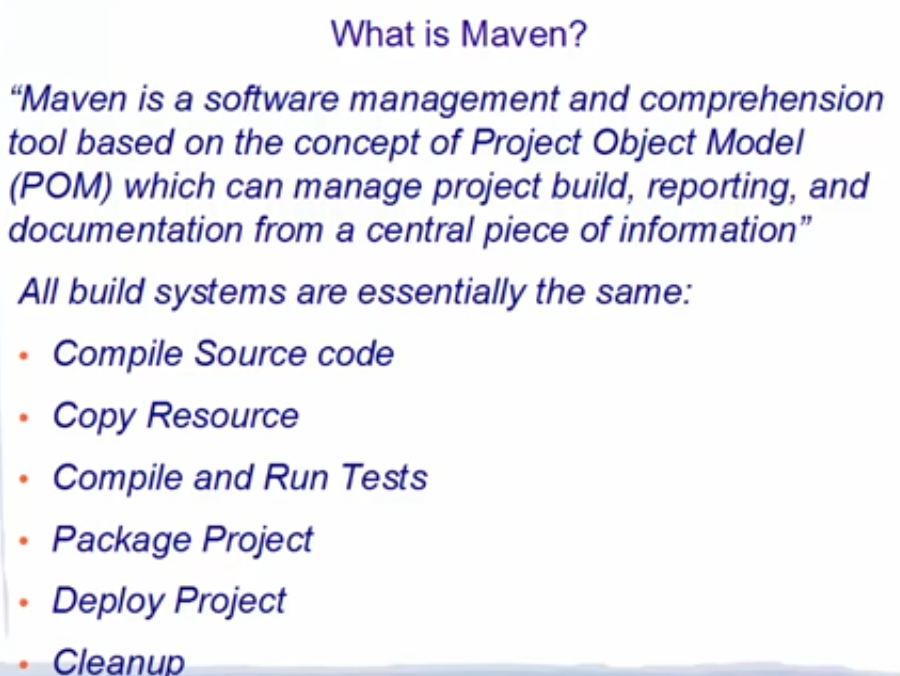
[mvn compile 9](#_Toc513841647)

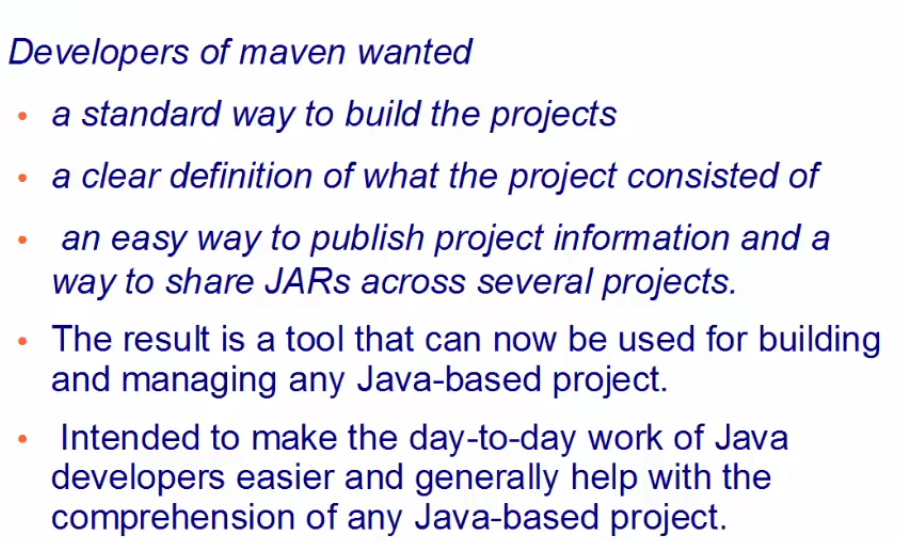
[mvn test-compile 10](#_Toc513841648)

# Daily Learning tracker

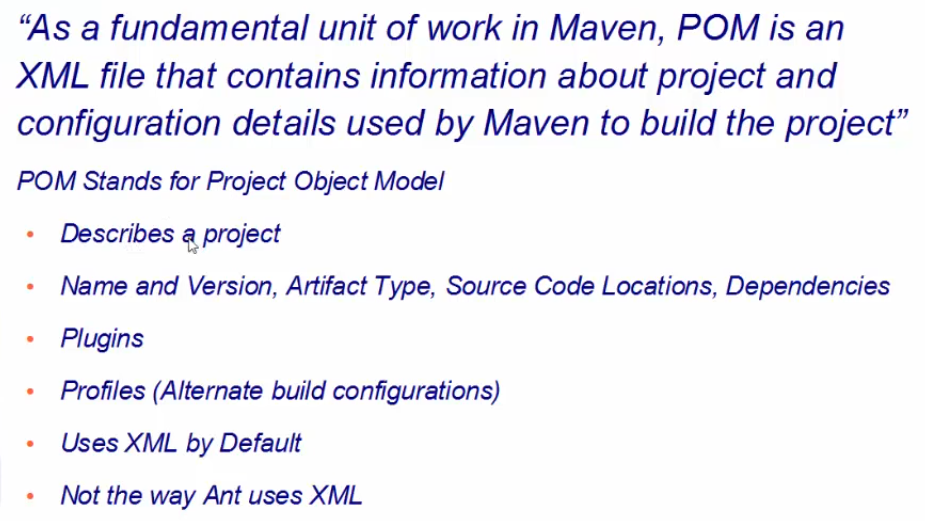
|  |  |
| --- | --- |
| Date | Topics |
| 05/11/2018 | * Learn maven basics * install maven * create maven project * Explore mvn commands * document learnings/hand-on |

# What is Maven

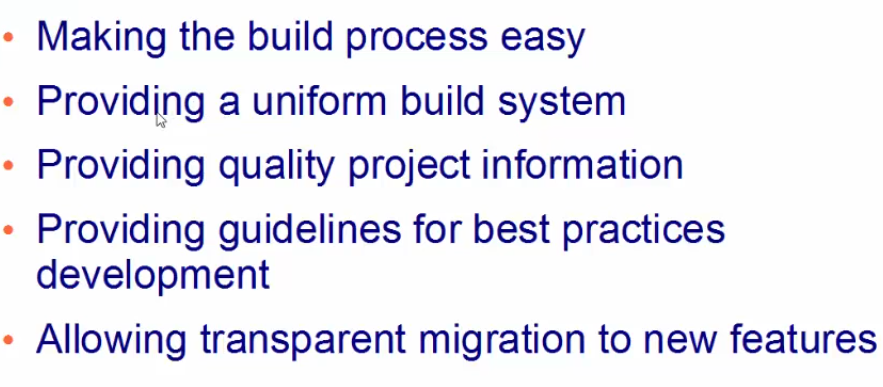




# What is POM

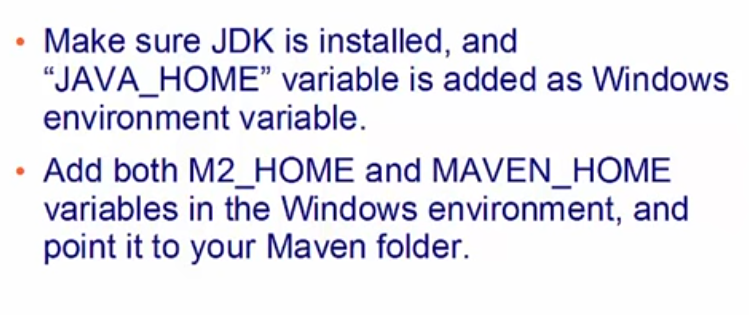


# Maven Objectives

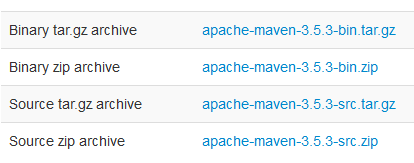


# Install Maven

## Prerequisites

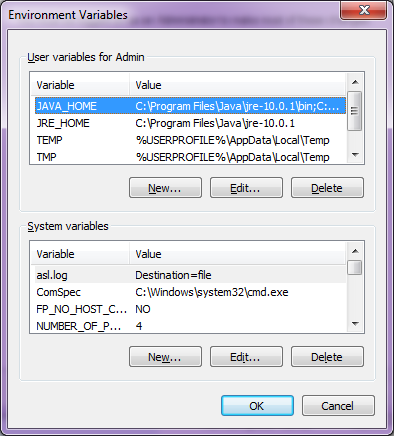


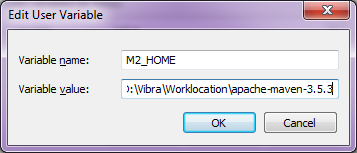
## Download Maven

URL: https://maven.apache.org/download.cgi  
  
Download 2nd option Binary Zip and extract it in your desire location.

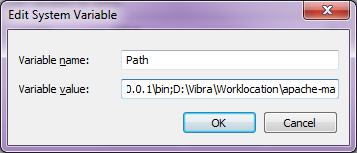
## Setting up Environment Variable

Go to computer properties and select Environment variable.

U  
Under user variable click new and add M2\_HOME variable with Maven extracted directory.



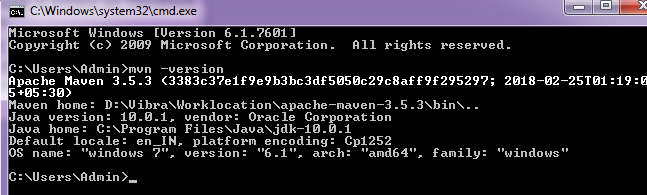
Click Ok and if needed added MAVEN\_HOME. since few projects use this variable with this name.



go to bin directory of maven and copy and append that path with existing path variable. done change existing variable. add ; at the end and past the path

Example: ; D:\Vibra\Worklocation\apache-maven-3.5.3\bin

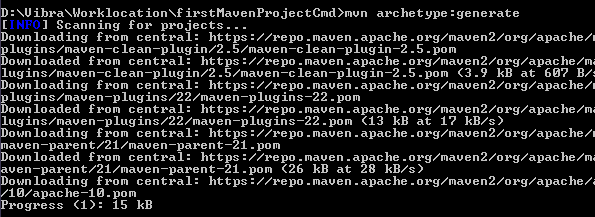
## Verify Maven Installation

execute mvn -version. This will list maven version. seems now all set with maven instalation.

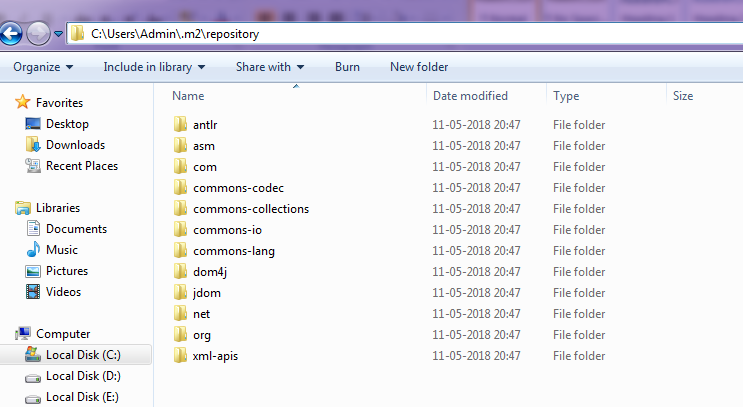
# Creating Maven Project Command Line

## Project Initiation

Step 1: create a project folder Ex: " D:\Vibra\Worklocation\**firstMavenProjectCmd**"  
Step 2: Navigate this folder in command line. and type mvn archetype: generate and enter  

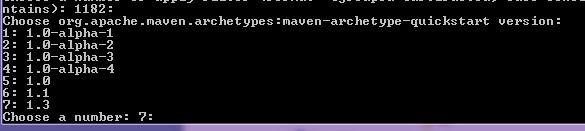
this will download required plug-ins and store it in your local for the first time.



These archetype plug-ins will be stored in your use directory ".m2"

once all archetype plug-ins downloaded this will ask for template id just select enter number shown (1182) or press enter.

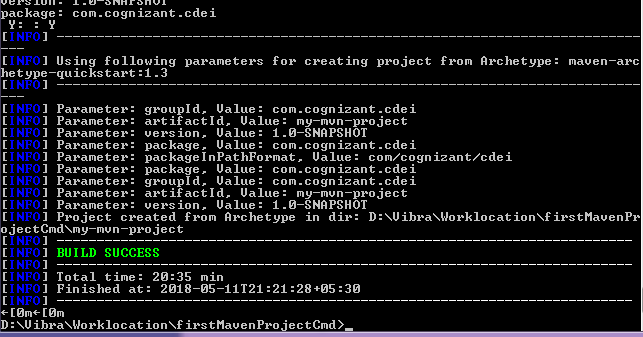




you can select required arche type. we will go with latest one (7) and enter. this will load required items.  
System will ask for groupid, artifactid and version. based on project need/hierarchy you can enter details given below.

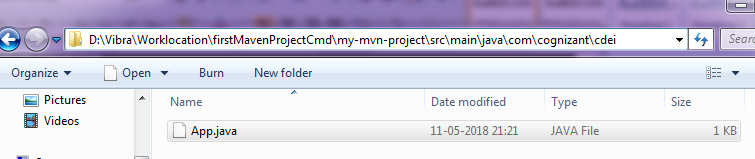


Once you enter all the details tie will start building folder structure for you. you can verify these in selected project location.



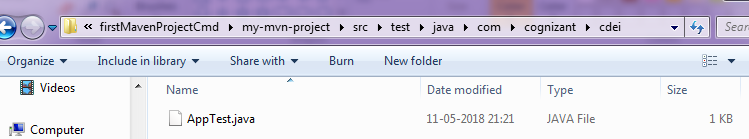
## Folder structure verification after mvn generation

### Java code path

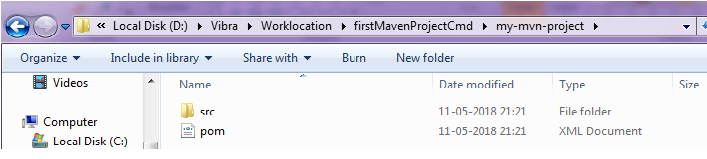


App.java created for us to use and all folder structure created based on our input given in groupid, artifactid and version.

### Junit path



### POM file path



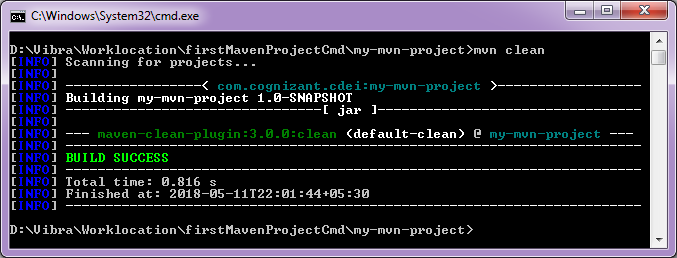
### POM file

  
This POM file created in this process with all the dependencies.

# Explore Maven goals/commands (Command Line)

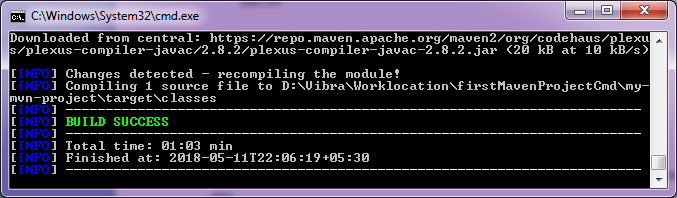
below commands can be used as goals in maven Jenkins Job.

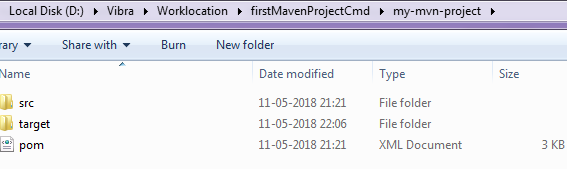
## mvn clean

Go to newly created project's POM file location. This command will clear already created artefacts if any.

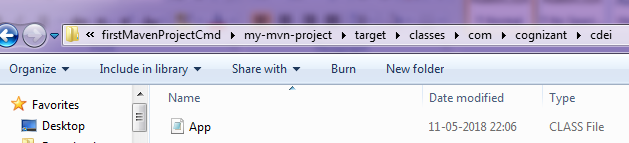
## mvn compile

This command will download required dependencies and compile our java and create .class files.





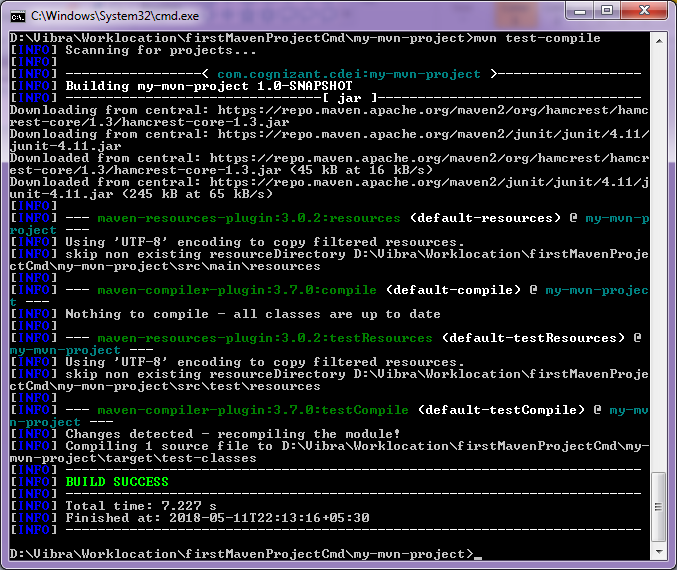
You can see target folder is created under our project, after running this mvn compile command.



Compiled class file generated under target location.

## mvn test-compile

This command will compile available Junit test case



test-classes created under target folder and class files available for further use.

