

Maven

...

Apache Maven Project

Learn More

Maven Tutorial - luv2code (14 videos / 1 hour)

<https://www.youtube.com/playlist?list=PLEAQNNR8IlB7uvr8EJbCNJq2I82A8cqE7>

Maven Overview

Mavens Solution

Apache Software Foundation

What is Maven

Maven Commands

Standard Directory Structure

POM file

What Problems Does Maven Solve?

We've pretty much have just used java api's for developing..

When building your Java project, you may need additional JAR files

JDBC, JUnit, Spring, Commons, etc.

One approach is to download the JAR files and manually add to build path

OR

Maven manages all these dependencies for you

Introduction

Standard way to build projects

Apache Software Foundation

Ant

Commons

Cordova

Groovy

Kafka

NetBeans

OpenJPA

Tomcat

Struts

What is Maven?

Making the build process easy

Providing a uniform build system

Providing quality project information

Encouraging better development practices

Maven Commands (phases - life cycle of the build)

Phase	Action
clean	Delete target directory.
validate	Validate if the project is correct.
compile	Compile source code, classes stored in target/classes.
test	Run tests.
package	Take the compiled code and package it in its distributable format, e.g. JAR, WAR.
verify	Run any checks to verify the MVN package is valid and meets quality criteria.
install	Install the package into the local repository.
deploy	Copies the final MVN package to the remote repository.

What we will go over

Create Maven projects

Add dependencies to Maven pom.xml

Build and run Maven projects

Run Maven from the Command-Line

Maven Solution

Like a personal shopper

Tell Maven what you want (JAR files for projects)

Maven will go out and get it (Download)

Maven will make those JAR files available during compile / run

Standard Directory Structure

AimUnit4ClassExercises

src

/main/java/com/aim/movie

/test/java/com/aim/movie

pom.xml

POM

Project Object Model file

Project Meta-Data

Dependencies

Plug ins

```
<project ...>

    <modelVersion>4.0.0</modelVersion>

    <groupId>com.aim.movie</groupId>
    <artifactId>movie</artifactId>
    <version>1.0-SNAPSHOT</version>

    <name>movie</name>

    <dependencies>
        <dependency>
            <groupId>junit</groupId>
            <artifactId>junit</artifactId>
            <version>4.11</version>
            <scope>test</scope>
        </dependency>
    </dependencies>

    <!-- add plugins for customizations -->

</project>
```

Project Coordinates

Group ID Name of company, etc. reverse domain name

`com.aim`

Artifact ID Name for the project

`AimJavaUnit4ClassExercises`

Version A specific release version

`1.0-SNAPSHOT`

Maven Summary

Mavens Solution

Apache Software Foundation

What is Maven

Maven Commands

Standard Directory Structure

POM file

Code together

Create a Maven Project together

Create a Maven Project

1. Create Maven Project
 - a. Open Command Palette (Cmd(Win)+Shift+P)
 - b. Type: Create Maven Project
 - c. Select maven-archtype-quickstart
 - d. Select 1.4
 - e. Select Destination
 - f. groupId: com.aim.movie
 - g. artifactId: AimUnit4JDBC
 - h. Version: <Hit Enter to accept default> (Default 1.0-SNAPSHOT)
 - i. Confirm properties configuration:
 - j. <Hit Enter to accept default>

1. Initialize Git Repository for Project
 - a. Open Command Palette
 - b. Type: Git Initialize Repository
 - c. Hit Enter twice
2. Add .gitignore file
 - a. Add generated project files you don't want to push to github
3. Stage and commit files
4. Create GitHub Repository
 - a. Github.com: Create a new repository
Called: AimUnit4JDBC
 - b. Accept Defaults; don't create readme, etc.
 - c. Copy Git URL
5. Push Maven project to Github
 - a. Open Command Palette
 - b. Type: Git Add Remote: Paste Git URL
 - c. Enter Name: origin