

Maven
Created by
github.com/farzadafi

General

Maven is a powerful project management tool that is based on POM (project object model).

It is used for projects build, dependency and documentation.

problem without Maven

- Adding set of Jars in each project
 - In case of struts, spring, hibernate frameworks, we need to add set of jar files in each project. It must include all the dependencies of jars also.
- Creating the right project structure
 - We must create the right project structure in servlet, struts etc, otherwise it will not be executed.
- Building and Deploying the project
 - We must have to build and deploy the project so that it may work.

What it does?

- Maven simplifies the above mentioned problems. It does mainly following tasks.
 - It makes a project easy to build
 - It provides uniform build process (maven project can be shared by all the maven projects)
 - It provides project information (log document, cross referenced sources, mailing list, dependency list, unit test reports etc.)
 - It is easy to migrate for new features of Maven

What is Build Tool

- A build tool takes care of everything for building a process. It does following:
 - Generates source code (if auto-generated code is used)
 - Generates documentation from source code
 - Compiles source code
 - Packages compiled code into JAR or ZIP file
 - Installs the packaged code in local repository, server repository, or central repository


JAR

- A JAR (Java ARchive) file is a compressed file format used to package Java classes, resources, and metadata into a single file.
- It allows you to distribute and deploy Java applications or libraries in a convenient manner.
- some key points about JAR files:
 - Compression
 - JAR files use the ZIP compression algorithm, allowing multiple files to be packaged together while reducing their overall size.
 - Portability
 - JAR files are platform-independent and can be executed on any system that has the Java Runtime Environment (JRE) installed.
 - Organization
 - A JAR file can contain compiled class files, resource files (such as images or configuration files), library dependencies, and other related artifacts.
 - Executable
 - Some JAR files may include an executable entry point called the Main-Class manifest attribute. This allows them to be run directly using the java -jar command.
 - Library Packaging
 - Libraries often distribute their code as JAR files for easy integration into other projects.
 - These libraries can then be added as dependencies in build tools like Maven or Gradle.

show in code :)

- create a project with some validation for get input from user
 - command for create a new project in maven
 - mvn archetype:generate -DgroupId=com.example -DartifactId=my-app -DarchetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false
- create a jar file from it and use it from another project
- create a library that use it in maven :)
 - command for create a library
 - mvn install:install-file -Dfile=(your jar file address) -DgroupId=com.asarshakam -DartifactId=(library-api) -Dversion=1.0.0 -Dpackaging=jar

some simple dependency

- please write code for these methods
 - countMatches
 - Counts how many times the substr appears in the larger string.
 - getDigits
 - get all digits from a string
 - Checks if the CharSequence contains only uppercase characters.
 - isAllUpperCase
- code of this class please click on this symbol: 
 - for use this code we have 3 way:
 - first copy and paste all method that i need in my code
 - second download jar file from internet and add to my project
 - thrid use from maven :)
 - commons-lang3
 - ```
<dependency>
<groupId>org.apache.commons</groupId>
<artifactId>commons-math3</artifactId>
<version>3.6.1</version>
</dependency>
```

.m2

- The .m2 directory is a default directory created by Maven in your user home directory
  - C:\Users\YourUsername\.m2 on Windows
  - /Users/YourUsername/.m2 on macOS/Linux
  - Note
    - The location of the .m2 directory can be customized by modifying the <localRepository> element in your settings.xml, but it defaults to being located within your user home directory if not explicitly changed.
- It serves as the local repository for Maven artifacts.
- what you'll find inside the .m2
  - Repository
    - The main subdirectory within .m2 is repository
    - This is where all downloaded and cached dependencies are stored.
    - Each dependency will have its own folder structure based on its groupId, artifactId, and version.
  - Settings
    - The settings.xml file resides in the .m2 directory and contains configuration settings for Maven.
    - You can modify this file to customize various aspects of Maven's behavior, such as specifying remote repositories or configuring proxy settings.
  - Plugins
    - If you use plugins that require additional configuration, their specific configurations may be stored under the .m2 directory as well.
- for what .m2?
  - The purpose of this local repository is to cache dependencies so that subsequent builds don't need to download them again from remote repositories unless there are updates or changes specified in your project's POM file.

Lifecycle

- Maven has several built-in lifecycle phases that define the order in which goals (plugins) are executed.
- Clean
  - This lifecycle is responsible for cleaning the project by removing any build artifacts generated from previous builds.
  - The primary goal of this phase is to ensure a clean and consistent starting point for building.
  - Deletes all files and directories generated during the build process.
- validate
  - Validates if the project's structure and dependencies are correct.
- compile
  - Compiles source code into bytecode.
- test
  - Runs unit tests using a suitable testing framework.
- package
  - Packages compiled code into a distributable format like JAR or WAR.
- verify
  - Performs checks on packaged code to ensure quality standards are met.
- install
  - Installs packaged artifact in your local repository for use as a dependency in other projects.
- deploy
  - Copies final package to remote repository for sharing with other developers or systems.