[CS304] Lab06 Maven

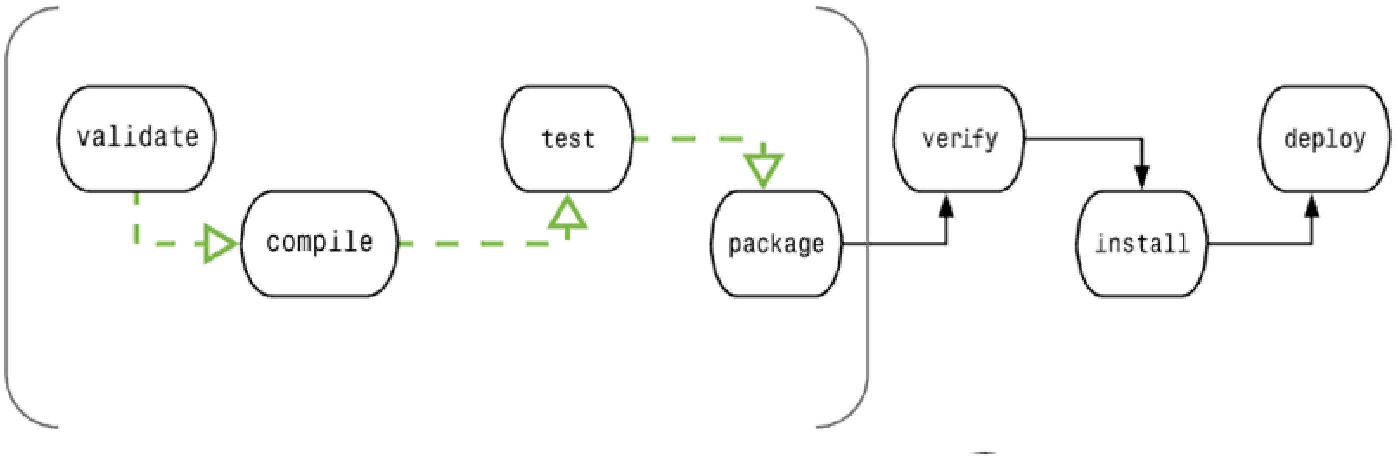
Part 1 Maven Introduction

Maven is a software project management and comprehension tool,can be used for building and managing any Java-based project.Based on the concept of a project object model (POM), Maven can manage a project's build, reporting and documentation from a central piece of information.

Maven deals with several areas of concern:

```
1.Making the build process easy
2.Providing a uniform build system
3.Providing quality project information
4.Encouraging better development practices
5.Making the build process easy
```

A Build Lifecycle is Made Up of Phases.
Each of these build lifecycles is defined by a different list of build phases, wherein a build phase represents a stage in the lifecycle.



When the default lifecycle is used, Maven will first validate the project, then will try to compile the sources, run those against the tests, package the binaries (e.g. jar), run integration tests against that package, verify the integration tests, install the verified package to the local repository, then deploy the installed package to a remote repository.
For example, the default lifecycle comprises of the following phases:

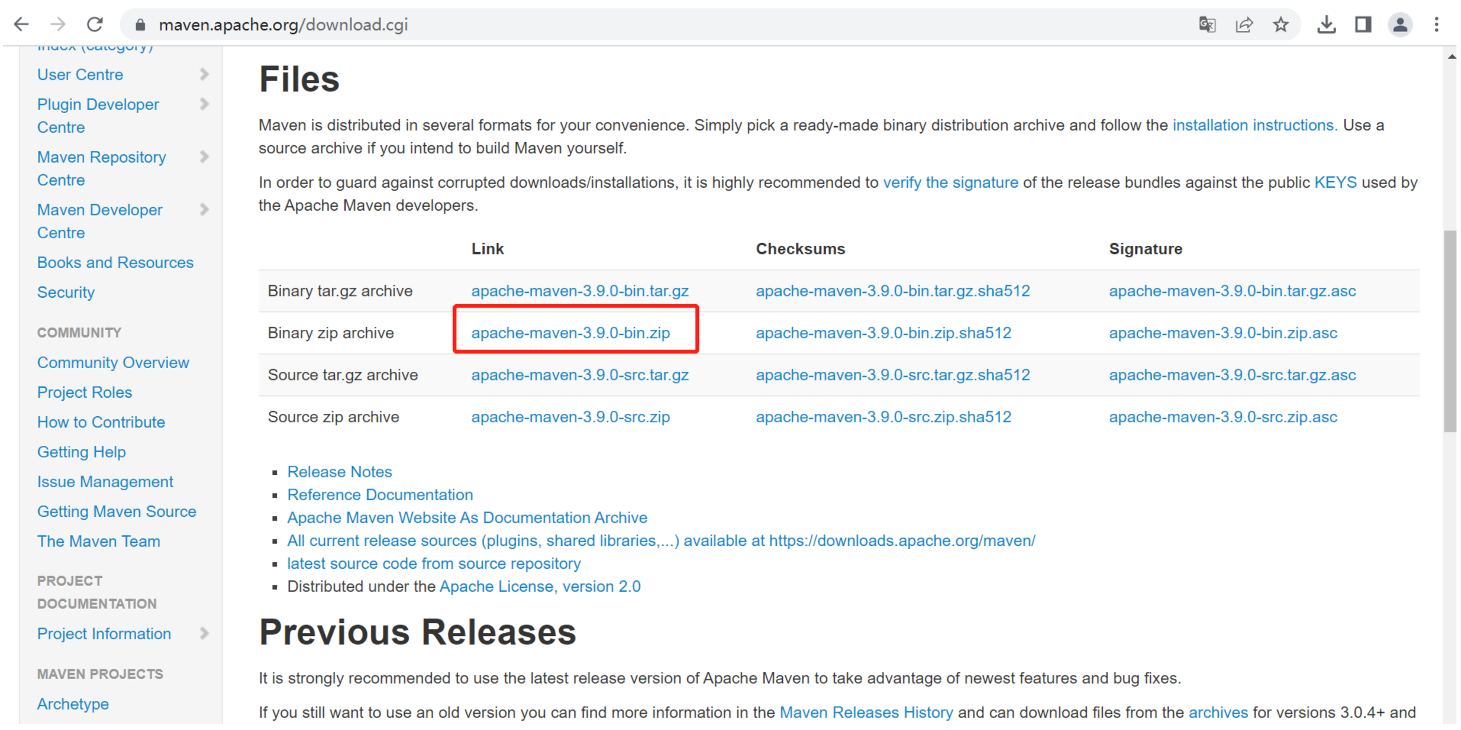
```
validate - validate the project is correct and all necessary information is available
compile  - compile the source code of the project
test     - test the compiled source code using a suitable unit testing framework. These tests
should not require the code be packaged or deployed
package  - take the compiled code and package it in its distributable format, such as a JAR.
verify   - run any checks on results of integration tests to ensure quality criteria are met
install  - install the package into the local repository, for use as a dependency in other
projects locally
deploy   - done in the build environment, copies the final package to the remote repository for
 sharing with other developers and projects.
```

These lifecycle phases (plus the other lifecycle phases not shown here) are executed sequentially to complete the default lifecycle.

You can refer below for Maven:

https://maven.apache.org/index.html

Download and install(you can use command also):
URL:https://maven.apache.org/download.cgi



Files

Maven is distributed in several formats for your convenience. Simply pick a ready-made binary distribution archive and follow the installation instructions. Use a source archive if you intend to build Maven yourself.

In order to guard against corrupted downloads/installations, it is highly recommended to verify the signature of the release bundles against the public KEYS used by the Apache Maven developers.

	Link	Checksums	Signature
Binary tar.gz archive	apache-maven-3.9.0-bin.tar.gz	apache-maven-3.9.0-bin.tar.gz.sha512	apache-maven-3.9.0-bin.tar.gz.asc
Binary zip archive	apache-maven-3.9.0-bin.zip	apache-maven-3.9.0-bin.zip.sha512	apache-maven-3.9.0-bin.zip.asc
Source tar.gz archive	apache-maven-3.9.0-src.tar.gz	apache-maven-3.9.0-src.tar.gz.sha512	apache-maven-3.9.0-src.tar.gz.asc
Source zip archive	apache-maven-3.9.0-src.zip	apache-maven-3.9.0-src.zip.sha512	apache-maven-3.9.0-src.zip.asc

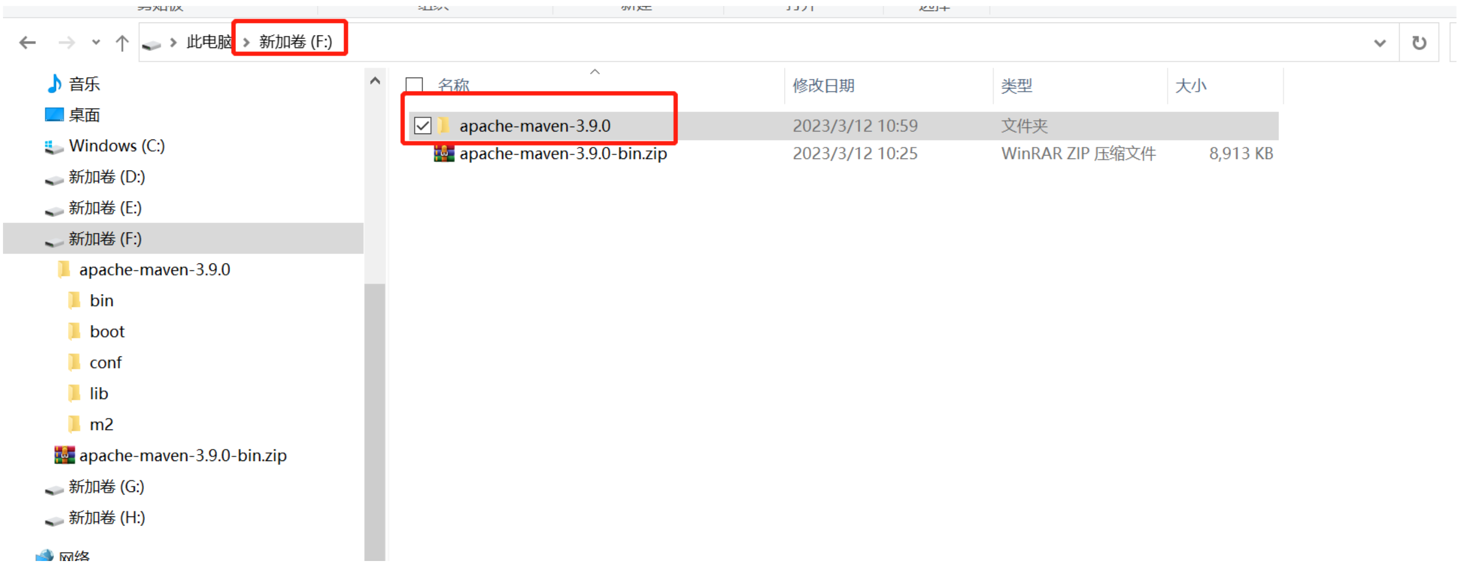
- Release Notes
- Reference Documentation
- Apache Maven Website As Documentation Archive
- All current release sources (plugins, shared libraries,...) available at https://downloads.apache.org/maven/
- latest source code from source repository
- Distributed under the Apache License, version 2.0

Previous Releases

It is strongly recommended to use the latest release version of Apache Maven to take advantage of newest features and bug fixes.

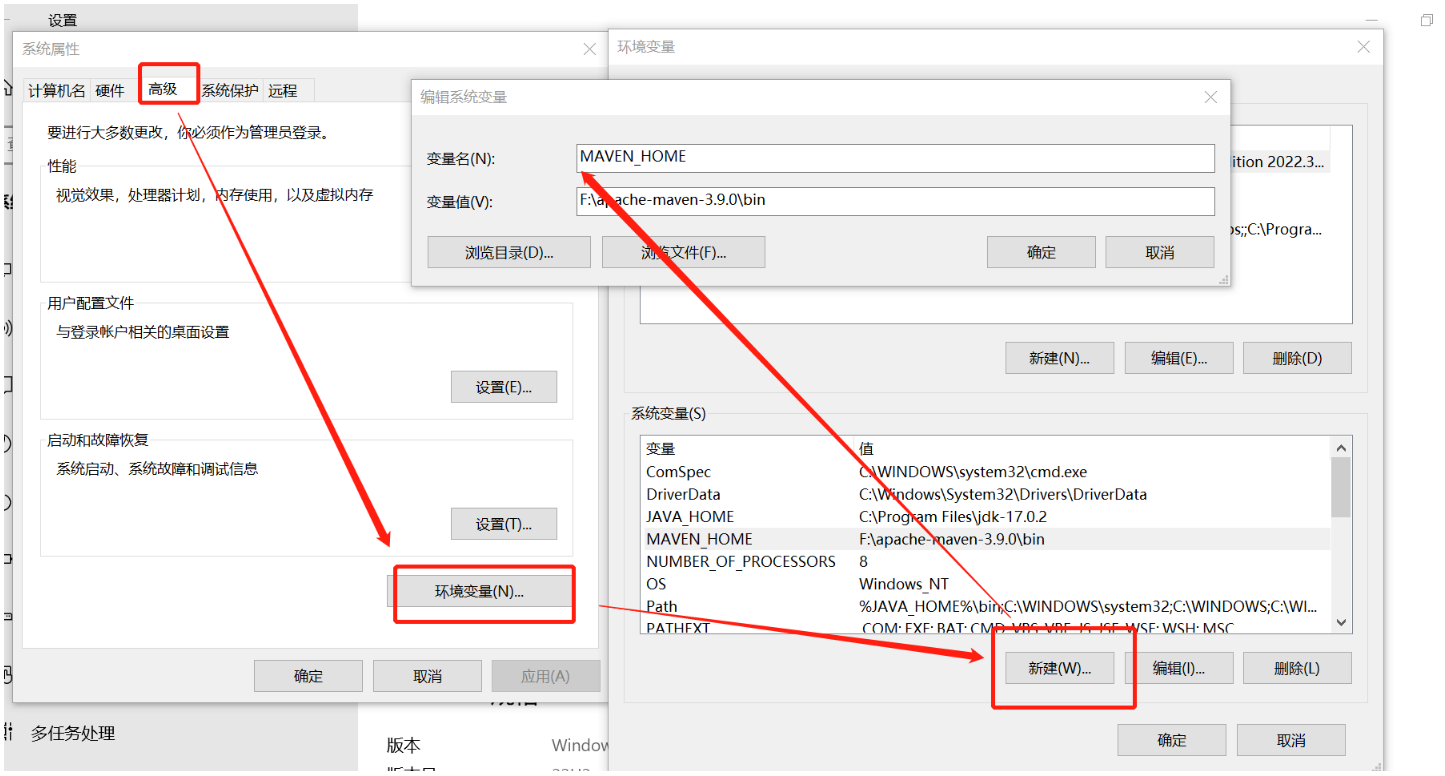
If you still want to use an old version you can find more information in the Maven Releases History and can download files from the archives for versions 3.0.4+ and

Extract distribution archive in any directory:



名称	修改日期	类型	大小
apache-maven-3.9.0	2023/3/12 10:59	文件夹	
apache-maven-3.9.0-bin.zip	2023/3/12 10:25	WinRAR ZIP 压缩文件	8,913 KB

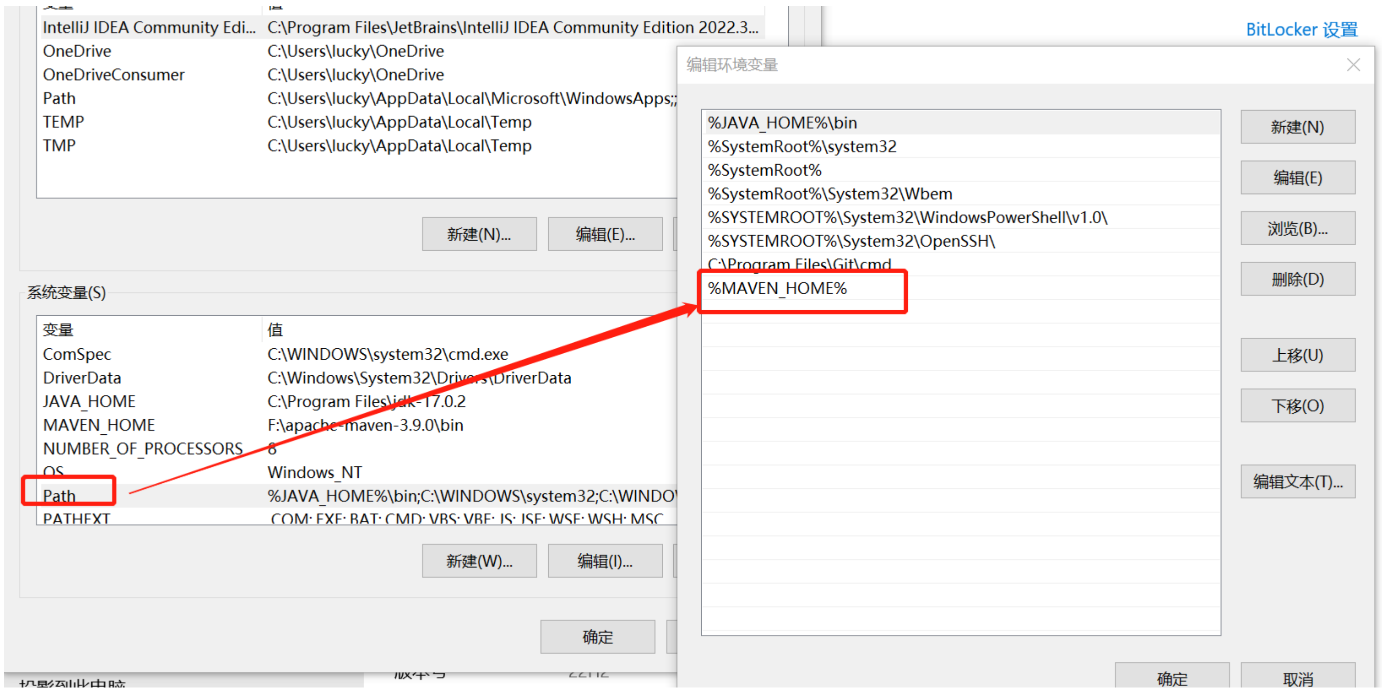
Config environment variables:MAVEN_HOME



变量名(N): MAVEN_HOME

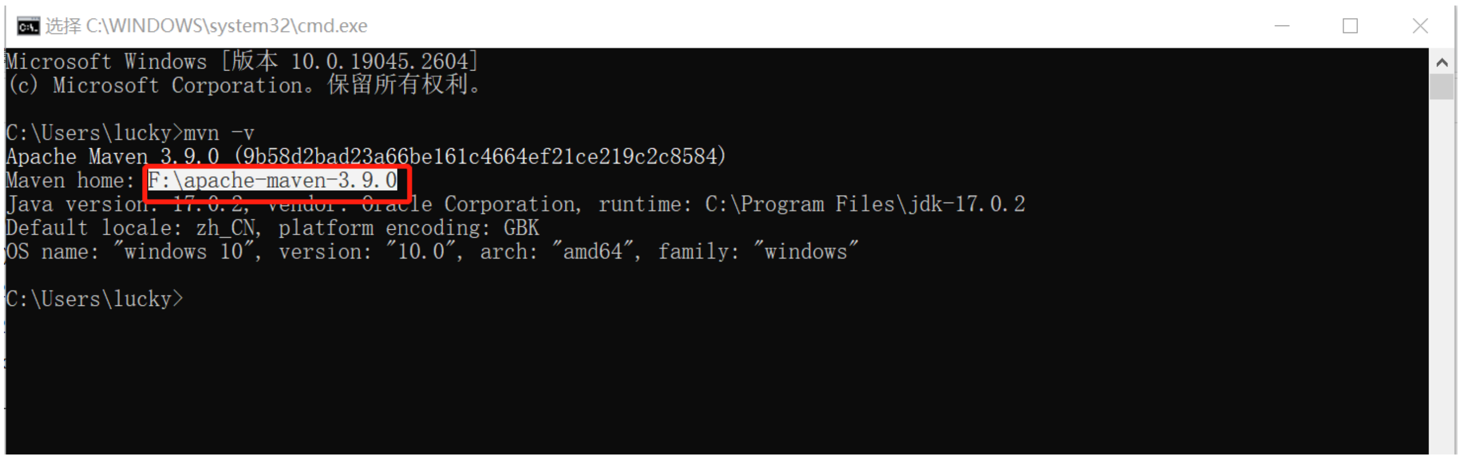
变量值(V): F:\apache-maven-3.9.0\bin

变量	值
ComSpec	C:\WINDOWS\system32\cmd.exe
DriverData	C:\Windows\System32\Drivers\DriverData
JAVA_HOME	C:\Program Files\jdk-17.0.2
MAVEN_HOME	F:\apache-maven-3.9.0\bin
NUMBER_OF_PROCESSORS	8
OS	Windows_NT
Path	%JAVA_HOME%\bin;C:\WINDOWS\system32;C:\WINDOWS;C:\WI...


```
IntelliJ IDEA Community Edi...   C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2022.3...
OneDrive                         C:\Users\lucky\OneDrive
OneDriveConsumer                 C:\Users\lucky\OneDrive
Path                             C:\Users\lucky\AppData\Local\Microsoft\WindowsApps;;
TEMP                             C:\Users\lucky\AppData\Local\Temp
TMP                              C:\Users\lucky\AppData\Local\Temp

系统变量(S)
变量                              值
ComSpec                          C:\WINDOWS\system32\cmd.exe
DriverData                       C:\Windows\System32\Drivers\DriverData
JAVA_HOME                        C:\Program Files\jdk-17.0.2
MAVEN_HOME                       F:\apache-maven-3.9.0\bin
NUMBER_OF_PROCESSORS             8
OS                               Windows_NT
Path                             %JAVA_HOME%\bin;C:\WINDOWS\system32;C:\WINDO
PATHEXT                          .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC

编辑环境变量
%JAVA_HOME%\bin
%SystemRoot%\system32
%SystemRoot%
%SystemRoot%\System32\Wbem
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\
%SYSTEMROOT%\System32\OpenSSH\
C:\Program Files\Git\cmd
%MAVEN_HOME%
```

Open dos and run command:

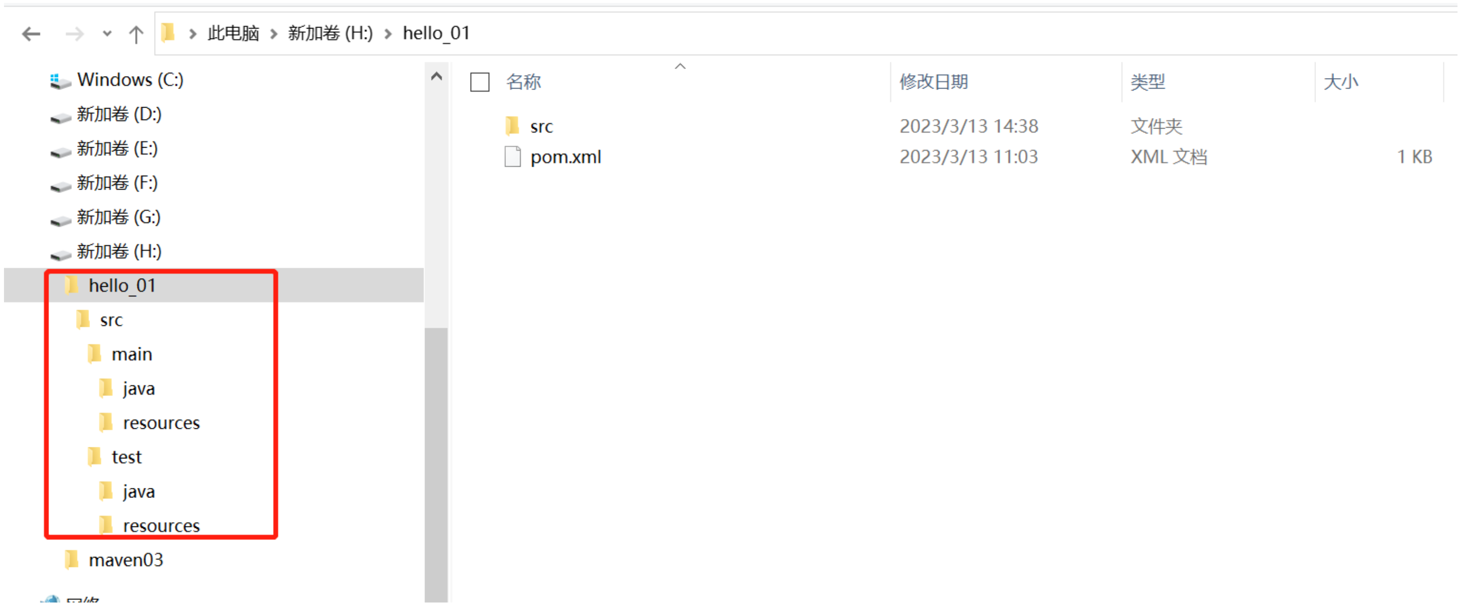
```
mvn -v
```


```
选择 C:\WINDOWS\system32\cmd.exe
Microsoft Windows [版本 10.0.19045.2604]
(c) Microsoft Corporation。保留所有权利。

C:\Users\lucky>mvn -v
Apache Maven 3.9.0 (9b58d2bad23a66be161c4664ef21ce219c2c8584)
Maven home: F:\apache-maven-3.9.0
Java version: 17.0.2, vendor: Oracle Corporation, runtime: C:\Program Files\jdk-17.0.2
Default locale: zh_CN, platform encoding: GBK
OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"

C:\Users\lucky>
```

Part 2 Run a simple Maven Java project

1. Simple Java project by hand

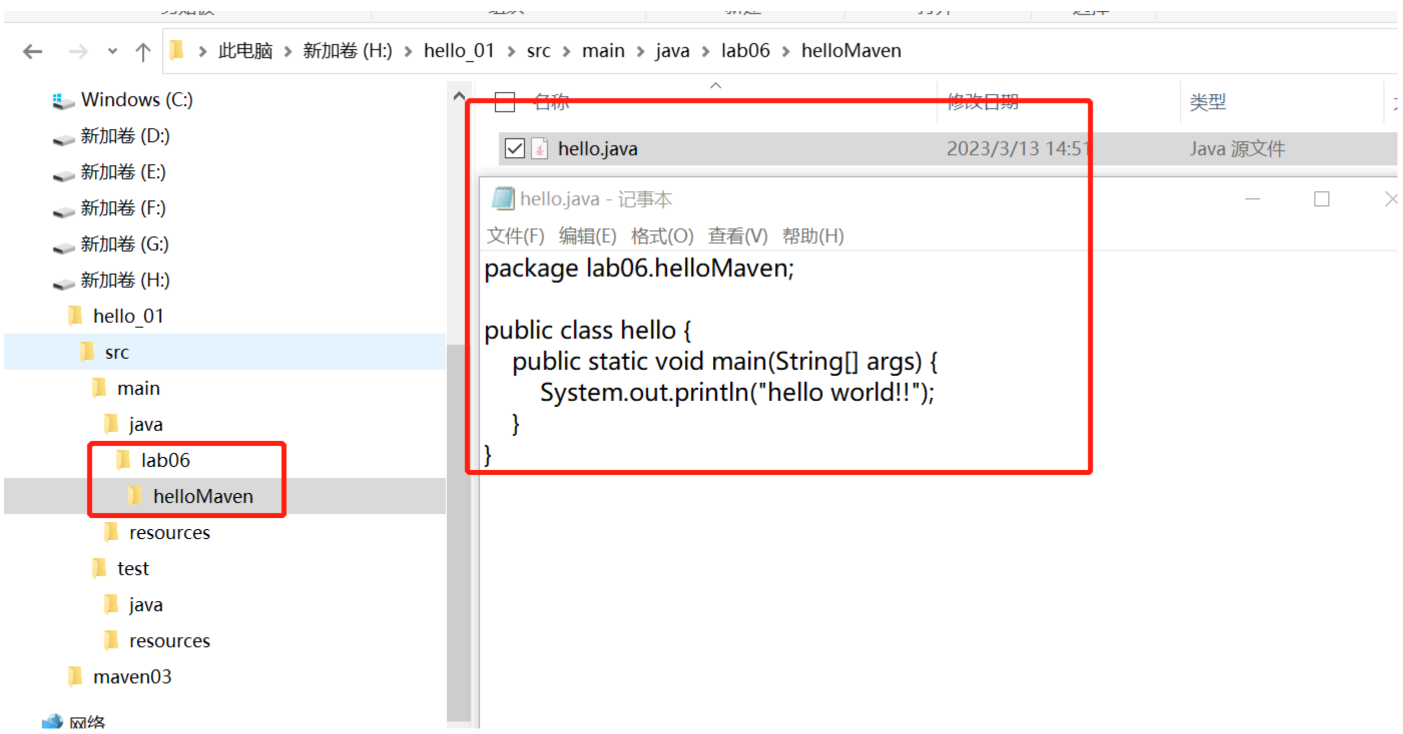
1. Create files and pom.xml below:
   The src/main/java directory contains the project source code, the src/test/java directory contains the test source, and the pom.xml file is the project's Project Object Model, or POM.



名称	修改日期	类型	大小
src	2023/3/13 14:38	文件夹	
pom.xml	2023/3/13 11:03	XML 文档	1 KB

2. Create files and hello.java:

```
package lab06.helloMaven;

public class hello {
    public static void main(String[] args) {
        System.out.println("hello world!!");
    }
}
```



```
package lab06.helloMaven;

public class hello {
    public static void main(String[] args) {
        System.out.println("hello world!!");
    }
}
```

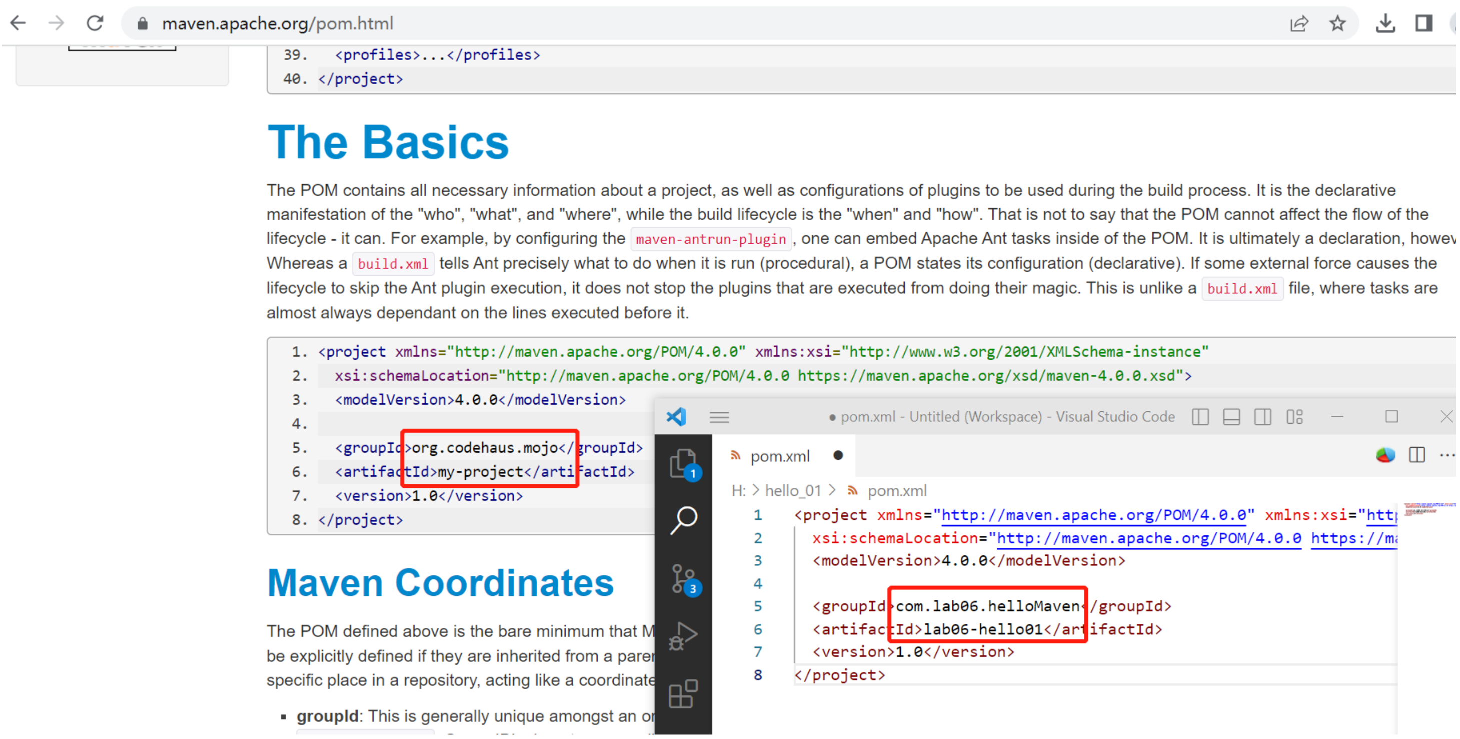
3. Refer official website and modify information:

https://maven.apache.org/pom.html

modelVersion:pom model version.

groupId: This is generally unique amongst an organization or a project.

artifactId: The artifactId is generally the name that the project is known by.


maven.apache.org/pom.html

```
  <profiles>...</profiles>
</project>
```

The Basics

The POM contains all necessary information about a project, as well as configurations of plugins to be used during the build process. It is the declarative manifestation of the "who", "what", and "where", while the build lifecycle is the "when" and "how". That is not to say that the POM cannot affect the flow of the lifecycle - it can. For example, by configuring the `maven-antrun-plugin`, one can embed Apache Ant tasks inside of the POM. It is ultimately a declaration, howev
Whereas a `build.xml` tells Ant precisely what to do when it is run (procedural), a POM states its configuration (declarative). If some external force causes the lifecycle to skip the Ant plugin execution, it does not stop the plugins that are executed from doing their magic. This is unlike a `build.xml` file, where tasks are almost always dependant on the lines executed before it.

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>org.codehaus.mojo</groupId>
  <artifactId>my-project</artifactId>
  <version>1.0</version>
</project>
```

Maven Coordinates

The POM defined above is the bare minimum that M
be explicitly defined if they are inherited from a pare
specific place in a repository, acting like a coordinate

- **groupId**: This is generally unique amongst an o

● pom.xml - Untitled (Workspace) - Visual Studio Code

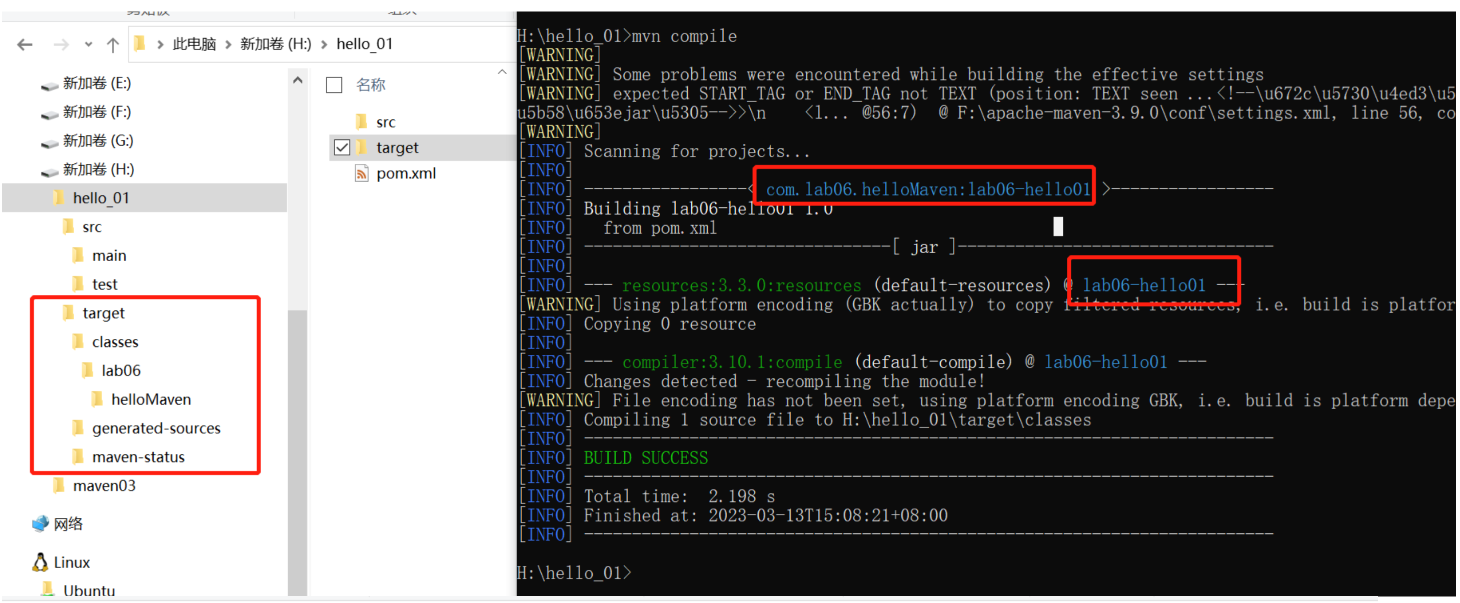
pom.xml

H: > hello_01 > pom.xml

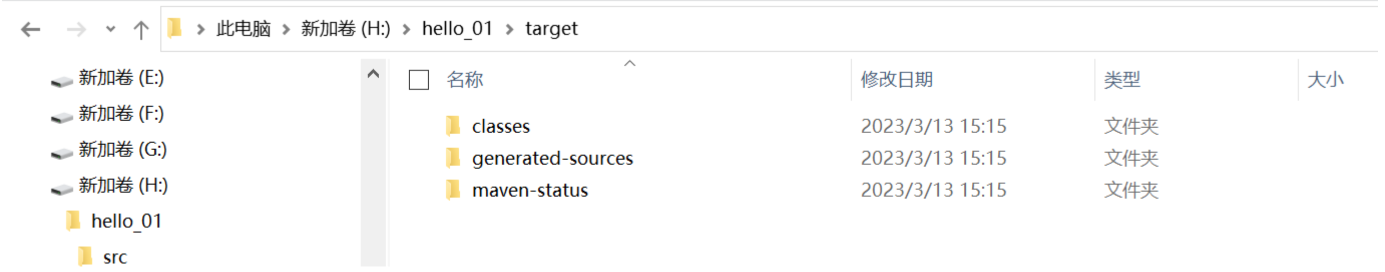
```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://ma
  <modelVersion>4.0.0</modelVersion>

  <groupId>com.lab06.helloMaven</groupId>
  <artifactId>lab06-hello01</artifactId>
  <version>1.0</version>
</project>
```

1. execute command below:

```
mvn compile
```

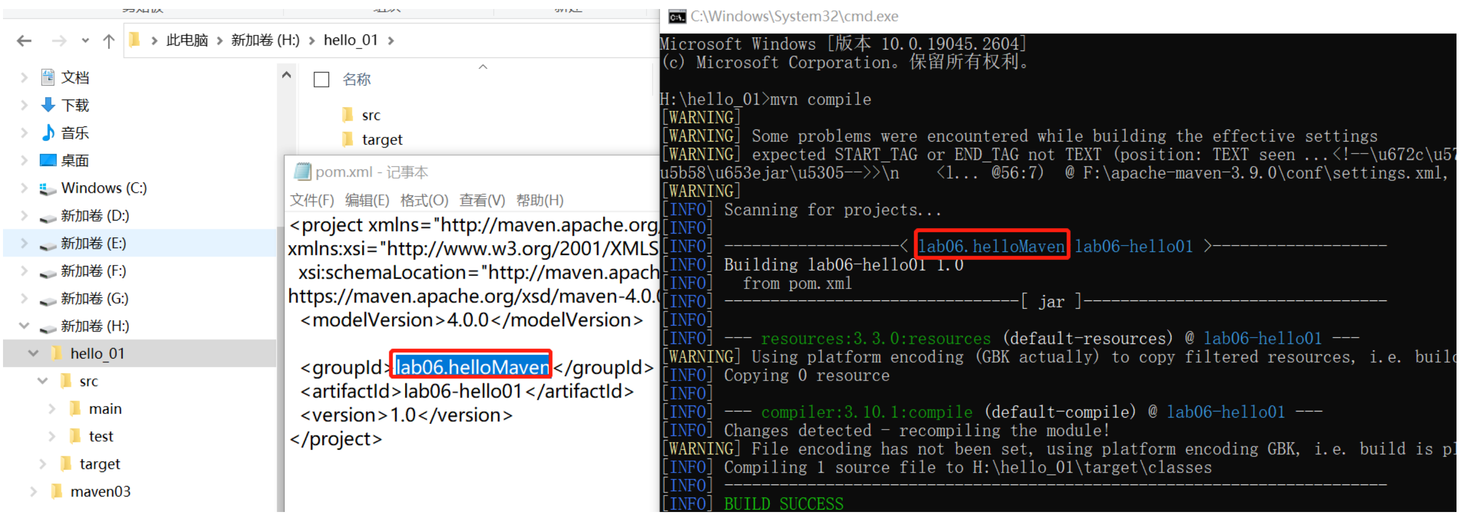


```
H:\hello_01>mvn compile
[WARNING]
[WARNING] Some problems were encountered while building the effective settings
[WARNING] expected START_TAG or END_TAG not TEXT (position: TEXT seen ...<!--\u672c\u5730\u4ed3\u5
u5b58\u653ejar\u5305-->>\n    <1... @56:7)  @ F:\apache-maven-3.9.0\conf\settings.xml, line 56, co
[WARNING]
[INFO] Scanning for projects...
[INFO]
[INFO] -----------------< com.lab06.helloMaven:lab06-hello01 >-----------------
[INFO] Building lab06-hello01 1.0
[INFO]   from pom.xml
[INFO] --------------------------------[ jar ]---------------------------------
[INFO]
[INFO] --- resources:3.3.0:resources (default-resources) @ lab06-hello01 ---
[WARNING] Using platform encoding (GBK actually) to copy filtered resources, i.e. build is platfor
[INFO] Copying 0 resource
[INFO]
[INFO] --- compiler:3.10.1:compile (default-compile) @ lab06-hello01 ---
[INFO] Changes detected - recompiling the module!
[WARNING] File encoding has not been set, using platform encoding GBK, i.e. build is platform depe
[INFO] Compiling 1 source file to H:\hello_01\target\classes
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  2.198 s
[INFO] Finished at: 2023-03-13T15:08:21+08:00
[INFO] ------------------------------------------------------------------------

H:\hello_01>
```


此电脑 > 新加卷 (H:) > hello_01 > target

名称	修改日期	类型	大小
classes	2023/3/13 15:15	文件夹	
generated-sources	2023/3/13 15:15	文件夹	
maven-status	2023/3/13 15:15	文件夹	

if I modify pom.xml:

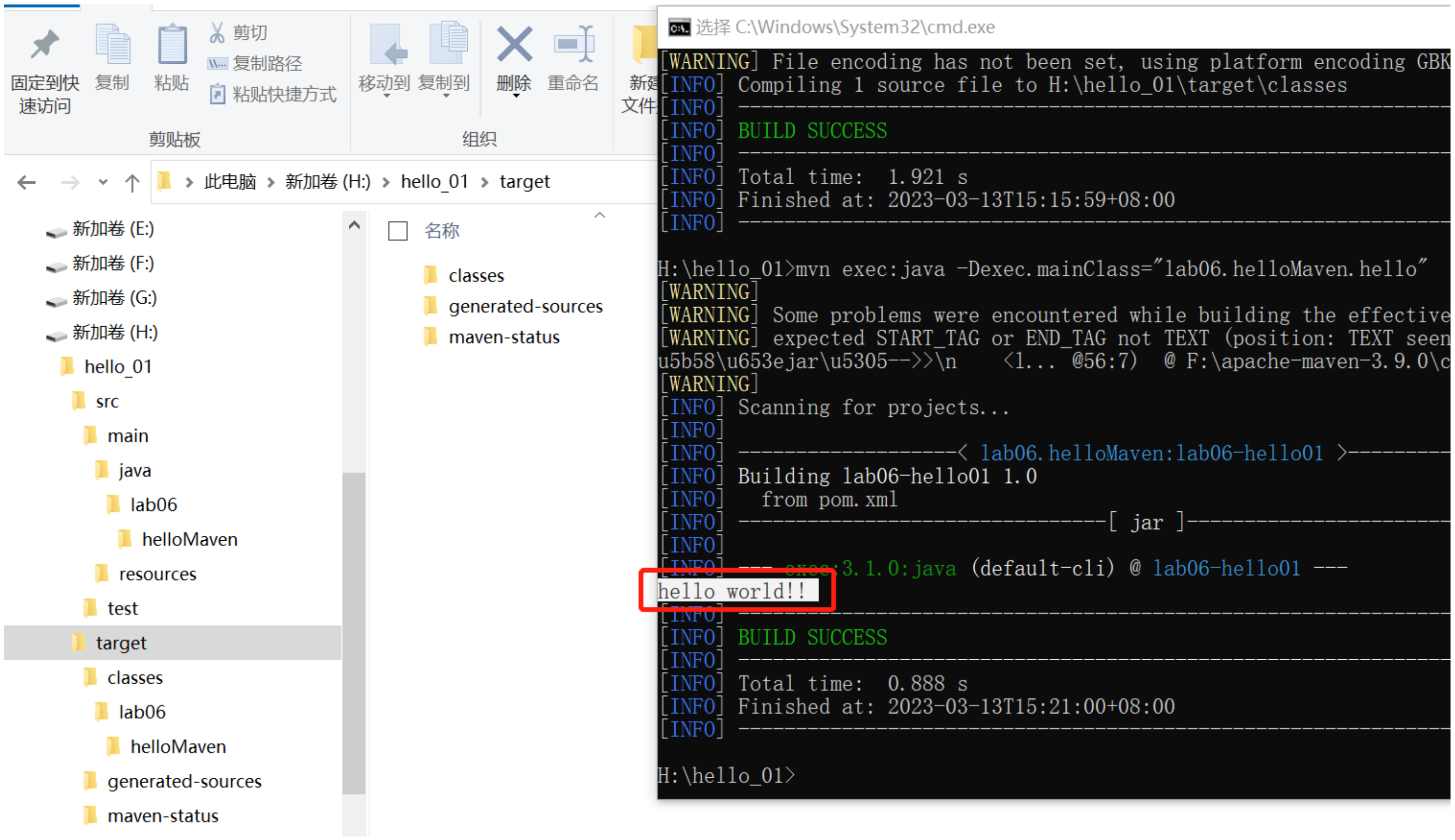


```
<project xmlns="http://maven.apache.org
xmlns:xsi="http://www.w3.org/2001/XMLS
  xsi:schemaLocation="http://maven.apach
https://maven.apache.org/xsd/maven-4.0.
  <modelVersion>4.0.0</modelVersion>

  <groupId>lab06.helloMaven</groupId>
  <artifactId>lab06-hello01</artifactId>
  <version>1.0</version>
</project>
```

```
Microsoft Windows [版本 10.0.19045.2604]
(c) Microsoft Corporation。保留所有权利。

H:\hello_01>mvn compile
[WARNING]
[WARNING] Some problems were encountered while building the effective settings
[WARNING] expected START_TAG or END_TAG not TEXT (position: TEXT seen ...<!--\u672c\u57
u5b58\u653ejar\u5305-->>\n    <1... @56:7)  @ F:\apache-maven-3.9.0\conf\settings.xml,
[WARNING]
[INFO] Scanning for projects...
[INFO]
[INFO] -------------------< lab06.helloMaven lab06-hello01 >-------------------
[INFO] Building lab06-hello01 1.0
[INFO]   from pom.xml
[INFO] --------------------------------[ jar ]---------------------------------
[INFO]
[INFO] --- resources:3.3.0:resources (default-resources) @ lab06-hello01 ---
[WARNING] Using platform encoding (GBK actually) to copy filtered resources, i.e. buil
[INFO] Copying 0 resource
[INFO]
[INFO] --- compiler:3.10.1:compile (default-compile) @ lab06-hello01 ---
[INFO] Changes detected - recompiling the module!
[WARNING] File encoding has not been set, using platform encoding GBK, i.e. build is pl
[INFO] Compiling 1 source file to H:\hello_01\target\classes
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
```

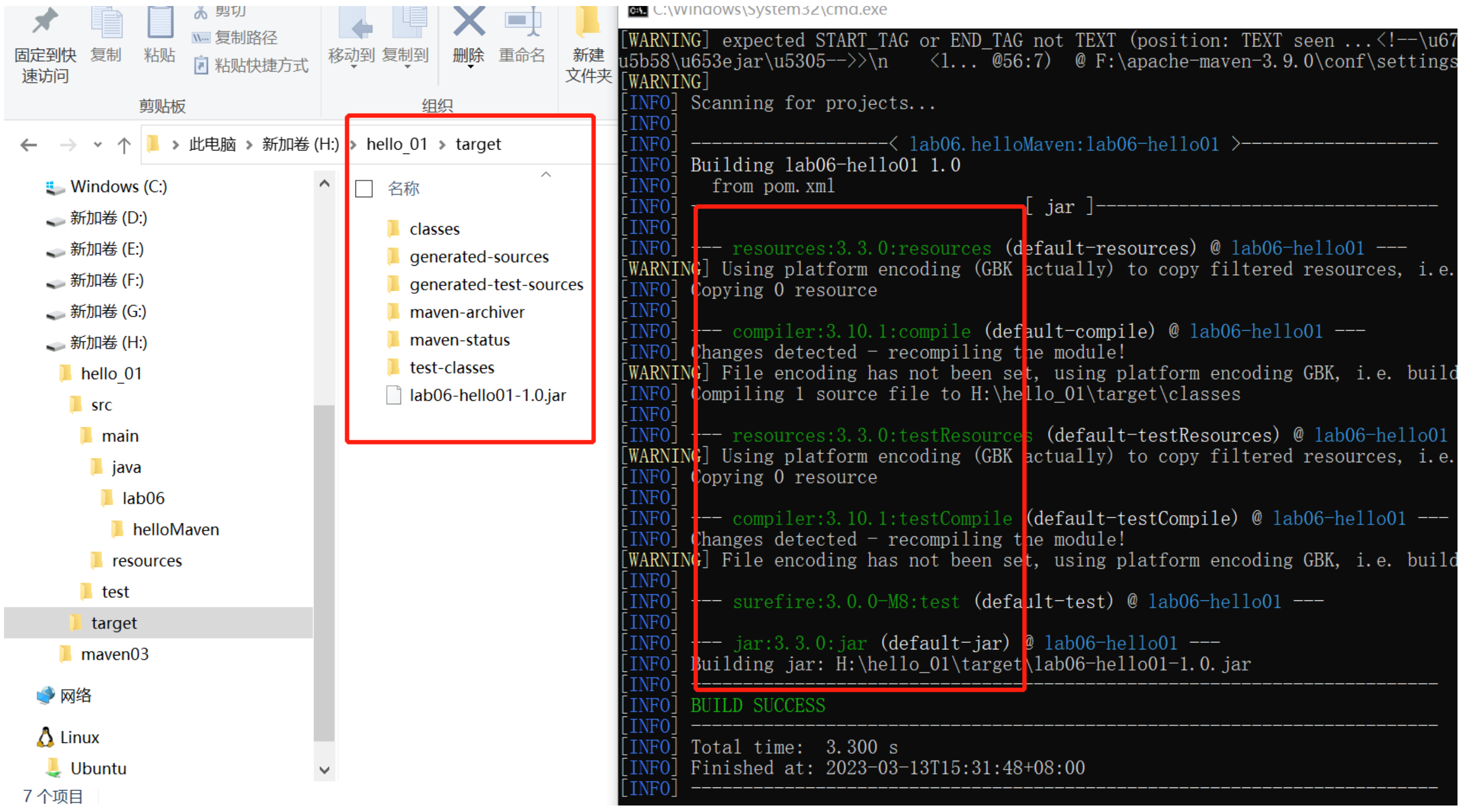
5.Execute java command:

```
mvn exec:java -Dexec.mainClass="lab06.helloMaven.hello"
```

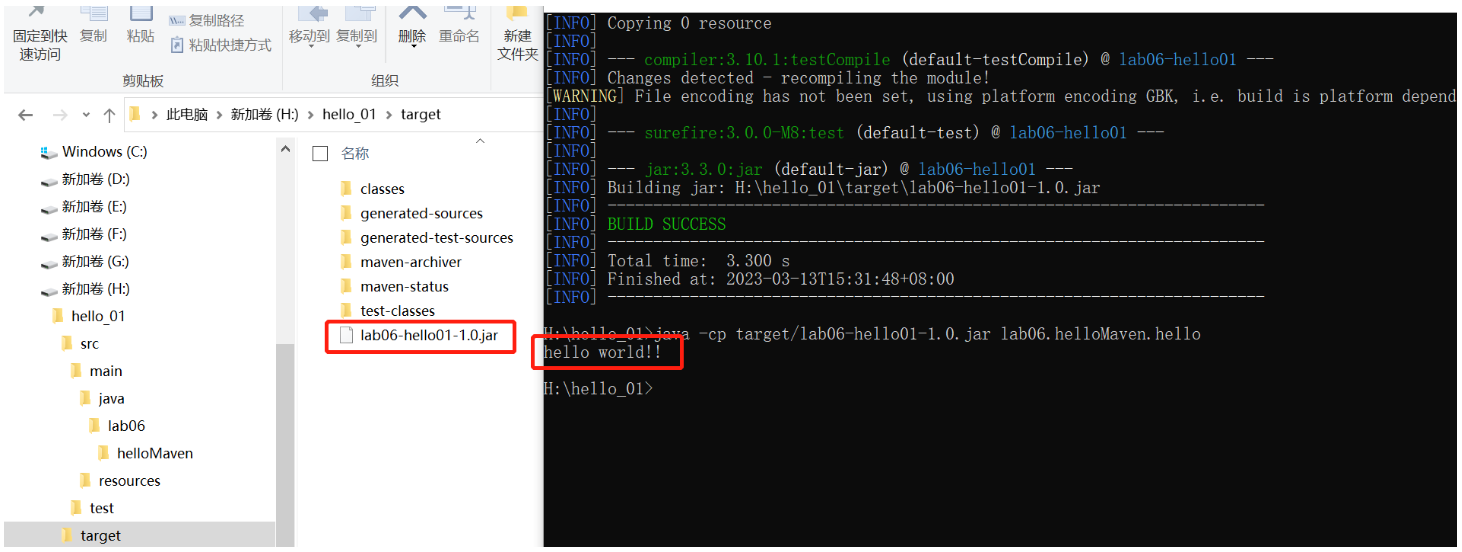


```
[WARNING] File encoding has not been set, using platform encoding GBK
[INFO] Compiling 1 source file to H:\hello_01\target\classes
[INFO] ------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------
[INFO] Total time:  1.921 s
[INFO] Finished at: 2023-03-13T15:15:59+08:00
[INFO] ------------------------------------------------------------

H:\hello_01>mvn exec:java -Dexec.mainClass="lab06.helloMaven.hello"
[WARNING]
[WARNING] Some problems were encountered while building the effective
[WARNING] expected START_TAG or END_TAG not TEXT (position: TEXT seen
u5b58\u653ejar\u5305-->>\n    <1... @56:7)  @ F:\apache-maven-3.9.0\c
[WARNING]
[INFO] Scanning for projects...
[INFO]
[INFO] -------------------< lab06.helloMaven:lab06-hello01 >---------
[INFO] Building lab06-hello01 1.0
[INFO]   from pom.xml
[INFO] --------------------------------[ jar ]-----------------------
[INFO]
[INFO] --- exec:3.1.0:java (default-cli) @ lab06-hello01 ---
hello world!!
[INFO] ------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------
[INFO] Total time:  0.888 s
[INFO] Finished at: 2023-03-13T15:21:00+08:00
[INFO] ------------------------------------------------------------

H:\hello_01>
```

If I use package command:



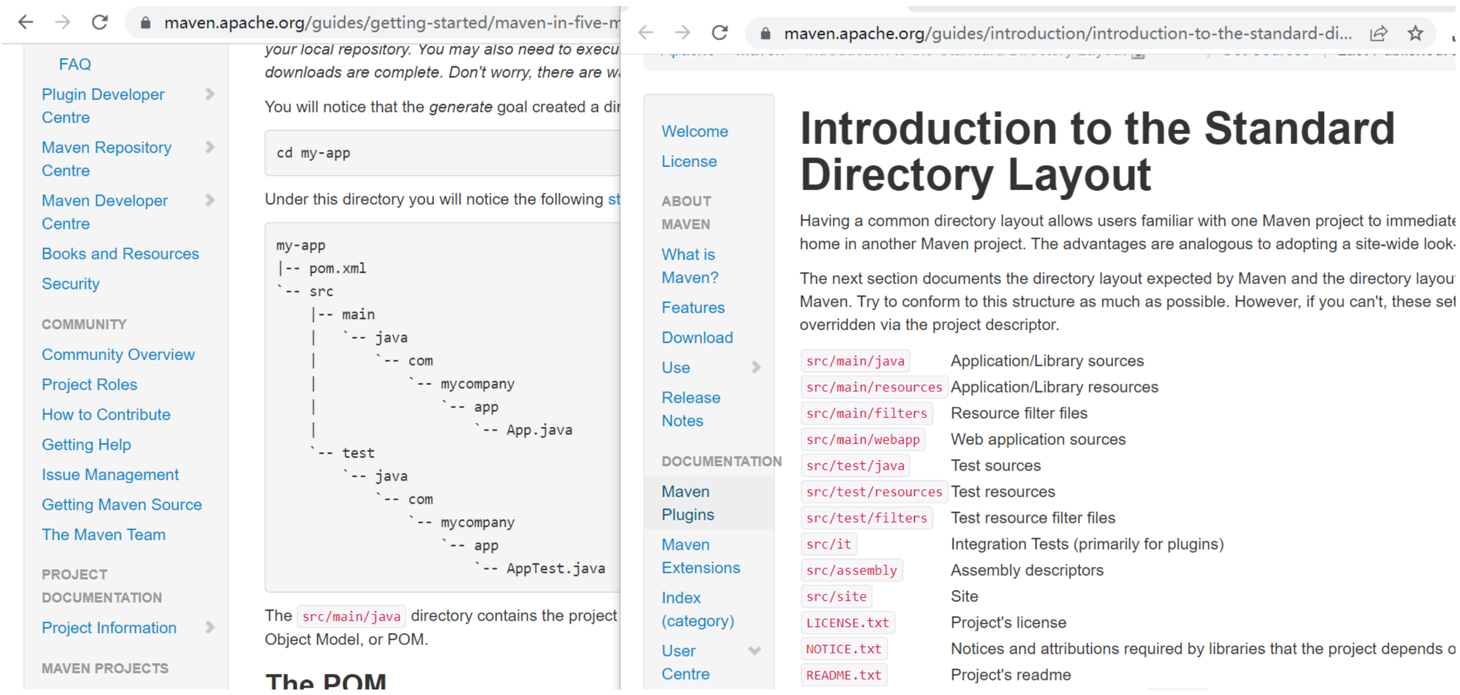
```
[WARNING] expected START_TAG or END_TAG not TEXT (position: TEXT seen ...<!--\u67
u5b58\u653ejar\u5305-->>\n    <1... @56:7)  @ F:\apache-maven-3.9.0\conf\settings
[WARNING]
[INFO] Scanning for projects...
[INFO]
[INFO] ------------------< lab06.helloMaven:lab06-hello01 >------------------
[INFO] Building lab06-hello01 1.0
[INFO]   from pom.xml
[INFO] --------------------------------[ jar ]---------------------------------
[INFO]
[INFO] --- resources:3.3.0:resources (default-resources) @ lab06-hello01 ---
[WARNING] Using platform encoding (GBK actually) to copy filtered resources, i.e.
[INFO] Copying 0 resource
[INFO]
[INFO] --- compiler:3.10.1:compile (default-compile) @ lab06-hello01 ---
[INFO] Changes detected - recompiling the module!
[WARNING] File encoding has not been set, using platform encoding GBK, i.e. build
[INFO] Compiling 1 source file to H:\hello_01\target\classes
[INFO]
[INFO] --- resources:3.3.0:testResources (default-testResources) @ lab06-hello01
[WARNING] Using platform encoding (GBK actually) to copy filtered resources, i.e.
[INFO] Copying 0 resource
[INFO]
[INFO] --- compiler:3.10.1:testCompile (default-testCompile) @ lab06-hello01 ---
[INFO] Changes detected - recompiling the module!
[WARNING] File encoding has not been set, using platform encoding GBK, i.e. build
[INFO]
[INFO] --- surefire:3.0.0-M8:test (default-test) @ lab06-hello01 ---
[INFO]
[INFO] --- jar:3.3.0:jar (default-jar) @ lab06-hello01 ---
[INFO] Building jar: H:\hello_01\target\lab06-hello01-1.0.jar
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  3.300 s
[INFO] Finished at: 2023-03-13T15:31:48+08:00
[INFO] ------------------------------------------------------------------------
```



```
[INFO] Copying 0 resource
[INFO]
[INFO] --- compiler:3.10.1:testCompile (default-testCompile) @ lab06-hello01 ---
[INFO] Changes detected - recompiling the module!
[WARNING] File encoding has not been set, using platform encoding GBK, i.e. build is platform depend
[INFO]
[INFO] --- surefire:3.0.0-M8:test (default-test) @ lab06-hello01 ---
[INFO]
[INFO] --- jar:3.3.0:jar (default-jar) @ lab06-hello01 ---
[INFO] Building jar: H:\hello_01\target\lab06-hello01-1.0.jar
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  3.300 s
[INFO] Finished at: 2023-03-13T15:31:48+08:00
[INFO] ------------------------------------------------------------------------

H:\hello_01>java -cp target/lab06-hello01-1.0.jar lab06.helloMaven.hello
hello world!!

H:\hello_01>
```

2. Simple project directory Automatic creation

Maven has its own standard Directory Layout:
https://maven.apache.org/guides/getting-started/maven-in-five-minutes.html
https://maven.apache.org/guides/introduction/introduction-to-the-standard-directory-layout.html



your local repository. You may also need to execu
downloads are complete. Don't worry, there are w

You will notice that the *generate* goal created a di

```
cd my-app
```

Under this directory you will notice the following st

```
my-app
|-- pom.xml
`-- src
    |-- main
    |   `-- java
    |       `-- com
    |           `-- mycompany
    |               `-- app
    |                   `-- App.java
    `-- test
        `-- java
            `-- com
                `-- mycompany
                    `-- app
                        `-- AppTest.java
```

The `src/main/java` directory contains the project
Object Model, or POM.

The POM

Introduction to the Standard Directory Layout

Having a common directory layout allows users familiar with one Maven project to immediate
home in another Maven project. The advantages are analogous to adopting a site-wide look-

The next section documents the directory layout expected by Maven and the directory layou
Maven. Try to conform to this structure as much as possible. However, if you can't, these set
overridden via the project descriptor.

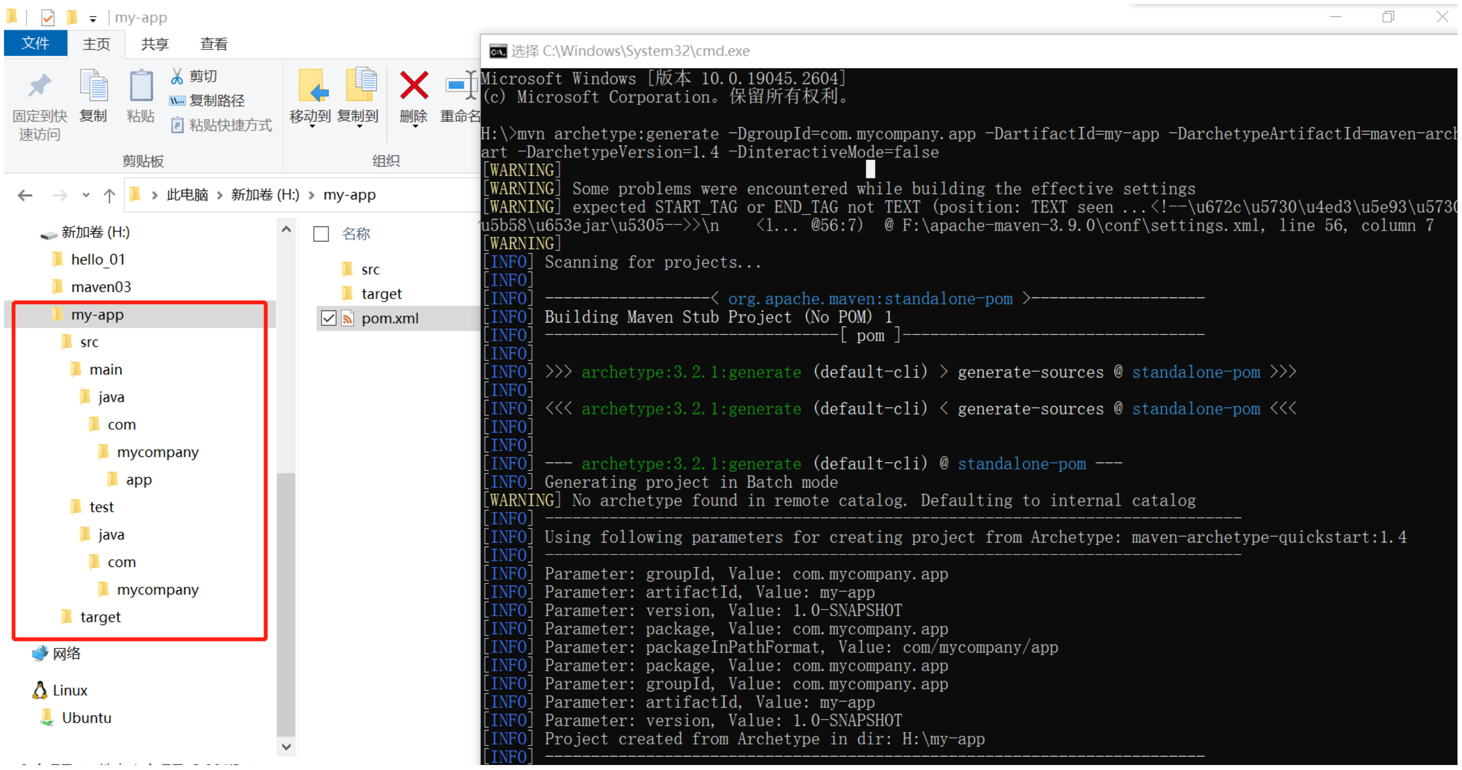
`src/main/java`	Application/Library sources
`src/main/resources`	Application/Library resources
`src/main/filters`	Resource filter files
`src/main/webapp`	Web application sources
`src/test/java`	Test sources
`src/test/resources`	Test resources
`src/test/filters`	Test resource filter files
`src/it`	Integration Tests (primarily for plugins)
`src/assembly`	Assembly descriptors
`src/site`	Site
`LICENSE.txt`	Project's license
`NOTICE.txt`	Notices and attributions required by libraries that the project depends o
`README.txt`	Project's readme

```
mvn archetype:generate -DgroupId=com.mycompany.app -DartifactId=my-app -DarchetypeArtifactId=maven-archetype-quickstart -DarchetypeVersion=1.4 -DinteractiveMode=false
```

We use this command to create a dirctory and start it .

This archetype:generate goal created a simple project based upon a maven-archetype-quickstart archetype.

(groupId:com.mycompany.app will create the path：/com/mycompany/app)

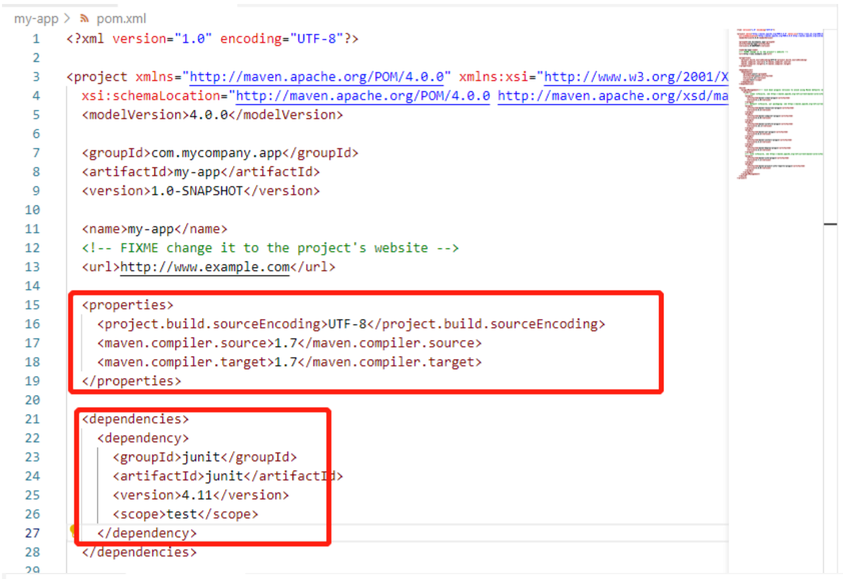


```
Microsoft Windows [版本 10.0.19045.2604]
(c) Microsoft Corporation。保留所有权利。

H:\>mvn archetype:generate -DgroupId=com.mycompany.app -DartifactId=my-app -DarchetypeArtifactId=maven-arch
art -DarchetypeVersion=1.4 -DinteractiveMode=false
[WARNING]
[WARNING] Some problems were encountered while building the effective settings
[WARNING] expected START_TAG or END_TAG not TEXT (position: TEXT seen ...<!--\u672c\u5730\u4ed3\u5e93\u5730
u5b58\u653ejar\u5305-->>\n    <1... @56:7)  @ F:\apache-maven-3.9.0\conf\settings.xml, line 56, column 7
[WARNING]
[INFO] Scanning for projects...
[INFO]
[INFO] ------------------< org.apache.maven:standalone-pom >-------------------
[INFO] Building Maven Stub Project (No POM) 1
[INFO] --------------------------------[ pom ]---------------------------------
[INFO]
[INFO] >>> archetype:3.2.1:generate (default-cli) > generate-sources @ standalone-pom >>>
[INFO]
[INFO] <<< archetype:3.2.1:generate (default-cli) < generate-sources @ standalone-pom <<<
[INFO]
[INFO]
[INFO] --- archetype:3.2.1:generate (default-cli) @ standalone-pom ---
[INFO] Generating project in Batch mode
[WARNING] No archetype found in remote catalog. Defaulting to internal catalog
[INFO] ----------------------------------------------------------------------------
[INFO] Using following parameters for creating project from Archetype: maven-archetype-quickstart:1.4
[INFO] ----------------------------------------------------------------------------
[INFO] Parameter: groupId, Value: com.mycompany.app
[INFO] Parameter: artifactId, Value: my-app
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[INFO] Parameter: package, Value: com.mycompany.app
[INFO] Parameter: packageInPathFormat, Value: com/mycompany/app
[INFO] Parameter: package, Value: com.mycompany.app
[INFO] Parameter: groupId, Value: com.mycompany.app
[INFO] Parameter: artifactId, Value: my-app
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[INFO] Project created from Archetype in dir: H:\my-app
[INFO] ------------------------------------------------------------------------
```

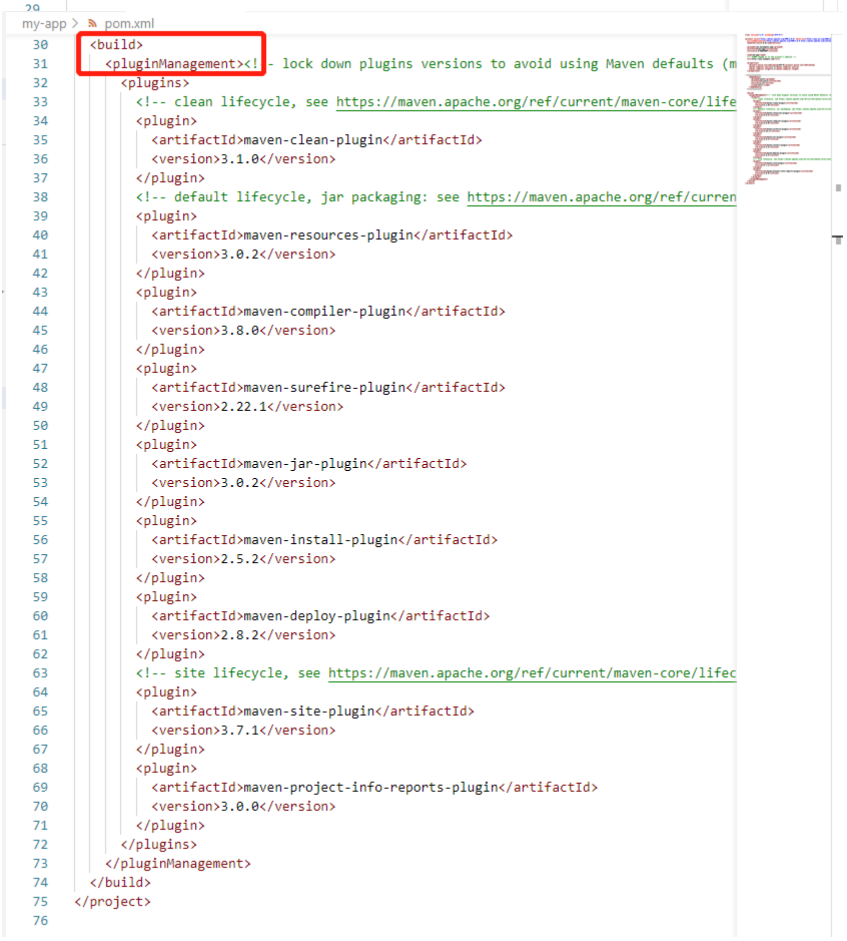
Then we can see the pom.xml:

Compare with previous xml,there are more information.

> Dependency management is a core feature of Maven. We can find more information by
> https://maven.apache.org/guides/introduction/introduction-to-dependency-mechanism.html
> Maven has two kinds of plugins build and reporting,are executed during the build and during the site generation.
> properties like a label,you can get plugin and dependency version by property.



```xml
<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/X
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/ma
  <modelVersion>4.0.0</modelVersion>

  <groupId>com.mycompany.app</groupId>
  <artifactId>my-app</artifactId>
  <version>1.0-SNAPSHOT</version>

  <name>my-app</name>
  <!-- FIXME change it to the project's website -->
  <url>http://www.example.com</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
  </properties>

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

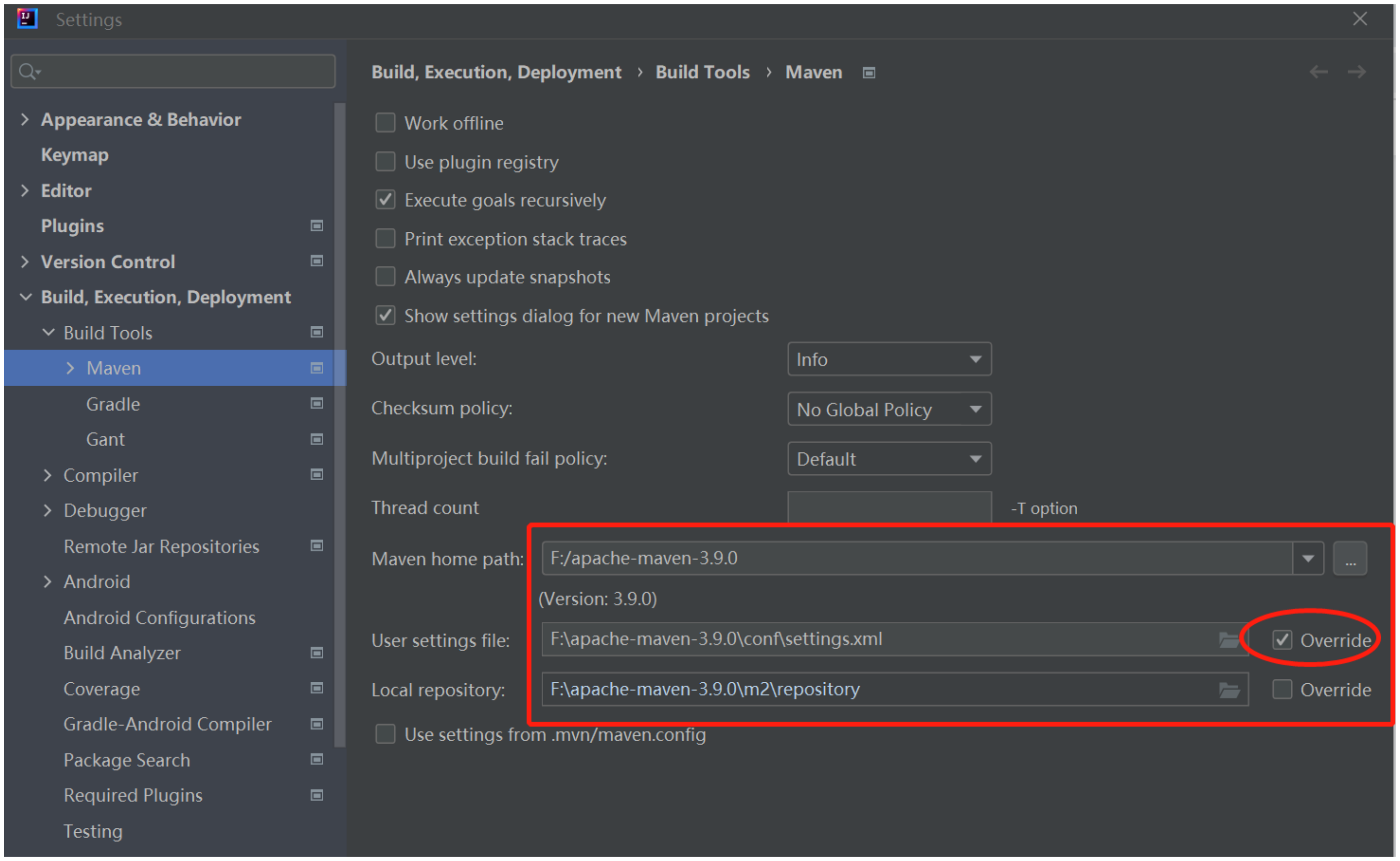


```xml
  <build>
    <pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (m
      <plugins>
        <!-- clean lifecycle, see https://maven.apache.org/ref/current/maven-core/life
        <plugin>
          <artifactId>maven-clean-plugin</artifactId>
          <version>3.1.0</version>
        </plugin>
        <!-- default lifecycle, jar packaging: see https://maven.apache.org/ref/curren
        <plugin>
          <artifactId>maven-resources-plugin</artifactId>
          <version>3.0.2</version>
        </plugin>
        <plugin>
          <artifactId>maven-compiler-plugin</artifactId>
          <version>3.8.0</version>
        </plugin>
        <plugin>
          <artifactId>maven-surefire-plugin</artifactId>
          <version>2.22.1</version>
        </plugin>
        <plugin>
          <artifactId>maven-jar-plugin</artifactId>
          <version>3.0.2</version>
        </plugin>
        <plugin>
          <artifactId>maven-install-plugin</artifactId>
          <version>2.5.2</version>
        </plugin>
        <plugin>
          <artifactId>maven-deploy-plugin</artifactId>
          <version>2.8.2</version>
        </plugin>
        <!-- site lifecycle, see https://maven.apache.org/ref/current/maven-core/lifec
        <plugin>
          <artifactId>maven-site-plugin</artifactId>
          <version>3.7.1</version>
        </plugin>
        <plugin>
          <artifactId>maven-project-info-reports-plugin</artifactId>
          <version>3.0.0</version>
        </plugin>
      </plugins>
    </pluginManagement>
  </build>
</project>
```

Part 3 Run a simple Maven JavaWeb project with IDEA

Let's config IDEA first:

File——new project setup——settings for new project(different version has different path):



Build, Execution, Deployment › Build Tools › Maven

- [ ] Work offline
- [ ] Use plugin registry
- [✓] Execute goals recursively
- [ ] Print exception stack traces
- [ ] Always update snapshots
- [✓] Show settings dialog for new Maven projects

Output level: Info

Checksum policy: No Global Policy

Multiproject build fail policy: Default

Thread count: -T option

Maven home path: F:/apache-maven-3.9.0

(Version: 3.9.0)

User settings file: F:\apache-maven-3.9.0\conf\settings.xml [✓] Override

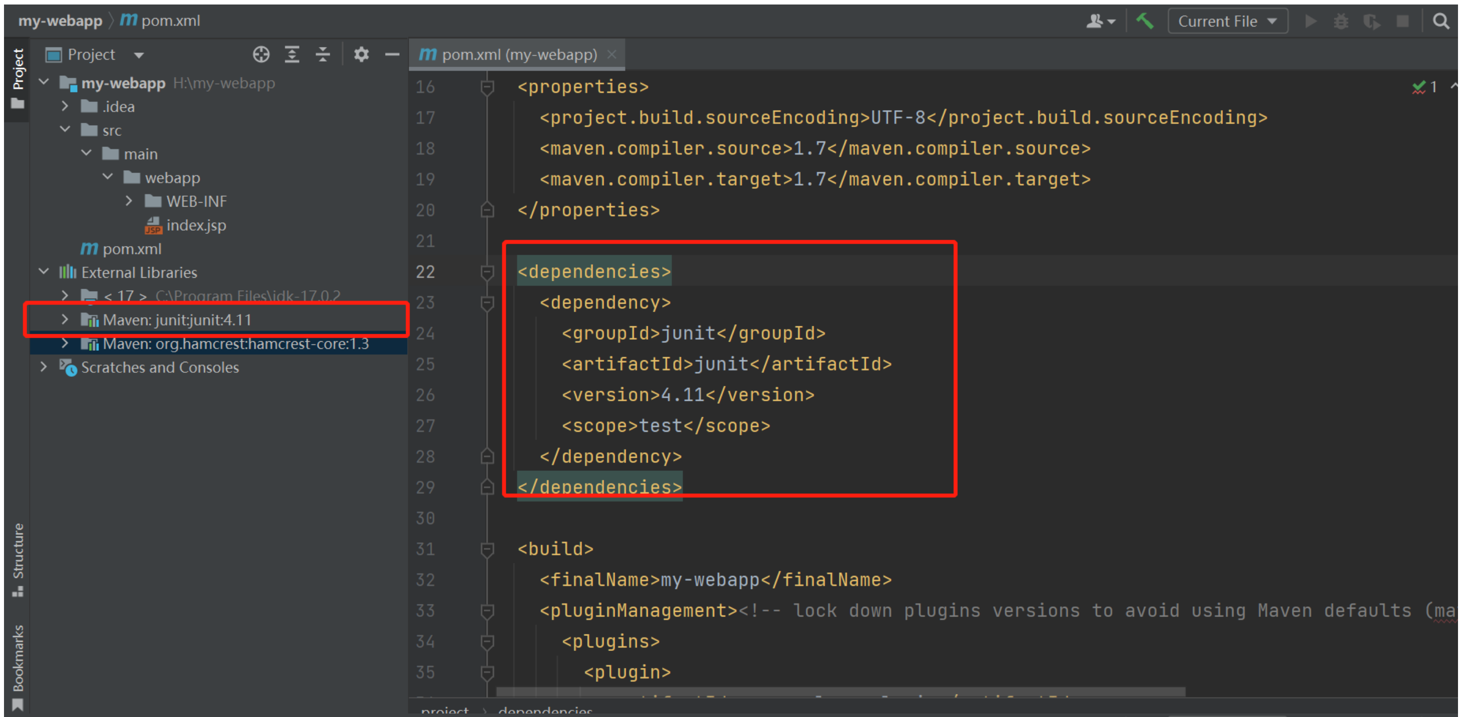
Local repository: F:\apache-maven-3.9.0\m2\repository [ ] Override

- [ ] Use settings from .mvn/maven.config

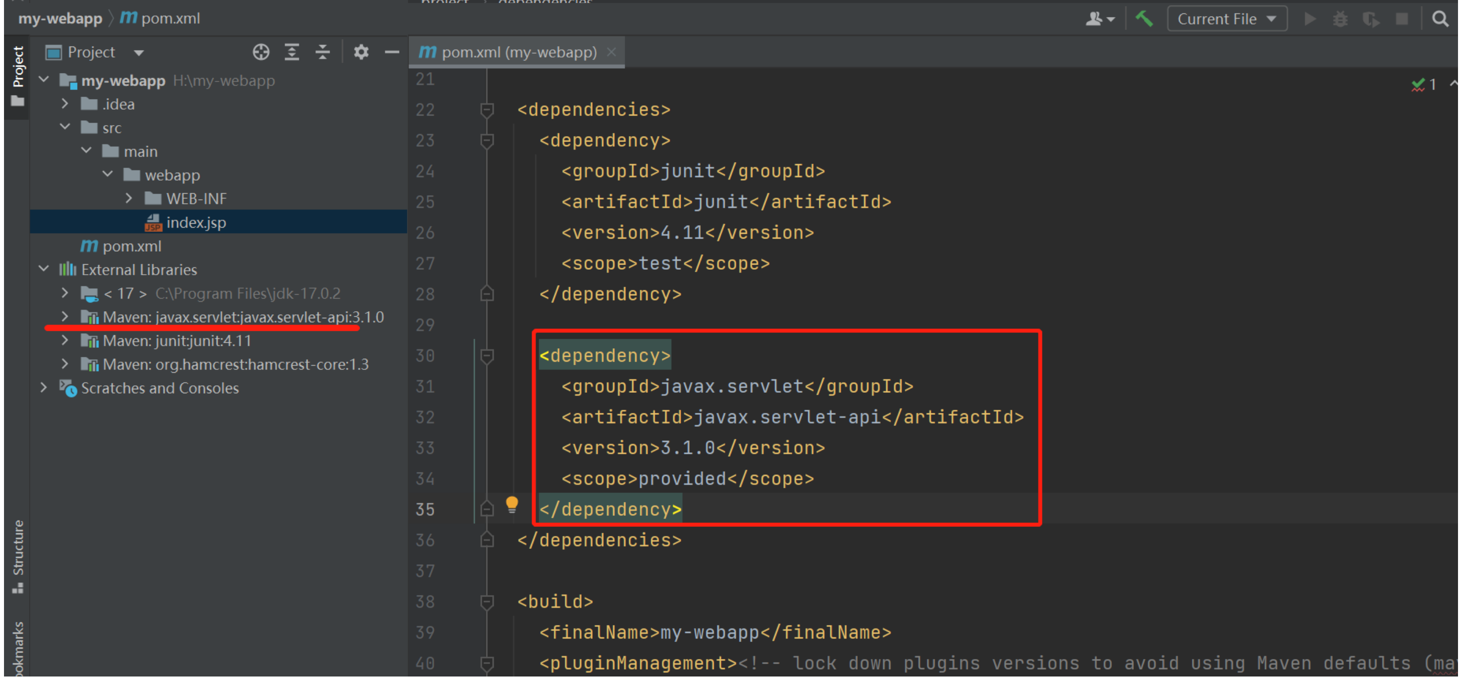
command to create a dirctory

```
mvn archetype:generate -DgroupId=com.mycompany.webapp -DartifactId=my-webapp -DarchetypeArtifactId=maven-archetype-webapp -DarchetypeVersion=1.4 -DinteractiveMode=false
```

Open with IDEA , Try to add a dependency:



```
<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  <maven.compiler.source>1.7</maven.compiler.source>
  <maven.compiler.target>1.7</maven.compiler.target>
</properties>

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

<build>
  <finalName>my-webapp</finalName>
  <pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (ma
    <plugins>
      <plugin>
```



```

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

  <dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>javax.servlet-api</artifactId>
    <version>3.1.0</version>
    <scope>provided</scope>
  </dependency>
</dependencies>

<build>
  <finalName>my-webapp</finalName>
  <pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (ma
```

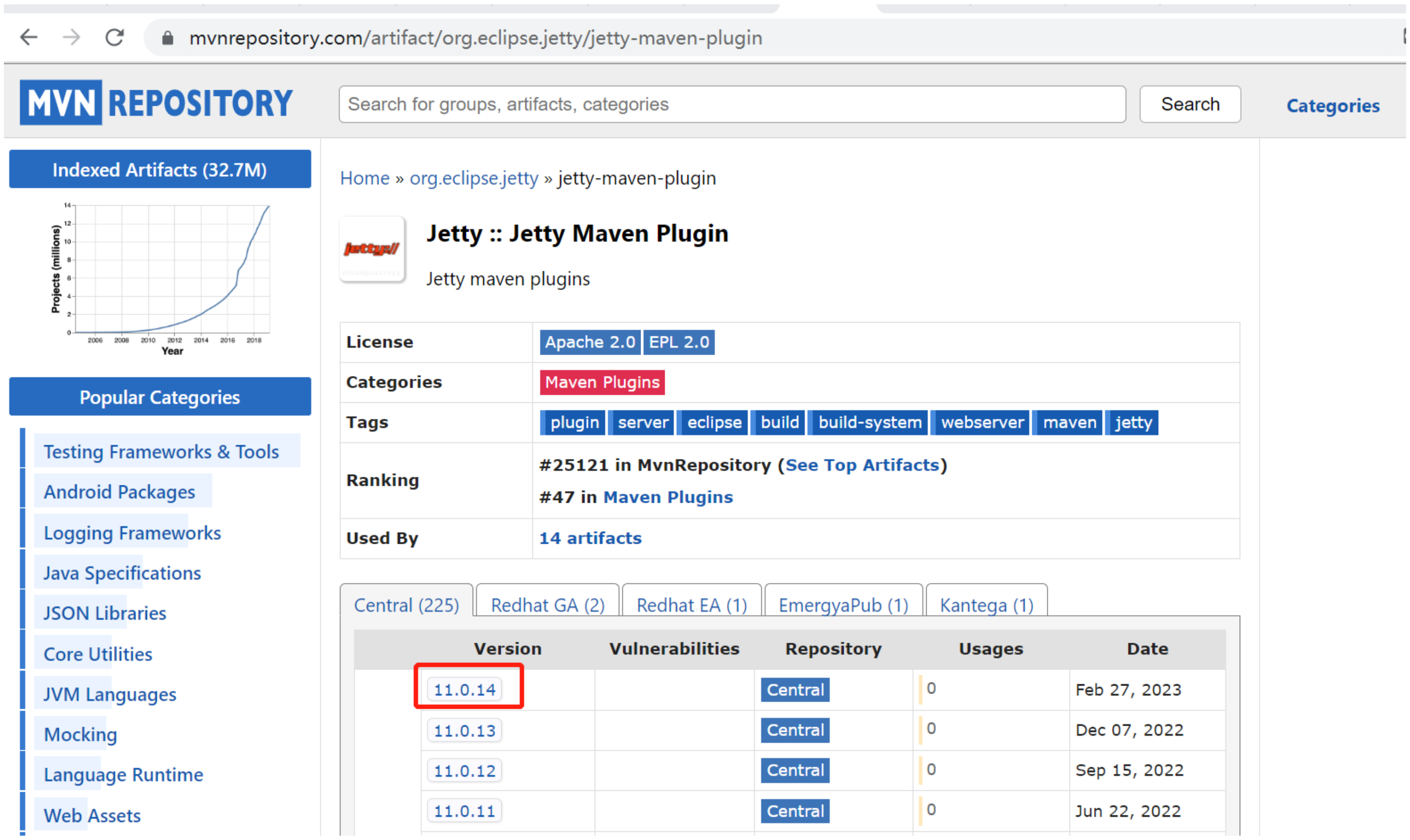
Add jetty server in build:

add jetty run command:

```
<plugin>
  <groupId>org.eclipse.jetty</groupId>
  <artifactId>jetty-maven-plugin</artifactId>
  <version>11.0.14</version>
</plugin>
```

Where can I find this config:

https://mvnrepository.com/


mvnrepository.com/artifact/org.eclipse.jetty/jetty-maven-plugin

Home » org.eclipse.jetty » jetty-maven-plugin

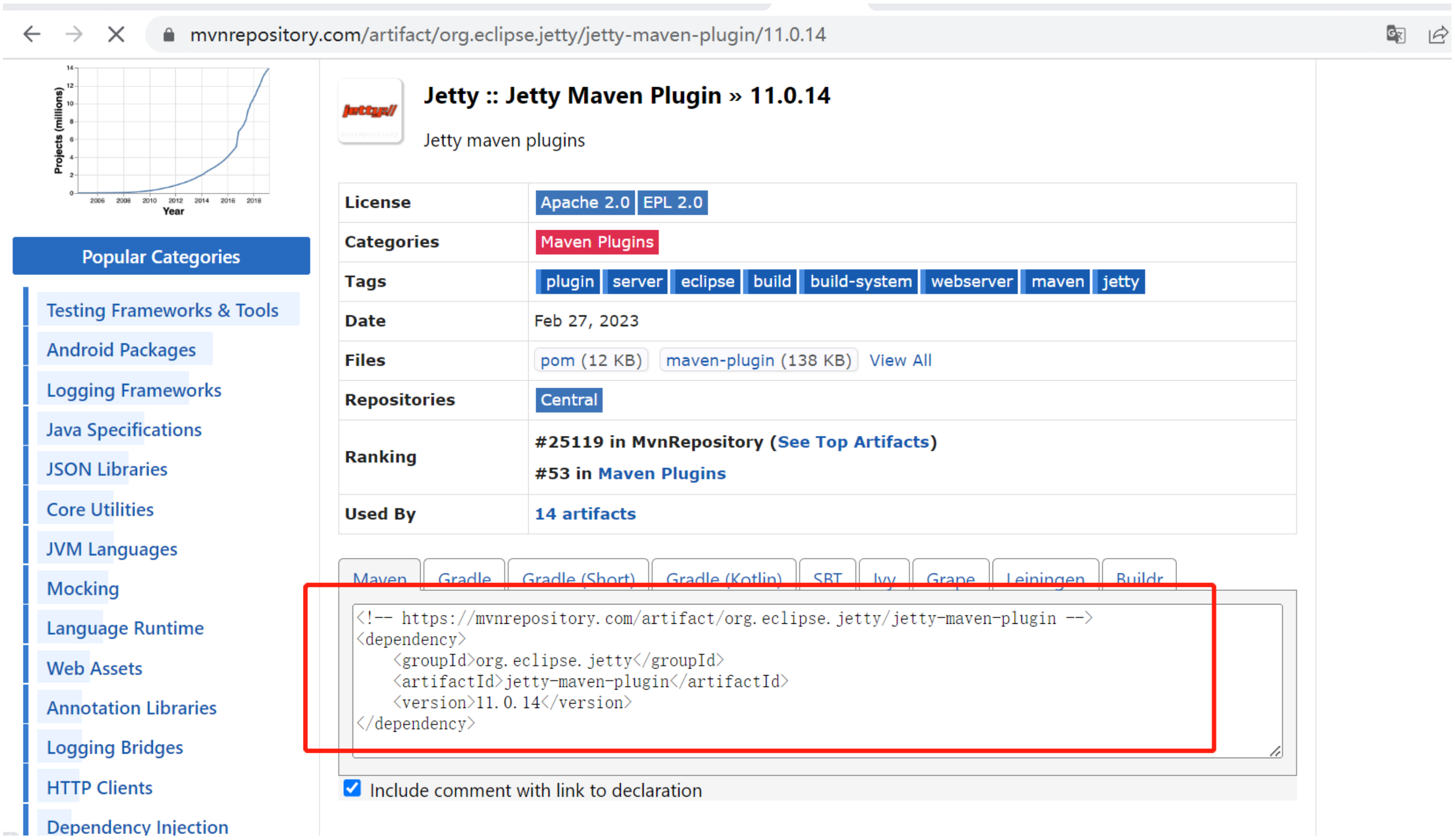
Jetty :: Jetty Maven Plugin

Jetty maven plugins

License	Apache 2.0 EPL 2.0
Categories	Maven Plugins
Tags	plugin server eclipse build build-system webserver maven jetty
Ranking	#25121 in MvnRepository (See Top Artifacts) #47 in Maven Plugins
Used By	14 artifacts

Central (225) | Redhat GA (2) | Redhat EA (1) | EmergyaPub (1) | Kantega (1)

Version	Vulnerabilities	Repository	Usages	Date
11.0.14		Central	0	Feb 27, 2023
11.0.13		Central	0	Dec 07, 2022
11.0.12		Central	0	Sep 15, 2022
11.0.11		Central	0	Jun 22, 2022


mvnrepository.com/artifact/org.eclipse.jetty/jetty-maven-plugin/11.0.14

Jetty :: Jetty Maven Plugin » 11.0.14

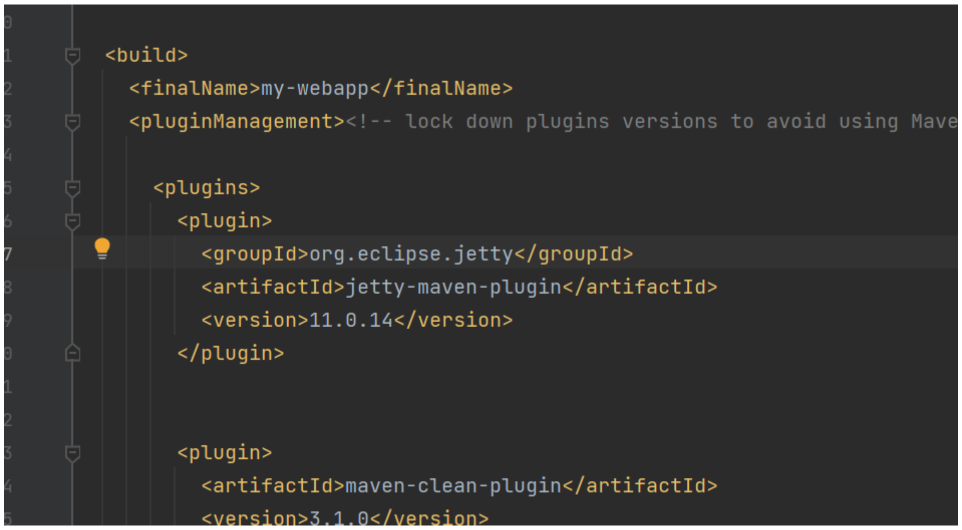
Jetty maven plugins

License	Apache 2.0 EPL 2.0
Categories	Maven Plugins
Tags	plugin server eclipse build build-system webserver maven jetty
Date	Feb 27, 2023
Files	pom (12 KB) maven-plugin (138 KB) View All
Repositories	Central
Ranking	#25119 in MvnRepository (See Top Artifacts) #53 in Maven Plugins
Used By	14 artifacts

Maven | Gradle | Gradle (Short) | Gradle (Kotlin) | SBT | Ivy | Grape | Leiningen | Buildr

```
<!-- https://mvnrepository.com/artifact/org.eclipse.jetty/jetty-maven-plugin -->
<dependency>
    <groupId>org.eclipse.jetty</groupId>
    <artifactId>jetty-maven-plugin</artifactId>
    <version>11.0.14</version>
</dependency>
```

☑ Include comment with link to declaration

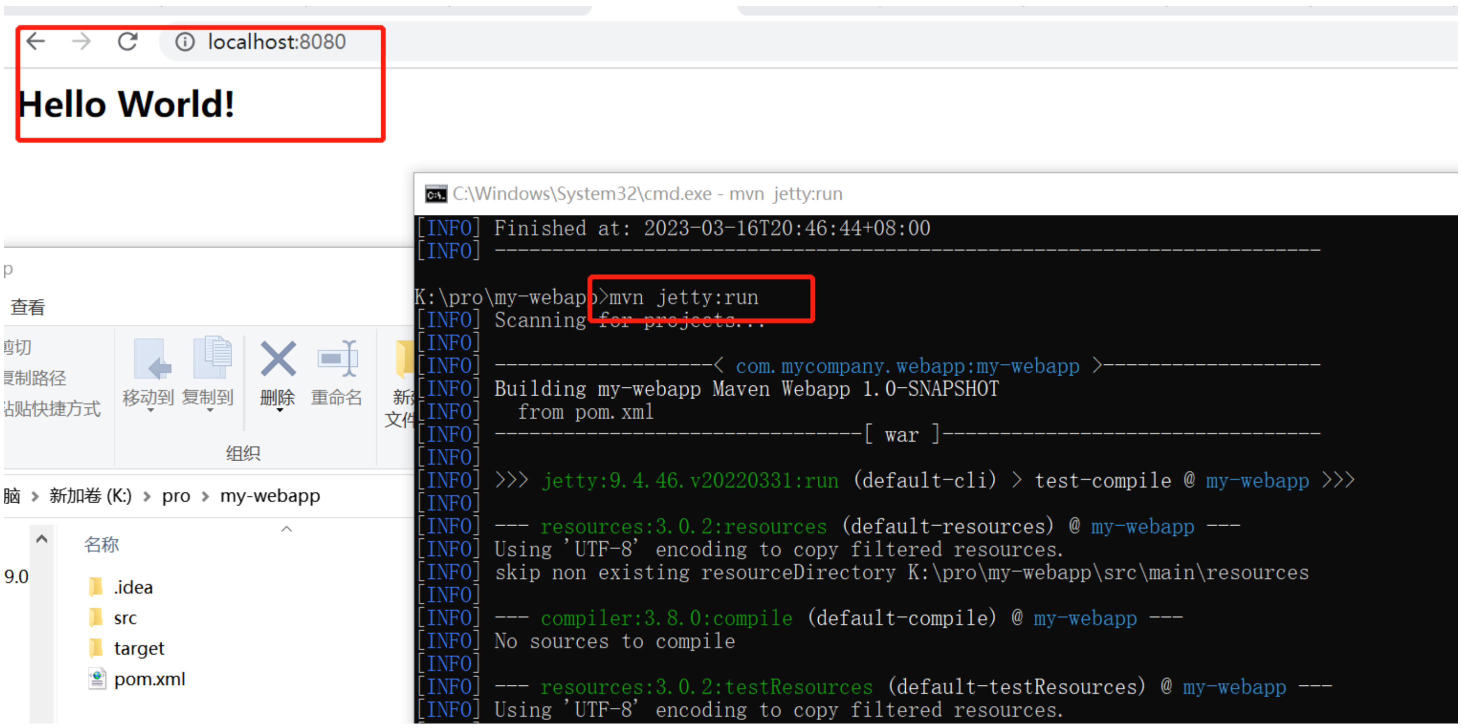


```
<build>
  <finalName>my-webapp</finalName>
  <pluginManagement><!-- lock down plugins versions to avoid using Mave

    <plugins>
      <plugin>
        <groupId>org.eclipse.jetty</groupId>
        <artifactId>jetty-maven-plugin</artifactId>
        <version>11.0.14</version>
      </plugin>


      <plugin>
        <artifactId>maven-clean-plugin</artifactId>
        <version>3.1.0</version>
```

compile and run jetty server:

1.by command:

```
mvn install
mvn jetty:run
```



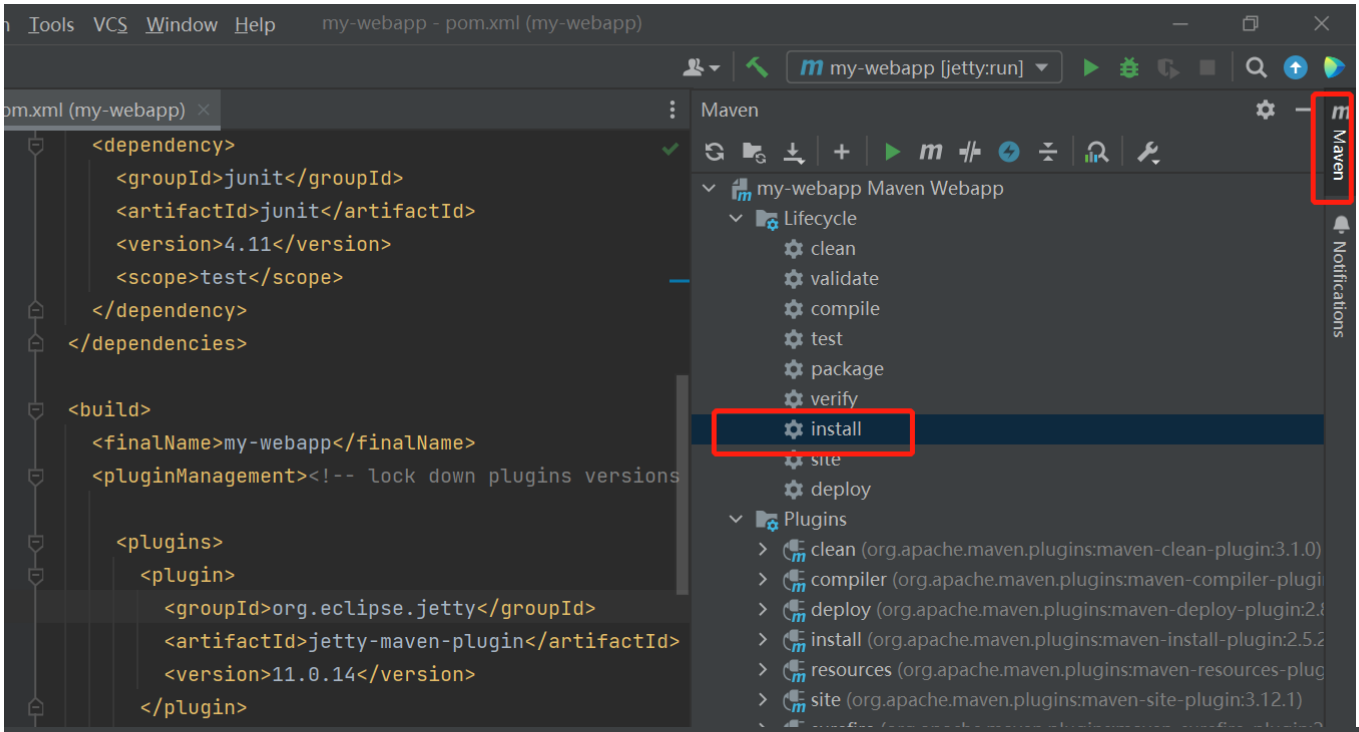
localhost:8080

Hello World!

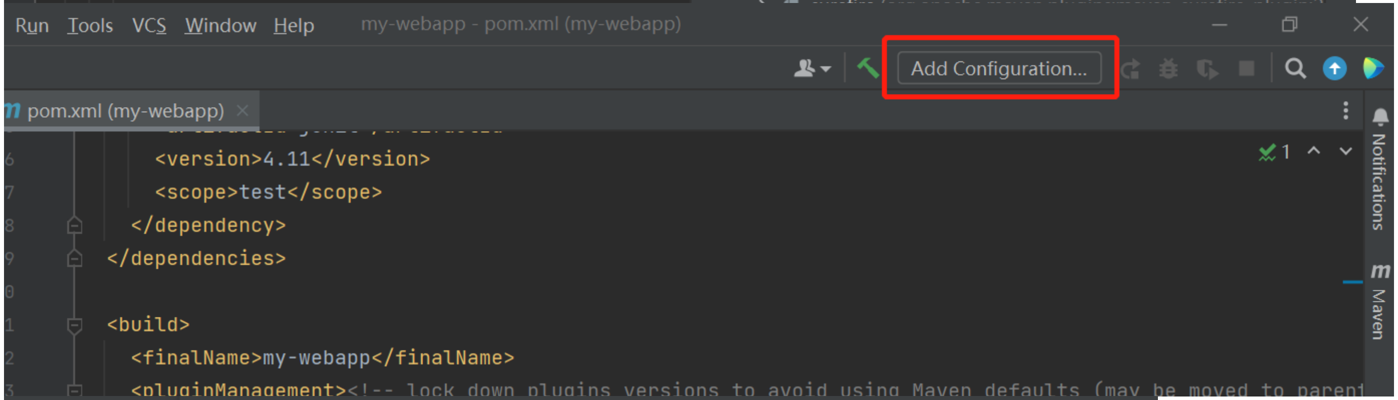
C:\Windows\System32\cmd.exe - mvn jetty:run

```
[INFO] Finished at: 2023-03-16T20:46:44+08:00
[INFO] ------------------------------------------------------------------------

K:\pro\my-webapp>mvn jetty:run
[INFO] Scanning for projects...
[INFO]
[INFO] -------------------< com.mycompany.webapp:my-webapp >-------------------
[INFO] Building my-webapp Maven Webapp 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] --------------------------------[ war ]---------------------------------
[INFO]
[INFO] >>> jetty:9.4.46.v20220331:run (default-cli) > test-compile @ my-webapp >>>
[INFO]
[INFO] --- resources:3.0.2:resources (default-resources) @ my-webapp ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory K:\pro\my-webapp\src\main\resources
[INFO]
[INFO] --- compiler:3.8.0:compile (default-compile) @ my-webapp ---
[INFO] No sources to compile
[INFO]
[INFO] --- resources:3.0.2:testResources (default-testResources) @ my-webapp ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
```

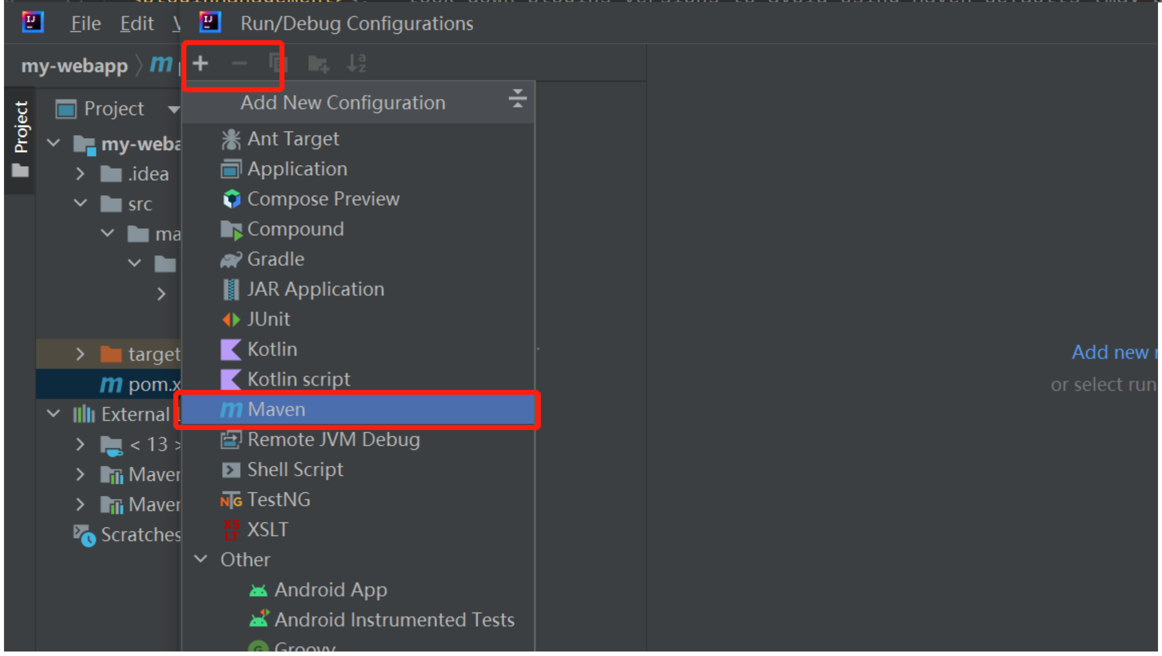
2.by IDEA(you can add command to configuration):

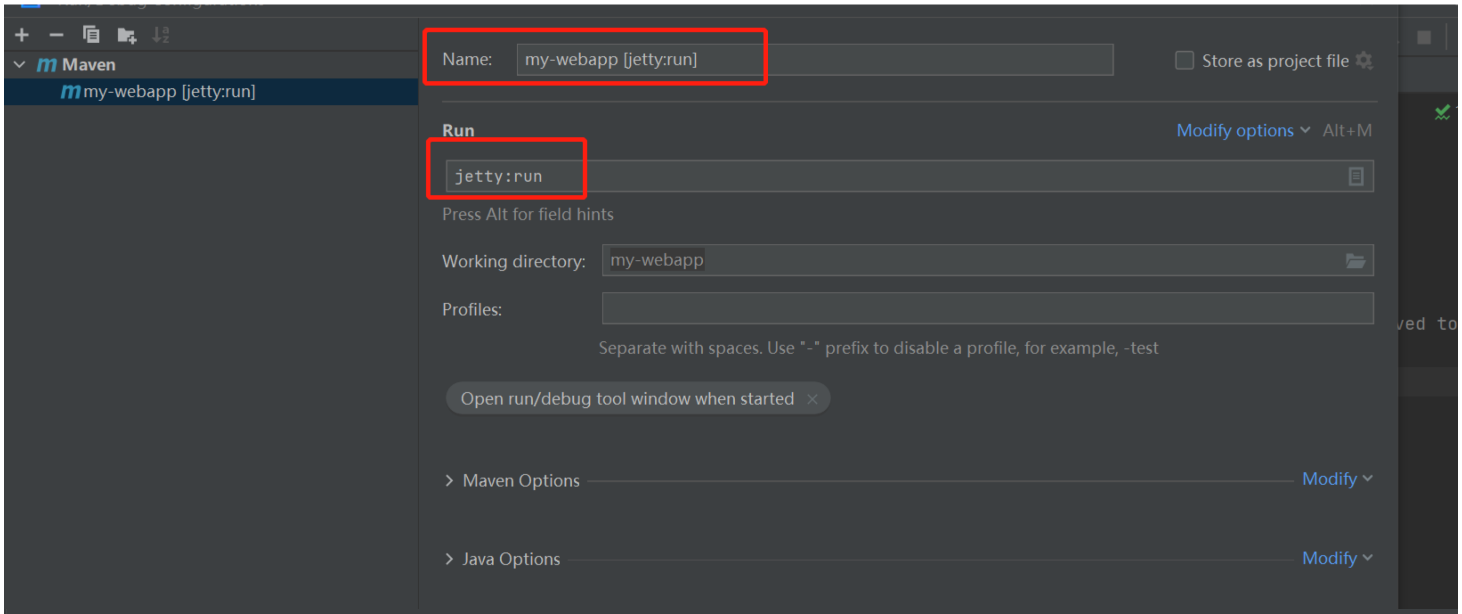

Tools VCS Window Help
my-webapp - pom.xml (my-webapp)
my-webapp [jetty:run]
pom.xml (my-webapp)
Maven
my-webapp Maven Webapp
Lifecycle
clean
validate
compile
test
package
verify
install
site
deploy
Plugins
clean (org.apache.maven.plugins:maven-clean-plugin:3.1.0)
compiler (org.apache.maven.plugins:maven-compiler-plugi
deploy (org.apache.maven.plugins:maven-deploy-plugin:2.8
install (org.apache.maven.plugins:maven-install-plugin:2.5.2
resources (org.apache.maven.plugins:maven-resources-plug
site (org.apache.maven.plugins:maven-site-plugin:3.12.1)
Maven
Notifications

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

  <build>
    <finalName>my-webapp</finalName>
    <pluginManagement><!-- lock down plugins versions

      <plugins>
        <plugin>
          <groupId>org.eclipse.jetty</groupId>
          <artifactId>jetty-maven-plugin</artifactId>
          <version>11.0.14</version>
        </plugin>
```



```
      <version>4.11</version>
      <scope>test</scope>
    </dependency>
  </dependencies>

  <build>
    <finalName>my-webapp</finalName>
    <pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent
```





Name: my-webapp [jetty:run]

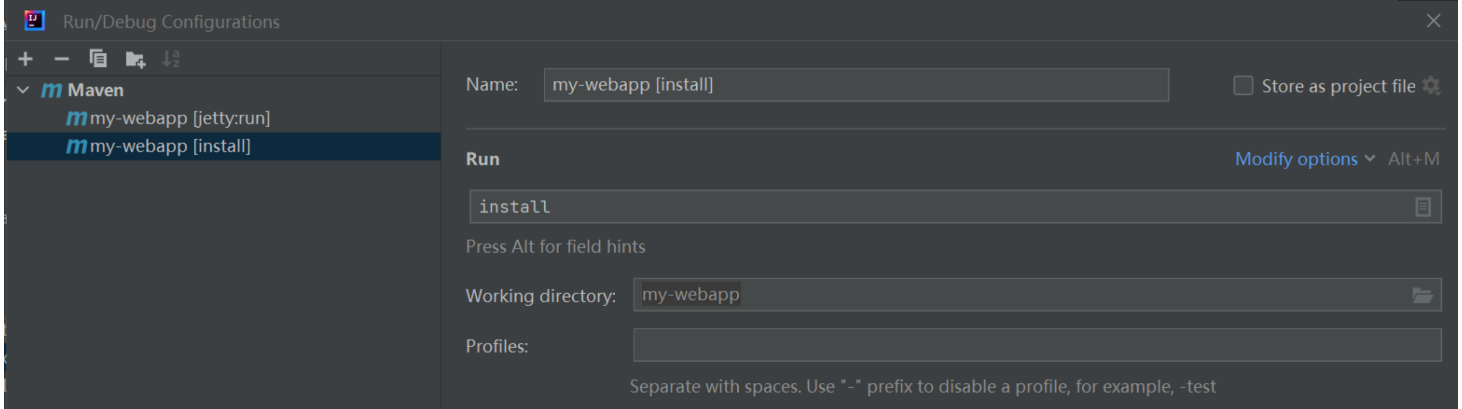
Run

```
jetty:run
```

Working directory: my-webapp

Profiles:

Separate with spaces. Use "-" prefix to disable a profile, for example, -test



Name: my-webapp [install]

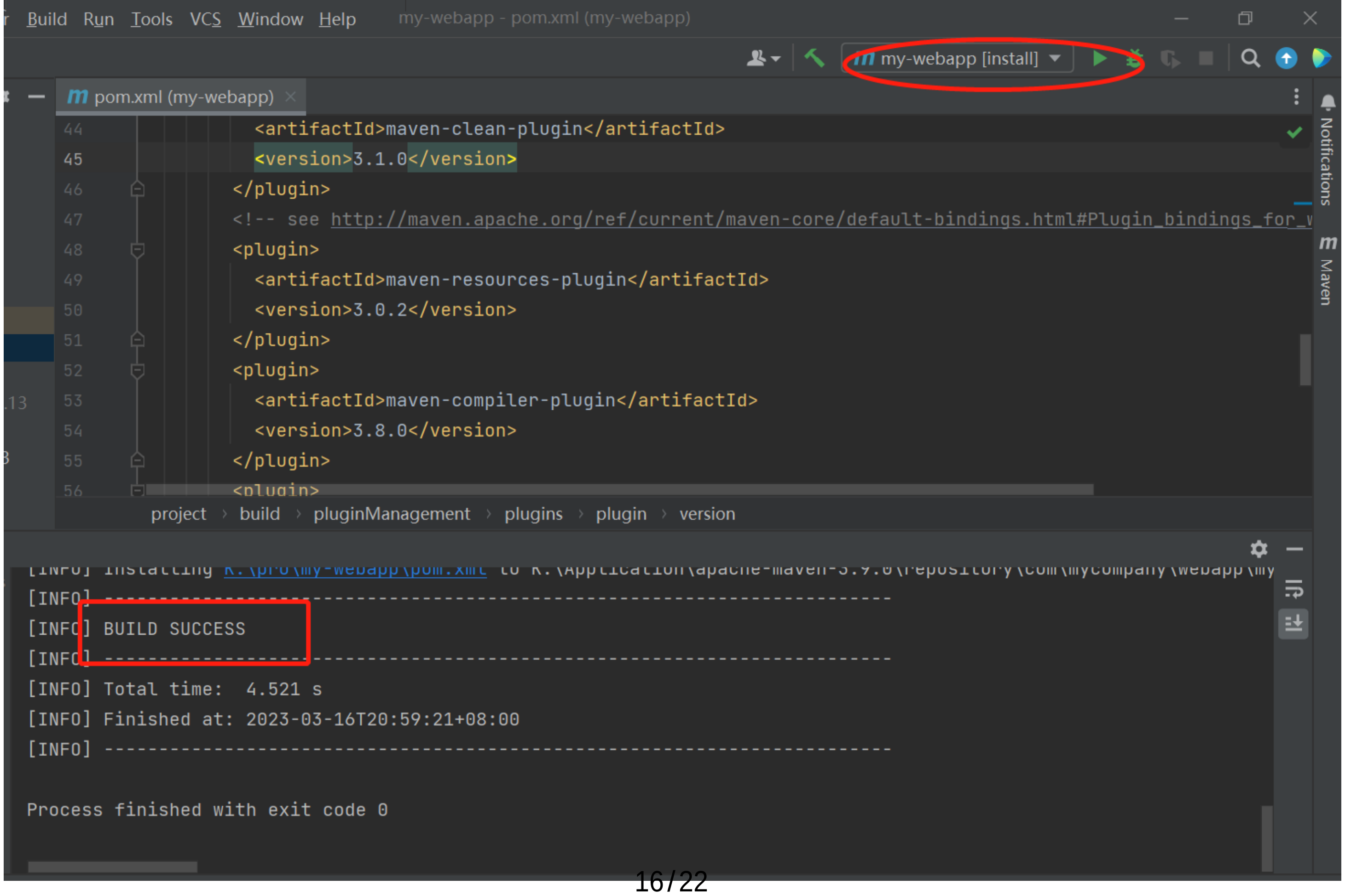
Run

```
install
```

Working directory: my-webapp

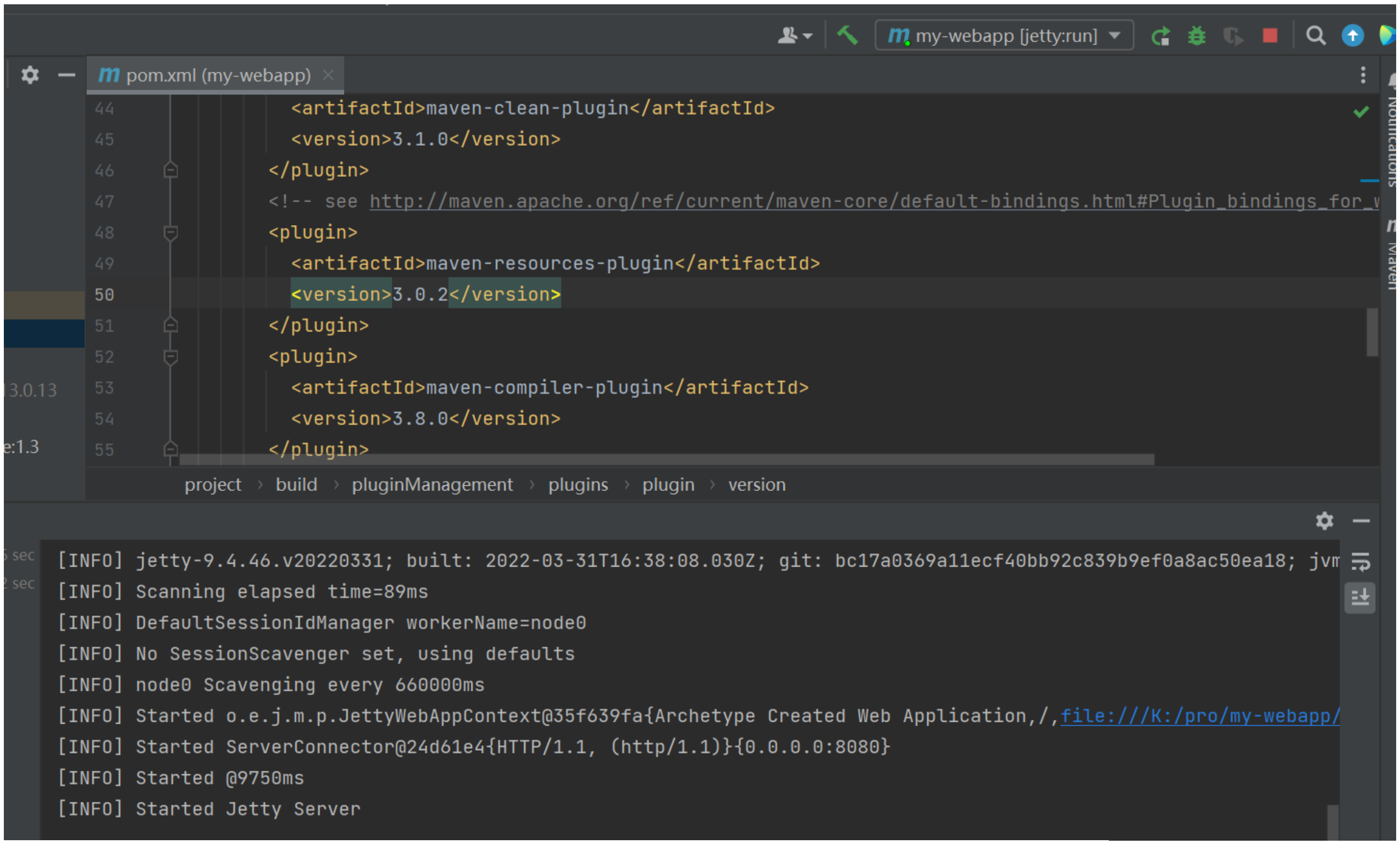
Profiles:

Separate with spaces. Use "-" prefix to disable a profile, for example, -test


Build Run Tools VCS Window Help
my-webapp - pom.xml (my-webapp)
my-webapp [install]
pom.xml (my-webapp)
Notifications
Maven
project › build › pluginManagement › plugins › plugin › version

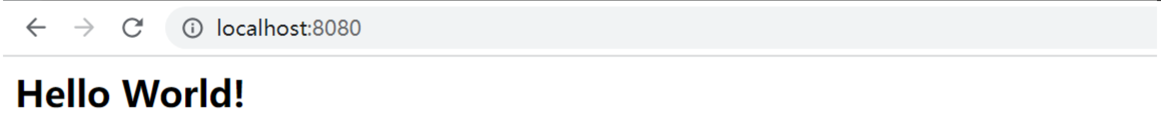
```
      <artifactId>maven-clean-plugin</artifactId>
      <version>3.1.0</version>
    </plugin>
    <!-- see http://maven.apache.org/ref/current/maven-core/default-bindings.html#Plugin_bindings_for_w
    <plugin>
      <artifactId>maven-resources-plugin</artifactId>
      <version>3.0.2</version>
    </plugin>
    <plugin>
      <artifactId>maven-compiler-plugin</artifactId>
      <version>3.8.0</version>
    </plugin>
    <plugin>
```

```
[INFO] Installing K:\pro\my-webapp\pom.xml to K:\Application\apache-maven-3.9.0\repository\com\mycompany\webapp\my
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  4.521 s
[INFO] Finished at: 2023-03-16T20:59:21+08:00
[INFO] ------------------------------------------------------------------------

Process finished with exit code 0
```


my-webapp [jetty:run]
pom.xml (my-webapp)
project › build › pluginManagement › plugins › plugin › version

```
          <artifactId>maven-clean-plugin</artifactId>
          <version>3.1.0</version>
        </plugin>
        <!-- see http://maven.apache.org/ref/current/maven-core/default-bindings.html#Plugin_bindings_for_w
        <plugin>
          <artifactId>maven-resources-plugin</artifactId>
          <version>3.0.2</version>
        </plugin>
        <plugin>
          <artifactId>maven-compiler-plugin</artifactId>
          <version>3.8.0</version>
        </plugin>
```

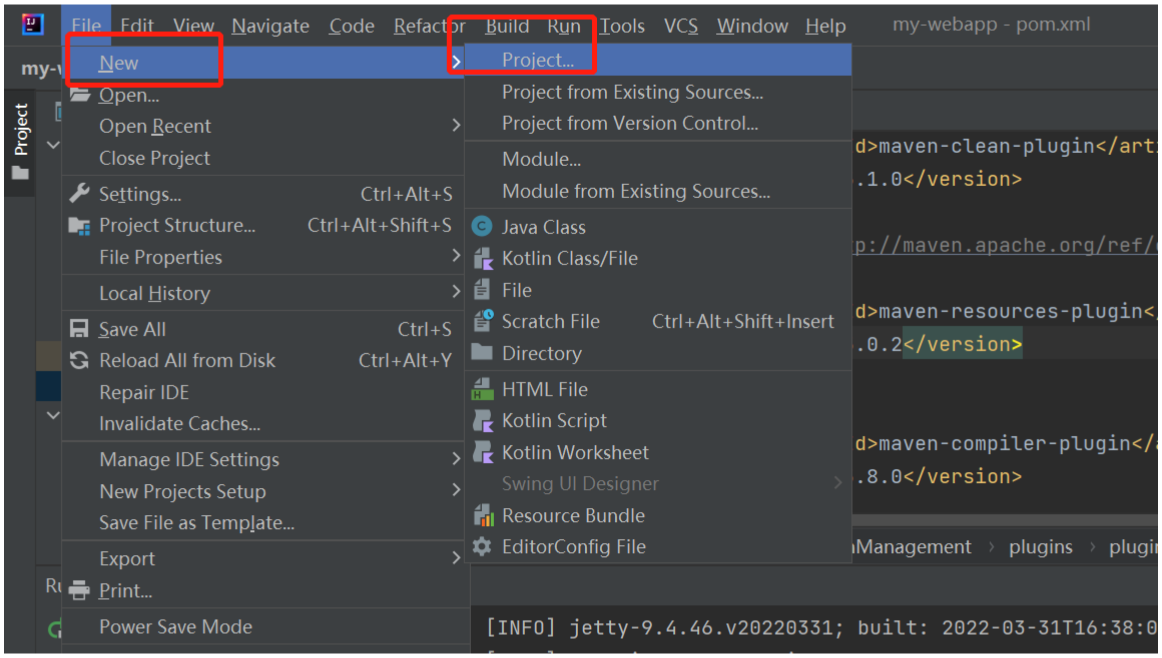
```
[INFO] jetty-9.4.46.v20220331; built: 2022-03-31T16:38:08.030Z; git: bc17a0369a11ecf40bb92c839b9ef0a8ac50ea18; jvm
[INFO] Scanning elapsed time=89ms
[INFO] DefaultSessionIdManager workerName=node0
[INFO] No SessionScavenger set, using defaults
[INFO] node0 Scavenging every 660000ms
[INFO] Started o.e.j.m.p.JettyWebAppContext@35f639fa{Archetype Created Web Application,/,file:///K:/pro/my-webapp/
[INFO] Started ServerConnector@24d61e4{HTTP/1.1, (http/1.1)}{0.0.0.0:8080}
[INFO] Started @9750ms
[INFO] Started Jetty Server
```



Hello World!

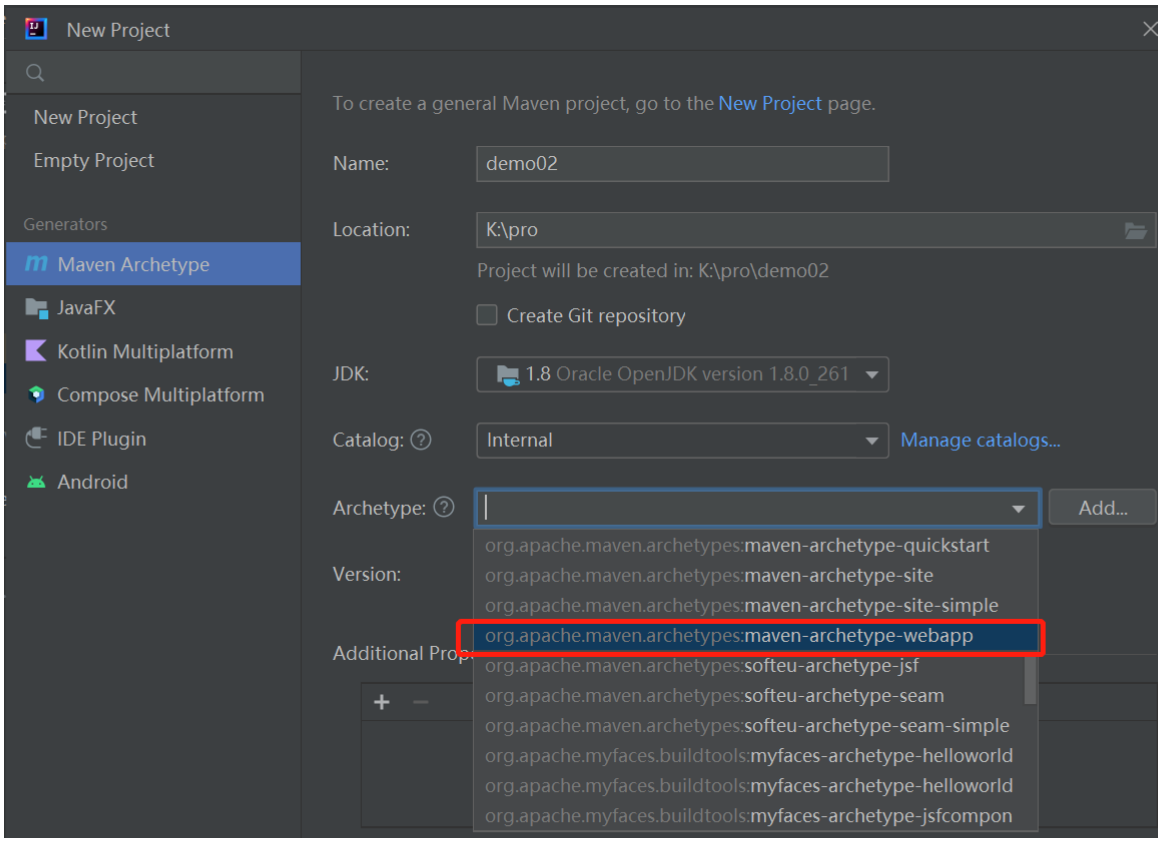
you can use tomcat also,please search tomcat7 in https://mvnrepository.com/

-- Create project by IDEA



```
d>maven-clean-plugin</art
.1.0</version>

p://maven.apache.org/ref/

d>maven-resources-plugin<
.0.2</version>

d>maven-compiler-plugin</
.8.0</version>
```

```
[INFO] jetty-9.4.46.v20220331; built: 2022-03-31T16:38:0
```



To create a general Maven project, go to the New Project page.

Name: demo02

Location: K:\pro

Project will be created in: K:\pro\demo02

Create Git repository

JDK: 1.8 Oracle OpenJDK version 1.8.0_261

Catalog: Internal

Archetype:

org.apache.maven.archetypes:maven-archetype-quickstart
org.apache.maven.archetypes:maven-archetype-site
org.apache.maven.archetypes:maven-archetype-site-simple
org.apache.maven.archetypes:maven-archetype-webapp
org.apache.maven.archetypes:softeu-archetype-jsf
org.apache.maven.archetypes:softeu-archetype-seam
org.apache.maven.archetypes:softeu-archetype-seam-simple
org.apache.myfaces.buildtools:myfaces-archetype-helloworld
org.apache.myfaces.buildtools:myfaces-archetype-helloworld
org.apache.myfaces.buildtools:myfaces-archetype-jsfcompon

Part 4 Teedy poms

1. /Teedy/pom.xml

1.Title:

```
<groupId>com.sismics.docs</groupId>
<artifactId>docs-parent</artifactId>

<!--packaging is the default: jar,this means the type is pom-->
<packaging>pom</packaging>

<version>1.10</version>

<!--project name-->
<name>Docs Parent</name>
```

2. Properties:
   version define: line 21 and 182-186

```
  <!--properties define the version of dependency-->
  <org.apache.commons.commons-compress.version>1.18</org.apache.commons.commons-compress.version>

     <dependency>
      <groupId>org.apache.commons</groupId>
      <artifactId>commons-compress</artifactId>
      <version>${org.apache.commons.commons-compress.version}</version>
    </dependency>
```

3. scm :line 69~70:
   You can config your repo by scm.

```
<scm>
  <connection>scm:git:https://github.com/sismics/docs.git</connection>
  <developerConnection>scm:git:https://github.com/docs/docs.git</developerConnection>
  <url>scm:git:https://github.com/sismics/docs.git</url>
  <tag>HEAD</tag>
</scm>
```

4. modules:list every modules that construct this project.

```
<modules>
  <module>docs-core</module>
  <module>docs-web-common</module>
  <module>docs-web</module>
</modules>
```

5. dependency:line 131~508
   dependencyManagement:if you have 3 modules, you can use dependencyManagement to manage dependency version.

```
Parent:
  <version>1.10</version>

  <dependencyManagement>
    <dependencies>
      <dependency>
        <groupId>com.sismics.docs</groupId>
        <artifactId>docs-core</artifactId>
        <version>${project.version}</version>
      </dependency>
    <dependencies>
  <dependencyManagement>

sub modules:
  <dependencies>
    <!-- Dependencies to Docs -->
    <dependency>
      <groupId>com.sismics.docs</groupId>
      <artifactId>docs-core</artifactId>
    </dependency>
  <dependencies>
```

From this pom we can see there are 3 modules:docs-core,docs-web-common,docs-web.

```
Teedy
|-- pom.xml
    |--dependency(core)
    |--dependency(web-common)
    |--dependency(web)
|-- docs-core
    |-- pom.xml
|-- docs-web-common
    |-- pom.xml
        |--dependency(core)
|-- docs-web
    |-- pom.xml
        |--dependency(core)
        |--dependency(web-common)
```

2. /Teedy/docs-core/pom.xml

1.Front part

```
/Teedy/pom.xml
  <groupId>com.sismics.docs</groupId>
  <artifactId>docs-parent</artifactId>
  <packaging>pom</packaging>
  <version>1.10</version>

/Teedy/docs-core/pom.xml
/Teedy/docs-web-common/pom.xml
/Teedy/docs-web/pom.xml
  <parent>
    <groupId>com.sismics.docs</groupId>
    <artifactId>docs-parent</artifactId>
    <version>1.10</version>
    <!--Parent Pom.xml relative path-->
    <relativePath>..</relativePath>
  </parent>
```

2. scope line 218~228:when you run test command this dependency is used.

```
....
    <dependency>
      <groupId>com.h2database</groupId>
      <artifactId>h2</artifactId>
      <scope>test</scope>
    </dependency>
....
```

3. profile:
   Teedy defined 2 profiles:dev and prod which correspond to different configurations,and this id affect docs-web configurations.

```
    <profiles>
      <!-- Development profile (active by default) -->
      <profile>
        <id>dev</id>
        <activation>
          <activeByDefault>true</activeByDefault>
          <property>
            <name>env</name>
            <value>dev</value>
          </property>
        </activation>

  ....

      <!-- Production profile -->
      <profile>
        <id>prod</id>
      </profile>
    </profiles>
```

3. /Teedy/docs-web-common/pom.xml

Two jar packages are exported here, and there are two dependencies in the web.

```
  ...
          <executions>
            <execution>
              <goals>
                <goal>test-jar</goal>
              </goals>
            </execution>
          </executions>
  ...
```

4. /Teedy/docs-web/pom.xml

line 152~285:
config 2 profiles:Development profile and Production profile.