

Lesson 01 Demo 02

Using Maven CLI to Create Java Web Project

Objective: To create a Java Web Project using Maven CLI and work with the Maven commands to configure and execute the project

Tools required: Visual Studio Code and Maven

Prerequisites: None

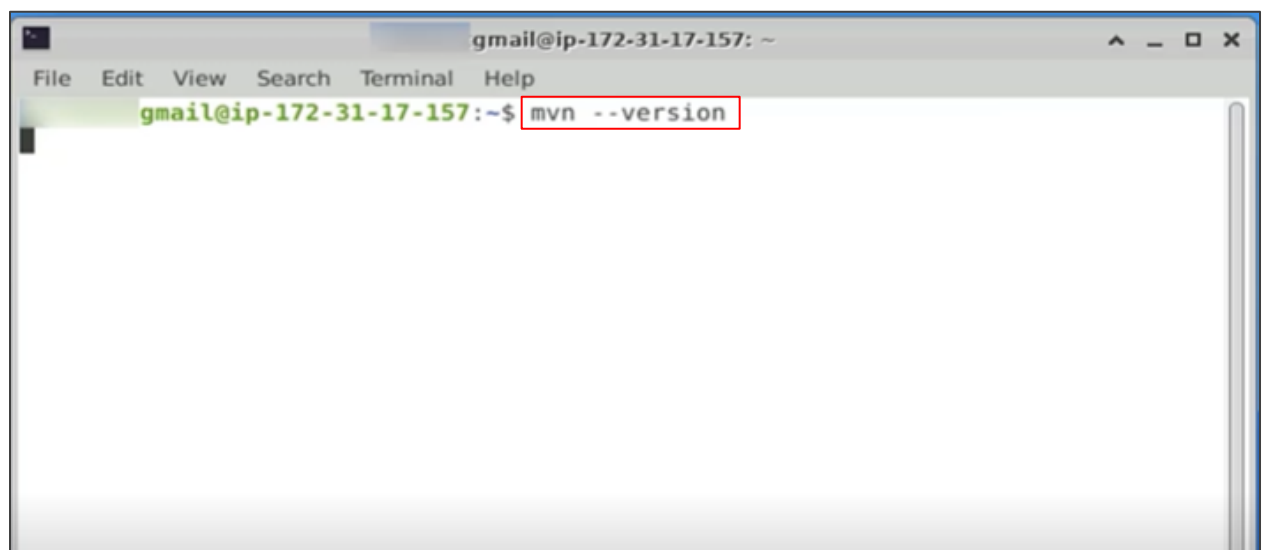
Steps to be followed:

1. Run the mvn package command
2. Open the CMS project
3. Add the dependency

Step 1: Run the mvn package command

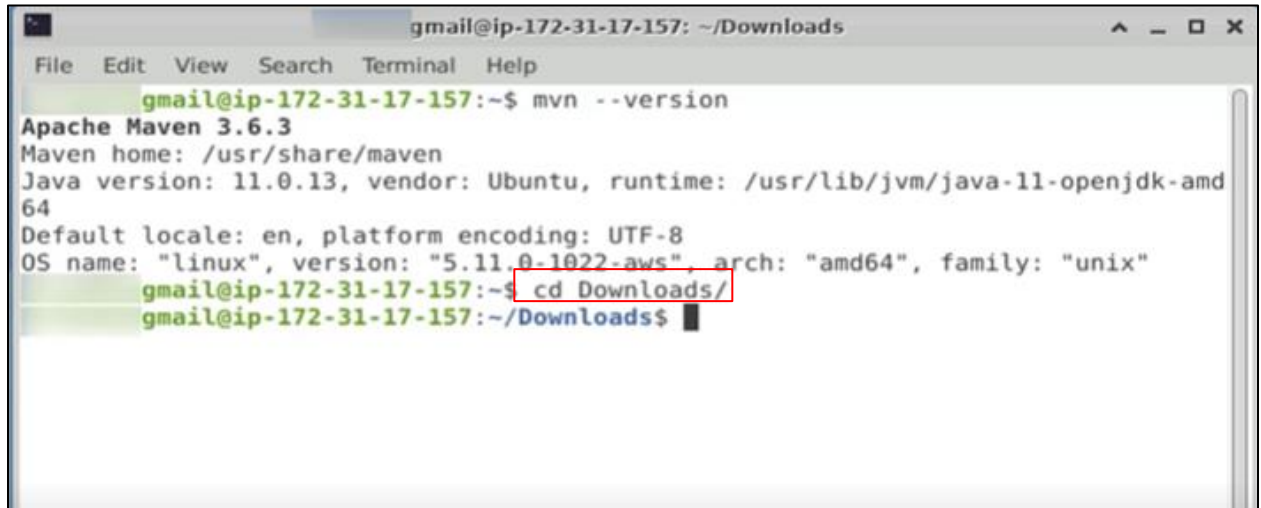
1.1 Open the terminal and check if Maven is installed using the following command:

mvn --version



The screenshot shows a terminal window with the title bar 'gmail@ip-172-31-17-157: ~'. The menu bar includes 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The prompt 'gmail@ip-172-31-17-157:~\$' is followed by the command 'mvn --version', which is highlighted with a red rectangular box.

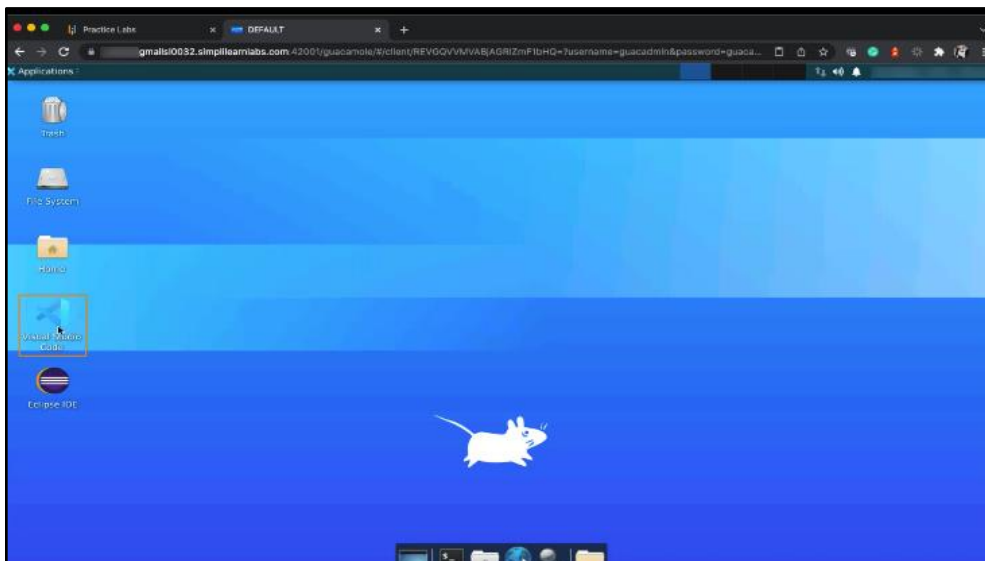
- 1.2 Use the `cd` command to navigate to the **Downloads** directory:
`cd Downloads/`



```

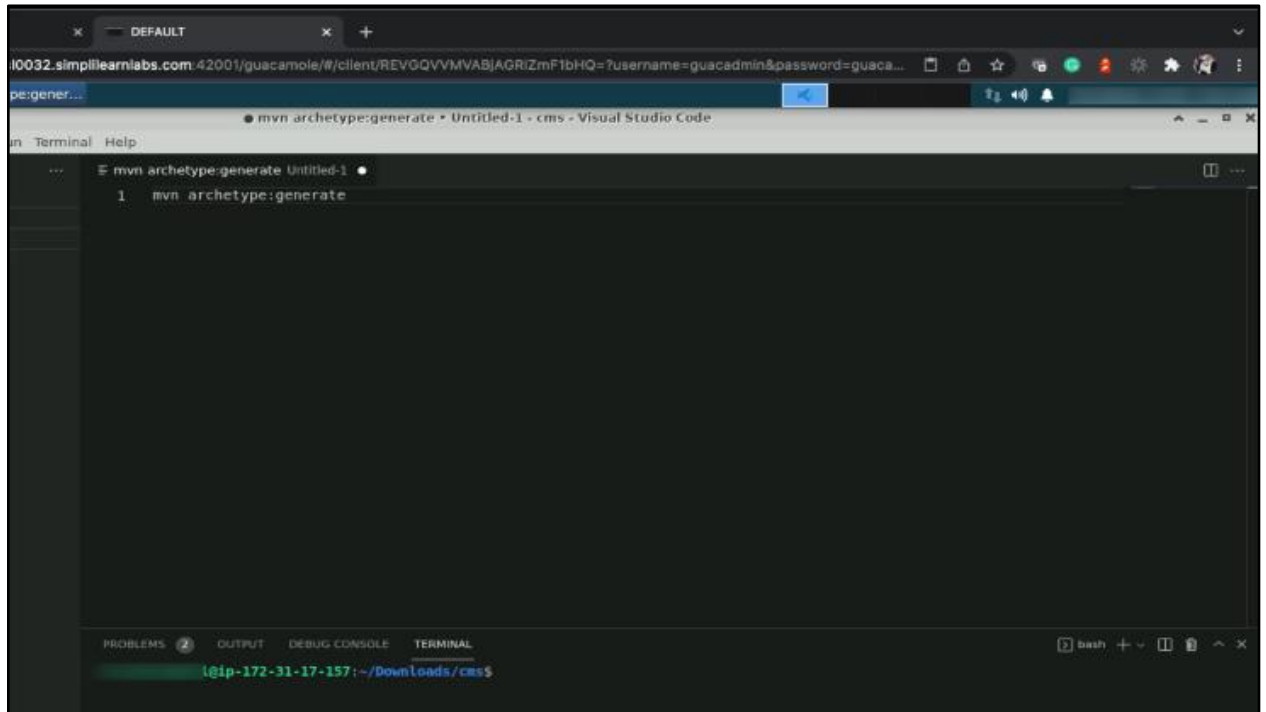
gmail@ip-172-31-17-157: ~/Downloads
File Edit View Search Terminal Help
gmail@ip-172-31-17-157:~$ mvn --version
Apache Maven 3.6.3
Maven home: /usr/share/maven
Java version: 11.0.13, vendor: Ubuntu, runtime: /usr/lib/jvm/java-11-openjdk-amd64
Default locale: en, platform encoding: UTF-8
OS name: "linux", version: "5.11.0-1022-aws", arch: "amd64", family: "unix"
gmail@ip-172-31-17-157:~$ cd Downloads/
gmail@ip-172-31-17-157:~/Downloads$
  
```

- 1.3 To execute the different commands, open the Visual Studio Code



1.4 Use the following command to create a project:

mvn archetype:generate



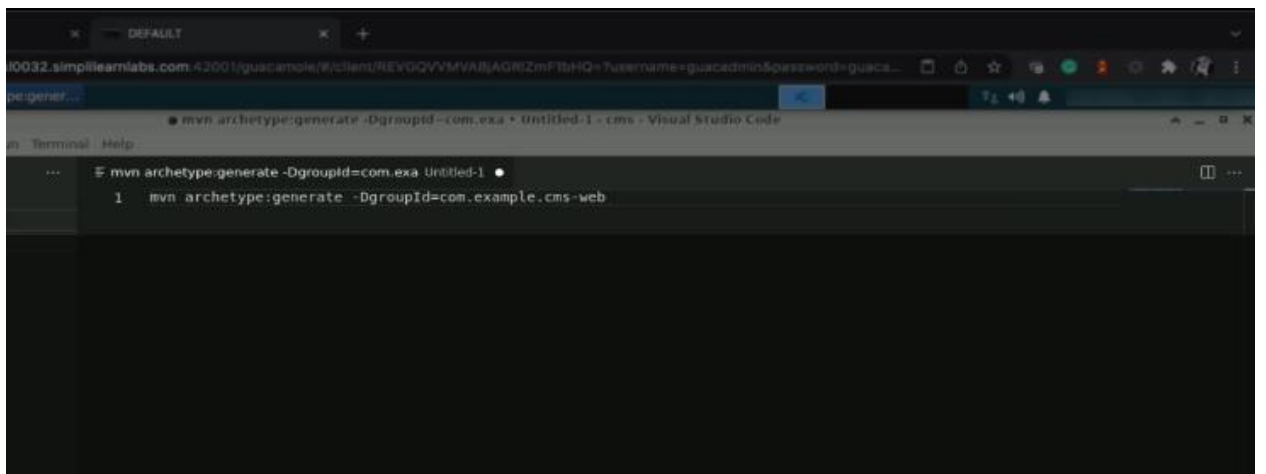
The screenshot shows the Visual Studio Code interface. The top panel displays a web browser with a URL from simplilearnlabs.com. The bottom panel shows a terminal window with the command `mvn archetype:generate` entered. The terminal output shows the command being executed in a bash shell on a system with IP 172-31-17-157.

```
mvn archetype:generate
1 mvn archetype:generate

l@ip-172-31-17-157:~/Downloads/cms$
```

1.5 The project must have the group ID, which can be the company's domain name in reverse order. Add group ID for the project:

-DgroupId=com.example.cms-web

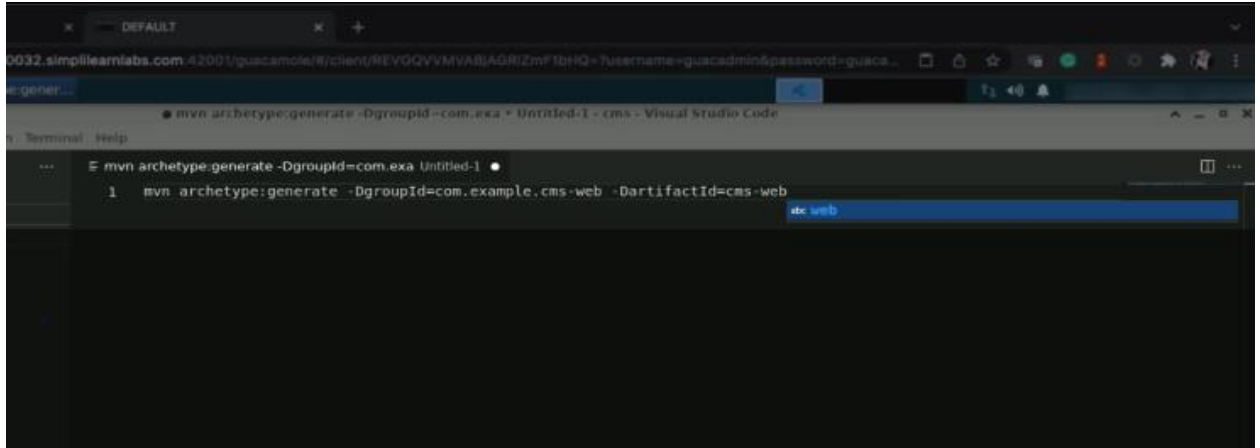


The screenshot shows the Visual Studio Code interface. The top panel displays a web browser with a URL from simplilearnlabs.com. The bottom panel shows a terminal window with the command `mvn archetype:generate -DgroupId=com.example.cms-web` entered. The terminal output shows the command being executed in a bash shell on a system with IP 172-31-17-157.

```
mvn archetype:generate -DgroupId=com.example.cms-web
1 mvn archetype:generate -DgroupId=com.example.cms-web
```

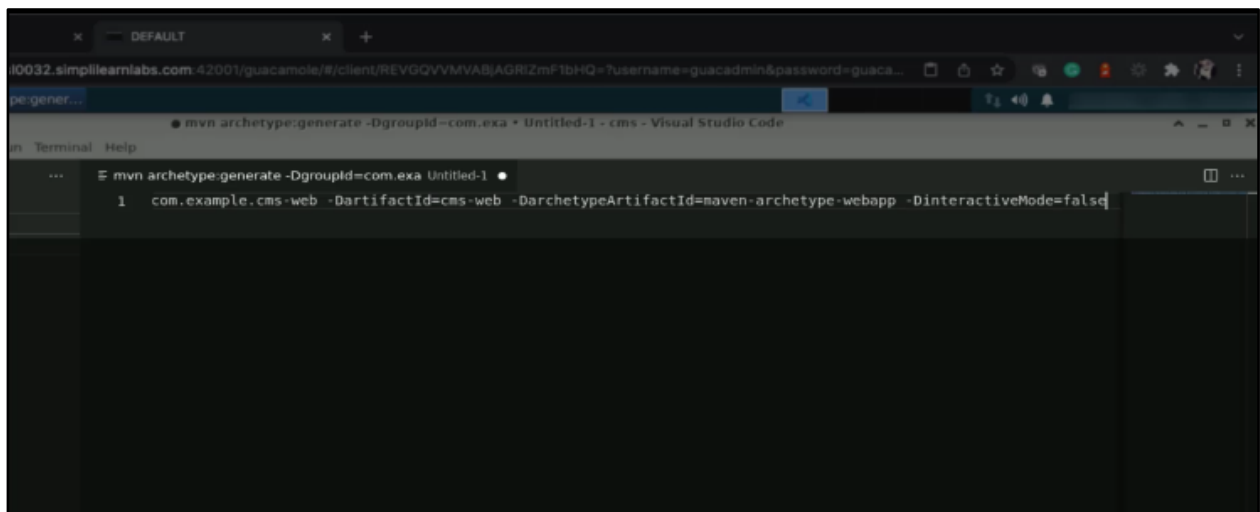
1.6 Add an artifactId for the project, which is the application name for the project

-DartifactId=cms-web



1.7 Add archetype and interactive mode:

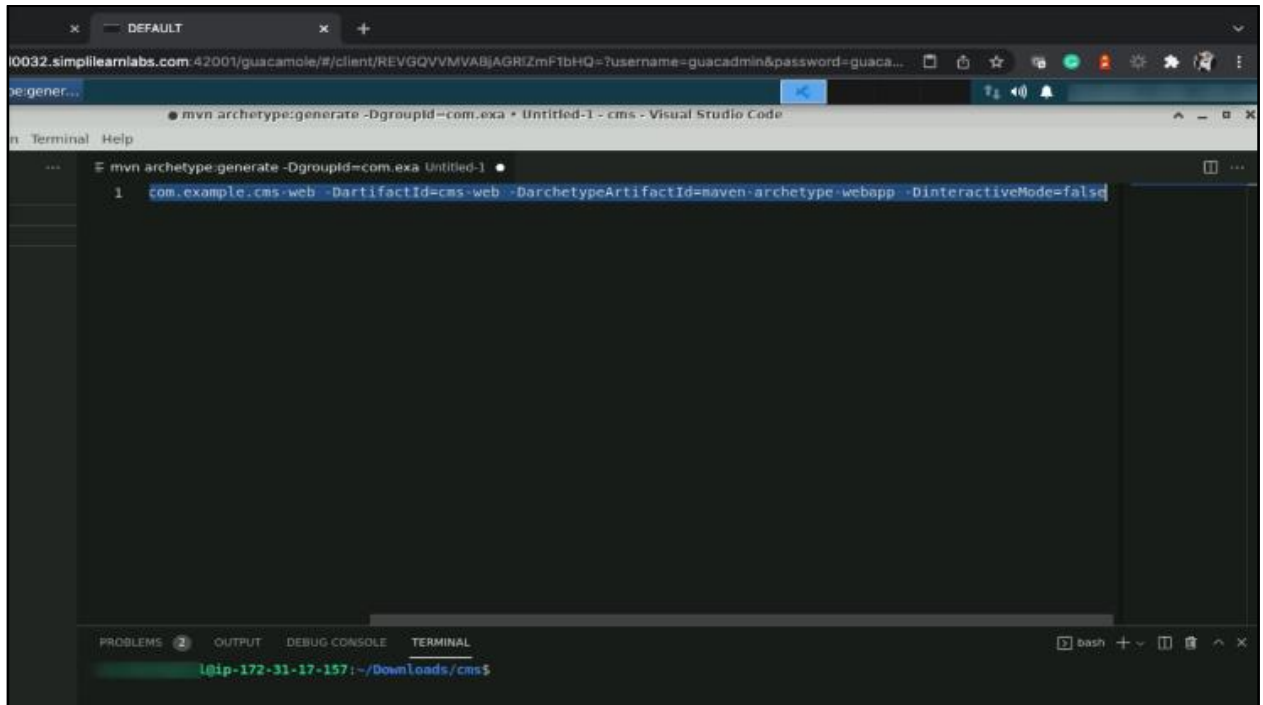
-DarchetypeArtifactId=maven-archetype-webapp -DinteractiveMode=false



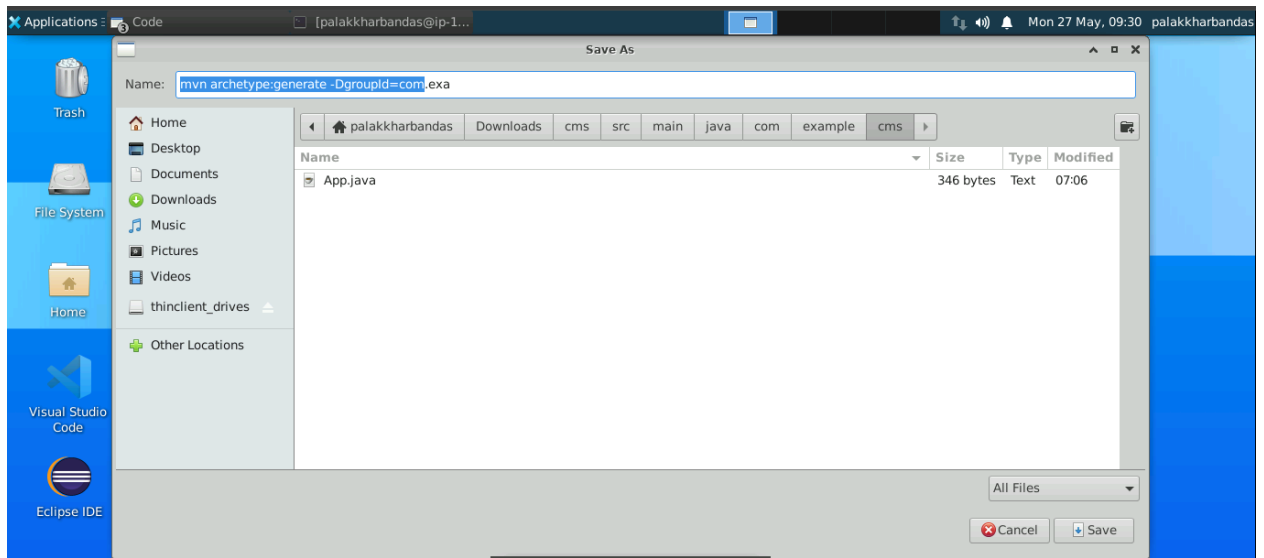
The **-DarchetypeArtifactId=maven-archetype-webapp** flag specifies the Maven archetype or project template for creating a new web application project. The **-DinteractiveMode=false** flag tells Maven to use non-interactive mode, which means that it won't prompt the user for input during the project creation process.

The complete command will look like: **mvn archetype:generate -DgroupId=com.example.cms-web -DartifactId=cms-web -DarchetypeArtifactId=maven-archetype-quickstart -DarchetypeVersion=1.4 -DinteractiveMode=false**

1.8 Copy the following command:

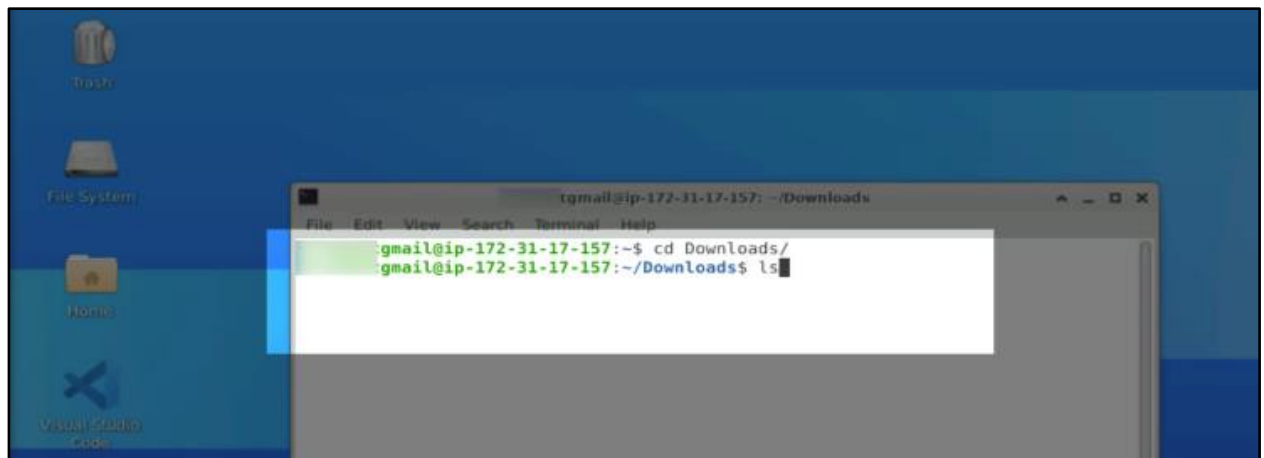


The screenshot shows a Visual Studio Code editor with a terminal window open. The terminal displays the command `mvn archetype:generate -DgroupId=com.example.cms-web -DartifactId=cms-web -DarchetypeArtifactId=maven-archetype-quickstart -DarchetypeVersion=1.4 -DinteractiveMode=false` being entered. The command is highlighted in blue. The terminal window is titled "mvn archetype:generate -DgroupId=com.example.cms-web -DartifactId=cms-web -DarchetypeArtifactId=maven-archetype-quickstart -DarchetypeVersion=1.4 -DinteractiveMode=false". The Visual Studio Code interface includes a sidebar on the left with "Terminal" and "Help" tabs, and a bottom status bar showing "bash" and "ip-172-31-17-157: ~/Downloads/cms\$".



1.9 Enter the **Downloads** directory in the terminal window:

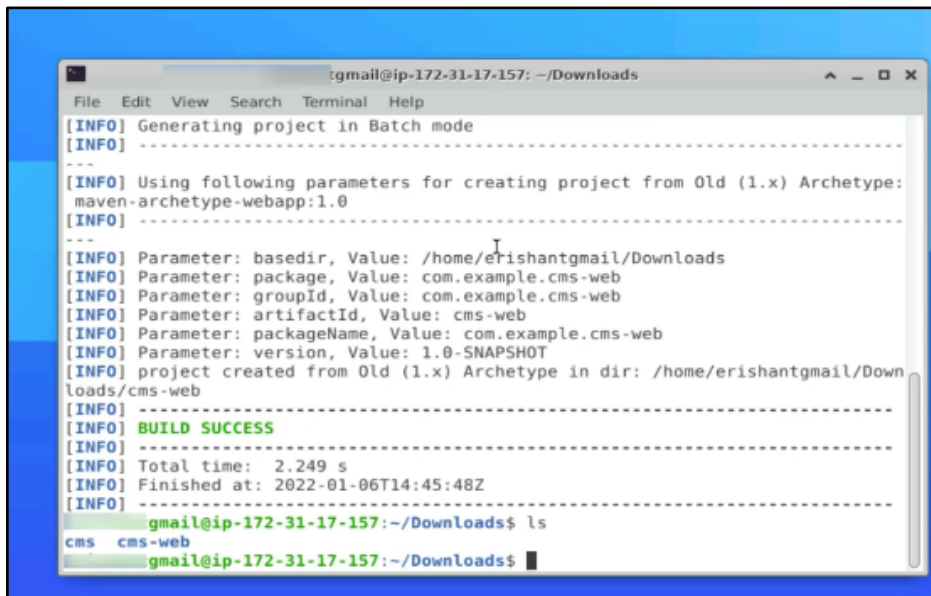
cd Downloads/



1.10 Paste the above command and press **Enter**:

```
mvn archetype:generate -DgroupId=com.example.cms-web -DartifactId=cms-web -DarchetypeArtifactId=maven-archetype-quickstart -DarchetypeVersion=1.4 -DinteractiveMode=false
```

```
palakkharbandas@ip-172-31-17-240:~/Downloads$ mvn archetype:generate -DgroupId=com.example.cms-web -DartifactId=cms-web -DarchetypeArtifactId=maven-archetype-quickstart -DarchetypeVersion=1.4 -DinteractiveMode=false
[INFO] Scanning for projects...
[INFO] -----< org.apache.maven:standalone-pom >-----
[INFO] Building Maven Stub Project (No POM) 1
[INFO] -----[ pom ]-----
[INFO] >>> maven-archetype-plugin:3.2.1:generate (default-cli) > generate-sources @ standalone-pom >>>
[INFO] <<< maven-archetype-plugin:3.2.1:generate (default-cli) < generate-sources @ standalone-pom <<<
[INFO] --- maven-archetype-plugin:3.2.1:generate (default-cli) @ standalone-pom ---
[INFO] Generating project in Batch mode
[INFO]
```



```
gmail@ip-172-31-17-157: ~/Downloads
File Edit View Search Terminal Help
[INFO] Generating project in Batch mode
[INFO] -----
[INFO] Using following parameters for creating project from Old (1.x) Archetype:
maven-archetype-webapp:1.0
[INFO] -----
[INFO] Parameter: basedir, Value: /home/erishantgmail/Downloads
[INFO] Parameter: package, Value: com.example.cms-web
[INFO] Parameter: groupId, Value: com.example.cms-web
[INFO] Parameter: artifactId, Value: cms-web
[INFO] Parameter: packageName, Value: com.example.cms-web
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[INFO] project created from Old (1.x) Archetype in dir: /home/erishantgmail/Downloads/cms-web
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 2.249 s
[INFO] Finished at: 2022-01-06T14:45:48Z
[INFO] -----
gmail@ip-172-31-17-157:~/Downloads$ ls
cms  cms-web
gmail@ip-172-31-17-157:~/Downloads$
```

```

gmail@ip-172-31-17-157: ~/Downloads
[INFO] Generating project in Batch mode
[INFO] -----
[INFO] Using following parameters for creating project from Old (1.x) Archetype:
maven-archetype-webapp:1.0
[INFO] -----
[INFO] Parameter: basedir, Value: /home/erishantgmail/Downloads
[INFO] Parameter: package, Value: com.example.cms-web
[INFO] Parameter: groupId, Value: com.example.cms-web
[INFO] Parameter: artifactId, Value: cms-web
[INFO] Parameter: packageName, Value: com.example.cms-web
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[INFO] project created from Old (1.x) Archetype in dir: /home/erishantgmail/Down
loads/cms-web
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 2.249 s
[INFO] Finished at: 2022-01-06T14:45:48Z
[INFO] -----
gmail@ip-172-31-17-157:~/Downloads$ ls
cms  cms-web
gmail@ip-172-31-17-157:~/Downloads$

```

1.11 Use the **ls** command to list the directories, and the **cms** folder will be present

```

[INFO] -----
gmail@ip-172-31-17-157:~/Downloads$ ls
cms  cms-web
gmail@ip-172-31-17-157:~/Downloads$

```

1.11 Use the **cd** command to navigate to the **cms-web** folder:

cd cms-web/

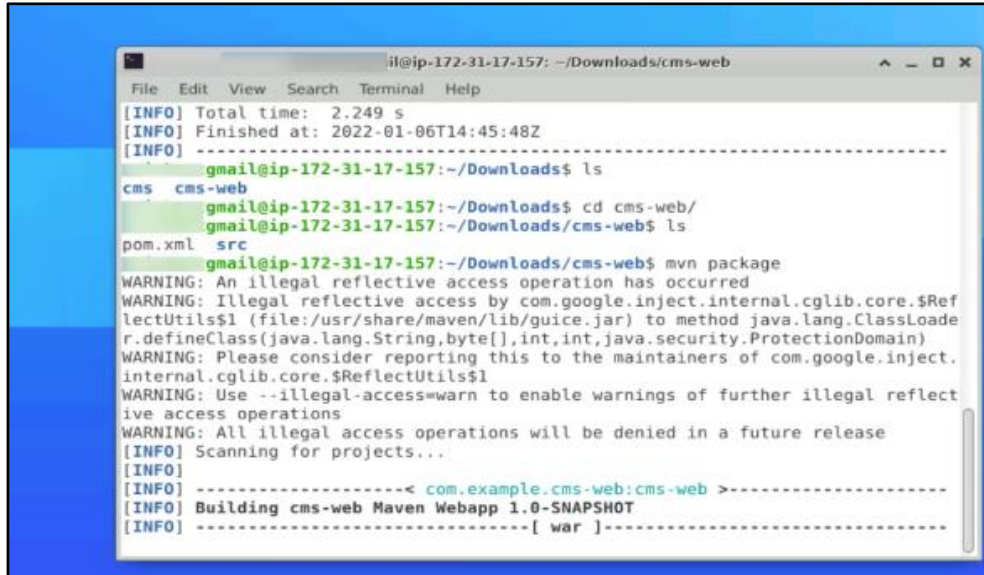
```

il@ip-172-31-17-157: ~/Downloads/cms-web
[INFO] -----
[INFO] Using following parameters for creating project from Old (1.x) Archetype:
maven-archetype-webapp:1.0
[INFO] -----
[INFO] Parameter: basedir, Value: /home/erishantgmail/Downloads
[INFO] Parameter: package, Value: com.example.cms-web
[INFO] Parameter: groupId, Value: com.example.cms-web
[INFO] Parameter: artifactId, Value: cms-web
[INFO] Parameter: packageName, Value: com.example.cms-web
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[INFO] project created from Old (1.x) Archetype in dir: /home/erishantgmail/Down
loads/cms-web
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 2.249 s
[INFO] Finished at: 2022-01-06T14:45:48Z
[INFO] -----
gmail@ip-172-31-17-157:~/Downloads$ ls
cms  cms-web
gmail@ip-172-31-17-157:~/Downloads$ cd cms-web/
gmail@ip-172-31-17-157:~/Downloads/cms-web$

```


1.12 Use the **mvn package** command to build the project:

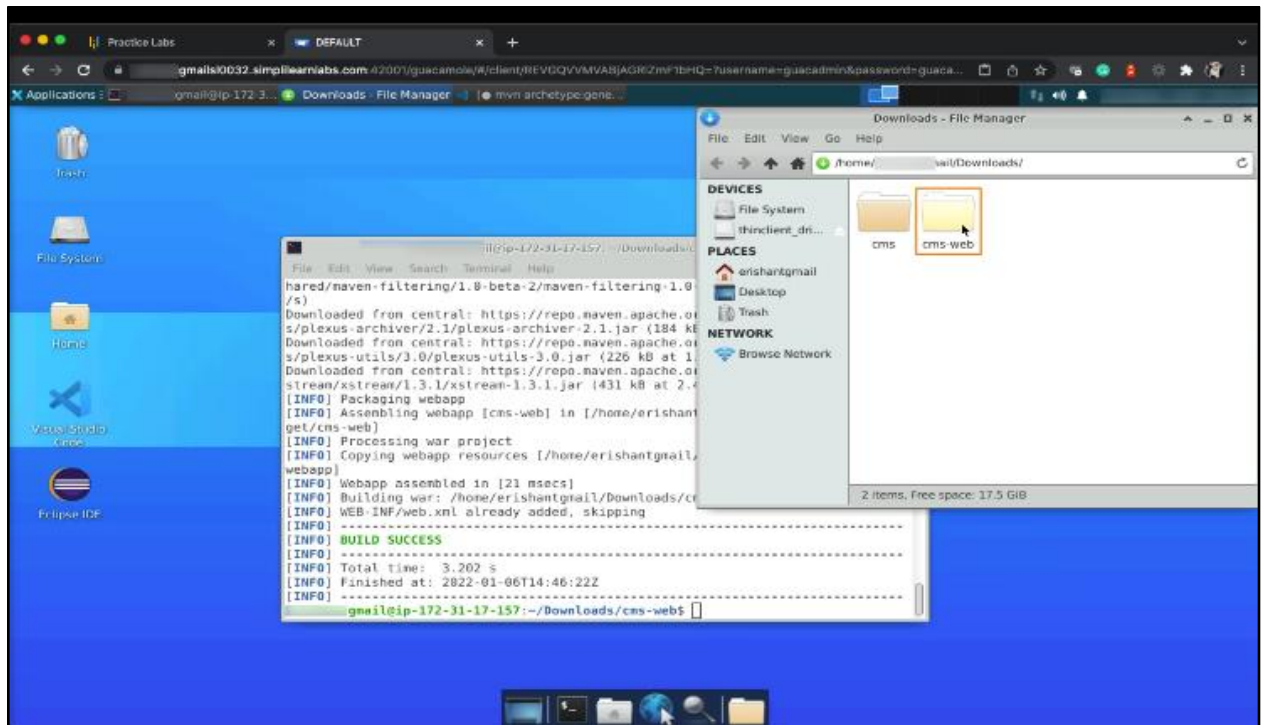
mvn package



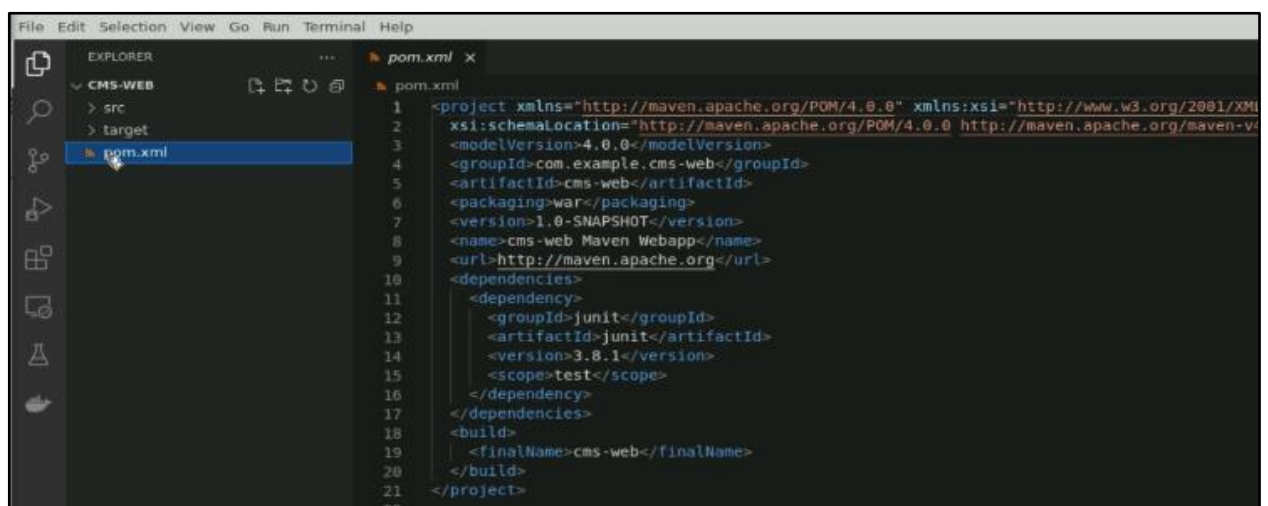
```
il@ip-172-31-17-157: ~/Downloads/cms-web
File Edit View Search Terminal Help
[INFO] Total time: 2.249 s
[INFO] Finished at: 2022-01-06T14:45:48Z
[INFO] -----
gmail@ip-172-31-17-157:~/Downloads$ ls
cms  cms-web
gmail@ip-172-31-17-157:~/Downloads$ cd cms-web/
gmail@ip-172-31-17-157:~/Downloads/cms-web$ ls
pom.xml  src
gmail@ip-172-31-17-157:~/Downloads/cms-web$ mvn package
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by com.google.inject.internal.cglib.core.$ReflectUtils$1 (file:/usr/share/maven/lib/guice.jar) to method java.lang.ClassLoader.defineClass(java.lang.String,byte[],int,int,java.security.ProtectionDomain)
WARNING: Please consider reporting this to the maintainers of com.google.inject.internal.cglib.core.$ReflectUtils$1
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
[INFO] Scanning for projects...
[INFO] -----< com.example.cms-web:cms-web >-----
[INFO] Building cms-web Maven Webapp 1.0-SNAPSHOT
[INFO] -----[ war ]-----
```

Step 2: Open the CMS project

2.1 Navigate to the **Downloads** directory and open the **cms-web** folder in the Visual Studio Code

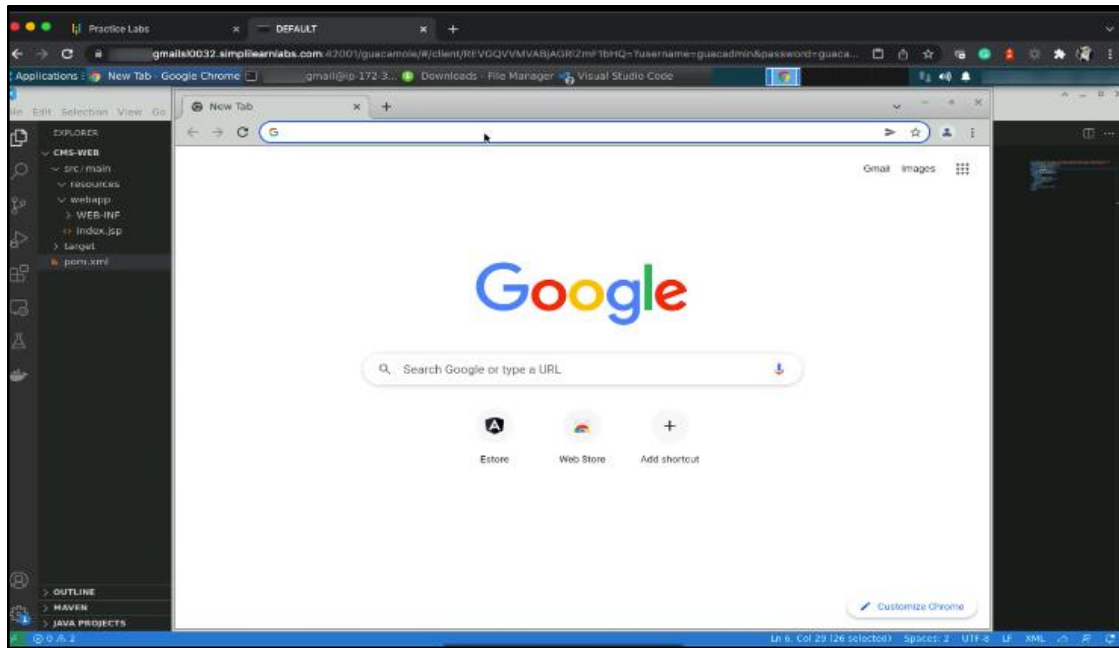


2.2 Open the **pom.xml** file. Check and ensure that the packaging is **war** and not **jar**.

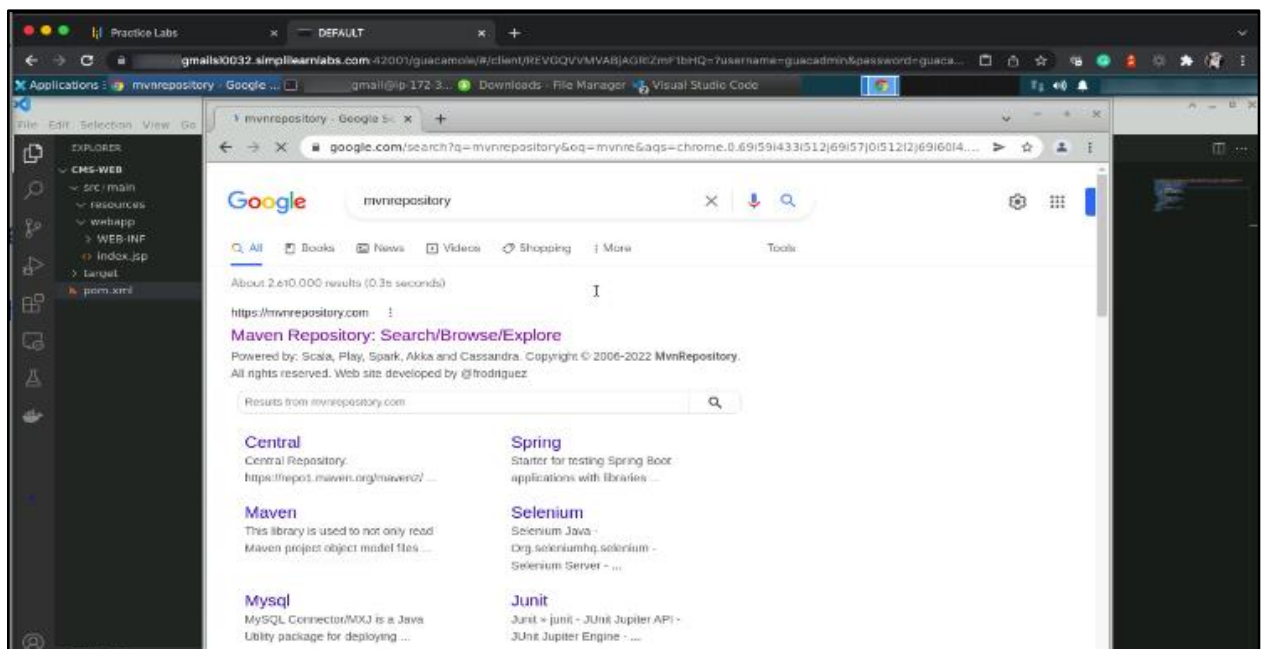


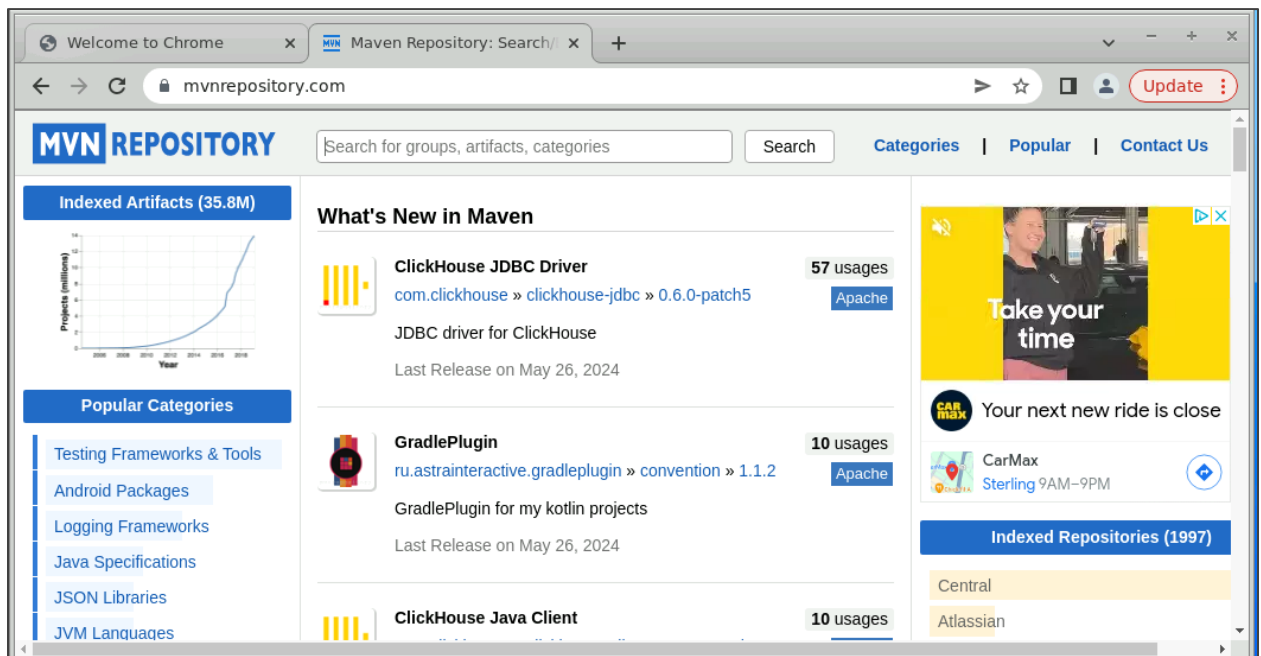
Step 3: Add the dependency

3.1 Open the Google browser

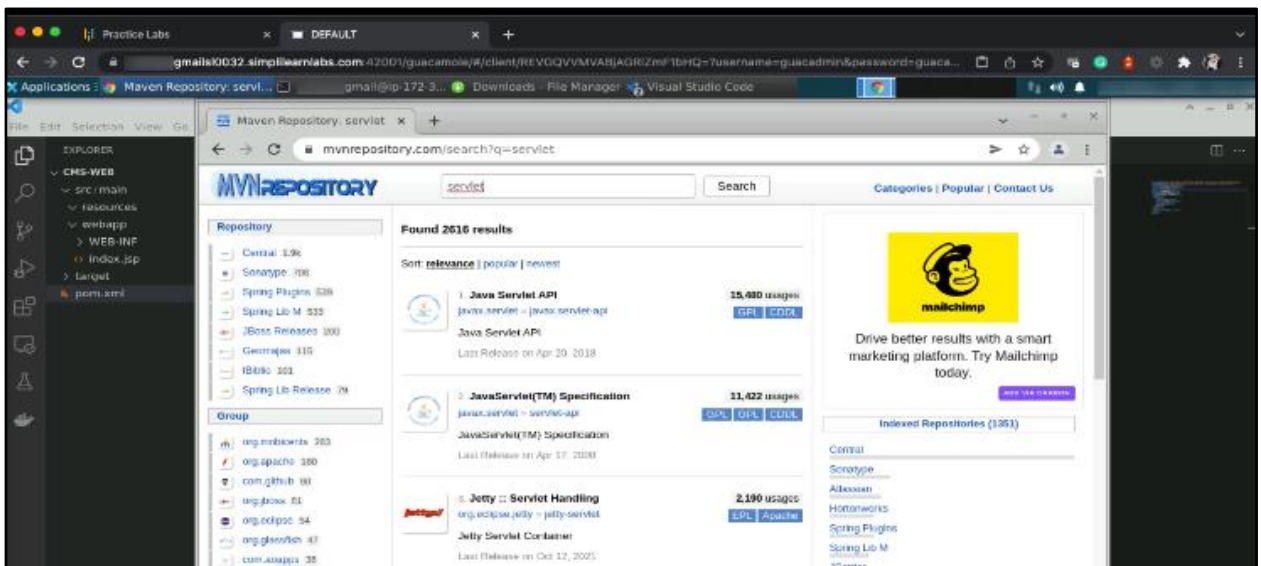


3.2 Search for mvn repository





3.3 Search for **servlet** in the search box and select **Java Servlet API**



3.4 Select the latest version of Servlet, which is 4.0.1

The screenshot shows the Maven Repository website for the Java Servlet API. The 'Version' table lists the following versions and their release dates:

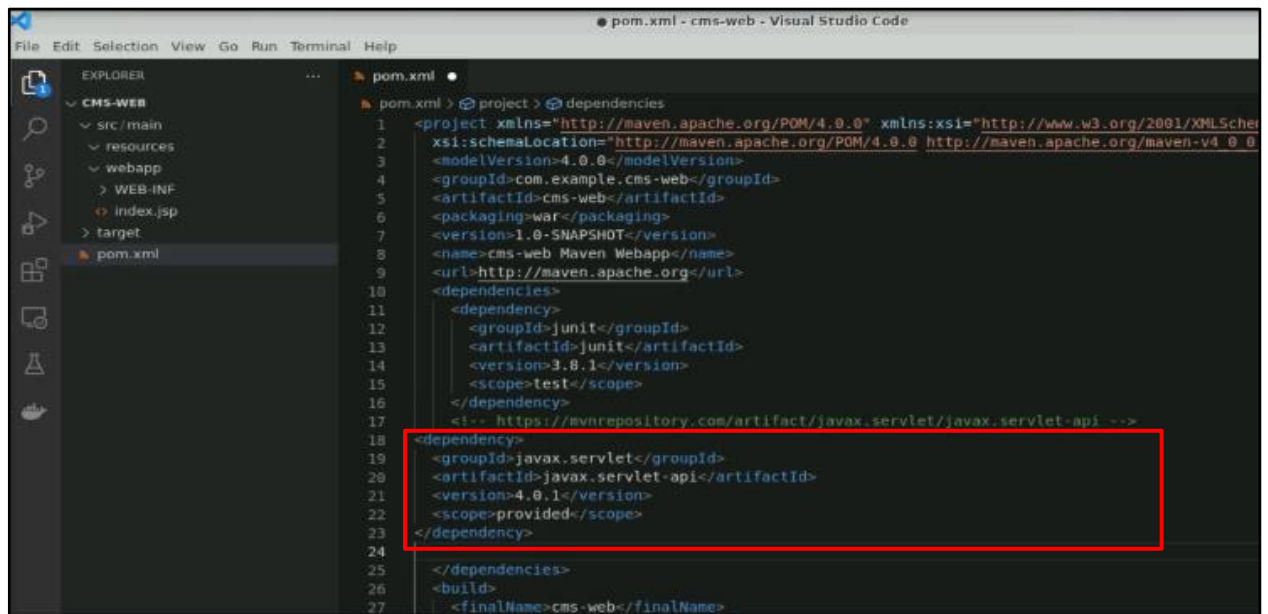
Version	Vulnerabilities	Repository	Usages	Date
4.0.1	0	Central	3,100	Apr, 2018
4.0.0	0	Central	410	Aug, 2017
4.0.0-507	0	Central	22	Jun, 2017
4.0.0-505	0	Central	0	May, 2017
4.0.x	0	Central	4	Mar, 2017
4.0.0-504	0	Central	0	Mar, 2017
4.0.0-503	2	Central	2	Mar, 2017
4.0.0-502	3	Central	3	Feb, 2017

3.5 Copy the dependency from the results shown

The screenshot shows the Maven Repository website for the Java Servlet API version 4.0.1. The dependency XML snippet is highlighted and copied to the clipboard:

```
<dependency>
  <groupId>javax.servlet</groupId>
  <artifactId>javax.servlet-api</artifactId>
  <version>4.0.1</version>
</dependency>
```


3.6 Paste it under the **dependency** in the **pom.xml** file



The screenshot shows the Visual Studio Code interface with the `pom.xml` file open. The Explorer view on the left shows the project structure: `CMS-WEB` with subfolders `src/main`, `resources`, `webapp`, `WEB-INF`, `target`, and files `Index.jsp` and `pom.xml`. The `pom.xml` file is selected, and its content is displayed in the editor. The XML content is as follows:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3         xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
4     <modelVersion>4.0.0</modelVersion>
5     <groupId>com.example.cms-web</groupId>
6     <artifactId>cms-web</artifactId>
7     <packaging>war</packaging>
8     <version>1.0-SNAPSHOT</version>
9     <name>cms-web Maven Webapp</name>
10    <url>http://maven.apache.org</url>
11    <dependencies>
12        <dependency>
13            <groupId>junit</groupId>
14            <artifactId>junit</artifactId>
15            <version>3.8.1</version>
16            <scope>test</scope>
17        </dependency>
18        <!-- https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api -->
19        <dependency>
20            <groupId>javax.servlet</groupId>
21            <artifactId>javax.servlet-api</artifactId>
22            <version>4.0.1</version>
23            <scope>provided</scope>
24        </dependency>
25    </dependencies>
26    <build>
27        <finalName>cms-web</finalName>
28    </build>
29 </project>
```

The new dependency block (lines 19-23) is highlighted with a red rectangle, indicating it has been added to the list of dependencies.

This is how you can easily add dependencies for the Servlet API. The same process can be used for other dependencies, such as Spring.

By following these steps, you have successfully created a Java Web Project using Maven CLI, built the project, opened it in Visual Studio Code, and added the necessary dependencies.