java -cp target\one-0.0.1-SNAPSHOT.jar logger.LogTest

**1. Install Maven**

First, ensure that Maven is installed on your system. You can download Maven from the [Apache Maven website](https://maven.apache.org/download.cgi). After downloading, unzip it and add the bin directory to your system's PATH.

To verify Maven installation, open Command Prompt (cmd) and run:

cmd

mvn -v

You should see version information if Maven is installed correctly.

**2. Navigate to Your Project Directory**

Open Command Prompt and navigate to the directory containing your pom.xml file:

cmd

cd path\to\your\project

**3. Compile Your Project**

Use Maven to compile your project. Maven will automatically handle dependencies specified in your pom.xml and compile the source code. Run:

cmd

mvn compile

This command will compile the source code and place the compiled classes in the target directory.

**4. Package Your Project**

If you want to create a JAR file, use the package goal. This command will compile, run tests, and package your application into a JAR file:

cmd

mvn package

The resulting JAR file will be located in the target directory, typically named something like your-project-name-version.jar.

**5. Run Your Project**

To run your application, use the java -cp command with the classpath pointing to the target directory. For example:

cmd

java -cp target\your-project-name-version.jar com.example.Main

java -cp target\one-0.0.1-SNAPSHOT.jar logger.LogTest

Replace com.example.Main with the fully qualified name of your main class (including the package). If you need to include other dependencies, Maven will handle this for you when you use the package command.

**Running a JAR with Dependencies**

If you want to run the JAR with all dependencies bundled together, you might need to use the Maven Shade Plugin or a similar tool to create an "uber-JAR" (a JAR with all dependencies included).

Here's how to configure it:

1. **Add the Maven Shade Plugin to pom.xml**:

xml

複製程式碼

<build>

<plugins>

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-shade-plugin</artifactId>

<version>3.2.4</version>

<executions>

<execution>

<phase>package</phase>

<goals>

<goal>shade</goal>

</goals>

<configuration>

<createDependencyReducedPom>false</createDependencyReducedPom>

<filters>

<filter>

<artifact>\*:\*</artifact>

<excludes>

<exclude>META-INF/\*.SF</exclude>

<exclude>META-INF/\*.DSA</exclude>

<exclude>META-INF/\*.RSA</exclude>

</excludes>

</filter>

</filters>

</configuration>

</execution>

</executions>

</plugin>

</plugins>

</build>

1. **Repackage Your Project**:

Run the following command to build the JAR with all dependencies included:

cmd

mvn package

The final JAR will be in the target directory and can be run with:

cmd

java -jar target/your-project-name-version.jar

**run-project.bat**

1. @echo off
2. REM Set the project root directory
3. set PROJECT\_DIR=C:\poi
4. REM Set the JAR file name
5. set JAR\_NAME=one-0.0.1-SNAPSHOT.jar
6. REM Navigate to the project directory
7. cd /d %PROJECT\_DIR%
8. REM Clean any previous builds
9. ::echo Cleaning project...
10. ::mvn clean
11. REM Compile and package the project
12. ::echo Building project...
13. ::mvn package
14. REM Check if the JAR file exists
15. if not exist target\%JAR\_NAME% (
16. echo Error: JAR file not found!
17. exit /b 1
18. )
19. REM Run the JAR file
20. echo Running project...
21. java -cp target\%JAR\_NAME% logger.LogTest
22. REM Pause to see the output
23. pause