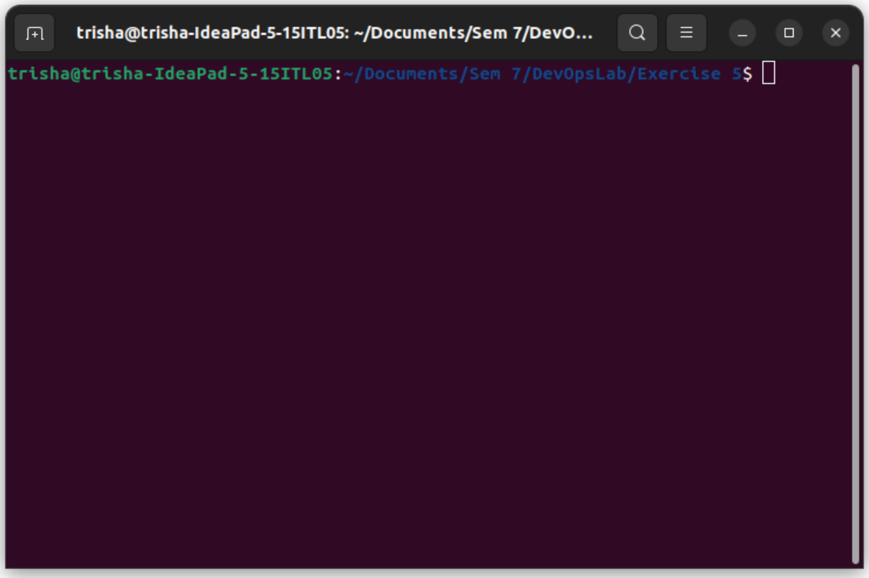
DevOps Lab - Trisha Balakrishnan

Exercise 7

Creating and Managing a Java Project Manually and with Maven

1. Creating a Java Project (Without Maven)

1. **Open Terminal** on your laptop.



```
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5$
```

Figure 1: 1-1

2. Navigate to your desired directory:

   ```
   cd path/to/your/directory
   ```

3. Create a new directory for your Java project:

   ```
   mkdir MyJavaProject
   cd MyJavaProject
   ```

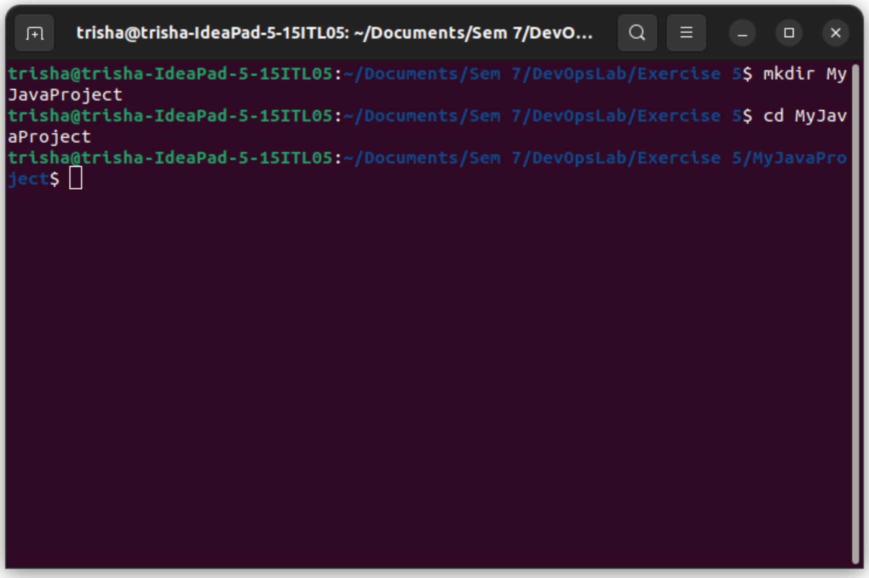
4. Create the necessary directory structure for a Java project:

   ```
   mkdir -p src/main/java/com/example
   ```

5. Create a simple Java class inside the `src/main/java/com/example` directory:

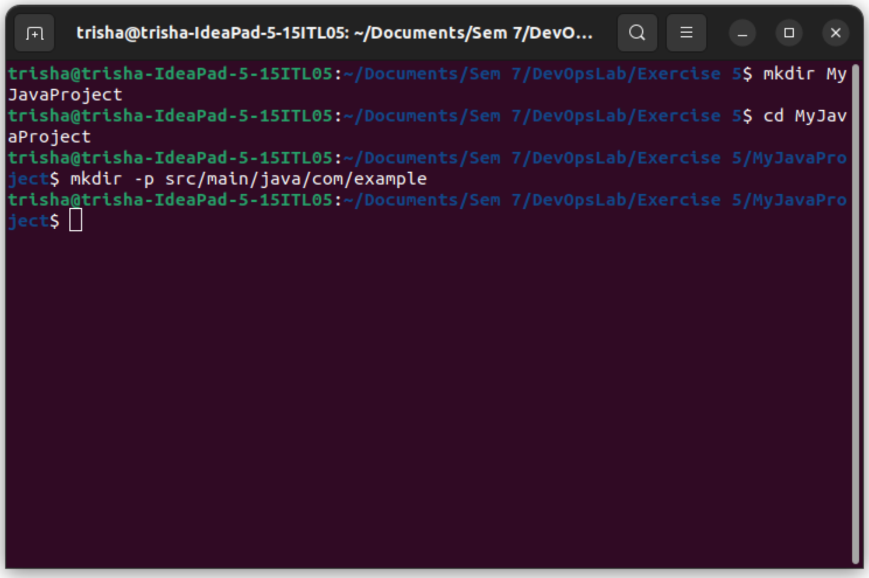
   ```
   nano src/main/java/com/example/App.java
   ```

   Add the following code to the file:



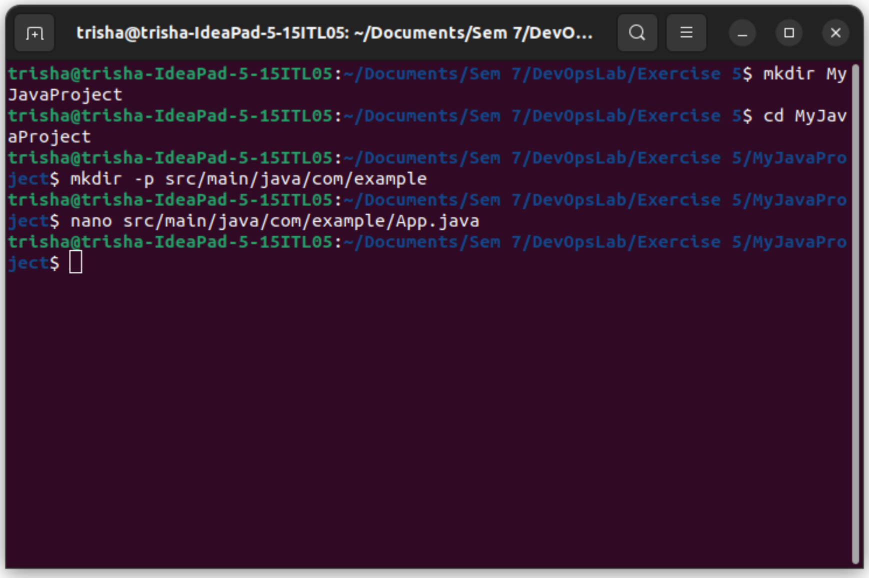
```
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5$ mkdir My
JavaProject
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5$ cd MyJav
aProject
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ 
```

Figure 2: 1-3



```
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5$ mkdir My
JavaProject
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5$ cd MyJav
aProject
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ mkdir -p src/main/java/com/example
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ 
```

Figure 3: 1-4



```
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5$ mkdir My
JavaProject
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5$ cd MyJav
aProject
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ mkdir -p src/main/java/com/example
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ nano src/main/java/com/example/App.java
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ 
```

Figure 4: 1-5

```java
package com.example;

public class App {
    public static void main(String[] args) {
        System.out.println("Hello, World!");
    }
}
```

Save the file and exit the editor.

2. Packaging the Project with Dependencies (Without Maven)

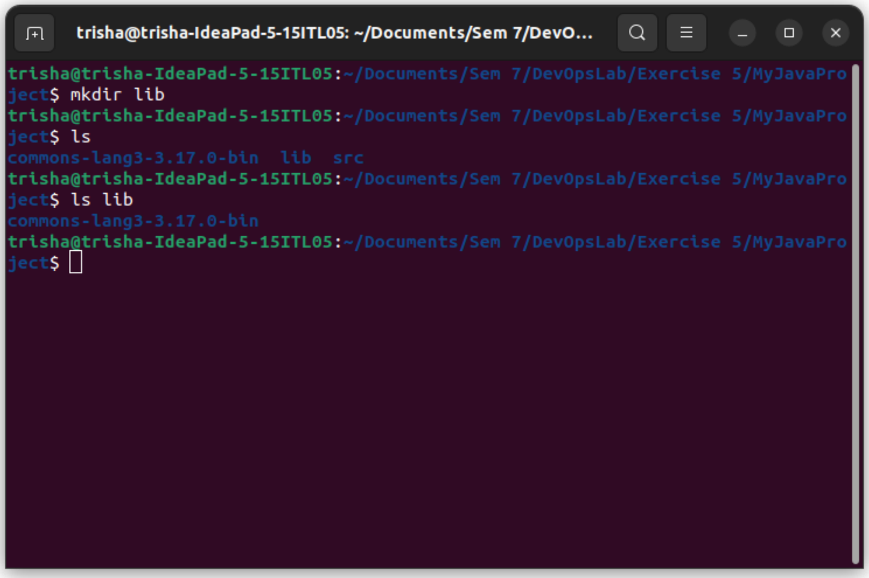
1. If you need external libraries (dependencies), you'll have to download the JAR files manually and include them in your project. For example, download a library like `commons-lang3` from Apache Commons.

2. Place the downloaded JAR files in a `lib` directory within your project:

```
mkdir lib
cp path/to/commons-lang3-3.12.0.jar lib/
```

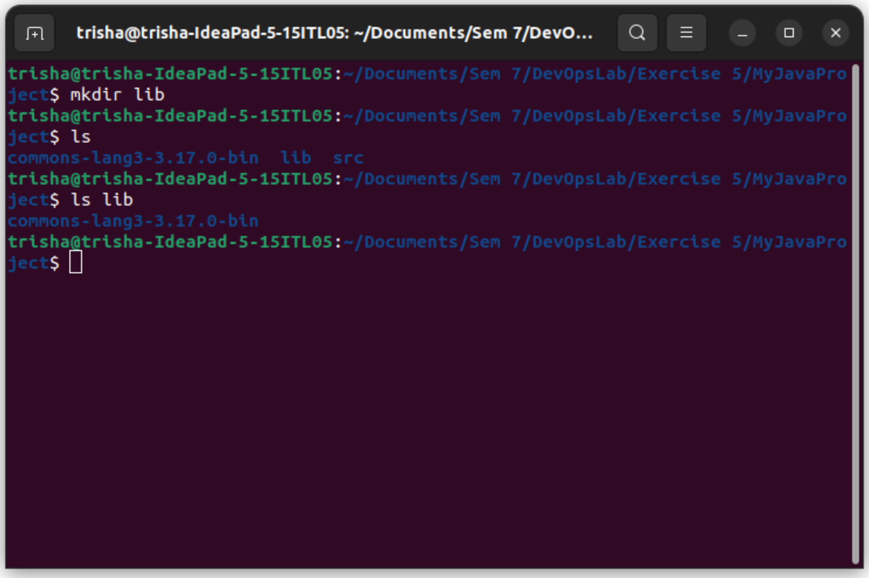
3. Compile the project with the dependencies:

```
javac -cp lib/commons-lang3-3.12.0.jar -d . src/main/java/com/example/App.java
```



```
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ mkdir lib
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ ls
commons-lang3-3.17.0-bin  lib  src
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ ls lib
commons-lang3-3.17.0-bin
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ 
```

Figure 5: 2-2



```
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ mkdir lib
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ ls
commons-lang3-3.17.0-bin  lib  src
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ ls lib
commons-lang3-3.17.0-bin
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ 
```

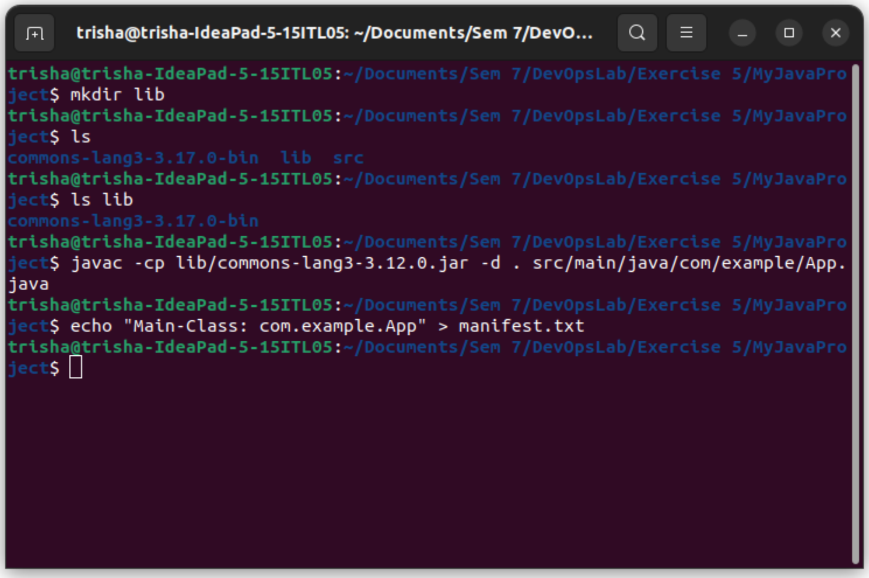
Figure 6: 2-3

3. Compiling and Building the JAR File Manually (Artifact 1 without Maven)

1. **Create a Manifest File:**

   Create a `manifest.txt` file:

   ```
   echo "Main-Class: com.example.App" > manifest.txt
   ```



```
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ mkdir lib
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ ls
commons-lang3-3.17.0-bin  lib  src
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ ls lib
commons-lang3-3.17.0-bin
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ javac -cp lib/commons-lang3-3.12.0.jar -d . src/main/java/com/example/App.
java
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ echo "Main-Class: com.example.App" > manifest.txt
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ 
```

Figure 7: 3-1

2. **Package the JAR File:**

   Use the `jar` command to package the compiled classes and the manifest file into a JAR:

   ```
   jar cvfm MyJavaProject.jar manifest.txt com/example/*.class -C lib .
   ```

3. **Verify the JAR File:**

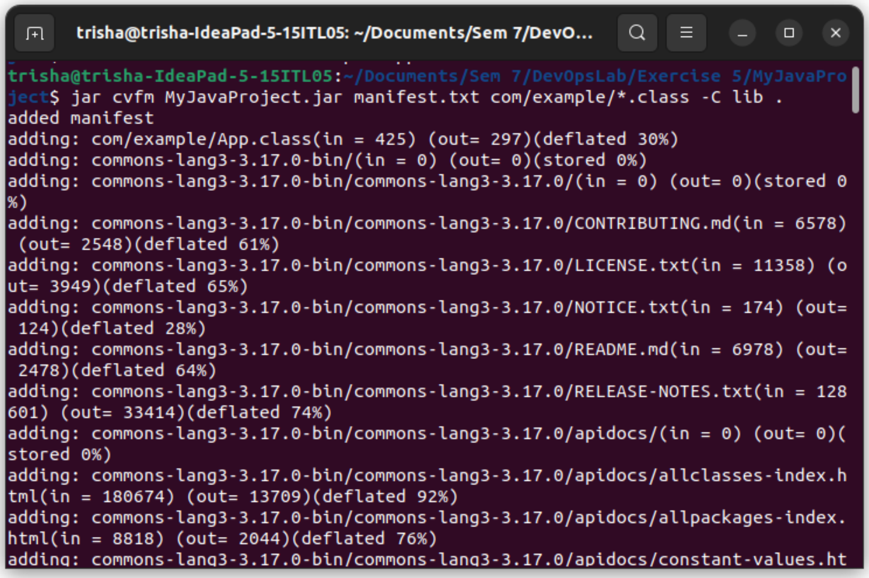
   Check that the JAR file has been created successfully:

   ```
   ls MyJavaProject.jar
   ```

4. **Run the JAR File:**

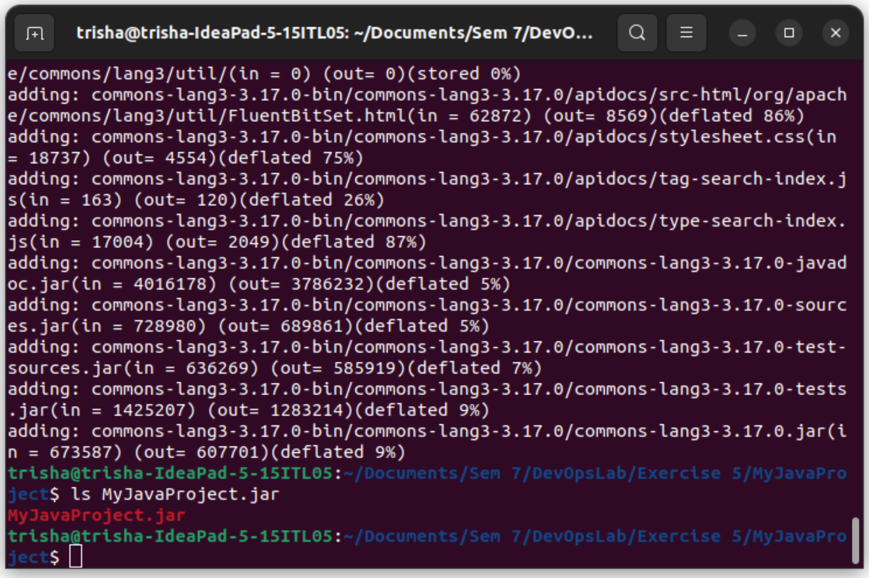
   To run your JAR file, use:

   ```
   java -cp MyJavaProject.jar:lib/commons-lang3-3.12.0.jar com.example.App
   ```



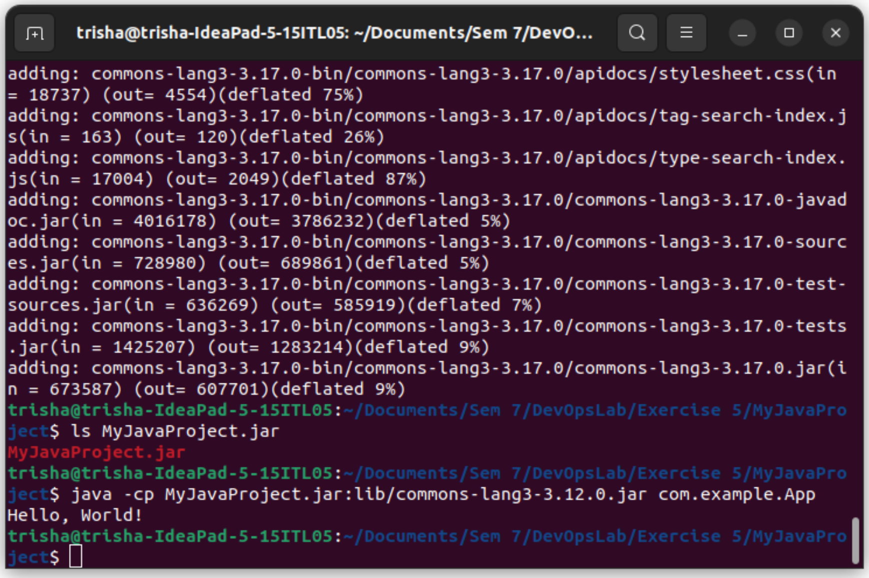
```
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ jar cvfm MyJavaProject.jar manifest.txt com/example/*.class -C lib .
added manifest
adding: com/example/App.class(in = 425) (out= 297)(deflated 30%)
adding: commons-lang3-3.17.0-bin/(in = 0) (out= 0)(stored 0%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/(in = 0) (out= 0)(stored 0
%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/CONTRIBUTING.md(in = 6578)
 (out= 2548)(deflated 61%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/LICENSE.txt(in = 11358) (o
ut= 3949)(deflated 65%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/NOTICE.txt(in = 174) (out=
 124)(deflated 28%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/README.md(in = 6978) (out=
 2478)(deflated 64%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/RELEASE-NOTES.txt(in = 128
601) (out= 33414)(deflated 74%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/apidocs/(in = 0) (out= 0)(
stored 0%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/apidocs/allclasses-index.h
tml(in = 180674) (out= 13709)(deflated 92%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/apidocs/allpackages-index.
html(in = 8818) (out= 2044)(deflated 76%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/apidocs/constant-values.ht
```

Figure 8: 3-2



```
e/commons/lang3/util/(in = 0) (out= 0)(stored 0%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/apidocs/src-html/org/apach
e/commons/lang3/util/FluentBitSet.html(in = 62872) (out= 8569)(deflated 86%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/apidocs/stylesheet.css(in
= 18737) (out= 4554)(deflated 75%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/apidocs/tag-search-index.j
s(in = 163) (out= 120)(deflated 26%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/apidocs/type-search-index.
js(in = 17004) (out= 2049)(deflated 87%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/commons-lang3-3.17.0-javad
oc.jar(in = 4016178) (out= 3786232)(deflated 5%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/commons-lang3-3.17.0-sourc
es.jar(in = 728980) (out= 689861)(deflated 5%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/commons-lang3-3.17.0-test-
sources.jar(in = 636269) (out= 585919)(deflated 7%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/commons-lang3-3.17.0-tests
.jar(in = 1425207) (out= 1283214)(deflated 9%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/commons-lang3-3.17.0.jar(i
n = 673587) (out= 607701)(deflated 9%)
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ ls MyJavaProject.jar
MyJavaProject.jar
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ 
```

Figure 9: 3-3



```
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/apidocs/stylesheet.css(in
= 18737) (out= 4554)(deflated 75%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/apidocs/tag-search-index.j
s(in = 163) (out= 120)(deflated 26%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/apidocs/type-search-index.
js(in = 17004) (out= 2049)(deflated 87%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/commons-lang3-3.17.0-javad
oc.jar(in = 4016178) (out= 3786232)(deflated 5%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/commons-lang3-3.17.0-sourc
es.jar(in = 728980) (out= 689861)(deflated 5%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/commons-lang3-3.17.0-test-
sources.jar(in = 636269) (out= 585919)(deflated 7%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/commons-lang3-3.17.0-tests
.jar(in = 1425207) (out= 1283214)(deflated 9%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/commons-lang3-3.17.0.jar(i
n = 673587) (out= 607701)(deflated 9%)
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ ls MyJavaProject.jar
MyJavaProject.jar
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ java -cp MyJavaProject.jar:lib/commons-lang3-3.12.0.jar com.example.App
Hello, World!
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$
```

Figure 10: 3-4

4. Checking Your Installed Maven Version

1. Now that you've manually built the project, check your Maven version:

   ```
   mvn -v
   ```

5. Creating `pom.xml`

1. **Initialize the Maven Project:**

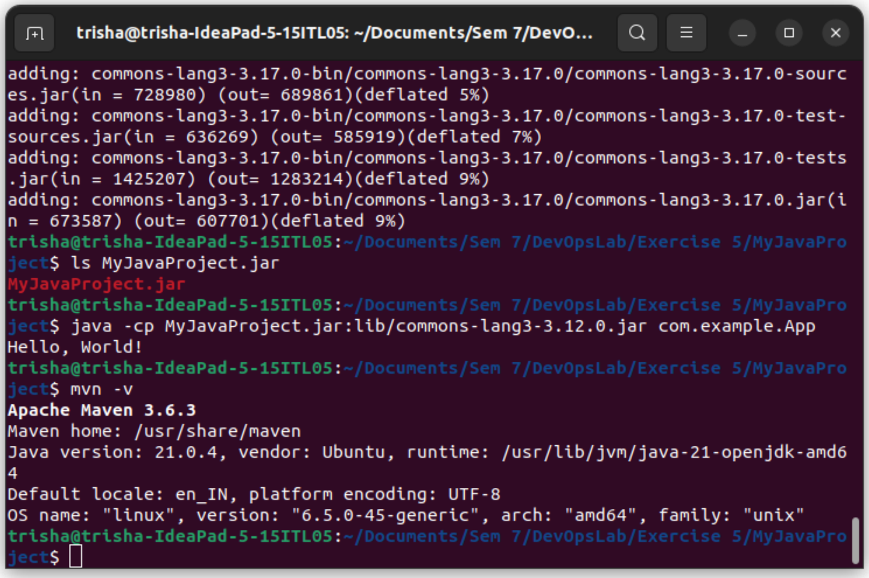
   Move your project to a Maven structure:

   ```
   mvn archetype:generate -DgroupId=com.example -DartifactId=MyJavaProject -DarchetypeArti
   ```

2. **Copy Your Files** into the newly generated Maven structure.

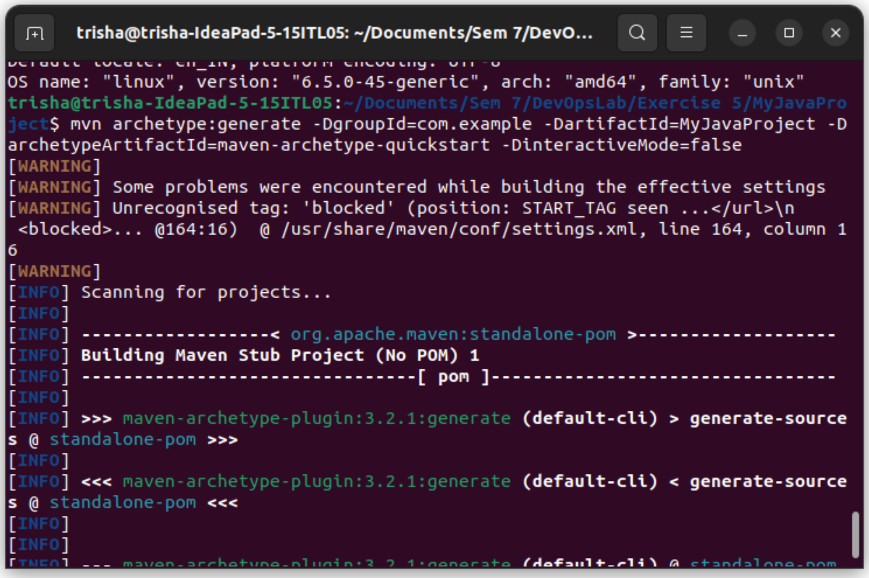
3. **Add Dependencies** in the `pom.xml`:

   ```
   <dependencies>
       <dependency>
           <groupId>org.apache.commons</groupId>
           <artifactId>commons-lang3</artifactId>
           <version>3.12.0</version>
       </dependency>
   </dependencies>
   ```



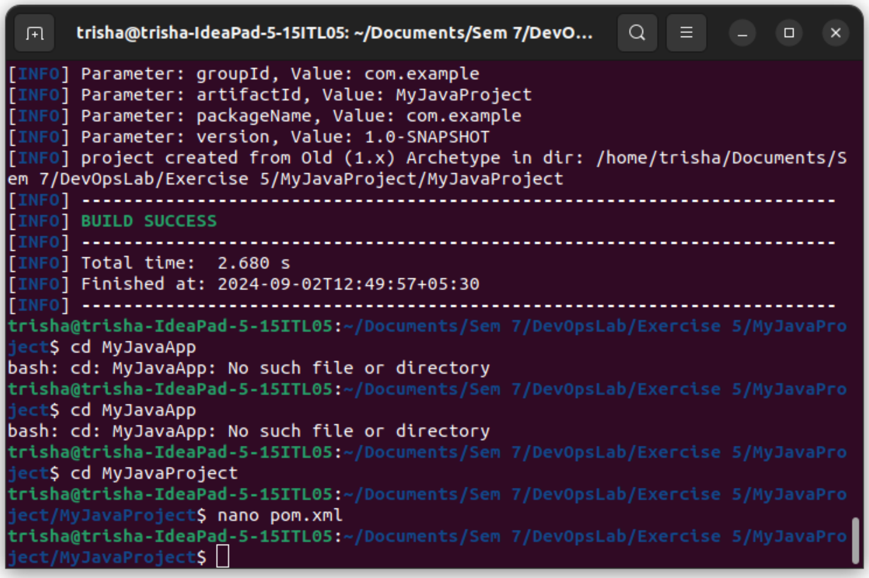
```
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/commons-lang3-3.17.0-sourc
es.jar(in = 728980) (out= 689861)(deflated 5%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/commons-lang3-3.17.0-test-
sources.jar(in = 636269) (out= 585919)(deflated 7%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/commons-lang3-3.17.0-tests
.jar(in = 1425207) (out= 1283214)(deflated 9%)
adding: commons-lang3-3.17.0-bin/commons-lang3-3.17.0/commons-lang3-3.17.0.jar(i
n = 673587) (out= 607701)(deflated 9%)
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ ls MyJavaProject.jar
MyJavaProject.jar
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ java -cp MyJavaProject.jar:lib/commons-lang3-3.12.0.jar com.example.App
Hello, World!
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ mvn -v
Apache Maven 3.6.3
Maven home: /usr/share/maven
Java version: 21.0.4, vendor: Ubuntu, runtime: /usr/lib/jvm/java-21-openjdk-amd6
4
Default locale: en_IN, platform encoding: UTF-8
OS name: "linux", version: "6.5.0-45-generic", arch: "amd64", family: "unix"
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$
```

Figure 11: 4-1



```
Default locale: en_IN, platform encoding: UTF-8
OS name: "linux", version: "6.5.0-45-generic", arch: "amd64", family: "unix"
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ mvn archetype:generate -DgroupId=com.example -DartifactId=MyJavaProject -D
archetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false
[WARNING]
[WARNING] Some problems were encountered while building the effective settings
[WARNING] Unrecognised tag: 'blocked' (position: START_TAG seen ...</url>\n
 <blocked>... @164:16)  @ /usr/share/maven/conf/settings.xml, line 164, column 1
6
[WARNING]
[INFO] Scanning for projects...
[INFO]
[INFO] ------------------< org.apache.maven:standalone-pom >-------------------
[INFO] Building Maven Stub Project (No POM) 1
[INFO] --------------------------------[ pom ]---------------------------------
[INFO]
[INFO] >>> maven-archetype-plugin:3.2.1:generate (default-cli) > generate-source
s @ standalone-pom >>>
[INFO]
[INFO] <<< maven-archetype-plugin:3.2.1:generate (default-cli) < generate-source
s @ standalone-pom <<<
[INFO]
[INFO]
[INFO] --- maven-archetype-plugin:3.2.1:generate (default-cli) @ standalone-pom
```

Figure 12: 5-1



```
[INFO] Parameter: groupId, Value: com.example
[INFO] Parameter: artifactId, Value: MyJavaProject
[INFO] Parameter: packageName, Value: com.example
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[INFO] project created from Old (1.x) Archetype in dir: /home/trisha/Documents/S
em 7/DevOpsLab/Exercise 5/MyJavaProject/MyJavaProject
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  2.680 s
[INFO] Finished at: 2024-09-02T12:49:57+05:30
[INFO] ------------------------------------------------------------------------
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ cd MyJavaApp
bash: cd: MyJavaApp: No such file or directory
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ cd MyJavaApp
bash: cd: MyJavaApp: No such file or directory
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject$ cd MyJavaProject
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject/MyJavaProject$ nano pom.xml
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject/MyJavaProject$ 
```

Figure 13: 5-3

6. Executing the Clean Lifecycle

1. Run the `clean` lifecycle in Maven:

   ```
   mvn clean
   ```

2. Verify that the `target` directory and its contents are removed:

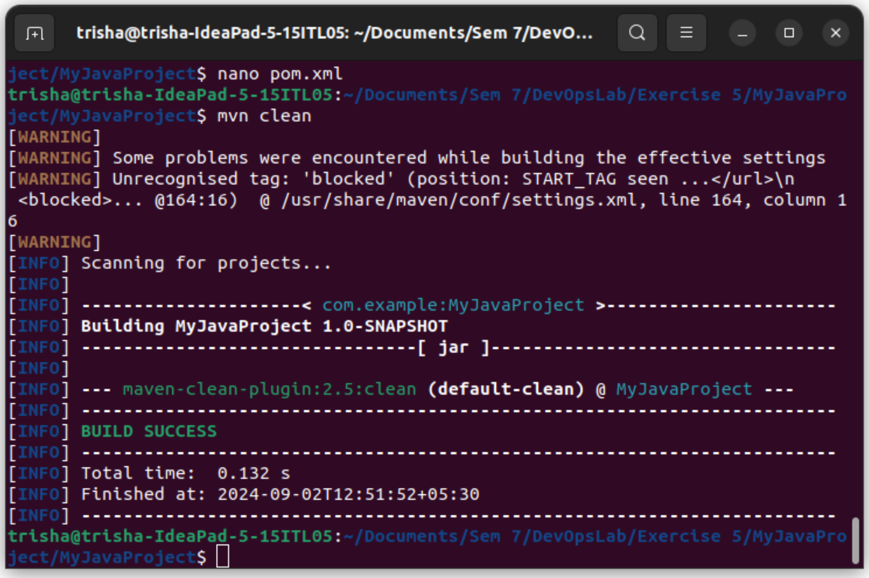
   ```
   ls target/
   ```

7. Exploring Plugins and Goals Associated with `clean`

1. Explore the plugins and goals associated with the `clean` lifecycle:

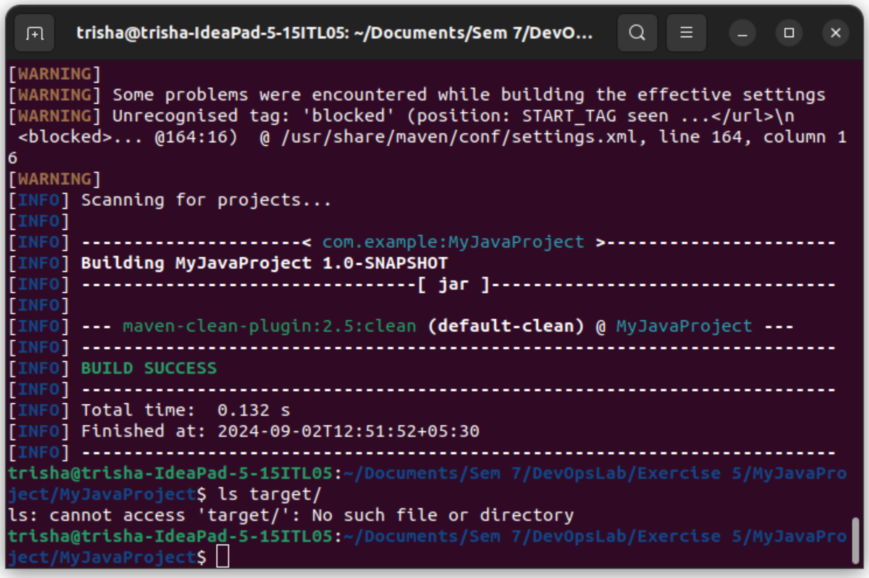
   ```
   mvn help:describe -Dcmd=clean
   ```

2. Review the output to understand the available goals and plugins.



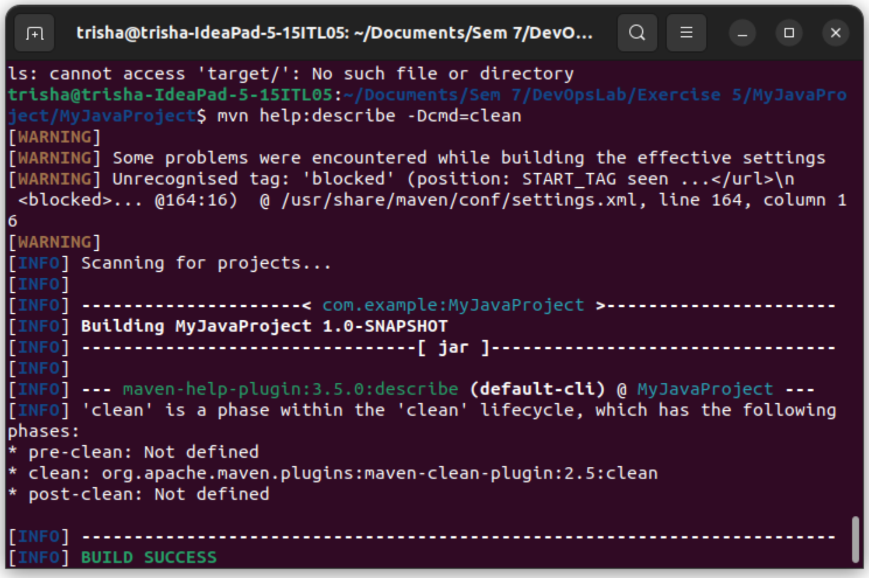
```
ject/MyJavaProject$ nano pom.xml
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject/MyJavaProject$ mvn clean
[WARNING]
[WARNING] Some problems were encountered while building the effective settings
[WARNING] Unrecognised tag: 'blocked' (position: START_TAG seen ...</url>\n
 <blocked>... @164:16)  @ /usr/share/maven/conf/settings.xml, line 164, column 1
6
[WARNING]
[INFO] Scanning for projects...
[INFO]
[INFO] --------------------< com.example:MyJavaProject >---------------------
[INFO] Building MyJavaProject 1.0-SNAPSHOT
[INFO] --------------------------------[ jar ]---------------------------------
[INFO]
[INFO] --- maven-clean-plugin:2.5:clean (default-clean) @ MyJavaProject ---
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  0.132 s
[INFO] Finished at: 2024-09-02T12:51:52+05:30
[INFO] ------------------------------------------------------------------------
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject/MyJavaProject$ 
```

Figure 14: 6-1



```
[WARNING]
[WARNING] Some problems were encountered while building the effective settings
[WARNING] Unrecognised tag: 'blocked' (position: START_TAG seen ...</url>\n
 <blocked>... @164:16)  @ /usr/share/maven/conf/settings.xml, line 164, column 1
6
[WARNING]
[INFO] Scanning for projects...
[INFO]
[INFO] --------------------< com.example:MyJavaProject >---------------------
[INFO] Building MyJavaProject 1.0-SNAPSHOT
[INFO] --------------------------------[ jar ]---------------------------------
[INFO]
[INFO] --- maven-clean-plugin:2.5:clean (default-clean) @ MyJavaProject ---
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  0.132 s
[INFO] Finished at: 2024-09-02T12:51:52+05:30
[INFO] ------------------------------------------------------------------------
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject/MyJavaProject$ ls target/
ls: cannot access 'target/': No such file or directory
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject/MyJavaProject$ 
```

Figure 15: 6-2



```
ls: cannot access 'target/': No such file or directory
trisha@trisha-IdeaPad-5-15ITL05:~/Documents/Sem 7/DevOpsLab/Exercise 5/MyJavaPro
ject/MyJavaProject$ mvn help:describe -Dcmd=clean
[WARNING]
[WARNING] Some problems were encountered while building the effective settings
[WARNING] Unrecognised tag: 'blocked' (position: START_TAG seen ...</url>\n
 <blocked>... @164:16)  @ /usr/share/maven/conf/settings.xml, line 164, column 1
6
[WARNING]
[INFO] Scanning for projects...
[INFO]
[INFO] --------------------< com.example:MyJavaProject >---------------------
[INFO] Building MyJavaProject 1.0-SNAPSHOT
[INFO] --------------------------------[ jar ]--------------------------------
[INFO]
[INFO] --- maven-help-plugin:3.5.0:describe (default-cli) @ MyJavaProject ---
[INFO] 'clean' is a phase within the 'clean' lifecycle, which has the following
phases:
* pre-clean: Not defined
* clean: org.apache.maven.plugins:maven-clean-plugin:2.5:clean
* post-clean: Not defined

[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
```

Figure 16: 7-1