TASK 5

Name: Swetha M

Rollno: 22CSR217

Step 1: Create a folder and navigate to the foder and install the git package using the following command:

mkdir pet

cd pet

apt install git

```
root@Swetha:~# mkdir pet
root@Swetha:~# cd pet
root@Swetha:~/pet# apt install git
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  git-man
Suggested packages:
  git-daemon-run | git-daemon-sysvinit git-doc git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn
The following packages will be upgraded:
  git git-man
2 upgraded, 0 newly installed, 0 to remove and 129 not upgraded.
Need to get 4779 kB of archives.
After this operation, 2048 B of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 git-man all 1:2.43.0-1ubuntu7.2 [1100 kB]
Get:2 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 git amd64 1:2.43.0-1ubuntu7.2 [3679 kB]
Fetched 4779 kB in 7s (717 kB/s)
(Reading database ... 42856 files and directories currently installed.)
Preparing to unpack .../git-man_1%3a2.43.0-1ubuntu7.2_all.deb ...
Unpacking git-man (1:2.43.0-1ubuntu7.2) over (1:2.43.0-1ubuntu7.1) ...
Preparing to unpack .../git_1%3a2.43.0-1ubuntu7.2_amd64.deb ...
Unpacking git (1:2.43.0-1ubuntu7.2) over (1:2.43.0-1ubuntu7.1) ...
Setting up git-man (1:2.43.0-1ubuntu7.2) ...
Setting up git (1:2.43.0-1ubuntu7.2) ...
Processing triggers for man-db (2.12.0-4build2) ...
root@Swetha:~/pet# git clone https://github.com/AranganathanPrakash/spring-framework-petclinic
Cloning into 'spring-framework-petclinic'...
remote: Enumerating objects: 7351, done.
remote: Counting objects: 100% (1111/1111), done.
remote: Compressing objects: 100% (78/78), done.
remote: Total 7351 (delta 1060), reused 1033 (delta 1033), pack-reused 6240 (from 1)
Receiving objects: 100% (7351/7351), 3.12 MiB | 3.74 MiB/s, done.
Resolving deltas: 100% (3600/3600), done.
root@Swetha:~/pet# ls
```

Step 2: By using the ls command check whether the required directory is exist or not

```
root@Swetha:~/pet# ls
spring-framework-petclinic
```

Step 3: Move to the directory and give ls command to check the file exist

```
root@Swetha:~/pet# cd spring-framework-petclinic
root@Swetha:~/pet/spring-framework-petclinic# ls
Jenkinsfile  LICENSE.txt  dockerfile  mvnw  mvnw.cmd  pom.xml  readme.md  src
```

Step 4: Install the maven package using the command:

install maven

```
root@Swetha:~/pet/spring-framework-petclinic# install maven
install: missing destination file operand after 'maven'
Try 'install --help' for more information.
root@Swetha:~/pet/spring-framework-petclinic# sudo apt install maven
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libaopalliance-java libapache-pom-java libatinject-jsr330-api-java libcdi-api-java libcommons-cli-java
  libcommons-io-java libcommons-lang3-java libcommons-parent-java liberror-prone-java
  libgeronimo-annotation-1.3-spec-java libgeronimo-interceptor-3.0-spec-java libguava-java libguice-java libjansi-java
  libjsr305-java libmaven-parent-java libmaven-resolver-java libmaven-shared-utils-java libmaven3-core-java
  libplexus-cipher-java libplexus-classworlds-java libplexus-component-annotations-java libplexus-interpolation-java
  libplexus-sec-dispatcher-java libplexus-utils2-java libsisu-inject-java libsisu-plexus-java libslf4j-java
  libwagon-file-java libwagon-http-shaded-java libwagon-provider-api-java
Suggested packages:
  libatinject-jsr330-api-java-doc libel-api-java libcommons-io-java-doc libasm-java libcglib-java libjsr305-java-doc
  libmaven-shared-utils-java-doc liblogback-java libplexus-utils2-java-doc junit4 testng libcommons-logging-java
  liblog4j1.2-java
The following NEW packages will be installed:
  libaopalliance-java libapache-pom-java libatinject-jsr330-api-java libcdi-api-java libcommons-cli-java
  libcommons-io-java libcommons-lang3-java libcommons-parent-java liberror-prone-java
  libgeronimo-annotation-1.3-spec-java libgeronimo-interceptor-3.0-spec-java libguava-java libguice-java libjansi-java
  libjsr305-java libmaven-parent-java libmaven-resolver-java libmaven-shared-utils-java libmaven3-core-java
  libplexus-cipher-java libplexus-classworlds-java libplexus-component-annotations-java libplexus-interpolation-java
  libplexus-sec-dispatcher-java libplexus-utils2-java libsisu-inject-java libsisu-plexus-java libslf4j-java
  libwagon-file-java libwagon-http-shaded-java libwagon-provider-api-java maven
```

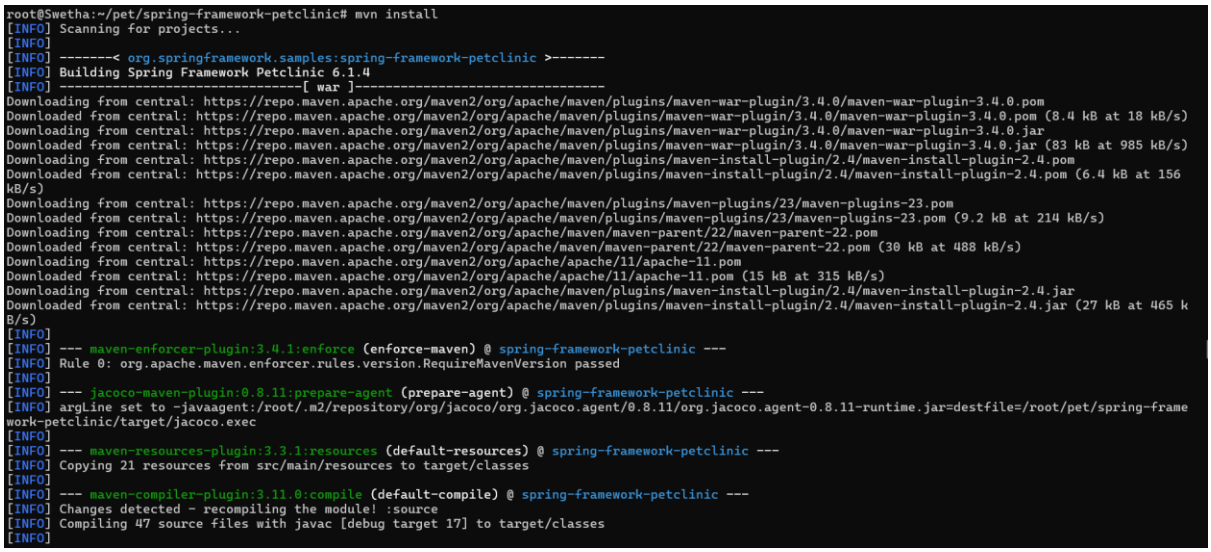
Step 5: Clean the maven packages using the command:

mvn clean

```
root@Swetha:~/pet/spring-framework-petclinic# mvn clean
[INFO] Scanning for projects...
[INFO]
[INFO] -------< org.springframework.samples:spring-framework-petclinic >-------
[INFO] Building Spring Framework Petclinic 6.1.4
[INFO] --------------------------------[ war ]---------------------------------
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/2.5/maven-clean-plugin-2.5.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/2.5/maven-clean-plugin-2.5.pom (3.9 kB at 14 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-plugins/22/maven-plugins-22.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-plugins/22/maven-plugins-22.pom (13 kB at 296 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-parent/21/maven-parent-21.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-parent/21/maven-parent-21.pom (26 kB at 527 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/apache/10/apache-10.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/apache/10/apache-10.pom (15 kB at 361 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/2.5/maven-clean-plugin-2.5.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/2.5/maven-clean-plugin-2.5.jar (25 kB at 560 kB/s)
[INFO]
[INFO] --- maven-clean-plugin:2.5:clean (default-clean) @ spring-framework-petclinic ---
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-plugin-api/2.0.6/maven-plugin-api-2.0.6.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-plugin-api/2.0.6/maven-plugin-api-2.0.6.pom (1.5 kB at 38 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven/2.0.6/maven-2.0.6.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven/2.0.6/maven-2.0.6.pom (9.0 kB at 215 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-parent/5/maven-parent-5.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-parent/5/maven-parent-5.pom (15 kB at 354 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/apache/3/apache-3.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/apache/3/apache-3.pom (3.4 kB at 95 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.0/plexus-utils-3.0.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.0/plexus-utils-3.0.pom (4.1 kB at 110 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/sonatype/spice/spice-parent/16/spice-parent-16.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/sonatype/spice/spice-parent/16/spice-parent-16.pom (8.4 kB at 164 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-plugin-api/2.0.6/maven-plugin-api-2.0.6.jar
Downloading from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.0/plexus-utils-3.0.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-plugin-api/2.0.6/maven-plugin-api-2.0.6.jar (13 kB at 306 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.0/plexus-utils-3.0.jar (226 kB at 1.2 MB/s)
```

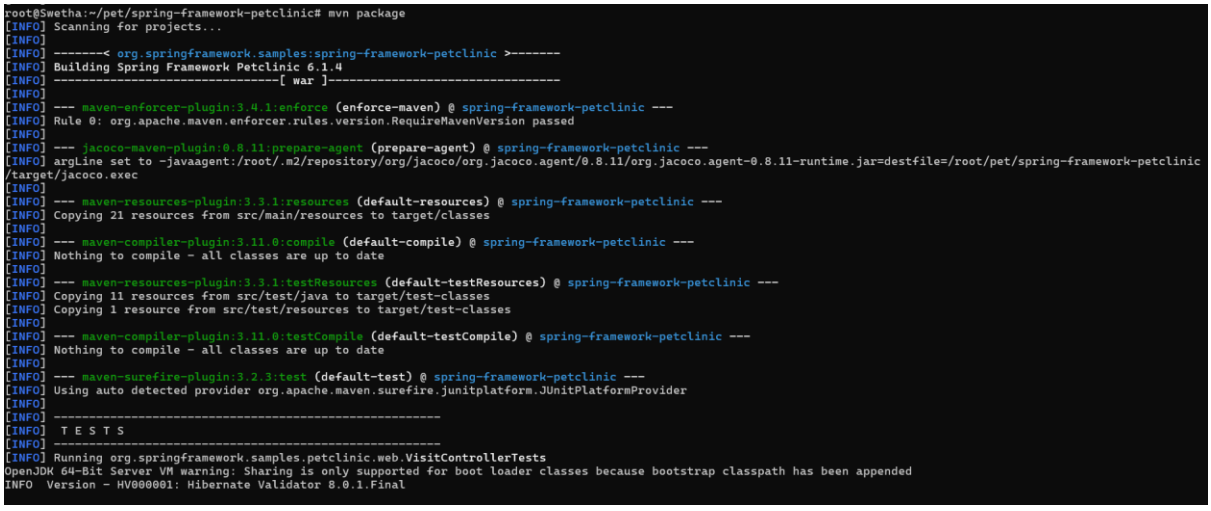
Step 6: Intsall maven using the following command

mvn install



Step 7: Install the maven packages using the command

mvn package



Step 8: Change the directory to target and check whether the .war file exists using the commands such as

cd target

ls

Step 9: Now build the image using the docker by the command

docker build -t <imagename>

Step 10: Now tag the builded image in the dockerhub and then login with docker to push the builded image in dockerhub by the commands such as

docker tag <imagename> <dockerhubusername>/<reponame>

docker login

docker push <dockerhubusername>/<reponame>

```
root@Swetha:~/pet/spring-framework-petclinic# docker tag pet swethamurugesan/devopsgit:latest
root@Swetha:~/pet/spring-framework-petclinic# docker login
Authenticating with existing credentials...
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
root@Swetha:~/pet/spring-framework-petclinic# docker push swethamurugesan/devopsgit:latest
The push refers to repository [docker.io/swethamurugesan/devopsgit]
d50f97ba3545: Pushed
5f70bf18a086: Mounted from library/tomcat
6fbdf02a6a33: Mounted from library/tomcat
49cb1bc2daeb: Mounted from library/tomcat
4e5b554b7345: Mounted from library/tomcat
39cf0ac89a5a: Mounted from library/tomcat
f844dcf94898: Mounted from library/tomcat
3359bc3d7a6a: Mounted from library/tomcat
4b7c01ed0534: Mounted from library/tomcat
latest: digest: sha256:3183103d70e7d0fb7df6a4de7c0067d653dc2dfef4c37c4d387d849f6156585a size: 2413
```

Step 11: Now start the minikube using the command

minikube start

```
$ minikube start
😄  minikube v1.35.0 on Ubuntu 24.04 (amd64)
✨  Using the docker driver based on existing profile
👍  Starting "minikube" primary control-plane node in "minikube" cluster
🚜  Pulling base image v0.0.46 ...
🏃  Updating the running docker "minikube" container ...
🐳  Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...
🔎  Verifying Kubernetes components...
    ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🌟  Enabled addons: default-storageclass, storage-provisioner
🏄  Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
```

Step 12: Now deploy the created image with the help of minikube by the following command

kubectl create deployment <imagename> --image=<username>/<reponame> --port=<portno>

```
$ kubectl create deployment pet --image=swethamurugesan/devopsgit --port=8080
deployment.apps/pet created
```

Step 13: Expose the deployed image using the command

Kubectl expose deployement <imagename> --port=<portno> --type=NodePort

```
$ kubectl expose deployment pet --port=8080 --type=NodePort
service/pet exposed
```

Step 14: Use the URL to view the output in the browser.