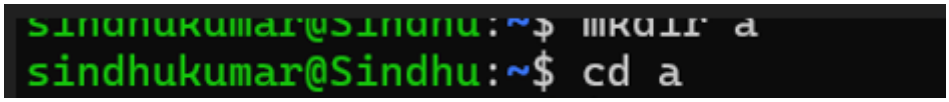
TASK 5 - MAVEN

NAME: SINDHU K

ROLL NO: 22CSR196

STEPS:

1. Create a folder and move into that folder.



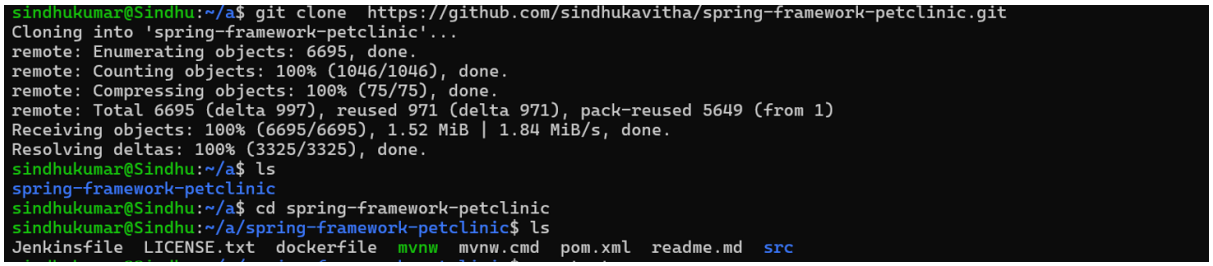
2. If Git is not already installed, install it using the following command:

   sudo apt install git

3. Clone the GitHub repository containing the project files.
4. Check the list of folders using the command:

   ls

5. Move into the cloned folder.



6. If Maven is not already installed, install it using the following command:

   sudo apt install maven

7. To verify the Maven installation, use the command:

   mvn –version (This checks the installed version.)

8. Execute the following commands one by one:
   - mvn test → Runs unit tests and generates a report for the project.
   - mvn clean → Deletes the previous Maven build.

- mvn install → Installs all the required plugins and packages.
- mvn package → Provides the jar or war file for the entire application.

```
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ mvn test
[INFO] Scanning for projects...
[INFO]
[INFO] -------< org.springframework.samples:spring-framework-petclinic >-------
[INFO] Building Spring Framework Petclinic 6.1.4
[INFO]   from pom.xml
[INFO] --------------------------------[ war ]---------------------------------
[INFO]
[INFO] --- enforcer:3.4.1:enforce (enforce-maven) @ spring-framework-petclinic ---
[INFO] Rule 0: org.apache.maven.enforcer.rules.version.RequireMavenVersion passed
[INFO]
[INFO] --- jacoco:0.8.11:prepare-agent (prepare-agent) @ spring-framework-petclinic ---
[INFO] argLine set to -javaagent:/home/sindhukumar/.m2/repository/org/jacoco/org.jacoco.agent/0.
e=/home/sindhukumar/a/spring-framework-petclinic/target/jacoco.exec
```

```
sindhukumar@Sindhu:~/a/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]   from pom.xml
[INFO] --------------------------------[ war ]---------------------------------
[INFO]
[INFO] --- clean:3.2.0:clean (default-clean) @ spring-framework-petclinic ---
[INFO] Deleting /home/sindhukumar/a/spring-framework-petclinic/target
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  0.281 s
[INFO] Finished at: 2025-03-21T09:40:29Z
[INFO] ------------------------------------------------------------------------
```

```
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ mvn install
[INFO] Scanning for projects...
[INFO]
[INFO] -------< org.springframework.samples:spring-framework-petclinic >-------
[INFO] Building Spring Framework Petclinic 6.1.4
[INFO]   from pom.xml
[INFO] --------------------------------[ war ]---------------------------------
[INFO]
[INFO] --- enforcer:3.4.1:enforce (enforce-maven) @ spring-framework-petclinic ---
[INFO] Rule 0: org.apache.maven.enforcer.rules.version.RequireMavenVersion passed
[INFO]
[INFO] --- jacoco:0.8.11:prepare-agent (prepare-agent) @ spring-framework-petclinic ---
[INFO] argLine set to -javaagent:/home/sindhukumar/.m2/repository/org/jacoco/org.jacoco.agent/0.8.11/org.jacoco.agent-0.8.11-runtime.jar=destfil
e=/home/sindhukumar/a/spring-framework-petclinic/target/jacoco.exec
[INFO]
```

9. Push the Docker image to DockerHub using the appropriate Docker commands.

```
[INFO] ------------------------------------------------------------------------
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ ls
Jenkinsfile  LICENSE.txt  dockerfile  mvnw  mvnw.cmd  pom.xml  readme.md  src  target
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ cd target
sindhukumar@Sindhu:~/a/spring-framework-petclinic/target$ ls
classes             generated-test-sources  maven-archiver  petclinic      site             test-classes
generated-sources   jacoco.exec             maven-status    petclinic.war  surefire-reports
sindhukumar@Sindhu:~/a/spring-framework-petclinic/target$ cd ..
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ cat dockerfile
FROM tomcat:latest
COPY target/*.war /usr/local/tomcat/webapps/ROOT.war
EXPOSE 8080
CMD ["catalina.sh", "run"]
```

```
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ docker build -t dog .
[+] Building 139.6s (8/8) FINISHED                                                                      docker:default
 => [internal] load build definition from dockerfile                                                              0.1s
 => => transferring dockerfile: 148B                                                                              0.0s
 => [internal] load metadata for docker.io/library/tomcat:latest                                                 24.5s
 => [auth] library/tomcat:pull token for registry-1.docker.io                                                     0.0s
 => [internal] load .dockerignore                                                                                 0.0s
 => => transferring context: 2B                                                                                   0.0s
 => [1/2] FROM docker.io/library/tomcat:latest@sha256:1374a565d5122fdb42807f3a5f2d4fcc245a5e15420ff5bb5123afedc8ef769d  114.0s
 => => resolve docker.io/library/tomcat:latest@sha256:1374a565d5122fdb42807f3a5f2d4fcc245a5e15420ff5bb5123afedc8ef769d    0.1s
 => => sha256:8dbbbc6af9dc7b3eec20b35797f66551a17f035a85f020fc99a0457dd268aae8 22.94MB / 22.94MB                  14.4s
 => => sha256:a10b6847b9f1913a9d34980e0354787e49b068cdbdd78c70bab054c6cfbd1660 157.59MB / 157.59MB               103.2s
 => => sha256:1374a565d5122fdb42807f3a5f2d4fcc245a5e15420ff5bb5123afedc8ef769d 6.64kB / 6.64kB                     0.0s
 => => sha256:88b0f1cee84c76bb84a450edacdc37fb3ee00a8706be9298dfe8ec69e5040cdb 12.50kB / 12.50kB                   0.0s
 => => sha256:5a7813e071bfadf18aaa6ca8318be4824a9b6297b3240f2cc84c1db6f4113040 29.75MB / 29.75MB                  20.7s
 => => sha256:ec01946b5efacf78477bdcfbd535c085881d83e873ae884c0cd44ea1e948d49d 2.72kB / 2.72kB                     0.0s
 => => sha256:dcc1c5ea3c7d921e35f64dce04af1c8a2cd97954281eb6af66f6067f6c2c319b 158B / 158B                        15.5s
 => => sha256:91e6cc55403ad09f9aeca15ab95bf547ad0b78be2b665c486beef7161150987d 2.28kB / 2.28kB                    15.8s
 => => sha256:5d4660d0a9e9f9a03ab7c4a134c5a0c8d39f649970092b17ebe6a619a9b838f5 139B / 139B                        16.2s
 => => sha256:4f4fb700ef54461cfa02571ae0db9a0dc1e0cdb5577484a6d75e68dc38e8acc1 32B / 32B                          16.5s
 => => sha256:e231914ca483a93d4915d672a47db6c2022215b7ce5de04ed3bb7214e89c814f 31.87MB / 31.87MB                  24.4s
 => => extracting sha256:5a7813e071bfadf18aaa6ca8318be4824a9b6297b3240f2cc84c1db6f4113040                          1.6s
 => => extracting sha256:8dbbbc6af9dc7b3eec20b35797f66551a17f035a85f020fc99a0457dd268aae8                          1.2s
 => => extracting sha256:a10b6847b9f1913a9d34980e0354787e49b068cdbdd78c70bab054c6cfbd1660                          7.5s
```

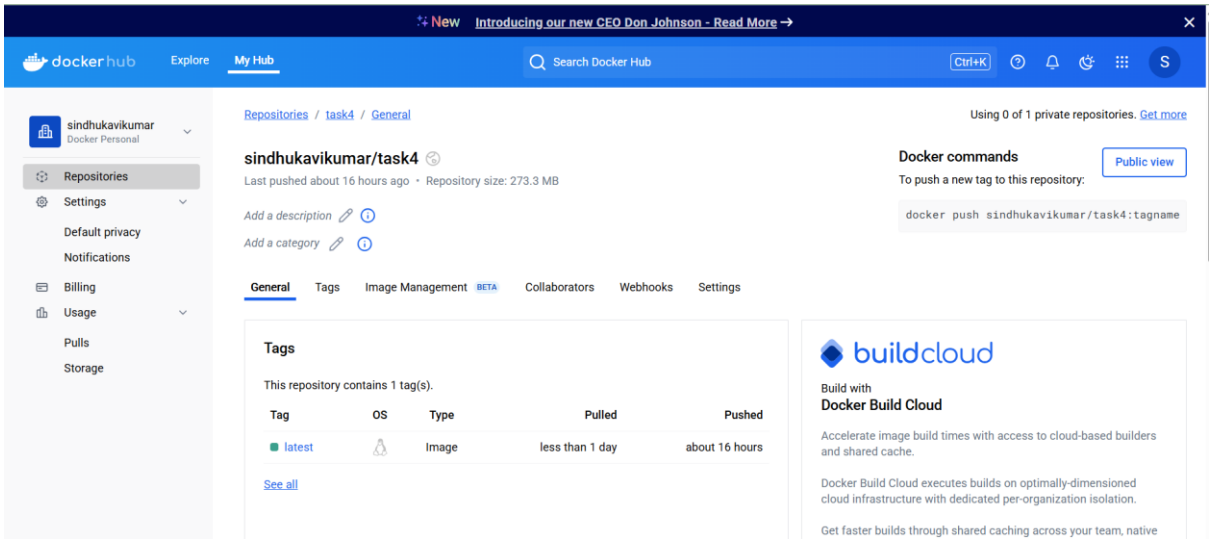
```
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ docker images
REPOSITORY                          TAG       IMAGE ID       CREATED          SIZE
dog                                 latest    4b2453f7d5c1   39 seconds ago   563MB
test2                               latest    0f3aef6de70e   2 hours ago      192MB
sindhukavikumar/task22              latest    0f3aef6de70e   2 hours ago      192MB
your-dockerhub-username/sample1     latest    1d28044a9cbe   4 hours ago      192MB
test                                latest    1d28044a9cbe   4 hours ago      192MB
sindhukavikumar/sample1             latest    1d28044a9cbe   4 hours ago      192MB
nginx                               latest    53a18edff809   6 weeks ago      192MB
gcr.io/k8s-minikube/kicbase         v0.0.46   e72c4cbe9b29   2 months ago     1.31GB
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ docker run -itd -p 96/8080 dog
docker: invalid proto: 8080

Run 'docker run --help' for more information
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ docker run -itd -p 96:8080 dog
37c0cb83f4d8290be03f0a3e03732f9557b3d0c031f907391a131f92c4db1683
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ docker tag dog sindhukavikumar/task4
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ docker images
REPOSITORY                          TAG       IMAGE ID       CREATED         SIZE
dog                                 latest    4b2453f7d5c1   4 minutes ago   563MB
sindhukavikumar/task4               latest    4b2453f7d5c1   4 minutes ago   563MB
test2                               latest    0f3aef6de70e   2 hours ago     192MB
sindhukavikumar/task22              latest    0f3aef6de70e   2 hours ago     192MB
test                                latest    1d28044a9cbe   4 hours ago     192MB
sindhukavikumar/sample1             latest    1d28044a9cbe   4 hours ago     192MB
your-dockerhub-username/sample1     latest    1d28044a9cbe   4 hours ago     192MB
nginx                               latest    53a18edff809   6 weeks ago     192MB
gcr.io/k8s-minikube/kicbase         v0.0.46   e72c4cbe9b29   2 months ago    1.31GB
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ docker login -u sindhukavikumar

i Info → A Personal Access Token (PAT) can be used instead.
         To create a PAT, visit https://app.docker.com/settings


Password:
```

```
Login Succeeded
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ docker push sindhukavikumar/task4
Using default tag: latest
The push refers to repository [docker.io/sindhukavikumar/task4]
861570bc1830: 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:c9f12c1f917c877551566f0d012d2663186de24005649e8481e88a7edaac3ccf size: 2413
```



10. Start Minikube.

11. Create the Kubernetes deployment.

12. Expose the deployment.

13. Expose the service using Minikube and obtain the URL.

```
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ 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
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ kubectl create deployment dogpic --image=sindhukavikumar/task4 --port=8080
deployment.apps/dogpic created
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ kubectl create deployment.apps/dogpic --port=8080 --type=NodePort
error: unknown flag: --port
See 'kubectl create --help' for usage.
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ kubectl expose deployment.apps/dogpic --port=8080 --type=NodePort
service/dogpic exposed
```

```
sindhukumar@Sindhu:~/a/spring-framework-petclinic$ minikube service dogpic
|-----------|--------|-------------|-------------------------|
| NAMESPACE |  NAME  | TARGET PORT |           URL           |
|-----------|--------|-------------|-------------------------|
| default   | dogpic |        8080 | http://192.168.49.2:32206 |
|-----------|--------|-------------|-------------------------|
🏃  Starting tunnel for service dogpic.
|-----------|--------|-------------|------------------------|
| NAMESPACE |  NAME  | TARGET PORT |          URL           |
|-----------|--------|-------------|------------------------|
| default   | dogpic |             | http://127.0.0.1:42571 |
|-----------|--------|-------------|------------------------|
🎉  Opening service default/dogpic in default browser...
👉  http://127.0.0.1:42571
❗  Because you are using a Docker driver on linux, the terminal needs to be open to run it.
```

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