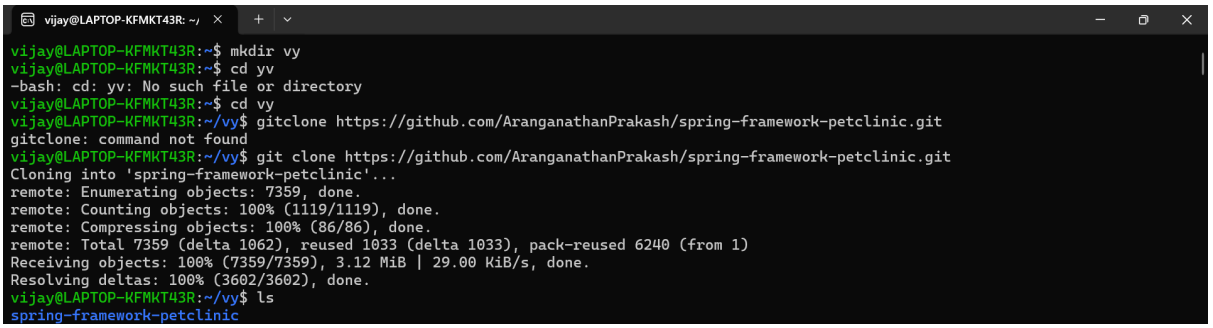
TASK 5 –Maven Creation

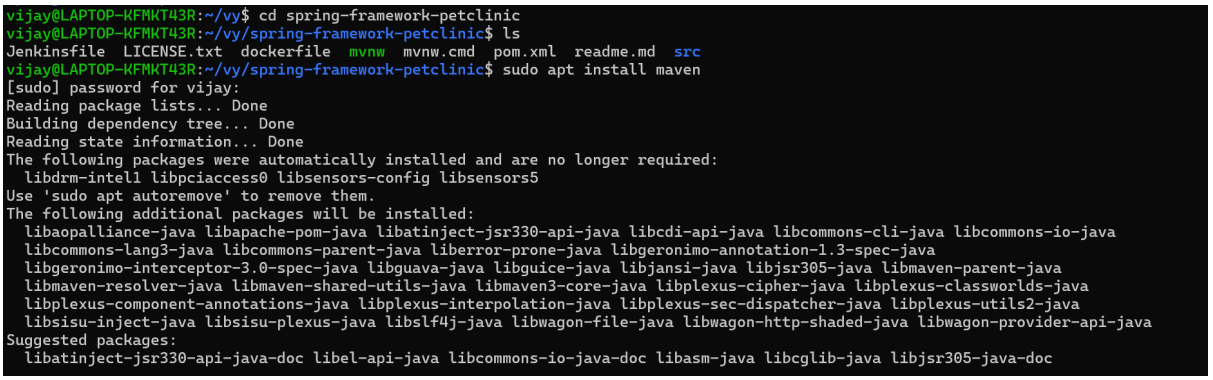
Step 1: Creating folder

Create a folder and clone the repository



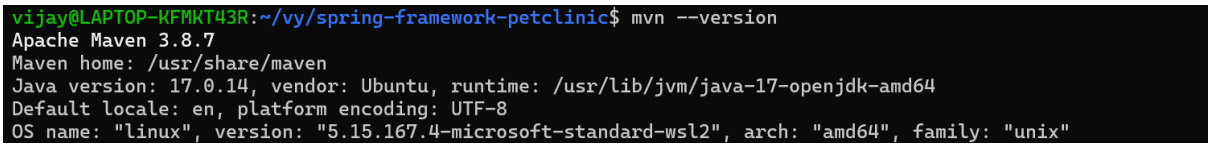
Step 2: Installing maven

Installing maven using -- sudo apt install maven



Step 3: Checking

See if the maven is installed or not



Step 4: Testing

Test the maven

```
vijay@LAPTOP-KFMKT43R:~/vy/spring-framework-petclinic$ mvn test
[INFO] Scanning for projects...
Downloading from central: https://repo.maven.apache.org/maven2/com/fasterxml/jackson/jackson-bom/2.16.1/jackson-bom-2.16.1.pom
Downloaded from central: https://repo.maven.apache.org/maven2/com/fasterxml/jackson/jackson-bom/2.16.1/jackson-bom-2.16.1.pom (18 kB
at 26 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/com/fasterxml/jackson/jackson-parent/2.16/jackson-parent-2.16.pom
Downloaded from central: https://repo.maven.apache.org/maven2/com/fasterxml/jackson/jackson-parent/2.16/jackson-parent-2.16.pom (6.5
kB at 102 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/com/fasterxml/oss-parent/56/oss-parent-56.pom
Downloaded from central: https://repo.maven.apache.org/maven2/com/fasterxml/oss-parent/56/oss-parent-56.pom (24 kB at 291 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/data/spring-data-bom/2023.1.1/spring-data-bom-2023
.1.1.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/data/spring-data-bom/2023.1.1/spring-data-bom-2023.
1.1.pom (5.5 kB at 15 kB/s)
[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/jacoco/jacoco-maven-plugin/0.8.11/jacoco-maven-plugin-0.8.11.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/jacoco/jacoco-maven-plugin/0.8.11/jacoco-maven-plugin-0.8.11.pom (4
.2 kB at 72 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/jacoco/org.jacoco.build/0.8.11/org.jacoco.build-0.8.11.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/jacoco/org.jacoco.build/0.8.11/org.jacoco.build-0.8.11.pom (44 kB a
t 492 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/ow2/asm/asm-bom/9.6/asm-bom-9.6.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/ow2/asm/asm-bom/9.6/asm-bom-9.6.pom (3.2 kB at 69 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/ow2/ow2/1.5.1/ow2-1.5.1.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/ow2/ow2/1.5.1/ow2-1.5.1.pom (11 kB at 191 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/jacoco/jacoco-maven-plugin/0.8.11/jacoco-maven-plugin-0.8.11.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/jacoco/jacoco-maven-plugin/0.8.11/jacoco-maven-plugin-0.8.11.jar (5
7 kB at 676 kB/s)
```

Step 5: Clean

Clean the maven

```
Downloaded from central: https://repo.maven.apache.org/maven2/org/sonatype/spice/spice-parent/16/spice-parent-16.pom (8.4 kB at 96 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 222 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 69
6 kB/s)
[INFO] Deleting /home/vijay/vy/spring-framework-petclinic/target
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  3.386 s
[INFO] Finished at: 2025-03-21T10:23:19Z
[INFO] ------------------------------------------------------------------------
vijay@LAPTOP-KFMKT43R:~/vy/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] --------------------------------[ war ]---------------------------------
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-war-plugin/3.4.0/maven-war-plugin-3.4.0
.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-war-plugin/3.4.0/maven-war-plugin-3.4.0.
pom (8.4 kB at 19 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-war-plugin/3.4.0/maven-war-plugin-3.4.0
.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-war-plugin/3.4.0/maven-war-plugin-3.4.0.
jar (83 kB at 114 kB/s)
```

Step 6: Login in docker

Login in the docker using the username

```
vijay@LAPTOP-KFMKT43R:~/vy/spring-framework-petclinic$ docker build -t petclinic .
[+] Building 283.4s (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                                                    3.4s
 => [auth] library/tomcat:pull token for registry-1.docker.io                                                       0.0s
 => [internal] load .dockerignore                                                                                   0.1s
 => => transferring context: 2B                                                                                     0.0s
 => [1/2] FROM docker.io/library/tomcat:latest@sha256:1374a565d5122fdb42807f3a5f2d4fcc245a5e15420ff5bb5123afedc8ef769d      279.3s
 => => resolve docker.io/library/tomcat:latest@sha256:1374a565d5122fdb42807f3a5f2d4fcc245a5e15420ff5bb5123afedc8ef769d        0.1s
 => => sha256:88b0f1cee84c76bb84a450edacdc37fb3ee00a8706be9298dfe8ec69e5040cdb 12.50kB / 12.50kB                    0.0s
 => => sha256:5a7813e071bfadf18aaa6ca8318be4824a9b6297b3240f2cc84c1db6f4113040 29.75MB / 29.75MB                   94.0s
 => => sha256:8dbbbc6af9dc7b3eec20b35797f66551a17f035a85f020fc99a0457dd268aae8 22.94MB / 22.94MB                   67.2s
 => => sha256:a10b6847b9f1913a9d34980e0354787e49b068cdbdd78c70bab054c6cfbd1660 157.59MB / 157.59MB               275.3s
 => => sha256:1374a565d5122fdb42807f3a5f2d4fcc245a5e15420ff5bb5123afedc8ef769d 6.64kB / 6.64kB                      0.0s
 => => sha256:ec01946b5efacf78477bdcfbd535c085881d83e873ae884c0cd44ea1e948d49d 2.72kB / 2.72kB                      0.0s
 => => sha256:dcc1c5ea3c7d921e35f64dce04af1c8a2cd97954281eb6af66f6067f6c2c319b 158B / 158B                         68.8s
 => => sha256:91e6cc55403ad09f9aeca15ab95bf547ad0b78be2b665c486beef7161150987d 2.28kB / 2.28kB                     69.3s
 => => sha256:5d4660d0a9e9f9a03ab7c4a134c5a0c8d39f649970092b17ebe6a619a9b838f5 139B / 139B                         70.8s
 => => sha256:4f4fb700ef54461cfa02571ae0db9a0dc1e0cdb5577484a6d75e68dc38e8acc1 32B / 32B                           71.7s
 => => sha256:e231914ca483a93d4915d672a47db6c2022215b7ce5de04ed3bb7214e89c814f 31.87MB / 31.87MB                 266.3s
 => => extracting sha256:5a7813e071bfadf18aaa6ca8318be4824a9b6297b3240f2cc84c1db6f4113040                           1.5s
 => => extracting sha256:8dbbbc6af9dc7b3eec20b35797f66551a17f035a85f020fc99a0457dd268aae8                           0.9s
 => => extracting sha256:a10b6847b9f1913a9d34980e0354787e49b068cdbdd78c70bab054c6cfbd1660                           1.9s
```

Step 7: Push

Push the image inside the docker

```
vijay@LAPTOP-KFMKT43R:~/vy/spring-framework-petclinic$ docker tag petclinic vijayvk10/dev
vijay@LAPTOP-KFMKT43R:~/vy/spring-framework-petclinic$ docker push vijayvk10/dev
Using default tag: latest
The push refers to repository [docker.io/vijayvk10/dev]
fc6966276563: 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: sha256:6baef3d63de919932b72bb12fe87d7623064f9c0c8d948c12c32902c369d9337 size: 2413
```

Step 8: Minikube

Start the minikube

```
vijay@LAPTOP-KFMKT43R:~$ 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 ...
🔄  Restarting existing docker container for "minikube" ...
🐳  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 9: Deployment creation

Create a deployment named petclinic

```
vijay@LAPTOP-KFMKT43R:~/vy/spring-framework-petclinic$ kubectl create deployment pet --image=vijayvk10/dev
deployment.apps/pet created
```

Step 10: Deployment exposure

Expose the deployment in the kubectl

```
vijay@LAPTOP-KFMKT43R:~/vy/spring-framework-petclinic$ kubectl expose deployment pet --type=LoadBalancer --port=8080
service/pet exposed
```

Step 11: Service

Check the service of the petclinic webpage

```
vijay@LAPTOP-KFMKT43R:~/vy/spring-framework-petclinic$ minikube service pet1
|-----------|------|-------------|---------------------------|
| NAMESPACE | NAME | TARGET PORT |            URL            |
|-----------|------|-------------|---------------------------|
| default   | pet1 |        8080 | http://192.168.49.2:31164 |
|-----------|------|-------------|---------------------------|
🏃  Starting tunnel for service pet1.
|-----------|------|-------------|------------------------|
| NAMESPACE | NAME | TARGET PORT |          URL           |
|-----------|------|-------------|------------------------|
| default   | pet1 |             | http://127.0.0.1:38871 |
|-----------|------|-------------|------------------------|
🎉  Opening service default/pet1 in default browser...
👉  http://127.0.0.1:38871
❗  Because you are using a Docker driver on linux, the terminal needs to be open to run it.
```

Step 12: Output

The output page is displayed in the localhost:44929