

Objective:

1. Create and download a Spring Boot application with Spring initializer.
2. Import the application into Eclipse.
3. Create controller class with endpoint.
4. Run and verify the application.
5. Write a Docker file to containerize the application
6. Build the docker image.
7. Push the docker image to Docker Hub
8. Run on application as container in detached mode and on system port 8089

Solution:-

Step 1: Installing docker and on server

Installation Source links :

Docker - <https://docs.docker.com/engine/install/ubuntu/>

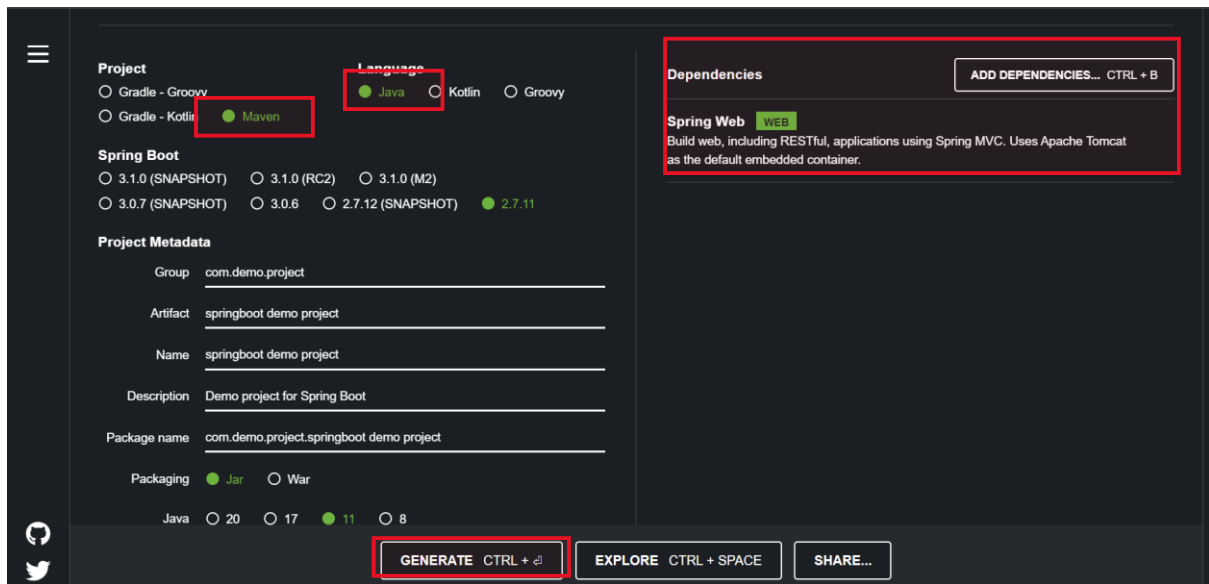
Maven: apt install maven

Java : apt install openjdk-11-jdk

Generating a spring boot project from spring initializer site

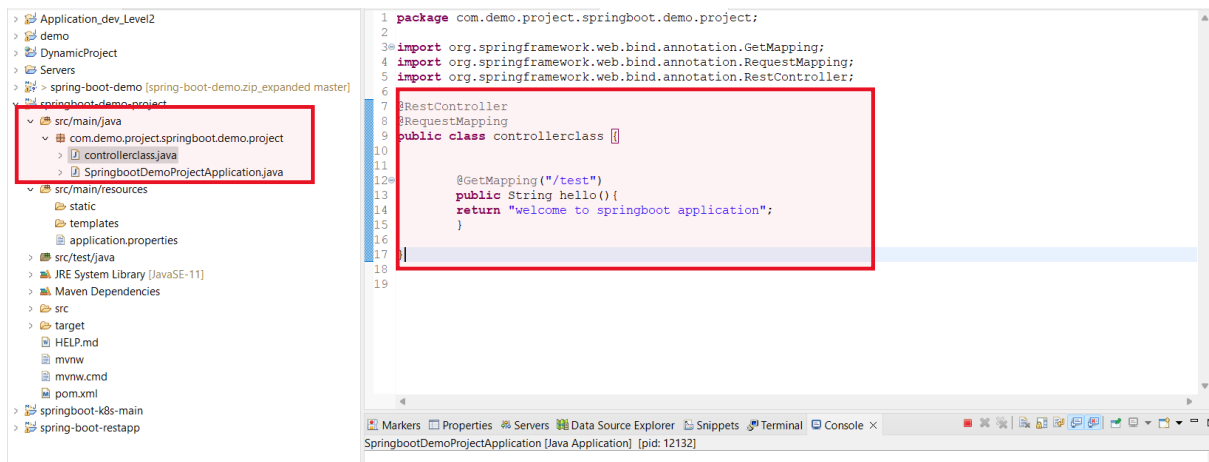
<https://start.spring.io/>

Add the **spring web** dependency.



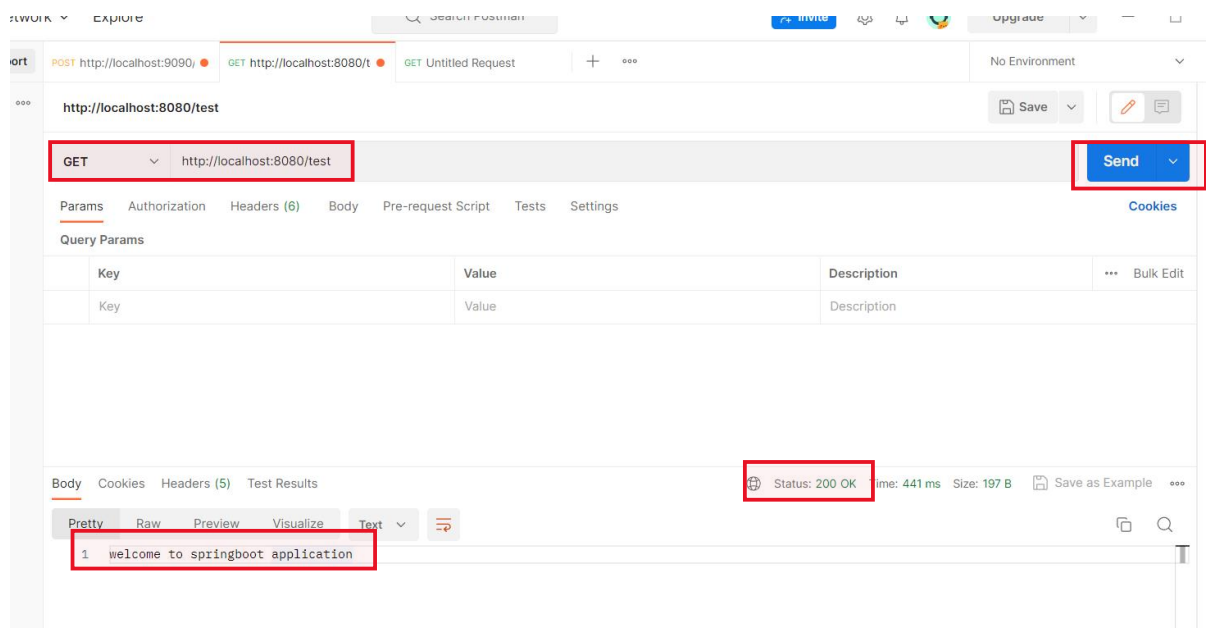
Importing into eclipse IDE

And creating a controller class which acts as a rest application



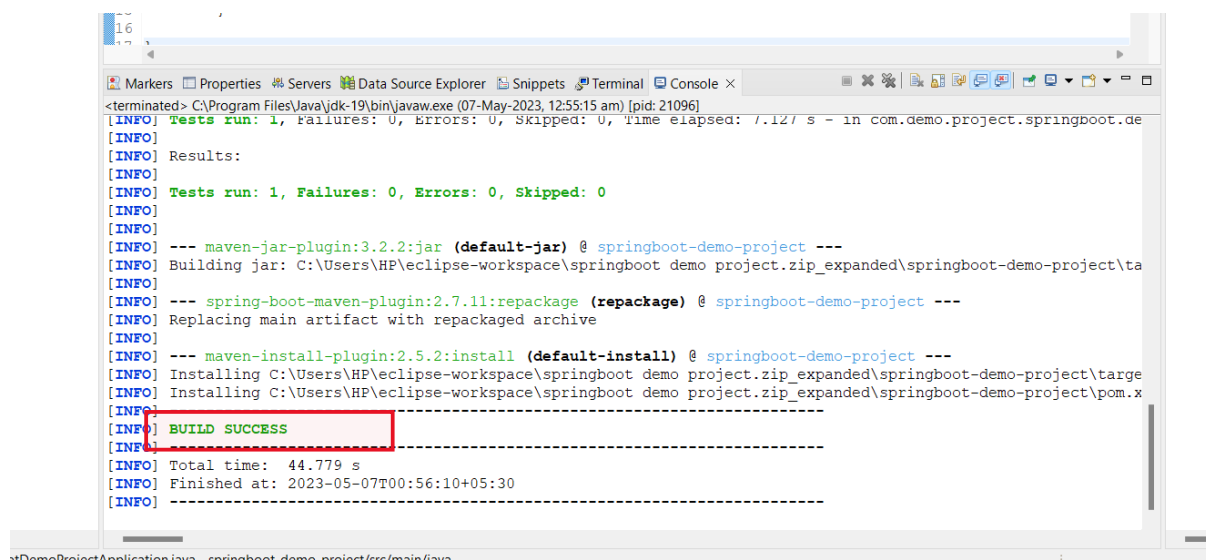
Running the application on local machine by using postman

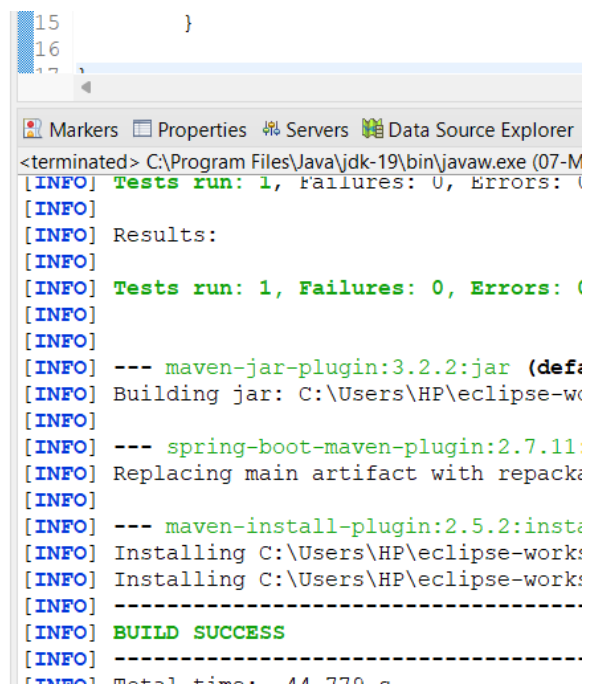
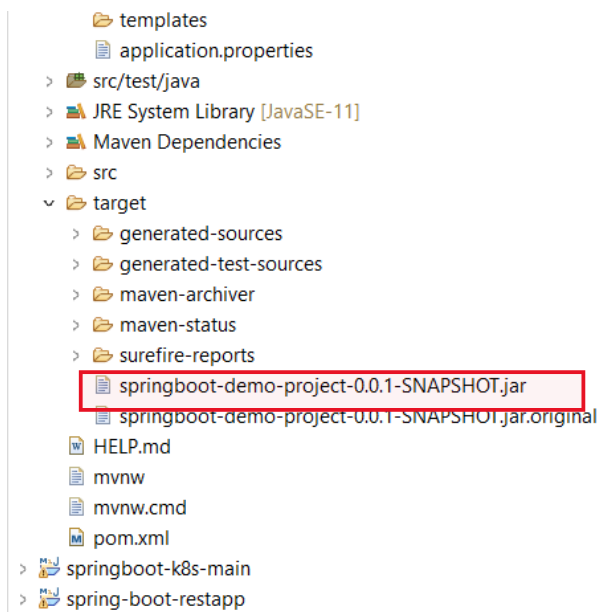
<http://localhost:8080/test>



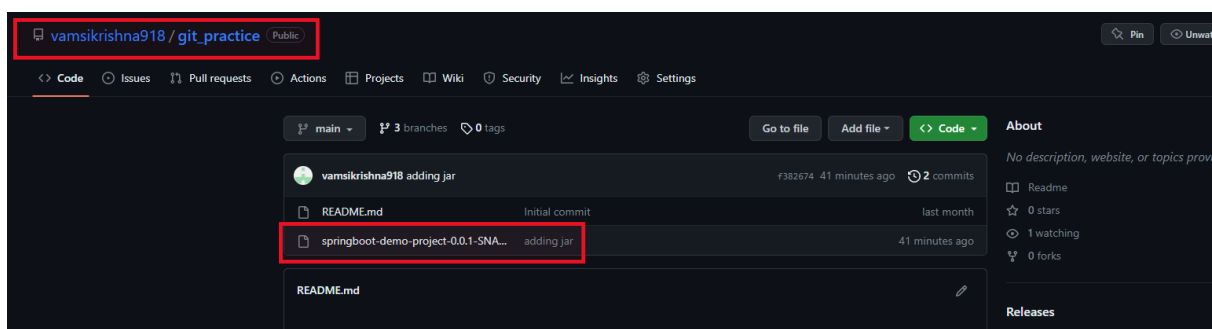
Building the application with **mvn clean install**

With which we can get a jar file





I have imported the jar file into **git hub** and **clone the repo in ubuntu machine**



Cloning the repository to get the jar

Git clone https://github.com/vamsikrishna918/git_practice

```
root@ip-172-31-55-81:~# git clone https://github.com/vamsikrishna918/git_practice
Cloning into 'git_practice'
remote: Enumerating objects: 34, done.
remote: Counting objects: 100% (34/34), done.
remote: Compressing objects: 100% (17/17), done.
remote: Total 34 (delta 2), reused 29 (delta 1), pack-reused 0
Receiving objects: 100% (34/34), 15.22 MiB | 30.98 MiB/s, done.
Resolving deltas: 100% (2/2), done.
root@ip-172-31-55-81:~# ls
git_practice  snap
root@ip-172-31-55-81:~# cd git_practice/
root@ip-172-31-55-81:~/git_practice# ls
README.md  springboot-demo-project-0.0.1-SNAPSHOT.jar
```

Running the jar

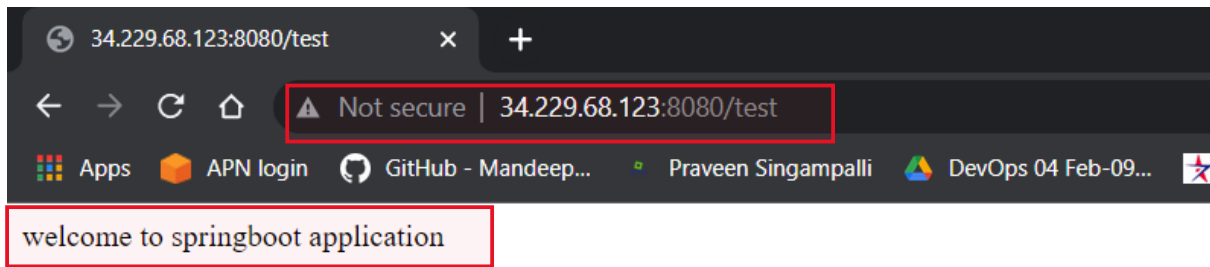
Command :

Java -jar springboot-demo-project.jar

```
root@ip-172-31-55-81:~/git_practice# java -jar springboot-demo-project-0.0.1-SNAPSHOT.jar
[Spring Boot] (v2.7.11)
2023-05-06 19:05:36.875 INFO 1752 --- [main] p.s.d.p.SpringbootDemoProjectApplication : Starting SpringbootDemoProjectApplication v0.0.1-SNAPSHOT using Java 11.0.18 on ip-172-31-55-81 with PID 1752 (/root/.git_practice/springboot-demo-project-0.0.1-SNAPSHOT.jar started by root in /root/.git_practice)
2023-05-06 19:05:36.887 INFO 1752 --- [main] p.s.d.p.SpringbootDemoProjectApplication : No active profile set, falling back to default profile: 'default'
2023-05-06 19:05:36.887 INFO 1752 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8080 (http)
2023-05-06 19:05:36.921 INFO 1752 --- [main] o.s.catalina.core.AprLifecycleListener : [Tomcat] [v1.7.0]
2023-05-06 19:05:36.921 INFO 1752 --- [main] o.s.catalina.core.AprLifecycleListener : APR capabilities: IPv6 [true], sendfile [true], accept filters [false], random [true], UDS [true].
2023-05-06 19:05:36.922 INFO 1752 --- [main] o.s.catalina.core.AprLifecycleListener : APR/OpenSSL configuration: useAprConnector [false], useOpenSSL [true]
2023-05-06 19:05:36.923 INFO 1752 --- [main] o.s.catalina.core.AprLifecycleListener : OpenSSL successfully initialized [OpenSSL 3.0.2 15 Mar 2022]
2023-05-06 19:05:36.929 INFO 1752 --- [main] org.apache.catalina.core.StandardEngine : Starting service [Tomcat]
2023-05-06 19:05:36.929 INFO 1752 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.74]
2023-05-06 19:05:36.948 INFO 1752 --- [main] o.s.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2023-05-06 19:05:36.948 INFO 1752 --- [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 2480 ms
2023-05-06 19:05:40.973 INFO 1752 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http) with context path ''
2023-05-06 19:05:40.993 INFO 1752 --- [main] p.s.d.p.SpringbootDemoProjectApplication : Started SpringbootDemoProjectApplication in 5.173 seconds (JVM running for 6.177)
2023-05-06 19:05:53.409 INFO 1752 --- [nio-8080-exec-1] o.s.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring DispatcherServlet 'dispatcherServlet'
2023-05-06 19:05:53.410 INFO 1752 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet : Initializing Servlet 'dispatcherServlet'
2023-05-06 19:05:53.418 INFO 1752 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet : Completed initialization in 1 ms
exit
```

Accessing the application

<http://Public ip:8080>



Creating the Docker file:

Command :

Vi Dockerfile

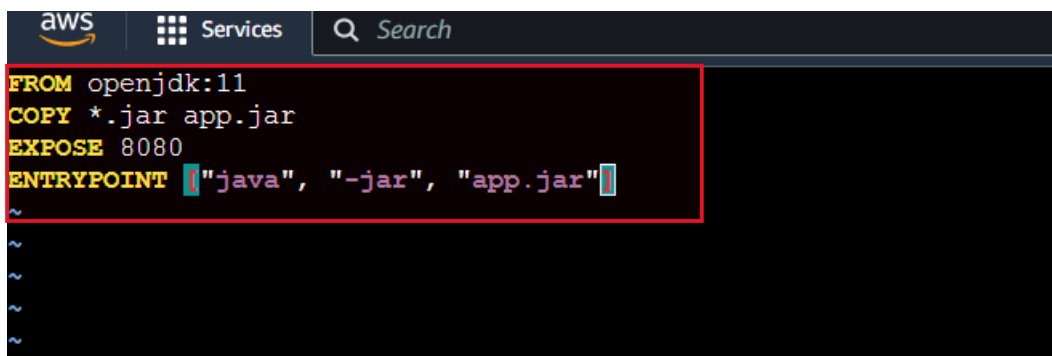
Insert below:

FROM openjdk:11

COPY *.jar app.jar

EXPOSE 8080

ENTRYPOINT ["java", "-jar", "app.jar"]



Building the docker image;

Command :

docker build -t springboot-demo .

```
root@ip-172-31-55-81:~/git_practice# docker build -t springboot-demo .
[+] Building 20.5s (9/8) FINISHED
=> [internal] load build definition from Dockerfile
=> [internal] load .dockerignore
=> [internal] load metadata for docker.io/library/openjdk:11
=> [auth] library/openjdk:pull token for registry-1.docker.io
=> [internal] load build context
=> [1/2] FROM docker.io/library/openjdk:11
=> resolve docker.io/library/openjdk:11
=> sha256:001c5c4e47e3b3e139e02f4693e5e0c22cdd82e00a917aee5e21452 55.00MB / 55.00MB
=> sha256:e01b7f317654b0226d3993e014b04bcb2250393b1b5de41e50feead4cd3c 1.79kB / 1.79kB
=> sha256:4fa522d582b743b5b0bccc55a9ede77de84eaf15335a7dec9d5e3b0ce0fa7 6.26kB / 6.26kB
=> sha256:d86430e3e3e4674e4f991b4e9373d6c4f08e4103b4e271e20262773d163 5.40MB / 5.40MB
=> sha256:2068746827ec1b043b571e4788e93eab7a9b2a9301176512731f2c317a2016a 10.68MB / 10.68MB
=> sha256:89bac5bf83633e3c7399ead723e8415e7b563b58a01e4599e580f9c9db7c21ab 1.04kB / 1.04kB
=> sha256:9aed2282308848e472aeb7c0eb407fa5380eb3b24cd220e78e971a6 54.40MB / 54.40MB
=> sha256:d85151f15b6683b98221c3827ac545188b16497e14a1049710ebc493de3ad5 5.42MB / 5.42MB
=> sha256:46223a710990a0ae7162eed80417d30303efaf224aaf57aa30348725e2230b 213B / 213B
=> sha256:db3089e6ca411b07220702978a6185ac0222a0c30c77f2516de8201 202.07MB / 202.07MB
=> extracting sha256:001c5c4e47e3b3e139e02f4693e5e0c22cdd82e00a917aee5e21452 5.2s
=> extracting sha256:d86430e3e3e4674e4f991b4e9373d6c4f08e4103b4e271e20262773d163 0.4s
=> extracting sha256:2068746827ec1b043b571e4788e93eab7a9b2a9301176512731f2c317a2016a 0.4s
=> extracting sha256:89bac5bf83633e3c7399ead723e8415e7b563b58a01e4599e580f9c9db7c21ab 1.8s
=> extracting sha256:d85151f15b6683b98221c3827ac545188b16497e14a1049710ebc493de3ad5 0.4s
=> extracting sha256:46223a710990a0ae7162eed80417d30303efaf224aaf57aa30348725e2230b 0.2s
=> extracting sha256:db3089e6ca411b07220702978a6185ac0222a0c30c77f2516de8201 3.5s
[2/2] COPY *.jar app.jar
=> exporting to image
=> exporting layers
=> writing image sha256:9583f6ab5103f6b2f3c056639e2c7fe22698650ab9cdf58301f28a50a7847
=> loading to registry
=> pushing to docker.io/library/springboot-demo
```

Verifying the docker images

Command :

Docker images

```
root@ip-172-31-55-81:~/git_practice# docker images
REPOSITORY          TAG          IMAGE ID          CREATED           SIZE
springboot-demo     latest       9583f6ab5103     46 seconds ago   672MB
vamsi12358/addressbook 1.0         0d7bce3f708c     3 hours ago      491MB
root@ip-172-31-55-81:~/git_practice# docker login
Authenticating with existing credentials...
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
To fix this warning, remove the password field from your config file.
```

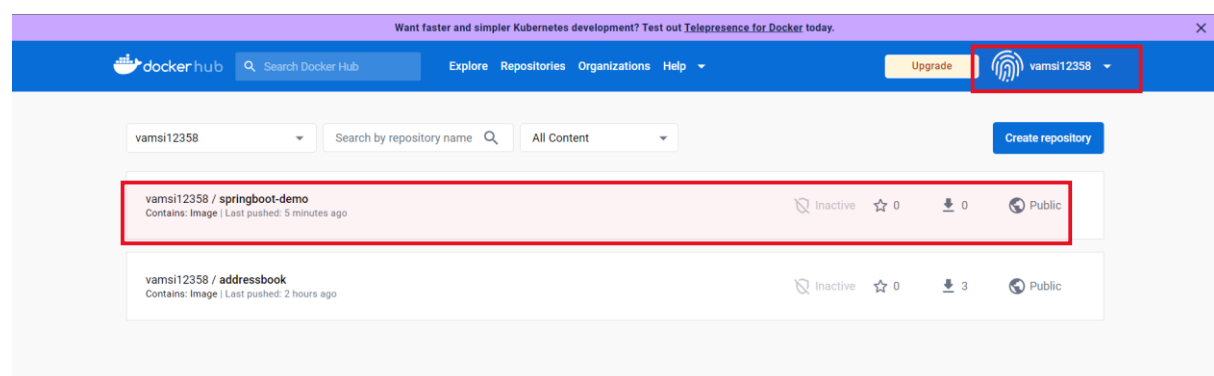
Login in to docker hub

```
root@ip-172-31-55-81:~/git_practice# 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@ip-172-31-55-81:~/git_practice# docker push springboot-demo
Using default tag: latest
```

Pushing the docker images into docker hub;

```
root@ip-172-31-55-81:~/git_practice# docker tag springboot-demo vamsi12358/springboot-demo
root@ip-172-31-55-81:~/git_practice# docker push vamsi12358/springboot-demo
Using default tag: latest
The push refers to repository [docker.io/vamsi12358/springboot-demo]
d56d81d8976d: Pushed
7b7f3078e1db: Pushed
826c3ddb29c: Pushed
b626401ef603: Pushed
9b55156abf26: Pushed
293d5db30c9f: Pushed
03127cdb479b: Pushed
9c742cd6c7a5: Pushed
latest: digest: sha256:1b018096dd5307d193c4bf5bdd6f1eb7e8f4a5f58d5a3edfabca62ff2c2bd23a size: 2007
root@ip-172-31-55-81:~/git_practice# docker images
```



Running the spring boot image with **port binding 8089**

Command :

```
docker run -itd -p 8089:8080 vamsi12358/springboot-demo
```

```
root@ip-172-31-55-81:~/git_practice# docker run -itd -p 8089:8080 vamsi12358/springboot-demo
5d2932a8b59a19bbbc1218b133ed116ad241dbfc02990f0dbfb5b424af9d7c74
root@ip-172-31-55-81:~/git_practice# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
5d2932a8b59a	vamsi12358/springboot-demo	"java -jar app.jar"	19 seconds ago	Up 18 seconds	0.0.0.0:8089->8080/tcp, :::8089->8080/tcp	cool_feistel

```
root@ip-172-31-55-81:~/git_practice# vi Dockerfile
root@ip-172-31-55-81:~/git_practice# docker build -t.springboot-demo
```

Verifying the spring boot application on port 8089

<http://Public ip:8089/test>

