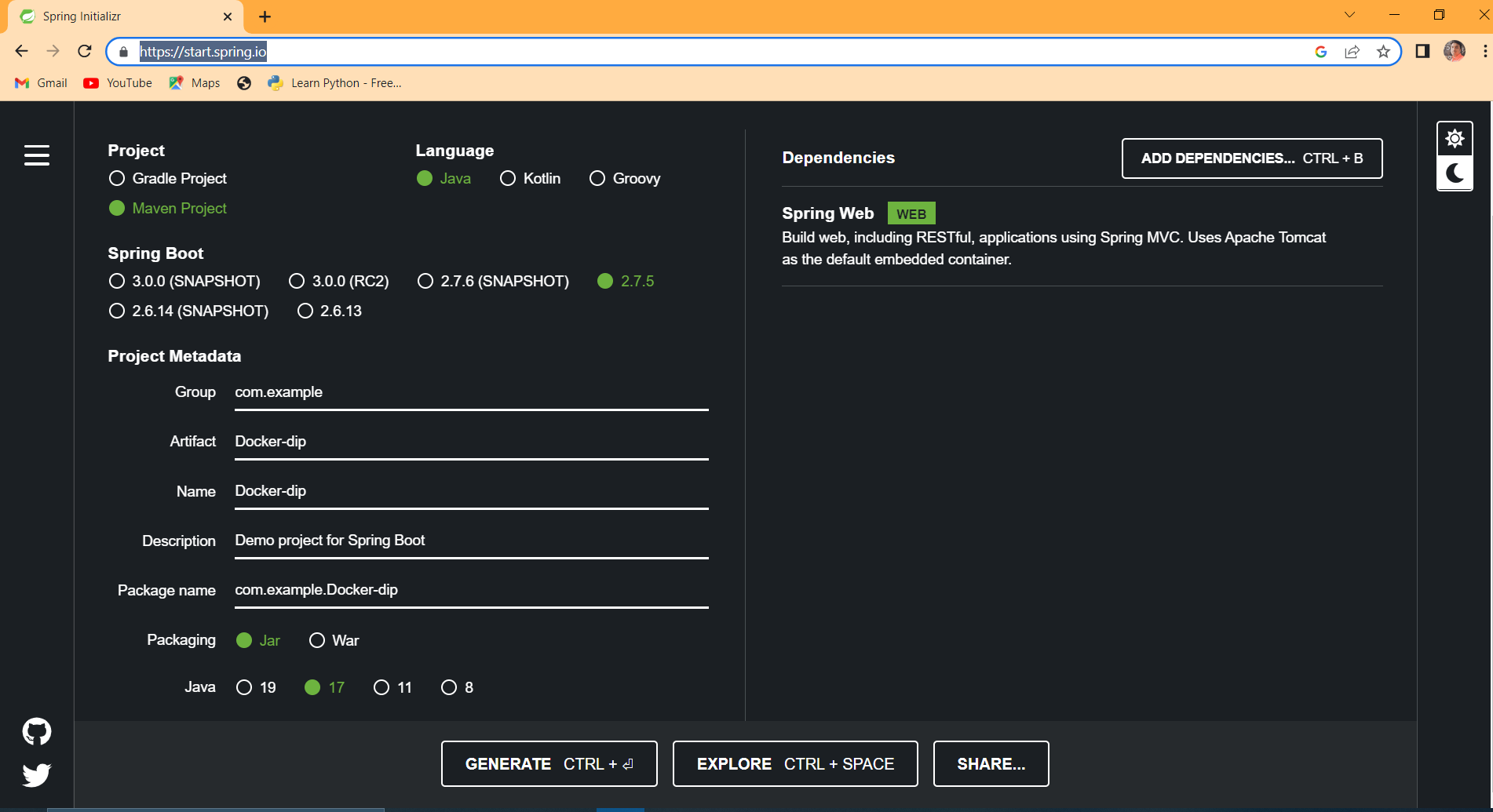
***Dockerazation with Springboot:-***

**Dockerazation or Deployment with SpringBoot, Angular, Mysql, Keycloak & Kubernetes Deployments Also Included**

Create Maven project by clicking this <https://start.spring.io/> like shown below



Add Dependencies like Spring Web and Generate

Add this code in main file

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.RestController;

@SpringBootApplication

@RestController

public class DockerDipApplication {

@GetMapping("/")

public String getMessage() {

return "Welcome to Spring Boot Application ";

}

public static void main(String[] args) {

SpringApplication.run(DockerDipApplication.class, args);

}

}

**And Next to this Create jar file**.

To do Go to pom.xml file and add this line

<finalName>Docker-Spring</finalName>

Docker-spring is a jar name you can give anything to that

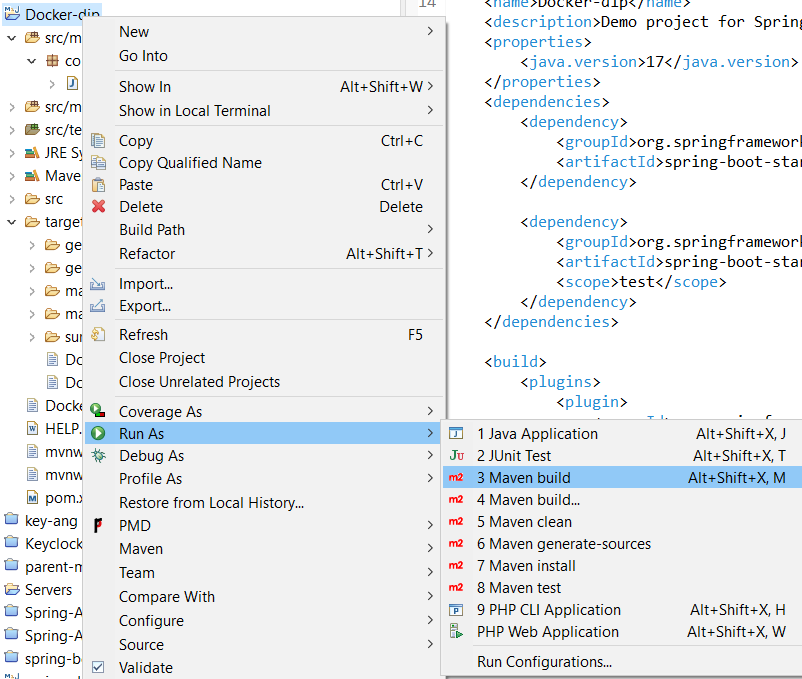
To Create jar right click on project and run as

1)**maven install** or

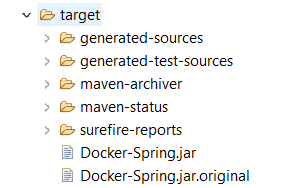
2)**Maven build** in this type **mvn clean** and enter it will create a jar. or

3)**Maven Clean Package or Mvn Build Packege or Mvn Package**

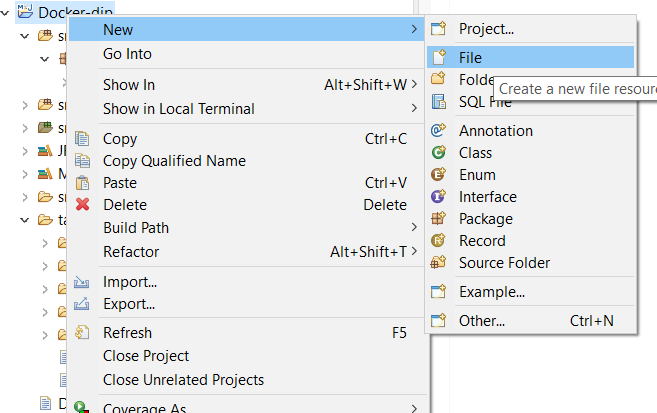
In applicaton property need to replace locacalhost with mysql docker ip address and and password also need to change



To check this click on Target Folder there you will find your jars.



**And Then You need to create a Docker file By naming Like Dockerfile**

****

**Like this**

**in this file you need to add this code**

**FROM openjdk:17-jdk-alpine**

**ARG JAR\_FILE=target/\*.jar**

**COPY ${JAR\_FILE} app.jar**

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

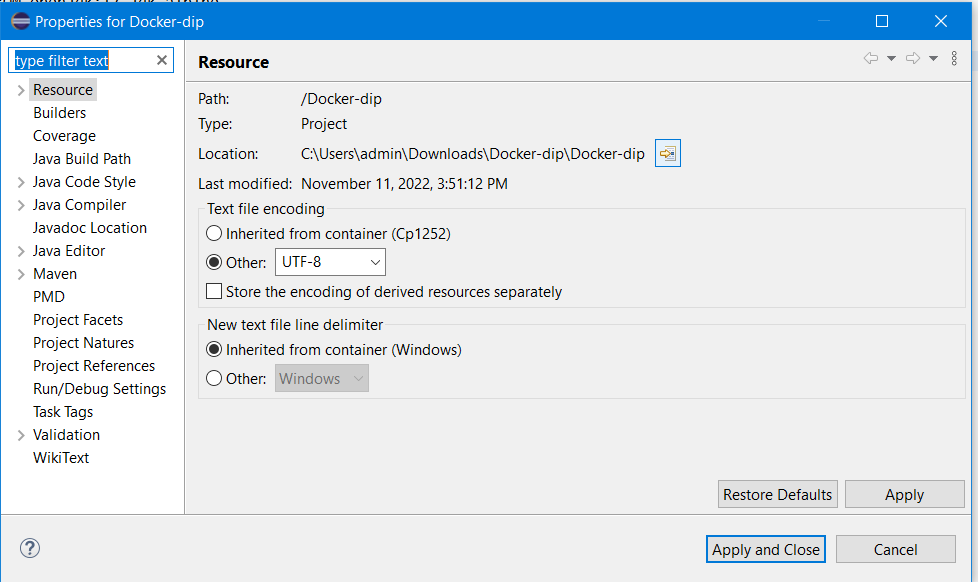
Make Sure you are using java **version or jdk** that is installed on your system .

Follow this Link For Reference <https://spring.io/guides/gs/spring-boot-docker/>

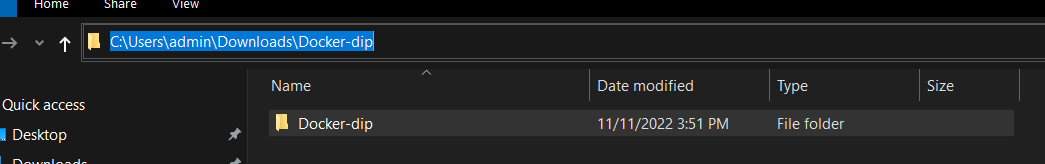
Next to this to run Project or To create Image in docker follow this

Open eclipse Terminal or you can do this

Right click on Your Project file and select Properties in this pop up click arrow in front of Location : like below shown



Then you will Locate to the file Location and then click on that like

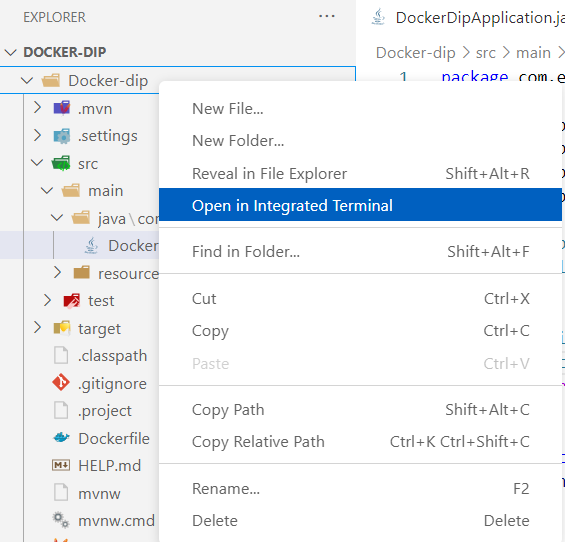


In selected Row you need to type ***cmd***

In command prompt type ***Code .***

This will open Your file in Runnable form of Author commands to Deploy in Docker

Open termail like this



**Before that make sure that Docker is running**

And then type this command in Integrated Thermal :

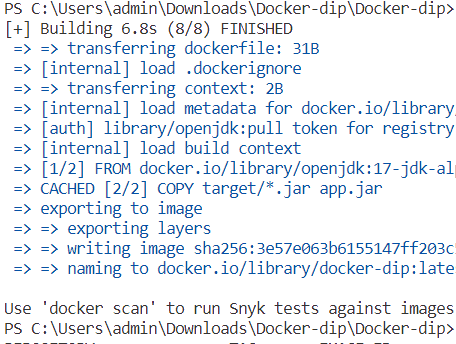
**docker build -t docker-dip:latest .**

**docker-dip**

is a maven filename or spring boot file name

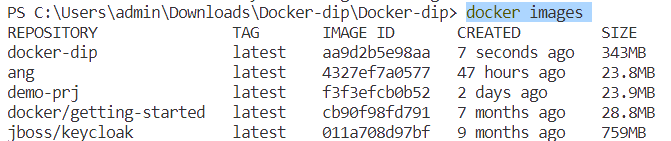
Press Enter

It need to run like this need to be finished



And then check the image in both Docker images or in a Terminal

To View in Terminal Type this Command **docker images**



Here You can see the docker-dip Image

Next to Run this or to create container Type This Command

docker run -it --name springboot -p 8082:8080 docker-dip

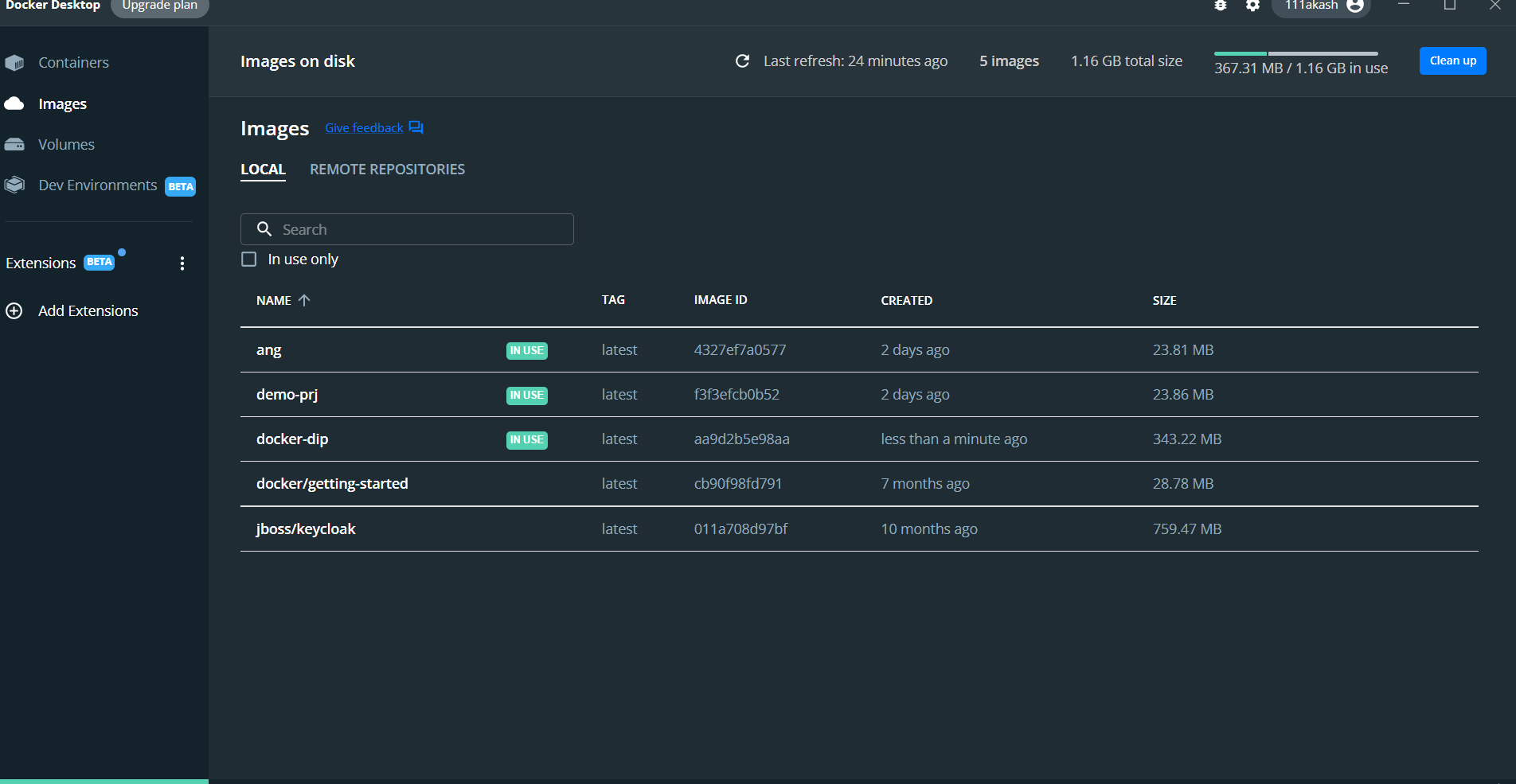
Here in this Command springboot is a Container name You can give any name **and 8082 is localhost or Host machine port and 8080 is default port or Docker image port of Springboot or Eclipse port** docker-dip is maven file name.

It Need to run Like this

****

**To Checker in docker :-**

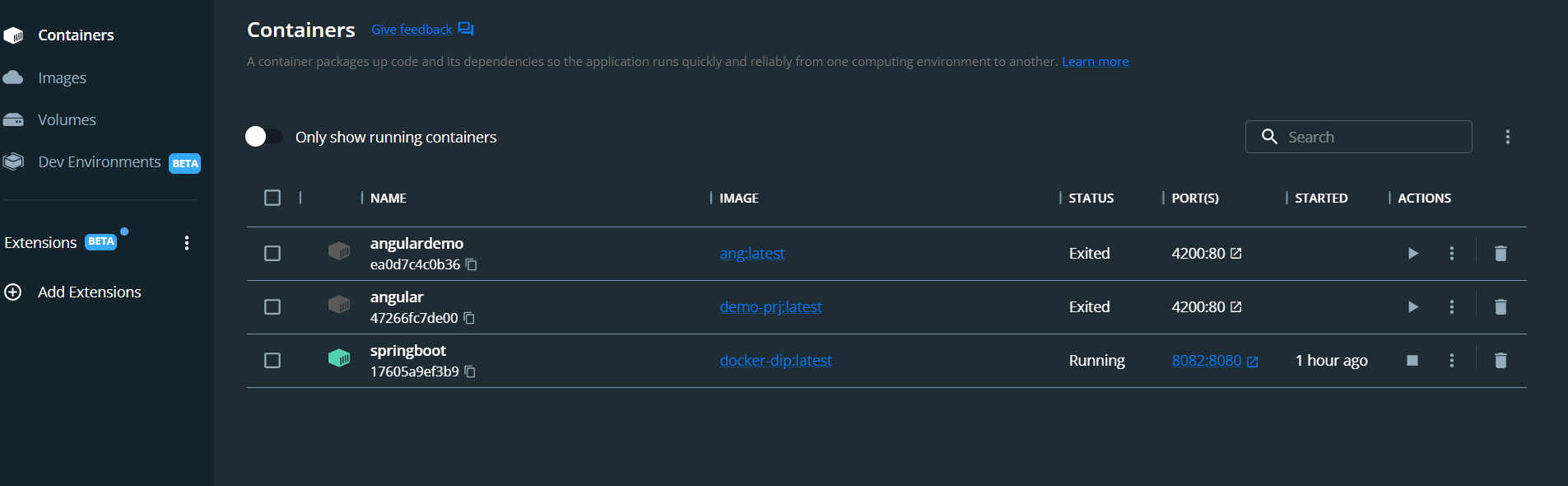
**Go to docker Destop file like this**

****

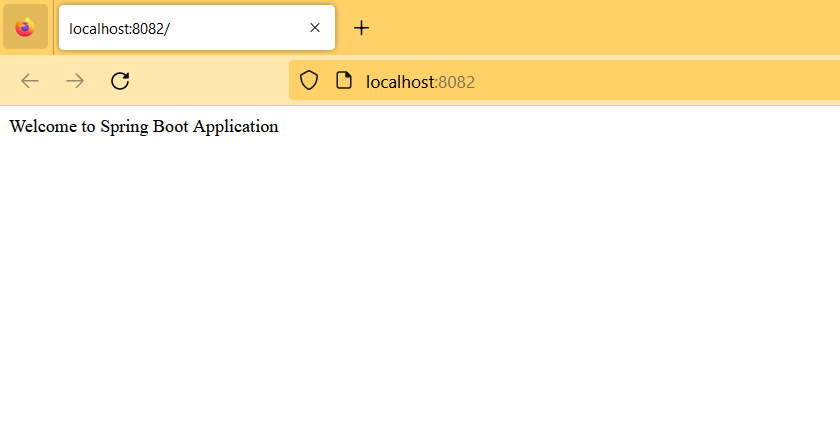
**Find your image is there are not if yes**

**Click on Containers :-**

**Find your Container name and click on port to view You out put or Successfully Deployment of Spring Boot**

****

**OUTPUT:-**

****

**To push in docker hub**

1. **docker tag docker-dip 111akash/ docker-dip**
2. **docker push 111akash/spring**

**Successfully Done With Spring Boot**

**References**

[**https://www.middlewareinventory.com/blog/deploy-docker-image-to-kubernetes/**](https://www.middlewareinventory.com/blog/deploy-docker-image-to-kubernetes/)

**Deployment of Angular:-**

Create a simple project for Hello World in angular.

And then Add Docker file or Create Docker file by right clicking on the project, name that to like

**Dockerfile and add this code.**

#stage 1

FROM node:latest as node or check the angular version by ng version if it is 15 then give here 14 or 15 instead of latest if angular prj is latest then okm

WORKDIR /app

COPY . .

RUN npm install

RUN npm run build --prod

#stage 2

FROM nginx:alpine

COPY --from=node /app/dist/ang-doc /usr/share/nginx/html

In here ang-doc is the project name you change it to your project name.

And in terminal paste below code and follow steps and make sure Docker is running behind

**1)docker build -t ang-doc .**

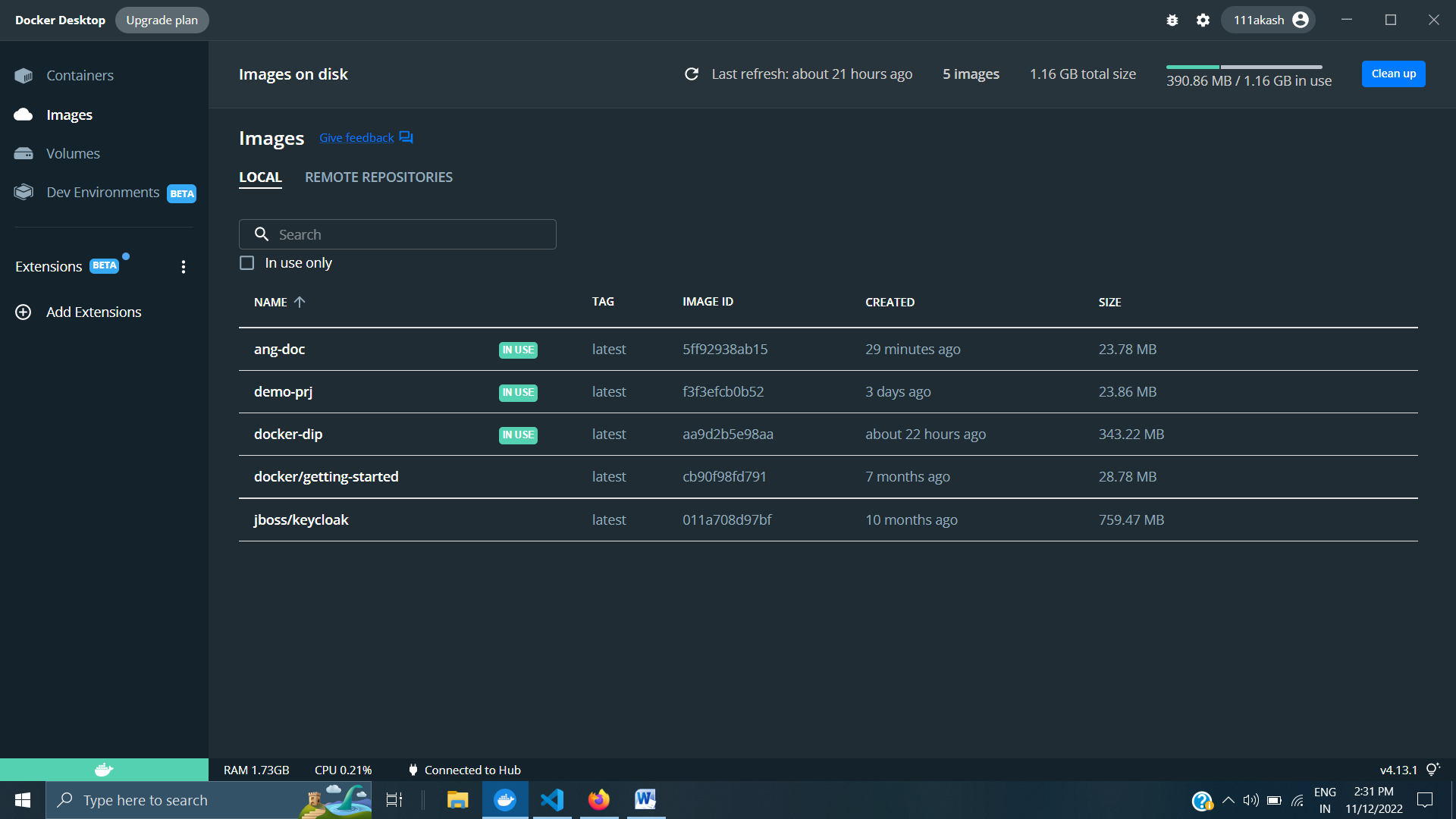
Here **ang-doc** . is project name

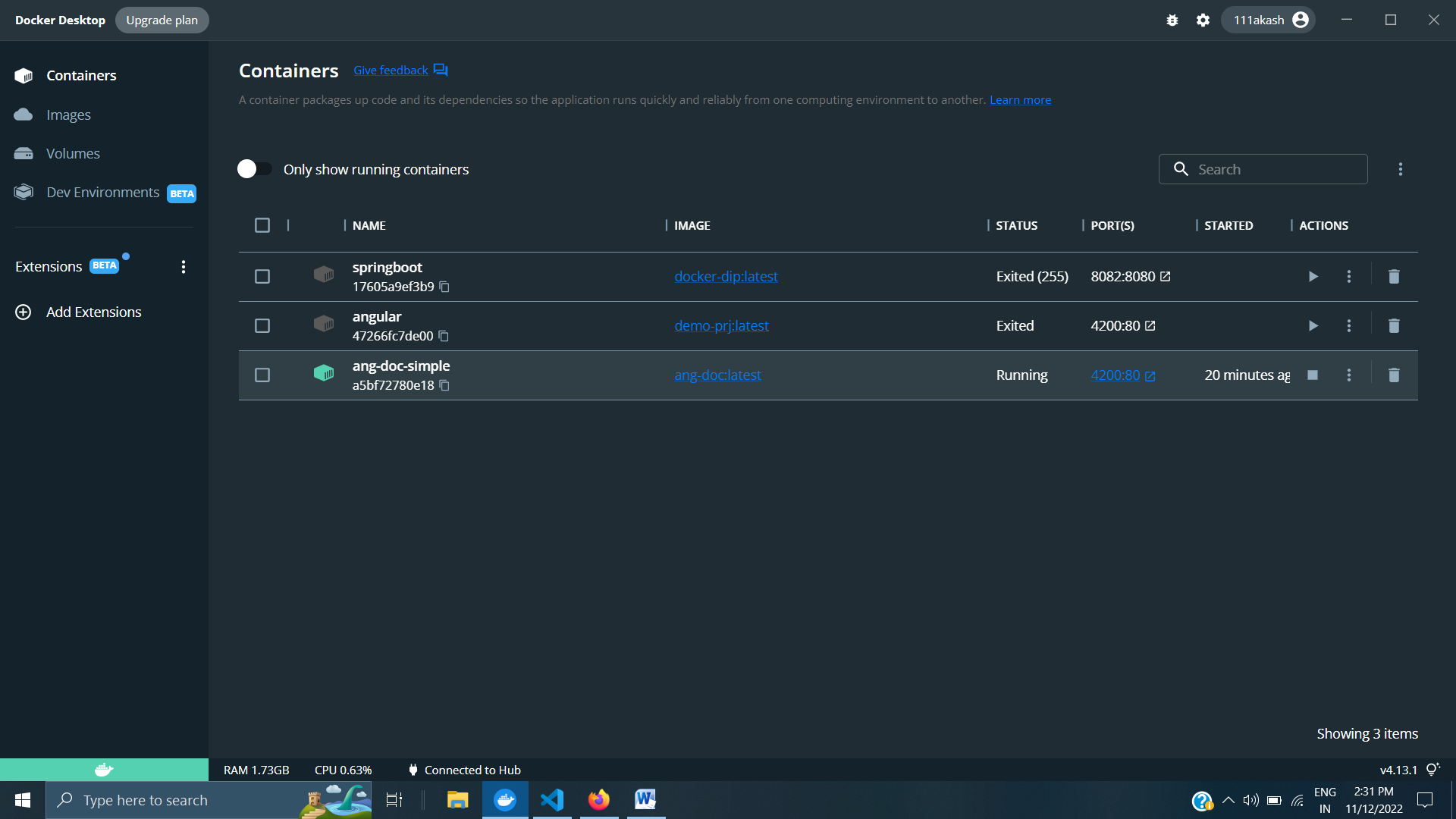
**2) docker images**

**3) docker run --name ang-doc-simple -it -p 4200:80 ang-doc**

Here ang-doc-simple is container name and ang-doc is Project name

Now you can see your image in docker and able to run that like below image.





Click on Link You are good to go.

**MySQL Deployment in Docker:-**

Open command Prompt & Run Docker Before go with Cmd.

Type this command in command prompt one by one

**1) docker ps -a**

**2) docker pull mysql**

**3) docker run -p 3307:3306 --name mysqldbbackend mysql**

**3) docker run --name mysqldb -e MYSQL\_ROOT\_PASSWORD=1234 -d mysql**

**Or the best one is**

**3)** docker run --name my-mysql -p 3306:3306 -e MYSQL\_ROOT\_PASSWORD=mypassword -d mysql

**In this the mysqldb is Container name & mysql is image name**

**4) docker exec -it mysqldb bash or (my-mysql bash)**

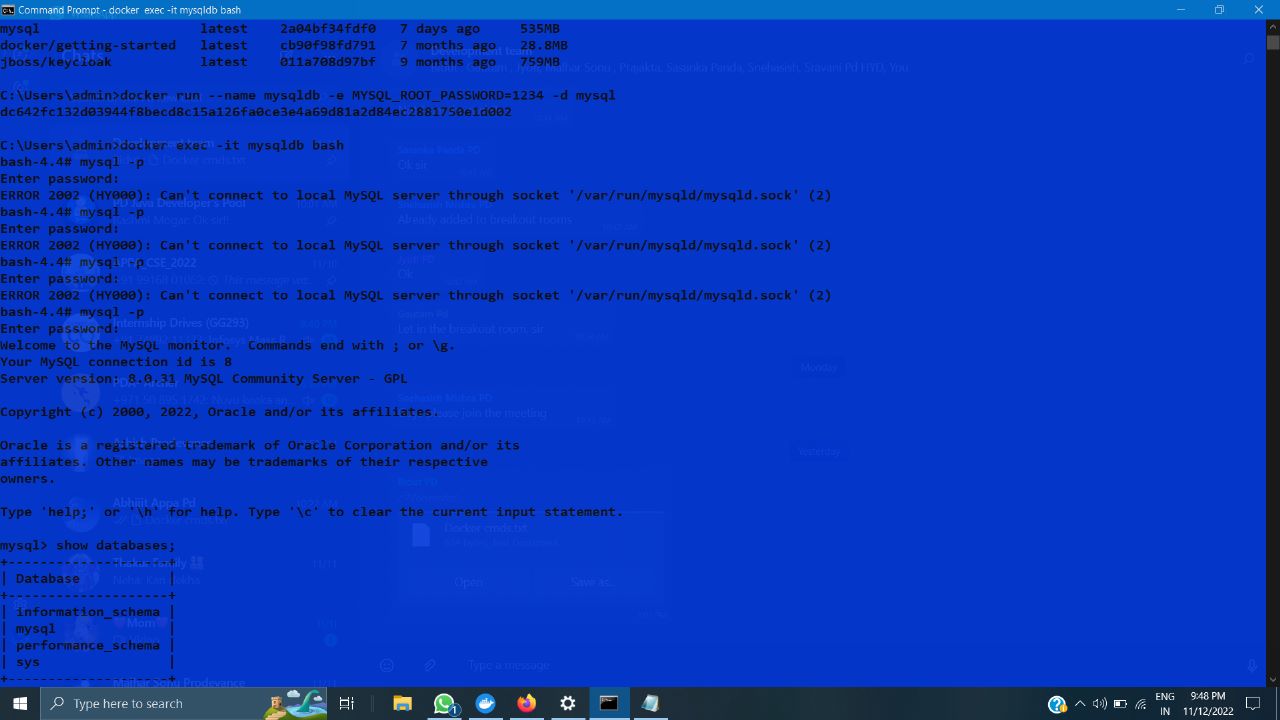
**After all done means 1 to 8 when again you are need to start then start from 4 stp**

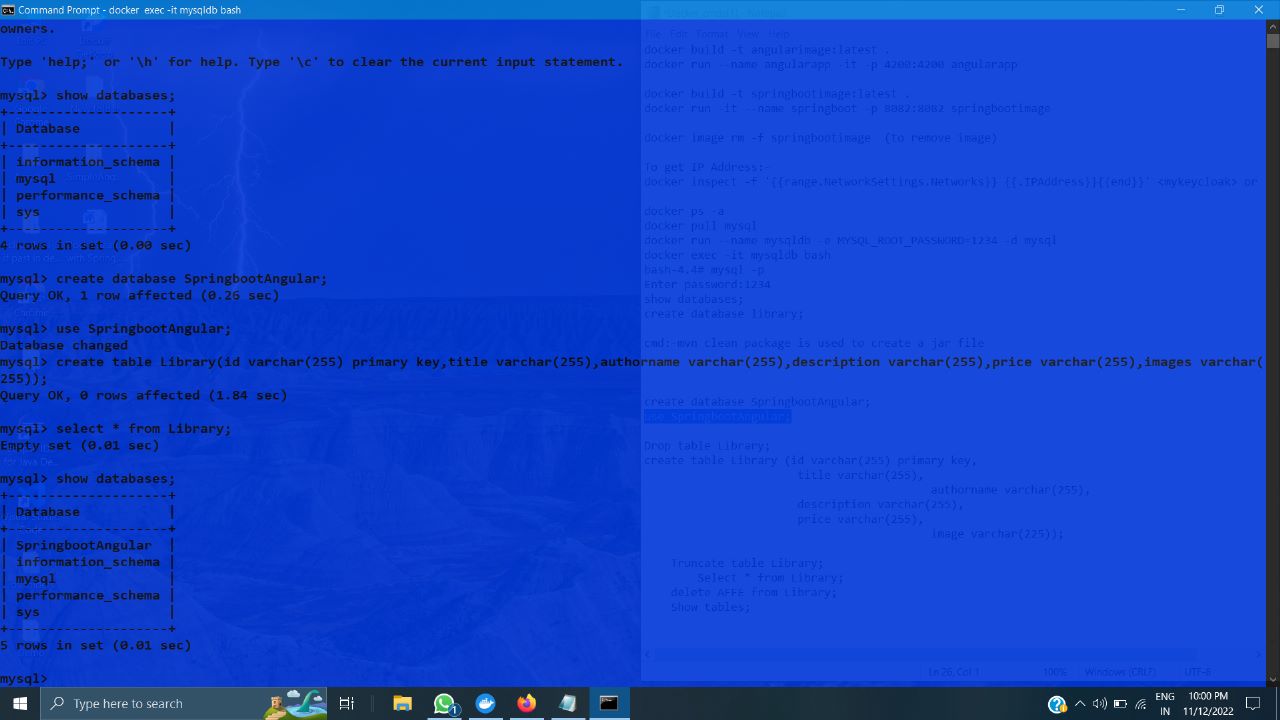
**5) bash-4.4# mysql -p**

**6) Enter password:1234 or (Akash@123)**

**7) show databases;**

**8) create database library;**

****

****

Keycloak Diployment:-

Open docker hub or use this link :- ***https://hub.docker.com/settings/general***

Search jboss/keycloak

Copy Docker Pull Command:- docker pull jboss/keycloak

1)Paste in cmd and Enter

And follow below by entering in cmd one by one-

2) docker run jboss/keycloak

3) docker run -p 8080:8080 jboss/keycloak

4) docker run -e KEYCLOAK\_USER=<USERNAME> -e KEYCLOAK\_PASSWORD=<PASSWORD> jboss/keycloak

Some time you avoide Step 4.

Now you are Good to go.

GetIP Address of Cointainers:-

Mysql cointainer Ip;-open cmd and type

docker inspect -f "{{range .NetworkSettings.Networks}}{{.IPAddress}}{{end}}" mysqldb

or

docker inspect -f '{{range .NetworkSettings.Networks}}{{.IPAddress}}{{end}}' container\_name\_or\_id

use Double Quates if not run

This are Sum Addition al CMD or same commands

docker build -t angularimage:latest .

docker run --name angularapp -it -p 4200:4200 angularapp

docker build -t springbootimage:latest .

docker run -it --name springboot -p 8082:8082 springbootimage

docker image rm -f springbootimage (to remove image)

To get IP Address:-

docker inspect -f “{{range.NetworkSettings.Networks}} {{.IPAddress}}{{end}}” <mykeycloak> or <mysqldb> (without this <>)

docker ps -a

docker pull mysql

docker run --name mysqldb -e MYSQL\_ROOT\_PASSWORD=1234 -d mysql

docker exec -it mysqldb bash

bash-4.4# mysql -p

Enter password:1234

show databases;

create database library;

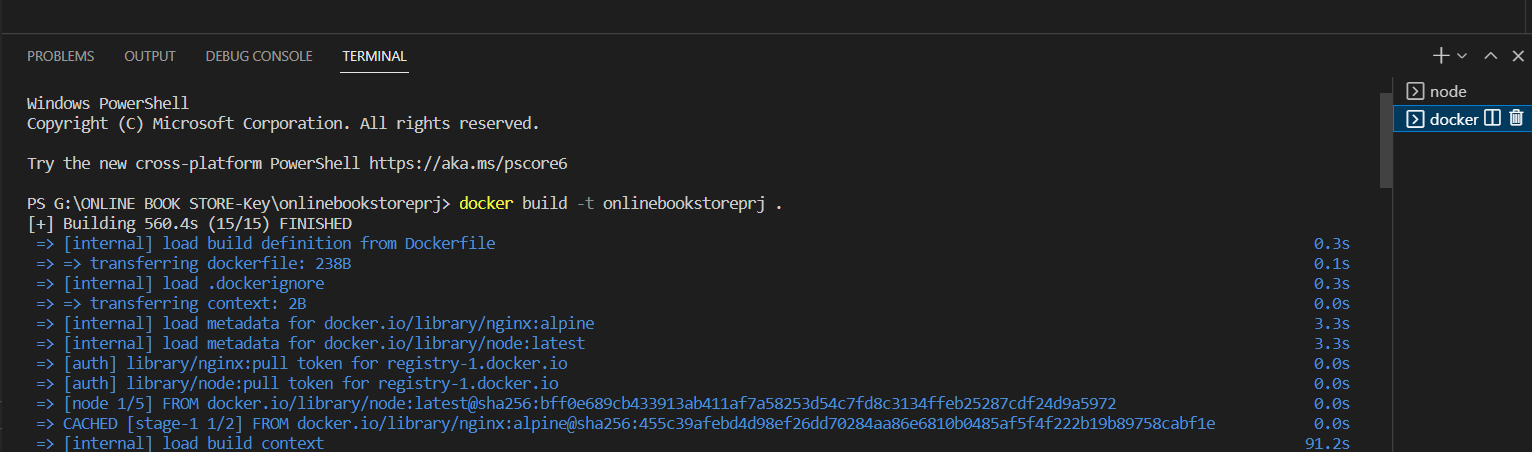
cmd:-mvn clean package is used to create a jar file

For Main project means mysql connected with spring,spring connected with anguler,angular connected with keycoak

1)Open anguler project and in terminal run or type ng build to create a disk file

2) run ng serve

1.0\_)open text terminal of angular like click + icon and run following command like this you need to have to terminal



Before that 1.0 Add Dockerfile to your project by File naming Dockerfile

Open it and add then below content on that

#stage 1

FROM node:latest as node

WORKDIR /app

COPY . .

RUN npm install

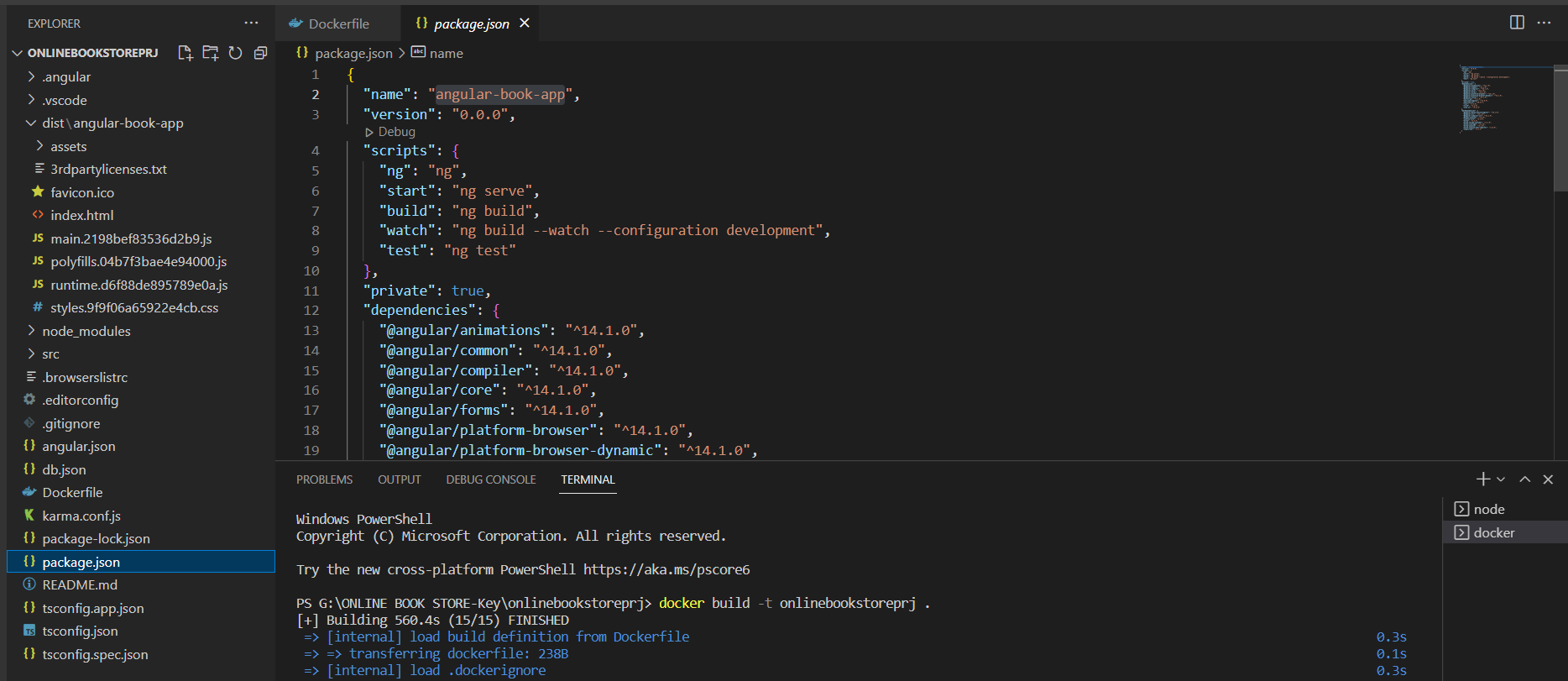
RUN npm run build --prod

#stage 2

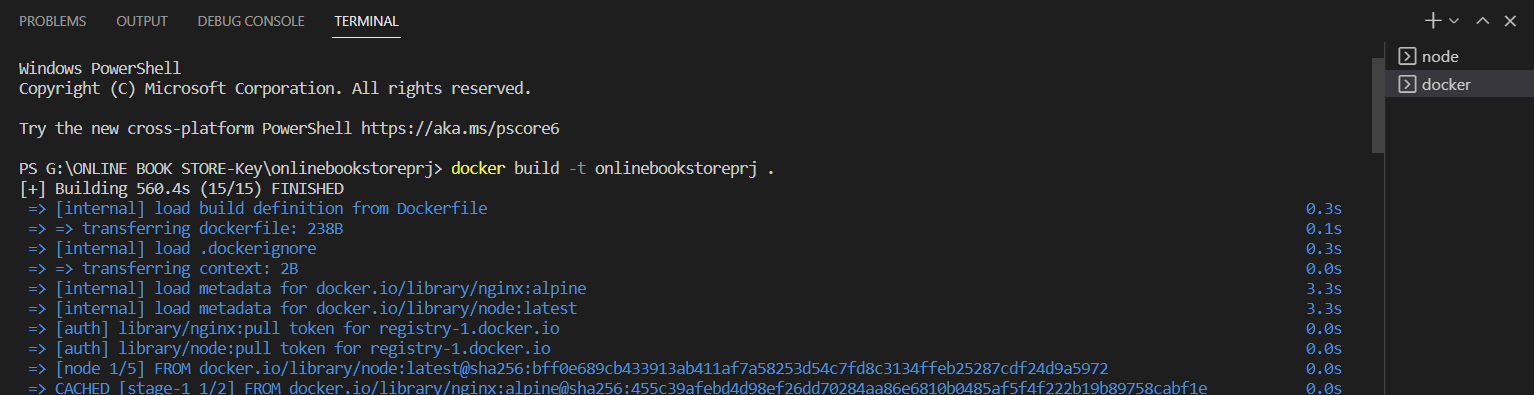
FROM nginx:alpine

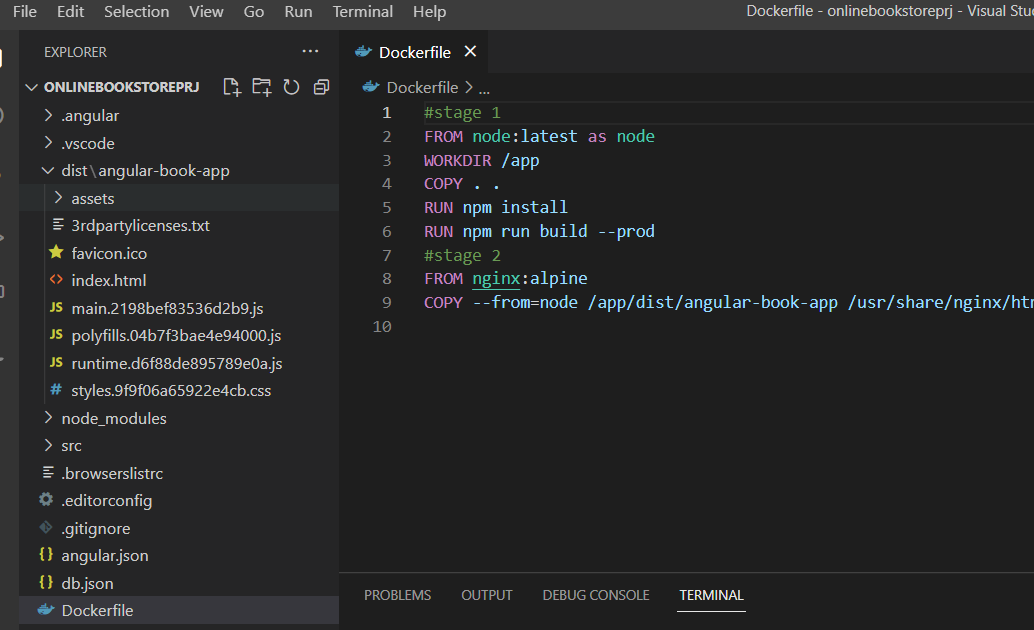
COPY --from=node /app/dist/angular-book-app /usr/share/nginx/html

In this last line angular-book-app is file name pick from package.json file like below at first line you can able to see

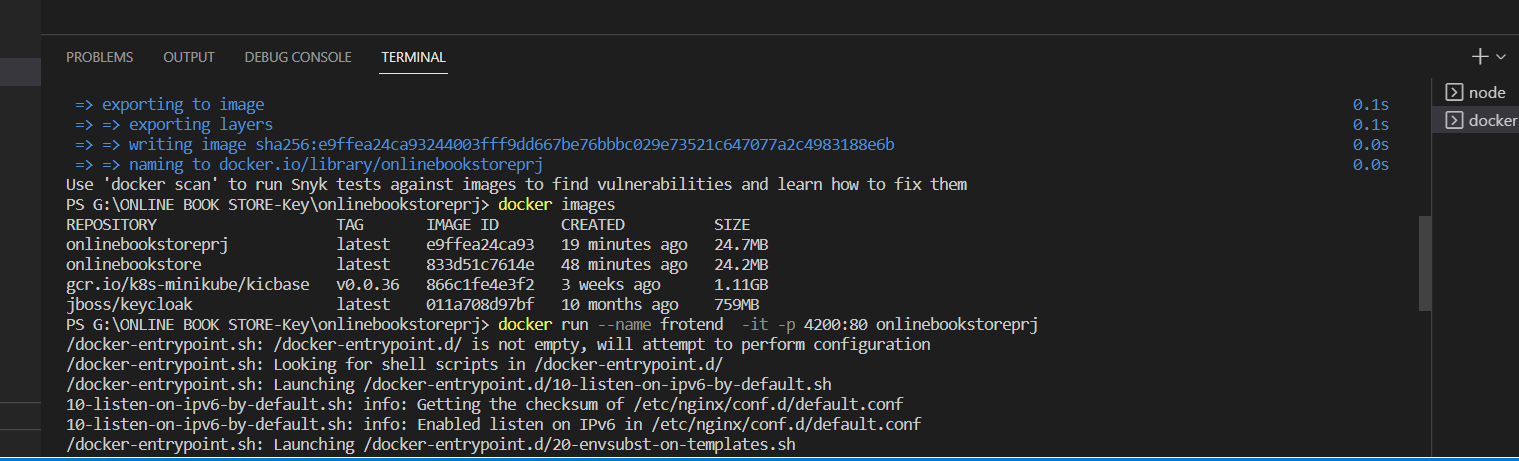


And run next terminal by following like this

here the onlinebookstoreprj is file name or prj name like below



And then fire this command in below like that



Done

==================Docker part==============

1 <https://www.youtube.com/watch?v=J9uKG22lBwA>

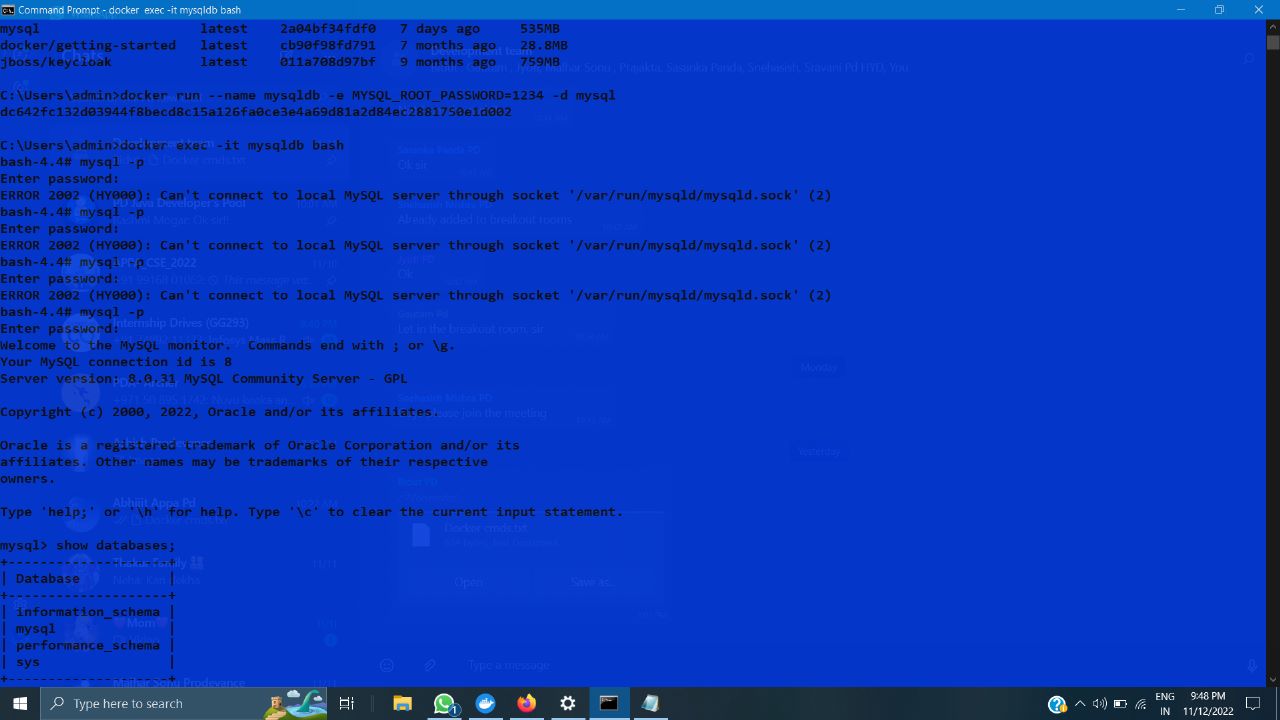
2 <https://wkrzywiec.medium.com/build-and-run-angular-application-in-a-docker-container-b65dbbc50be8>

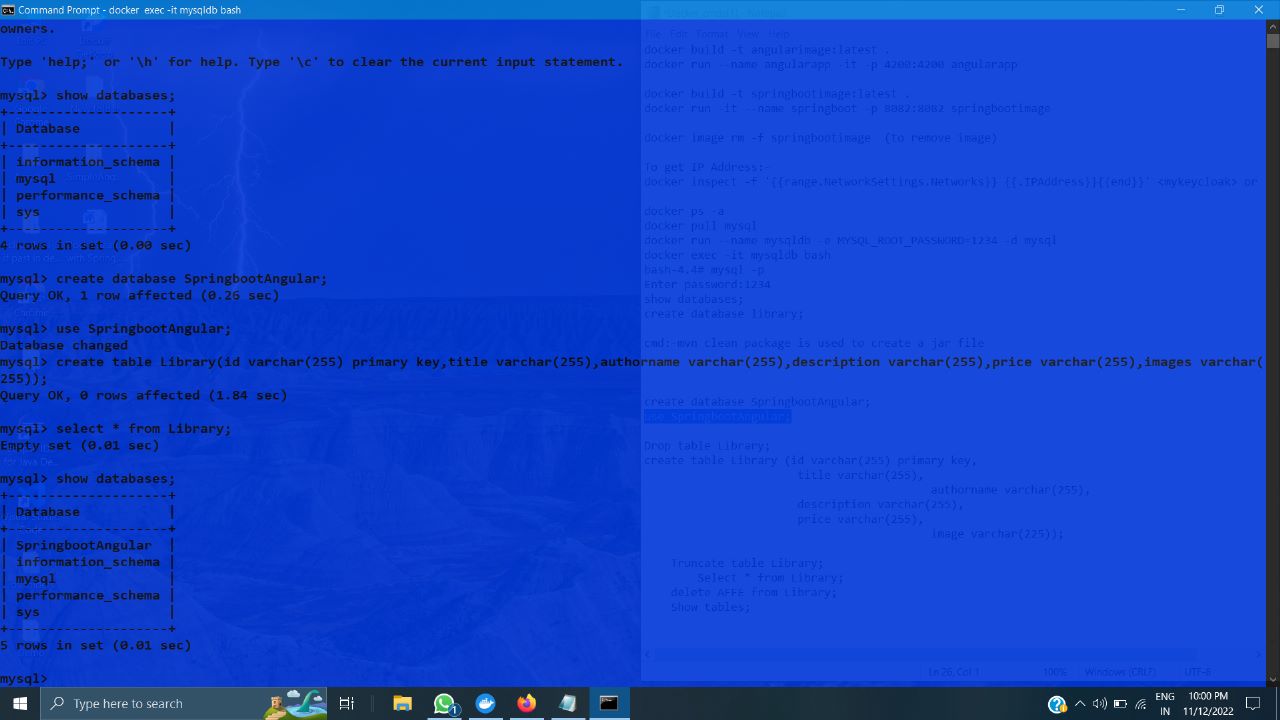
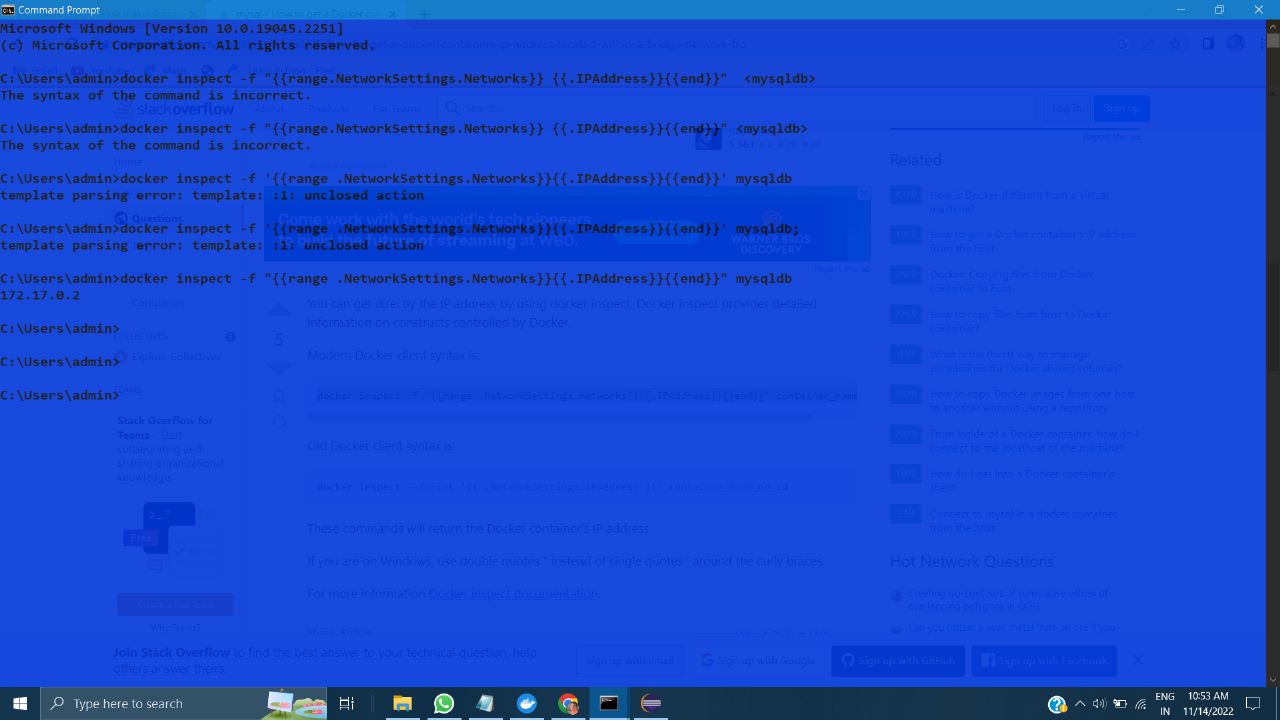
3 <https://dev.to/oneofthedevs/docker-angular-nginx-37e4>

4 <https://hub.docker.com/r/jboss/keycloak>

5 <https://medium.com/codex/dockerize-angular-application-69e7503d1816>

Working angular docker ,this below fo rmysql

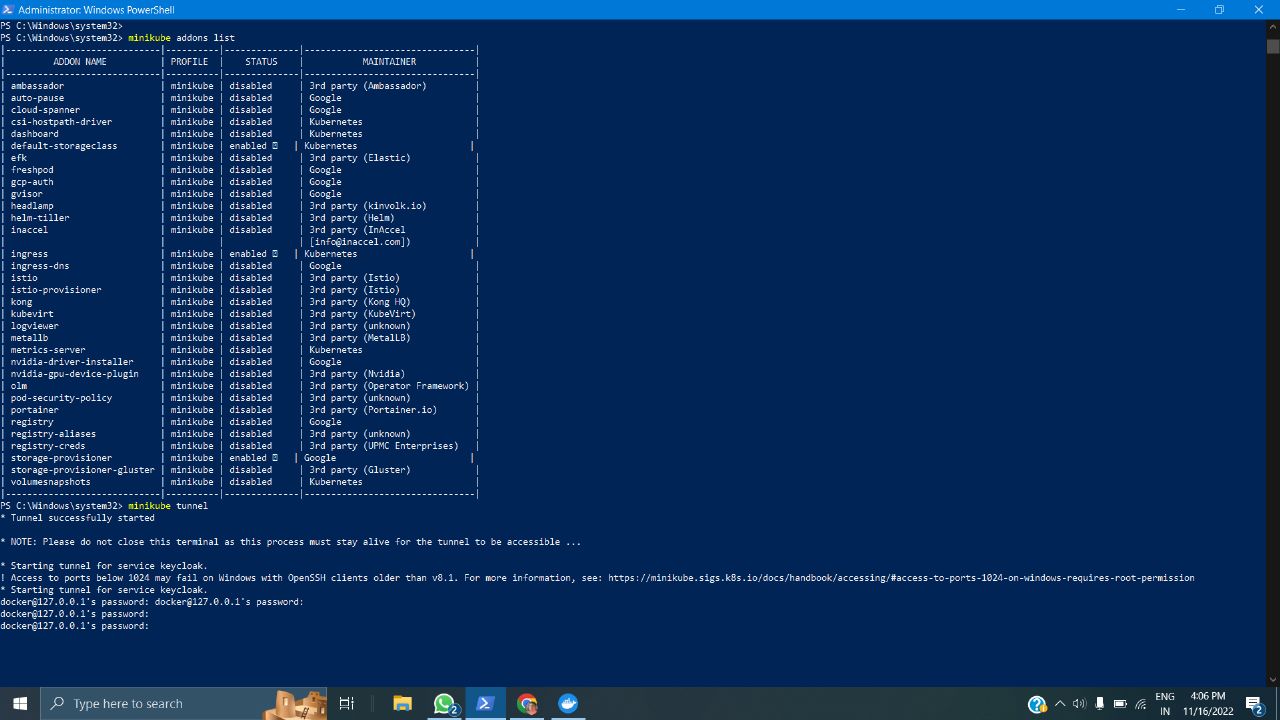


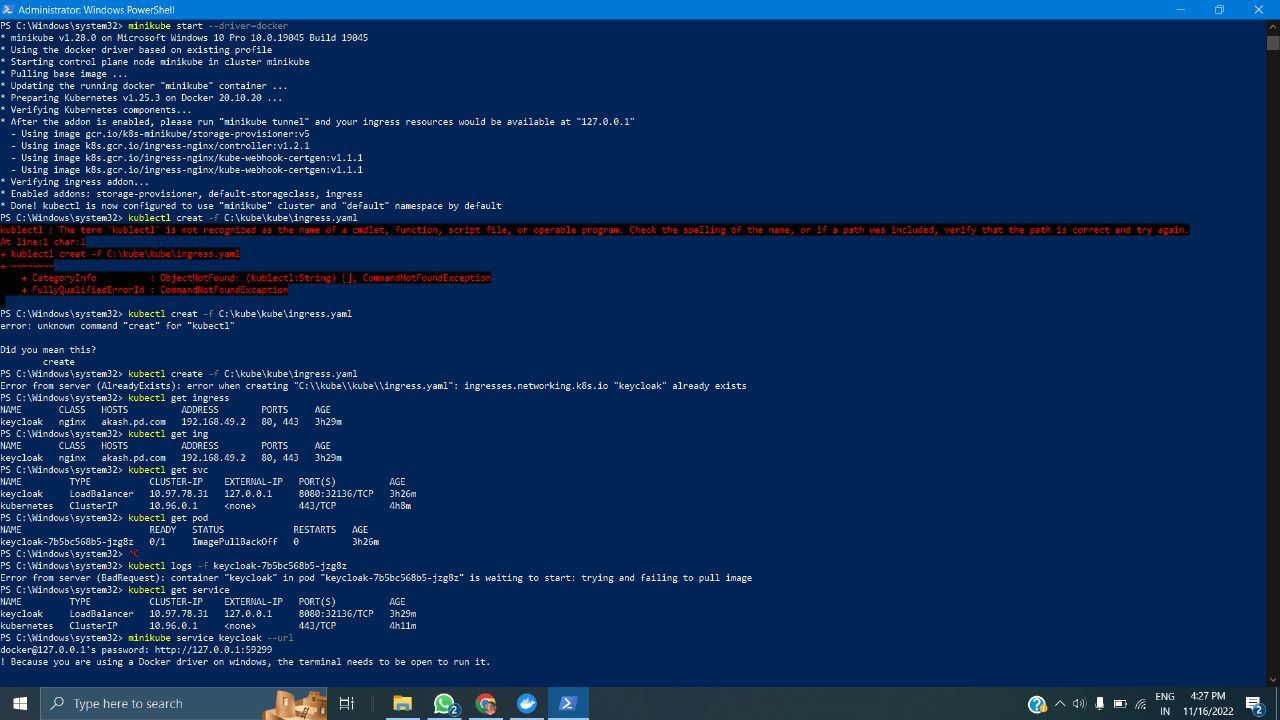


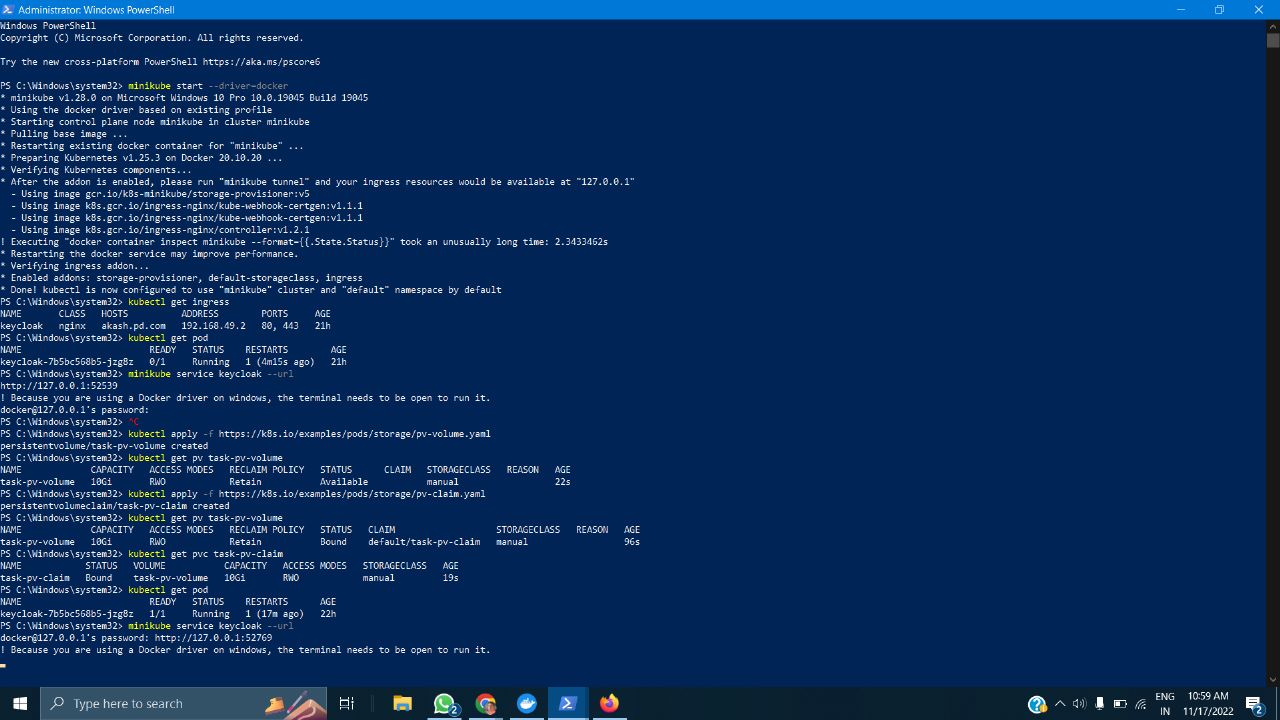
<https://minikube.sigs.k8s.io/docs/start/>

<https://kubernetes.io/docs/tasks/tools/install-kubectl-windows/>

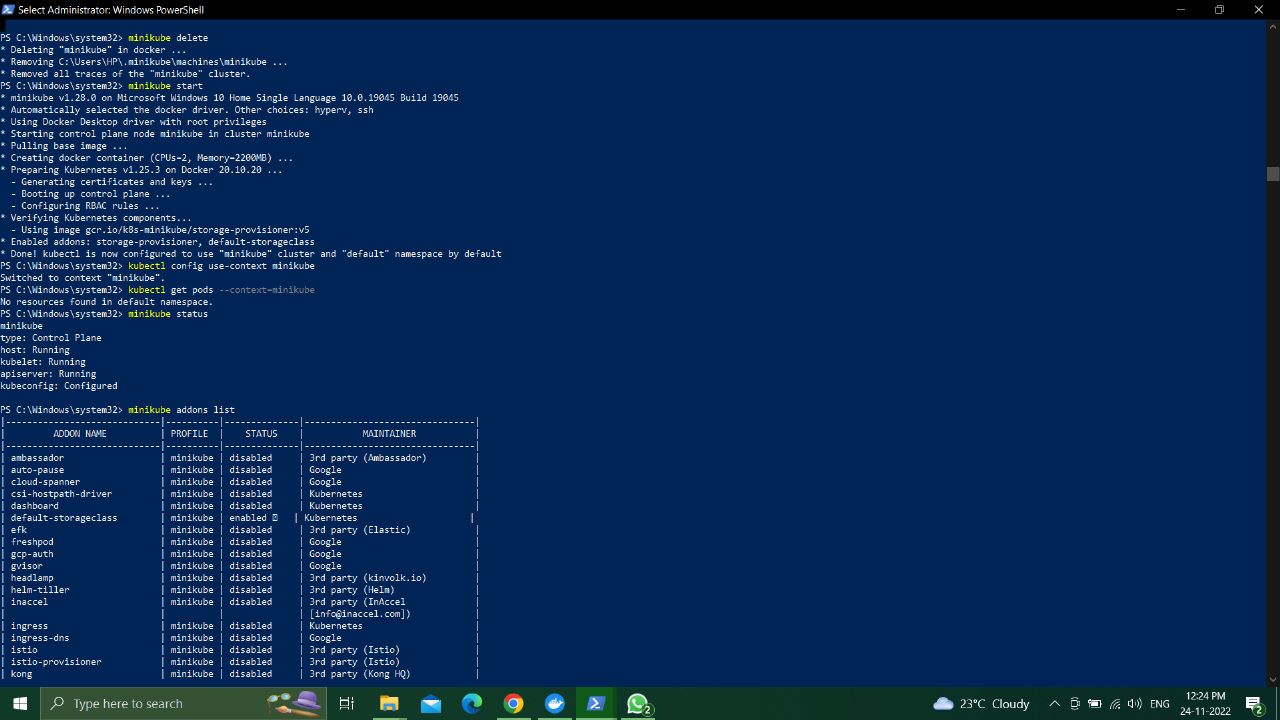
Installation on minikube and kubectl for kubernetes

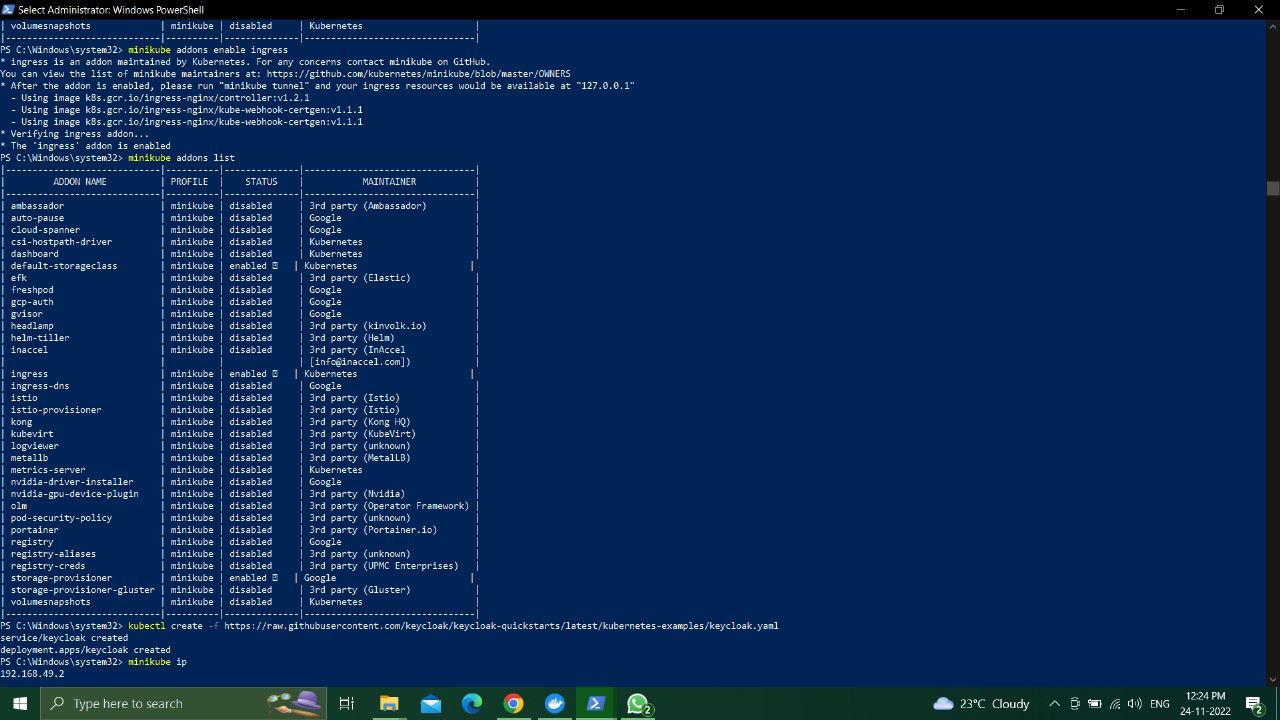


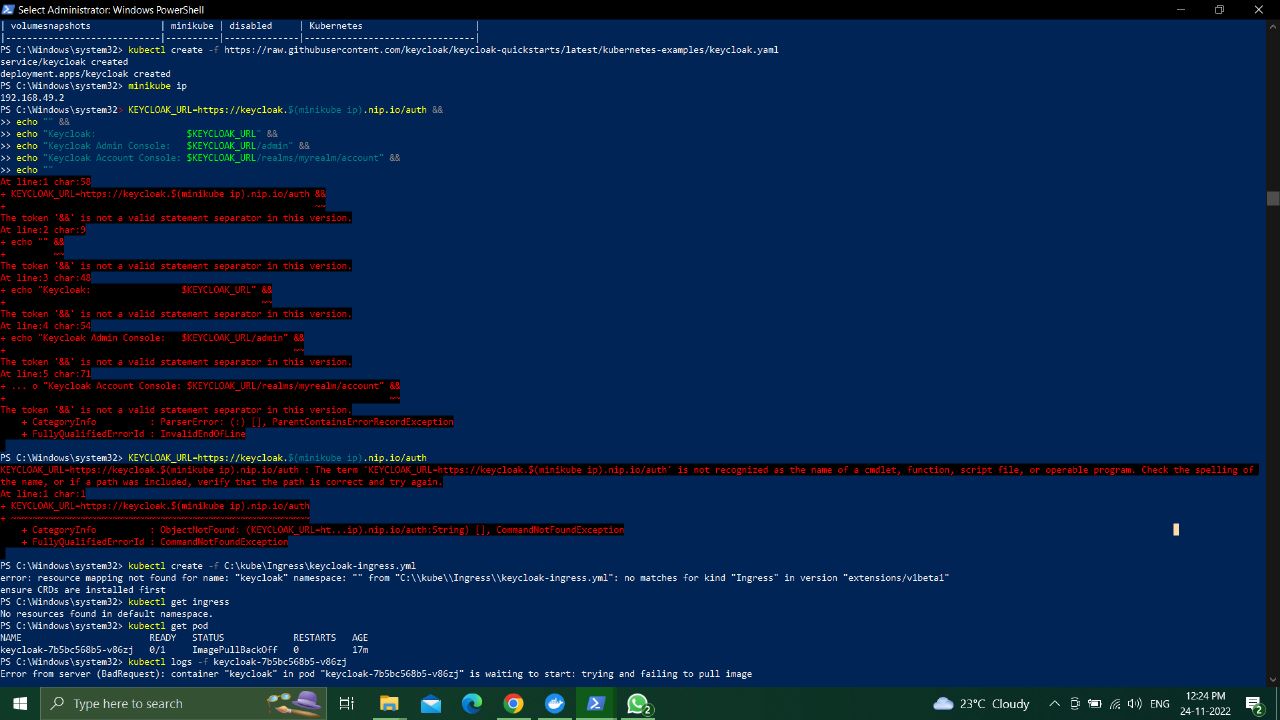


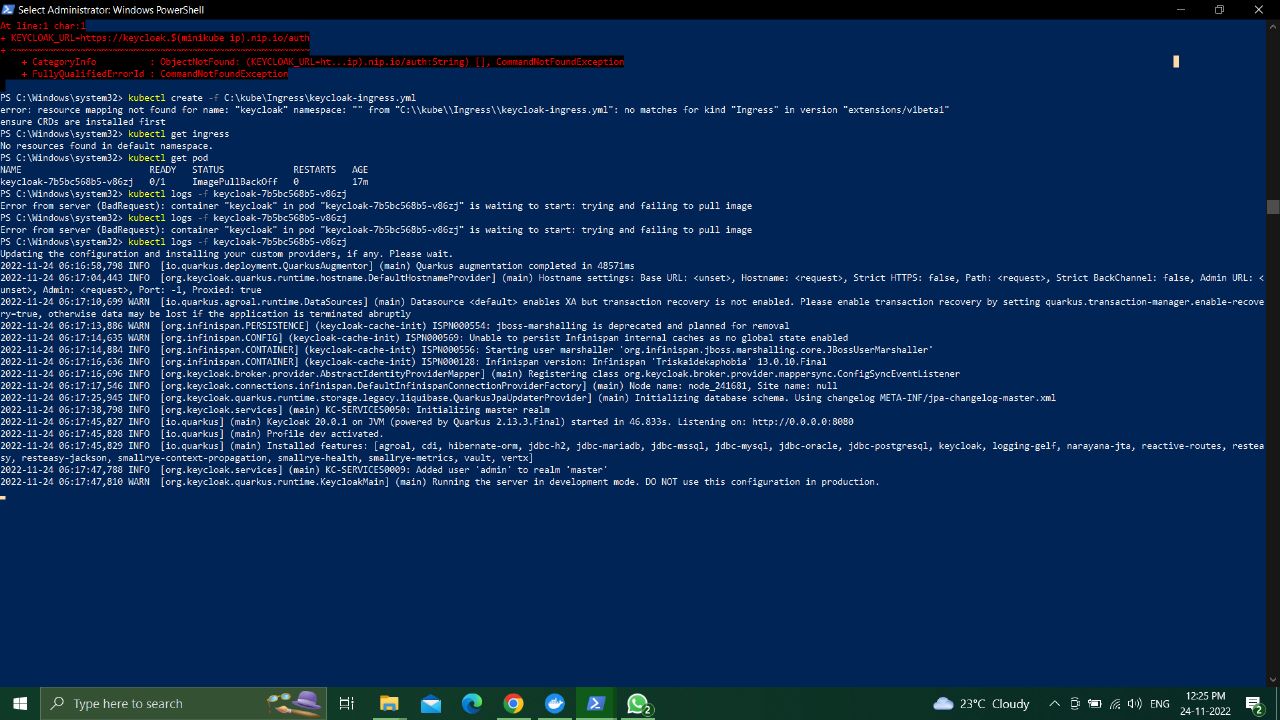


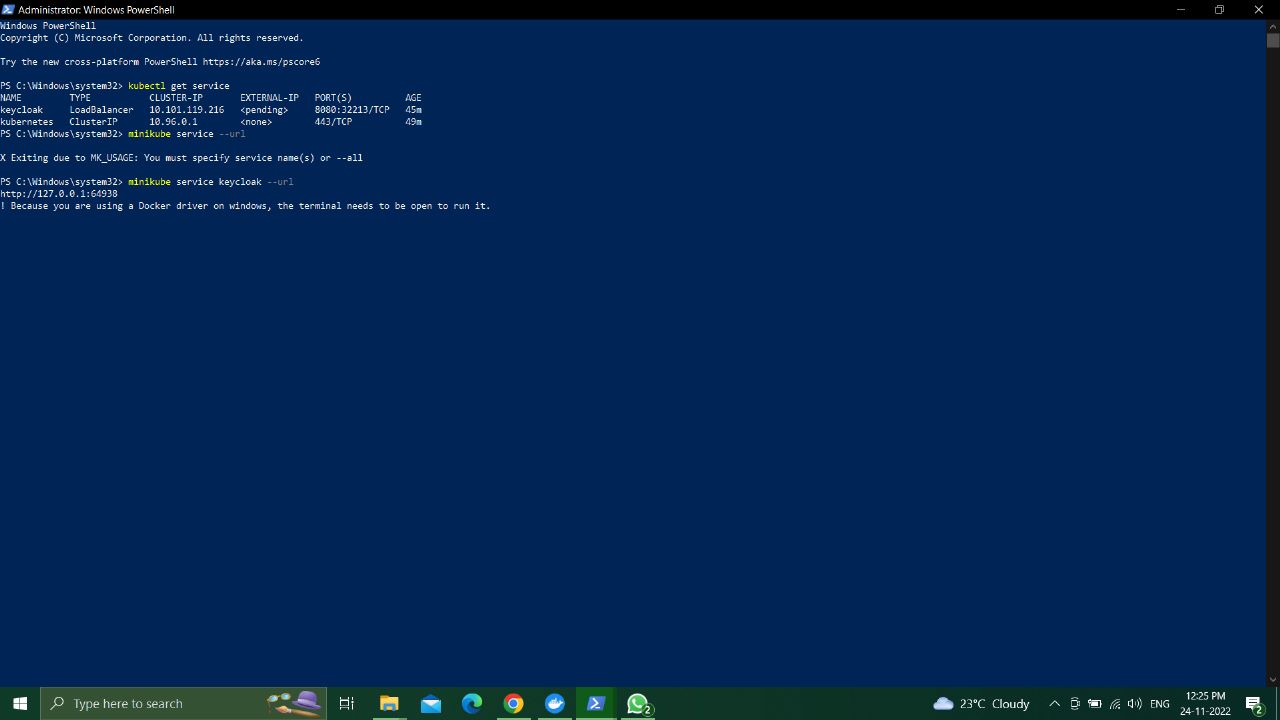
Minikube in Keycloak

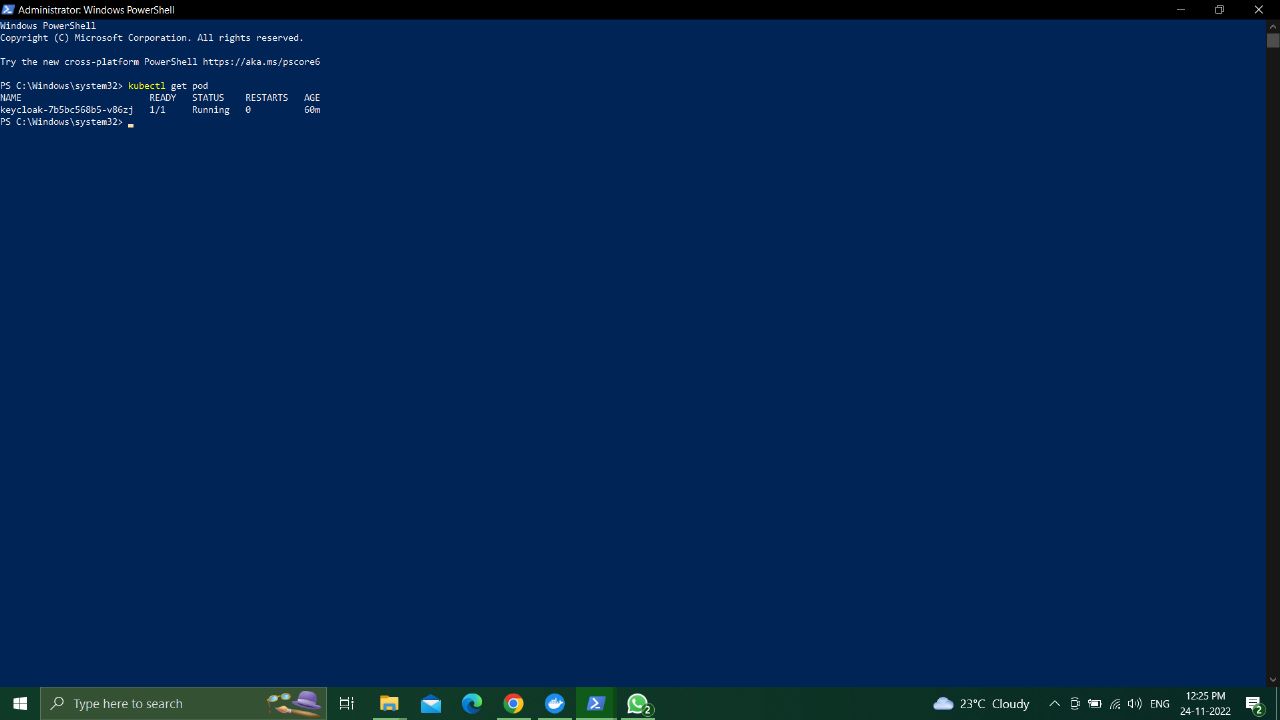












reference for minikube keycloak:- <https://nani2ratna.medium.com/keycloak-in-minikube-c78439327a79>

https://kubernetes.io/docs/tasks/configure-pod-container/configure-persistent-volume-storage/#create-a-persistentvolume

For mount volume like 1/1

https://www.keycloak.org/getting-started/getting-started-kube

Other reference

====Spring push with Docker hub=========

<https://www.youtube.com/watch?v=SzbeDqBSRkc&t=1211s>

<https://www.youtube.com/watch?v=aH1IwAPHe1w>

==spring yaml in kubernetes=== <https://www.section.io/engineering-education/spring-boot-kubernetes/>

<https://www.youtube.com/watch?v=ICkQwGlj9Zo>

for add delete yaml file In spring boot <https://www.youtube.com/watch?v=ICkQwGlj9Zo>

for angular application to diploy in kubernetes, see once yaml file in onlinebookstore angular file in = <https://dev.to/saiprasad2595/deploy-angular-application-on-kubernetes-3n2f>

FOR KUBERNETES CRUD REFERE THIS one by part.

<https://www.youtube.com/watch?v=gMmcRbd8L5Y&list=PLVz2XdJiJQxybsyOxK7WFtteH42ayn5i9&index=11>