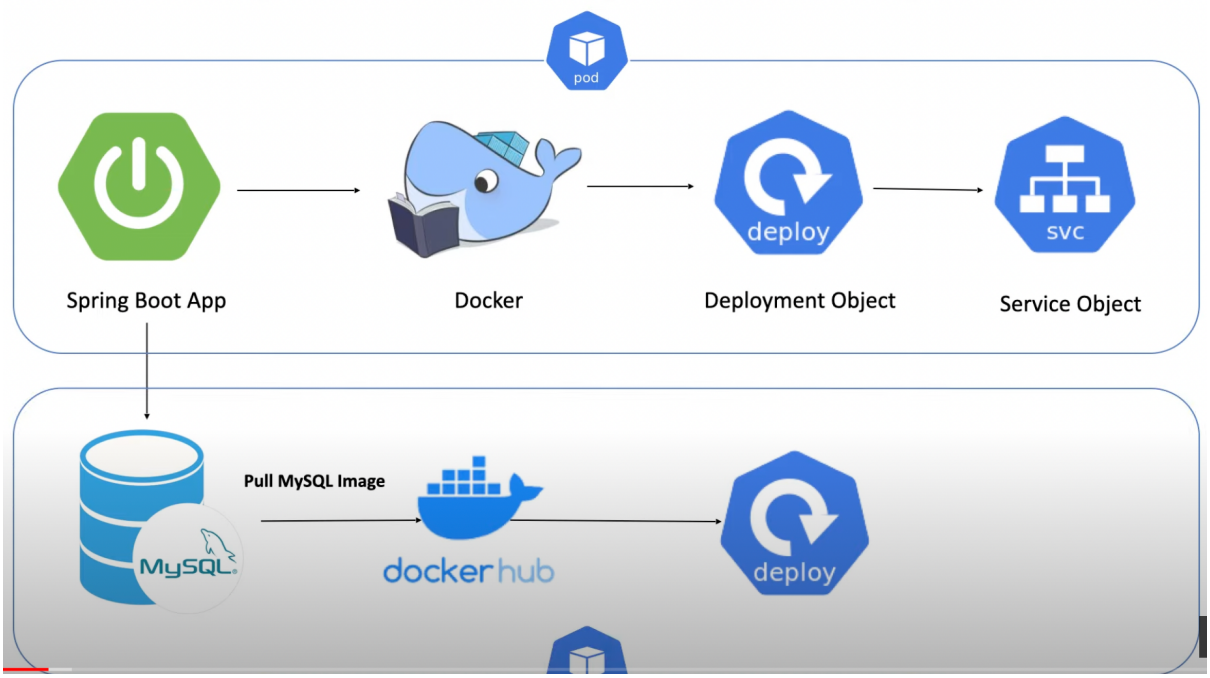


IN STS CLONE BELOW CODE :

https://github.com/DEVOPS-WITH-WEB-DEV/springboot-k8s.git

PLAYLIST FOR OUR BOOTCAMP2 WITH HANDSON VIDEO :

https://www.youtube.com/playlist?list=PLj-3PZlPbUVThOSi1QRqNQoTlE04Kl4zN

1. Launch an instance from an Amazon Linux 2 or Amazon Linux AMI with t2.medium
2. Connect to your instance.
3. Update the packages and package caches you have installed on your instance.
`

yum update -y

4. Install the latest Docker Engine packages.

amazon-linux-extras install docker
OR
yum install docker -y

docker -v

5. Start the Docker service.

systemctl start docker
systemctl enable docker

6. Install Conntrack and git:

yum install conntrack -y

yum install git -y

7. Install k8

curl -LO
https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64

sudo install minikube-linux-amd64 /usr/local/bin/minikube

8. Start Minikube

/usr/local/bin/minikube start --force --driver=docker

/usr/local/bin/minikube version

9. Install kubectl

curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"

sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl

/usr/local/bin/kubectl version

9.1 – Clone the repo

cd /opt/
git clone https://github.com/DEVOPS-WITH-WEB-DEV/springboot-k8s.git

10. Make the DB UP

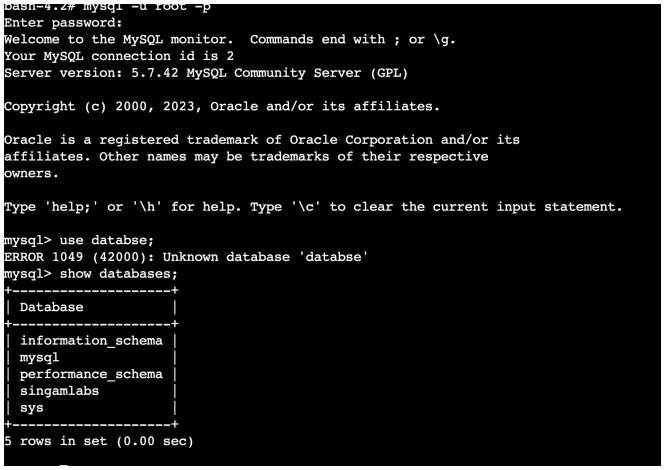
/usr/local/bin/kubectl create -f db-deployment.yaml

/usr/local/bin/kubectl get pods

/usr/local/bin/kubectl exec -it mysql-f759455cd-2dh8m /bin/bash

mysql -u root -p root

show databases;



11. yum install maven -y

12. Create the docker image

docker build -t praveensingam1994/springboot-crud-k8s:1.0 .

13. docker login [CREATE A DOCKER HUB ACCOUNT BEFORE]

Give dockehub username and password

docker image push praveensingam1994/springboot-crud-k8s:1.0

14. /usr/local/bin/kubectl apply -f app-deployment.yaml

15. /usr/local/bin/kubectl get svc

16. /usr/local/bin/minikube ip

17. PUT PORT FORWARD

/usr/local/bin/kubectl port-forward --address 0.0.0.0 svc/springboot-crud-svc 8080:8080 &

[HOST PORT TO CONTAINER PORT]

kubectl port-forward --address 0.0.0.0 svc/{your service name} {external port to the Internet}:{your service port, the port your app is listening on in it's container}

for example, if my service is named store and is listening on 80

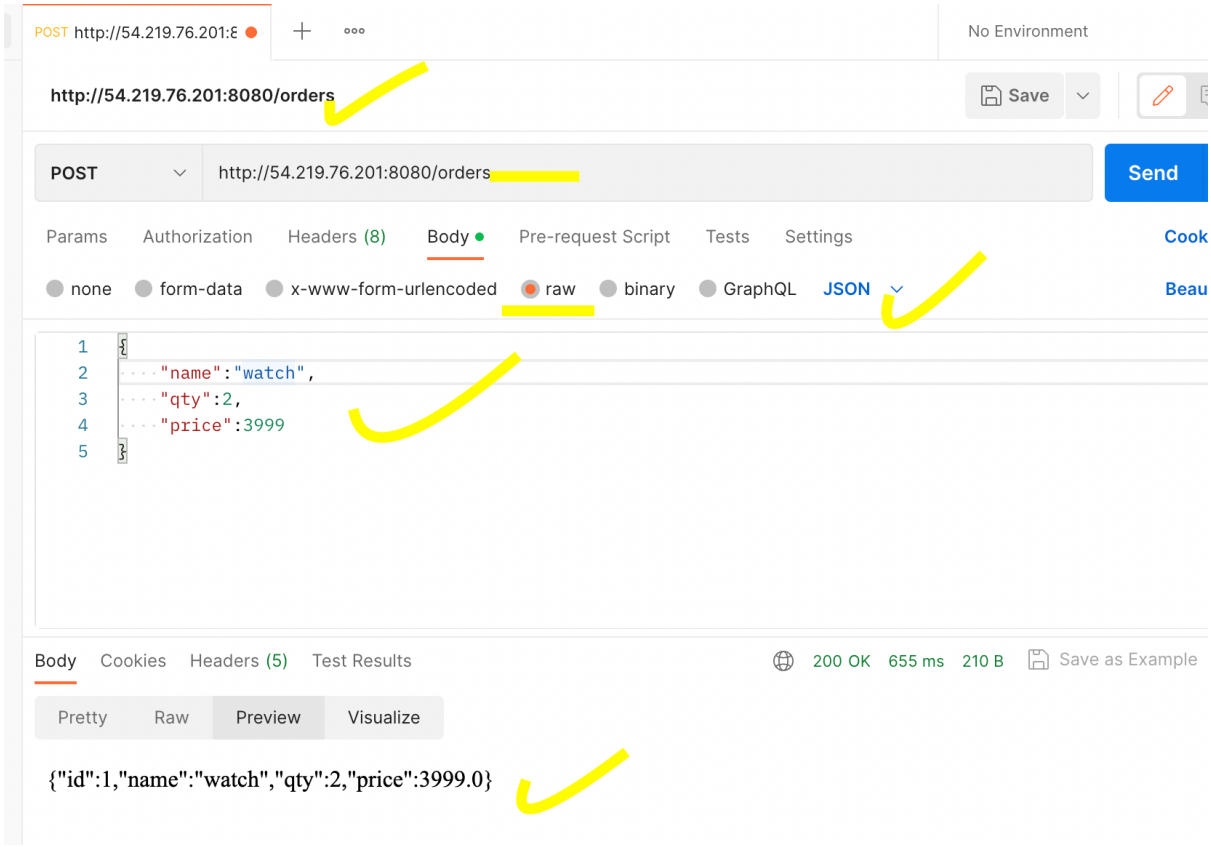
kubectl port-forward --address 0.0.0.0 svc/store 8888:80

18) JSON DATA TO BE HITTED WITH POST

1) URL POST - http://<EC2IP>:8080/orders

DATA TO SENT IN RAW TAB


```
{
   "name":"shoes",
   "qty":5,
   "price":6999
}
```

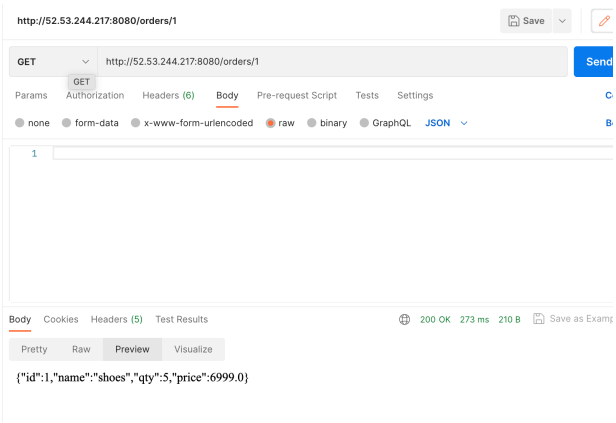


18.1 -> Check Database

```
 singamlabs         |
 sys                |
--------------------+
 rows in set (0.00 sec)

ysql> use singamlabs;
eading table information for completion of table and colu
ou can turn off this feature to get a quicker startup wit

atabase changed
ysql> show tables;
----------------------+
 Tables_in_singamlabs |
----------------------+
 hibernate_sequence   |
 orders_tbl           |
----------------------+
 rows in set (0.00 sec)

ysql> select * from orders_tbl;
----+-------+-------+-----+
 id | name  | price | qty |
----+-------+-------+-----+
  1 | shoes |  6999 |   5 |
  2 | chair |    99 |   3 |
----+-------+-------+-----+
 rows in set (0.00 sec)
```

URL GET - http://<EC2IP>:8080/orders/1

PRAVEEN SINGAMPALLI



19 - FOR DASHBOARD:

IN ONE TERMINAL

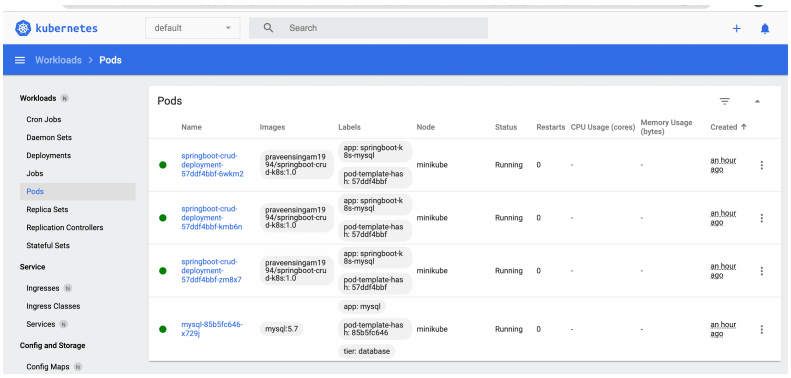
/usr/local/bin/kubectl proxy --address='0.0.0.0' --accept-hosts='^*$'

IN OTHER TERMINAL

/usr/local/bin/minikube dashboard

20 - Hit this url in browser

http://<EC2IP>:8001/api/v1/namespaces/kubernetes-dashboard/services/http:kubernetes-dashboard:/proxy/#/pod?namespace=default



PRAVEEN SINGAMPALLI