Docker Commands:

docker network create my-network

docker pull mysql

docker images

docker run –name mysql-container -e MYSQL\_ROOT\_PASSWORD=admin -e MYSQL\_DATABASE=classicbusinessmodel –network my-network -d mysql

docker inspect my-network

docker ps

docker exec -it mysql-container bash

mysql -u root -p admin

In duplicate session:

git clone <https://github.com/Ravindra08933/Classic_Business_Model.git>

mvn clean package

docker build -t shanmukh8907/classic-business-model .

docker login

docker push shanmukh8907/classic-business-model

docker run –name cbm-container -p 8080:8080 –network my-network -d classic-business-model

docker ps

Deploying our application using EKS:

Kubectl commands:

curl -O <https://s3.us-west-2.amazonaws.com/amazon-eks/1.27.1/2023-04-19/bin/linux/amd64/kubectl>

chmod +x ./kubectl

mkdir -p $HOME/bin && cp ./kubectl $HOME/bin/kubectl && export PATH=$HOME/bin:$PATH

kubectl version –short

sudo apt install unzip

curl "https://awscli.amazonaws.com/awscli-exe-linux-x86\_64.zip" -o "awscliv2.zip"  
unzip awscliv2.zip  
sudo ./aws/install –update

aws configure

aws eks update-kubeconfig –region ap-south-1 –name CBM-cluster

nano mysql-secret.yml

nano mysql-storage.yml

nano mysql-deployment-service.yml

kubectl apply -f mysql-secret.yml

kubectl apply -f mysql-storage.yml

kubectl apply -f mysql-deployment-service.yml

kubectl get all

kubectl exec -it pod/mysql-deployment-fccb94f9f-xmsc9 bash

mysql -u root -p admin

In duplicate session:

nano cbm-deployment-service.yml

kubectl apply -f cbm-deployment-service.yml

kubectl get all

kubectl logs pod/cbm-deployment-5974cbf7dc-bwk4m