Prerequisites:

1. Kubectl installed.
2. Kubernetes cluster is up and running.
3. Docker images for all the microservices

Task 1 command execution:-

Once creation of all the 3 deployment file and 3 service file we can do the apply for the creation of PODS and Service

kubectl apply -f frontend\_deployment.yaml

kubectl apply -f frontend\_service.yaml

kubectl apply -f backend\_deployment.yaml

kubectl apply -f backend\_service.yaml

kubectl apply -f database\_deployment.yaml

kubectl apply -f database\_service.yaml

Once after running the command the pods and service will be created and we can check there status from the following command

kubectl get pods

kubectl get services

Task 2: -

We can directly update the image name in backend\_deployement.yaml file and apply back the changes using command

kubectl apply -f backend\_deployment.yaml

kubectl apply -f backend\_service.yaml

The older PODS will be destroyed and updated with the new image 1by 1

kubectl rollout status backend\_deployment this command can be used to check the deployment status. We can see how many pod is been updated till now and how many are pending

Once after running the command the pods will be updated with new images and its status can be get from the following commands

kubectl get pods

***2nd Option***

We can directly use kubectl set image backend\_deployment backend=backendupdateimage:latest

Where backend is the container name follow by the update image name. Ensure container name is same what we have used in our backend\_deployment.yaml file