

TASK 3- Minikube Deployment Task

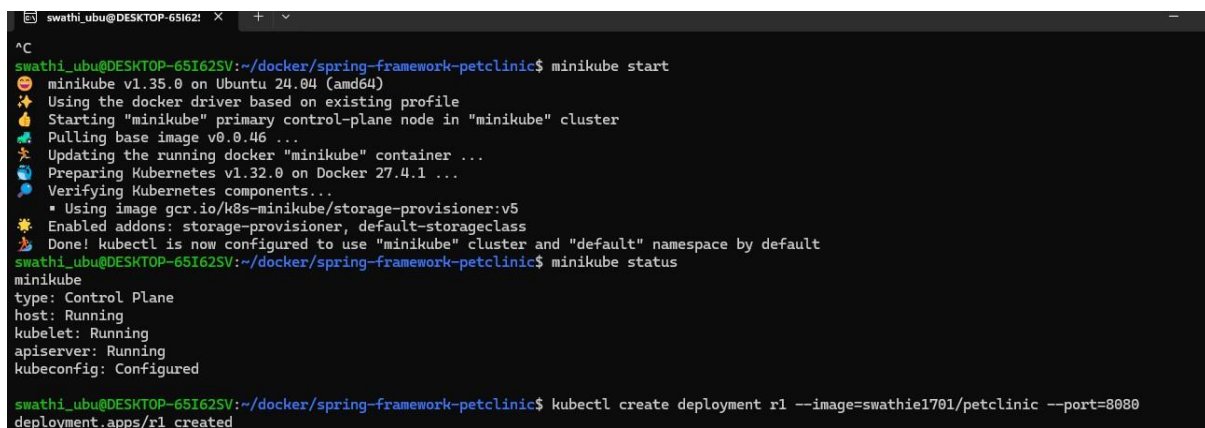
Name:SWATHI K

Rollno: 22CSR216

Step 1:

Start the minikube cluster using the command:

minikube start

A terminal window with a dark background and light green text. The prompt is 'swathi_ubu@DESKTOP-65I62SV:~/docker/spring-framework-petclinic\$'. The first command is 'minikube start', which outputs several status messages including 'minikube v1.35.0 on Ubuntu 24.04 (amd64)', 'Using the docker driver based on existing profile', 'Starting "minikube" primary control-plane node in "minikube" cluster', 'Pulling base image v0.0.46 ...', 'Updating the running docker "minikube" container ...', 'Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...', and 'Verifying Kubernetes components...'. It then lists enabled addons: 'storage-provisioner, default-storageclass'. The second command is 'minikube status', which outputs: 'minikube', 'type: Control Plane', 'host: Running', 'kubelet: Running', 'apiserver: Running', and 'kubeconfig: Configured'. The third command is 'kubectl create deployment r1 --image=swathie1701/petclinic --port=8080', which outputs 'deployment.apps/r1 created'.

This initializes the Minikube cluster using Docker as the driver.

Step 2: Install kubectl

sudo snap install kubectl --classic

Step 3: Verify kubectl Installation

Kubectl version

Step 4: Create a deployment

kubectl create deployment r2 --image=swethamurugesan/devopsgit --port=80

Step 5: Expose the document

kubectl expose deployment r2 --port=80 --type=NodePort

Step 6: Access the service

minikube service r2

```
swathi_ubu@DESKTOP-65I62SV:~/docker/spring-framework-petclinic$ kubectl get pods
NAME                READY   STATUS    RESTARTS   AGE
r1-6c8698c4b6-nhr8l 1/1     Running   0           2m50s
r1-5758bcd4b4-dlwfg 0/1     InvalidImageName 0           103m
swathi_ubu@DESKTOP-65I62SV:~/docker/spring-framework-petclinic$ kubectl expose deployment r1 --port=8080 --type=NodePort
service/r1 exposed
swathi_ubu@DESKTOP-65I62SV:~/docker/spring-framework-petclinic$ minikube service r1

```

NAMESPACE	NAME	TARGET PORT	URL
default	r1	8080	http://192.168.49.2:31484

Starting tunnel for service r1.

NAMESPACE	NAME	TARGET PORT	URL
default	r1		http://127.0.0.1:39883

Opening service default/r1 in default browser...
http://127.0.0.1:39883
Because you are using a Docker driver on linux, the terminal needs to be open to run it.

Step 7: Check the Output in Browser

