Task 3

CREATING A MINIKUBE SERVICE TO HANDLE CONTAINERS

```
archita@DESKTOP-U3693UJ:~$ curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube_latest_amd64.deb
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 37.0M  100 37.0M    0     0  1368k      0  0:00:27  0:00:27 --:--:-- 1311k
```

```
archita@DESKTOP-U3693UJ:~$ sudo dpkg -i minikube_latest_amd64.deb
Selecting previously unselected package minikube.
(Reading database ... 42869 files and directories currently installed.)
Preparing to unpack minikube_latest_amd64.deb ...
Unpacking minikube (1.35.0-0) ...
Setting up minikube (1.35.0-0) ...
```

STARTING MINIKUBE:

```
archita@DESKTOP-U3693UJ:~$ minikube start
😄  minikube v1.35.0 on Ubuntu 24.04 (amd64)
✨  Automatically selected the docker driver. Other choices: none, ssh
📌  Using Docker driver with root privileges
👍  Starting "minikube" primary control-plane node in "minikube" cluster
🚜  Pulling base image v0.0.46 ...
💾  Downloading Kubernetes v1.32.0 preload ...
    > preloaded-images-k8s-v18-v1...:  133.81 MiB / 333.57 MiB  40.11% 739.64 K
    > preloaded-images-k8s-v18-v1...:  333.57 MiB / 333.57 MiB  100.00% 716.85
    > gcr.io/k8s-minikube/kicbase...:  500.31 MiB / 500.31 MiB  100.00% 821.02
🔥  Creating docker container (CPUs=2, Memory=2200MB) ...
🐳  Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...
    ▪ Generating certificates and keys ...
    ▪ Booting up control plane ...
    ▪ Configuring RBAC rules ...
🔗  Configuring bridge CNI (Container Networking Interface) ...
🔎  Verifying Kubernetes components...
    ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🌟  Enabled addons: storage-provisioner, default-storageclass
💡  kubectl not found. If you need it, try: 'minikube kubectl -- get pods -A'
🏄  Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
```

INSTALLING KUBECTL:

```
archita@DESKTOP-U3693UJ:~$ curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/li
nux/amd64/kubectl"
chmod +x kubectl
sudo mv kubectl /usr/local/bin/
kubectl version --client
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   138  100   138    0     0    231      0 --:--:-- --:--:-- --:--:--   231
100 54.6M  100 54.6M    0     0  1773k      0  0:00:31  0:00:31 --:--:-- 3583k
Client Version: v1.32.1
Kustomize Version: v5.5.0
```

Task 3

FINAL OUTPUT:

```
asvita@DESKTOP-MTASIFM:~$ kubectl apply -f task1.txt
deployment.apps/test unchanged
asvita@DESKTOP-MTASIFM:~$ kubectl apply -f task2.txt
service/test-service unchanged
asvita@DESKTOP-MTASIFM:~$ minikube service test-service
|-----------|--------------|-------------|---------------------------|
| NAMESPACE |     NAME     | TARGET PORT |            URL            |
|-----------|--------------|-------------|---------------------------|
| default   | test-service | http/80     | http://192.168.49.2:31210 |
|-----------|--------------|-------------|---------------------------|
🏃  Starting tunnel for service test-service.
|-----------|--------------|-------------|------------------------|
| NAMESPACE |     NAME     | TARGET PORT |          URL           |
|-----------|--------------|-------------|------------------------|
| default   | test-service |             | http://127.0.0.1:34609 |
|-----------|--------------|-------------|------------------------|
🎉  Opening service default/test-service in default browser...
👉  http://127.0.0.1:34609
❗  Because you are using a Docker driver on linux, the terminal needs to be open to run it.
```