

## Kubernetes Assignment – 2

### 1. Create a Pod using a YAML Manifest

```
master@master-vm:~/Desktop$ nano nginx-pod.yaml
master@master-vm:~/Desktop$ kubectl apply -f nginx-pod.yaml
pod/nginx-pod configured
master@master-vm:~/Desktop$ kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
nginx-pod     1/1     Running   1 (22m ago) 18h
nginxpod      0/1     Completed 0           18h
master@master-vm:~/Desktop$ kubectl delete -f nginx-pod.yaml
pod "nginx-pod" deleted
master@master-vm:~/Desktop$
```

### 2. Create and use a ConfigMap

```
master@master-vm:~/Desktop$ minikube start
🐳 minikube v1.35.0 on Ubuntu 20.04
👉 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 ...
👉 Verifying Kubernetes components...
    ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🌟 Enabled addons: default-storageclass, storage-provisioner

! /usr/bin/kubectl is version 1.28.15, which may have incompatibilities with Kubernetes 1.32.0.
  ▪ Want kubectl v1.32.0? Try 'minikube kubectl -- get pods -A'
🏁 Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
master@master-vm:~/Desktop$ kubectl create secret generic db-secret --from-literal=DB_USER=admin --from-literal=DB_PASS=password123
secret/db-secret created
master@master-vm:~/Desktop$ kubectl get secrets
NAME          TYPE          DATA   AGE
db-secret     Opaque        2       22s
my-registry-secret  kubernetes.io/dockerconfigjson  1       18h
master@master-vm:~/Desktop$ kubectl describe secret db-secret
Name:         db-secret
Namespace:    default
Labels:       <none>
Annotations:  <none>

Type: Opaque

Data
====
DB_PASS: 11 bytes
DB_USER: 5 bytes
```