

# Kubernetes Assignment - 01

## Exercise 1: Deploy an Nginx Pod

Step 1: Start a Kubernetes cluster (Minikube or other cluster):

```
master@master-vm:~$ curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 119M 100 119M 0 0 14.2M 0 0:00:08 0:00:08 --:--:-- 18.8M
master@master-vm:~$ sudo install minikube-linux-amd64 /usr/local/bin/minikube
master@master-vm:~$ minikube version
minikube version: v1.35.0
commit: dd5d320e41b5451cdf3c01891bc4e13d189586ed-dirty

master@master-vm:~$ minikube start --driver=docker
minikube v1.35.0 on Ubuntu 20.04
Using the docker driver based on user configuration
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...: 333.57 MiB / 333.57 MiB 100.00% 9.28 Mi
> gcr.io/k8s-minikube/kicbase...: 500.31 MiB / 500.31 MiB 100.00% 8.06 Mi
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

/usr/bin/kubectctl is version 1.29.15, which may have incompatibilities with Kubernetes 1.32.0.
- Want kubectctl v1.32.0? Try 'minikube kubectctl -- get pods -A'
Done! kubectctl is now configured to use "minikube" cluster and "default" namespace by default
master@master-vm:~$ kubectctl run nginx-pod --image=nginx --restart=Never
```

Step 2: Create an Nginx pod

```
master@master-vm:~$ kubectctl run nginx-pod --image=nginx --restart=Never
pod/nginx-pod created
```

Step 3: Verify the pod is running:

```
master@master-vm:~$ kubectctl get pods
NAME      READY   STATUS    RESTARTS   AGE
nginx-pod 1/1     Running   0           27m
```

Step 4: Check pod details:

```
master@master-vm:~$ kubectctl describe pod nginx-pod
Name:         nginx-pod
Namespace:    default
Priority:      0
Service Account: default
Node:         minikube/192.168.49.2
Start Time:   Thu, 13 Mar 2025 11:00:02 +0530
Labels:       run=nginx-pod
Annotations:   <none>
Status:       Running
IP:           10.244.0.4
IPs:          IP: 10.244.0.4
Containers:
```

Step 5: Delete the pod:

```
master@master-vm:~$ kubectl delete pod nginx-pod
pod "nginx-pod" deleted
master@master-vm:~$
```

## Exercise 2: Create an Nginx Deployment and Scale It

Step 1: Create a deployment with Nginx

```
master@master-vm:~$ kubectl create deployment nginx-deployment --image=nginx
deployment.apps/nginx-deployment created
```

Step 2 : Check the deployment

```
master@master-vm:~$ kubectl get deployments
NAME          READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment 1/1     1            1           19s
```

Step 3: Scale the deployment to 3 replicas

```
master@master-vm:~$ kubectl scale deployment nginx-deployment --replicas=3
deployment.apps/nginx-deployment scaled
```

Step 4: Check the running pods

```
master@master-vm:~$ kubectl get pods -o wide
NAME                                READY   STATUS    RESTARTS   AGE   IP            NODE       NOMINATED NODE
nginx-deployment-6cfb98644c-f9fbd   1/1     Running   0          19s   10.244.0.9    minikube   <none>
nginx-deployment-6cfb98644c-lzvqs   1/1     Running   0          84s   10.244.0.7    minikube   <none>
nginx-deployment-6cfb98644c-rhdhc   1/1     Running   0          19s   10.244.0.8    minikube   <none>
```

Step 5: Delete the deployment

```
master@master-vm:~$ kubectl delete deployment nginx-deployment
deployment.apps "nginx-deployment" deleted
master@master-vm:~$
```