Name: Kiran Yadav

Email: kiranyadav1102003@gmail.com

Course Name:Devops and Cloud Computing Course

Assignment Name: Kubernetes Assignment 2

Git Link:https://github.com/Hydra-Dev110/Kubernetes

Drive Link: https://docs.google.com/document/d/1Oj6FsWff1IKw4dExz74CyIDhJnYWb1JWjbWYdImMXww/edit?usp=sharing

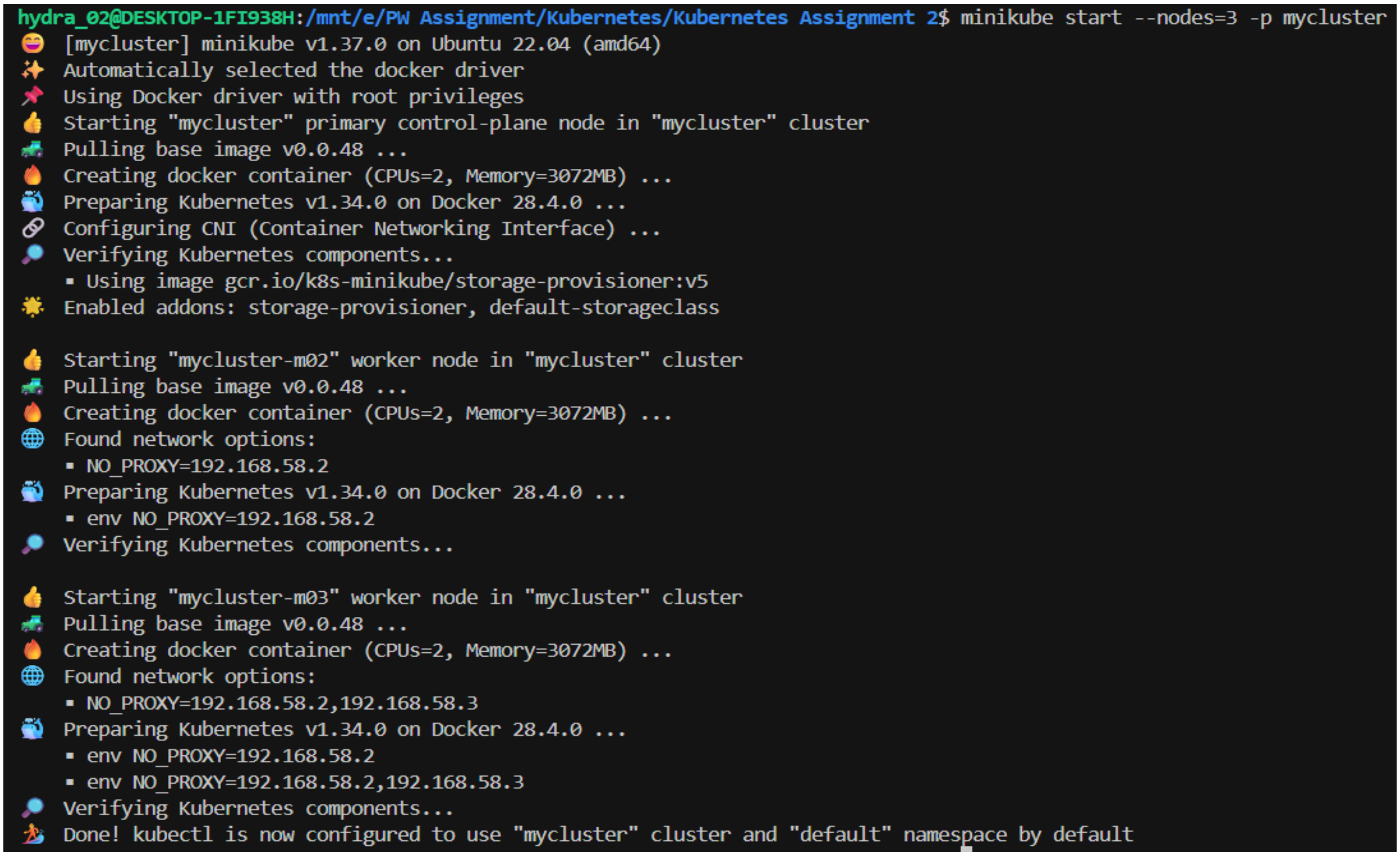
---

Kubernetes Assignment 1

Tasks:

1. Deploy a Kubernetes cluster with 3 nodes.
2. Create an NGINX deployment with 3 replicas.
Steps:

1. Deploy 3-node cluster
   - Using Minikube
     minikube start --nodes=3 -p mycluster



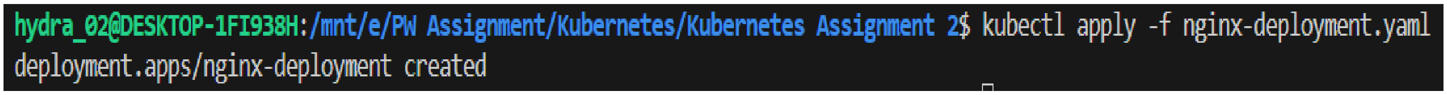
```
hydra_02@DESKTOP-1FI938H:/mnt/e/PW Assignment/Kubernetes/Kubernetes Assignment 2$ minikube start --nodes=3 -p mycluster
😄  [mycluster] minikube v1.37.0 on Ubuntu 22.04 (amd64)
✨  Automatically selected the docker driver
📌  Using Docker driver with root privileges
👍  Starting "mycluster" primary control-plane node in "mycluster" cluster
🚜  Pulling base image v0.0.48 ...
🔥  Creating docker container (CPUs=2, Memory=3072MB) ...
🐳  Preparing Kubernetes v1.34.0 on Docker 28.4.0 ...
🔗  Configuring CNI (Container Networking Interface) ...
🔎  Verifying Kubernetes components...
    ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🌟  Enabled addons: storage-provisioner, default-storageclass

👍  Starting "mycluster-m02" worker node in "mycluster" cluster
🚜  Pulling base image v0.0.48 ...
🔥  Creating docker container (CPUs=2, Memory=3072MB) ...
🌐  Found network options:
    ▪ NO_PROXY=192.168.58.2
🐳  Preparing Kubernetes v1.34.0 on Docker 28.4.0 ...
    ▪ env NO_PROXY=192.168.58.2
🔎  Verifying Kubernetes components...

👍  Starting "mycluster-m03" worker node in "mycluster" cluster
🚜  Pulling base image v0.0.48 ...
🔥  Creating docker container (CPUs=2, Memory=3072MB) ...
🌐  Found network options:
    ▪ NO_PROXY=192.168.58.2,192.168.58.3
🐳  Preparing Kubernetes v1.34.0 on Docker 28.4.0 ...
    ▪ env NO_PROXY=192.168.58.2
    ▪ env NO_PROXY=192.168.58.2,192.168.58.3
🔎  Verifying Kubernetes components...
🏄  Done! kubectl is now configured to use "mycluster" cluster and "default" namespace by default
```

2. Create Nginx Deployment

- YAML(nginx-deployment.yaml)

```
  apiVersion: apps/v1
  kind: Deployment
 metadata:
      name: nginx-deployment
 spec:
      replicas: 3
      selector:
      matchLabels:
      app: nginx
template:
   metadata:
      labels:
         app: nginx
   spec:
     containers:
     - name: nginx
       image: nginx:1.21.6
      ports:
      - containerPort: 80
```

kubectl apply -f nginx-deployment.yaml



```
hydra_02@DESKTOP-1FI938H:/mnt/e/PW Assignment/Kubernetes/Kubernetes Assignment 2$ kubectl apply -f nginx-deployment.yaml
deployment.apps/nginx-deployment created
```

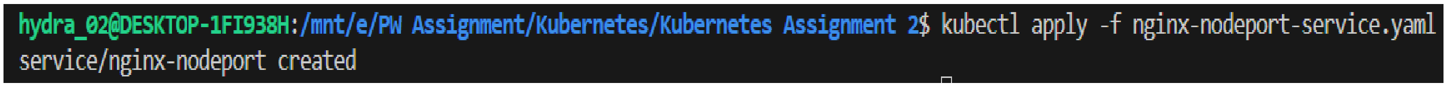
Kubernetes Assignment 2

Tasks:
1. Use the previous deployment
2. Create a Nodeport for NGINX
3.Check the service in a browser

YAML (`nginx-nodeport-service.yaml`):

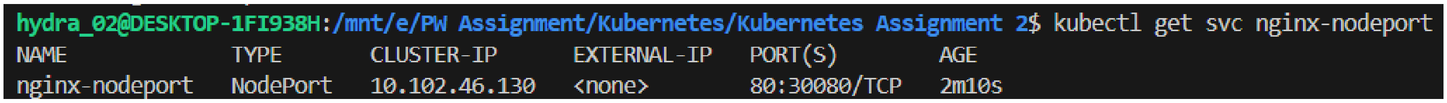
```
apiVersion: v1
kind: Service
metadata:
  name: nginx-nodeport
spec:
  type: NodePort
  selector:
    app: nginx
  ports:
  - port: 80
    targetPort: 80
    nodePort: 30080
```

kubectl apply -f nginx-nodeport-service.yaml



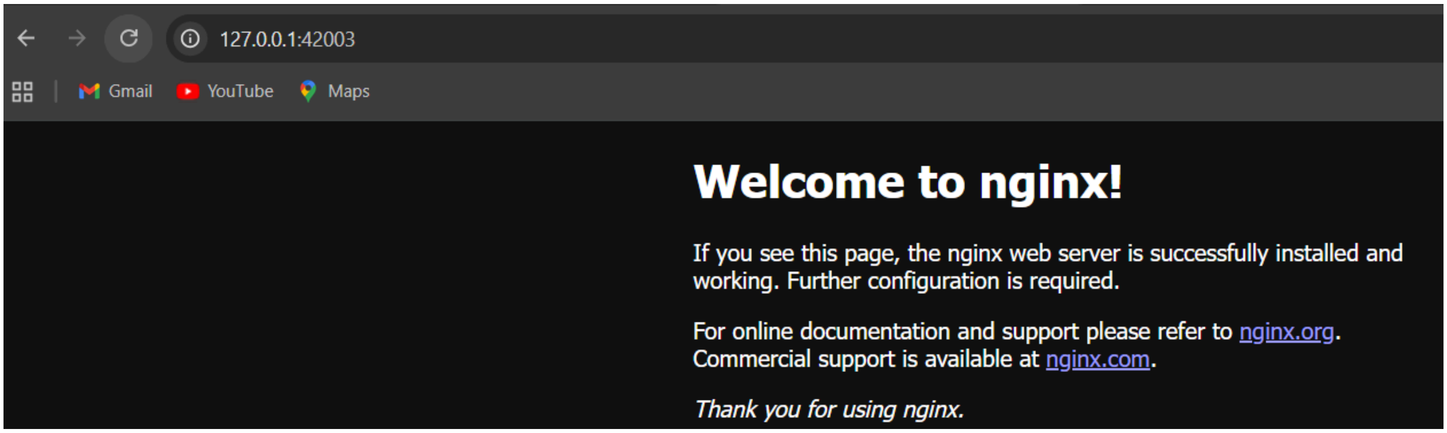
```
hydra_02@DESKTOP-1FI938H:/mnt/e/PW Assignment/Kubernetes/Kubernetes Assignment 2$ kubectl apply -f nginx-nodeport-service.yaml
service/nginx-nodeport created
```

Kubectl get svc nginx-nodeport



```
hydra_02@DESKTOP-1FI938H:/mnt/e/PW Assignment/Kubernetes/Kubernetes Assignment 2$ kubectl get svc nginx-nodeport
NAME             TYPE       CLUSTER-IP       EXTERNAL-IP   PORT(S)        AGE
nginx-nodeport   NodePort   10.102.46.130    <none>        80:30080/TCP   2m10s
```

minikube service nginx-nodeport



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

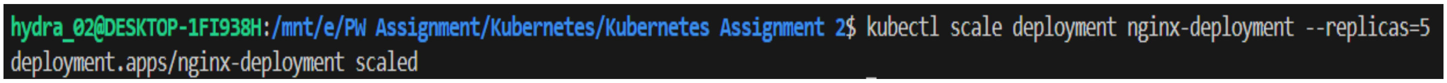
Thank you for using nginx.

Kubernetes Assignment 3

Tasks:

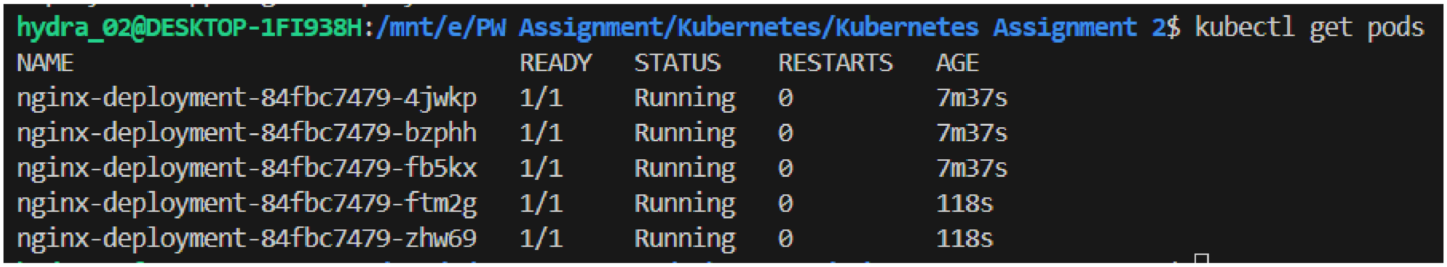
- **Scale NGINX deployment to 5 replicas.**

kubectl scale deployment nginx-deployment --replicas=5



```
hydra_02@DESKTOP-1FI938H:/mnt/e/PW Assignment/Kubernetes/Kubernetes Assignment 2$ kubectl scale deployment nginx-deployment --replicas=5
deployment.apps/nginx-deployment scaled
```

kubectl get pods



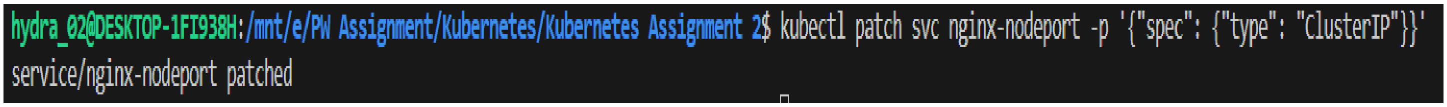
```
hydra_02@DESKTOP-1FI938H:/mnt/e/PW Assignment/Kubernetes/Kubernetes Assignment 2$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
nginx-deployment-84fbc7479-4jwkp   1/1     Running   0          7m37s
nginx-deployment-84fbc7479-bzphh   1/1     Running   0          7m37s
nginx-deployment-84fbc7479-fb5kx   1/1     Running   0          7m37s
nginx-deployment-84fbc7479-ftm2g   1/1     Running   0          118s
nginx-deployment-84fbc7479-zhw69   1/1     Running   0          118s
```

Kubernetes Assignment 4

Tasks:

- **Change the service type to ClusterIP.**

kubectl patch svc nginx-nodeport -p '{"spec": {"type": "ClusterIP"}}'



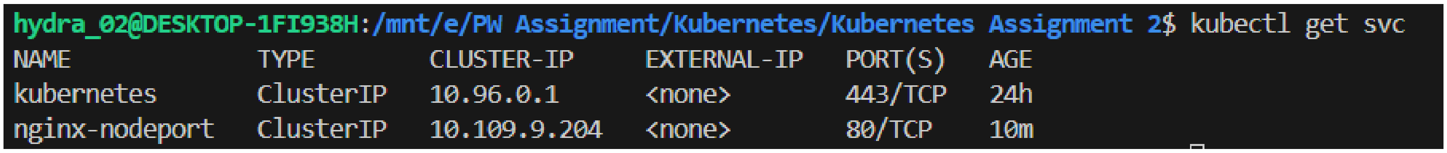
```
hydra_02@DESKTOP-1FI938H:/mnt/e/PW Assignment/Kubernetes/Kubernetes Assignment 2$ kubectl patch svc nginx-nodeport -p '{"spec": {"type": "ClusterIP"}}'
service/nginx-nodeport patched
```

Or update YAML:

```
spec:
  type: ClusterIP
```

Verify:

Kubectl get svc



```
hydra_02@DESKTOP-1FI938H:/mnt/e/PW Assignment/Kubernetes/Kubernetes Assignment 2$ kubectl get svc
NAME             TYPE        CLUSTER-IP     EXTERNAL-IP   PORT(S)   AGE
kubernetes       ClusterIP   10.96.0.1      <none>        443/TCP   24h
nginx-nodeport   ClusterIP   10.109.9.204   <none>        80/TCP    10m
```

Kubernetes Assignment 5

Tasks:

1. **Deploy another NGINX deployment with 3 replicas.**

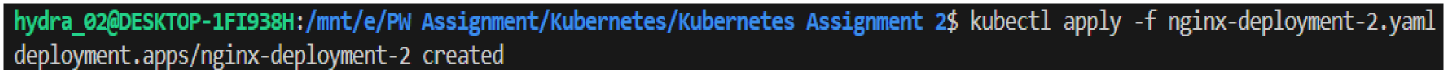
2. **Create an NGINX service of type ClusterIP.**

3. **Create an Ingress service to route traffic to NGINX.**

YAML (`nginx-deployment-2.yaml`):

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment-2
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx2
  template:
    metadata:
      labels:
        app: nginx2
    spec:
      containers:
      - name: nginx
        image: nginx:latest
        ports:
        - containerPort: 80
```

kubectl apply -f nginx-deployment-2.yaml

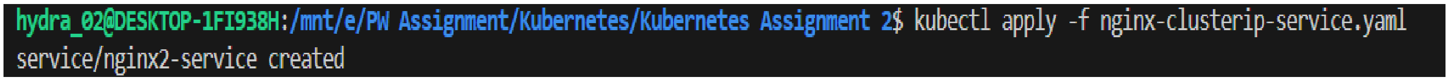


```
hydra_02@DESKTOP-1FI938H:/mnt/e/PW Assignment/Kubernetes/Kubernetes Assignment 2$ kubectl apply -f nginx-deployment-2.yaml
deployment.apps/nginx-deployment-2 created
```

Service (`nginx-clusterip-service.yaml`):

```
apiVersion: v1
kind: Service
metadata:
  name: nginx2-service
spec:
  type: ClusterIP
  selector:
    app: nginx2
  ports:
  - port: 80
    targetPort: 80
```

kubectl apply -f nginx-clusterip-service.yaml

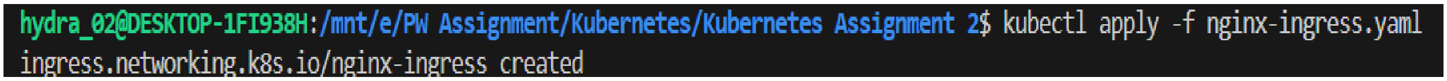


```
hydra_02@DESKTOP-1FI938H:/mnt/e/PW Assignment/Kubernetes/Kubernetes Assignment 2$ kubectl apply -f nginx-clusterip-service.yaml
service/nginx2-service created
```

Ingress (`nginx-ingress.yaml`)

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: nginx-ingress
spec:
  rules:
  - host: nginx.example.com
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: nginx2-service
            port:
              number: 80
```

kubectl apply -f nginx-ingress.yaml



```
hydra_02@DESKTOP-1FI938H:/mnt/e/PW Assignment/Kubernetes/Kubernetes Assignment 2$ kubectl apply -f nginx-ingress.yaml
ingress.networking.k8s.io/nginx-ingress created
```