KUBERNETES ASSIGNMENT – 3

Name: Vikram
Assignment: Scale NGINX Deployment to 5 Replicas

Problem Statement

Using the existing Kubernetes cluster and NGINX deployment:

1. Modify the deployment
2. Increase the number of replicas from 3 to 5

TASK 1: Use Existing Deployment

Verify the deployment from previous assignment:

kubectl get deployments

```
ubuntu@ip-10-0-15-63:~$ kubectl get deployments
NAME                    READY   UP-TO-DATE   AVAILABLE   AGE
assignment1-deployment   3/3     3            3           41m
ubuntu@ip-10-0-15-63:~$ nano assignment1.yaml
ubuntu@ip-10-0-15-63:~$
```

TASK 2: Update Replicas Using YAML

Edit the deployment file:

nano assignment1.yaml

Locate this line:

replicas: 3

Change it to:

replicas: 5

Save and exit.

Apply it:

kubectl apply -f assignment1.yaml

```
ubuntu@ip-10-0-15-63:~$ kubectl get deployments
NAME                    READY   UP-TO-DATE   AVAILABLE   AGE
assignment1-deployment   3/3     3            3           41m
ubuntu@ip-10-0-15-63:~$ nano assignment1.yaml
ubuntu@ip-10-0-15-63:~$
```

```
  GNU nano 7.2                                                    assignme
apiVersion: apps/v1
kind: Deployment
metadata:
  name: assignment1-deployment
  labels:
    app: nginx
spec:
  replicas: 5
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx
        ports:
        - containerPort: 80


^G Help          ^O Write Out     ^W Where Is      ^K Cut           ^T Execute       ^C Loca
^X Exit          ^R Read File     ^\ Replace       ^U Paste         ^J Justify       ^/ Go T
```

i-0ec6451fd14969f99 (k8s-master)

PublicIPs: 54.82.221.39 PrivateIPs: 10.0.15.63

```
ubuntu@ip-10-0-15-63:~$ kubectl get deployments
NAME                        READY   UP-TO-DATE   AVAILABLE   AGE
assignment1-deployment      3/3     3            3           41m
ubuntu@ip-10-0-15-63:~$ nano assignment1.yaml
ubuntu@ip-10-0-15-63:~$ kubectl apply -f assignment1.yaml
deployment.apps/assignment1-deployment configured
```

TASK 3: Verify Scaling

Check deployment status:

kubectl get deployments

Expected output:

READY UP-TO-DATE AVAILABLE

5/5 5 5

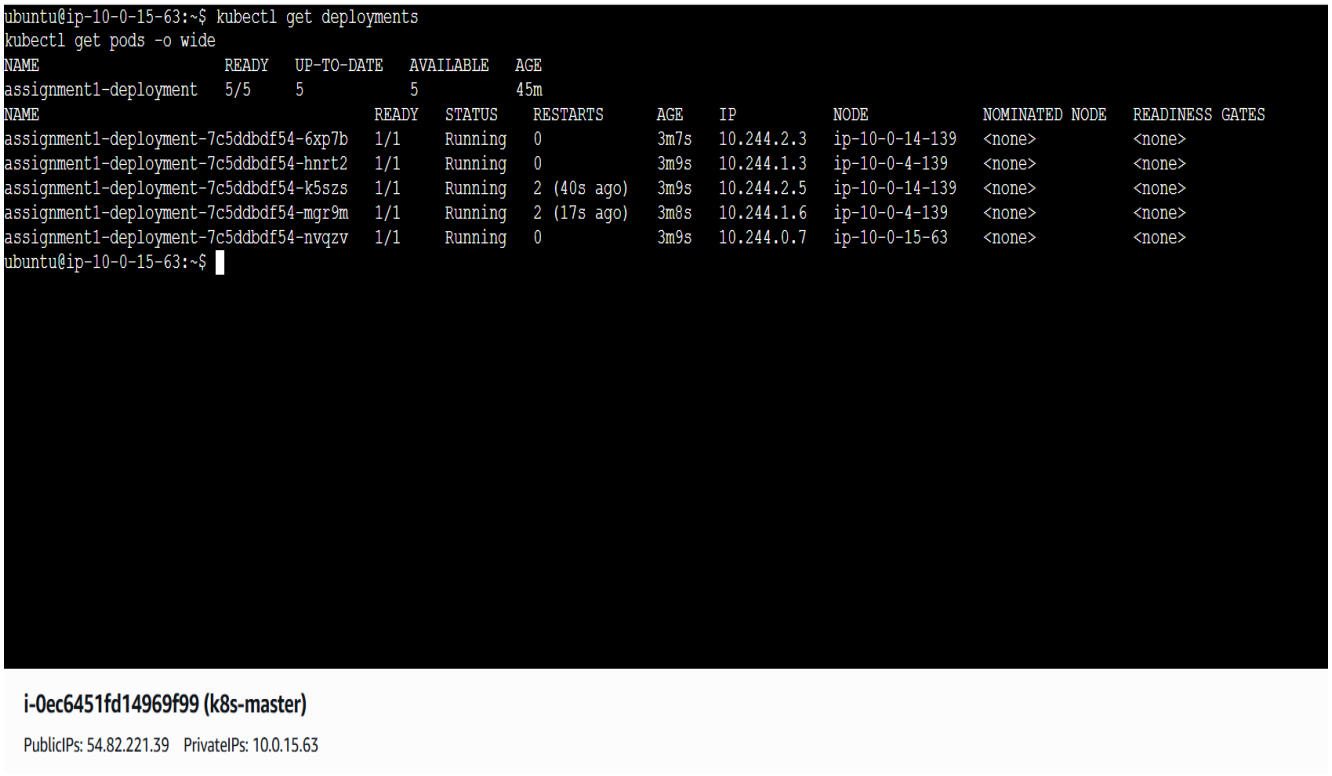
Check pods:

kubectl get pods

Expected:

5 pods running

```
ubuntu@ip-10-0-15-63:~$ kubectl get deployments
NAME                    READY   UP-TO-DATE   AVAILABLE   AGE
assignment1-deployment  5/5     5            5           42m
ubuntu@ip-10-0-15-63:~$ kubectl get pods
NAME                                    READY   STATUS      RESTARTS   AGE
assignment1-deployment-7c5ddbdf54-6xp7b 1/1     Running     0          85s
assignment1-deployment-7c5ddbdf54-hnrt2 1/1     Running     0          87s
assignment1-deployment-7c5ddbdf54-k5szs 0/1     Completed   0          87s
assignment1-deployment-7c5ddbdf54-mgr9m 1/1     Running     0          86s
assignment1-deployment-7c5ddbdf54-nvqzv 1/1     Running     0          87s
ubuntu@ip-10-0-15-63:~$
```



Conclusion

Successfully updated the existing NGINX deployment to 5 replicas.

Kubernetes automatically created new pods and balanced the workload across the cluster.

✓ Used existing deployment

✓ Scaled replicas from 3 → 5

✓ Verified successful scaling