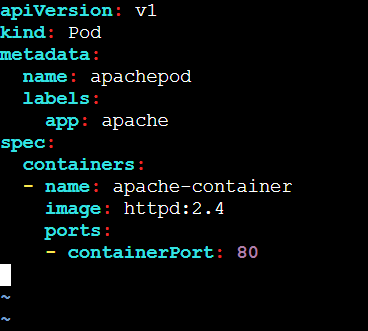
**Kubernetes Tasks-03**

1. **Create a ClusterIP service for an Apache web server pod.**

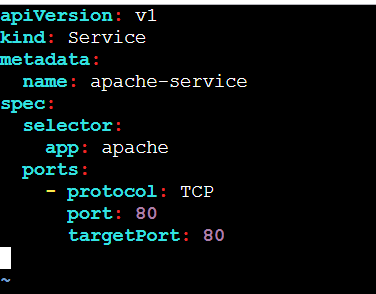
**>>>** vi apache-pod.yaml

|  |
| --- |
| apiVersion: v1  kind: Pod  metadata:  name: apachepod  labels:  app: apache  spec:  containers:  - name: apache-container  image: httpd:2.4  ports:  - containerPort: 80 |

****

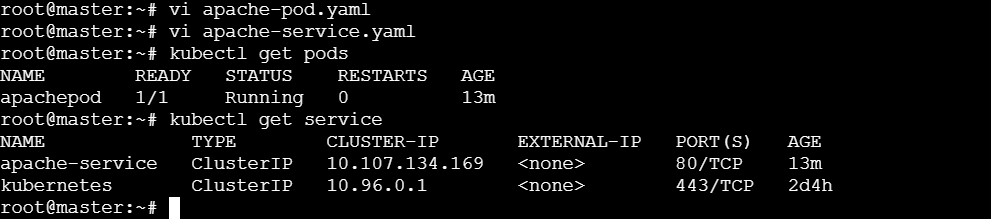
**>>>** vi apache-service.yaml

|  |
| --- |
| apiVersion: v1  kind: Service  metadata:  name: apache-service  spec:  selector:  app: apache  ports:  - protocol: TCP  port: 80  targetPort: 80 |

****

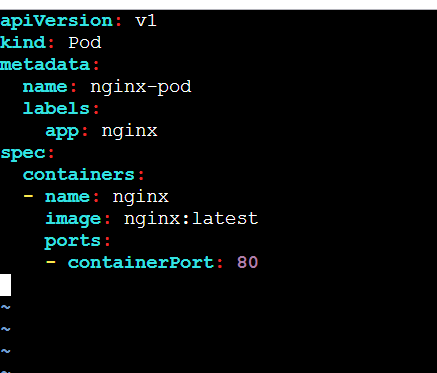
**>>>** kubectl apply -f apache-pod.yaml

**>>>** kubectl apply -f apache-service.yaml

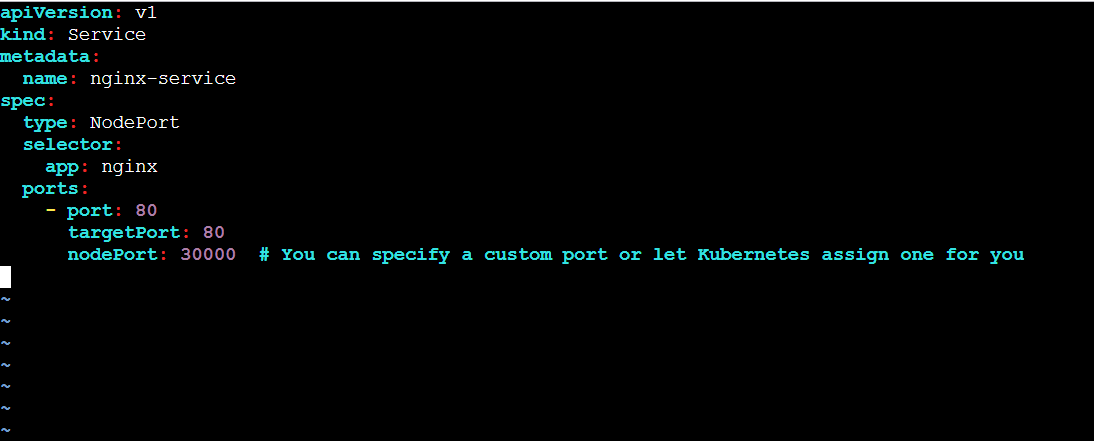


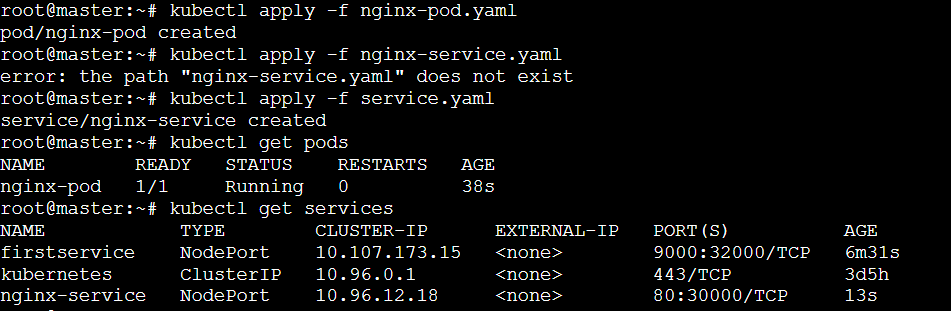
1. **Expose a Nginx pod externally using a NodePort service.**

* **Create a pod:**

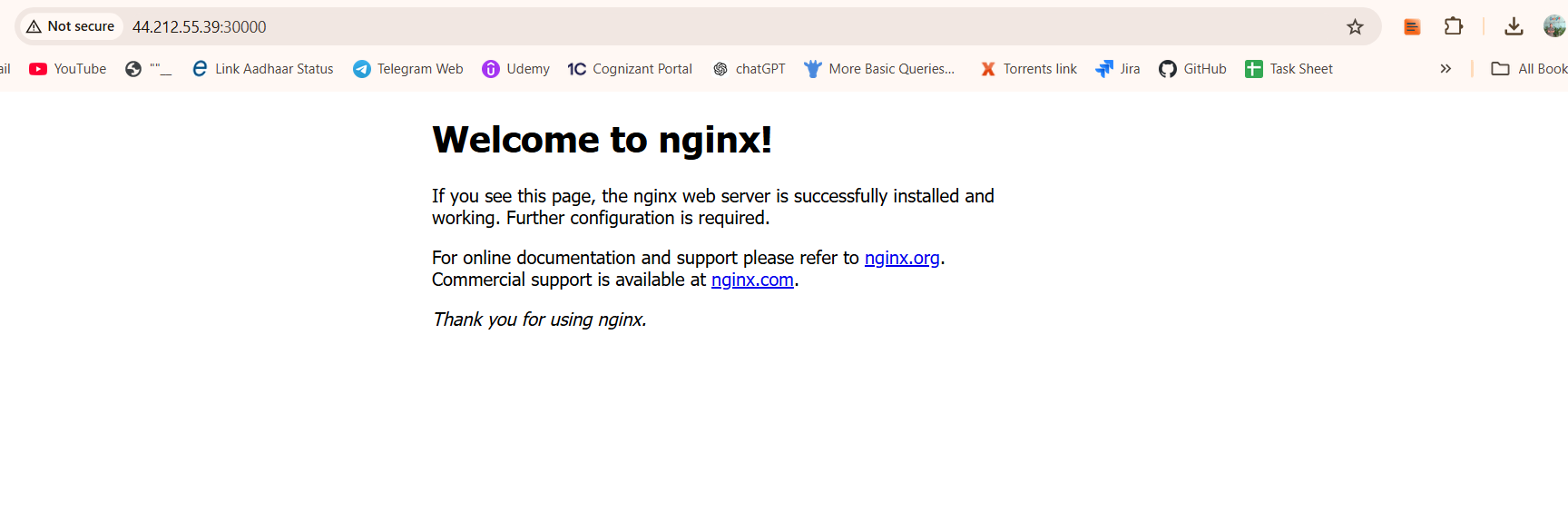
****

* **Create a service:**

****

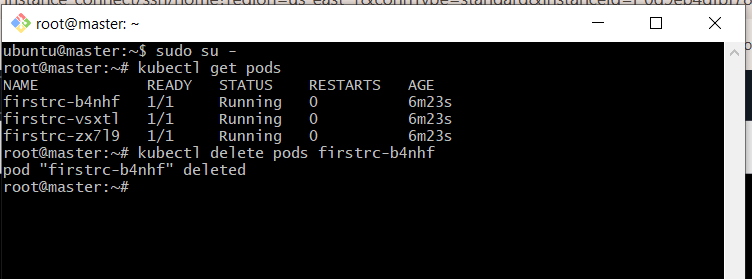
****

* **Using the public IP address of any node(Worker-01) along with the NodePort(3000) to access the Nginx service**

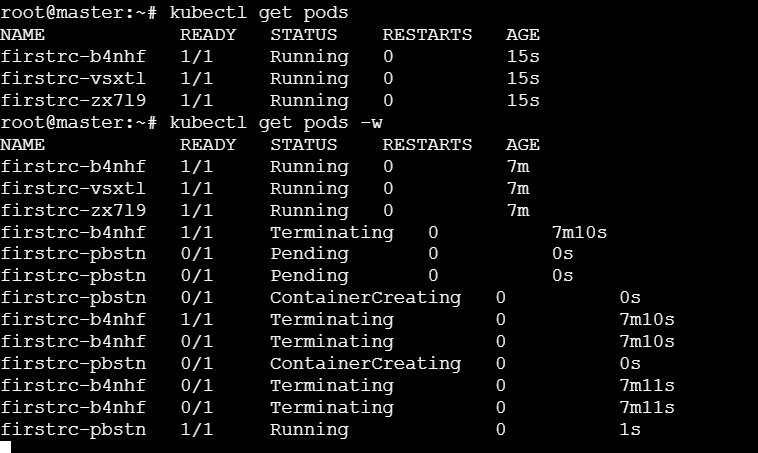
****

1. **Deploy a Replication Controller to maintain 3 replicas of an Nginx pod.**

* **Connected Using another terminal:**
* **Deleted on pod**

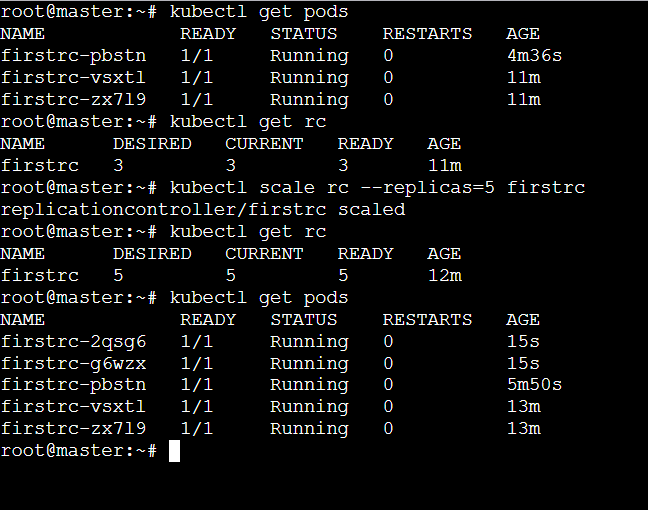
****

**>>> kubectl get pods -w**

****

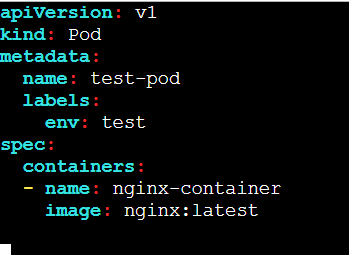
1. **Scale the ReplicationController from 3 replicas to 5 replicas.**

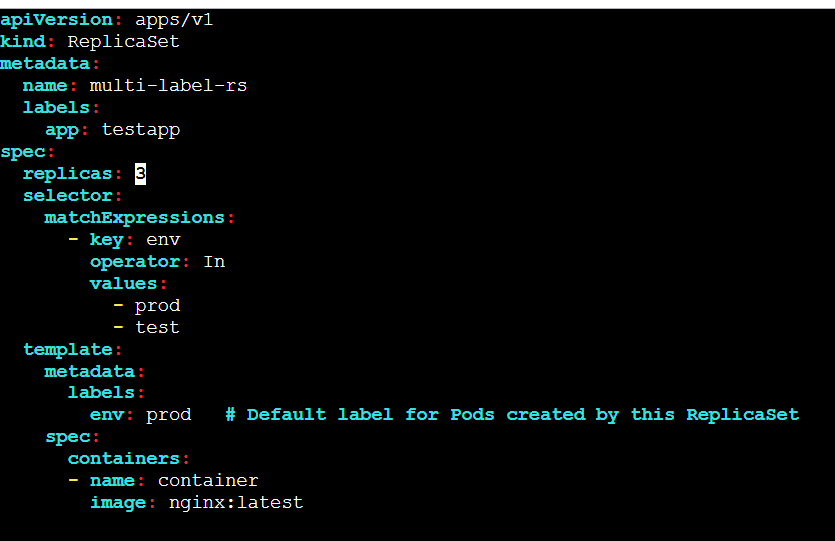
**>>> kubectl scale rc --replicas=5 firstrc**

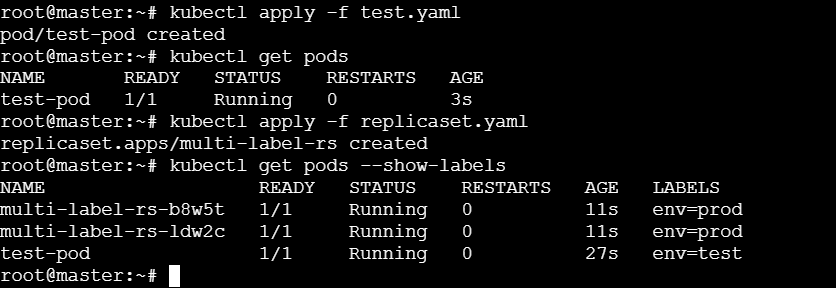
****

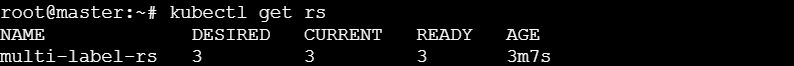
1. **Create a ReplicaSet to manage pods based on multiple labels (prod and test)**

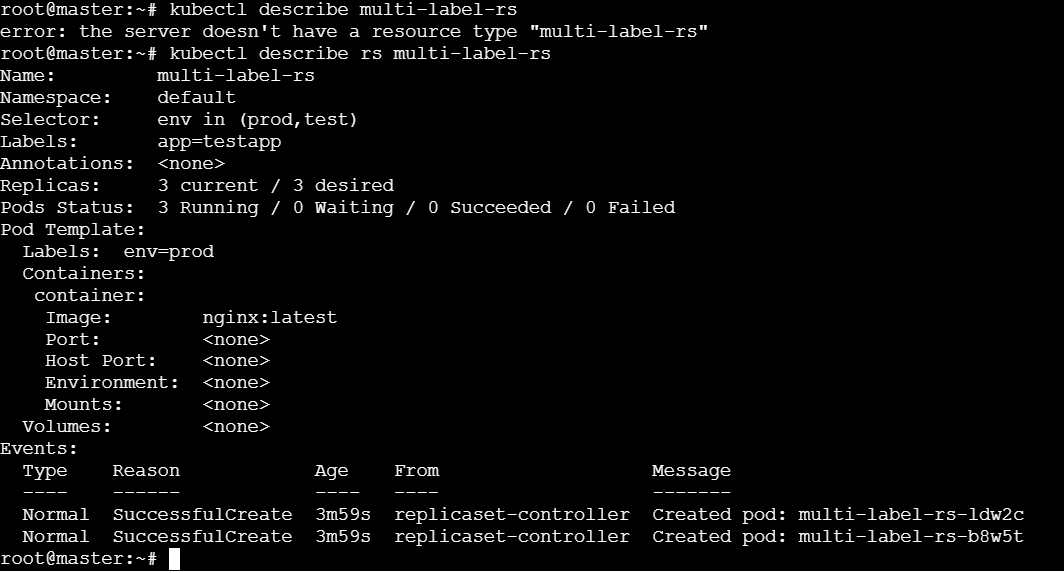
**>>> vi test.yaml**

****

>>> **vi replicaset.yaml**

****

****

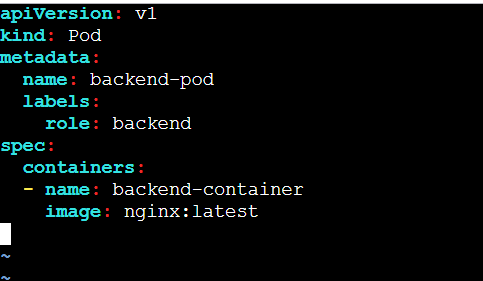
****

1. **Deploy a ReplicaSet that excludes pods with the label backend.**

* **Create a pod with the label as “backend”**

**>>> vi backend.yaml**

**>>> kubectl create -f backend.yaml**

****

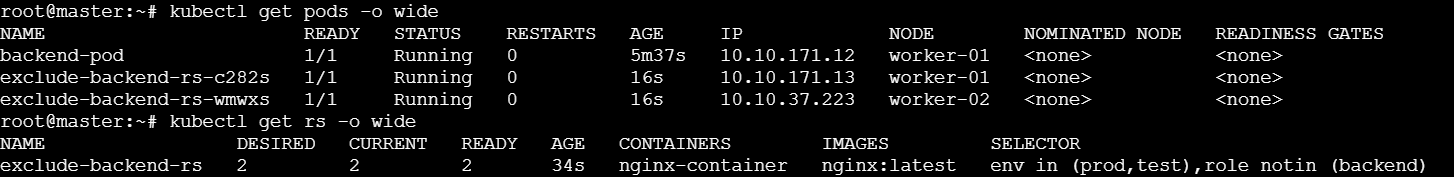
* **Create a replicaset that should map with labels like env: prod and env: test but must ignore label backend:**

**>>>vi replicaset.yaml**

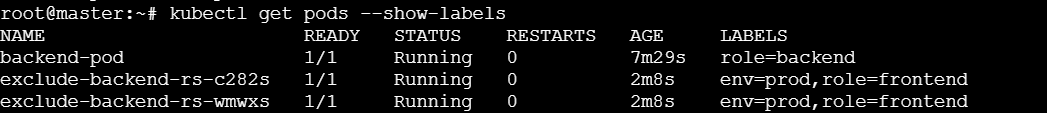
|  |
| --- |
| apiVersion: apps/v1  kind: ReplicaSet  metadata:  name: exclude-backend-rs  labels:  app: testapp  spec:  replicas: 2  selector:  matchExpressions:  - key: role  operator: NotIn  values:  - backend  - key: env  operator: In  values:  - prod  - test  template:  metadata:  labels:  role: frontend # Ensure this Pod does not have the label role: backend  env: prod # Default label for Pods created by this ReplicaSet  spec:  containers:  - name: nginx-container  image: nginx:latest |

**>>> kubectl get pods -o wide**

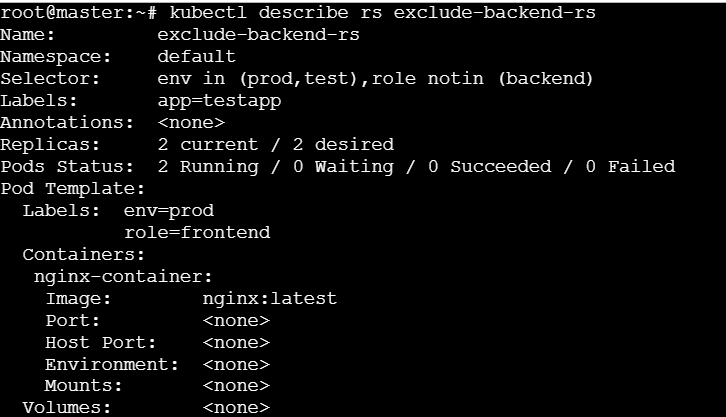
**>>> kubectl get rs -o wide**

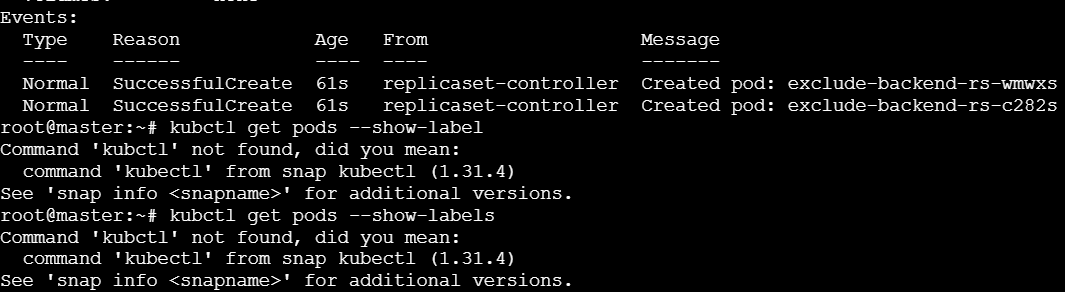
****

**>>> kubectl get pods –show-labels**

****

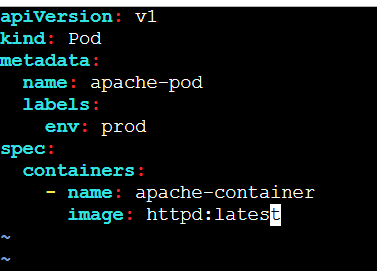
**>>> kubectl describe rs exclude-backend-rs**

****

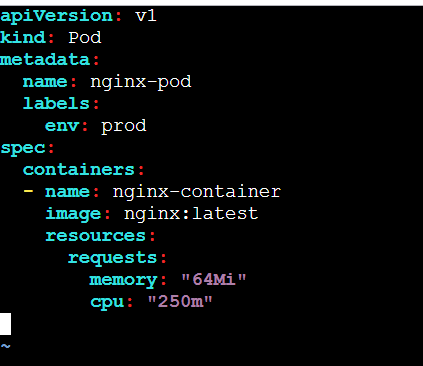
****

**7)Test load balancing across multiple pods using a NodePort service.**

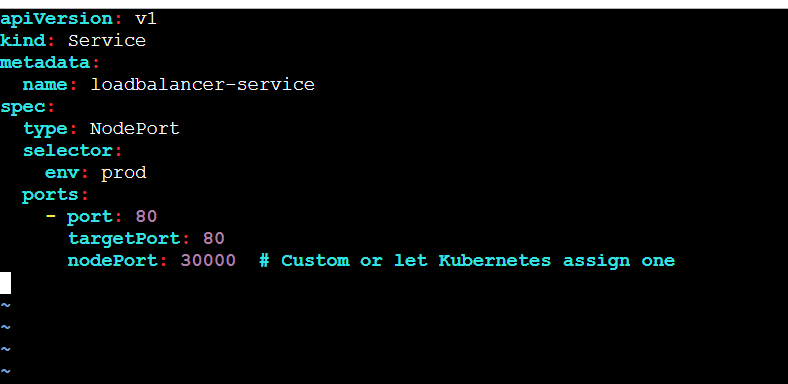
* **For Apache pod**

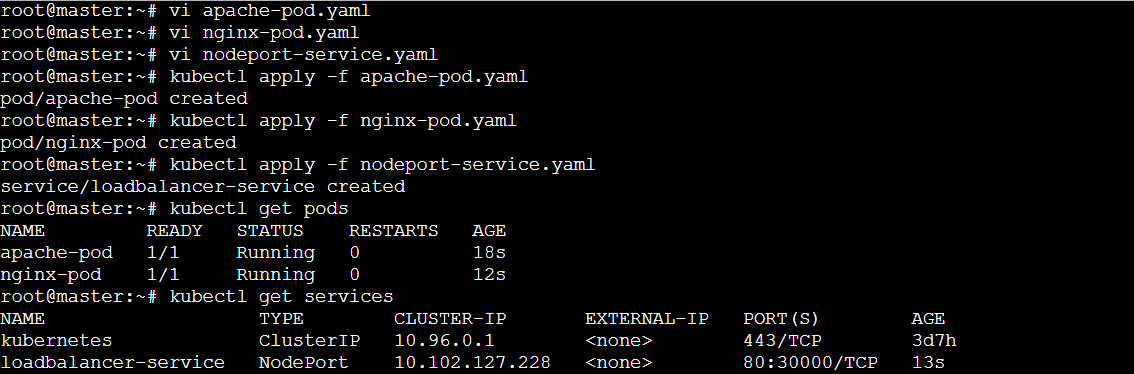
****

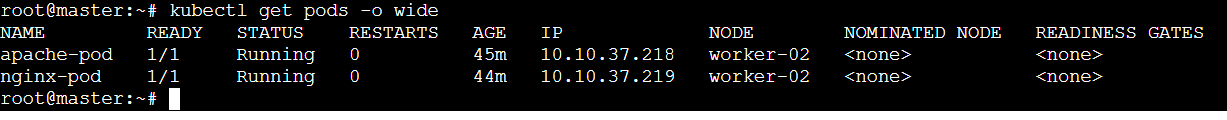
* **For nginx pod**

****

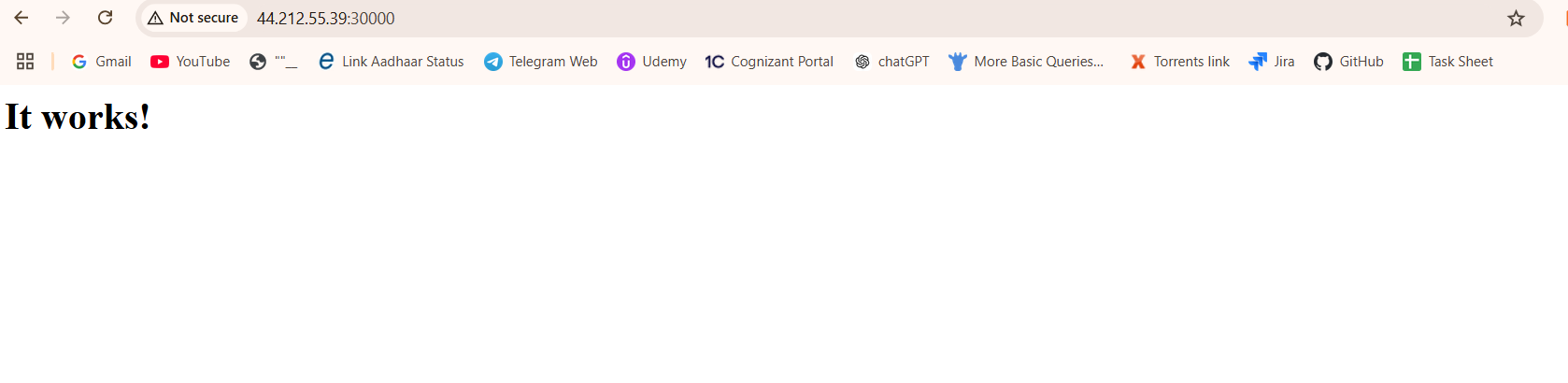
* **Create a service:**

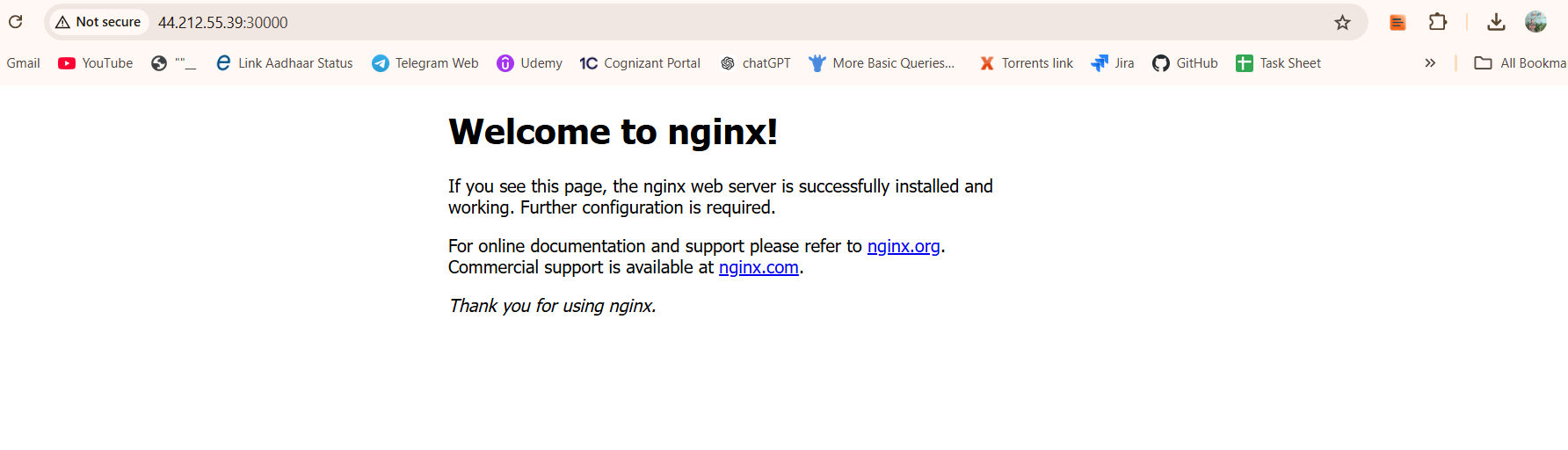
****

****

****

* **Check on the browser:**

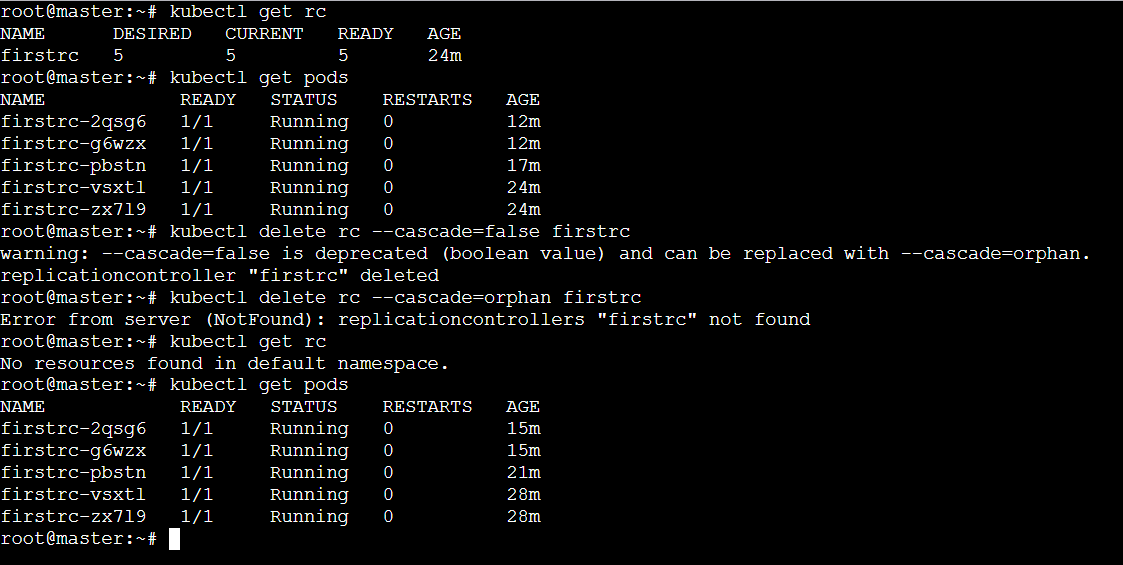
****

****

**8) Delete a ReplicationController without affecting the running pods.**

* **Delete Rc without deleting the pods using the command:**

**>>> kubectl delete rc --cascade=orphan firstrc**

****