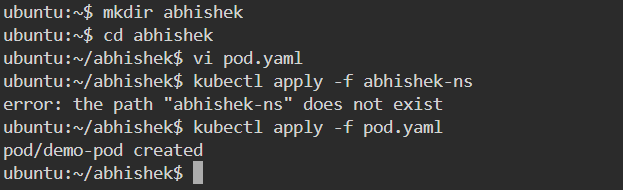
Kubarnetes

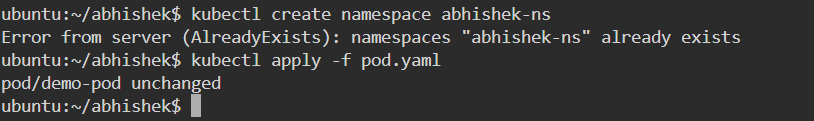
**Pod YAML**

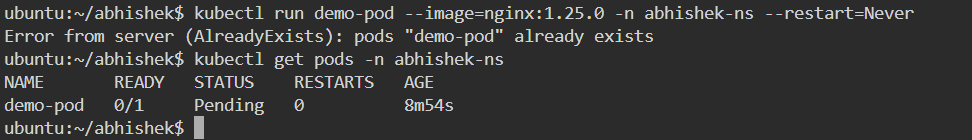


kubectl create namespace abhishek-ns

# Create pod from YAML

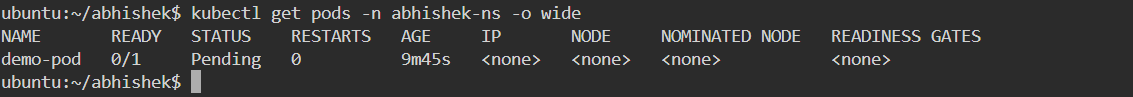
kubectl apply -f pod.yaml





# Alternative: create a pod quickly

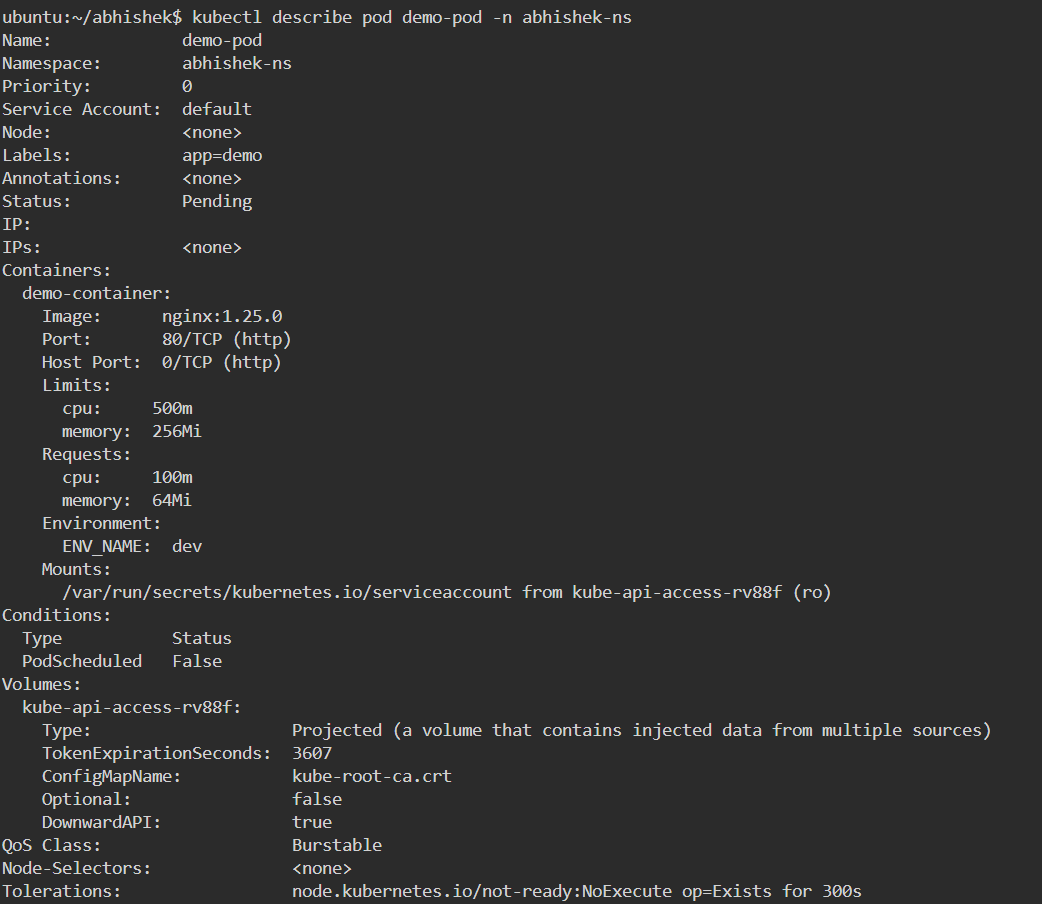
kubectl run demo-pod --image=nginx:1.25.0 -n abhishek-ns --restart=Never



# Get pods

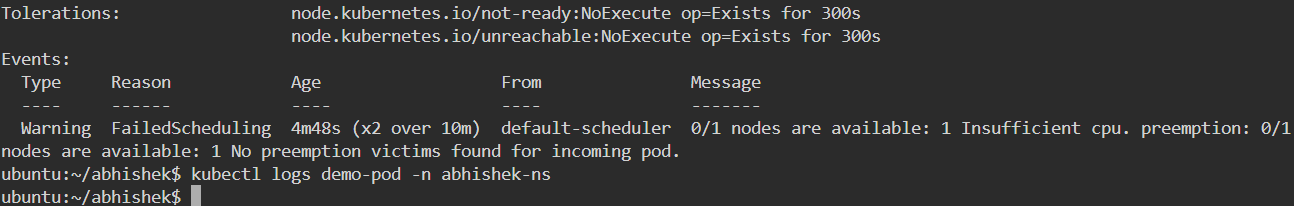
kubectl get pods -n abhishek-ns

kubectl get pods -n abhishek-ns -o wide



# Describe pod (detailed)

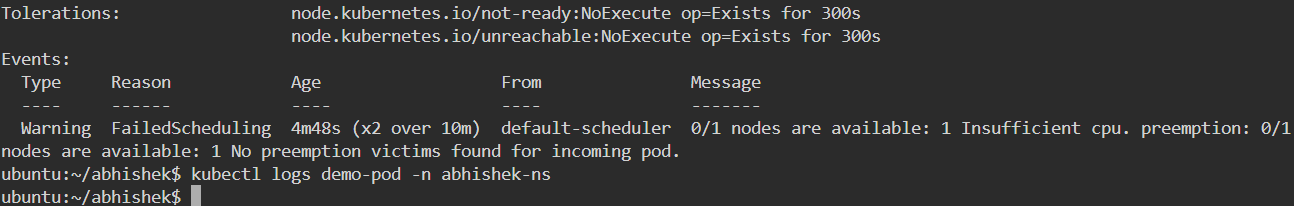
kubectl describe pod demo-pod -n abhishek-ns

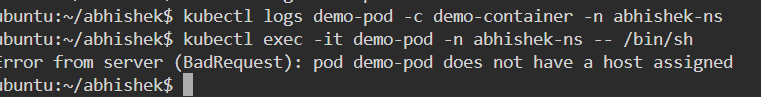


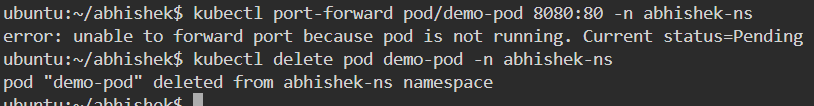
# View logs (container name optional)

kubectl logs demo-pod -n abhishek-ns

kubectl logs demo-pod -c demo-container -n abhishek-ns





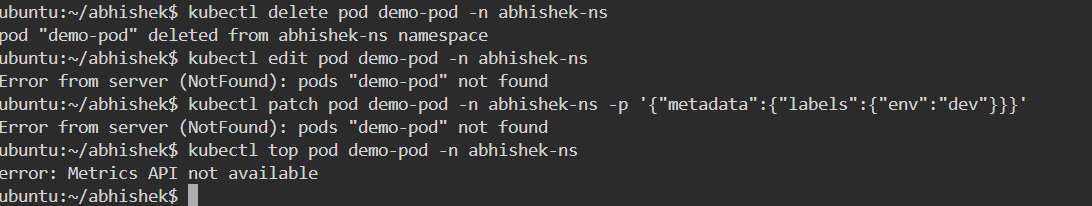


# Exec into pod (interactive shell)

kubectl exec -it demo-pod -n abhishek-ns -- /bin/sh

# Port-forward pod localPort:podPort (useful for testing)

kubectl port-forward pod/demo-pod 8080:80 -n abhishek-ns



# Delete pod

kubectl delete pod demo-pod -n abhishek-ns

# Edit live pod (will restart container)

kubectl edit pod demo-pod -n abhishek-ns

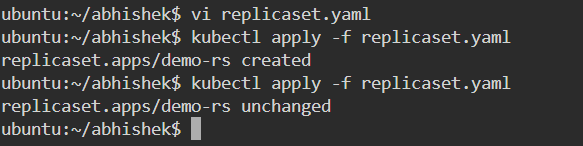
# Patch a pod (labels, annotations) — example: add label

kubectl patch pod demo-pod -n abhishek-ns -p '{"metadata":{"labels":{"env":"dev"}}}'

# Check resource usage (if metrics server installed)

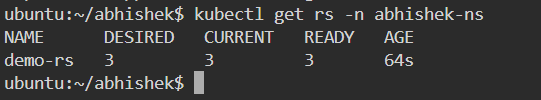
kubectl top pod demo-pod -n abhishek-ns

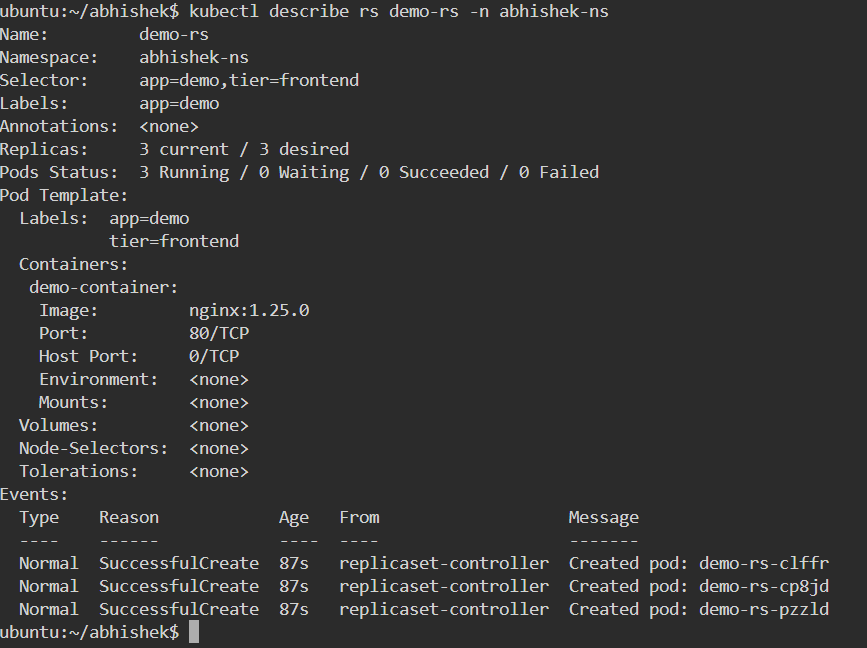
**ReplicaSet YAML + explanation**



**# Create RS**

**kubectl apply -f replicaset.yaml**

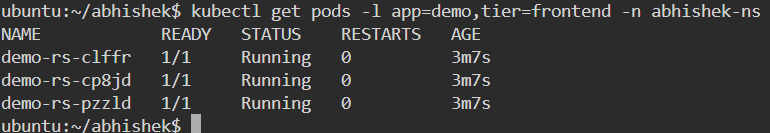




**# Get replica sets**

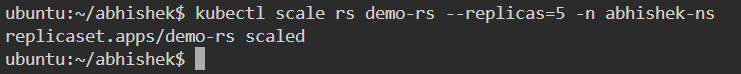
**kubectl get rs -n abhishek-ns**

**kubectl describe rs demo-rs -n abhishek-ns**



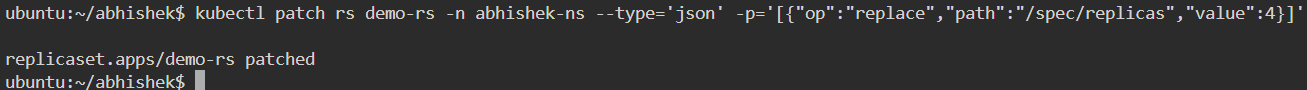
**# See pods created by RS**

**kubectl get pods -l app=demo,tier=frontend -n abhishek-ns**



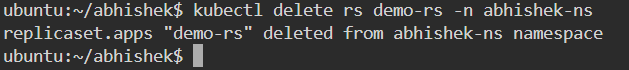
**# Scale RS**

**kubectl scale rs demo-rs --replicas=5 -n abhishek-ns**



**# Patch RS (e.g., change image) — but RS is not ideal for updates; use deployment for rolling updates**

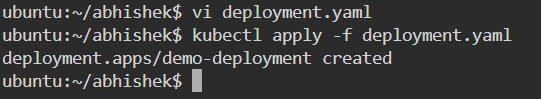
**kubectl patch rs demo-rs -n abhishek-ns --type='json' -p='[{"op":"replace","path":"/spec/replicas","value":4}]'**



**# Delete RS (this also deletes its pods)**

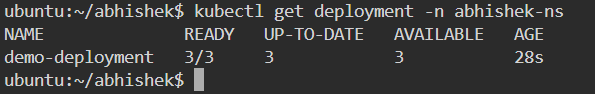
**kubectl delete rs demo-rs -n abhishek-ns**

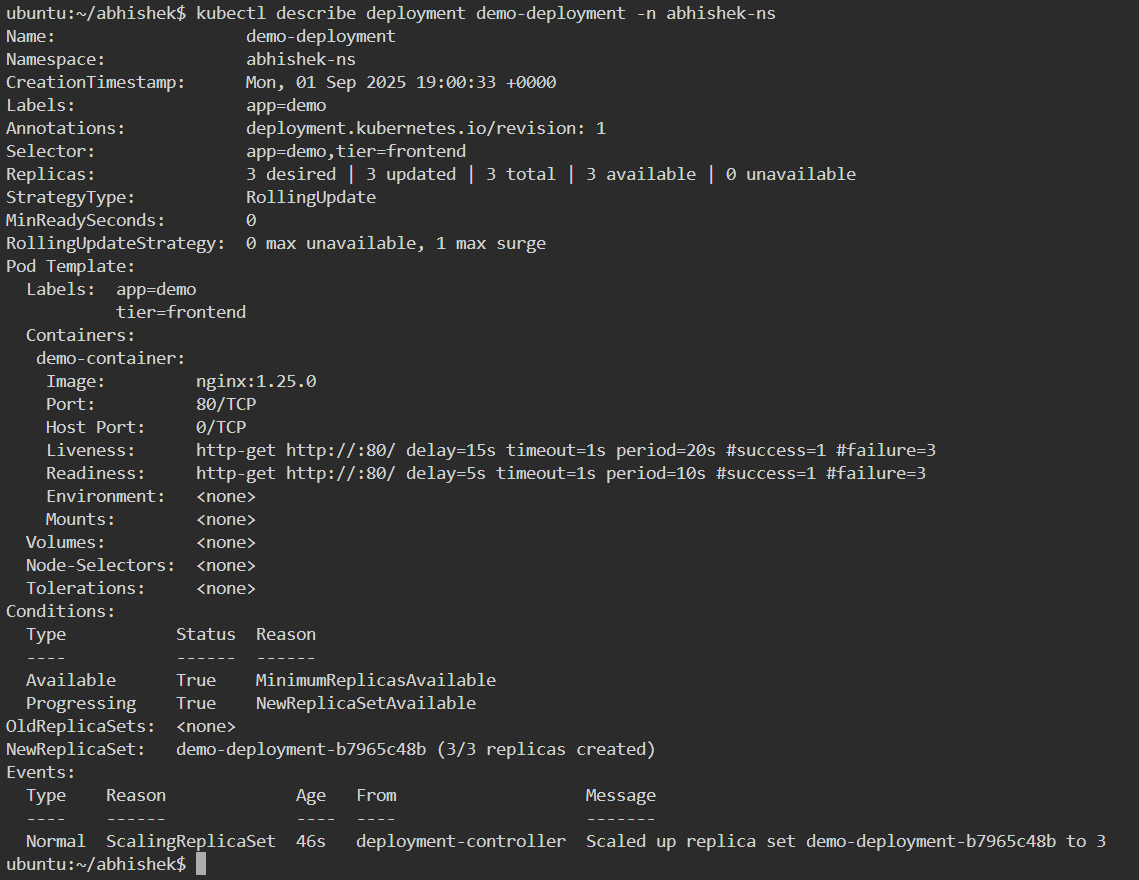
**Deployment YAML + explanation**



**# Create deployment**

**kubectl apply -f deployment.yaml**

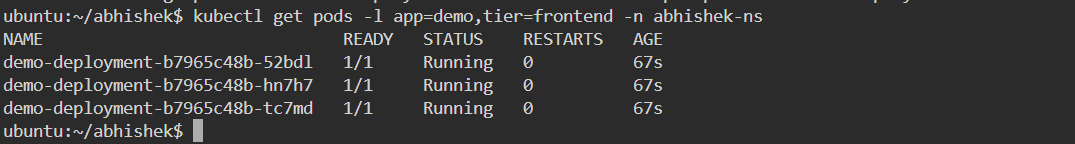




**# Get deployments**

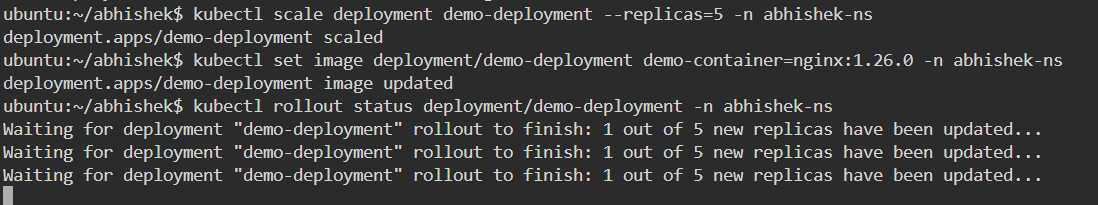
**kubectl get deployment -n abhishek-ns**

**kubectl describe deployment demo-deployment -n abhishek-ns**



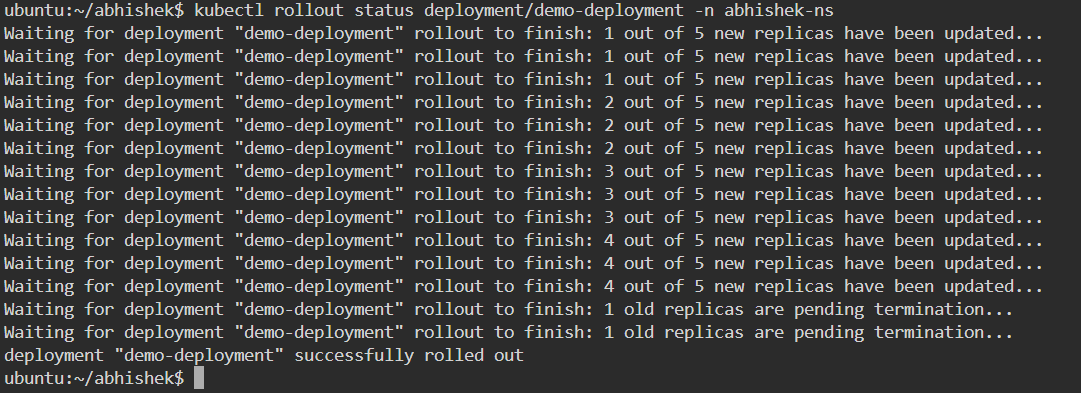
**# Get pods created by deployment**

**kubectl get pods -l app=demo,tier=frontend -n abhishek-ns**



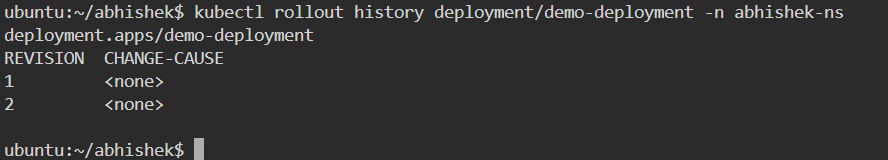
**# Scale the deployment**

**kubectl scale deployment demo-deployment --replicas=5 -n abhishek-ns**



**# Rolling update: change image**

**kubectl set image deployment/demo-deployment demo-container=nginx:1.26.0 -n abhishek-ns**

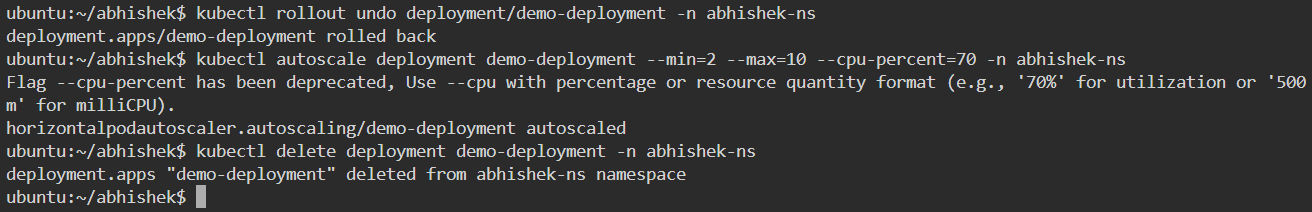


**# Check rollout status**

**kubectl rollout status deployment/demo-deployment -n abhishek-ns**

**# See rollout history**

**kubectl rollout history deployment/demo-deployment -n abhishek-ns**



**# Rollback to previous revision**

**kubectl rollout undo deployment/demo-deployment -n abhishek-ns**

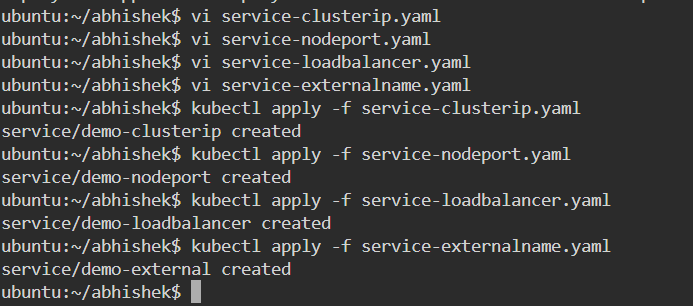
**# Autoscale (HPA) example**

**kubectl autoscale deployment demo-deployment --min=2 --max=10 --cpu-percent=70 -n abhishek-ns**

**# Delete deployment (and pods/replica sets it created)**

**kubectl delete deployment demo-deployment -n abhishek-ns**

**Service YAML files (4 types) with explanations**



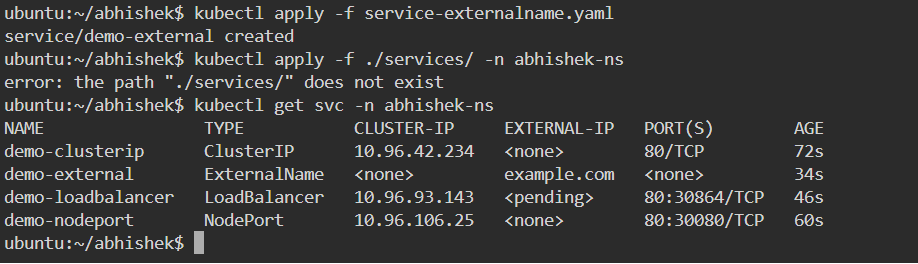
**# Create all services**

**kubectl apply -f service-clusterip.yaml**

**kubectl apply -f service-nodeport.yaml**

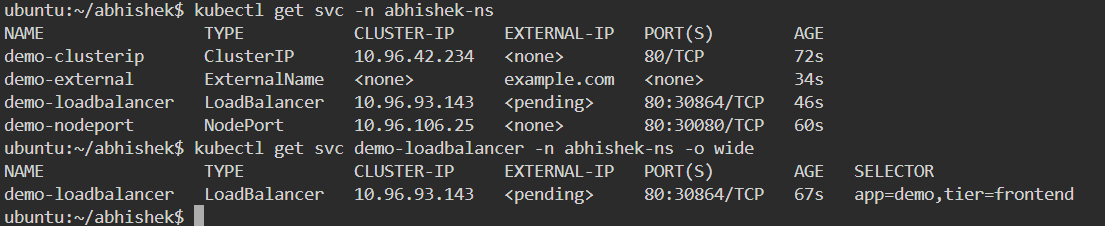
**kubectl apply -f service-loadbalancer.yaml**

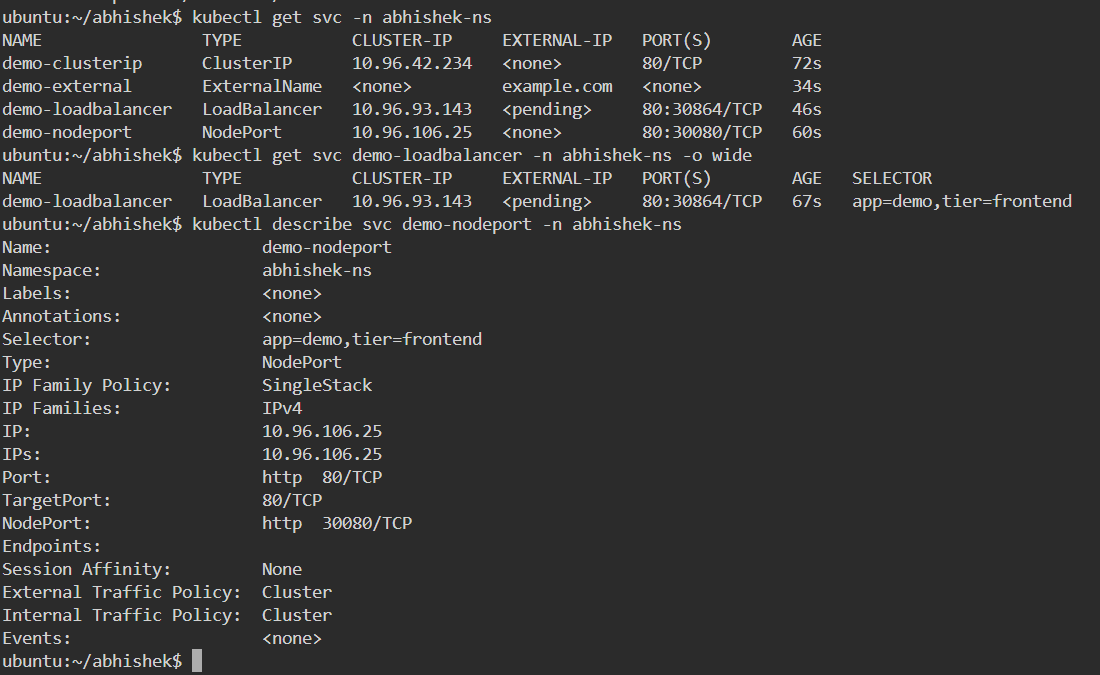
**kubectl apply -f service-externalname.yaml**



**# Or apply a folder**

**kubectl apply -f ./services/ -n abhishek-ns**





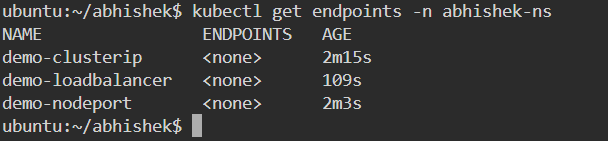
**# Get services**

**kubectl get svc -n abhishek-ns**

**kubectl get svc demo-loadbalancer -n abhishek-ns -o wide**

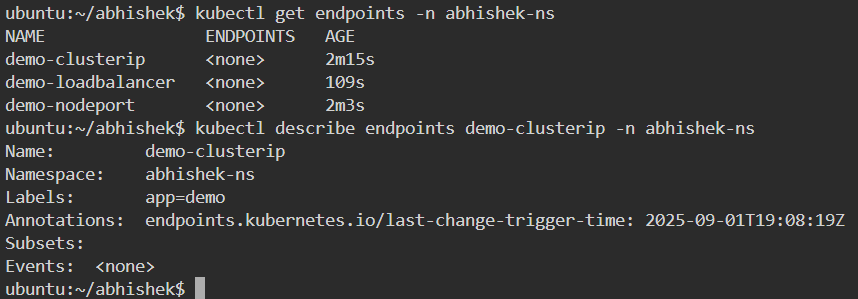
**# Describe service**

**kubectl describe svc demo-nodeport -n abhishek-ns**

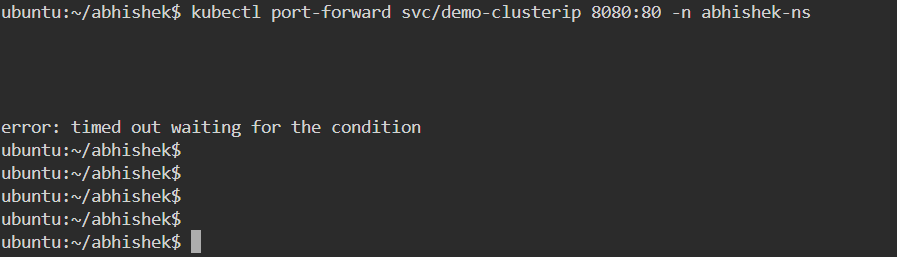


**# Get endpoints (IP:port of backing pods)**

**kubectl get endpoints -n abhishek-ns**

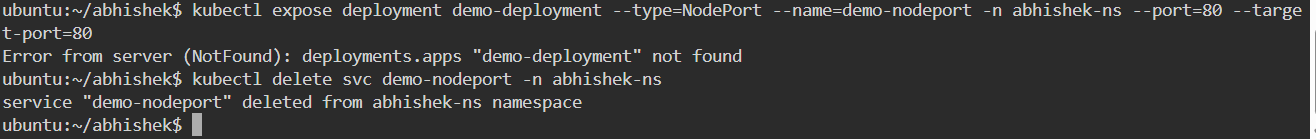


**kubectl describe endpoints demo-clusterip -n abhishek-ns**



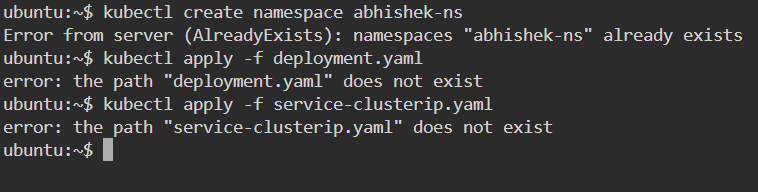
**# Port-forward service to local (service -> pod)**

**kubectl port-forward svc/demo-clusterip 8080:80 -n abhishek-ns**



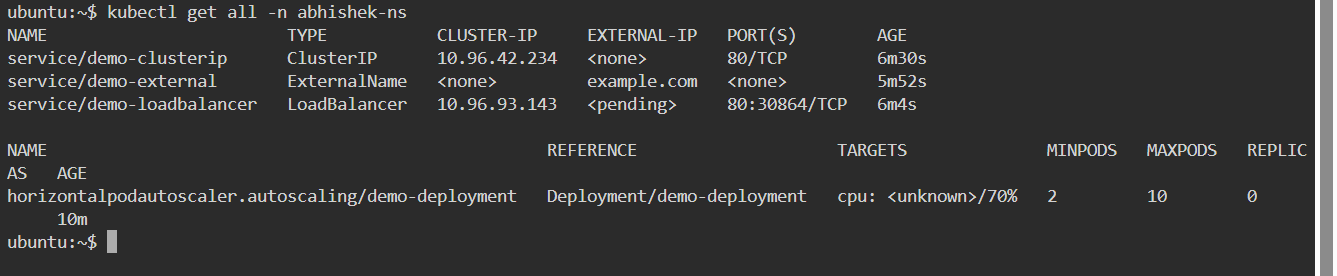
**# Expose a deployment quickly as NodePort (alternative to YAML)**

**kubectl expose deployment demo-deployment --type=NodePort --name=demo-nodeport -n abhishek-ns --port=80 --target-port=80**



**# Delete a service**

**kubectl delete svc demo-nodeport -n abhishek-ns**



**# 1. Make namespace**

**kubectl create namespace abhishek-ns**

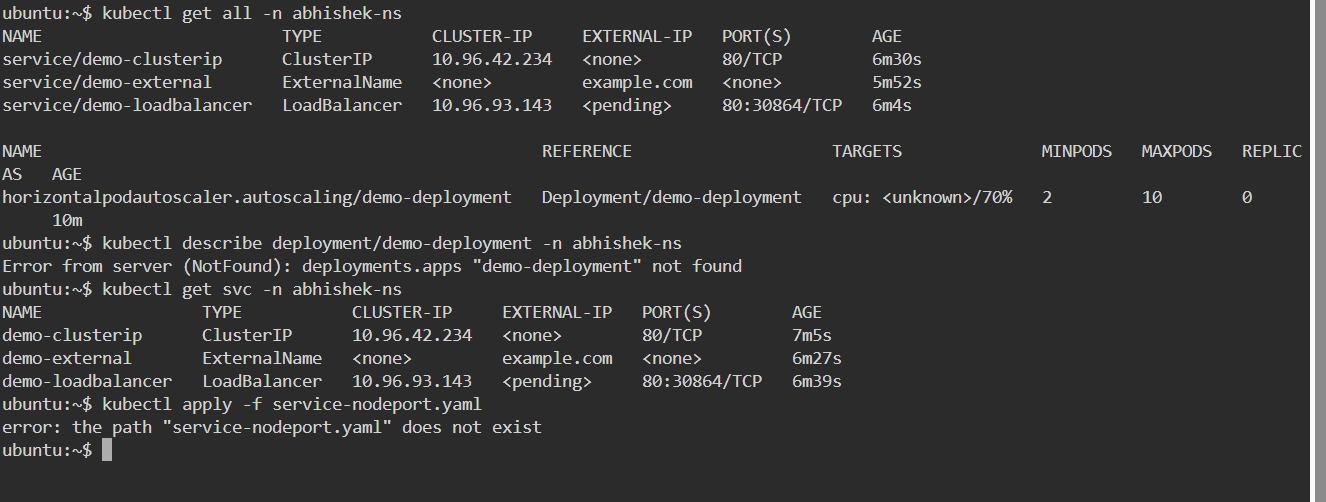
**# 2. Apply deployment and service**

**kubectl apply -f deployment.yaml**

**kubectl apply -f service-clusterip.yaml**

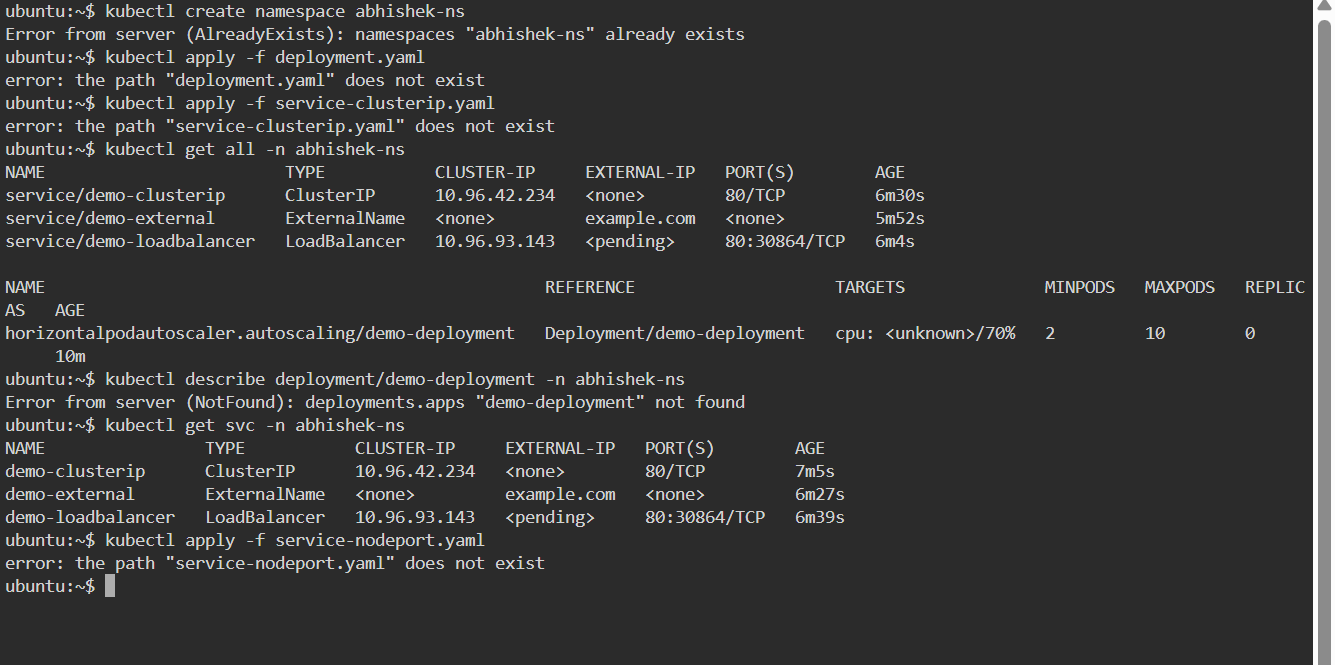
**# 3. Check**

**kubectl get all -n abhishek-ns**



**kubectl describe deployment/demo-deployment -n abhishek-ns**

**kubectl get svc -n abhishek-ns**



**# 4. For external testing using NodePort**

**kubectl apply -f service-nodeport.yaml**

**# Then access NodeIP:30080 (or the assigned nodePort)**