

# **Lab Exercise 7 :- Start and Access**

## **Kubernetes Dashboard**

**Name:- Vansh Bhatt**

**Sap ID:- 500125395**

**Batch:- DevOps B1**

**To:- Hitesh Sharma Sir**

---

### **Objective**

To enable Kubernetes in Docker Desktop, deploy the Kubernetes Dashboard, and access it securely using a web browser on Windows.

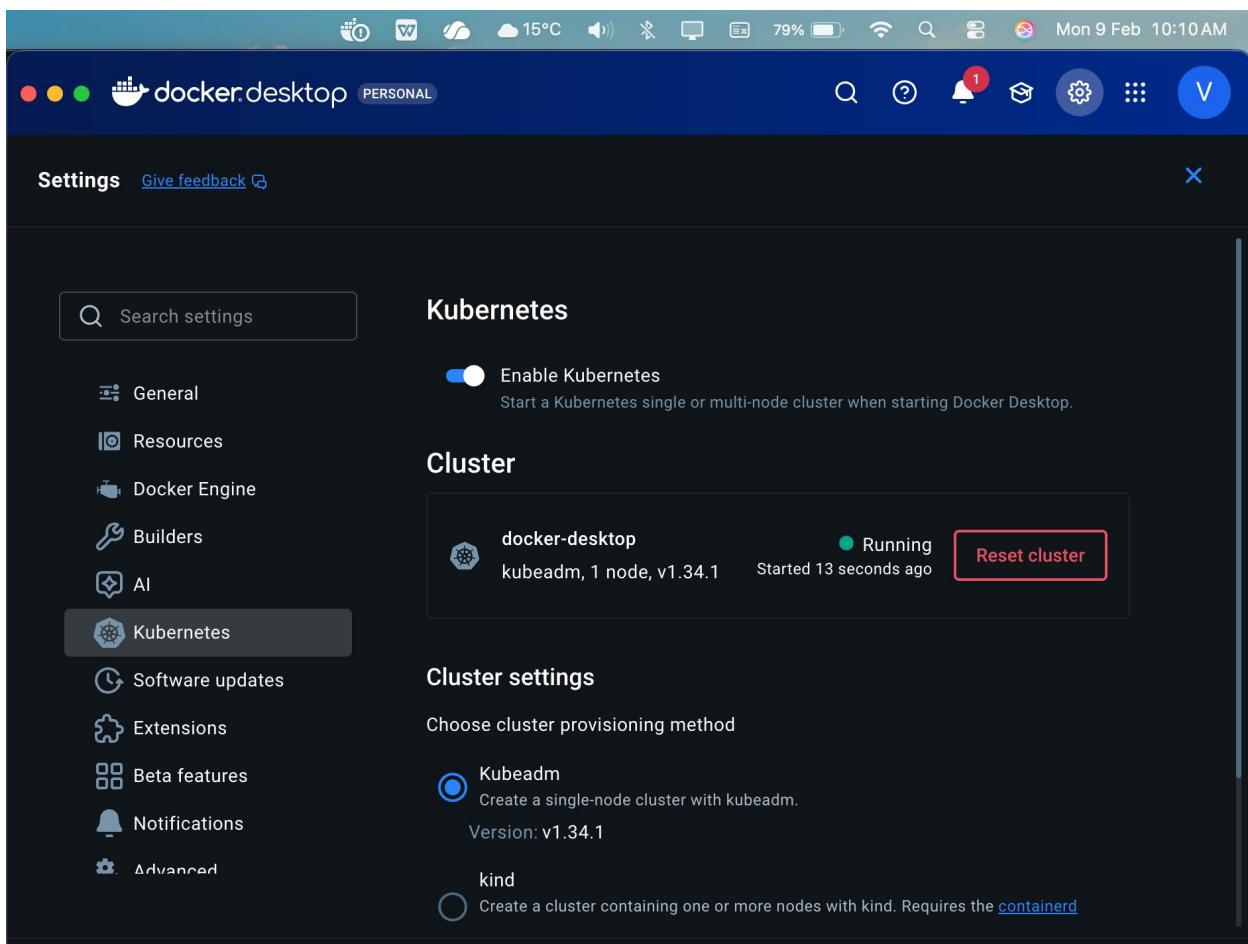
---

### **Prerequisites**

- Windows 10 / 11
  - Docker Desktop installed
  - Docker Desktop Kubernetes enabled
  - Internet connection
  - kubectl (comes bundled with Docker Desktop)
-

## Step 1: Enable Kubernetes in Docker Desktop

1. Open **Docker Desktop**
2. Go to **Settings**
3. Select **Kubernetes**
4. Check **Enable Kubernetes**
5. Click **Apply & Restart**



Wait until Kubernetes status shows **Running** (green).

---

## Step 2: Verify Kubernetes Cluster

Open **PowerShell** or **Command Prompt** and run:

- `kubectl version --client`

```
Last login: Sat Feb  7 13:19:53 on console
vanshssupermac@Vanshs-MacBook-Air ~ % kubectl version --client
Client Version: v1.34.1
Kustomize Version: v5.7.1
vanshssupermac@Vanshs-MacBook-Air ~ %
```

- Check cluster status:
- `kubectl cluster-info`

```
vanshssupermac@Vanshs-MacBook-Air ~ % kubectl cluster-info
Kubernetes control plane is running at https://127.0.0.1:6443
CoreDNS is running at https://127.0.0.1:6443/api/v1/namespaces/kube-system/services/
kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
vanshssupermac@Vanshs-MacBook-Air ~ %
```

Check nodes:

```
kubectl get nodes
```

```
vanshssupermac@Vanshs-MacBook-Air ~ % kubectl get nodes
NAME           STATUS    ROLES          AGE     VERSION
docker-desktop   Ready    control-plane   40d    v1.34.1
vanshssupermac@Vanshs-MacBook-Air ~ %
```

Expected output:

Node status should be **Ready**

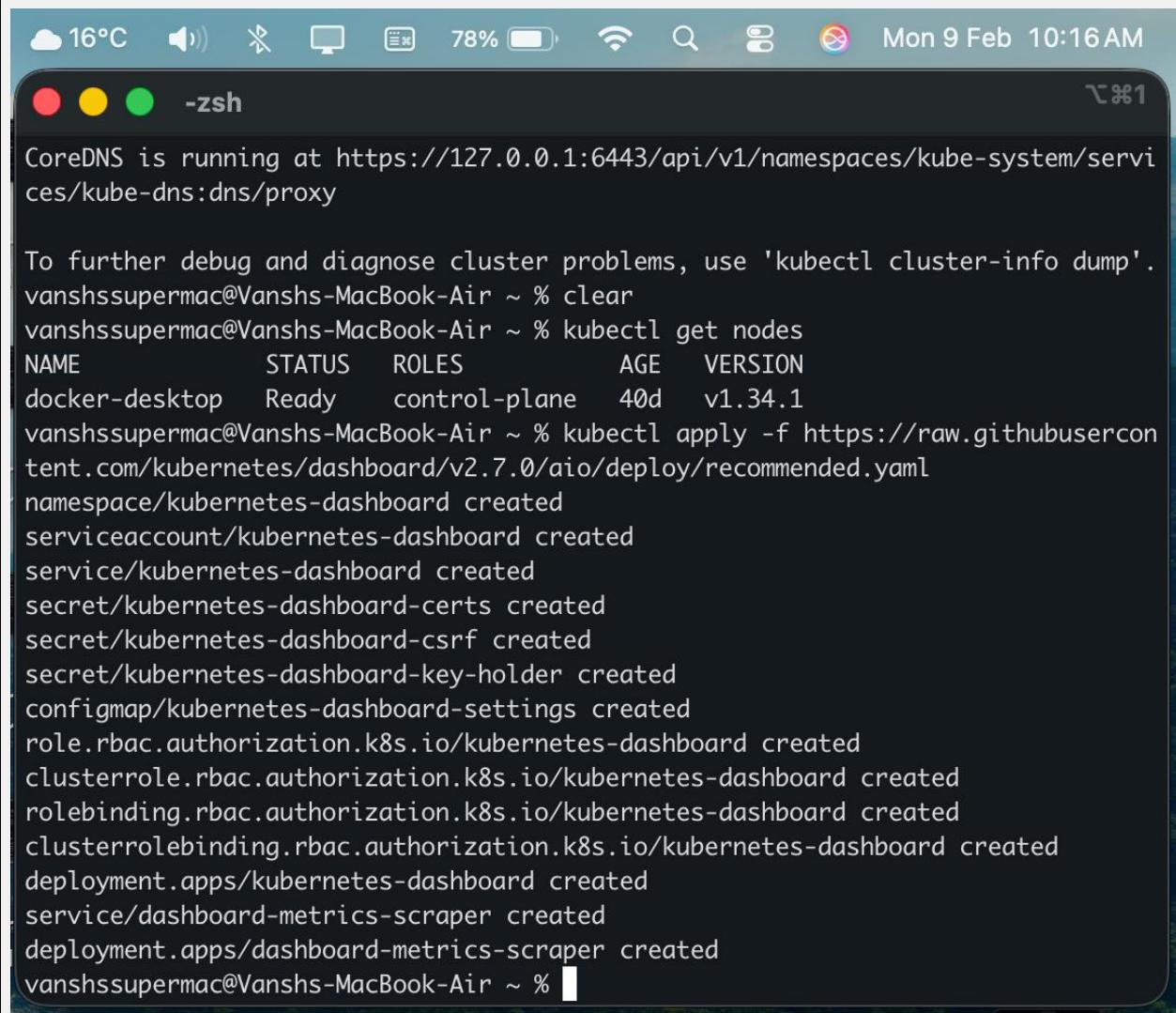
---

### Step 3: Deploy Kubernetes Dashboard

Apply the official Kubernetes Dashboard manifest:

```
kubectl apply -f
```

<https://raw.githubusercontent.com/kubernetes/dashboard/v2.7.0/aio/deploy/recommended.yaml>



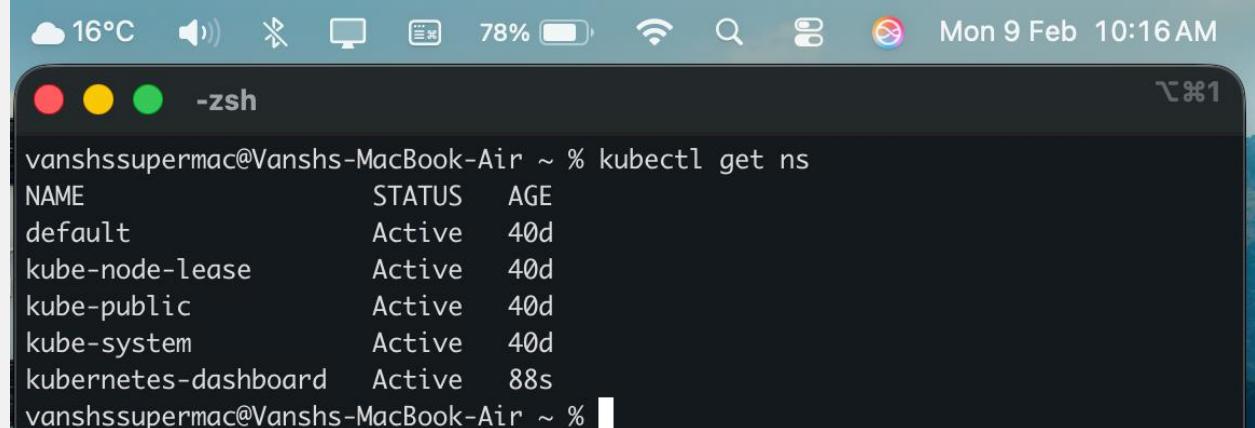
The screenshot shows a macOS terminal window with a dark theme. At the top, there's a status bar with icons for weather (16°C), volume, battery (78%), and network. The title bar says "-zsh". The main terminal area has a blue header with the command "kubectl apply -f https://raw.githubusercontent.com/kubernetes/dashboard/v2.7.0/aio/deploy/recommended.yaml". Below the header, the terminal displays the deployment process. It starts with a message from CoreDNS running at https://127.0.0.1:6443. Then, it lists nodes: "docker-desktop Ready control-plane 40d v1.34.1". Following this, it shows the creation of various resources: "namespace/kubernetes-dashboard created", "serviceaccount/kubernetes-dashboard created", "service/kubernetes-dashboard created", "secret/kubernetes-dashboard-certs created", "secret/kubernetes-dashboard-csrftoken created", "secret/kubernetes-dashboard-key-holder created", "configmap/kubernetes-dashboard-settings created", "role.rbac.authorization.k8s.io/kubernetes-dashboard created", "clusterrole.rbac.authorization.k8s.io/kubernetes-dashboard created", "rolebinding.rbac.authorization.k8s.io/kubernetes-dashboard created", "clusterrolebinding.rbac.authorization.k8s.io/kubernetes-dashboard created", "deployment.apps/kubernetes-dashboard created", "service/dashboard-metrics-scraper created", and "deployment.apps/dashboard-metrics-scraper created". The session ends with "vanshssupermac@Vanshs-MacBook-Air ~ %".

```
CoreDNS is running at https://127.0.0.1:6443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
vanshssupermac@Vanshs-MacBook-Air ~ % clear
vanshssupermac@Vanshs-MacBook-Air ~ % kubectl get nodes
NAME           STATUS   ROLES      AGE    VERSION
docker-desktop   Ready    control-plane   40d   v1.34.1
vanshssupermac@Vanshs-MacBook-Air ~ % kubectl apply -f https://raw.githubusercontent.com/kubernetes/dashboard/v2.7.0/aio/deploy/recommended.yaml
namespace/kubernetes-dashboard created
serviceaccount/kubernetes-dashboard created
service/kubernetes-dashboard created
secret/kubernetes-dashboard-certs created
secret/kubernetes-dashboard-csrftoken created
secret/kubernetes-dashboard-key-holder created
configmap/kubernetes-dashboard-settings created
role.rbac.authorization.k8s.io/kubernetes-dashboard created
clusterrole.rbac.authorization.k8s.io/kubernetes-dashboard created
rolebinding.rbac.authorization.k8s.io/kubernetes-dashboard created
clusterrolebinding.rbac.authorization.k8s.io/kubernetes-dashboard created
deployment.apps/kubernetes-dashboard created
service/dashboard-metrics-scraper created
deployment.apps/dashboard-metrics-scraper created
vanshssupermac@Vanshs-MacBook-Air ~ %
```

Verify namespace creation:

```
kubectl get ns
```



```
vanshssupermac@Vanshs-MacBook-Air ~ % kubectl get ns
NAME        STATUS   AGE
default     Active   40d
kube-node-lease Active   40d
kube-public  Active   40d
kube-system  Active   40d
kubernetes-dashboard Active  88s
vanshssupermac@Vanshs-MacBook-Air ~ %
```

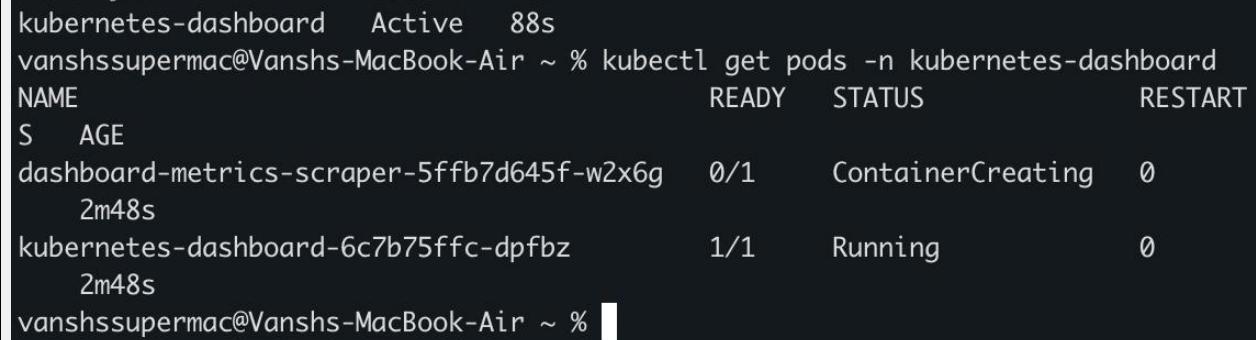
You should see:

```
kubernetes-dashboard
```

#### Step 4: Verify Dashboard Pods

Check dashboard pods:

```
kubectl get pods -n kubernetes-dashboard
```



```
kubernetes-dashboard  Active   88s
vanshssupermac@Vanshs-MacBook-Air ~ % kubectl get pods -n kubernetes-dashboard
NAME                           READY   STATUS            RESTARTS
S   AGE
dashboard-metrics-scraper-5ffb7d645f-w2x6g   0/1    ContainerCreating   0
2m48s
kubernetes-dashboard-6c7b75ffc-dpfbz          1/1    Running           0
2m48s
vanshssupermac@Vanshs-MacBook-Air ~ %
```

Expected status:

Running

---

### Step 5: Create Admin User for Dashboard Access

Create a service account:

```
kubectl create serviceaccount dashboard-admin -n kubernetes-dashboard
```

```
vanshssupermac@Vanshs-MacBook-Air ~ % kubectl create serviceaccount dashboard-admin -n kubernetes-dashboard
serviceaccount/dashboard-admin created
vanshssupermac@Vanshs-MacBook-Air ~ %
```

Create cluster role binding:

```
kubectl create clusterrolebinding dashboard-admin-binding --clusterrole=cluster-admin --serviceaccount=kubernetes-dashboard:dashboard-admin
```

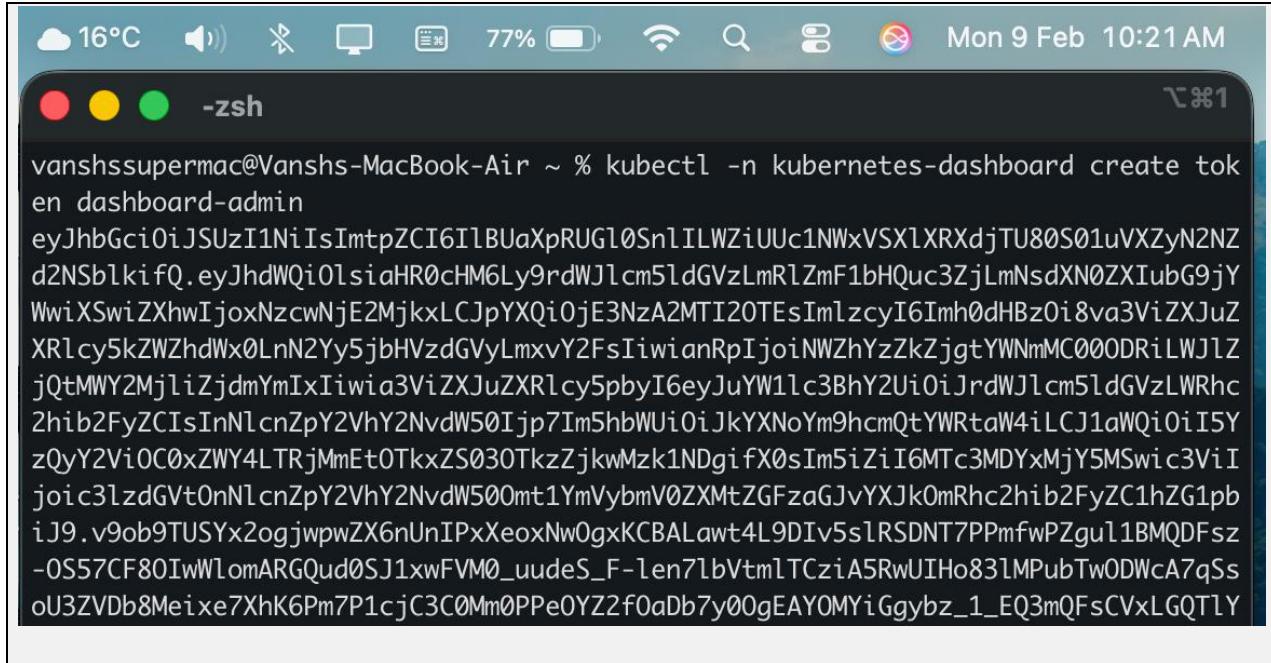
```
16°C Mon 9 Feb 10:20AM
-zsh
vanshssupermac@Vanshs-MacBook-Air ~ % kubectl create clusterrolebinding dashboard-admin-binding --clusterrole=cluster-admin --serviceaccount=kubernetes-dashboard:dashboard-admin
clusterrolebinding.rbac.authorization.k8s.io/dashboard-admin-binding created
vanshssupermac@Vanshs-MacBook-Air ~ %
```

---

### Step 6: Generate Dashboard Login Token

Run the following command to get the token:

```
kubectl -n kubernetes-dashboard create token dashboard-admin
```



```
vanshssupermac@Vanshs-MacBook-Air ~ % kubectl -n kubernetes-dashboard create token dashboard-admin
eyJhbGciOiJSUzI1NiIsImtpZCI6IlBUaXpRUGl0SnLILWZiUUc1NWxVSXlRXdjTU80S01uVXZyN2NZd2NSblkifQ.eyJhdWQiolsiaHR0cHM6Ly9rdWJlc5ldGVzLmRlZmF1bHQuc3ZjLmNsdxN0ZXIubG9jYWwiXSwiZXhwIjoxNzcwNjE2MjkxLCJpYXQiOjE3NzA2MTI20TEsImlzcyI6Imh0dHBzOi8va3ViZXJuZXlcy5kZWZhWx0LnN2Yy5jbHVzdGVyLmxvY2FsIiwianRpIjoiNWZhYzZkZjgtYWNmMC000DRiLWJlZjQtMWY2MjliZjdmYmIxIiwiq3ViZXJuZXlcy5pbvI6eyJuYW1lc3BhY2Ui0iJrdWJlc5ldGVzLWRhc2hib2FyZCIsInNlcnZpY2VhY2NvdW50Ijp7Im5hbWUi0iJkYXNoYm9hcmQtYWRtaW4iLCJ1aWQi0iI5YzQyY2ViOC0xZWY4LTrjMmEtOTkxZS030TkzZjkwMzk1NDgifX0sIm5iZiI6MTc3MDYxMjY5MSwiC3ViIjoic3lzdGVtOnNlcnZpY2VhY2NvdW50mt1YmVybmV0ZXMtZGFzaGJvYXJkOmRh2hib2FyZC1hZG1pbij9.v9ob9TUSYx2ogjwpwZX6nUnIPxXeo9Nw0gxKCBALawt4L9DIv5s1RSDNT7PPmfwPZgul1BMQDFsz-OS57CF80IwWlomARGQud0SJ1xwFVM0_uudeS_F-1en7lbVtmLTCziA5RwUIHo83lMPubTw0DWcA7qSsoU3ZVDb8Meixe7XhK6Pm7P1cjC3C0Mm0PPe0YZ2f0aDb7y00gEAYOMYiGgybz_1_EQ3mQFsCVxLGQTLY
```

Copy the generated token (you will paste it in the browser later).

---

### Step 7: Start Kubernetes Dashboard

Run the proxy command:

```
kubectl proxy
```

```
vanshssupermac@Vanshs-MacBook-Air ~ % kubectl proxy
Starting to serve on 127.0.0.1:8001
```

---

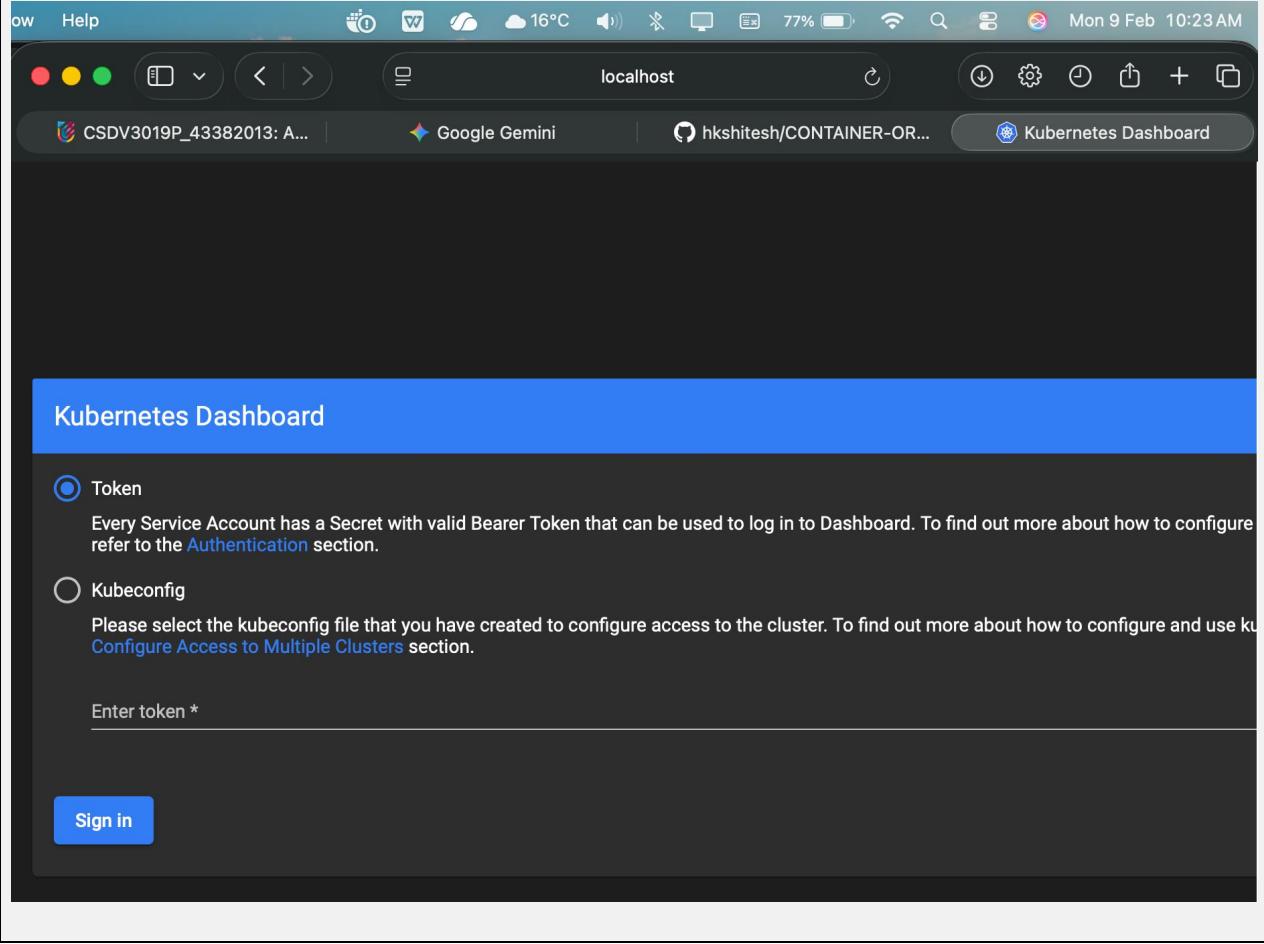
Keep this terminal **running**.

---

### Step 8: Access Kubernetes Dashboard in Browser

Open a web browser and paste the following URL:

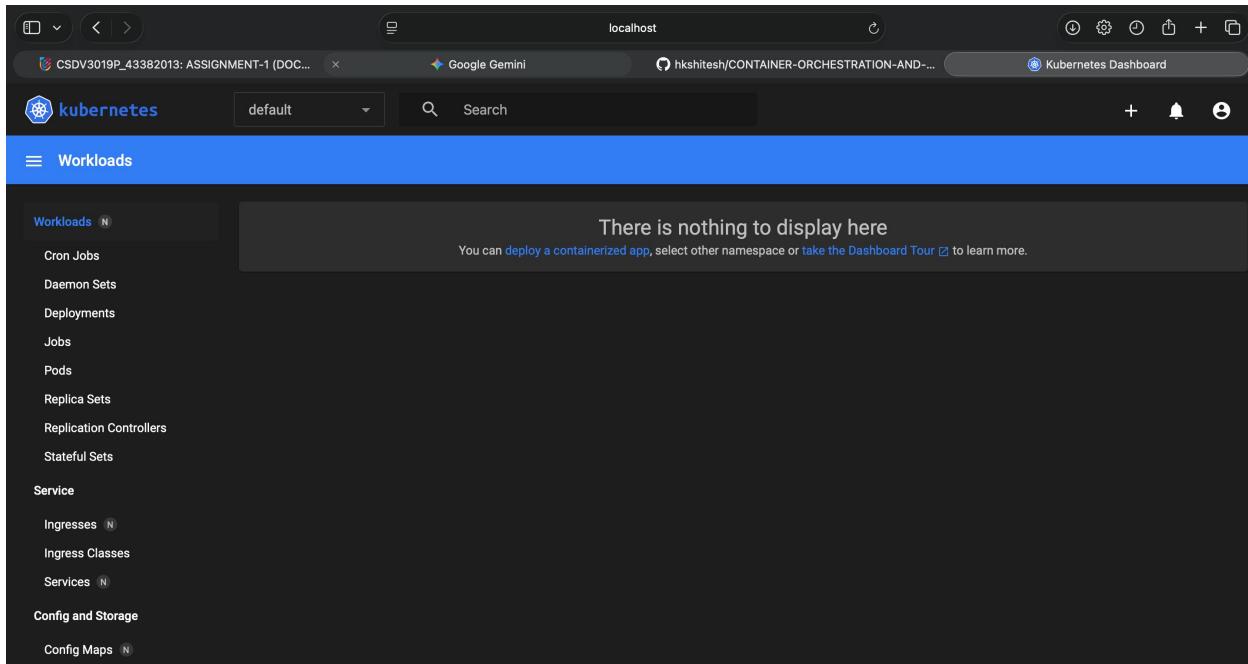
<http://localhost:8001/api/v1/namespaces/kubernetes-dashboard/services/https:kubernetes-dashboard:/proxy/>



## Step 9: Login to Dashboard

1. Select **Token** authentication
2. Paste the token generated earlier
3. Click **Sign In**

You should now see the **Kubernetes Dashboard UI**.



---

## Step 10: Explore Dashboard

You can now view:

- Nodes
- Pods
- Deployments
- Services
- Namespaces
- ConfigMaps and Secrets

# Thank You