TASK 5: Build a Kubernetes Cluster Locally with Minikube

Objective:

- Deploy and manage an app using Kubernetes on a local/ec2 instance Minikube cluster.

Tools Needed

- Minikube
- Kubectl
- Docker
- Ec2/locally

Step-by-Step Process

Launch the server first (or) we can use the locally also

Step 1: Install Tools:

For minikube install purpose required **2cpu and 2gib ram**

----------docker----------

- sudo dnf install docker -y -----Install Docker first
- sudo systemctl start docker.service ---to start the docker service
- sudo systemctl enable docker.service --- to enable docker
- sudo usermod -a -G docker ec2-user --- to add the user ec2-user to the docker group
- newgrp docker --to run new group make changes immediatly

login to ec2-user (su - ec2-user)

----------minikube----------

- curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64
- sudo install minikube-linux-amd64 /usr/local/bin/minikube
- to start minikube ---minikube start
- to check status ----minikube status

----------kubectl----------

- curl -LO https://storage.googleapis.com/kubernetes-release/release/$(curl -s https://storage.googleapis.com/kubernetes-release/release/stable.txt)/bin/linux/amd64/kubectl
- chmod +x ./kubectl -----Make the kubectl binary executable
- sudo mv ./kubectl /usr/local/bin/kubectl

Step 2: Create a deployment.yml and service.yml for sample App

- **deployment.yml**

```
# deployment.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: myapp-deployment
spec:
  replicas: 2
  selector:
    matchLabels:
      app: myapp
  template:
    metadata:
      labels:
        app: myapp
    spec:
      containers:
      - name: myapp
        image: nginx
        ports:
        - containerPort: 80
```

- **service.yml**

```
# service.yaml
apiVersion: v1
kind: Service
metadata:
  name: myapp-service
spec:
  type: NodePort
```

```
  selector:
    app: myapp
  ports:
    - port: 80
      targetPort: 80
      nodePort: 30007
```

 # Then execute this cmds:

- kubectl apply -f .
- kubectl get pods

```
[ec2-user@ip-172-31-19-95 ~]$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
myapp-deployment-58dbdb69b8-9vjhv   1/1     Running   0          18s
myapp-deployment-58dbdb69b8-gnwdf   1/1     Running   0          18s
```

- kubectl get nodes

```
[ec2-user@ip-172-31-19-95 ~]$ kubectl get nodes
NAME       STATUS   ROLES           AGE     VERSION
minikube   Ready    control-plane   5m54s   v1.32.0
```

- kubectl get svc # In ec2 the svc cannot show the external ip so it shows like, so we have to access the application from powershell , we choose for port forwarder.

```
css                                                                    Copy code

myapp-service   NodePort   10.96.22.143   <none>   80:30007/TCP   5m
```

- nohup kubectl port-forward svc/myapp-service 8080:80 > port-forward.log 2>&1 & **# execute this cmd in ec2**

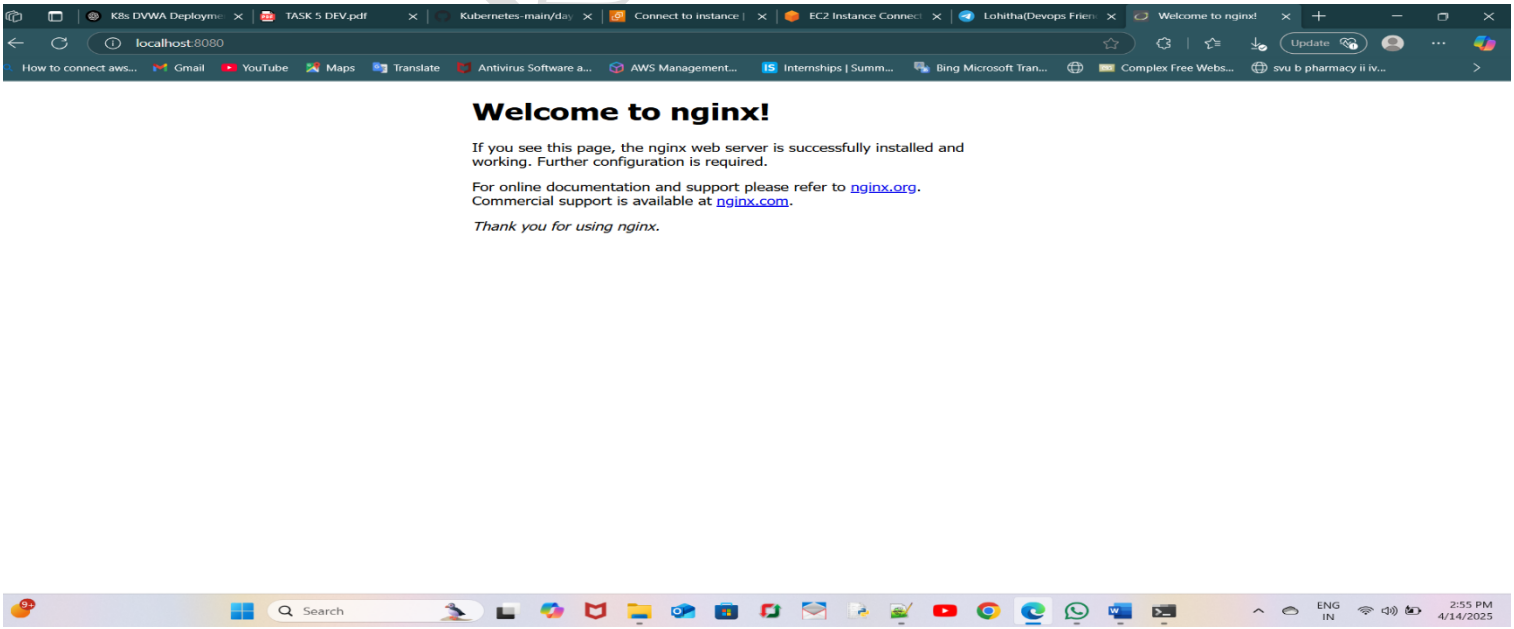
- ssh -i "your-key.pem" -L 8080:localhost:8080 ec2-user@<EC2_PUBLIC_IP> **# execute this cmd in local powershell or cmd**

```
PS C:\Users\[redacted]\Downloads> ssh -i "virginia.pem" -L 8080:localhost:8080 ec2-user@52.201.236.104
The authenticity of host '52.201.236.104 (52.201.236.104)' can't be established.
ED25519 key fingerprint is SHA256:U61otEM62GNwlpM2Aar0B5M9nHiCV1Gkv2EE5fH25zg.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '52.201.236.104' (ED25519) to the list of known hosts.
ec2-user@52.201.236.104: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
PS C:\Users\[redacted]\Downloads> ssh -i "virginia.pem" -L 8080:localhost:8080 ec2-user@52.201.236.104
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Mon Apr 14 09:08:48 2025
[ec2-user@ip-172-31-19-95 ~]$ sudo -i
[root@ip-172-31-19-95 ~]#
```

After connection in powershell only the application can access in locally otherwise it not possible

Then open this in your **local browser**:

http://localhost:8080 (a secure tunnel will form from ec2 and locally)



After change the replica set in deployment.yml do manually or execute the below cmd

- kubectl scale deployment myapp-deployment --replicas=4
- kubectl get pods

```
[ec2-user@ip-172-31-19-95 ~]$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
myapp-deployment-58dbdb69b8-9vjhv   1/1     Running   0          12m
myapp-deployment-58dbdb69b8-gnwdf   1/1     Running   0          12m
myapp-deployment-58dbdb69b8-m4mbg   1/1     Running   0          9s
myapp-deployment-58dbdb69b8-tvcgf   1/1     Running   0          9s
```

Describe and Check Logs

- kubectl describe pod <pod-name>

```
[ec2-user@ip-172-31-19-95 ~]$ kubectl describe pod myapp-deployment-58dbdb69b8-9vjhv
Name:             myapp-deployment-58dbdb69b8-9vjhv
Namespace:        default
Priority:         0
Service Account:  default
Node:             minikube/192.168.49.2
Start Time:       Mon, 14 Apr 2025 09:15:57 +0000
Labels:           app=myapp
                  pod-template-hash=58dbdb69b8
Annotations:      <none>
Status:           Running
IP:               10.244.0.4
IPs:
  IP:           10.244.0.4
Controlled By:  ReplicaSet/myapp-deployment-58dbdb69b8
Containers:
  myapp:
    Container ID:   docker://07dae5a1fca354942f1278ce2ee05c3c8cab1efd5804e6d92cb1389a0d5fbaf8
```

- kubectl logs <pod-name>

```
[ec2-user@ip-172-31-19-95 ~]$ kubectl describe pod myapp-deployment-58dbdb69b8-9vjhv
Name:             myapp-deployment-58dbdb69b8-9vjhv
Namespace:        default
Priority:         0
Service Account:  default
Node:             minikube/192.168.49.2
Start Time:       Mon, 14 Apr 2025 09:15:57 +0000
Labels:           app=myapp
                  pod-template-hash=58dbdb69b8
Annotations:      <none>
Status:           Running
IP:               10.244.0.4
IPs:
  IP:           10.244.0.4
Controlled By:  ReplicaSet/myapp-deployment-58dbdb69b8
Containers:
  myapp:
    node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type    Reason     Age   From               Message
  ----    ------     ----  ----               -------
  Normal  Scheduled  13m   default-scheduler  Successfully assigned default/myapp-deployment-58dbdb69b8-9vjhv to minikube
  Normal  Pulling    13m   kubelet            Pulling image "nginx"
  Normal  Pulled     13m   kubelet            Successfully pulled image "nginx" in 146ms (4.842s including waiting). Image size: 192056179 bytes.
  Normal  Created    13m   kubelet            Created container: myapp
  Normal  Started    13m   kubelet            Started container myapp
[ec2-user@ip-172-31-19-95 ~]$
```

- kubectl describe node <node-name>

```
[ec2-user@ip-172-31-19-95 ~]$ kubectl describe minikube
error: the server doesn't have a resource type "minikube"
[ec2-user@ip-172-31-19-95 ~]$ kubectl describe node minikube
Name:               minikube
Roles:              control-plane
Labels:             beta.kubernetes.io/arch=amd64
                    beta.kubernetes.io/os=linux
                    kubernetes.io/arch=amd64
                    kubernetes.io/hostname=minikube
                    kubernetes.io/os=linux
                    minikube.k8s.io/commit=dd5d320e41b5451cdf3c01891bc4e13d189586ed-dirty
                    minikube.k8s.io/name=minikube
                    minikube.k8s.io/primary=true
                    minikube.k8s.io/updated_at=2025_04_14T09_10_15_0700
                    minikube.k8s.io/version=v1.35.0
                    node-role.kubernetes.io/control-plane=
                    node.kubernetes.io/exclude-from-external-load-balancers=
Annotations:        kubeadm.alpha.kubernetes.io/cri-socket: unix:///var/run/cri-dockerd.sock
                    node.alpha.kubernetes.io/ttl: 0
                    volumes.kubernetes.io/controller-managed-attach-detach: true
CreationTimestamp:  Mon, 14 Apr 2025 09:10:12 +0000
Taints:             <none>
Unschedulable:      false
Lease:
  HolderIdentity:  minikube
```

 # For minikube install purpose required 2cpu and 2gib ram

```
Capacity:
  cpu:                2
  ephemeral-storage:  8310764Ki
  hugepages-2Mi:      0
  memory:             3997868Ki
  pods:               110
Allocatable:
  cpu:                2
  ephemeral-storage:  8310764Ki
  hugepages-2Mi:      0
  memory:             3997868Ki
  pods:               110
```