**Lab Exercise 18- Using NodeSelector in Kubernetes**

This exercise will guide you through creating a Kubernetes deployment with a NodeSelector to ensure that pods are scheduled on specific nodes based on labels.

**Step 1: Label the Nodes**

Identify the nodes in your cluster:

kubectl get nodes

Label a node:

Choose one of the nodes and label it. Replace <node-name> with the name of your chosen node and mylabel and myvalue with your desired label key and value:

kubectl label nodes <node-name> mylabel=myvalue

Example:

kubectl label nodes node1 mylabel=myvalue

Verify the node label:

kubectl get nodes --show-labels

**Step 2: Create a Deployment with NodeSelector**

Create a YAML file for the deployment:

Create a file named **nginx-deployment.yaml** with the following content:

apiVersion: apps/v1

kind: Deployment

metadata:

name: nginx-deployment

spec:

replicas: 2

selector:

matchLabels:

app: nginx

template:

metadata:

labels:

app: nginx

spec:

nodeSelector:

mylabel: myvalue

containers:

- name: nginx

image: nginx:1.21

ports:

- containerPort: 80

Apply the deployment:

kubectl apply -f nginx-deployment.yaml

Verify that the pods are scheduled on the labeled node:

kubectl get pods -o wide

Check the NODE column to ensure that the pods are scheduled on the node with the label mylabel=myvalue.

**Step 3: Clean Up**

Delete the deployment:

kubectl delete -f nginx-deployment.yaml

Remove the label from the node (optional):

kubectl label nodes <node-name> mylabel-

Example:

kubectl label nodes node1 mylabel-