1. **For Instaling KubeCtl-**

curl.exe -LO "https://dl.k8s.io/release/v1.27.0/bin/windows/amd64/kubectl.exe"

2. **For Installing MiniKube-**

New-Item -Path 'c:\' -Name 'minikube' -ItemType Directory -Force

Invoke-WebRequest -OutFile 'c:\minikube\minikube.exe' -Uri 'https://github.com/kubernetes/minikube/releases/latest/download/minikube-windows-amd64.exe' -UseBasicParsing

Then-

$oldPath = [Environment]::GetEnvironmentVariable('Path', [EnvironmentVariableTarget]::Machine)

if ($oldPath.Split(';') -inotcontains 'C:\minikube'){ `

[Environment]::SetEnvironmentVariable('Path', $('{0};C:\minikube' -f $oldPath), [EnvironmentVariableTarget]::Machine) `

}

- **minikube start**

**deployment.yaml**

----------------

```

apiVersion: apps/v1

kind: Deployment

metadata:

name: django-deployment

labels:

app: myapp

spec:

replicas: 2

selector:

matchLabels:

app: myapp

template:

metadata:

labels:

app: myapp

spec:

containers:

- name: nidhi

image: nidhisharma28/docker-test1

ports:

- containerPort: 8000

```

-----------------------------------------------------------------------------------------------------

**service.yaml**

--------------

```

apiVersion: v1

kind: Service

metadata:

name: service

spec:

type: NodePort

selector:

app: myapp

ports:

- protocol: TCP

port: 80

targetPort: 8000

nodePort: 31257

```

-------------------------------------------------------------------------------------------------

3. **echo $oldPath**

4. **kubectl apply -f deployment.yaml**

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

6. **minikube service <service-name>**

7. **Create ingress file**

Ingress.yaml

apiVersion: networking.k8s.io/v1

kind: Ingress

metadata:

name: nidhi-ingress

spec:

ingressClassName: nginx

rules:

- host: nidhi-sharma.com

http:

paths:

- path: /

pathType: Prefix

backend:

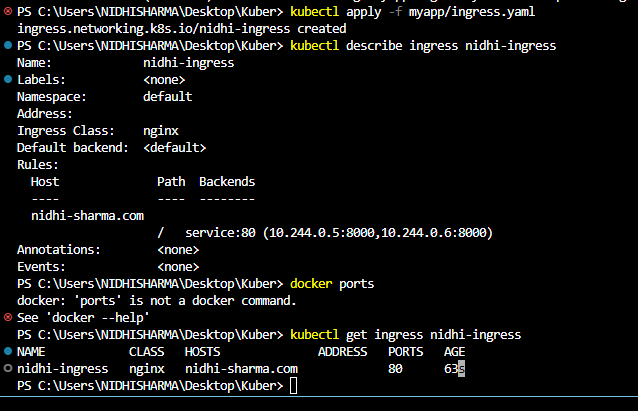
service:

name: service

port:

number: 80

8**. Run commands**



----------------------------------------END-------------------------------------------------