task 1

Reconfigure the existing deployment front-end and add a port specification named http

#exposing port 80/tcp of the existing container nginx.

Create a new service named front-end-svc exposing the container port http.

Configure the new service to also expose the individual Pods

#via a NodePort on the nodes on which they are scheduled.

Before editing

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: front-end
spec:
  replicas: 2
  selector:
    matchLabels:
      app: front-end
  template:
    metadata:
      labels:
        app: front-end
    spec:
      containers:
      - name: nginx
        image: nginx:latest
~
```

```yaml
    app: front-end
  spec:
    containers:
    - image: nginx:latest
      imagePullPolicy: Always
      name: nginx
      ports:
      - containerPort: 80
        name: http
        protocol: TCP
      resources: {}
      terminationMessagePath: /dev/termination-log
      terminationMessagePolicy: File
    dnsPolicy: ClusterFirst
    restartPolicy: Always
```

Edit pod yaml

```yaml
  name: front-end
spec:
  replicas: 2
  selector:
    matchLabels:
      app: front-end
  template:
    metadata:
      labels:
        app: front-end

    spec:
      containers:
      - name: nginx
        image: nginx:latest
        ports:
        - containerPort: 80
          name: http

~
```

```
[fouad2@client ~]$ kubectl apply -f front-end-deploy.yaml
deployment.apps/front-end configured
[fouad2@client ~]$
```

```
[fouad2@client ~]$ vim front-end-svc.yaml
[fouad2@client ~]$
[fouad2@client ~]$ kubectl apply -f front-end-svc.yaml
```

Svc.yaml

```yaml
apiVersion: v1
kind: Service
metadata:
  name: front-end-svc
spec:
  selector:
    app: front-end
  type: NodePort
  ports:
  - name: http
    port: 80
    targetPort: http
    nodePort: 30080
~
```

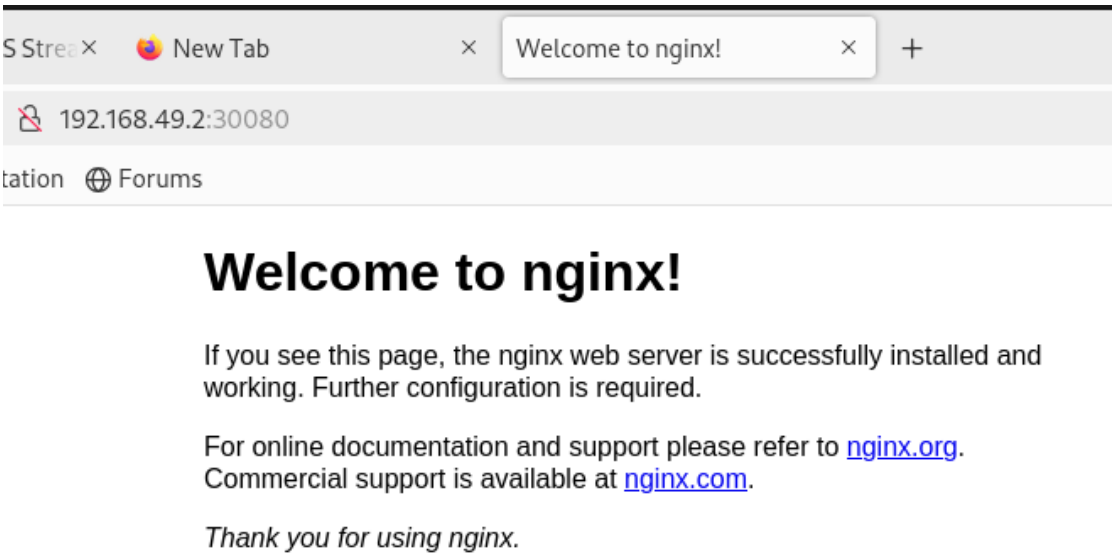
Verify

```
[fouad2@client ~]$ kubectl get svc front-end-svc
NAME            TYPE       CLUSTER-IP      EXTERNAL-IP   PORT(S)        AGE
front-end-svc   NodePort   10.96.205.198   <none>        80:30080/TCP   3h37m
[fouad2@client ~]$
[fouad2@client ~]$
[fouad2@client ~]$ minikube ip
192.168.49.2
```

From cluster

```
[fouad2@client ~]$ curl http://192.168.49.2:30080
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
```

From web



Task 2

```
[fouad2@client ~]$
[fouad2@client ~]$ kubectl label node minikube disk=ssd
node/minikube labeled
```

```
]$ vim nginx-pod.yaml
```

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx-kusc00401
spec:
  nodeSelector:
    disk: ssd
  containers:
    - name: nginx
      image: nginx
      ports:
        - containerPort: 80
```

```
[fouad2@client ~]$ kubectl get pod nginx-kusc00401 -o wide
NAME              READY   STATUS    RESTARTS   AGE     IP           NODE       NOMINATED NODE   REA
DINESS GATES
nginx-kusc00401   1/1     Running   0          2m14s   10.244.0.27   minikube   <none>           <no
ne>
```

```
kubectl get pod nginx-kusc00401 -o wide
```

```
Volumes:
  kube-api-access-8pk6d:
    Type:                    Projected (a volume that contains injected da
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    Optional:                false
    DownwardAPI:             true
QoS Class:                   BestEffort
Node-Selectors:              disk=ssd
```

Task 3

```
apiVersion: v1
kind: Pod
metadata:
  name: foo
spec:
  containers:
  - name: foo-container
    image: busybox
    command: ["sh", "-c", "while true; do echo 'error file-not-found'; sleep 5; done"]
```

```
[fouad2@client ~]$ vim foo-pod.yaml
[fouad2@client ~]$
[fouad2@client ~]$
[fouad2@client ~]$ kubectl apply -f foo-pod.yaml
pod/foo created
```

```
[fouad2@client ~]$ kubectl logs foo
error file-not-found
error file-not-found
error file-not-found
error file-not-found
error file-not-found
```

```
[fouad2@client ~]$ sudo mkdir -p /opt/KUTR00101
[sudo] password for fouad2:
[fouad2@client ~]$
```

```
[fouad2@client ~]$
[fouad2@client ~]$ kubectl logs foo | grep "file-not-found" | sudo tee /opt/KUTR00101/foo
error file-not-found
error file-not-found
error file-not-found
```

```
[fouad2@client ~]$ cat /opt/KUTR00101/foo
error file-not-found
error file-not-found
error file-not-found
error file-not-found
error file-not-found
error file-not-found
```