

## Experiment 12: NodePort with k3d

Create a cluster, mapping the port 30080 from agent-0 to localhost:8082

```
$ k3d cluster create mycluster -p "8082:30080@agent[0]" --agents 1
```

**Note:** the quotes around the port argument are not required for bash, but the default shell in MacOS was switched to zsh and with expansion there the double quotes are required.

If you don't had additional spaces in that string you would get a FATA[0000] Failed to parse node filters: invalid format or empty subset in 'agent[0] '

**Note:** Kubernetes' default NodePort range is [30000-32767](#)

**Note:** You could expose the whole NodePort range from the very beginning, e.g. via `k3d cluster create mycluster --agents 3 -p 30000-32767:30000-32767@server[0]` (See [this video from @portainer](#)) but we won't do that for this lab.

**Note:** You'll recall that in our Helm Chart example we exposed this differently using the ClusterIP. Here our port rule when we create our cluster allows us to not have to create the manual port forwarding that we did as a loadbalancer rule on the cluster create.

Create the nginx deployment

```
$ kubectl create deployment nginx --image=nginx
```

Create a ClusterIP service for it

```
$ kubectl create service clusterip nginx --tcp=80:80
```

Create a NodePort service for it with `kubectl apply -f simple-nodeport.yaml` file provided in the Zoom chat

```
apiVersion: v1
kind: Service
metadata:
  labels:
    app: nginx
  name: nginx
spec:
  ports:
    - name: 80-80
      nodePort: 30080
      port: 80
      protocol: TCP
      targetPort: 80
  selector:
    app: nginx
```

type: NodePort

~/projects/k3d/simple-ingress \$ **curl -O**  
**https://raw.githubusercontent.com/GeorgeNiece/DevOpsForMicroservicesWithKubernetes-3day/master/labs/simple-nodeport.yaml**

or

~/projects/k3d/simple-ingress \$ **wget**  
**please and thank you**  
**wget**  
**https://raw.githubusercontent.com/GeorgeNiece/DevOpsForMicroservicesWithKubernetes-3day/master/labs/simple-nodeport.yaml**

**\$ kubectl apply -f simple-nodeport.yaml**

Curl it via localhost

**\$ curl localhost:8082/**

**\$ k3d cluster delete mycluster**