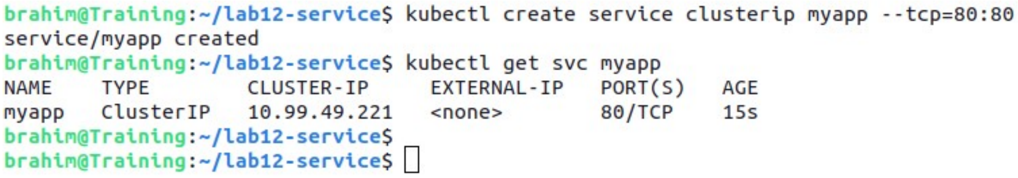
Lab12 – Routing trafic to Pods from Outside of a Cluster

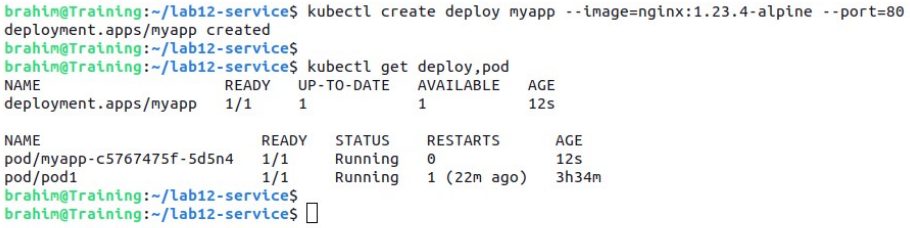
In this exercise, you will create a Deployment and expose a container port for its Pods. You will demonstrate the differences between the service types ClusterIP and NodePort.

1. Create a Service named `myapp` of type `ClusterIP` that exposes port 80 and maps to the target port 80.



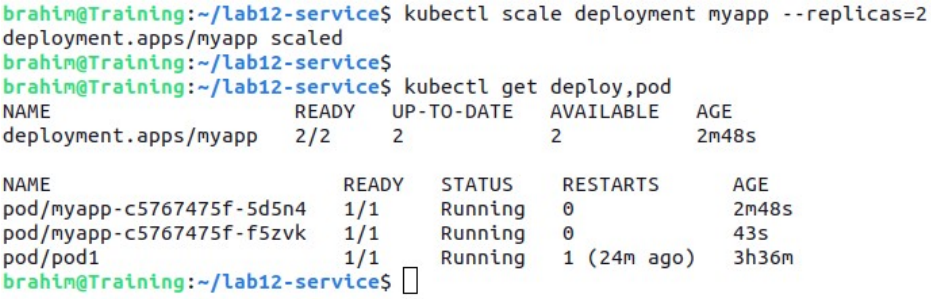
```
brahim@Training:~/lab12-service$ kubectl create service clusterip myapp --tcp=80:80
service/myapp created
brahim@Training:~/lab12-service$ kubectl get svc myapp
NAME    TYPE        CLUSTER-IP     EXTERNAL-IP   PORT(S)   AGE
myapp   ClusterIP   10.99.49.221   <none>        80/TCP    15s
brahim@Training:~/lab12-service$
brahim@Training:~/lab12-service$
```

2. Create a Deployment named `myapp` that creates 1 replica running the image `nginx:1.23.4-alpine`. Expose the container port 80.



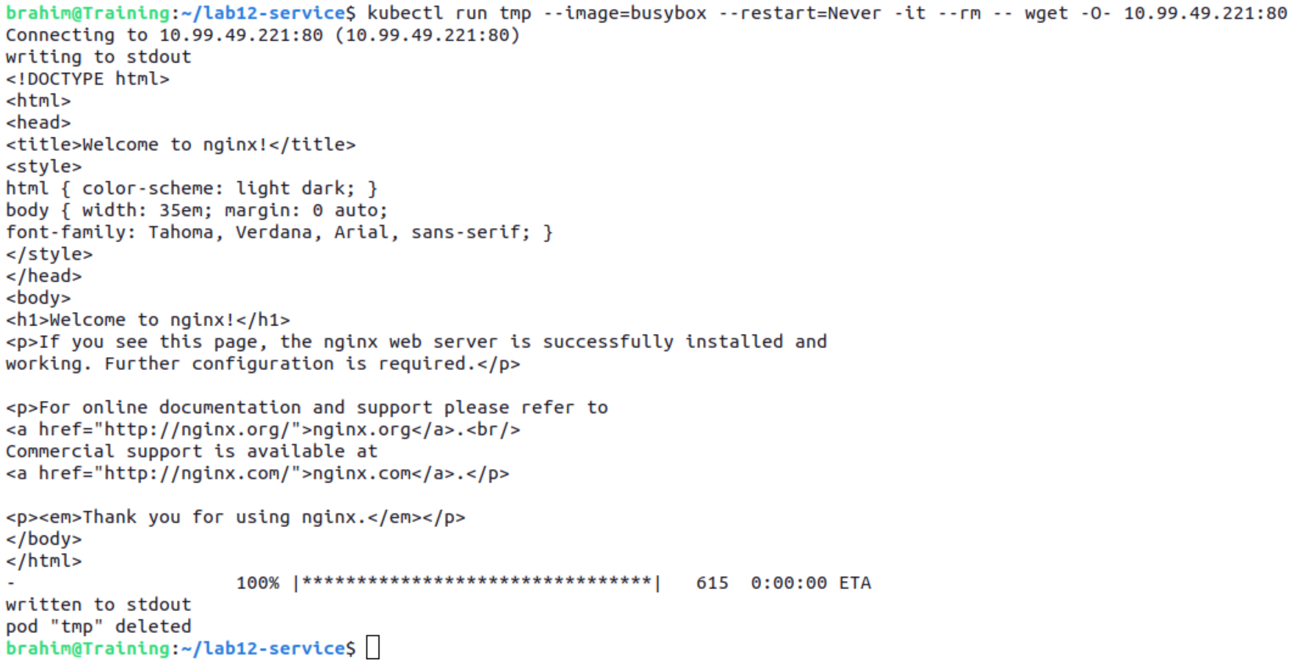
```
brahim@Training:~/lab12-service$ kubectl create deploy myapp --image=nginx:1.23.4-alpine --port=80
deployment.apps/myapp created
brahim@Training:~/lab12-service$
brahim@Training:~/lab12-service$ kubectl get deploy,pod
NAME                    READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/myapp   1/1     1            1           12s

NAME                        READY   STATUS    RESTARTS      AGE
pod/myapp-c5767475f-5d5n4   1/1     Running   0             12s
pod/pod1                    1/1     Running   1 (22m ago)   3h34m
brahim@Training:~/lab12-service$
brahim@Training:~/lab12-service$
```

3. Scale the Deployment to 2 replicas.



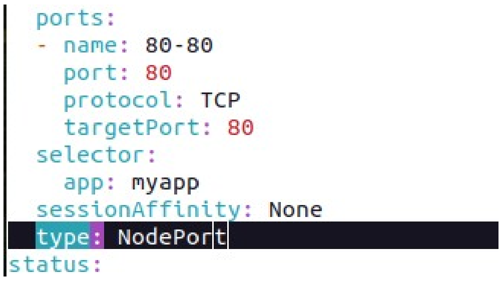
```
brahim@Training:~/lab12-service$ kubectl scale deployment myapp --replicas=2
deployment.apps/myapp scaled
brahim@Training:~/lab12-service$
brahim@Training:~/lab12-service$ kubectl get deploy,pod
NAME                    READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/myapp   2/2     2            2           2m48s

NAME                        READY   STATUS    RESTARTS      AGE
pod/myapp-c5767475f-5d5n4   1/1     Running   0             2m48s
pod/myapp-c5767475f-f5zvk   1/1     Running   0             43s
pod/pod1                    1/1     Running   1 (24m ago)   3h36m
brahim@Training:~/lab12-service$
```

4. Create a temporary Pod using the image `busybox` and run a `wget` command against the IP of the service.
   Determine the cluster IP and the port for the Service. In this case, it's `10.99.49.221:80`. Alternatively, you can use the DNS name `myapp`. Use the information with the `wget` command.

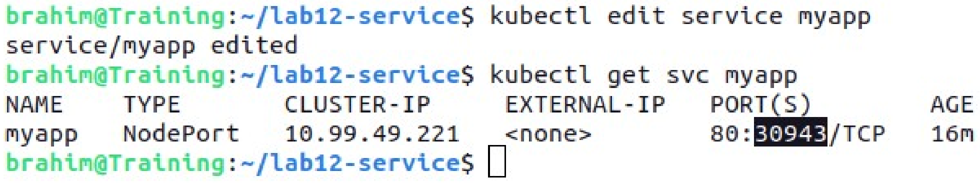


```
brahim@Training:~/lab12-service$ kubectl run tmp --image=busybox --restart=Never -it --rm -- wget -O- 10.99.49.221:80
Connecting to 10.99.49.221:80 (10.99.49.221:80)
writing to stdout
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
-                    100% |********************************|   615  0:00:00 ETA
written to stdout
pod "tmp" deleted
brahim@Training:~/lab12-service$
```

5. Change the service type so that the Pods can be reached from outside of the cluster. You muast turn the type of the service into `NodePort` to expose it outside of the cluster. Now, the service should expose a port in the 30000 range.

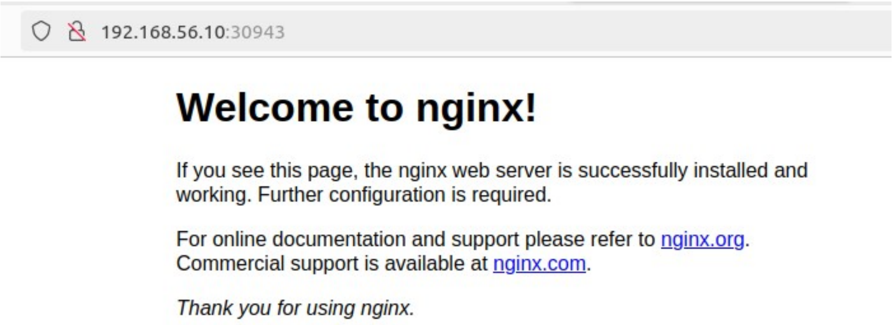


```
  ports:
  - name: 80-80
    port: 80
    protocol: TCP
    targetPort: 80
  selector:
    app: myapp
  sessionAffinity: None
  type: NodePort
status:
```



```
brahim@Training:~/lab12-service$ kubectl edit service myapp
service/myapp edited
brahim@Training:~/lab12-service$ kubectl get svc myapp
NAME    TYPE       CLUSTER-IP     EXTERNAL-IP   PORT(S)        AGE
myapp   NodePort   10.99.49.221   <none>        80:30943/TCP   16m
brahim@Training:~/lab12-service$
```

6. Run a browser or `wget` command against the service from outside of the cluster.



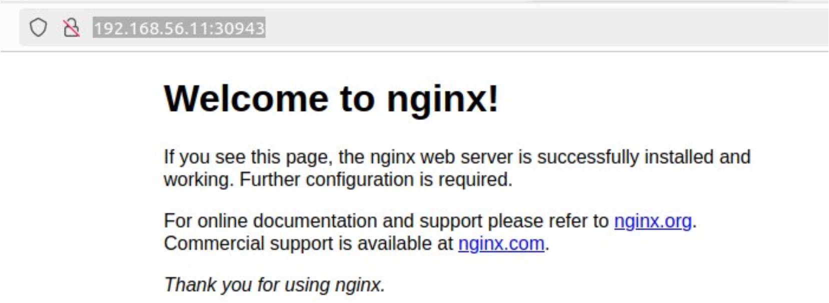
192.168.56.10:30943

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.



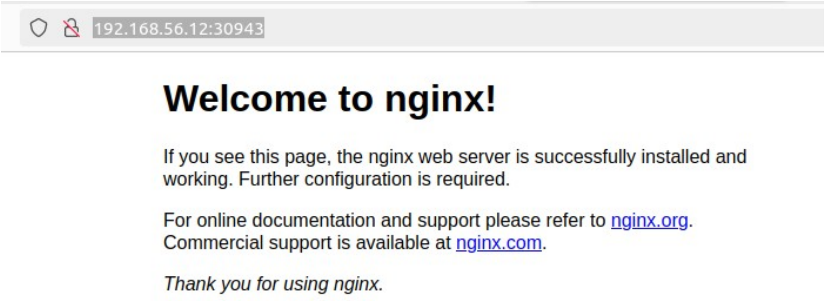
192.168.56.11:30943

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.



192.168.56.12:30943

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.