

Name: Shweta Singh

Sap Id: 500098159

Roll No: R2142211484

Course and Batch: Btech CSE(DevOps)-B4

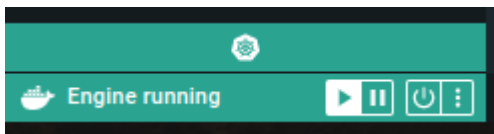
Submitted to: Dr Hitesh Kumar Sharma

Lab Exercise 8– Creating Service in Kubernetes

Below is a lab exercise that will help you understand and practice creating a service in Kubernetes:

Task 1: Start Kubernetes in Docker-Desktop

- Start Kubernetes service in Docker-Desktop



Task 2: Creating a Service

Create a service to expose the deployed application within the Kubernetes cluster. You can use the following sample YAML manifest as a reference:

```
apiVersion: v1
kind: Service
metadata:
  name: my-service
spec:
  selector:
    app: lbnginx
```

```
ports:

- protocol: TCP

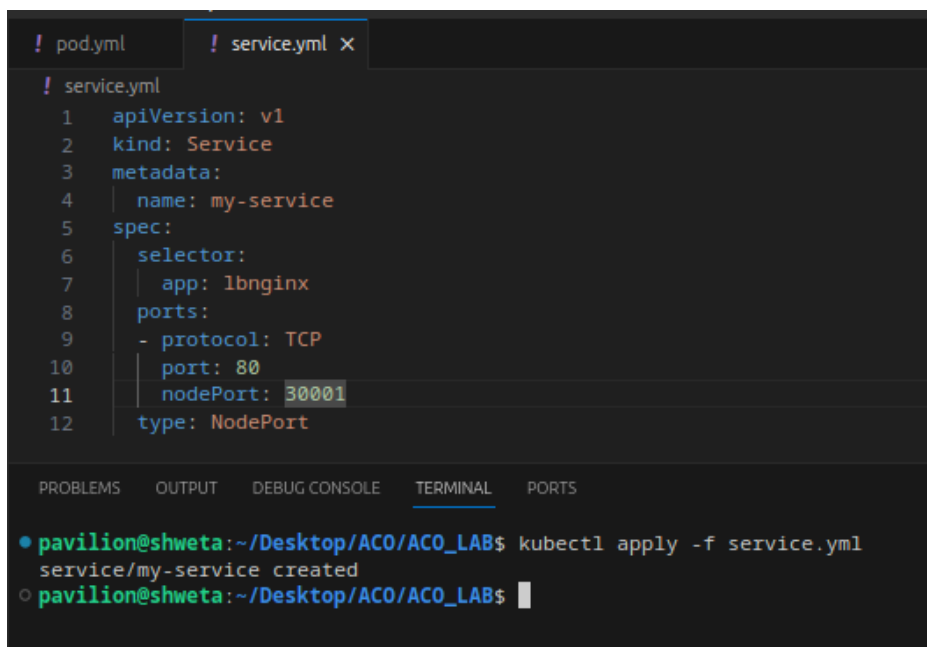
  port: 80

  nodePort: 30001

type: NodePort
```

- Apply the service using the following command:

```
kubectl apply -f service.yaml
```



The screenshot shows a code editor with two tabs: 'pod.yml' and 'service.yml'. The 'service.yml' tab is active, displaying the following YAML content:

```
! service.yml
1  apiVersion: v1
2  kind: Service
3  metadata:
4    name: my-service
5  spec:
6    selector:
7      app: lbnginx
8    ports:
9      - protocol: TCP
10        port: 80
11        nodePort: 30001
12    type: NodePort
```

Below the code editor is a terminal window with the following output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

● pavilion@shweta:~/Desktop/ACO/ACO_LAB$ kubectl apply -f service.yml
service/my-service created
○ pavilion@shweta:~/Desktop/ACO/ACO_LAB$
```

- Verify that the service is created by running the following command:

```
kubectl get services
```

```

pavilion@shweta:~/Desktop/ACO/ACO_LAB$ kubectl get all
NAME                                READY   STATUS    RESTARTS   AGE
pod/my-nginx-pod                   1/1     Running   0           84s

NAME                                TYPE               CLUSTER-IP   EXTERNAL-IP   PORT(S)          AGE
service/kubernetes                 ClusterIP          10.96.0.1    <none>         443/TCP           2d16h
service/my-service                 NodePort           10.107.144.167 <none>         80:30001/TCP      123m
pavilion@shweta:~/Desktop/ACO/ACO_LAB$ kubectl get services
NAME            TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)          AGE
kubernetes     ClusterIP   10.96.0.1    <none>         443/TCP           2d16h
my-service     NodePort    10.107.144.167 <none>         80:30001/TCP      124m
pavilion@shweta:~/Desktop/ACO/ACO_LAB$

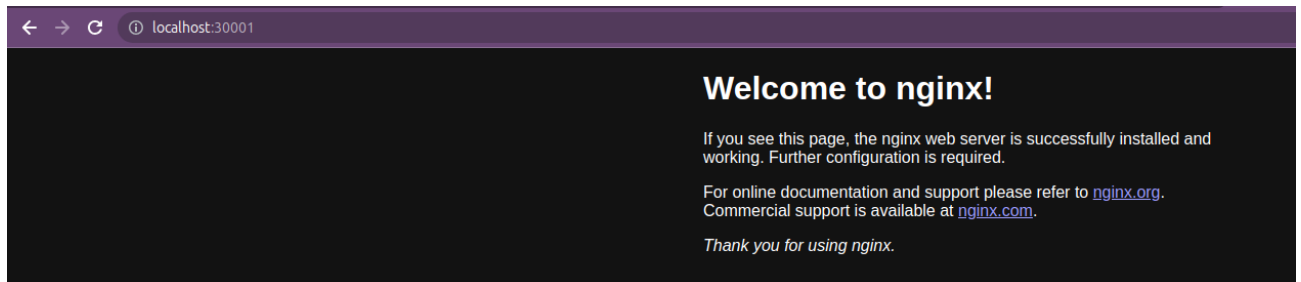
```

Task 4: Accessing the Service

- Access the service using port forwarding. Run the following command:

Access the Nginx server running in the service by opening a web browser and navigating to

```
http://localhost:30001
```



Task 5: Deleting the Service

Delete the service using the following command:

```
kubectl delete service my-service
```

Verify that the service has been deleted by running the kubectl get services command.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
● pavilion@shweta:~/Desktop/ACO/ACO_LAB$ kubectl delete service my-service
service "my-service" deleted
● pavilion@shweta:~/Desktop/ACO/ACO_LAB$ kubectl get services
NAME          TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)    AGE
kubernetes    ClusterIP   10.96.0.1    <none>        443/TCP    2d16h
○ pavilion@shweta:~/Desktop/ACO/ACO_LAB$
```

Task 6: Cleanup

Delete any remaining deployments, services, and resources created during the exercise using the appropriate kubectl delete commands.

Task 7: Documentation and Best Practices

Document your findings and the best practices for creating and managing services in Kubernetes.

Through this exercise, you'll gain a better understanding of how to create and manage services to expose applications within a Kubernetes cluster. Adjust the exercise based on your specific use case and requirements.