### **Lab Exercise 2- Create Service in Kubernetes**

### **Objective:**

* Understand the syntax and structure of a Kubernetes Service definition file (YAML).
* Learn to create different types of Services: ClusterIP, NodePort, and LoadBalancer.
* Comprehend how Services operate independently of specific Pods.

### **Prerequisites**

* Kubernetes Cluster: Have a running Kubernetes cluster (locally using Minikube or kind, or a cloud-based service).
* kubectl: Install and configure kubectl to interact with your Kubernetes cluster.
* Basic Knowledge of YAML: Familiarity with YAML format will be helpful for understanding Kubernetes resource definitions.

### **Step-by-Step Guide**

**Step 1: Understand Kubernetes Services**

In Kubernetes, Services provide a way to expose applications running on Pods. They allow these applications to communicate internally and externally, providing a stable interface despite the dynamic nature of Pods.

**Service Types**

* **ClusterIP:** Exposes the Service on a cluster-internal IP. This is the default type and is accessible only within the cluster.
* **NodePort:** Exposes the Service on each Node's IP at a static port. This makes the Service accessible from outside the cluster using the Node's IP and the NodePort.
* **LoadBalancer:** Exposes the Service externally using a cloud provider's load balancer. It provides an external IP address that forwards traffic to the Service.

**1. NodePort Service**

To expose the Service on a port on each Node in the cluster, modify the Service type to NodePort.

Create a YAML file named **service.yaml** with the following content:

apiVersion: v1

kind: Service

metadata:

name: nodeport-service

spec:

selector:

app: my-app

ports:

- protocol: TCP

port: 80

targetPort: 80

nodePort: 30007 # A specific port in the range 30000-32767

type: NodePort

**Explanation:**

* The primary difference from the ClusterIP Service is the addition of nodePort, which specifies the static port on each Node.
* type: Set to NodePort, exposing the Service on a specific port across all Nodes.

**Apply this YAML to create the NodePort Service:**

kubectl apply -f service.yaml

**Verify the Service:**

kubectl get services

You should see the nodeport-service listed with a NodePort and details about the port exposed.

**Step 3: Cleaning Up Resources**

To delete the resources you created:

kubectl delete -f nodeport-service.yaml

This command deletes all the Services you defined.