**What is a Service in Kubernetes?**

A **Service** gives your app a stable way to talk to Pods, even though Pods can come and go (and their IPs change). It acts like a permanent doorway to a group of Pods.

**Types of Services**

1. **ClusterIP (default)**
   * Only works **inside** the cluster.
   * Used when Pods need to talk to each other.
2. **NodePort**
   * Opens a fixed port on **every Node**.
   * You can access your app using <NodeIP>:<nodePort> from **outside** the cluster.
3. **LoadBalancer**
   * For **cloud** environments.
   * Creates a public IP to forward traffic to your app.

**🔌 NodePort Explained**

A NodePort has three ports:

* **port** (required): The internal port used by the service.
* **targetPort** (optional): The port on the Pod. Defaults to port if not set.
* **nodePort** (optional): The external port. If not set, Kubernetes picks one from 30000–32767.

You can reach the app using:  
📍<NodeIP>:<nodePort>

**How it works:**  
When you hit the NodePort, Kubernetes sends the traffic to the right Pod behind the scenes.

**⚖️ Load Balancing with NodePort**

* If **multiple Pods** are on the **same Node**, Kubernetes splits the traffic between them.
* If Pods are on **different Nodes**, traffic is still spread out across them.  
  This keeps things fast and available.

**🔐 What is ClusterIP?**

* It gives a **private IP** for talking to Pods **inside** the cluster.
* Not accessible from outside.
* Best for internal communication (like frontend → backend).

**🏗️ How ClusterIP Helps in Multi-Tier Apps**

Let’s say you have 3 layers:

1. **Frontend Pods**
2. **Backend Pods**
3. **Redis Pods**

**Example Flow:**

* Frontend Pods talk to **Backend** using the backend **ClusterIP service**.
* Backend Pods talk to **Redis** using the redis **ClusterIP service**.

No need to remember Pod IPs — just use the service name!

**✅ Why Use ClusterIP?**

* **Stable Access**: Pod IPs change, service IPs don’t.
* **Built-in Load Balancing**: Traffic is spread evenly.
* **Easy to Use**: Just connect using the service name like http://backend.
* **Internal DNS**: Kubernetes automatically helps Pods find services by name.

**🧠 Summary**

A **ClusterIP service** is like a private phone number inside the cluster that always connects you to the right group of Pods. It makes app parts (like frontend and backend) talk easily and safely.