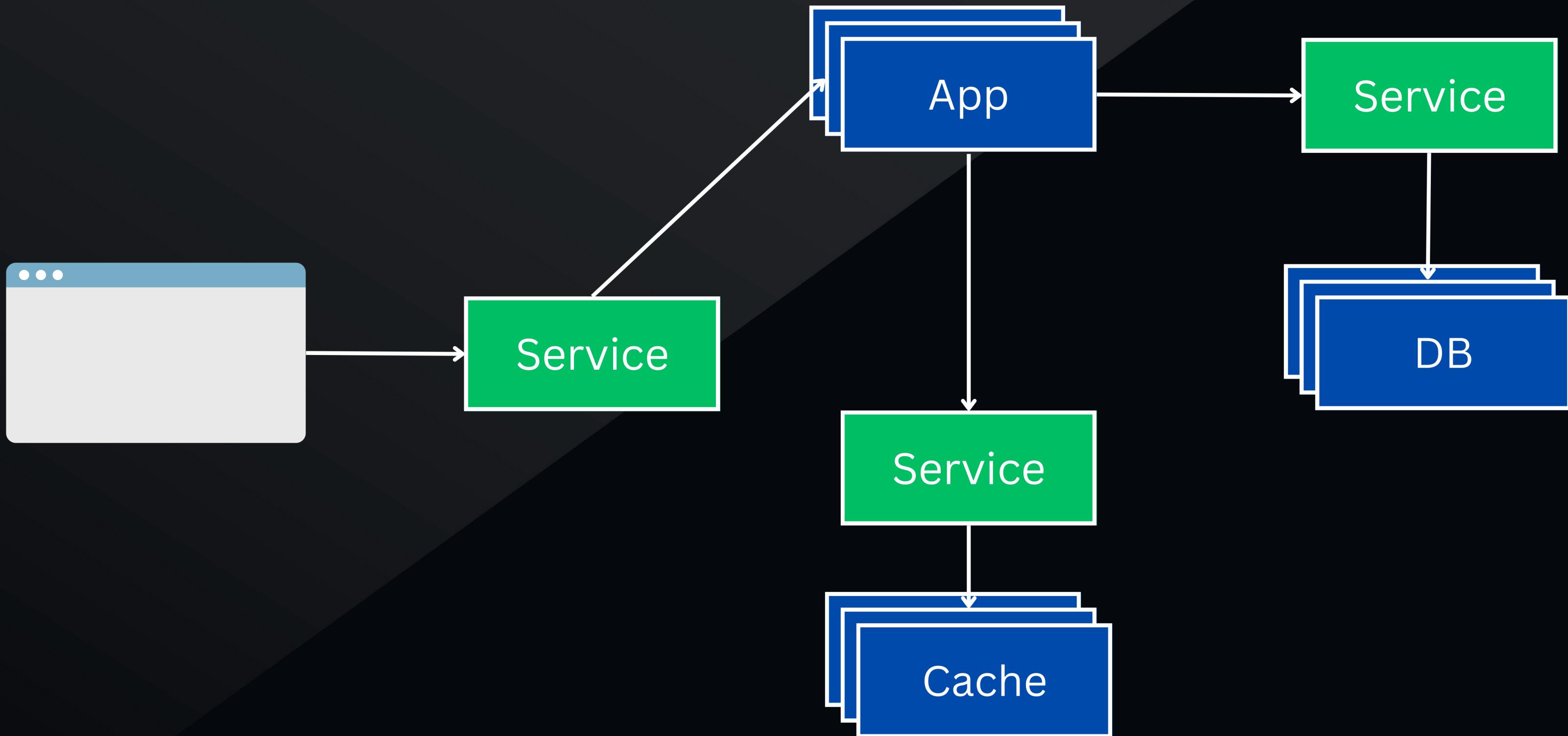


# SERVICES

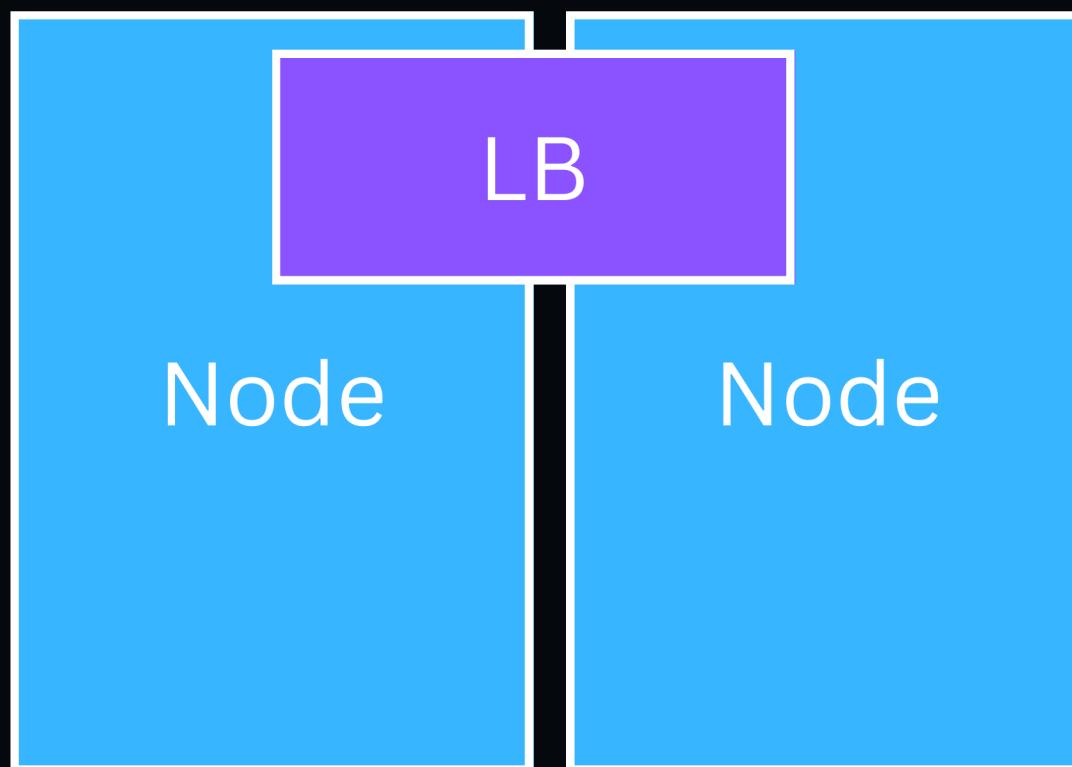
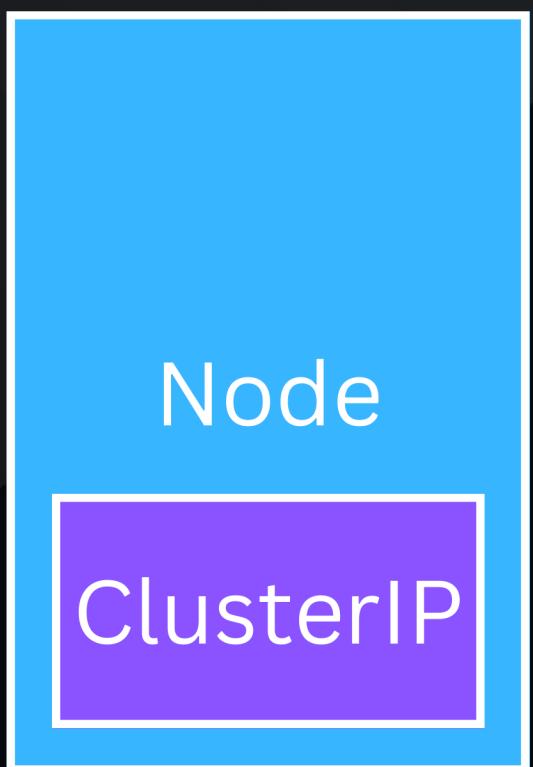
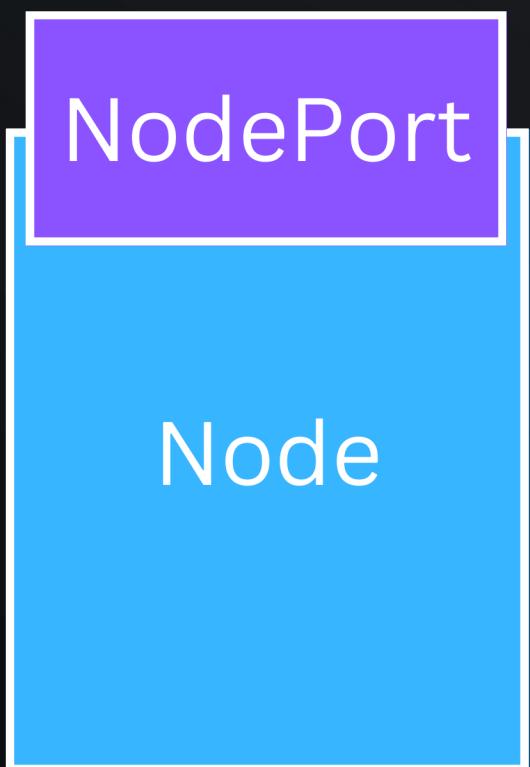
- It is an abstraction that provides a **stable cluster internal IP address and DNS name** for a set of pods that provide the same functionality
- It enables communication between different parts of an application running in a Kubernetes cluster by providing a single entry point for network traffic that is load-balanced across the pods that make up the service
- Service names in Kubernetes **must be unique within the same namespace**

# SERVICES



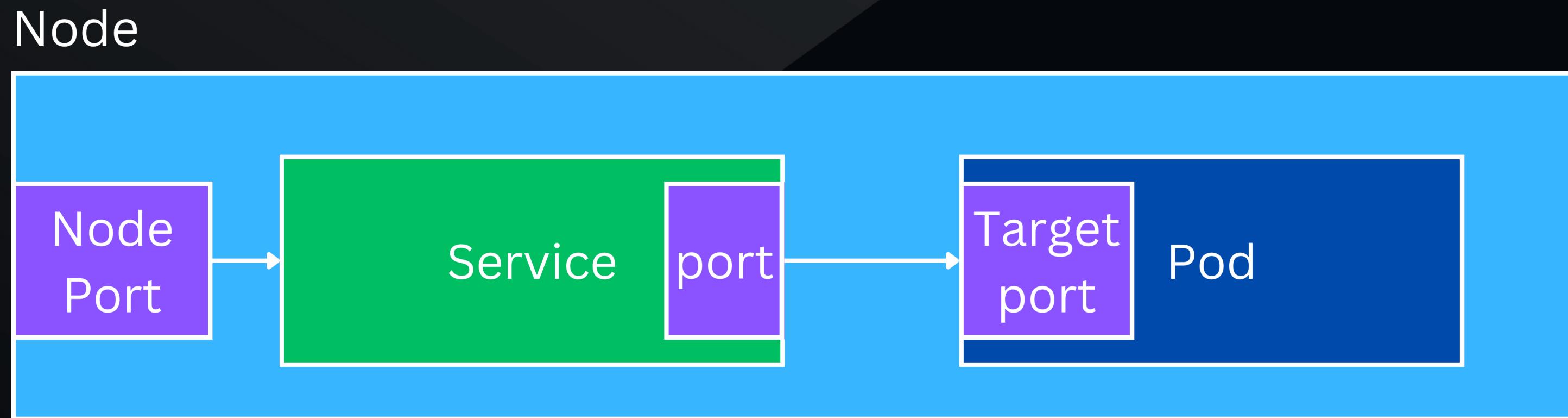
# SERVICES

- NodePort
- ClusterIP
- LoadBalancer



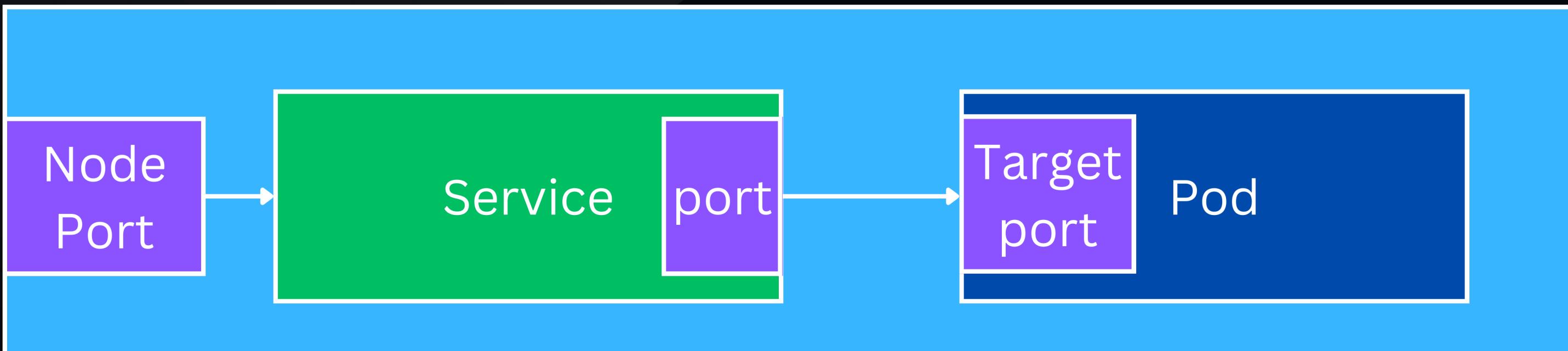
# NODE PORT

- NodePort is a service type that **exposes the service on a specific port on each node in the cluster**, allowing external traffic to be directed to the service



# NODE PORT

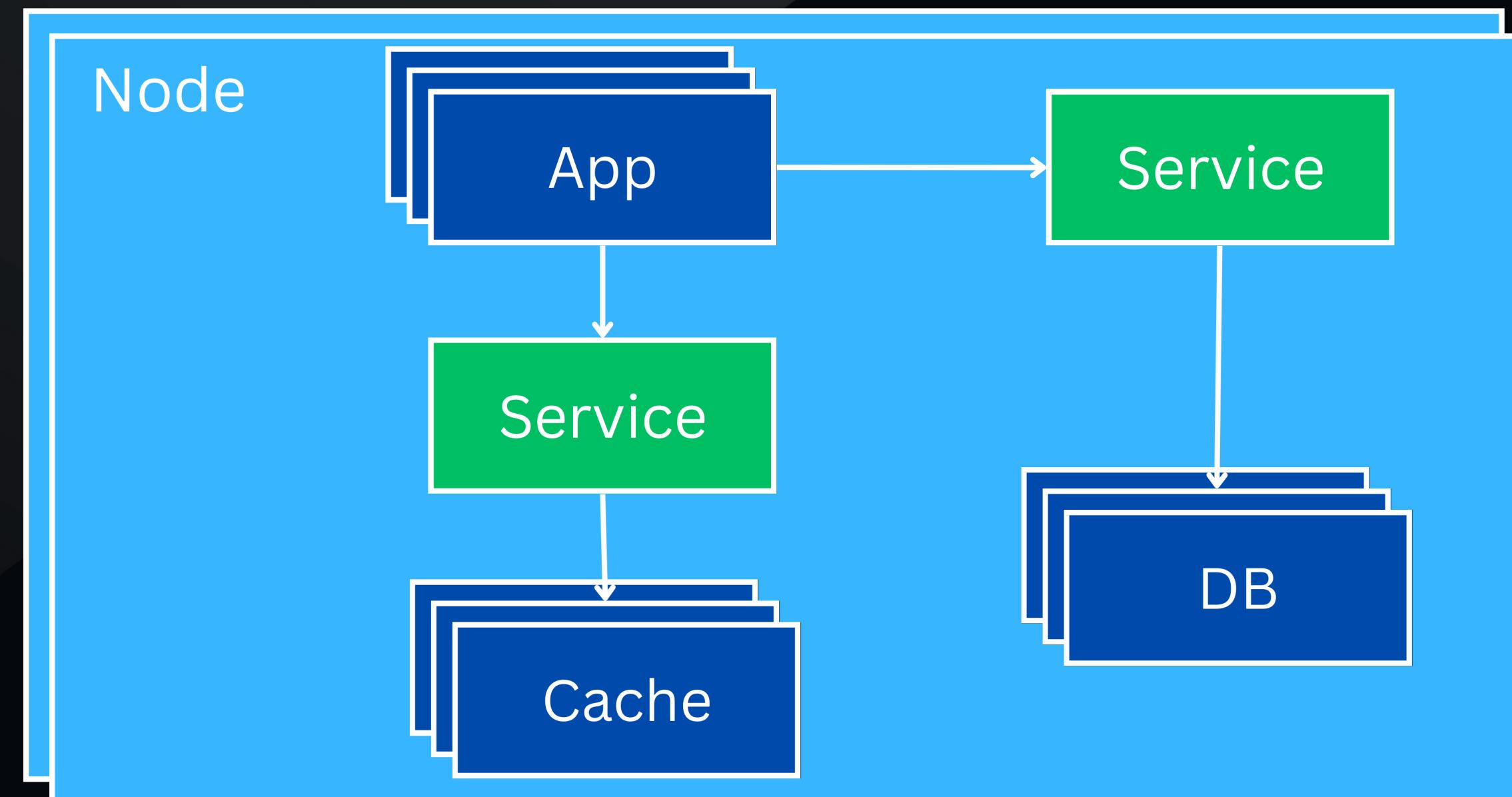
Node



```
● ● ●  
1  apiVersion: v1  
2  kind: Service  
3  metadata:  
4    name: nginx-service  
5  spec:  
6    type: NodePort  
7    selector:  
8      app.kubernetes.io/name: proxy  
9    ports:  
10   - name: nginx-service-port  
11     protocol: TCP  
12     port: 80  
13     targetPort: http-web-svc
```

# CLUSTER IP

- ClusterIP is the **default service type** in Kubernetes. It provides **a stable IP address and DNS name** for the service within the cluster
- ClusterIP **allows internal communication** between different parts of an application running in a Kubernetes cluster.



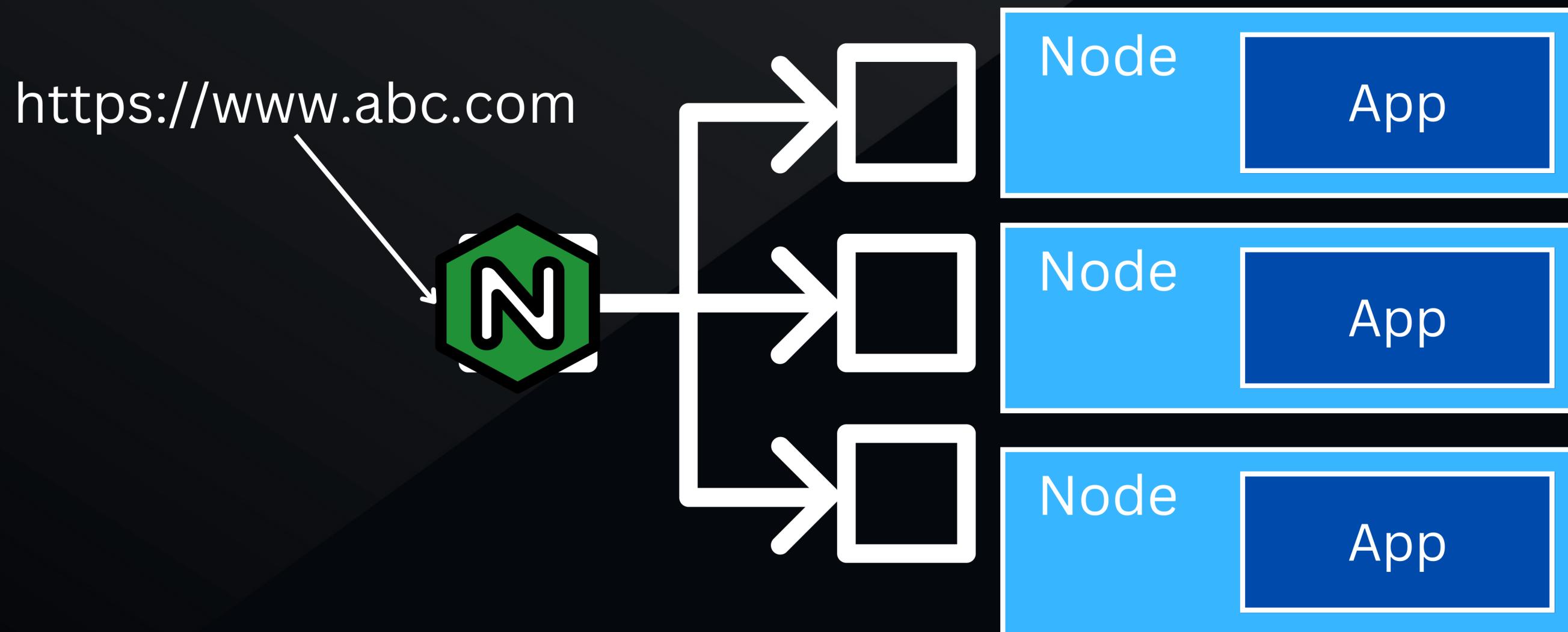
# CLUSTER IP



```
1 apiVersion: v1
2 kind: Service
3 metadata:
4   name: nginx-cl-svc
5 spec:
6   type: ClusterIP
7   selector:
8     app: nginx
9   ports:
10  - name: http
11    port: 80
12    targetPort: 80
13
```

# LOAD BALANCER

- LoadBalancer is a service type that provides **external access to the service using a cloud provider's load balancer**
- LoadBalancer **assigns a public IP address to the service and distributes incoming traffic** across the pods that make up the service



# LOAD BALANCER

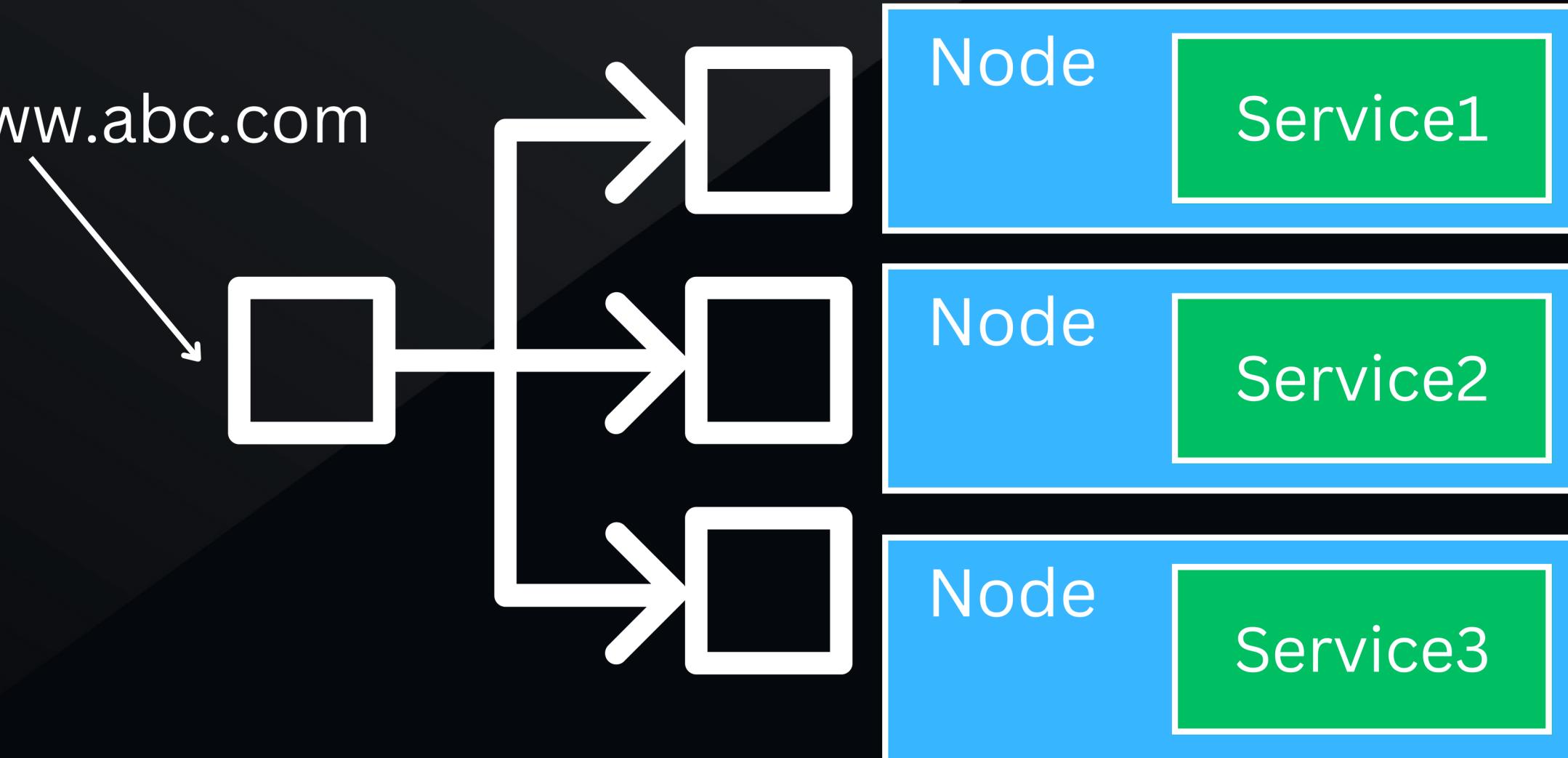


```
1 apiVersion: v1
2 kind: Service
3 metadata:
4   name: nginx-lb-svc
5 spec:
6   selector:
7     app: nginx
8   type: LoadBalancer
9   ports:
10  - name: http
11    port: 80
12    targetPort: 80
```

# INGRESS

- API object that provides **an entry point for external traffic to reach services** within a cluster
- An Ingress **defines rules for how incoming traffic should be routed to different services** based on the host name, path, or other attributes of the incoming request

https://www.abc.com



# INGRESS



```
1  apiVersion: networking.k8s.io/v1
2  kind: Ingress
3  metadata:
4    name: my-ingress
5  spec:
6    rules:
7      - host: mydomain.com
8        http:
9          paths:
10            - path: /app1
11              pathType: Prefix
12              backend:
13                service:
14                  name: app1-service
15                  port:
16                      name: http
17            - path: /app2
18              pathType: Prefix
19              backend:
20                service:
21                  name: app2-service
22                  port:
23                      name: http
24
```