## Introduction

1. Kubernetes is open-source.
2. Kubernetes is an container management tool.
3. It is used to run and manage containers running on virtual cloud machines.
4. Kubernetes is also known as k8s.

Working of Kubernetes

Kubernetes uses, objects to represent the desired state of the cluster. These objects are defined in manifest files, written in YAML format. Once we create the objects, the kubernetes system will constantly check and ensure that the objects exist and ensures that the cluster's actual state matches the desired state specified in the manifest files.

Kubernetes objects include:

- Pods
- Volumes
- Service
- ConfigMaps
- Jobs
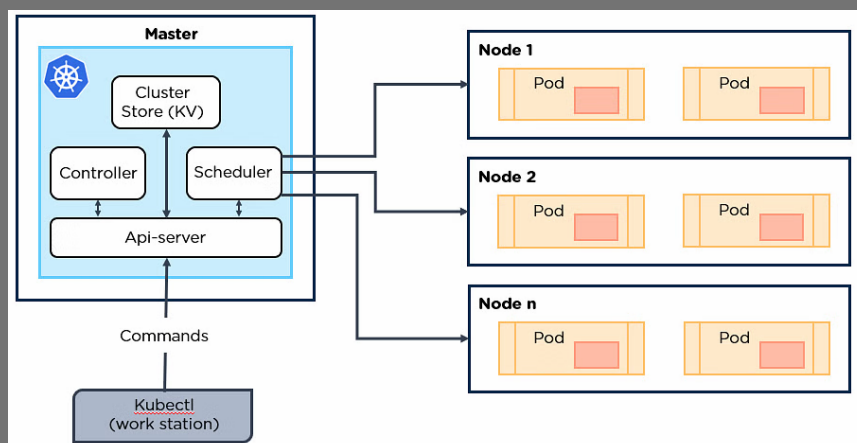
High-level and Relationships b/w Kubernetes Objects

1. Service: static (non-ephemeral) IP and Networking.
2. Replica sets: Auto-scaling and manage pods.
3. Volume: Non-ephemeral storage.(Ephemeral means the storage outside the Node).
4. Deployment: Versioning and Rollback.
5. Pod manages containers.
6. Replica sets manage pods.
7. Config maps and secrets helps you to configure pods.
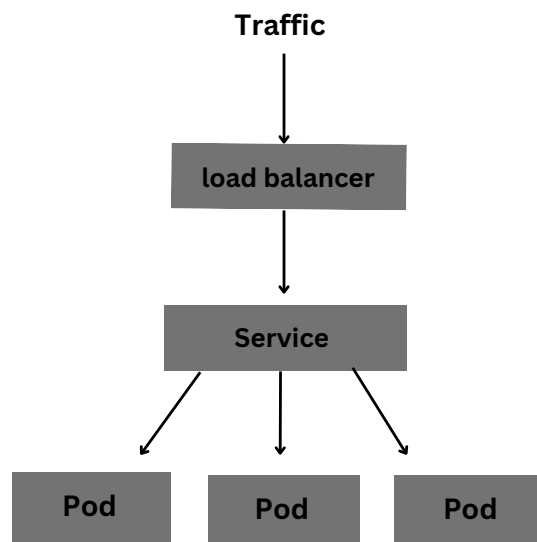
There are 2 Methods to run manifest.

1. Imperative Method: we directly issue commands and execute the work.
2. Declarative Method: we create a proper manifest write all the requirements in file and then execute the file.

Pod

- Pods runs on node, which is controlled by master.
- Pod is the smallest unit in Kubernetes.
- Pod manages containers.
- Pod is a groups of containers that are deployed together on the same host.
- When, a pod get created, it is scheduled accoding to master. If we want the pod to be created in our selected node then we need to specify in the manifest.
- A pod remains in the node. It will be deleted only in case when the process is terminated or if the node fails.
- Once the pod is terminated we cannot restart that exact pod. Instead we can create a new pod we same features and things.
- If the pod is deleted the volume attached to it will also be deleted.

Service:

```
                    Traffic

                       |
                       v

              ┌─────────────────┐
              │  load balancer  │
              └─────────────────┘

                       |
                       v

              ┌─────────────────┐
              │     Service     │
              └─────────────────┘
               /        |       \
              v         v        v
        ┌────────┐  ┌────────┐  ┌────────┐
        │  Pod   │  │  Pod   │  │  Pod   │
        └────────┘  └────────┘  └────────┘
```

Service is used to load balance the traffic among the pods. It constantly monitors the pods, in case a pod gets unhealthy, the service will start deploying the traffic to the other healthy pods.