Cluster contains multiple nodes but there will be only one master node.

**kubectl version**: to get version of kubernetes

**POD**

**kubectl run <pod\_name> --image=<image\_name>** -> it will run a new pod with image.

**kubectl run nginx --image=nginx** -> it will run a new pod called nginx with image as nginx.

**kubectl get pods** -> to get all the pods

**kubectl get pods -o wide** -> to get more details for the pods

**kubectl describe pod <pod-name>** -> it will give us the detailed info about the specific pod

Kubernetes yaml file must contain these fields -> apiVersion, kind, metadata, spec

|  |  |
| --- | --- |
| **kind** | **version** |
| Pod | v1 |
| Service | v1 |
| ReplicationController(deprecated) | v1 |
| ReplicaSet | apps/v1 |
| Deployment | apps/v1 |

nginx-pod.yaml

apiVersion: v1

kind: Pod

metadata:

  name: nginx

  tier: frontend

spec:

  containers:

    - name: nginx

      image: nginx

**kubectl apply/create -f nginx-pod.yaml** -> to create a pod with this yaml file.

**kubectl describe pod nginx** -> to give more description of the nginx pod

**kubectl edit pod nginx** -> it will open a vi editor where we can change the pod definition file also this is a in memory pod definition file which is maintained by Kubernetes.

**kubectl delete pod nginx** -> to delete the nginx pod

**kubectl delete --all pods** -> delete all the pods

**kubectl run redis --image=redis --dry-run=client -o yaml > redis-pod.yaml** -> It will not create any pod rather it’s a imperative style of writing definition file where a pod definition file will created with the necessary fields.

**ReplicaSet**

ReplicaSet is a group of same pods where we can scale in(reduce) or scale out(increase) the number of the pods.

**kubectl create/apply -f <replicaset-definition.yaml>** -> it will create a replicaset from the definition file.

**kubectl get replicateset** -> to get all the replicaset in the default namespace

**kubectl describe replicaset** -> to get all the replicaset in the default namespace

**kubectl delete replicaset <replicaset-name>** -> to delete the replicaset

nginx-replicaset.yaml

apiVersion: apps/v1

kind: ReplicaSet

metadata:

  name: nginx-replicaset

  labels:

    type: frontend

spec:

  selector:

    matchLabels:

      name: nginx-pod

      type: frontend

  replicas: 1

  template:

    metadata:

      labels:

        name: nginx-pod

        type: frontend

    spec:

      containers:

        - name: nginx

          image: nginx

labels under template section and mathLabels under selector should be same. That is how replicate set identifies the pod and controls the number of the pods. If we try to delete any pod or anyhow any pod got crashed then Kubernetes automatically brings another pod in.

Scale commands ->

**Kubectl replace -f nginx-replicaset.yaml** -> replace the previous nginx-repicaset with the current replicas given in the definition file

**Kubectl scale --replicas=6 -f nginx-replicaset.yaml** -> it will override the replicas given in the yaml

**Kubectl edit replicaset nginx-replicaset** -> it will open an editor and show the current configuration of the replicaset then we can easily scale out.

**Kubectl scale --replicas=6 replicaset nginx-replicaset** -> it will scale out an existing replicaset