**1) Creating Pod:**

Name of pod: webapp, image: nginx

* kubectl run webapp --image=nginx
* kubectl describe pod webapp
* kubectl delete pod webapp
* kubectl edit pod webapp
* kubectl run webapp --image=nginx --restart=Never --port=80 --labels="app=frontend"
* **If you are not given a pod file**, you may extract the definition to a file using the below command:

kubectl get pod <pod-name> -o yaml > pod.yaml

kubectl edit pod <pod-name> to edit.

* kubectl run redis –image=redis123 –dry-run=client -o yaml > pod.yaml

kubectl apply -f pod.yaml

**2) Get info pods and nodes:**

- kubectl get pods

- kubectl get nodes

- kubectl get pods -o wide

**3) Logs of pods:**

- kubectl logs webapp (see the logs)

- kubectl logs -f webapp (see the logs live)

**4) Replicaset**

- kubectl create -f replica-define.yml

- kubectl get replicaset

- kubectl delete replicaset myapp-replicaset

- kubectl replace -f replica-define.yml (replicas number changed in yaml file)

- kubectl scale replicaset - -replicas=6 myapp-replicaset

- kubectl edit replicaset myapp-replicaset (you can edit replicaset without on yaml file)

**5) Deployment**

- kubectl get all (the list of deployment and replicaset)

- kubectl create -f deployment.yml

- kubectl get deployments

- kubectl describe deployment

- kubectl apply -f deployment.yml ( Update)

- kubectl edit deployment frontend (Edit deployment name frontend)

- kubectl set image deployment//myapp-deployment nginx=nginx:1.9.1

- kubectl rollout status deployment/myapp-deployment

- kubectl rollout hoistory deployment/myapp-deployment

- kubectl rollout undo deployment/myapp-deployment