**Kubernets deployments objects**

* vi mydeploy.yml ( create YAML )

( eg - kind: Deployment

apiVersion: apps/v1

metadata:

name: mydeployments

spec:

replicas: 2

selector:

matchLabels:

name: deployment

template:

metadata:

name: testpod

labels:

name: deployment

spec:

containers:

- name: c00

image: ubuntu

command: ["/bin/bash", "-c", "while true; do echo version-1; sleep 5; done"]

* kubectl get pods ( to check how many pods we have currently )
* kubectl apply -f mydeploy.yml ( to run yml file )
* kubectl get deploy ( to check deployment was created or not )
* kubectl describe deploy deployments ( to check how deployments creates RS and pods )
* kubectl get rs ( to check rs )
* kubectl get pods ( detail of pods )
* kubectl delete pod “mydeployments- fkdafhbjk\*\*\*” ( pod can’t delete coz our desire state in yml file is 2 )
* kubectl get rs (desired state=2)
* kubectl scale --replicas=1 deploy mydeployments ( to scale up or down pods )
* kubectl get deploy ( it will delete newly created pod after scale up )
* kubectl get rs
* kubectl get pods
* kubectl logs -f “mydeployments- fkdafhbjk\*\*\*” ( to see the output )
* vi mydeploy.yml ( make some changes )

( for eg :- kind: Deployment

apiVersion: apps/v1

metadata:

name: mydeployments

spec:

replicas: 2

selector:

matchLabels:

name: deployment

template:

metadata:

name: testpod

labels:

name: deployment

spec:

containers:

- name: c00

image: centos

command: ["/bin/bash", "-c", "while true; do echo version-1.1; sleep 5; done"]

* kubectl apply -f mydeploy.yml ( to run yml file )
* kubectl get deploy ( to check deployment was created or not )
* kubectl get rs ( previous version rs will be 0 and new version will up and running with 2 pods )
* kubectl get pods
* kubectl logs -f “mydeployments- fkdafhbjk\*\*\*” ( you will get new output of version-1.1 )
* kubectl exec my deployments-7dknfr -- cat / etc/os-release ( new image of centos )
* kubectl scale --replicas=4 deploy mydeployments
* kubectl get rs
* kubectl scale --replicas=2deploy mydeployments
* kubectl rollout status deployments mydeployments ( rollback to your previous version-1 )
* kubectl rollout history deployment mydeployments ( to check the version history )
* kubectl rollout undo deploy/mydeployments ( to go previous version-1.1 )