Kubernetes:

Kubernetes is a “Container Orchestration System”

It can be use for

1-Automatic deployments

2-Distribution of loads

3-Auto scaling

4-Monitoring and health checks

POD is a smallest unit in Kubernetes

Most common use is One container per POD

Kubernetes Cluster

Master Node

Worker Node

Worker Node

Master Node

Worker Node

API Server

kubelet

Container

Runtime

Scheduler

Kube Controller Manager

Kube-proxy

Cloud Controller Manager

Worker Node

dns

etcd

kubelet

Container

Runtime

kubelet

Container

Runtime

Kube-proxy

Kube-proxy

POD

container

Node

🡪Install chocolatey

🡪Install minikube using chocolatey

Run Powershell as administrator

> Choco install minikube

> minikube version

> minikube help

🡪start minikube cluster

> minikube status

> Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Hyper-V-Tools-All -All

> minikube start - - driver=hyperv

> minikube status

>minikube ip

>minikube delete

> kubectl cluster-info

>kubectl get nodes

> kubectl get pods (by default it will list POD under “default” namespace)

>kubectl get namespaces

>kubectl get pods - -namespace=kube-system

🡪Create single POD

>kubectl run nginx - - image=nginx

> kubectl get pods

>kubectl describe pod nginx

>kubectl get pods -o wide

If “container runtime” is Docker, “/pause” container is created for each container in POD to keep the namespace of POD

🡪Creating and Exploring Deployments

>kubectl get deployments |kubectl get deploy

>kubectl create deployment nginx-deployment - - image=nginx

>kubectl describe deployment nginx-deployment

Labels, Selector, Replicas, NewReplicaSet, Events

>kubectl get pods

>kubectl describe pod nginx-deployment-6d6565499c-fl5tc(nginx-deployment-6d6565499c is ReplicaSet Name)

🡪 Upscale deployment

>kubectl scale deployment nginx-deployment - -replicas=5

🡪 Downscale deployment

>kubectl scale deployment nginx-deployment - -replicas=3

🡺 Service (ClusterIP service, NodePort Service, LoadBalancer service)

🡪Creating and Exploring ClusterIP Service

>kubectl expose deployment nginx-deployment - -port=8080 - -target-port=80

Above command will create “Cluster IP Service”.

Cluster IP Service is a service which maps virtual IP address to deployment, using which we can access deployment and its POD from cluster

>kubectl get services | kubectl get svc

>kubectl delete deployment nginx-deployment

>kubectl delete service nginx-deployment

🡪create application

🡪create Dockerfile

🡪create docker image

🡪push docker image to docker hub

🡪create deployment in kubernetes using docker image

🡪create service using expose deployment

🡪access server

🡪scale up the deployment using scale

🡪 Create NodePort Service

>kubectl expose deployment nginx-deployment - -type=NodePort - -port=80

>kubectl get services |kubectl get svc

>minikube ip

>minikube service nginx-deployment

🡪Create LoadBalancer Service

>kubectl expose deployment nginx-deployment - -type=LoadBalancer - -port=80

>kubectl get services

>minikube ip

>minikube service nginx-deployment

🡪Rolling Update of Deployment

>kubectl set image deployment nginx-deployment nginx-deployment=nginx:2.0.0

Eg: kubectl set image deployment [deployment-name] [pod-name]=[image-name-with-tag]

>kubectl rollout status deploy nginx-deployment | kubectl rollout status deploy [deployment-name]

🡪what happens when one of the PODs is deleted

>kubectl delete pod [pod-name]

Kubernetes will create new POD to match replication factor

🡪Kubernetes dashboard

>minikube dashboard

🡪Delete all

>kubectl delete all - -all

🡪 Create Deployment.yml file and apply

>kubectl apply -f “./deployment.yml”

Task 1:

Create Express Application

Create docker image of application and push it to public repository.

Create deployment and use public docker image.

Run deployment and service.

Task 2:

Update the Express Application

Create new Tag of application image and push it to docker repository

Update deployment

Re-Run deployment and service

Task 3:

Combine service and deployment file

Task 4: Connect different deployments together

Communication between 2 services

Connect to 1st service using LoadBalancer

1st service will use 2nd service name to make call

2nd service is using ClusterIP