**MINKUBE INSTALLATION**

Install ec2

Go to Google Minkube and install Ubuntu 22.04

<https://www.linuxbuzz.com/install-minikube-on-ubuntu/>

open mobex and sudo su , update, upgrade

docker install ubuntu 22.04

<https://docs.docker.com/engine/install/ubuntu/>

commands give 1 by 1 not together

sudo apt-get update

sudo apt-get install ca-certificates curl

sudo install -m 0755 -d /etc/apt/keyrings

sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc

sudo chmod a+r /etc/apt/keyrings/docker.asc

echo \

"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \

$(. /etc/os-release && echo "$VERSION\_CODENAME") stable" | \

sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

sudo apt-get update

sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

sudo docker run hello-world

docker ps

docker ps -a(we need to get HelloWorld image)

sudo usermod -aG docker $USER

newgrp docker

curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64

sudo install minikube-linux-amd64 /usr/local/bin/minikube

minikube version

curl -LO https://storage.googleapis.com/kubernetes-release/release/`curl -s https://storage.googleapis.com/kubernetes-release/release/stable.txt`/bin/linux/amd64/kubectl

chmod +x kubectl

sudo mv kubectl /usr/local/bin/

kubectl version -o yaml

minikube start --driver=docker(\*\*\*)

minikube start --driver=docker –force

minikube status

minkube get nodes(which is attached to cluster master nodes or work nodes)

kubectl describe node minikube(we can see all installation and pod cir too)

**what is their in pods?**

**The pod will have a container pod that needs to get to one particular IP address that will be within this CIDR block**

**There are 3 main elements.**

**ETCD**

**Controller**

**Scheduler**

**There are 3 namespaces**

**kube-node-lease**

**kube-system, kube-public**

kubectl get namespaces

kubectl get pods

kubectl get pods -A

kubectl get pods -n(to get particular )

kubectl get pods -n kube-node-lease

kubectl get pods -n default

kubectl create deployment nginx-web --image=nginx

kubectl get all

kubectl describe pod/nginx-web-5b757f798d-6zp5z

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

kubectl get all

**service** [**url**:service](url:service) url is defined to connect with in the kubernate claster

kubectl get pods -n default

kubectl exec -it nginx-web-5b757f798d-6zp5z -n default /bin/bash

ls

cd etc/

ls

cd nginx/

ls

cd nginx/

ls

exit

kubectl get all -A (TO check deployment)

kubectl edit deployment.apps/nginx-web -n default(no changes)

**create k8 depolyment YAML file for nginx**

**create k8 depolyment YAML file for nginx along with service.YAML**

kubectl create ns traningnavya(creating namespace)

kubectl get ns(we will namespace)

ls

mkdir nginx

cd nginx/

vi deployment.yaml(we need to give same namespace)

vi service.yaml(type:NodePort)

to apply this for kuburanatics

kubectl apply -f deployment.yaml

kubectl apply -f service.yaml -n traningnavya

kubectl get all -n traningnavya

kubectl edit service/nginx-traning-service -n traningnavya(to edit service loadbalancer to NodePort)

kubectl get all -n traningnavya

minikube addons list

minikube addons enable ingress

minikube addons enable dashboard

minikube addons metrics-server

minikube addons enable metrics-server

kubectl get all -A

kubectl get ns

kubectl get all -n kube-system

kubectl get ns

kubectl get all -n kubernetes-dashboard

kubectl edit service/kubernetes-dashboard -n kubernetes-dashboard

(change type:NodePort)

kubectl get all -n kubernetes-dashboard

minikube ip

kubectl get all -A

curl 192.168.49.2:32286(training port 1st)

ls

**porting:**

**public ip**

**first 80**

**minikube ip,port(31069)**

**ec2 public ip,ubuntu,22**