Create single t2.medium machine in aws

sudo su

**########Now install docker###############**

sudo apt update && apt -y install docker.io

**###install Kubectl###**

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

**###install Minikube####**

curl -Lo minikube https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64 && chmod +x minikube && sudo mv minikube /usr/local/bin/

**##Start minikube####**

apt install conntrack

minikube start --vm-driver=none

minikube status

vi pod1.yml \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* press i \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

kind: Pod

apiVersion: v1

metadata:

name: testpod

annotations:

description: Our first testing pod

spec:

containers:

- name: c00

image: ubuntu

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

restartPolicy: Never # Defaults to Always

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Press esc and then :wq \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

kubectl apply -f pod1.yml

Now ,  
kubectl get pods

(you will find a pod with the name testpod)

kubectl get pods -o wide

kubectl describe pod testpod

kubectl logs -f testpod (to see what is running in container)

kubectl logs -f testpod -c c00 (c00 is the container name as mentioned in the pod1.yml file )

kubectl get nodes

kubectl exec testpod -it -c c00 -- /bin/bash (to enter inside the container)

\*\*After your enter container \*\*\*

Ps

Ps -ef

exit

\*\*\*\*Delete a pod\*\*\*\*\*\*\*\*\*\*\*\*\*

Kubectl delete pod testpod

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**\*\*\*\*\*\*\*\*MULTI CONTAINER POD ENVIRONMENT (If you want to create multiple container in a single pod)\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*vi testpod3.yml \*\*\***

kind: Pod

apiVersion: v1

metadata:

name: testpod3

spec:

containers:

- name: c00

image: ubuntu

command: ["/bin/bash", "-c", "while true; do echo My first message; sleep 5 ; done"]

- name: c01

image: centos

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

\*\*then escape and :wq\*\*

Ce

kubectl apply -f testpod3.yml

Now ,  
kubectl get pods

(you will find a pod with the name testpod)

kubectl get pods -o wide

kubectl describe pod testpod3

kubectl logs -f testpod3 c00 (to see what is running in container)

kubectl logs -f testpod3 c01

kubectl get nodes

kubectl exec testpod3 -it -c c00 -- /bin/bash (to enter inside the container)

\*\*After your enter container \*\*\*

Ps

Ps -ef

exit

\*\*\*\*Delete a pod\*\*\*\*\*\*\*\*\*\*\*\*\*

Kubectl delete pod testpod3

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

LAB 3: POD ENVIRONMENT VARIABLES

vi pod3.yml

press i

kind: Pod

apiVersion: v1

metadata:

name: environments

spec:

containers:

- name: c00

image: ubuntu

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

env: # List of environment variables to be used inside the pod

- name: MYNAME

value: AKSHAT

\*\*\*\*\*\*\*\*\*\*\*now press escape and press :wq \*\*\*\*\*\*\*\*\*\*\*\*

kubectl apply -f testpod2.yml

Now ,  
kubectl get pods

(you will find all the pods in your node)

kubectl get pods -o wide

kubectl describe pod environments

kubectl logs -f environments c00 (to see what is running in container)

kubectl get nodes

kubectl exec environments -it -c c00 -- /bin/bash (to enter inside the container)

\*\*After your enter container \*\*\*

Env

Echo $MYNAME

exit