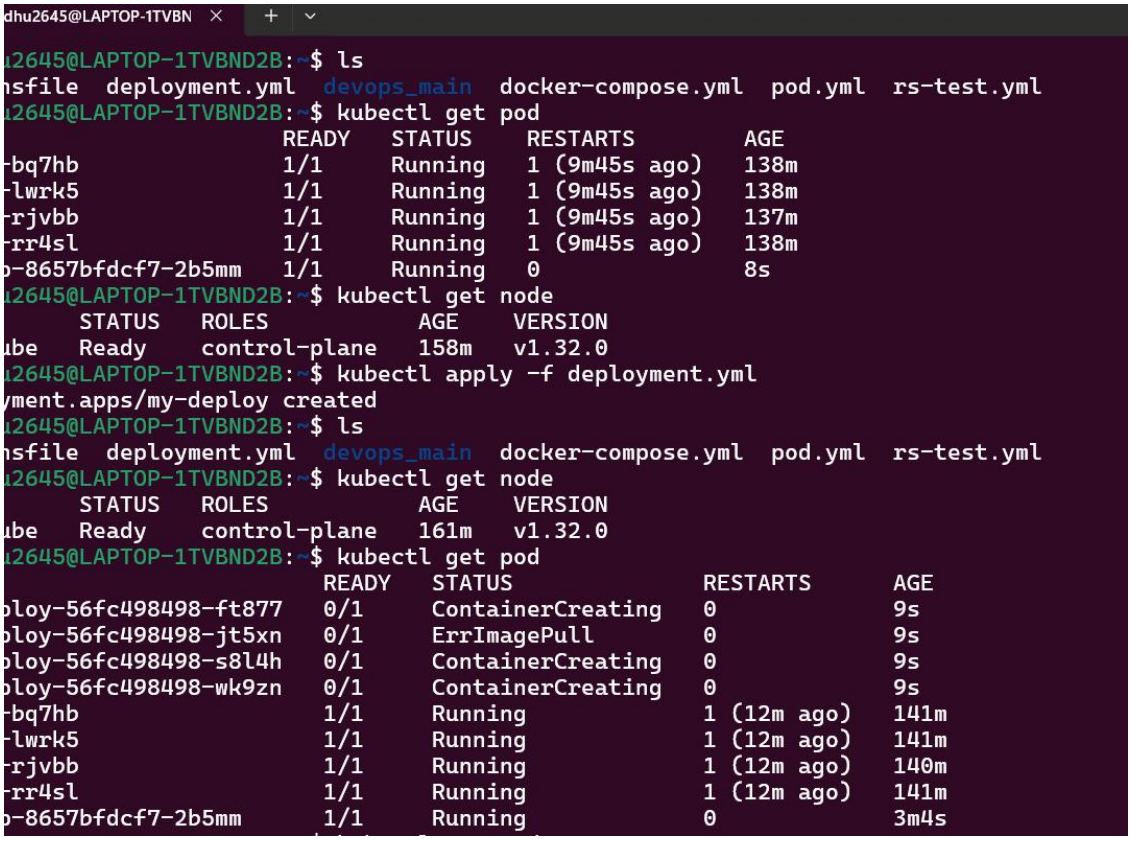
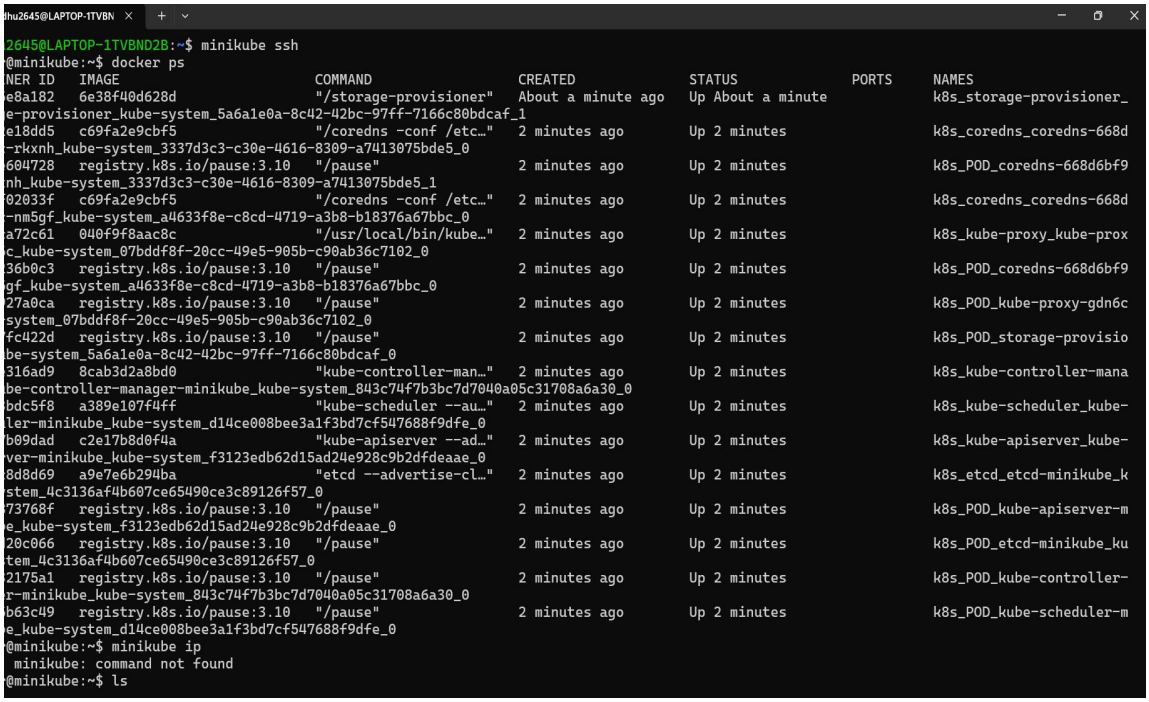
DAY-4

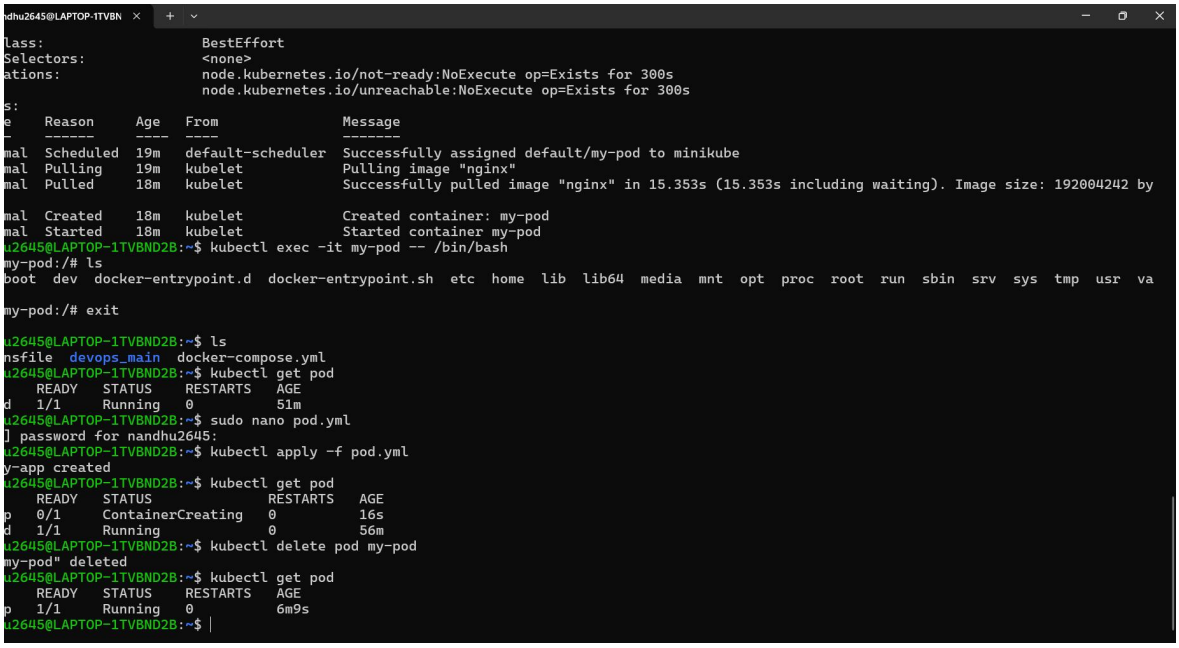
DATE: 20.03.2025

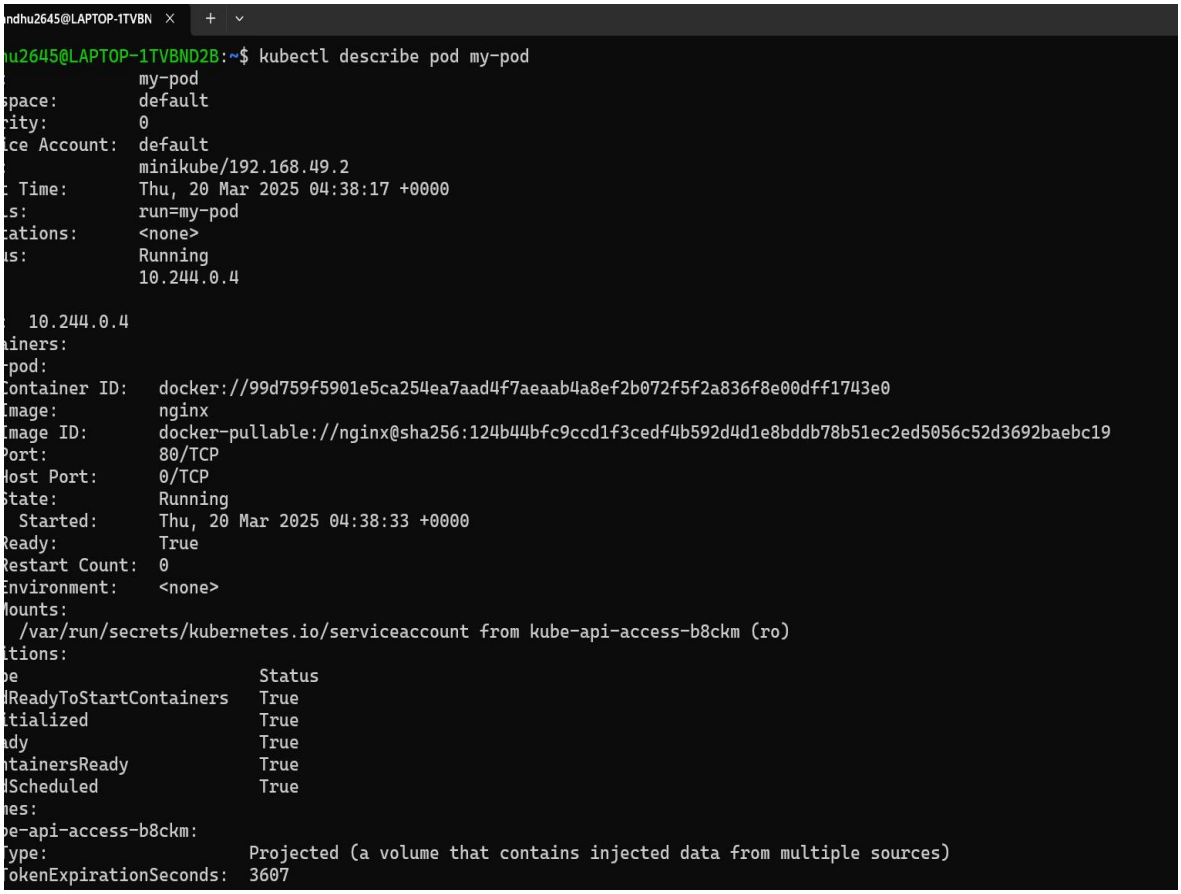


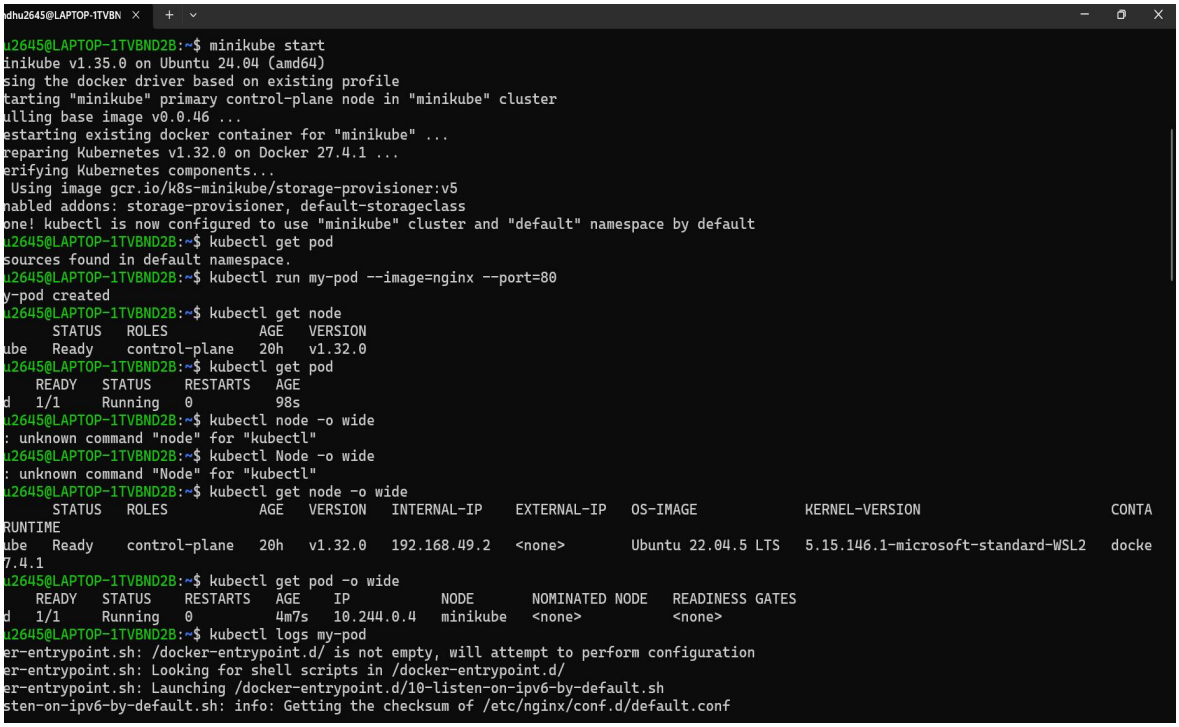
```
dhu2645@LAPTOP-1TVBN
u2645@LAPTOP-1TVBND2B:~$ ls
nsfile  deployment.yml  devops_main  docker-compose.yml  pod.yml  rs-test.yml
u2645@LAPTOP-1TVBND2B:~$ kubectl get pod
                    READY   STATUS    RESTARTS        AGE
-bq7hb              1/1     Running   1 (9m45s ago)   138m
-lwrk5              1/1     Running   1 (9m45s ago)   138m
-rjvbb              1/1     Running   1 (9m45s ago)   137m
-rr4sl              1/1     Running   1 (9m45s ago)   138m
b-8657bfdcf7-2b5mm  1/1     Running   0               8s
u2645@LAPTOP-1TVBND2B:~$ kubectl get node
      STATUS   ROLES           AGE    VERSION
ube   Ready    control-plane   158m   v1.32.0
u2645@LAPTOP-1TVBND2B:~$ kubectl apply -f deployment.yml
yment.apps/my-deploy created
u2645@LAPTOP-1TVBND2B:~$ ls
nsfile  deployment.yml  devops_main  docker-compose.yml  pod.yml  rs-test.yml
u2645@LAPTOP-1TVBND2B:~$ kubectl get node
      STATUS   ROLES           AGE    VERSION
ube   Ready    control-plane   161m   v1.32.0
u2645@LAPTOP-1TVBND2B:~$ kubectl get pod
                    READY   STATUS              RESTARTS      AGE
oloy-56fc498498-ft877   0/1     ContainerCreating   0             9s
oloy-56fc498498-jt5xn   0/1     ErrImagePull        0             9s
oloy-56fc498498-s8l4h   0/1     ContainerCreating   0             9s
oloy-56fc498498-wk9zn   0/1     ContainerCreating   0             9s
-bq7hb                  1/1     Running             1 (12m ago)   141m
-lwrk5                  1/1     Running             1 (12m ago)   141m
-rjvbb                  1/1     Running             1 (12m ago)   140m
-rr4sl                  1/1     Running             1 (12m ago)   141m
b-8657bfdcf7-2b5mm      1/1     Running             0             3m4s
```



```
dhu2645@LAPTOP-1TVBN
2645@LAPTOP-1TVBND2B:~$ minikube ssh
@minikube:~$ docker ps
NER ID   IMAGE                  COMMAND                  CREATED              STATUS              PORTS     NAMES
e8a182   6e38f40d628d           "/storage-provisioner"   About a minute ago   Up About a minute             k8s_storage-provisioner_
e-provisioner_kube-system_5a6a1e0a-8c42-42bc-97ff-7166c80bdcaf_1
e18dd5   c69fa2e9cbf5           "/coredns -conf /etc…"   2 minutes ago        Up 2 minutes                  k8s_coredns_coredns-668d
-rkxnh_kube-system_3337d3c3-c30e-4616-8309-a7413075bde5_0
604728   registry.k8s.io/pause:3.10   "/pause"           2 minutes ago        Up 2 minutes                  k8s_POD_coredns-668d6bf9
nh_kube-system_3337d3c3-c30e-4616-8309-a7413075bde5_1
02033f   c69fa2e9cbf5           "/coredns -conf /etc…"   2 minutes ago        Up 2 minutes                  k8s_coredns_coredns-668d
-nm5gf_kube-system_a4633f8e-c8cd-4719-a3b8-b18376a67bbc_0
a72c61   040f9f8aac8c           "/usr/local/bin/kube…"   2 minutes ago        Up 2 minutes                  k8s_kube-proxy_kube-prox
c_kube-system_07bddf8f-20cc-49e5-905b-c90ab36c7102_0
36b0c3   registry.k8s.io/pause:3.10   "/pause"           2 minutes ago        Up 2 minutes                  k8s_POD_coredns-668d6bf9
gf_kube-system_a4633f8e-c8cd-4719-a3b8-b18376a67bbc_0
27a0ca   registry.k8s.io/pause:3.10   "/pause"           2 minutes ago        Up 2 minutes                  k8s_POD_kube-proxy-gdn6c
system_07bddf8f-20cc-49e5-905b-c90ab36c7102_0
fc422d   registry.k8s.io/pause:3.10   "/pause"           2 minutes ago        Up 2 minutes                  k8s_POD_storage-provisio
be-system_5a6a1e0a-8c42-42bc-97ff-7166c80bdcaf_0
316ad9   8cab3d2a8bd0           "kube-controller-man…"   2 minutes ago        Up 2 minutes                  k8s_kube-controller-mana
be-controller-manager-minikube_kube-system_843c74f7b3bc7d7040a05c31708a6a30_0
bdc5f8   a389e107f4ff           "kube-scheduler --au…"   2 minutes ago        Up 2 minutes                  k8s_kube-scheduler_kube-
ler-minikube_kube-system_d14ce008bee3a1f3bd7cf547688f9dfe_0
b09dad   c2e17b8d0f4a           "kube-apiserver --ad…"   2 minutes ago        Up 2 minutes                  k8s_kube-apiserver_kube-
ver-minikube_kube-system_f3123edb62d15ad24e928c9b2dfdeaae_0
8d8d69   a9e7e6b294ba           "etcd --advertise-cl…"   2 minutes ago        Up 2 minutes                  k8s_etcd_etcd-minikube_k
stem_4c3136af4b607ce65490ce3c89126f57_0
73768f   registry.k8s.io/pause:3.10   "/pause"           2 minutes ago        Up 2 minutes                  k8s_POD_kube-apiserver-m
e_kube-system_f3123edb62d15ad24e928c9b2dfdeaae_0
20c066   registry.k8s.io/pause:3.10   "/pause"           2 minutes ago        Up 2 minutes                  k8s_POD_etcd-minikube_ku
tem_4c3136af4b607ce65490ce3c89126f57_0
2175a1   registry.k8s.io/pause:3.10   "/pause"           2 minutes ago        Up 2 minutes                  k8s_POD_kube-controller-
r-minikube_kube-system_843c74f7b3bc7d7040a05c31708a6a30_0
b63c49   registry.k8s.io/pause:3.10   "/pause"           2 minutes ago        Up 2 minutes                  k8s_POD_kube-scheduler-m
e_kube-system_d14ce008bee3a1f3bd7cf547688f9dfe_0
@minikube:~$ minikube ip
 minikube: command not found
@minikube:~$ ls
```


lass: BestEffort
Selectors: <none>
ations: node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
s:
e Reason Age From Message
mal Scheduled 19m default-scheduler Successfully assigned default/my-pod to minikube
mal Pulling 19m kubelet Pulling image "nginx"
mal Pulled 18m kubelet Successfully pulled image "nginx" in 15.353s (15.353s including waiting). Image size: 192004242 by
mal Created 18m kubelet Created container: my-pod
mal Started 18m kubelet Started container my-pod
u2645@LAPTOP-1TVBND2B:~$ kubectl exec -it my-pod -- /bin/bash
my-pod:/# ls
boot dev docker-entrypoint.d docker-entrypoint.sh etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr va
my-pod:/# exit
u2645@LAPTOP-1TVBND2B:~$ ls
nsfile devops_main docker-compose.yml
u2645@LAPTOP-1TVBND2B:~$ kubectl get pod
READY STATUS RESTARTS AGE
d 1/1 Running 0 51m
u2645@LAPTOP-1TVBND2B:~$ sudo nano pod.yml
] password for nandhu2645:
u2645@LAPTOP-1TVBND2B:~$ kubectl apply -f pod.yml
y-app created
u2645@LAPTOP-1TVBND2B:~$ kubectl get pod
READY STATUS RESTARTS AGE
p 0/1 ContainerCreating 0 16s
d 1/1 Running 0 56m
u2645@LAPTOP-1TVBND2B:~$ kubectl delete pod my-pod
my-pod" deleted
u2645@LAPTOP-1TVBND2B:~$ kubectl get pod
READY STATUS RESTARTS AGE
p 1/1 Running 0 6m9s
u2645@LAPTOP-1TVBND2B:~$


u2645@LAPTOP-1TVBND2B:~$ kubectl describe pod my-pod
my-pod
space: default
rity: 0
ice Account: default
minikube/192.168.49.2
t Time: Thu, 20 Mar 2025 04:38:17 +0000
ls: run=my-pod
tations: <none>
us: Running
10.244.0.4
10.244.0.4
ainers:
pod:
Container ID: docker://99d759f5901e5ca254ea7aad4f7aeaab4a8ef2b072f5f2a836f8e00dff1743e0
Image: nginx
Image ID: docker-pullable://nginx@sha256:124b44bfc9ccd1f3cedf4b592d4d1e8bddb78b51ec2ed5056c52d3692baebc19
Port: 80/TCP
Host Port: 0/TCP
State: Running
Started: Thu, 20 Mar 2025 04:38:33 +0000
Ready: True
Restart Count: 0
Environment: <none>
Mounts:
/var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-b8ckm (ro)
itions:
pe Status
dReadyToStartContainers True
tialized True
ady True
ntainersReady True
dScheduled True
es:
be-api-access-b8ckm:
Type: Projected (a volume that contains injected data from multiple sources)
TokenExpirationSeconds: 3607



```
u2645@LAPTOP-1TVBND2B:~$ minikube start
inikube v1.35.0 on Ubuntu 24.04 (amd64)
sing the docker driver based on existing profile
tarting "minikube" primary control-plane node in "minikube" cluster
ulling base image v0.0.46 ...
estarting existing docker container for "minikube" ...
reparing Kubernetes v1.32.0 on Docker 27.4.1 ...
erifying Kubernetes components...
 Using image gcr.io/k8s-minikube/storage-provisioner:v5
nabled addons: storage-provisioner, default-storageclass
one! kubectl is now configured to use "minikube" cluster and "default" namespace by default
u2645@LAPTOP-1TVBND2B:~$ kubectl get pod
sources found in default namespace.
u2645@LAPTOP-1TVBND2B:~$ kubectl run my-pod --image=nginx --port=80
y-pod created
u2645@LAPTOP-1TVBND2B:~$ kubectl get node
      STATUS   ROLES           AGE   VERSION
ube   Ready    control-plane   20h   v1.32.0
u2645@LAPTOP-1TVBND2B:~$ kubectl get pod
    READY   STATUS    RESTARTS   AGE
d   1/1     Running   0          98s
u2645@LAPTOP-1TVBND2B:~$ kubectl node -o wide
: unknown command "node" for "kubectl"
u2645@LAPTOP-1TVBND2B:~$ kubectl Node -o wide
: unknown command "Node" for "kubectl"
u2645@LAPTOP-1TVBND2B:~$ kubectl get node -o wide
      STATUS   ROLES           AGE   VERSION   INTERNAL-IP    EXTERNAL-IP   OS-IMAGE             KERNEL-VERSION                      CONTA
RUNTIME
ube   Ready    control-plane   20h   v1.32.0   192.168.49.2   <none>        Ubuntu 22.04.5 LTS   5.15.146.1-microsoft-standard-WSL2   docke
7.4.1
u2645@LAPTOP-1TVBND2B:~$ kubectl get pod -o wide
    READY   STATUS    RESTARTS   AGE    IP           NODE       NOMINATED NODE   READINESS GATES
d   1/1     Running   0          4m7s   10.244.0.4   minikube   <none>           <none>
u2645@LAPTOP-1TVBND2B:~$ kubectl logs my-pod
er-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
er-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
er-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
sten-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
```

```
2645@LAPTOP-1TVBND2B:~$ kubectl scale deploy my-deploy --replicas=1
ment.apps/my-deploy scaled
2645@LAPTOP-1TVBND2B:~$ kubectl get pod
                      READY   STATUS             RESTARTS      AGE
loy-56fc498498-s8l4h  0/1     ImagePullBackOff   0             7m44s
bq7hb                 1/1     Running            1 (20m ago)   149m
lwrk5                 1/1     Running            1 (20m ago)   149m
rjvbb                 1/1     Running            1 (20m ago)   147m
rr4sl                 1/1     Running            1 (20m ago)   149m
-8657bfdcf7-2b5mm     1/1     Running            0             10m
2645@LAPTOP-1TVBND2B:~$ kubectl delete deploy my-deploy
ment.apps "my-deploy" deleted
2645@LAPTOP-1TVBND2B:~$ kubectl get pod
                    READY   STATUS    RESTARTS      AGE
bq7hb               1/1     Running   1 (20m ago)   149m
lwrk5               1/1     Running   1 (20m ago)   149m
rjvbb               1/1     Running   1 (20m ago)   148m
rr4sl               1/1     Running   1 (20m ago)   149m
-8657bfdcf7-2b5mm   1/1     Running   0             11m
2645@LAPTOP-1TVBND2B:~$ kubectl delete rs all --all
 name cannot be provided when a selector is specified
2645@LAPTOP-1TVBND2B:~$ kubectl delete rs-test all --all
 name cannot be provided when a selector is specified
2645@LAPTOP-1TVBND2B:~$ kubectl delete all --all
y-rs-bq7hb" deleted
y-rs-lwrk5" deleted
y-rs-rjvbb" deleted
y-rs-rr4sl" deleted
ebapp-8657bfdcf7-2b5mm" deleted
e "kubernetes" deleted
ment.apps "webapp" deleted
aset.apps "my-rs" deleted
2645@LAPTOP-1TVBND2B:~$ kubectl get pod
ources found in default namespace.
2645@LAPTOP-1TVBND2B:~$
```

COMMANDS:

1.MINIKUBE COMMANDS:

minikube start

minikube status

kubectl get pod

kubectl run my_pod --image=nginx --port=80

kubectl get node

kubectl get pod

kubectl get node -o wide

kubectl get pod -o wide

kubectl logs my-pod

kubectl describe pod my-pod

kubectl exec -it my-pod -- /bin/bash

/usr/local/tomcat#ls

cd webapps

ls

exit

sudo nano pod.yml

then paste the command in grp

kubectl apply -f pod.yml

kubectl get pod

kubectl delete pod my-pod

minikube ssh

docker ps

minikube ip

kubectl get rs

kubectl get pod

kubectl create deployment web-nginx --image=nginx --replicas=1

kubectl get deploy

kubectl get pod

kubectl delete deployment web-nginx

kubectl get pod

kubectl delete pod my-app

kubectl get pod

Replica Set:

sudo nano rs-test.yml

past from grp

kubectl apply -f rs-test.yml

kubectl get rs

kubectl get pod

kubectl delete pod my-rs-jclds // even if one pod is deleted other pod is automatically created

kubectl get pod

2. Create Deployment by executing above YAML file

$ kubectl create -f web-deploy.yml

Do necessary modifications if exist, else create new

$ kubectl create -f web-deploy.yml

Completely Modify Pod Template

$ kubectl replace –f web-deploy.yml

3. View Deployments

$ kubectl get deployments

$ kubectl get deploy

$ kubectl get deploy -o wide

$ kubectl get deploy <deployment-name> -o json

$ kubectl get deploy <deployment-name> -o yaml

4. View Deployment Description

$ kubectl describe deploy <deployment-name>

5. We can modify generated/updated YAML file

$ kubectl edit deploy <deployment-name>

change replicas: count to any other value then (ESC):wq

# We can modify our YAML file and then execute apply command
$ kubectl apply -f web-deploy.yml

We can Even scale using command also
$ kubectl scale deploy <deployment-name> --replicas=<desired-replica-count>

6. Delete Deployment

$ kubectl delete deploy <deployment-name>
$ kubectl delete -f web-deploy.yml

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-deploy
  labels:
    name: my-deploy
spec:
  replicas: 1
  selector:
    matchLabels:
      apptype: web-backend
  strategy:
    type: RollingUpdate
  template:
    metadata:
      labels:
        apptype: web-backend
    spec:
      containers:
      - name: my-app
        image:
        ports:
        - containerPort: 9000
```

7. Create ReplicaSet by executing above YAML file

$ kubectl create -f rs-test.yml

Do necessary modifications if exist, else create new

$ kubectl apply -f rs-test.yml

Completely Modify Pod Template

$ kubectl replace –f rs-test.yml

8. View ReplicaSets

$ kubectl get replicasets

$ kubectl get rs

$ kubectl get rs –o wide

$ kubectl get rs <replica-set-name> –o json

$ kubectl get rs <replica-set-name> –o yaml

9. View ReplicaSet Description

$ kubectl describe rs <replica-set-name>

10. We can modify generated/updated YAML file

$ kubectl edit rs <replica-set-name>

change replicas: count to any other value then (ESC):wq

We can modify our YAML file and then execute apply command

$ kubectl apply -f rs-test.yml

We can Even scale using command also

$ kubectl scale replicaset <replicaset-name> --replicas=<desired-replica-count>

11. Delete ReplicaSet

$ kubectl delete rs <replica-set-name>

$ kubectl delete -f rs-test.yml

12.Services (short name = svc):

Service is an abstraction that defines a logical set of pods and a policy to access them. Services enable network connectivity and load balancing to the pods that are part of

the service, allowing other components within or outside the cluster to interact with the application.

Service Types: Kubernetes supports different types of services:

1. NodePort: Exposes the service on a static port on each selected node's IP. This type makes the service accessible from outside the cluster by the <NodeIP>:<NodePort> combination.

2. ClusterIP: Exposes the service on a cluster-internal IP. This type makes the service only reachable within the cluster.

3. LoadBalancer: Creates an external load balancer in cloud environments, which routes traffic to the service.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-deploy
  labels:
    name: my-deploy
spec:
  replicas: 1
  selector:
    matchLabels:
      apptype: web-backend
  strategy:
    type: RollingUpdate
  template:
    metadata:
      labels:
        apptype: web-backend
    spec:
      containers:
      - name: my-app
```

```
        image:
        ports:
        - containerPort: 9000

---

apiVersion: v1
kind: Service
metadata:
  name: my-service
  labels:
    app: my-service
spec:
  type: NodePort
  ports:
    - port: 9000
      targetPort: 8080
      nodePort: 30002
  selector:
    apptype: web-backend
```

13. Namespace (short name = ns):

namespace is a virtual cluster or logical partition within a cluster that provides a way to organize and isolate resources.

It allows multiple teams or projects to share the same physical cluster while maintaining resource separation and access control.

```
# To create a namespace:
$ kubectl create namespace <namespace-name>
$ kubectl create ns my-bank
# To switch to a specific namespace: (make this as default type)
$ kubectl config set-context --current --namespace=<namespace-name>
# To list all namespaces:
$ kubectl get namespaces
```

To get resources within a specific namespace:

$ kubectl get <resource-type> -n <namespace-name>

$ kubectl get deploy -n my-bank

$ kubectl get deploy --namespace my-bank

$ kubectl get all --namespace my-bank

To delete a namespace and all associated resources:

$ kubectl delete namespace <namespace-name>

$ kubectl delete ns my-bank