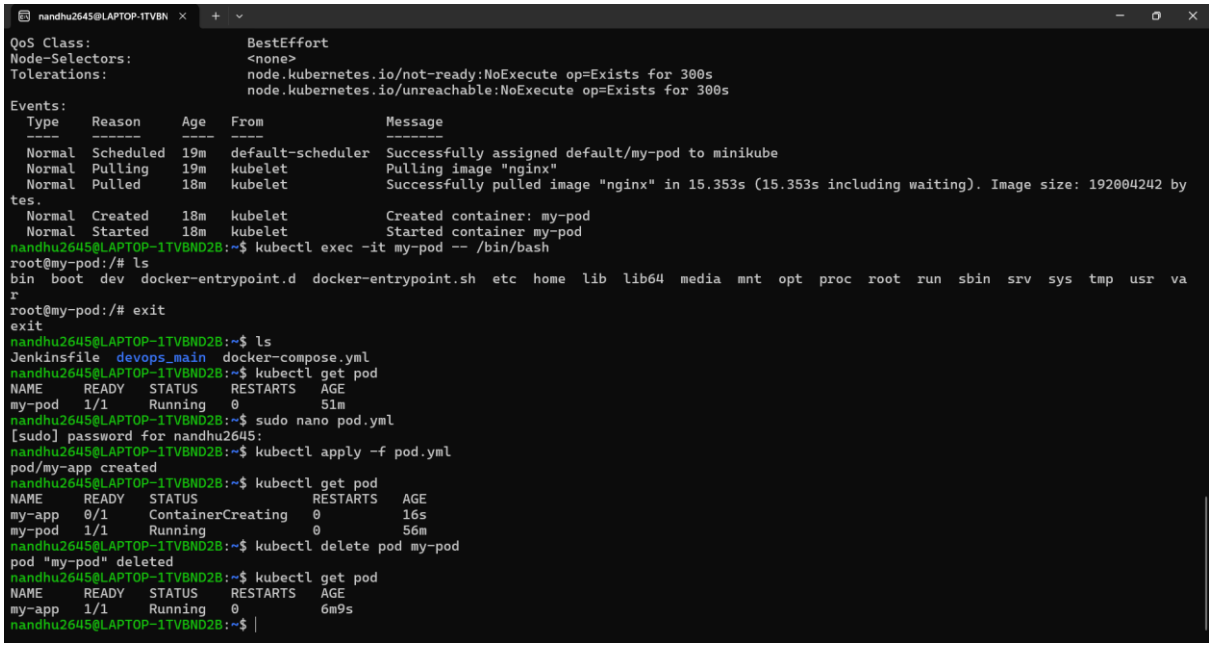
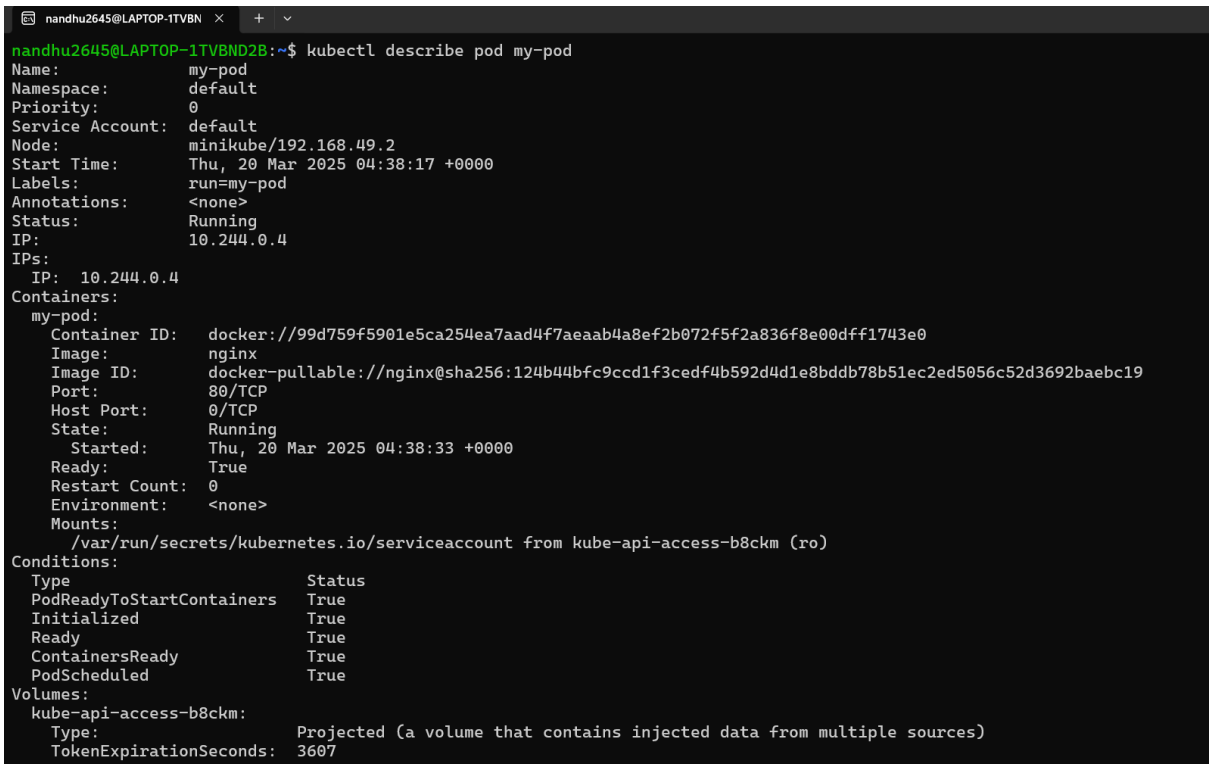
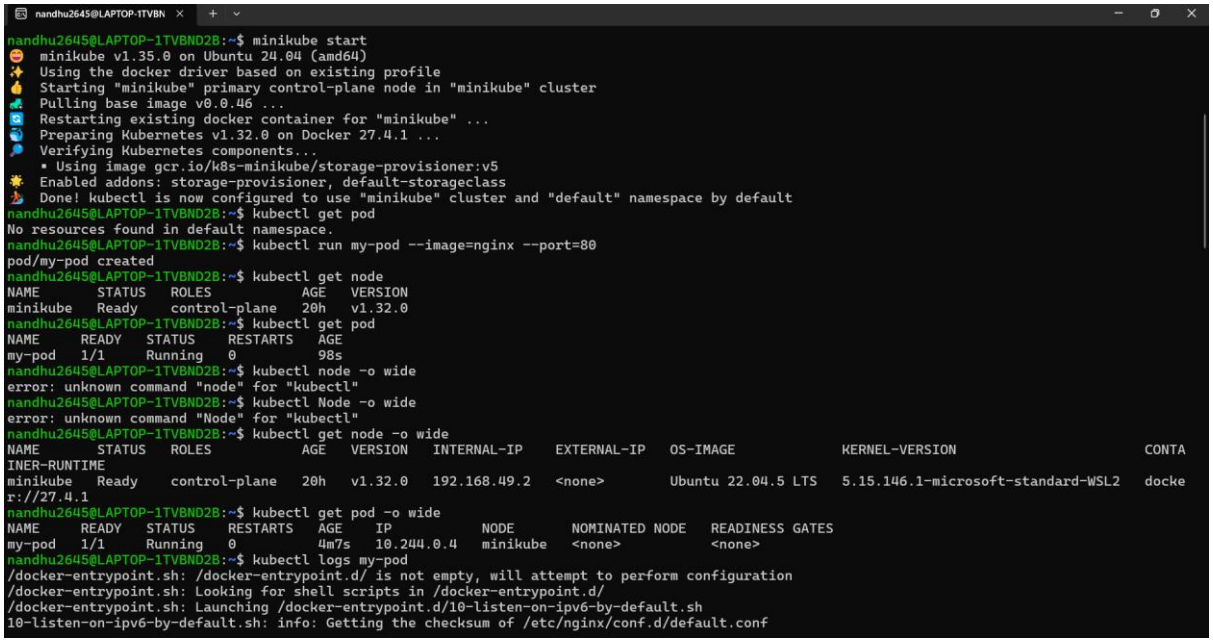
```
nandhu2645@LAPTOP-1TVBN
nandhu2645@LAPTOP-1TVBND2B:~$ ls
Jenkinsfile  deployment.yml  devops_main  docker-compose.yml  pod.yml  rs-test.yml
nandhu2645@LAPTOP-1TVBND2B:~$ kubectl get pod
NAME                      READY   STATUS    RESTARTS        AGE
my-rs-bq7hb               1/1     Running   1 (9m45s ago)   138m
my-rs-lwrk5               1/1     Running   1 (9m45s ago)   138m
my-rs-rjvbb               1/1     Running   1 (9m45s ago)   137m
my-rs-rr4sl               1/1     Running   1 (9m45s ago)   138m
webapp-8657bfdcf7-2b5mm   1/1     Running   0               8s
nandhu2645@LAPTOP-1TVBND2B:~$ kubectl get node
NAME       STATUS   ROLES           AGE    VERSION
minikube   Ready    control-plane   158m   v1.32.0
nandhu2645@LAPTOP-1TVBND2B:~$ kubectl apply -f deployment.yml
deployment.apps/my-deploy created
nandhu2645@LAPTOP-1TVBND2B:~$ ls
Jenkinsfile  deployment.yml  devops_main  docker-compose.yml  pod.yml  rs-test.yml
nandhu2645@LAPTOP-1TVBND2B:~$ kubectl get node
NAME       STATUS   ROLES           AGE    VERSION
minikube   Ready    control-plane   161m   v1.32.0
nandhu2645@LAPTOP-1TVBND2B:~$ kubectl get pod
NAME                         READY   STATUS              RESTARTS       AGE
my-deploy-56fc498498-ft877   0/1     ContainerCreating   0              9s
my-deploy-56fc498498-jt5xn   0/1     ErrImagePull        0              9s
my-deploy-56fc498498-s8l4h   0/1     ContainerCreating   0              9s
my-deploy-56fc498498-wk9zn   0/1     ContainerCreating   0              9s
my-rs-bq7hb                  1/1     Running             1 (12m ago)    141m
my-rs-lwrk5                  1/1     Running             1 (12m ago)    141m
my-rs-rjvbb                  1/1     Running             1 (12m ago)    140m
my-rs-rr4sl                  1/1     Running             1 (12m ago)    141m
webapp-8657bfdcf7-2b5mm      1/1     Running             0              3m4s
```

```
nandhu2645@LAPTOP-1TVBN
nandhu2645@LAPTOP-1TVBND2B:~$ minikube ssh
docker@minikube:~$ docker ps
CONTAINER ID   IMAGE                     COMMAND                  CREATED              STATUS              PORTS     NAMES
451f35e8a182   6e38f40d628d              "/storage-provisioner"   About a minute ago   Up About a minute             k8s_storage-provisioner_
storage-provisioner_kube-system_5a6a1e0a-8c42-42bc-97ff-7166c80bdcaf_1
4ab8b2e18dd5   c69fa2e9cbf5              "/coredns -conf /etc…"   2 minutes ago        Up 2 minutes                  k8s_coredns_coredns-668d
6bf9bc-rkxnh_kube-system_3337d3c3-c30e-4616-8309-a7413075bde5_0
9adace604728   registry.k8s.io/pause:3.10   "/pause"                 2 minutes ago        Up 2 minutes                  k8s_POD_coredns-668d6bf9
bc-rkxnh_kube-system_3337d3c3-c30e-4616-8309-a7413075bde5_1
8d16bf02033f   c69fa2e9cbf5              "/coredns -conf /etc…"   2 minutes ago        Up 2 minutes                  k8s_coredns_coredns-668d
6bf9bc-nm5gf_kube-system_a4633f8e-c8cd-4719-a3b8-b18376a67bbc_0
8f926ca72c61   040f9f8aac8c              "/usr/local/bin/kube…"   2 minutes ago        Up 2 minutes                  k8s_kube-proxy_kube-prox
y-gdn6c_kube-system_07bddf8f-20cc-49e5-905b-c90ab36c7102_0
c59f0236b0c3   registry.k8s.io/pause:3.10   "/pause"                 2 minutes ago        Up 2 minutes                  k8s_POD_coredns-668d6bf9
bc-nm5gf_kube-system_a4633f8e-c8cd-4719-a3b8-b18376a67bbc_0
3294a927a0ca   registry.k8s.io/pause:3.10   "/pause"                 2 minutes ago        Up 2 minutes                  k8s_POD_kube-proxy-gdn6c
_kube-system_07bddf8f-20cc-49e5-905b-c90ab36c7102_0
4a2b87fc422d   registry.k8s.io/pause:3.10   "/pause"                 2 minutes ago        Up 2 minutes                  k8s_POD_storage-provisio
ner_kube-system_5a6a1e0a-8c42-42bc-97ff-7166c80bdcaf_0
395a0e316ad9   8cab3d2a8bd0              "kube-controller-man…"   2 minutes ago        Up 2 minutes                  k8s_kube-controller-mana
ger_kube-controller-manager-minikube_kube-system_843c74f7b3bc7d7040a05c31708a6a30_0
527983bdc5f8   a389e107f4ff              "kube-scheduler --au…"   2 minutes ago        Up 2 minutes                  k8s_kube-scheduler_kube-
scheduler-minikube_kube-system_d14ce008bee3a1f3bd7cf547688f9dfe_0
d93617b09dad   c2e17b8d0f4a              "kube-apiserver --ad…"   2 minutes ago        Up 2 minutes                  k8s_kube-apiserver_kube-
apiserver-minikube_kube-system_f3123edb62d15ad24e928c9b2dfdeaae_0
bc6bac8d8d69   a9e7e6b294ba              "etcd --advertise-cl…"   2 minutes ago        Up 2 minutes                  k8s_etcd_etcd-minikube_k
ube-system_4c3136af4b607ce65490ce3c89126f57_0
cf613873768f   registry.k8s.io/pause:3.10   "/pause"                 2 minutes ago        Up 2 minutes                  k8s_POD_kube-apiserver-m
inikube_kube-system_f3123edb62d15ad24e928c9b2dfdeaae_0
abfb9d20c066   registry.k8s.io/pause:3.10   "/pause"                 2 minutes ago        Up 2 minutes                  k8s_POD_etcd-minikube_ku
be-system_4c3136af4b607ce65490ce3c89126f57_0
8c6c832175a1   registry.k8s.io/pause:3.10   "/pause"                 2 minutes ago        Up 2 minutes                  k8s_POD_kube-controller-
manager-minikube_kube-system_843c74f7b3bc7d7040a05c31708a6a30_0
6d1dd6b63c49   registry.k8s.io/pause:3.10   "/pause"                 2 minutes ago        Up 2 minutes                  k8s_POD_kube-scheduler-m
inikube_kube-system_d14ce008bee3a1f3bd7cf547688f9dfe_0
docker@minikube:~$ minikube ip
-bash: minikube: command not found
docker@minikube:~$ ls
```


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


nandhu2645@LAPTOP-1TVBN
nandhu2645@LAPTOP-1TVBND2B:~$ kubectl describe pod my-pod
Name: my-pod
Namespace: default
Priority: 0
Service Account: default
Node: minikube/192.168.49.2
Start Time: Thu, 20 Mar 2025 04:38:17 +0000
Labels: run=my-pod
Annotations: <none>
Status: Running
IP: 10.244.0.4
IPs:
IP: 10.244.0.4
Containers:
my-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)
Conditions:
Type Status
PodReadyToStartContainers True
Initialized True
Ready True
ContainersReady True
PodScheduled True
Volumes:
kube-api-access-b8ckm:
Type: Projected (a volume that contains injected data from multiple sources)
TokenExpirationSeconds: 3607




```
nandhu2645@LAPTOP-1TVBND2B:~$ kubectl scale deploy my-deploy --replicas=1
deployment.apps/my-deploy scaled
nandhu2645@LAPTOP-1TVBND2B:~$ kubectl get pod
NAME                         READY   STATUS             RESTARTS      AGE
my-deploy-56fc498498-s8l4h   0/1     ImagePullBackOff   0             7m44s
my-rs-bq7hb                  1/1     Running            1 (20m ago)   149m
my-rs-lwrk5                  1/1     Running            1 (20m ago)   149m
my-rs-rjvbb                  1/1     Running            1 (20m ago)   147m
my-rs-rr4sl                  1/1     Running            1 (20m ago)   149m
webapp-8657bfdcf7-2b5mm      1/1     Running            0             10m
nandhu2645@LAPTOP-1TVBND2B:~$ kubectl delete deploy my-deploy
deployment.apps "my-deploy" deleted
nandhu2645@LAPTOP-1TVBND2B:~$ kubectl get pod
NAME                         READY   STATUS    RESTARTS      AGE
my-rs-bq7hb                  1/1     Running   1 (20m ago)   149m
my-rs-lwrk5                  1/1     Running   1 (20m ago)   149m
my-rs-rjvbb                  1/1     Running   1 (20m ago)   148m
my-rs-rr4sl                  1/1     Running   1 (20m ago)   149m
webapp-8657bfdcf7-2b5mm      1/1     Running   0             11m
nandhu2645@LAPTOP-1TVBND2B:~$ kubectl delete rs all --all
error: name cannot be provided when a selector is specified
nandhu2645@LAPTOP-1TVBND2B:~$ kubectl delete rs-test all --all
error: name cannot be provided when a selector is specified
nandhu2645@LAPTOP-1TVBND2B:~$ kubectl delete all --all
pod "my-rs-bq7hb" deleted
pod "my-rs-lwrk5" deleted
pod "my-rs-rjvbb" deleted
pod "my-rs-rr4sl" deleted
pod "webapp-8657bfdcf7-2b5mm" deleted
service "kubernetes" deleted
deployment.apps "webapp" deleted
replicaset.apps "my-rs" deleted
nandhu2645@LAPTOP-1TVBND2B:~$ kubectl get pod
No resources found in default namespace.
nandhu2645@LAPTOP-1TVBND2B:~$
```

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
```

2. 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

3. 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

4. View ReplicaSet Description

$ kubectl describe rs <replica-set-name>

5. 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>

6. Delete ReplicaSet

$ kubectl delete rs <replica-set-name>

$ kubectl delete -f rs-test.yml

4.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
```

5.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