

DevOps
Day – 4
Assignment

Name: Debehaa J

Roll No: 22CSR037

1. Namespace

A Namespace is a logical partition within a Kubernetes cluster.

It allows you to divide resources like Pods, Services, and Deployments into different environments (e.g., dev, test, prod).

Think of it as a virtual cluster inside the physical Kubernetes cluster.

Useful for managing large projects with multiple teams.

2. Replica

A Replica ensures that a specified number of identical Pods are running in your cluster.

If a Pod fails, Kubernetes automatically replaces it using replicas to maintain the desired state.

It provides scalability and fault tolerance.

3. Pod

A **Pod** is the smallest deployable unit in Kubernetes.

It contains one or more containers (e.g., Docker containers).

Containers in a Pod share the same network namespace, storage, and lifecycle.

4. Deployment

A Deployment manages the creation and scaling of Pods using ReplicaSets.

It provides automated rollouts and rollbacks.

It ensures your application is always available by managing its state.

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
```

```
kubectl create ns mydeploy
```

```
kubectl apply -f deploy.yml -n mydeploy
```

```
apiVersion: v1
```

```
kind: Namespace
```

```
metadata:
```

```
  name: my-demo-ns
```

```
apiVersion: v1
```

```
kind: Pod
```

```
metadata:
```

```
  name: my-pod
```

```
  namespace: my-demo-ns
```

```
spec:
```

```
  containers:
```

```
    - name: my-container
```

```
      image: nginx:latest
```

```
akashine@Advik: ~$ kubectl get pod
NAME                                STATUS    RESTARTS   AGE
my-app                              1/1      Running    0          44s
nginx-5869d7778c-tpg4r             1/1      Running    0          44s
akashine@Advik:~$
```

Conditions:

Type	Status
PodReadyToStartContainers	True
Initialized	True
Ready	True
ContainersReady	True
PodScheduled	True

Volumes:

kube-api-access-tmbvj:

- Type: Projected (a volume that contains injected data from multiple sources)
- TokenExpirationSeconds: 3607
- ConfigMapName: kube-root-ca.crt
- ConfigMapOptional: <nil>
- DownwardAPI: true

QoS Class: BestEffort

Node-Selectors: <none>

Tolerations:

- node.kubernetes.io/not-ready:NoExecute opExists for 300s
- node.kubernetes.io/unreachable:NoExecute opExists for 300s

Events:

Type	Reason	Age	From	Message
Normal	Scheduled	3m1s	default-scheduler	Successfully assigned default/my-app to minikube
Normal	Pulling	3m	kubelet	Pulling image "nginx"
Normal	Pulled	2m56s	kubelet	Successfully pulled image "nginx" in 4.224s (4.224s including waiting). Image size: 19.2804242 bytes.
Normal	Created	2m56s	kubelet	Created container: my-app-container
Normal	Started	2m56s	kubelet	Started container my-app-container

```
akashine@Advik:~$ kubectl get pod
NAME                                STATUS    RESTARTS   AGE
my-app                              1/1      Running    0          44s
nginx-5869d7778c-tpg4r             1/1      Running    0          44s
akashine@Advik:~$
```

```
akashine@Advik: ~$ kubectl get pod
NAME                                STATUS    RESTARTS   AGE
my-app                              1/1      Running    0          44s
nginx-5869d7778c-tpg4r             1/1      Running    0          44s
akashine@Advik:~$
```

Conditions:

Type	Status
PodReadyToStartContainers	True
Initialized	True
Ready	True
ContainersReady	True
PodScheduled	True

Volumes:

kube-api-access-tmbvj:

- Type: Projected (a volume that contains injected data from multiple sources)
- TokenExpirationSeconds: 3607
- ConfigMapName: kube-root-ca.crt
- ConfigMapOptional: <nil>
- DownwardAPI: true

QoS Class: BestEffort

Node-Selectors: <none>

Tolerations:

- node.kubernetes.io/not-ready:NoExecute opExists for 300s
- node.kubernetes.io/unreachable:NoExecute opExists for 300s

Events:

Type	Reason	Age	From	Message
Normal	Scheduled	3m1s	default-scheduler	Successfully assigned default/my-app to minikube
Normal	Pulling	3m	kubelet	Pulling image "nginx"
Normal	Pulled	2m56s	kubelet	Successfully pulled image "nginx" in 4.224s (4.224s including waiting). Image size: 19.2804242 bytes.
Normal	Created	2m56s	kubelet	Created container: my-app-container
Normal	Started	2m56s	kubelet	Started container my-app-container

```
akashine@Advik:~$ kubectl get pod
NAME                                STATUS    RESTARTS   AGE
my-app                              1/1      Running    0          44s
nginx-5869d7778c-tpg4r             1/1      Running    0          44s
akashine@Advik:~$
```

```
akashine@Advik: ~$ kubectl get pod
NAME                                STATUS    RESTARTS   AGE
my-app                              1/1      Running    0          44s
nginx-5869d7778c-tpg4r             1/1      Running    0          44s
akashine@Advik:~$
```

Conditions:

Type	Status
PodReadyToStartContainers	True
Initialized	True
Ready	True
ContainersReady	True
PodScheduled	True

Volumes:

kube-api-access-tmbvj:

- Type: Projected (a volume that contains injected data from multiple sources)
- TokenExpirationSeconds: 3607
- ConfigMapName: kube-root-ca.crt
- ConfigMapOptional: <nil>
- DownwardAPI: true

QoS Class: BestEffort

Node-Selectors: <none>

Tolerations:

- node.kubernetes.io/not-ready:NoExecute opExists for 300s
- node.kubernetes.io/unreachable:NoExecute opExists for 300s

Events:

Type	Reason	Age	From	Message
Normal	Scheduled	3m1s	default-scheduler	Successfully assigned default/my-app to minikube
Normal	Pulling	3m	kubelet	Pulling image "nginx"
Normal	Pulled	2m56s	kubelet	Successfully pulled image "nginx" in 4.224s (4.224s including waiting). Image size: 19.2804242 bytes.
Normal	Created	2m56s	kubelet	Created container: my-app-container
Normal	Started	2m56s	kubelet	Started container my-app-container

```
akashine@Advik:~$ kubectl get pod
NAME                                STATUS    RESTARTS   AGE
my-app                              1/1      Running    0          44s
nginx-5869d7778c-tpg4r             1/1      Running    0          44s
akashine@Advik:~$
```

```
akashine@Advik: ~$ kubectl get pod
NAME                                READY    STATUS    RESTARTS   AGE
my-deploy-6f6bc8b6d-hsrhb           1/1      Running   0           2m57s
my-rs-pfck2                          1/1      Running   0           175m
my-rs-rjg76                         1/1      Running   0           175m
my-rs-twzsd                         1/1      Running   0           175m
webnginx-689ff98b69-lbq4p           1/1      Running   0           137m
akashine@Advik:~$ kubectl exec -it my-deploy-6f6bc8b6d-hsrhb -- /bin/bash/
OCI runtime exec failed: exec failed: unable to start container process: exec: "/bin/bash/": stat /bin/bash/: not a directory: unknown
n: Are you trying to mount a directory onto a file (or vice-versa)? Check if the specified host path exists and is the expected type
command terminated with exit code 126
akashine@Advik:~$ kubectl exec -it my-deploy-6f6bc8b6d-hsrhb -- /bin/bash
root@my-deploy-6f6bc8b6d-hsrhb:/usr/local/tomcat# ls
bin          conf         filtered-KEYS  LICENSE  native-jni-lib  README.md  RUNNING.txt  upstream-KEYS  webapps.dist
BUILDING.txt  CONTRIBUTING.md  lib           NOTICE  RELEASE-NOTES  webapps    webapps
root@my-deploy-6f6bc8b6d-hsrhb:/usr/local/tomcat# exit
exit
akashine@Advik:~$ curl http://192.168.49.2:30002/webapps/
<doctype html><html lang="en"><head><title>HTTP Status 404 - Not Found</title><style type="text/css">body {font-family:Tahoma,Arial,
sans-serif;} h1, h2, h3, b {color:white;background-color:#525D76;} h1 {font-size:22px;} h2 {font-size:16px;} h3 {font-size:14px;} p {
font-size:12px;} a {color:black;} a {line {height:1px;background-color:#525D76;border:none;}}/style></head><body><h1>HTTP Status 404 -
Not Found</h1><hr class="line" /><p><b>Type</b> Status Report</p><p><b>Description</b> The origin server did not find a current repre
sentation for the target resource or is not willing to disclose that one exists.</p><hr class="line" /><h3>Apache Tomcat/9.0.102</h3>
</body></html></html>
akashine@Advik:~$ curl http://192.168.49.2:30002/maven-web-app
<html>
<body>
<h2>Hello World!</h2>
</body>
</html>
akashine@Advik:~$
```

Namespace

```
akashine@Advik: ~$ service/my-service created
akashine@Advik:~$ kubectl get pod -n my-deploy
NAME                                READY    STATUS    RESTARTS   AGE
my-deploy-6f6bc8b6d-m5mzq           1/1      Running   0           4m37s
akashine@Advik:~$ kubectl get pod
NAME                                READY    STATUS    RESTARTS   AGE
my-deploy-6f6bc8b6d-hsrhb           1/1      Running   0           29m
my-rs-pfck2                          1/1      Running   0           3h21m
my-rs-rjg76                         1/1      Running   0           3h21m
my-rs-twzsd                         1/1      Running   0           3h21m
webnginx-689ff98b69-lbq4p           1/1      Running   0           164m
akashine@Advik:~$ kubectl get pod -n my-deploy
NAME                                READY    STATUS    RESTARTS   AGE
my-deploy-6f6bc8b6d-m5mzq           1/1      Running   0           5m16s
akashine@Advik:~$ kubectl get deploy -n my-deploy
NAME                                READY    UP-TO-DATE    AVAILABLE    AGE
my-deploy 1/1      1              1            5m38s
akashine@Advik:~$ kubectl get all --namespaces
error: unknown flag: --namespaces
See 'kubectl get --help' for usage.
akashine@Advik:~$ kubectl get all --namespace my-deploy
NAME                                READY    STATUS    RESTARTS   AGE
pod/my-deploy-6f6bc8b6d-m5mzq       1/1      Running   0           6m27s
akashine@Advik:~$ kubectl get all --namespace my-deploy
NAME                                TYPE     CLUSTER-IP    EXTERNAL-IP    PORT(S)          AGE
service/my-service                 NodePort  10.99.26.253  <none>         7070:30001/TCP   6m27s
akashine@Advik:~$ kubectl get all --namespace my-deploy
NAME                                READY    UP-TO-DATE    AVAILABLE    AGE
deployment.apps/my-deploy          1/1      1              1            6m27s
akashine@Advik:~$ kubectl get all --namespace my-deploy
NAME                                DESIRED    CURRENT    READY    AGE
replicaset.apps/my-deploy-6f6bc8b6d 1          1          1        6m27s
akashine@Advik:~$
```

Namespace yaml

```
akashine@Advik: ~$ kubectl get pod -n my-deploy
NAME                                READY    STATUS    RESTARTS   AGE
my-deploy-6f6bc8b6d-m5mzq           1/1      Running   0           5m16s
akashine@Advik:~$ kubectl get deploy -n my-deploy
NAME                                READY    UP-TO-DATE    AVAILABLE    AGE
my-deploy 1/1      1              1            5m38s
akashine@Advik:~$ kubectl get all --namespaces
error: unknown flag: --namespaces
See 'kubectl get --help' for usage.
akashine@Advik:~$ kubectl get all --namespace my-deploy
NAME                                READY    STATUS    RESTARTS   AGE
pod/my-deploy-6f6bc8b6d-m5mzq       1/1      Running   0           6m27s
akashine@Advik:~$ kubectl get all --namespace my-deploy
NAME                                TYPE     CLUSTER-IP    EXTERNAL-IP    PORT(S)          AGE
service/my-service                 NodePort  10.99.26.253  <none>         7070:30001/TCP   6m27s
akashine@Advik:~$ kubectl get all --namespace my-deploy
NAME                                READY    UP-TO-DATE    AVAILABLE    AGE
deployment.apps/my-deploy          1/1      1              1            6m27s
akashine@Advik:~$ kubectl get all --namespace my-deploy
NAME                                DESIRED    CURRENT    READY    AGE
replicaset.apps/my-deploy-6f6bc8b6d 1          1          1        6m27s
akashine@Advik:~$ sudo nano mydeploy.yml
[sudo] password for akashine:
akashine@Advik:~$ kubectl apply -f mydeploy.yml
namespace/my-demon created
akashine@Advik:~$ sudo nano nspod.yml
akashine@Advik:~$ kubectl apply -f nspod.yml
pod/my-pod created
akashine@Advik:~$ kubectl get pod -n my-demon
NAME                                READY    STATUS    RESTARTS   AGE
my-pod 1/1      Running   0           36s
akashine@Advik:~$
```