Namespace

Introduction:

Namespaces provide a mechanism for isolating groups of resources within a single cluster. Names of resources need to be unique within a namespace, but not across namespaces.

Kubernetes starts with four initial namespaces:

Default: Kubernetes includes this namespace so that you can start using your new cluster without first creating a namespace. By default, every resource will be a part of this namespace initially unless you explicitly assign a namespace.

kube-node-lease: This namespace holds Lease objects associated with each node. Node leases allow the kubelet to send heartbeats so that the control plane can detect node failure.

kube-public: This namespace is readable by *all* clients (including those not authenticated). This namespace is mostly reserved for cluster usage, in case that some resources should be visible and readable publicly throughout the whole cluster. The public aspect of this namespace is only a convention, not a requirement.

kube-system: The namespace for objects created by the Kubernetes system where all the management related pods are deployed.

Objectives:

1. Create a Namespace
2. Create any resource in a Namespace

1. Create a Namespace:

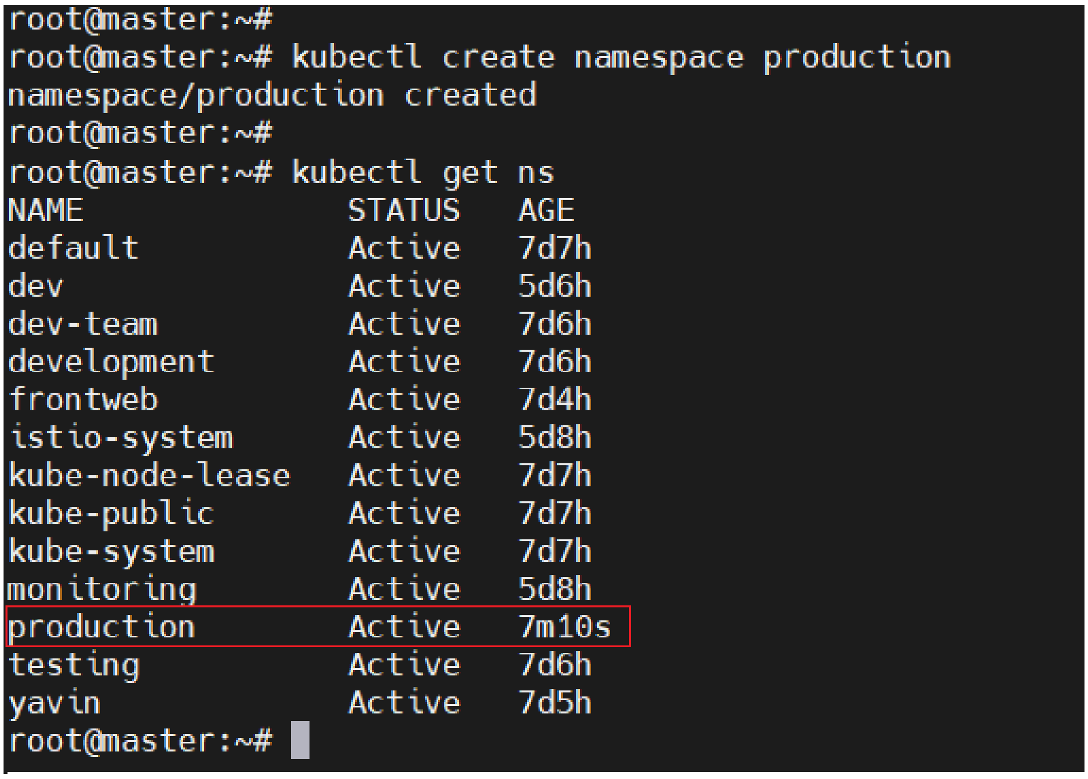
Use the below command to create a namespace.

```
kubectl create ns production
```

Check for the namespaces in the cluster.

```
kubectl get ns
```

above command will list all the namespaces within the cluster.



```
root@master:~#
root@master:~# kubectl create namespace production
namespace/production created
root@master:~#
root@master:~# kubectl get ns
NAME               STATUS   AGE
default            Active   7d7h
dev                Active   5d6h
dev-team           Active   7d6h
development        Active   7d6h
frontweb           Active   7d4h
istio-system       Active   5d8h
kube-node-lease    Active   7d7h
kube-public        Active   7d7h
kube-system        Active   7d7h
monitoring         Active   5d8h
production         Active   7m10s
testing            Active   7d6h
yavin              Active   7d5h
root@master:~#
```

2. **Create any resource in a Namespace:**

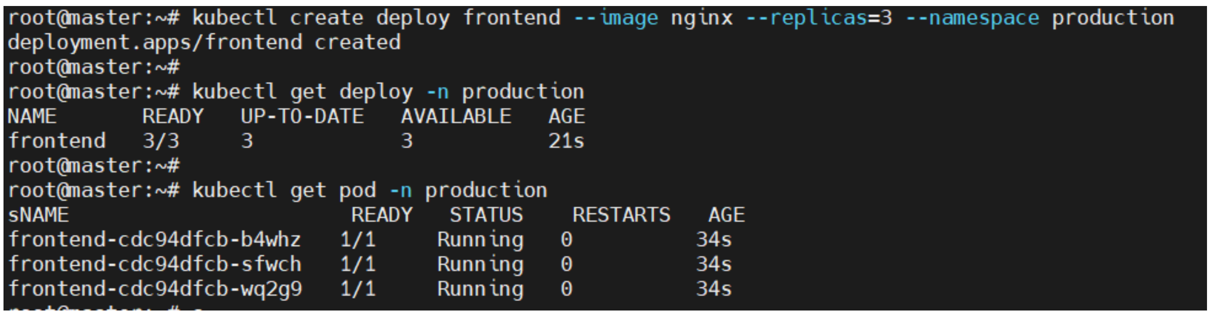
Let's create a deployment in production namespace using the below command.

```
kubectl create deploy frontend –image nginx –replicas 3 –namespace production
```

We can check the resources in the namespace using the below command.

```
kubectl get deploy -n production
```

```
kubectl get pods -n production
```



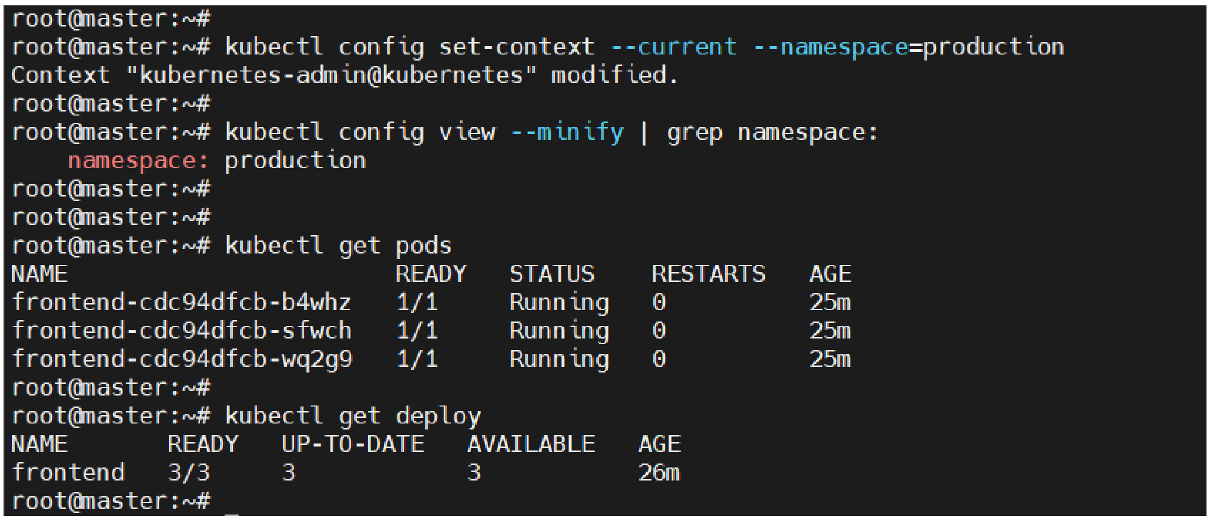
```
root@master:~# kubectl create deploy frontend --image nginx --replicas=3 --namespace production
deployment.apps/frontend created
root@master:~#
root@master:~# kubectl get deploy -n production
NAME       READY   UP-TO-DATE   AVAILABLE   AGE
frontend   3/3     3            3           21s
root@master:~#
root@master:~# kubectl get pod -n production
sNAME                       READY   STATUS    RESTARTS   AGE
frontend-cdc94dfcb-b4whz   1/1     Running   0          34s
frontend-cdc94dfcb-sfwch   1/1     Running   0          34s
frontend-cdc94dfcb-wq2g9   1/1     Running   0          34s
```

Use the below command to set the current context to the namespace Production. So now we need not to mention the namespace while getting or creating any resource.

```
kubectl config set-context –current –namespace=production
```

To validate the current context, we can use below command.

```
kubectl config view –minify | grep namespace:
```



```
root@master:~#
root@master:~# kubectl config set-context --current --namespace=production
Context "kubernetes-admin@kubernetes" modified.
root@master:~#
root@master:~# kubectl config view --minify | grep namespace:
    namespace: production
root@master:~#
root@master:~#
root@master:~# kubectl get pods
NAME                       READY   STATUS    RESTARTS   AGE
frontend-cdc94dfcb-b4whz   1/1     Running   0          25m
frontend-cdc94dfcb-sfwch   1/1     Running   0          25m
frontend-cdc94dfcb-wq2g9   1/1     Running   0          25m
root@master:~#
root@master:~# kubectl get deploy
NAME       READY   UP-TO-DATE   AVAILABLE   AGE
frontend   3/3     3            3           26m
root@master:~# _
```