

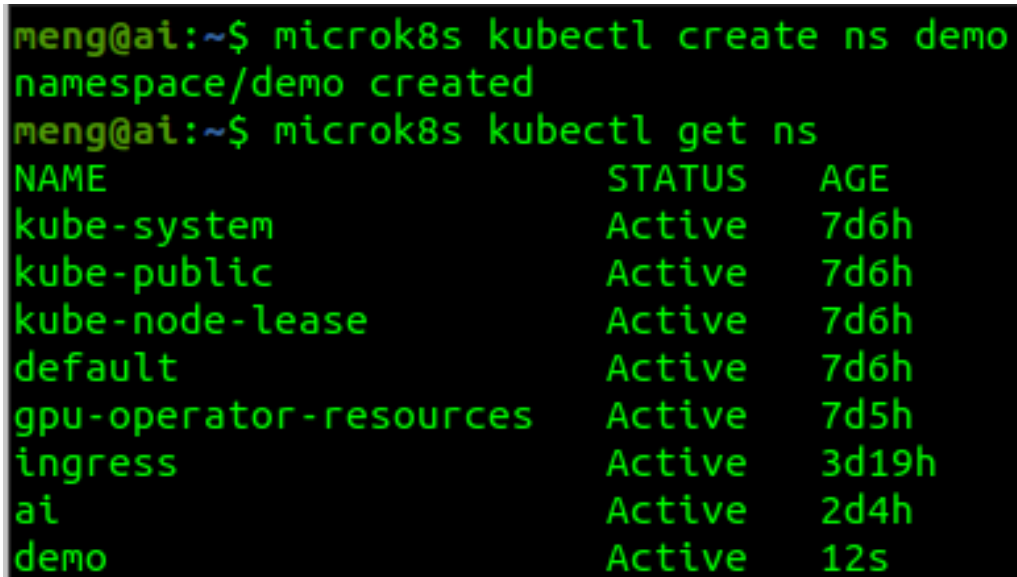
Kubernetes resources

Meng Oon Lee

July 5, 2021

1 Imperative approach

1. Create demo namespace
`kubectl create ns demo`

A terminal window with a black background and green text. The first command is 'meng@ai:~\$ microk8s kubectl create ns demo', followed by the output 'namespace/demo created'. The second command is 'meng@ai:~\$ microk8s kubectl get ns', followed by a table of namespaces.

```
meng@ai:~$ microk8s kubectl create ns demo
namespace/demo created
meng@ai:~$ microk8s kubectl get ns
NAME                                STATUS    AGE
kube-system                        Active    7d6h
kube-public                       Active    7d6h
kube-node-lease                   Active    7d6h
default                          Active    7d6h
gpu-operator-resources            Active    7d5h
ingress                          Active    3d19h
ai                                Active    2d4h
demo                             Active    12s
```

2. Label demo namespace
`kubectl label ns demo tier=test`

```

meng@ai:~$ microk8s kubectl label ns demo tier=test
namespace/demo labeled
meng@ai:~$ microk8s kubectl describe ns/demo
Name:          demo
Labels:        kubernetes.io/metadata.name=demo
               tier=test
Annotations:    <none>
Status:        Active

No resource quota.

No LimitRange resource.

```

3. Create nginx-alpine deployment
kubectl create deploy nginx-alpine --image=nginx:alpine --replicas=3 --namespace=demo

```

meng@ai:~$ microk8s kubectl create deploy nginx-alpine --image=nginx:alpine --replicas=3 --namespace=demo
deployment.apps/nginx-alpine created
meng@ai:~$ microk8s kubectl get deploy -n demo

```

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
nginx-alpine	3/3	3	3	107s

4. Label nginx-alpine deployment
kubectl label deploy nginx-alpine app=nginx tag=alpine --namespace=demo --overwrite

```

meng@ai:~$ microk8s kubectl label deploy nginx-alpine app=nginx tag=alpine --namespace=demo --overwrite
deployment.apps/nginx-alpine labeled
meng@ai:~$ microk8s kubectl describe deploy/nginx-alpine -n demo

```

Name:	nginx-alpine
Namespace:	demo
CreationTimestamp:	Mon, 05 Jul 2021 15:14:31 +0800
Labels:	app=nginx tag=alpine

5. Expose nginx-alpine deployment
kubectl expose deploy/nginx-alpine --port=8111 --namespace=demo

```

meng@ai:~$ microk8s kubectl expose deploy/nginx-alpine --port=8111 --namespace=demo
service/nginx-alpine exposed
meng@ai:~$ microk8s kubectl get svc --namespace=demo

```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
nginx-alpine	ClusterIP	10.152.183.20	<none>	8111/TCP	27s

6. Create config map
kubectl create configmap nginx-version --from-literal=version=alpine --namespace=demo

```
meng@ai:~$ microk8s kubectl create configmap nginx-version --from-literal=version=alpine --namespace=demo
configmap/nginx-version created
meng@ai:~$ microk8s kubectl describe cm/nginx-version -n demo
Name:         nginx-version
Namespace:    demo
Labels:       <none>
Annotations:  <none>

Data
====
version:
----
alpine
Events: <none>
```