Namespaces

In Kubernetes, a `namespace` is a way to divide cluster resources between multiple users/teams. Namespaces are intended for use in environments with many users spread across multiple teams, or projects, or environments like development, staging, and production.

When you do

```
kubectl get pods
```

it gets you the `pods` in the `default` namespace

Creating a new namespace

- Create a new namespace

```
kubectl create namespace backend-team
```

- Get all the namespaces

```
kubectl get namespaces
```

- Get all pods in the namespace

```
kubectl get pods -n my-namespace
```

- Create the manifest for a deployment in the namespace

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  namespace: backend-team
spec:
  matchLabels:
```

```
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:latest
        ports:
        - containerPort: 80
```

- Apply the manifest

```
kubectl apply -f deployment-ns.yml
```

- Get the deployments in the namespace

```
kubectl get deployment -n backend-team
```

- Get the pods in the namespace

```
kubectl get pods -n backend-team
```

- Set the default context to be the namespace

```
kubectl config set-context --current --namespace=backend-team
```

- Try seeing the pods now

```
kubectl get pods
```

- Revert back the kubectl config

```
kubectl config set-context --current --namespace=default
```