Kubernetes Part 2 3 of 23

Recapping how to run this locally

Creating a cluster

- Create a `kind.yml` file locally

```
kind: Cluster
apiVersion: kind.x-k8s.io/v1alpha4
nodes:
- role: control-plane
  extraPortMappings:
  - containerPort: 30007
    hostPort: 30007
- role: worker
  extraPortMappings:
  - containerPort: 30007
    hostPort: 30008
- role: worker
```

- Run the cluster locally

```
kind create cluster --config kind.yml --name local2
```

- Run docker ps to confirm that the cluster is running

```
docker ps
```

Creating a pod

- Create a pod manifest (pod.yml)

↗ ↑

```
metadata:
   c
   e
  containers:
  - name: nginx
    image: nginx
    ports:
    - containerPort: 80
```

- Apply the pod manifest

```
kubectl apply -f pod.yml
```

- Check if the pod exists now

```
kube get pods
```

- Check the logs

```
 kubectl logs -f nginx
```

- Delete the pod

```
kubectl delete pod nginx
```

Creating a replicaset

- Create the replicaset manifest

```
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: nginx-replicaset
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:

        app: nginx
    spec:
```

```
        image: nginx:latest
        ports:
        - containerPort: 80
```

- Apply the replicaset manifest

```
kubectl apply -f rs.yml
```

- Check the number of pods running now

```
kubectl get pods
```

- Try deleting a pod, and ensure it gets restarted
- Delete the replicaset

```
kubectl delete rs nginx-replicaset
```

Creating a Deployment

- Create a deployment manifest (deployment.yml)

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        ports:
```

- Apply the manifest

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

- Check the rs that exist now

```
kubectl get rs
```

- Check the pods that exist now

```
kubectl get pods
```

- Try creating a new deployment with a wrong image name

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx2:latest
        ports:
        - containerPort: 80
```

- Ensure that the old pods are still running

```
kubectl get pods
```

Keep the deployment, it'll come in handy in the 4th slide