What are deployments

- Deployment provides declarative updates for Pods and Replica Sets
- Deployment defines the state of the application
  - Kubernetes ensures that the cluster maintains the desired state of application
- Replication Controllers and Replica Sets fall short of key requirements to manage application deployments
- Deployment object is flexible for managing and scaling applications in Kubernetes

Why to use deployment

- Create a deployment
  - Deploy an application
- Update a deployment
  - Deploy a new version of application
- Perform rolling updates
  - Zero downtime during upgrades
- Perform rollback
  - Undo the last deployment
- Pause/Resume a deployment
  - Selective upgrades

- Create a Deployment to bring up a Replica Set and Pods
- Check the status of a Deployment to see if it succeeds or not
- Later, update that Deployment to recreate the Pods
- Rollback to an earlier Deployment revision if the current Deployment isn't stable
- Pause and resume a Deployment

Deployment

- A Deployment declaration in Kubernetes allows you to do app **deployments** and **updates**
- When using the deployment object, you define the **state** of your application
  - Kubernetes will then make sure the cluster matches your **desired** state
- Just using the **replication-controller** or **replication set** might be **cumbersome** to deploy apps
  - The **Deployment Object** is easier to use and gives you more possibilities

Deployment

- With a deployment object you can:
  - **Create** a deployment (e.g. deploying an app)
  - **Update** a deployment (e.g. deploying an new version)
  - Do **rolling updates** (Zero downtime deployment)
  - **Roll Back** to previous version
  - **Pause/Resume** a deployment (e.g. to roll-out to only certain percentage)

Useful Commands

Command	Description
kubectl get deployment	Get info on current deployments
kubectl get rs	Get info about the replica sets
kubectl get pods --show-labels	Get info about the labels attached to those pods
kubectl rollout status deployment/helloworld-deployment	Get deployment status
kubectl set image deployment/helloworld-deployment k8s-demo=k8s-demo:2	Run k8s-demo with image label version
kubectl edit deployment/helloworld-deployment	Edit the deployment Object
kubectl rollout history deployment/helloworld-deployment	Get the rollout history
kubectl rollout undo deployment/helloworld-deployment	Rollback to pervious version
kubectl rollout undo deployment/helloworld-deployment --to-revision=n	Rollback to any pervious version

Demo Placeholder

- A deployment