



Exercise 7.2: Working with Deployments

A Deployment is a watch loop object which we have been working with in the previous labs. A Deployment provides a declarative update to Pods and ReplicaSets and ensure a particular number of pods are created in general, several could be on a single node. Deployment is a high-level resource object that is used to manage the rollout and scaling of containerized applications. A Deployment describes the desired state of the application, such as the number of replicas, and the container image to use. When a Deployment is created, Kubernetes will automatically create and manage the necessary replica sets, which in turn will create and manage the necessary pods to ensure that the desired state of the application is met. Deployment also provide rolling updates, which allow for updating an application to a new version without downtime by gradually replacing the old replicas with new ones. Using Deployment in Kubernetes makes it easy to manage and scale containerized applications while ensuring high availability and reliability.

1. We begin by creating a yaml file. In this case the kind would be set to deployment. We can generate the yaml file using the imperative method

```
student@cp:~$ kubectl create deploy webserver --image nginx:1.22.1 --replicas=2 \
--dry-run=client -o yaml | tee dep.yaml
```

```
student@cp:~$ cat dep.yaml
```

YAML

dep.yaml

```
1  ....
2  kind: Deployment
3  ....
4  name: webserver
5  ....
6  replicas: 2
7  ....
8      app: webserver
9  ....
10
```

2. Create and verify the newly formed Deployment. There should be two replicas of Pods created in the cluster.

```
student@cp:~$ kubectl create -f dep.yaml
```

```
deployment.apps/webserver created
```

```
student@cp:~$ kubectl get deploy
```

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
webserver	2/2	2	2	14s

```
student@cp:~$ kubectl get pod
```

NAME	READY	STATUS	RESTARTS	AGE
webserver-6cbc654ddc-lssbm	1/1	Running	0	42s
webserver-6cbc654ddc-xpmtl	1/1	Running	0	42s

3. Verify the image running inside the Pods. We will use this information in the next section.

```
student@cp:~$ kubectl describe pod webserver-6cbc654ddc-lssbm | grep Image:
```

```
Image:          nginx:1.22.1
```