Exercise 4.4: Using Labels

Create and work with labels. We will understand how the deployment, replicaSet, and pod labels interact.

1. Create a new deployment called `design2`

```
student@cp:~$ kubectl create deployment design2 --image=nginx
```

```
deployment.apps/design2 created
```

2. View the wide **kubectl get** output for the `design2` deployment and make note of the `SELECTOR`

```
student@cp:~$ kubectl get deployments.apps design2 -o wide
```

```
NAME     READY  UP-TO-DATE  AVAILABLE  AGE    CONTAINERS  IMAGES  SELECTOR
design2  1/1    1           1          2m13s  nginx       nginx   app=design2
```

3. Use the **-l** option to use the selector to list the pods running inside the deployment. There should be only one pod running.

```
student@cp:~$ kubectl get -l app=design2 pod
```

```
NAME                       READY   STATUS    RESTARTS   AGE
design2-766d48574f-5w274   1/1     Running   0          3m1s
```

4. View the pod details in YAML format using the deployment selector. This time use the **–selector** option. Find the pod label in the output. It should match that of the deployment.

```
student@cp:~$ kubectl get --selector app=design2 pod  -o yaml
```

```
apiVersion: v1
kind: Pod
metadata:
  creationTimestamp: "2024-06-11T16:29:37Z"
  generateName: design2-766d48574f-
  labels:
    app: design2
    pod-template-hash: 766d48574f
....
```

5. Edit the pod label to be your favorite color.

```
student@cp:~$ kubectl edit pod design2-766d48574f-5w274
```

YAML

```
....
  labels:
    app: orange                        #<<-- Edit this line
    pod-template-hash: 766d48574f
  name: design2-766d48574f-5w274
....
```

6. Now view how many pods are in the deployment. Then how many have `design2` in their name. Note the `AGE` of the pods.

```
student@cp:~$ kubectl get deployments.apps design2 -o wide
```

```
NAME      READY  UP-TO-DATE  AVAILABLE  AGE  CONTAINERS  IMAGES  SELECTOR
design2   1/1    1           1          56s  nginx       nginx   app=design2
```

```
student@cp:~$ kubectl get pods | grep design2
```

```
design2-766d48574f-5w274   1/1     Running            0          82s
design2-766d48574f-xttgg   1/1     Running            0          2m12s
```

7. Delete the `design2` deployment.

```
student@cp:~$ kubectl delete deploy design2
```

```
deployment.apps "design2" deleted
```

8. Check again for pods with `design2` in their names. You should find one pod, with an `AGE` of when you first created the deployment. Once the label was edited the deployment created a new pod in order that the status matches the spec and there be a replica running with the intended label.

```
student@cp:~$ kubectl get pods | grep design2
```

```
design2-766d48574f-5w274  1/1    Running           0         38m
```

9. Delete the pod using the **-l** and the label you edited to be your favorite color in a previous step. The command details have been omitted. Use previous steps to figure out these commands.