Task

1. Create a Deployment named `nginx` with 3 replicas. The Pods should use the `nginx:1.23.0` image and the name `nginx`. The Deployment uses the label `tier=backend`. The Pod template should use the label `app=v1`.

Readme.md ! nginx-deployment.yaml 1, U

Day-8 > ! nginx-deployment.yaml > {} spec > {} template > {} spec > [ ] containers > {} 0

io.k8s.api.apps.v1.Deployment (v1@deployment.json)

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx
  labels:
    tier: backend
spec:
  replicas: 3
  selector:
    matchLabels:
      app: v1
  template:
    metadata:
      labels:
        app: v1
    spec:
      containers:
      - name: nginx
        image: nginx:1.23.0
```

2. List the Deployment and ensure the correct number of replicas is running.

```
Bhakti@LAPTOP-DNC3NQI0 MINGW64 /d/Github/CKA2024 (main)
$ kubectl apply -f D:/Github/CKA2024/Day-8/nginx-deployment.yaml
deployment.apps/nginx created

Bhakti@LAPTOP-DNC3NQI0 MINGW64 /d/Github/CKA2024 (main)
$ kubectl get deployments
NAME    READY   UP-TO-DATE   AVAILABLE   AGE
nginx   3/3     3            3           64s
```

```
Bhakti@LAPTOP-DNC3NQI0 MINGW64 /d/Github/CKA2024 (main)
$ kubectl get pods
NAME                     READY   STATUS    RESTARTS   AGE
nginx-5d9b68bdd7-hs5nl   1/1     Running   0          79s
nginx-5d9b68bdd7-qz722   1/1     Running   0          79s
nginx-5d9b68bdd7-szqsc   1/1     Running   0          79s
```

3. Update the image to `nginx:1.23.4`.

```
Bhakti@LAPTOP-DNC3NQI0 MINGW64 /d/Github/CKA2024 (main)
$ kubectl set image deployment/nginx nginx=nginx=1.23.4
deployment.apps/nginx image updated
```

4. Verify that the change has been rolled out to all replicas.

```
Bhakti@LAPTOP-DNC3NQI0 MINGW64 /d/Github/CKA2024 (main)
$ kubectl rollout status deployment/nginx
Waiting for deployment "nginx" rollout to finish: 1 out of 3 new replicas have been updated...
Waiting for deployment "nginx" rollout to finish: 1 out of 3 new replicas have been updated...
Waiting for deployment "nginx" rollout to finish: 1 out of 3 new replicas have been updated...
Waiting for deployment "nginx" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "nginx" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "nginx" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "nginx" rollout to finish: 1 old replicas are pending termination...
Waiting for deployment "nginx" rollout to finish: 1 old replicas are pending termination...
deployment "nginx" successfully rolled out
```

5. Assign the change cause "Pick up patch version" to the revision.

```
Bhakti@LAPTOP-DNC3NQI0 MINGW64 /d/Github/CKA2024 (main)
$ kubectl annotate deployment nginx kubernetes.io/change-cause="Pick up patch version"
deployment.apps/nginx annotated
```

6. Scale the Deployment to 5 replicas.

```
Bhakti@LAPTOP-DNC3NQI0 MINGW64 /d/Github/CKA2024 (main)
$ kubectl scale deployment/nginx --replicas=5
deployment.apps/nginx scaled
```

7. Have a look at the Deployment rollout history.

```
Bhakti@LAPTOP-DNC3NQI0 MINGW64 /d/Github/CKA2024 (main)
$ kubectl rollout history deployment/nginx
deployment.apps/nginx
REVISION  CHANGE-CAUSE
2         <none>
3         Pick up patch version
4         <none>
```

8. Revert the Deployment to revision 1.

```
Bhakti@LAPTOP-DNC3NQI0 MINGW64 /d/Github/CKA2024 (main)
$ kubectl rollout undo deployment/nginx --to-revision=1
```

9. Ensure that the Pods use the image `nginx:1.23.0`.

```
Bhakti@LAPTOP-DNC3NQI0 MINGW64 /d/Github/CKA2024 (main)
$ kubectl get pods -l app=v1 -o jsonpath="{.items[*].spec.containers[*].image}"
nginx:1.23.0 nginx:1.23.0 nginx:1.23.0 nginx:1.23.0 nginx:1.23.0
```