Kubernetes Lab 3

1-Create a deployment called my-first-deployment of image nginx:alpine in the default namespace.
Check to make sure the deployment is healthy.

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-first-deployment
  labels:
    app: nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:alpine
        ports:
        - containerPort: 80
```

```
controlplane $ vim my-first-deployment.yaml
controlplane $ kubectl apply -f my-first-deployment.yaml
deployment.apps/my-first-deployment created
controlplane $ kubectl get deployment
NAME                  READY   UP-TO-DATE   AVAILABLE   AGE
my-first-deployment   2/2     2            2           27s
controlplane $
```

2-Scale my-first-deployment up to run 3 replicas.
Check to make sure all 3 replicas are ready.

```
controlplane $ vim my-first-deployment.yaml
controlplane $ kubectl apply -f my-first-deployment.yaml
deployment.apps/my-first-deployment configured
controlplane $ kubectl get deployment
NAME                  READY   UP-TO-DATE   AVAILABLE   AGE
my-first-deployment   3/3     3            3           4m8s
controlplane $ kubectl get pod
NAME                                   READY   STATUS    RESTARTS   AGE
my-first-deployment-774f96d4d9-dhvbm   1/1     Running   0          4m17s
my-first-deployment-774f96d4d9-pww9s   1/1     Running   0          22s
my-first-deployment-774f96d4d9-sdlxn   1/1     Running   0          4m17s
controlplane $
```

another way using scale command

```
controlplane $ kubectl scale deployment my-first-deployment --replicas=3
deployment.apps/my-first-deployment scaled
controlplane $ kubectl get deployment
NAME                  READY   UP-TO-DATE   AVAILABLE   AGE
my-first-deployment   3/3     3            3           11m
controlplane $ kubectl apply -f my-first-deployment.yaml
deployment.apps/my-first-deployment configured
controlplane $ kubectl get deployment
NAME                  READY   UP-TO-DATE   AVAILABLE   AGE
my-first-deployment   2/2     2            2           12m
controlplane $
```

3-Scale my-first-deployment down to run 2 replicas.

```
controlplane $ vim my-first-deployment.yaml
controlplane $ kubectl apply -f my-first-deployment.yaml
deployment.apps/my-first-deployment configured
controlplane $ kubectl get deployment
NAME                  READY   UP-TO-DATE   AVAILABLE   AGE
my-first-deployment   2/2     2            2           9m17s
controlplane $
```

4-Change the image my-first-deployment runs from nginx:alpine to httpd:alpine .
k set image deployment my-first-deployment nginx=httpd:alpine

```
controlplane $ vim my-first-deployment.yaml
controlplane $ kubectl apply -f my-first-deployment.yaml
deployment.apps/my-first-deployment configured
controlplane $ kubectl rollout status
error: required resource not specified
controlplane $ kubectl rollout status deployment my-first-deployment
deployment "my-first-deployment" successfully rolled out
controlplane $ kubectl get deployment
NAME                  READY   UP-TO-DATE   AVAILABLE   AGE
my-first-deployment   2/2     2            2           22m
controlplane $ kubectl get pod
NAME                                   READY   STATUS    RESTARTS   AGE
my-first-deployment-84f6f77b7c-bxh4l   1/1     Running   0          75s
my-first-deployment-84f6f77b7c-q6d5q   1/1     Running   0          72s
controlplane $ kubectl describe my-first-deployment-84f6f77b7c-bxh4l
error: the server doesn't have a resource type "my-first-deployment-84f6f77b7c-bxh4l"
controlplane $ kubectl describe pod my-first-deployment-84f6f77b7c-bxh4l
Name:             my-first-deployment-84f6f77b7c-bxh4l
Namespace:        default
Priority:         0
Service Account:  default
Node:             node01/172.30.2.2
Start Time:       Mon, 02 Jan 2023 12:04:52 +0000
Labels:           app=nginx
                  pod-template-hash=84f6f77b7c
Annotations:      cni.projectcalico.org/containerID: db4adddaeb93869b66b06ea1ed41e1b2ca
                  cni.projectcalico.org/podIP: 192.168.1.6/32
                  cni.projectcalico.org/podIPs: 192.168.1.6/32
Status:           Running
IP:               192.168.1.6
IPs:
  IP:           192.168.1.6
Controlled By:  ReplicaSet/my-first-deployment-84f6f77b7c
Containers:
  nginx:
    Container ID:   containerd://09b14e65868e8e71bc5471da2cf5d931812d723ceeeecc2db7edef
    Image:          httpd:alpine
    Image ID:       docker.io/library/httpd@sha256:86ed18b4670b3be349e62f05c34bf0c28f3e
    Port:           80/TCP
    Host Port:      0/TCP
    State:          Running
      Started:      Mon, 02 Jan 2023 12:04:54 +0000
    Ready:          True
    Restart Count:  0
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-m7hkt (ro)
```

5-Delete the deployment my-first-deployment

```
controlplane $ kubectl delete deployment  my-first-deployment
deployment.apps "my-first-deployment" deleted
controlplane $ kubectl get  deployment
No resources found in default namespace.
controlplane $
```

6-Create deployment from the below yaml

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: frontend-deployment
  namespace: default
spec:
  replicas: 4
  selector:
    matchLabels:
      name: busybox-pod
  strategy:
    rollingUpdate:
      maxSurge: 25%
      maxUnavailable: 25%
    type: RollingUpdate
  template:
    metadata:
      labels:
        name: busybox-pod
    spec:
      containers:
      - command:
        - sh
        - -c
        - echo Hello Kubernetes! && sleep 3600
        image: busybox888
        imagePullPolicy: Always
        name: busybox-container
```

```
controlplane $ vim frontend-deployment.yaml
controlplane $ kubectl apply -f frontend-deployment.yaml
deployment.apps/frontend-deployment created
controlplane $
```

7-How many ReplicaSets exist on the system now?

```
controlplane $ kubectl get replicaset
NAME                           DESIRED   CURRENT   READY   AGE
frontend-deployment-7fbf4f5cd9   4         4         0       103s
controlplane $
```

8-How many PODs exist on the system now?

```
controlplane $ kubectl get pod
NAME                                 READY   STATUS             RESTARTS   AGE
frontend-deployment-7fbf4f5cd9-8plhm   0/1     ImagePullBackOff   0          3m56s
frontend-deployment-7fbf4f5cd9-f2wbr   0/1     ImagePullBackOff   0          3m56s
frontend-deployment-7fbf4f5cd9-hgd47   0/1     ImagePullBackOff   0          3m56s
frontend-deployment-7fbf4f5cd9-r6w9k   0/1     ImagePullBackOff   0          3m56s
controlplane $
```

9-Out of all the existing PODs, how many are ready?

```
controlplane $ kubectl get replicaset
NAME                            DESIRED   CURRENT   READY   AGE
frontend-deployment-7fbf4f5cd9   4         4         0       103s
controlplane $
```

10-What is the image used to create the pods in the new deployment?

```
Containers:
  busybox-container:
    Container ID:
    Image:          busybox888
    Image ID:
    Port:           <none>
    Host Port:      <none>
    Command:
      sh
      -c
      echo Hello Kubernetes! && sleep 3600
    State:          Waiting
      Reason:       ImagePullBackOff
    Ready:          False
    Restart Count:  0
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-4xz8z (ro)
```

11-Why do you think the deployment is not ready?

```
Events:
  Type     Reason     Age                   From               Message
  ----     ------     ----                  ----               -------
  Normal   Scheduled  5m28s                 default-scheduler  Successfully assigned default/frontend-deployment-7fbf4f5cd9-8plhm to controlplane
  Normal   Pulling    3m57s (x4 over 5m28s)  kubelet            Pulling image "busybox888"
  Warning  Failed     3m55s (x4 over 5m27s)  kubelet            Failed to pull image "busybox888": rpc error: code = Unknown desc = failed to pull and unpack
image "docker.io/library/busybox888:latest": failed to resolve reference "docker.io/library/busybox888:latest": pull access denied, repository does not exist
or may require authorization: server message: insufficient_scope: authorization failed
  Warning  Failed     3m55s (x4 over 5m27s)  kubelet            Error: ErrImagePull
  Warning  Failed     3m45s (x6 over 5m26s)  kubelet            Error: ImagePullBackOff
  Normal   BackOff    26s (x21 over 5m26s)   kubelet            Back-off pulling image "busybox888"
controlplane $
```

there is no images with this name busybox888

12-Create a new Deployment using the below yaml

```yaml
apiVersion: apps/v1
kind: deployment
metadata:
  name: deployment-1
spec:
  replicas: 2
  selector:
    matchLabels:
      name: busybox-pod
  template:
    metadata:
      labels:
        name: busybox-pod
    spec:
      containers:
      - name: busybox-container
        image: busybox888
        command:
        - sh
        - "-c"
        - echo Hello Kubernetes! && sleep 3600
~
```

```
controlplane $ vim deployment-1.yaml
controlplane $ kubectl apply -f deployment-1.yaml
Error from server (BadRequest): error when creating "deployment-1.yaml": deployment in version "v1" cannot be handled as a Deployment: no kind "deployment" is
 registered for version "apps/v1" in scheme "pkg/api/legacyscheme/scheme.go:30"
controlplane $
```

13-There is an issue with the file, so try to fix it. and correct the value of kind.

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: deployment-1
spec:
  replicas: 2
  selector:
    matchLabels:
      name: busybox-pod
  template:
    metadata:
      labels:
        name: busybox-pod
    spec:
      containers:
      - name: busybox-container
        image: busybox
        command:
        - sh
        - "-c"
        - echo Hello Kubernetes! && sleep 3600
```

```
controlplane $ vim deployment-1.yaml
controlplane $ kubectl apply -f deployment-1.yaml
deployment.apps/deployment-1 created
controlplane $ kubectl get deployment
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
deployment-1   2/2     2            2           16s
controlplane $ kubectl get pod
NAME                            READY   STATUS    RESTARTS   AGE
deployment-1-6b9644f597-4xfdh   1/1     Running   0          25s
deployment-1-6b9644f597-cxrnc   1/1     Running   0          25s
controlplane $
```