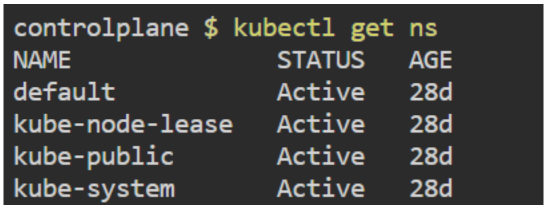
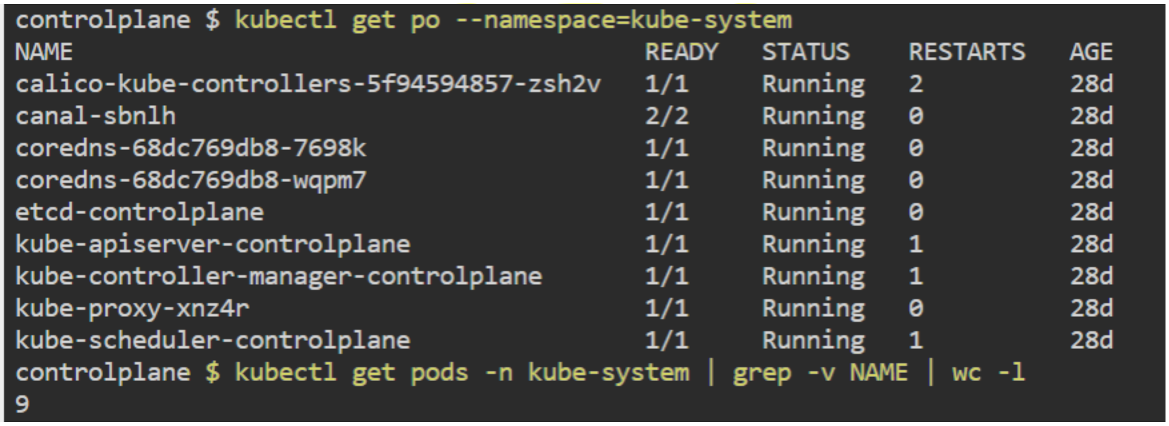
Lab2

1- How many Namespaces exist on the system?



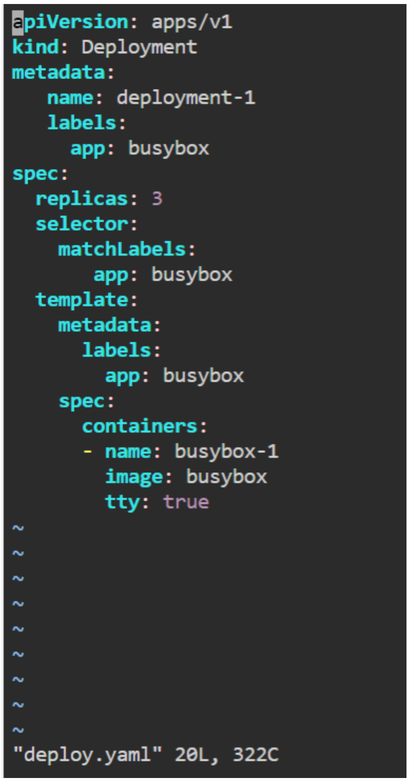
```
controlplane $ kubectl get ns
NAME              STATUS   AGE
default           Active   28d
kube-node-lease   Active   28d
kube-public       Active   28d
kube-system       Active   28d
```

2-How many pods exist in the kube-system namespace?

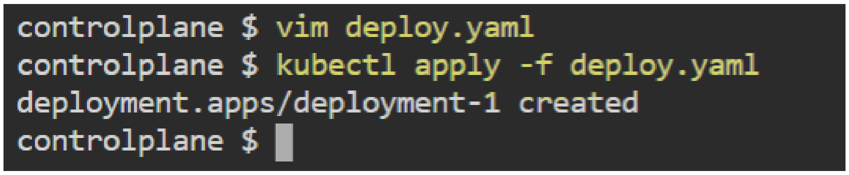


```
controlplane $ kubectl get po --namespace=kube-system
NAME                                      READY   STATUS    RESTARTS   AGE
calico-kube-controllers-5f94594857-zsh2v  1/1     Running   2          28d
canal-sbnlh                               2/2     Running   0          28d
coredns-68dc769db8-7698k                  1/1     Running   0          28d
coredns-68dc769db8-wqpm7                  1/1     Running   0          28d
etcd-controlplane                         1/1     Running   0          28d
kube-apiserver-controlplane               1/1     Running   1          28d
kube-controller-manager-controlplane      1/1     Running   1          28d
kube-proxy-xnz4r                          1/1     Running   0          28d
kube-scheduler-controlplane               1/1     Running   1          28d
controlplane $ kubectl get pods -n kube-system | grep -v NAME | wc -l
9
```

3- create a Deployment with, name= deployment-1, image= busybox, replicas= 3

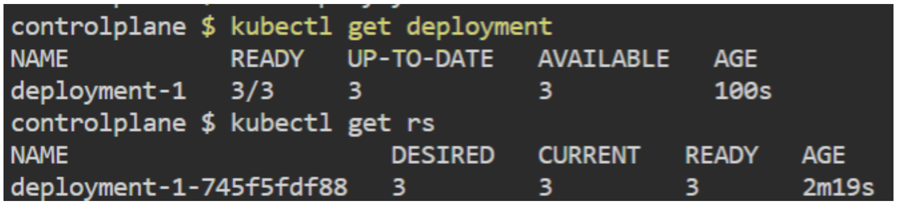


```
apiVersion: apps/v1
kind: Deployment
metadata:
   name: deployment-1
   labels:
     app: busybox
spec:
  replicas: 3
  selector:
    matchLabels:
       app: busybox
  template:
    metadata:
      labels:
        app: busybox
    spec:
      containers:
      - name: busybox-1
        image: busybox
        tty: true
~
~
~
~
~
~
~
~
~
"deploy.yaml" 20L, 322C
```



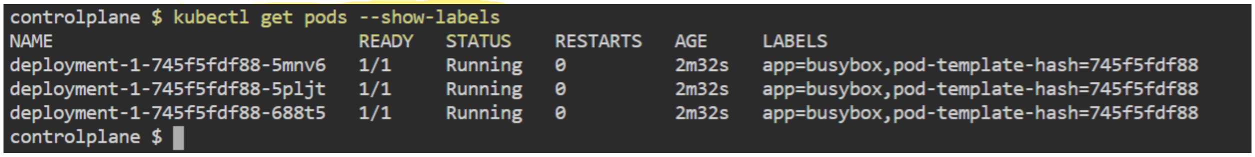
```
controlplane $ vim deploy.yaml
controlplane $ kubectl apply -f deploy.yaml
deployment.apps/deployment-1 created
controlplane $
```

4- How many Deployments and ReplicaSets exist on the system now?



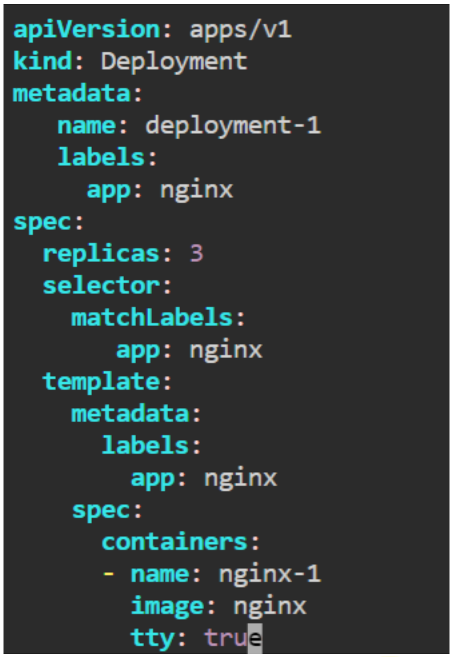
```
controlplane $ kubectl get deployment
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
deployment-1   3/3     3            3           100s
controlplane $ kubectl get rs
NAME                      DESIRED   CURRENT   READY   AGE
deployment-1-745f5fdf88   3         3         3       2m19s
```

5- How many pods are ready with the deployment-1?

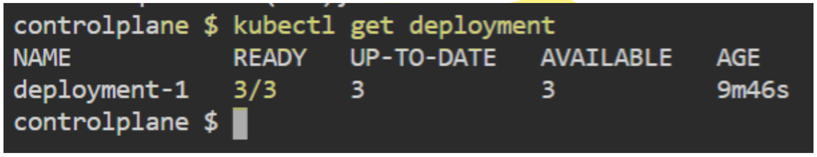


```
controlplane $ kubectl get pods --show-labels
NAME                            READY   STATUS    RESTARTS   AGE     LABELS
deployment-1-745f5fdf88-5mnv6   1/1     Running   0          2m32s   app=busybox,pod-template-hash=745f5fdf88
deployment-1-745f5fdf88-5pljt   1/1     Running   0          2m32s   app=busybox,pod-template-hash=745f5fdf88
deployment-1-745f5fdf88-688t5   1/1     Running   0          2m32s   app=busybox,pod-template-hash=745f5fdf88
controlplane $
```

6- Update deployment-1 image to nginx then check the ready pods again

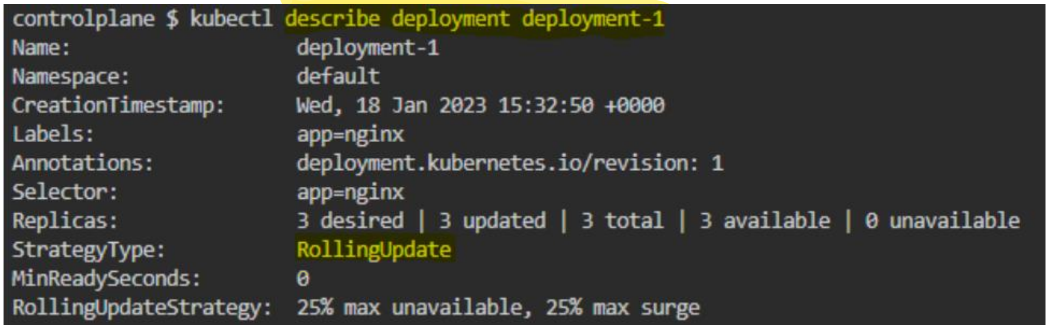


```
apiVersion: apps/v1
kind: Deployment
metadata:
   name: deployment-1
   labels:
     app: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
       app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx-1
        image: nginx
        tty: true
```



```
controlplane $ kubectl get deployment
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
deployment-1   3/3     3            3           9m46s
controlplane $
```

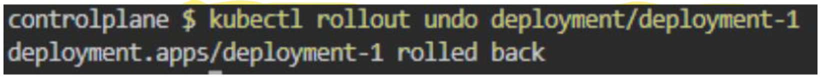
7- Run kubectl describe deployment deployment-1 and check events What is the deployment strategy used to upgrade the deployment-1?



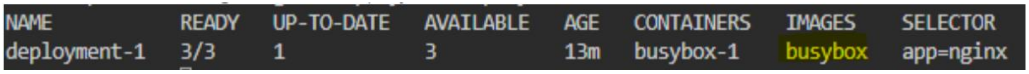
```
controlplane $ kubectl describe deployment deployment-1
Name:                   deployment-1
Namespace:              default
CreationTimestamp:      Wed, 18 Jan 2023 15:32:50 +0000
Labels:                 app=nginx
Annotations:            deployment.kubernetes.io/revision: 1
Selector:               app=nginx
Replicas:               3 desired | 3 updated | 3 total | 3 available | 0 unavailable
StrategyType:           RollingUpdate
MinReadySeconds:        0
RollingUpdateStrategy:  25% max unavailable, 25% max surge
```

8- Rollback the deployment-1

What is the used image with the deployment-1?



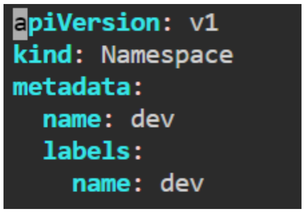
```
controlplane $ kubectl rollout undo deployment/deployment-1
deployment.apps/deployment-1 rolled back
```



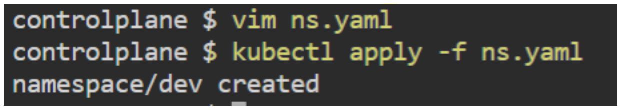
```
NAME           READY   UP-TO-DATE   AVAILABLE   AGE   CONTAINERS   IMAGES    SELECTOR
deployment-1   3/3     1            3           13m   busybox-1    busybox   app=nginx
```

10- Create a deployment with, Name: dev-deploy, Image: redis, Replicas: 2, Namespace: dev

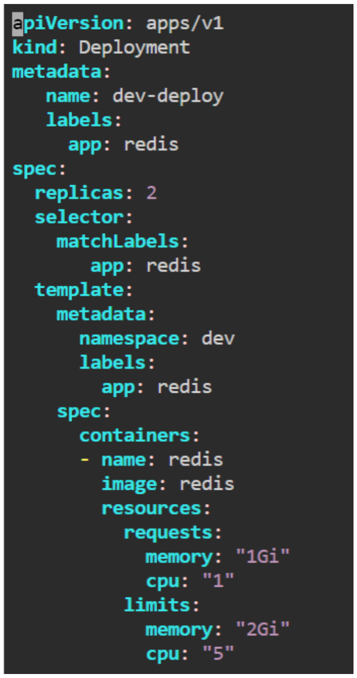
Resources Requests: CPU: .5 vcpu, Mem: 1G, Resources Limits:, CPU: 1 vcpu, Mem: 2G



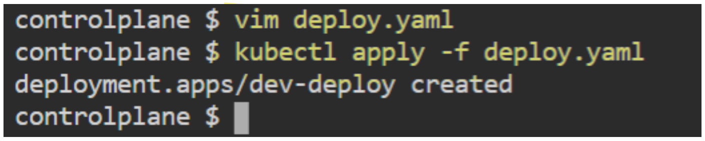
```
apiVersion: v1
kind: Namespace
metadata:
  name: dev
  labels:
    name: dev
```



```
controlplane $ vim ns.yaml
controlplane $ kubectl apply -f ns.yaml
namespace/dev created
```



```
apiVersion: apps/v1
kind: Deployment
metadata:
   name: dev-deploy
   labels:
     app: redis
spec:
  replicas: 2
  selector:
    matchLabels:
       app: redis
  template:
    metadata:
      namespace: dev
      labels:
        app: redis
    spec:
      containers:
      - name: redis
        image: redis
        resources:
          requests:
            memory: "1Gi"
            cpu: "1"
          limits:
            memory: "2Gi"
            cpu: "5"
```



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