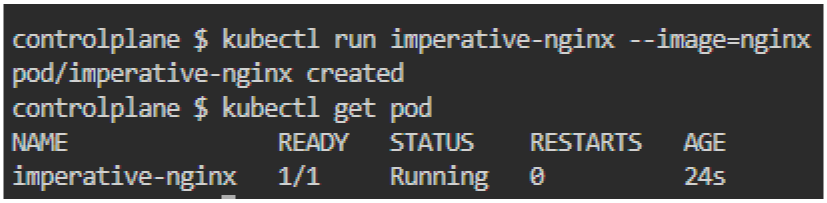
Kubernetes

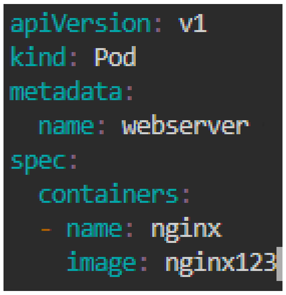
Lab1

1- Create a pod with the name “`imperative-nginx`” and with the image `nginx and latest tag`. using Imperative command (not yaml).

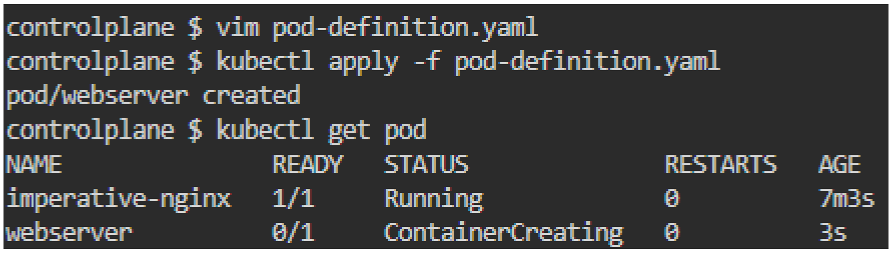


```
controlplane $ kubectl run imperative-nginx --image=nginx
pod/imperative-nginx created
controlplane $ kubectl get pod
NAME               READY   STATUS    RESTARTS   AGE
imperative-nginx   1/1     Running   0          24s
```

2- Create a pod with the name `webserver` and with the image “`nginx123`” Use a pod-definition YAML file.



```
apiVersion: v1
kind: Pod
metadata:
  name: webserver
spec:
  containers:
  - name: nginx
    image: nginx123
```

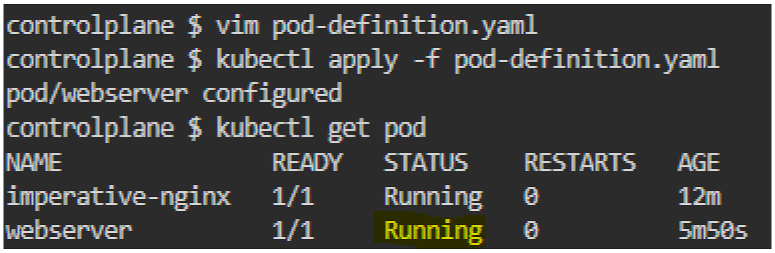


```
controlplane $ vim pod-definition.yaml
controlplane $ kubectl apply -f pod-definition.yaml
pod/webserver created
controlplane $ kubectl get pod
NAME               READY   STATUS              RESTARTS   AGE
imperative-nginx   1/1     Running             0          7m3s
webserver          0/1     ContainerCreating   0          3s
```

3- What is the nginx pod status?

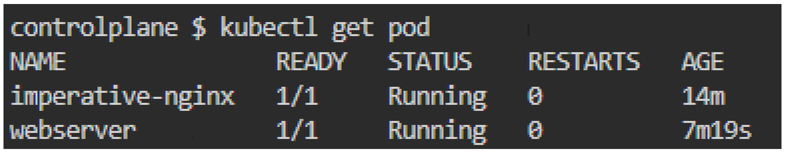
State: ImagePullBackOff

4- Change the nginx pod image to “nginx” check the status again



```
controlplane $ vim pod-definition.yaml
controlplane $ kubectl apply -f pod-definition.yaml
pod/webserver configured
controlplane $ kubectl get pod
NAME               READY   STATUS    RESTARTS   AGE
imperative-nginx   1/1     Running   0          12m
webserver          1/1     Running   0          5m50s
```

5- How many pods are running in the system? Type the command to show this

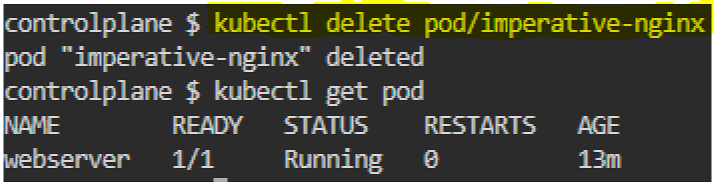


```
controlplane $ kubectl get pod
NAME               READY   STATUS    RESTARTS   AGE
imperative-nginx   1/1     Running   0          14m
webserver          1/1     Running   0          7m19s
```

6- What does READY column in the output of get pods command indicate?

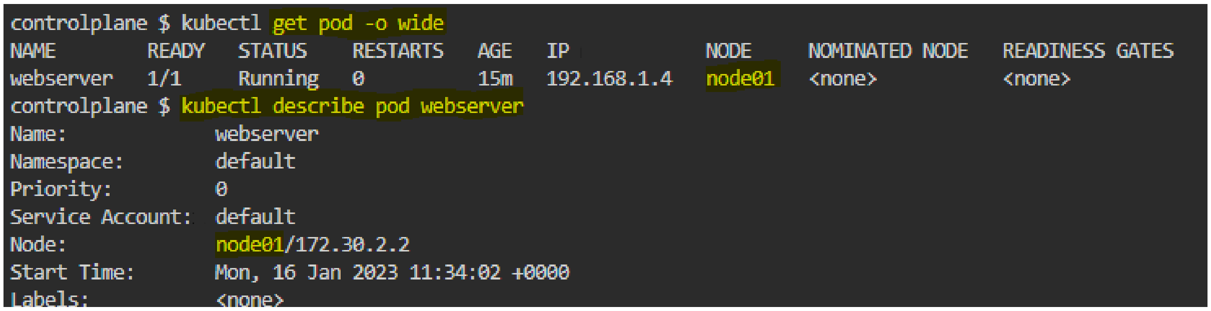
Ans: it shows how many containers in a pod are considered ready

7- Delete the first pod named imperative-nginx you just created. Type the command to do this



```
controlplane $ kubectl delete pod/imperative-nginx
pod "imperative-nginx" deleted
controlplane $ kubectl get pod
NAME        READY   STATUS    RESTARTS   AGE
webserver   1/1     Running   0          13m
```

8- Which node is pod named webserver running on (list two commands to do this)

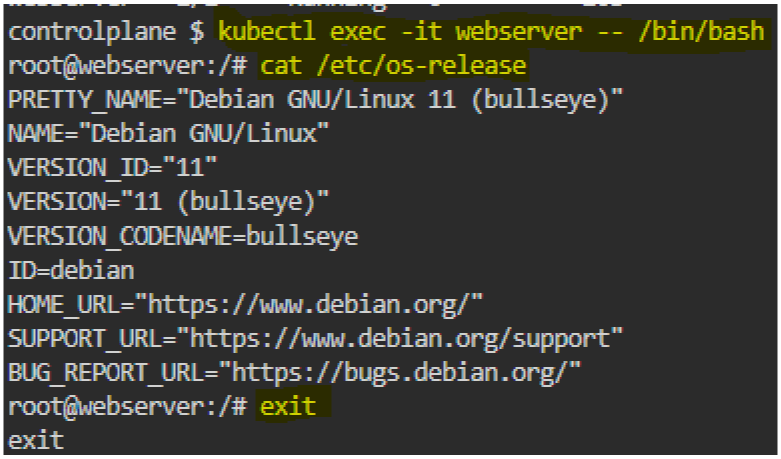


```
controlplane $ kubectl get pod -o wide
NAME        READY   STATUS    RESTARTS   AGE   IP            NODE     NOMINATED NODE   READINESS GATES
webserver   1/1     Running   0          15m   192.168.1.4   node01   <none>           <none>
controlplane $ kubectl describe pod webserver
Name:             webserver
Namespace:        default
Priority:         0
Service Account:  default
Node:             node01/172.30.2.2
Start Time:       Mon, 16 Jan 2023 11:34:02 +0000
Labels:           <none>
```

9- Get a shell to the running container i.e ssh into it (figure out the command)

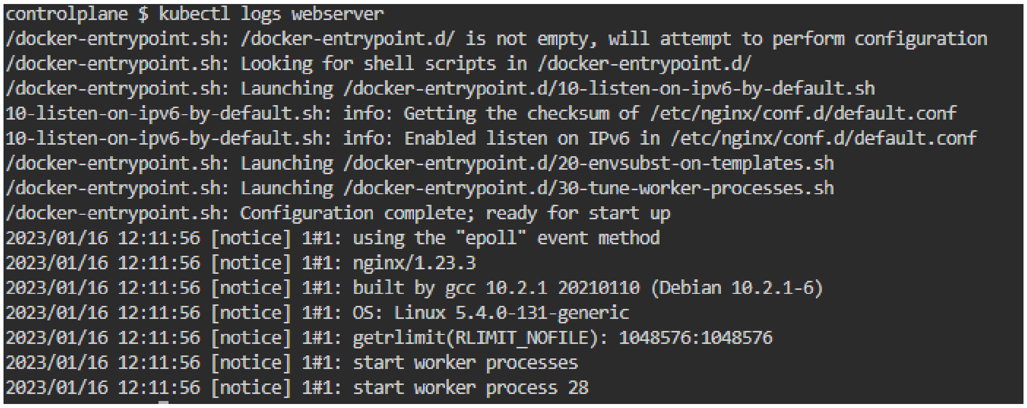
10- Run cat /etc/os-release inside the container

11- Exit from the shell (/bin/bash) session



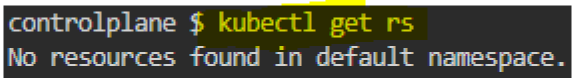
```
controlplane $ kubectl exec -it webserver -- /bin/bash
root@webserver:/# cat /etc/os-release
PRETTY_NAME="Debian GNU/Linux 11 (bullseye)"
NAME="Debian GNU/Linux"
VERSION_ID="11"
VERSION="11 (bullseye)"
VERSION_CODENAME=bullseye
ID=debian
HOME_URL="https://www.debian.org/"
SUPPORT_URL="https://www.debian.org/support"
BUG_REPORT_URL="https://bugs.debian.org/"
root@webserver:/# exit
exit
```

12- Get logs of pod, what are logs and what they are used for?



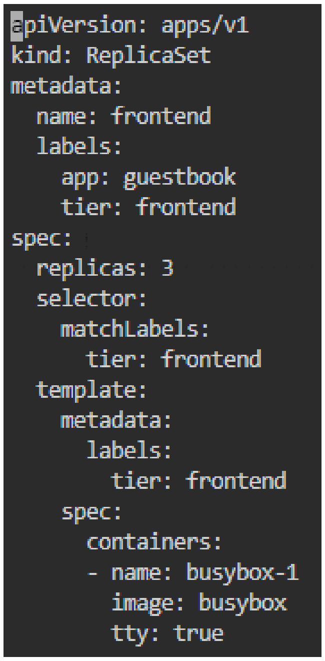
```
controlplane $ kubectl logs webserver
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2023/01/16 12:11:56 [notice] 1#1: using the "epoll" event method
2023/01/16 12:11:56 [notice] 1#1: nginx/1.23.3
2023/01/16 12:11:56 [notice] 1#1: built by gcc 10.2.1 20210110 (Debian 10.2.1-6)
2023/01/16 12:11:56 [notice] 1#1: OS: Linux 5.4.0-131-generic
2023/01/16 12:11:56 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2023/01/16 12:11:56 [notice] 1#1: start worker processes
2023/01/16 12:11:56 [notice] 1#1: start worker process 28
```

13- How many ReplicaSets exist on the system?

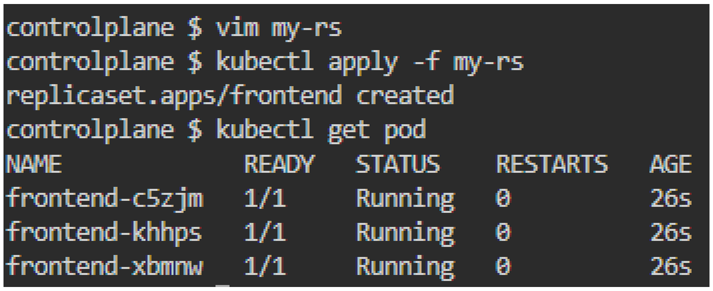


```
controlplane $ kubectl get rs
No resources found in default namespace.
```

14- create a ReplicaSet with name= replica-set-1 image= busybox replicas= 3

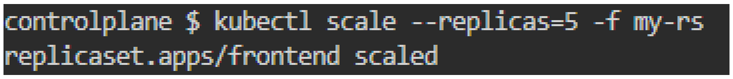


```
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: frontend
  labels:
    app: guestbook
    tier: frontend
spec:
  replicas: 3
  selector:
    matchLabels:
      tier: frontend
  template:
    metadata:
      labels:
        tier: frontend
    spec:
      containers:
      - name: busybox-1
        image: busybox
        tty: true
```



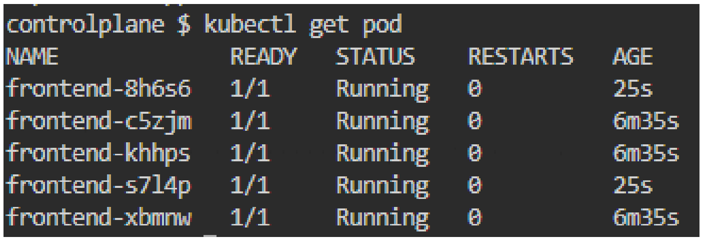
```
controlplane $ vim my-rs
controlplane $ kubectl apply -f my-rs
replicaset.apps/frontend created
controlplane $ kubectl get pod
NAME             READY   STATUS    RESTARTS   AGE
frontend-c5zjm   1/1     Running   0          26s
frontend-khhps   1/1     Running   0          26s
frontend-xbmnw   1/1     Running   0          26s
```

15- Scale the ReplicaSet replica-set-1 to 5 PODs.



```
controlplane $ kubectl scale --replicas=5 -f my-rs
replicaset.apps/frontend scaled
```

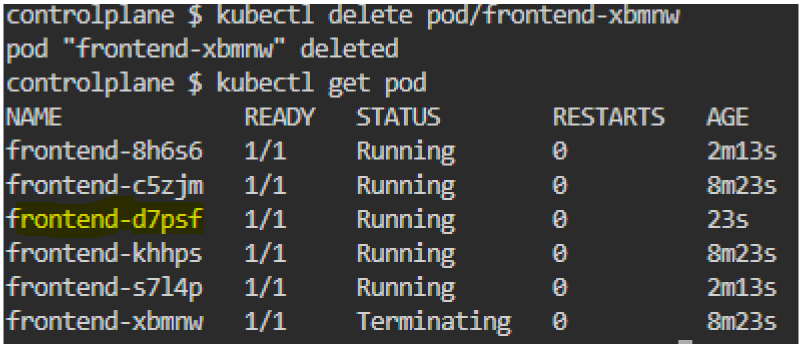
16- How many PODs are READY in the `replica-set-1`?



```
controlplane $ kubectl get pod
NAME             READY   STATUS    RESTARTS   AGE
frontend-8h6s6   1/1     Running   0          25s
frontend-c5zjm   1/1     Running   0          6m35s
frontend-khhps   1/1     Running   0          6m35s
frontend-s7l4p   1/1     Running   0          25s
frontend-xbmnw   1/1     Running   0          6m35s
```

Ans: 5 pods

17- Delete any one of the 5 PODs then check How many PODs exist now? Why are there still 5 PODs, even after you deleted one?



```
controlplane $ kubectl delete pod/frontend-xbmnw
pod "frontend-xbmnw" deleted
controlplane $ kubectl get pod
NAME             READY   STATUS        RESTARTS   AGE
frontend-8h6s6   1/1     Running       0          2m13s
frontend-c5zjm   1/1     Running       0          8m23s
frontend-d7psf   1/1     Running       0          23s
frontend-khhps   1/1     Running       0          8m23s
frontend-s7l4p   1/1     Running       0          2m13s
frontend-xbmnw   1/1     Terminating   0          8m23s
```

Ans:5 pods, because Replica set make sure that 5 pods running over all time. If one terminated, it creates another pod