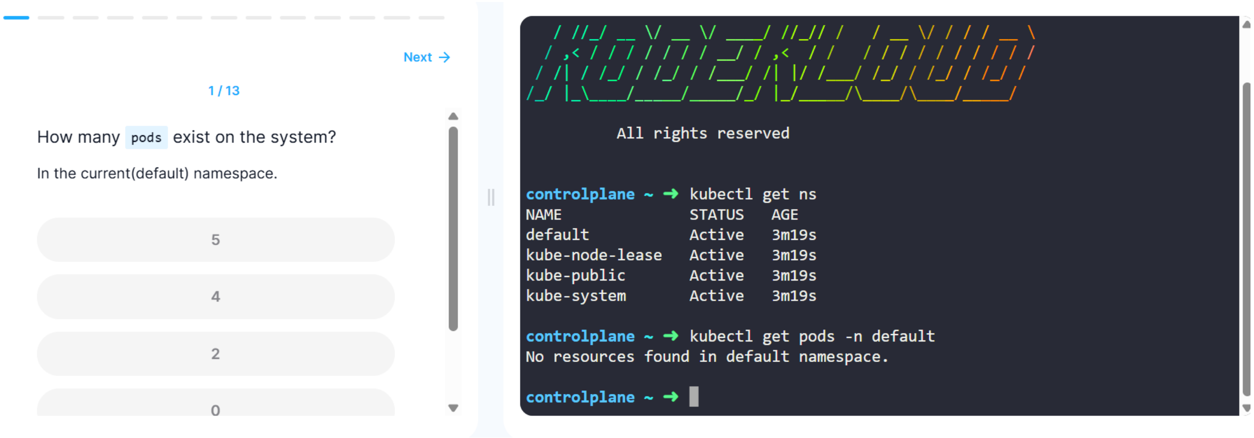
Lab2



How many `pods` exist on the system?

In the current(default) namespace.

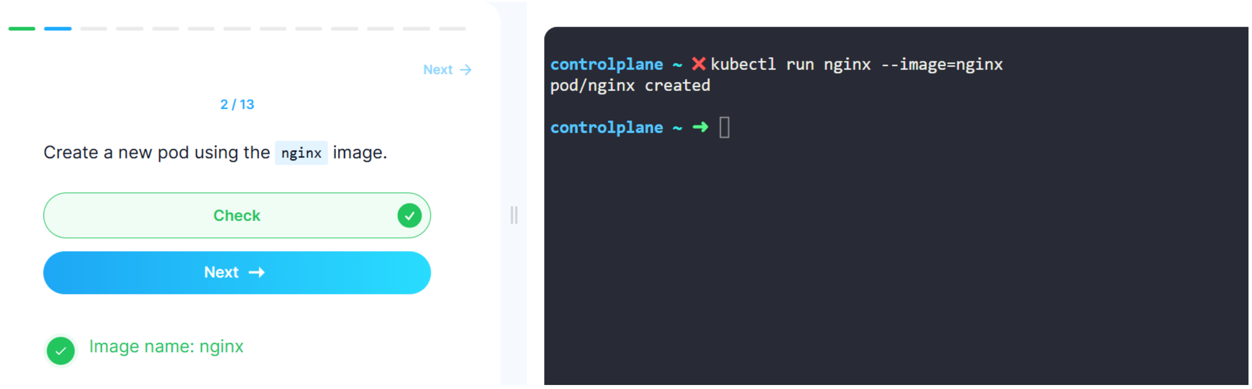
```
   / //_/ __ \/ __ \/ ____/ //_// /   / __ \/ / / / __ \
  / ,< / / / / / / / __/ / ,<  / /   / / / / / / / / / /
 / /| / /_/ / /_/ / /___/ /| |/ /___/ /_/ / /_/ / /_/ /
/_/ |_\____/_____/_____/_/ |_/_____/\____/\____/_____/

         All rights reserved


controlplane ~ ➜ kubectl get ns
NAME              STATUS   AGE
default           Active   3m19s
kube-node-lease   Active   3m19s
kube-public       Active   3m19s
kube-system       Active   3m19s

controlplane ~ ➜ kubectl get pods -n default
No resources found in default namespace.

controlplane ~ ➜
```

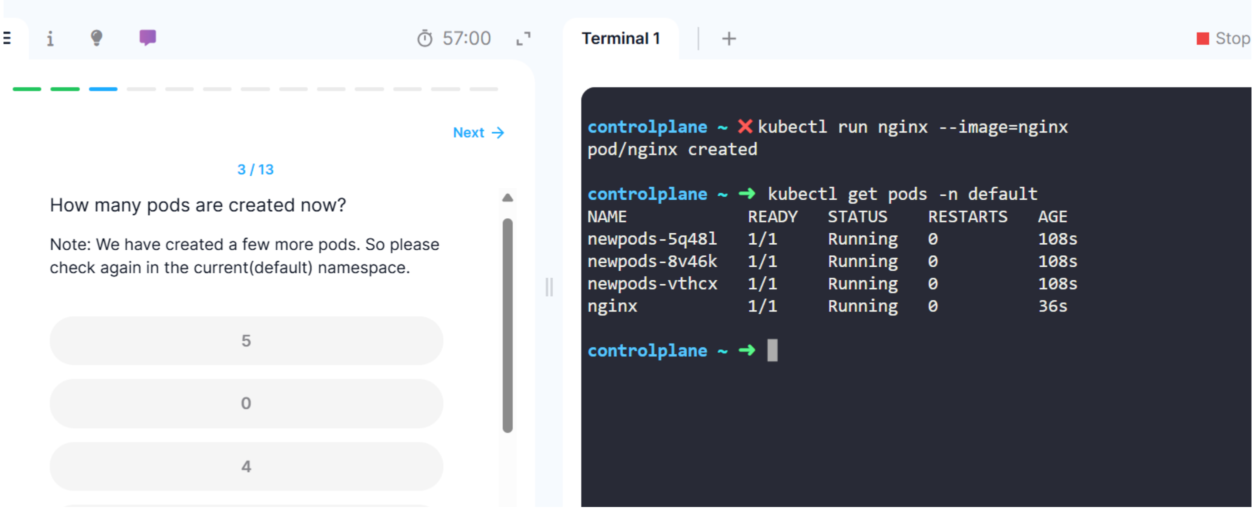


Create a new pod using the `nginx` image.

✓ Image name: nginx

```
controlplane ~ ✗ kubectl run nginx --image=nginx
pod/nginx created

controlplane ~ ➜
```



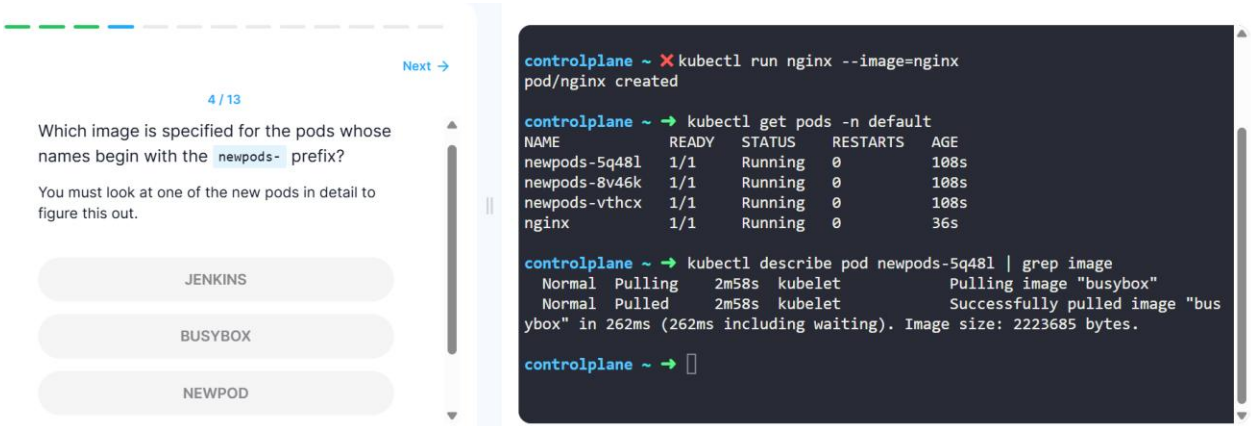
How many pods are created now?

Note: We have created a few more pods. So please check again in the current(default) namespace.

```
controlplane ~ ✗ kubectl run nginx --image=nginx
pod/nginx created

controlplane ~ ➜  kubectl get pods -n default
NAME            READY   STATUS    RESTARTS   AGE
newpods-5q48l   1/1     Running   0          108s
newpods-8v46k   1/1     Running   0          108s
newpods-vthcx   1/1     Running   0          108s
nginx           1/1     Running   0          36s

controlplane ~ ➜
```



4 / 13

Which image is specified for the pods whose names begin with the `newpods-` prefix?

You must look at one of the new pods in detail to figure this out.

JENKINS

BUSYBOX

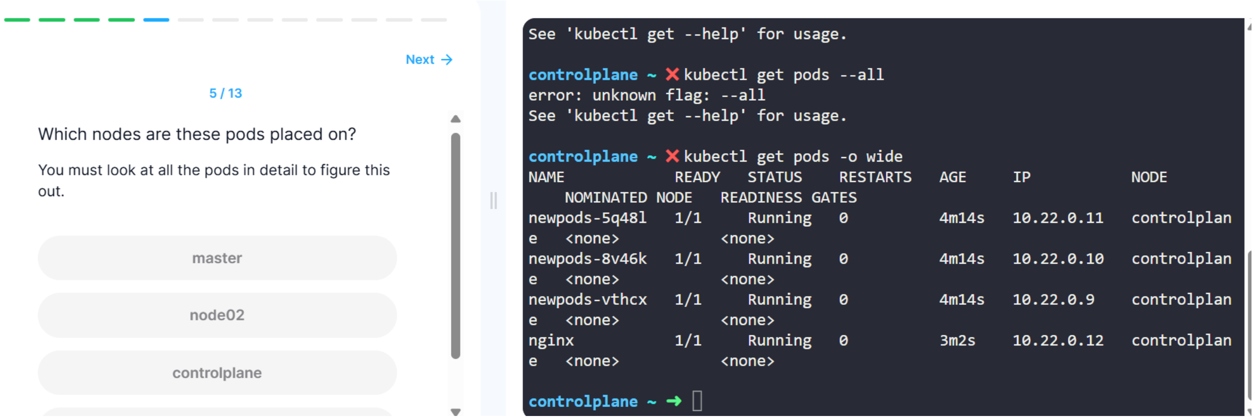
NEWPOD

```
controlplane ~ ✗ kubectl run nginx --image=nginx
pod/nginx created

controlplane ~ ➜ kubectl get pods -n default
NAME            READY   STATUS    RESTARTS   AGE
newpods-5q48l   1/1     Running   0          108s
newpods-8v46k   1/1     Running   0          108s
newpods-vthcx   1/1     Running   0          108s
nginx           1/1     Running   0          36s

controlplane ~ ➜ kubectl describe pod newpods-5q48l | grep image
  Normal  Pulling    2m58s  kubelet            Pulling image "busybox"
  Normal  Pulled     2m58s  kubelet            Successfully pulled image "bus
ybox" in 262ms (262ms including waiting). Image size: 2223685 bytes.

controlplane ~ ➜ 
```



5 / 13

Which nodes are these pods placed on?

You must look at all the pods in detail to figure this out.

master

node02

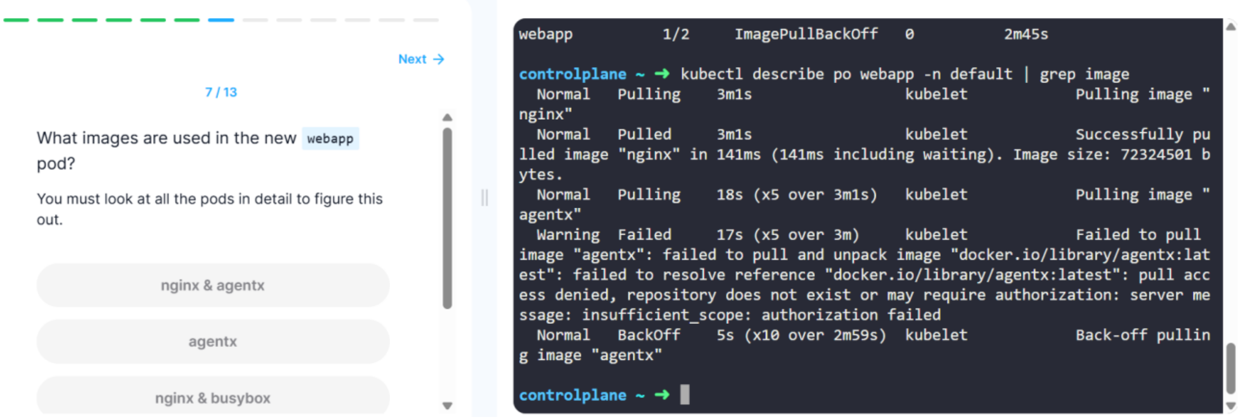
controlplane

```
See 'kubectl get --help' for usage.

controlplane ~ ✗ kubectl get pods --all
error: unknown flag: --all
See 'kubectl get --help' for usage.

controlplane ~ ✗ kubectl get pods -o wide
NAME            READY   STATUS    RESTARTS   AGE     IP           NODE
    NOMINATED NODE   READINESS GATES
newpods-5q48l   1/1     Running   0          4m14s   10.22.0.11   controlplan
e   <none>           <none>
newpods-8v46k   1/1     Running   0          4m14s   10.22.0.10   controlplan
e   <none>           <none>
newpods-vthcx   1/1     Running   0          4m14s   10.22.0.9    controlplan
e   <none>           <none>
nginx           1/1     Running   0          3m2s    10.22.0.12   controlplan
e   <none>           <none>

controlplane ~ ➜ 
```



7 / 13

What images are used in the new `webapp` pod?

You must look at all the pods in detail to figure this out.

nginx & agentx

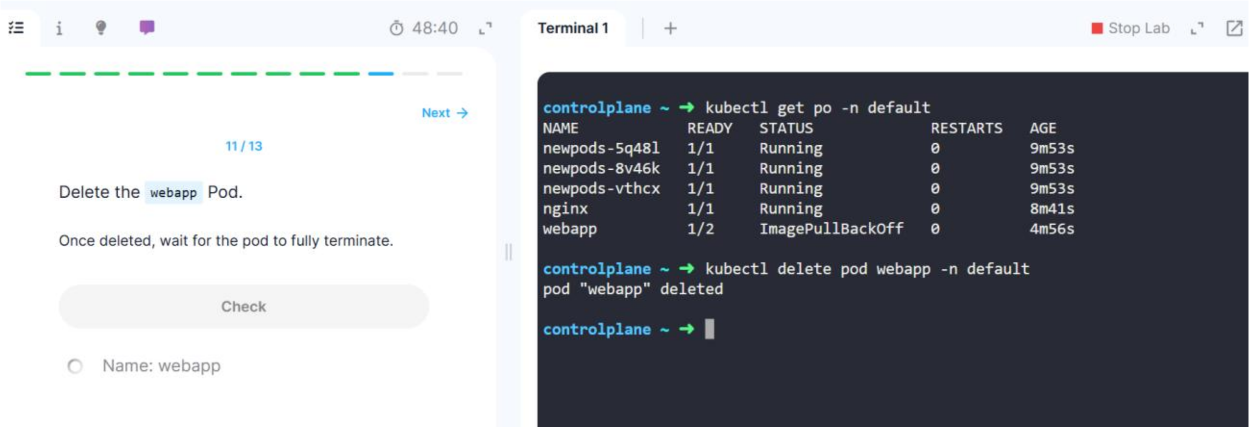
agentx

nginx & busybox

```
webapp          1/2     ImagePullBackOff   0          2m45s

controlplane ~ ➜ kubectl describe po webapp -n default | grep image
  Normal   Pulling    3m1s                kubelet            Pulling image "
nginx"
  Normal   Pulled     3m1s                kubelet            Successfully pu
lled image "nginx" in 141ms (141ms including waiting). Image size: 72324501 b
ytes.
  Normal   Pulling    18s (x5 over 3m1s)  kubelet            Pulling image "
agentx"
  Warning  Failed     17s (x5 over 3m)    kubelet            Failed to pull 
image "agentx": failed to pull and unpack image "docker.io/library/agentx:lat
est": failed to resolve reference "docker.io/library/agentx:latest": pull acc
ess denied, repository does not exist or may require authorization: server me
ssage: insufficient_scope: authorization failed
  Normal   BackOff    5s (x10 over 2m59s)  kubelet           Back-off pullin
g image "agentx"

controlplane ~ ➜ 
```



11 / 13

Delete the `webapp` Pod.

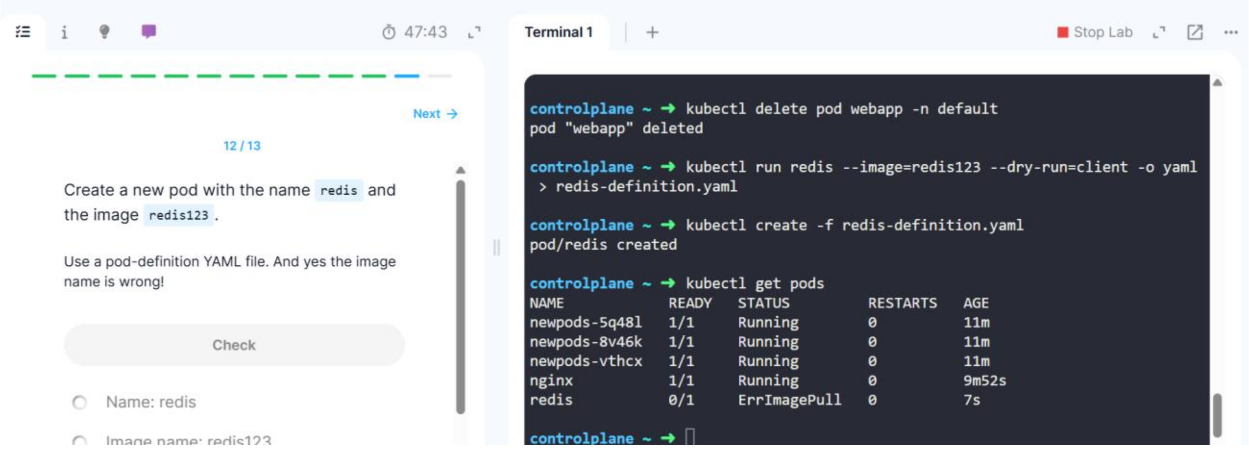
Once deleted, wait for the pod to fully terminate.

○ Name: webapp

```
controlplane ~ ➜  kubectl get po -n default
NAME             READY   STATUS             RESTARTS   AGE
newpods-5q48l    1/1     Running            0          9m53s
newpods-8v46k    1/1     Running            0          9m53s
newpods-vthcx    1/1     Running            0          9m53s
nginx            1/1     Running            0          8m41s
webapp           1/2     ImagePullBackOff   0          4m56s

controlplane ~ ➜  kubectl delete pod webapp -n default
pod "webapp" deleted

controlplane ~ ➜  
```



12 / 13

Create a new pod with the name `redis` and the image `redis123`.

Use a pod-definition YAML file. And yes the image name is wrong!

○ Name: redis

○ Image name: redis123

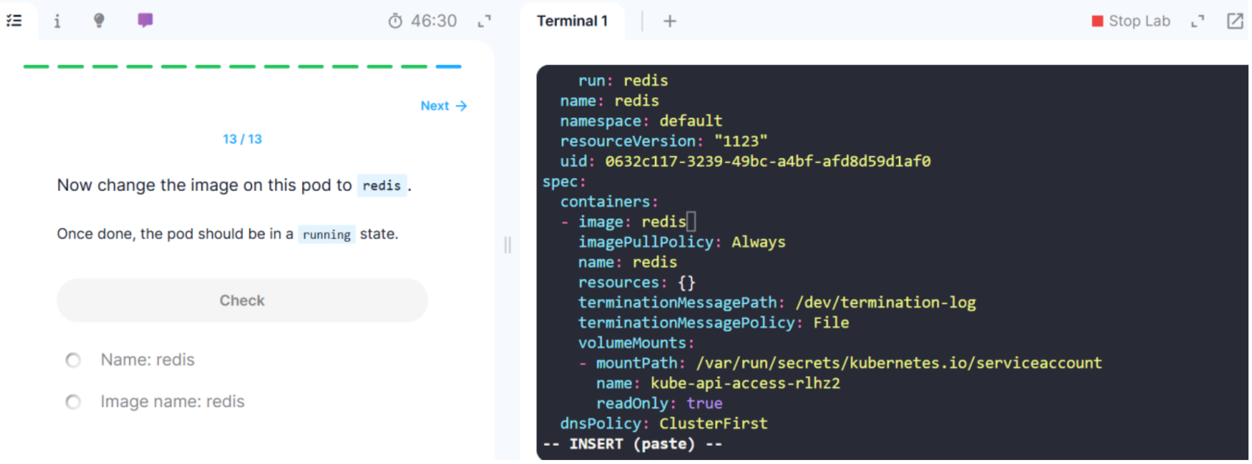
```
controlplane ~ ➜  kubectl delete pod webapp -n default
pod "webapp" deleted

controlplane ~ ➜  kubectl run redis --image=redis123 --dry-run=client -o yaml
 > redis-definition.yaml

controlplane ~ ➜  kubectl create -f redis-definition.yaml
pod/redis created

controlplane ~ ➜  kubectl get pods
NAME            READY   STATUS         RESTARTS   AGE
newpods-5q48l   1/1     Running        0          11m
newpods-8v46k   1/1     Running        0          11m
newpods-vthcx   1/1     Running        0          11m
nginx           1/1     Running        0          9m52s
redis           0/1     ErrImagePull   0          7s

controlplane ~ ➜  
```



13 / 13

Now change the image on this pod to `redis`.

Once done, the pod should be in a `running` state.

○ Name: redis

○ Image name: redis

```
    run: redis
  name: redis
  namespace: default
  resourceVersion: "1123"
  uid: 0632c117-3239-49bc-a4bf-afd8d59d1af0
spec:
  containers:
  - image: redis
    imagePullPolicy: Always
    name: redis
    resources: {}
    terminationMessagePath: /dev/termination-log
    terminationMessagePolicy: File
    volumeMounts:
    - mountPath: /var/run/secrets/kubernetes.io/serviceaccount
      name: kube-api-access-rlhz2
      readOnly: true
  dnsPolicy: ClusterFirst
-- INSERT (paste) --
```