

1- How many pods exist on the system?

Zero

2- How many Nodes exist on the system?

2 → node01 ..controlplane

3- Create a new pod with the nginx image.
Image name: nginx

pod/nginx created

4- Which nodes are these pods placed on?

node01

5- Create pod from the below yaml using kubectl apply command

```
apiVersion: v1
kind: Pod
metadata:
  name: webapp
  namespace: default
spec:
  containers:
  - image: nginx
    imagePullPolicy: Always
    name: nginx
  - image: agentx
    imagePullPolicy: Always
    name: agentx
```

pod/webapp created

6- How many containers are part of the pod webapp

```
controlplane:~$ kubectl get pod webapp
NAME      READY   STATUS    RESTARTS   AGE
webapp    1/2     ErrImagePull  0          69s
```

7- What images are used in the new webapp pod?

- Nginx - agentX

8- What is the state of the container `agentx` in the pod `webapp`

Type	Reason	Age	From	Message
Normal	Scheduled	24m	default-scheduler	Successfully assigned default/webapp to node01
Normal	Pulling	24m	kubelet	Pulling image "nginx"
Normal	Pulled	24m	kubelet	Successfully pulled image "nginx" in 5.516s (5.516s including waiting). Image size : 72403299 bytes.
Normal	Created	24m	kubelet	Created container: nginx
Normal	Started	24m	kubelet	Started container nginx
Normal	Pulling	20m (x5 over 24m)	kubelet	Pulling image "agentx"
Warning	Failed	20m (x5 over 23m)	kubelet	Failed to pull image "agentx": failed to pull and unpack image "docker.io/library/agentx:latest": failed to resolve reference "docker.io/library/agentx:latest": pull access denied, repository does not exist or may require authentication

Failed To Pull and unpack image

9- Why do you think the container `agentx` in pod `webapp` is in error?

repository does not exist or may require authorization: server message: insufficient_scope: authorization failed

10- Delete the `webapp` Pod.

-pod "webapp" deleted

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

- Name: redis
- Image Name: redis123
-

```
controlplane:~$ vim redis-pod.yaml
controlplane:~$ kubectl apply -f redis-pod.yaml
pod/redis created
```

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

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

```
controlplane:~$ kubectl delete pod redis -n default
pod "redis" deleted
controlplane:~$ vim redis-pod.yaml
controlplane:~$ kubectl apply -f redis-pod.yaml
error: error parsing redis-pod.yaml: error converting YAML to JSON: yaml: line 1: mapping values are not allowed here
controlplane:~$ vim redis-pod.yaml
controlplane:~$ vim redis-pod.yaml
controlplane:~$ kubectl apply -f redis-pod.yaml
pod/redis created
```

13- Create a pod called `my-pod` of image `nginx:alpine`

```
controlplane:~$ vim my-pod.yaml
controlplane:~$ kubectl apply -f my-pod.yaml
pod/my-pod created
```

14- Delete the pod called `my-pod`

```
controlplane:~$ kubectl delete -f my-pod.yaml  
pod "my-pod" deleted
```