1-How many static pods exist in this cluster in all namespaces?

kubectl get pods -A -o custom-columns=NAME:.metadata.name,CONTROLLER:.metadata.ownerReferences[].kind,NAMESPACE:.metadata.namespace | grep Node

2-On which nodes are the static pods created currently?

kubectl get pods -A -o custom-columns=NAME:.metadata.name,CONTROLLER:.metadata.ownerReferences[].kind,NODE\_NAME:.spec.nodeName,NAMESPACE:.metadata.namespace | grep Node

-master node contains static pods like API Server , scheduler

3- What is the path of the directory holding the static pod definition files?

/etc/kubernetes/manifests

Its configured in /var/lib/kubelet/config.yaml in worker/master node and can be changed

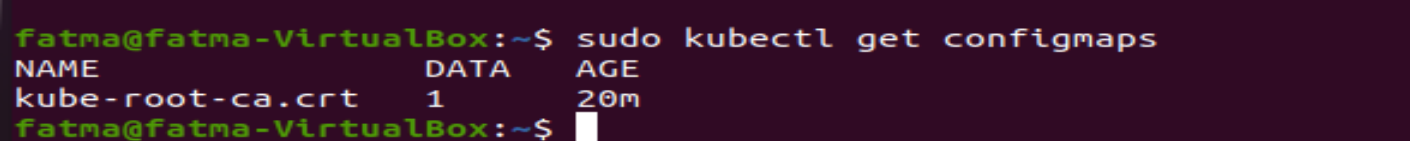
4- Create a static pod named static-busybox that uses the busybox image and the command sleep 1000



5- Edit the image on the static pod to use busybox:1.28.4

6- How many ConfigMaps exist in the environment?

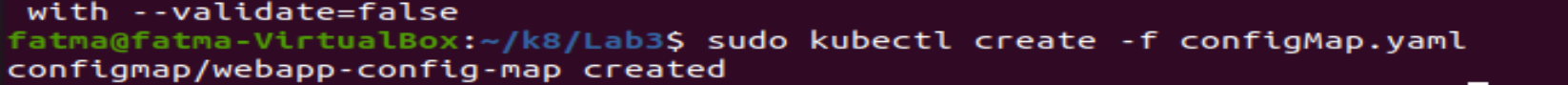
One



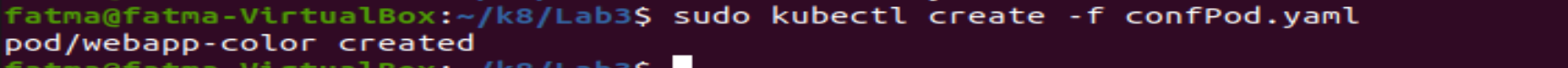
7- Create a new ConfigMap Use the spec given below.

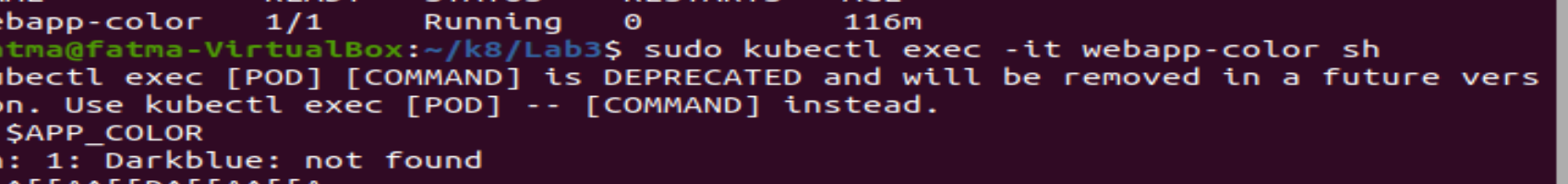
ConfigName Name: webapp-config-map

Data: APP\_COLOR=darkblue



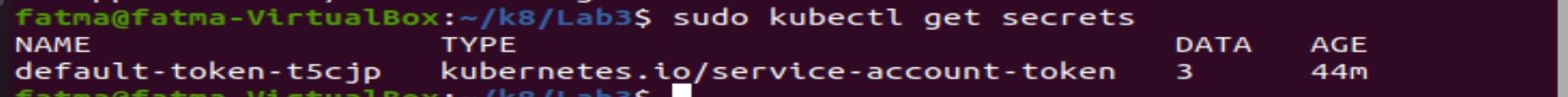
8- Create a webapp-color POD with nginx image and use the created ConfigMap

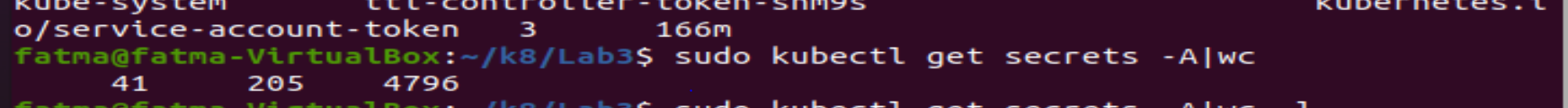




9- How many Secrets exist on the system?

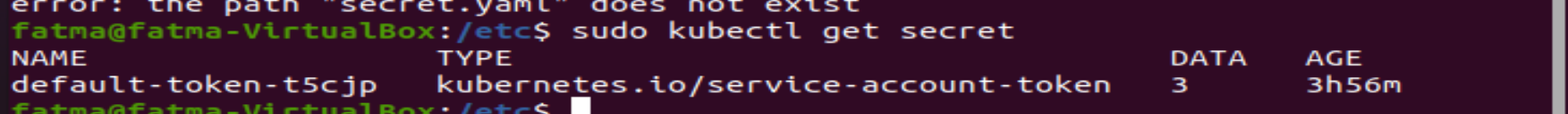
41 -One default



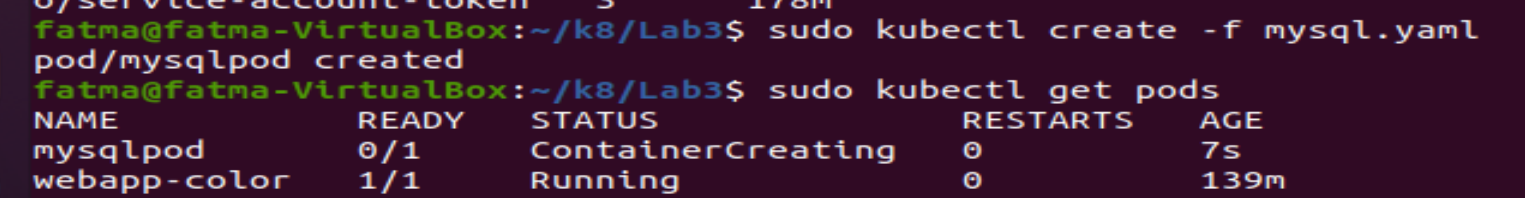


10- How many secrets are defined in the default-token secret?

3



11- create a POD called db-pod with the image mysql:5.7 then check the POD status

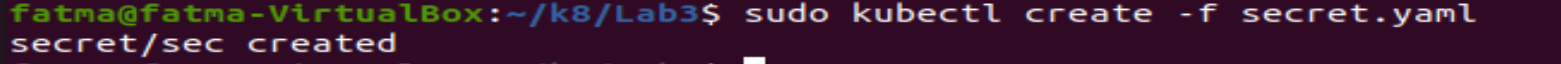


12- why the db-pod status not ready

Becouse Mysql need -> MySQL\_USERNAME-ROOTPASSWORD to lunch

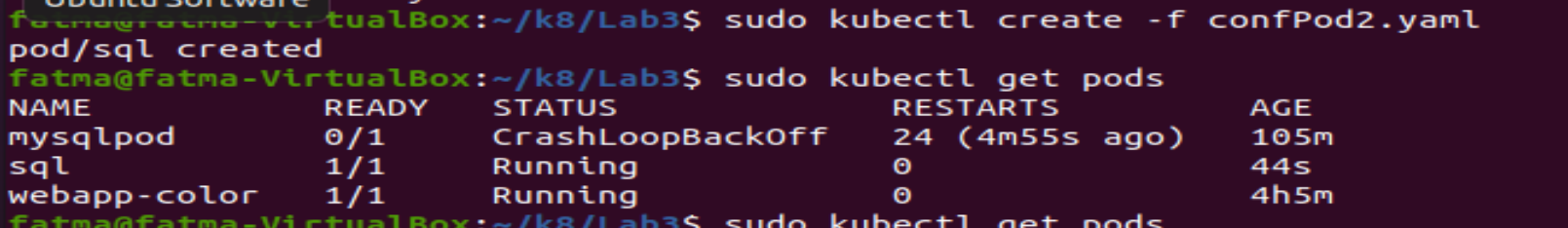
13- Create a new secret named db-secret with the data given below.  
 Secret Name: db-secret

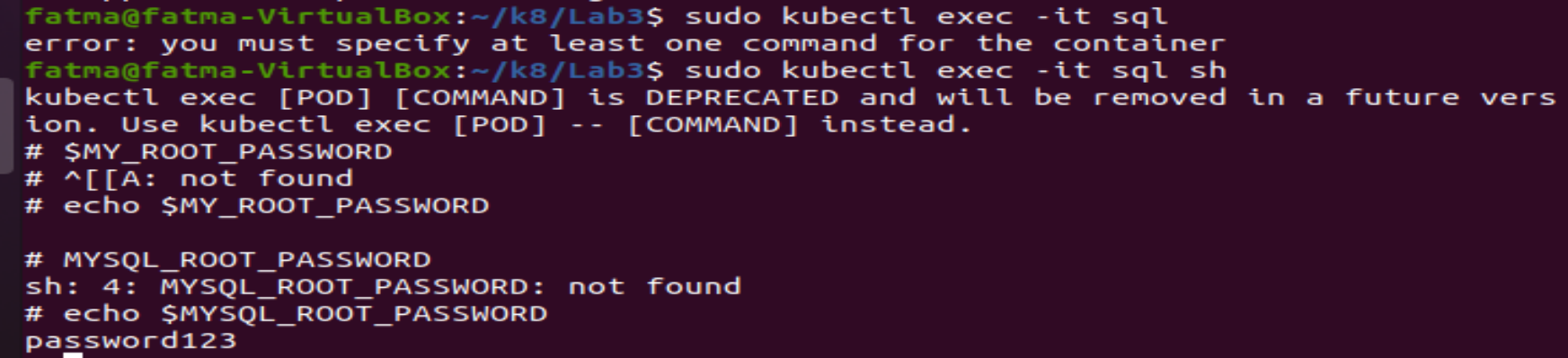
* Secret 1: MYSQL\_DATABASE=sql01
* Secret 2: MYSQL\_USER=user1
* Secret3: MYSQL\_PASSWORD=password
* Secret 4: MYSQL\_ROOT\_PASSWORD=password123



14- Configure db-pod to load environment variables from the newly created secret.

Delete and recreate the pod if required.





15- Create a multi-container pod with 2 containers.

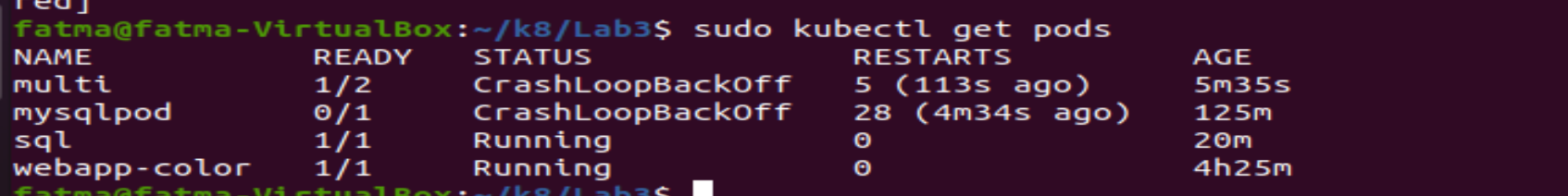
Name: yellow

Container 1 Name: lemon

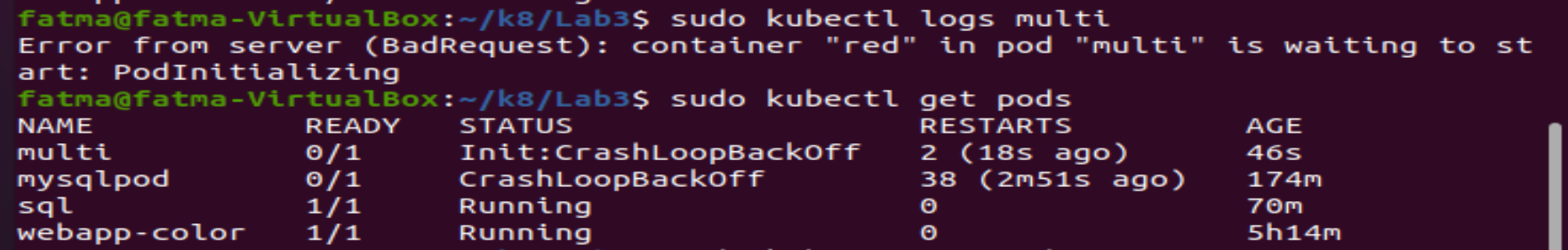
Container 1 Image: busybox

Container 2 Name: gold

Container 2 Image: redis

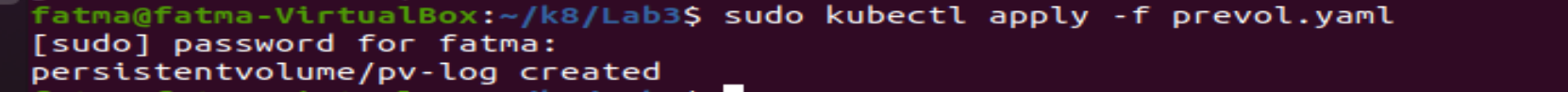


16- Create a pod red with redis image and use an initContainer that uses the busybox image and sleeps for 20 seconds



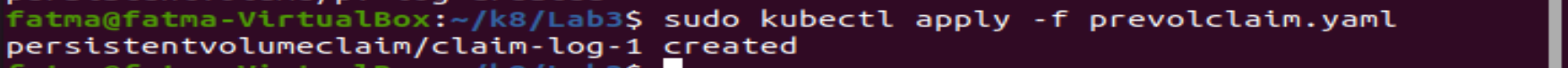
17- Create a Persistent Volume with the given specification.  
 Volume Name: pv-log

* Storage: 100Mi
* Access Modes: ReadWriteMany
* Host Path: /pv/log



18- Create a Persistent Volume Claim with the given specification.

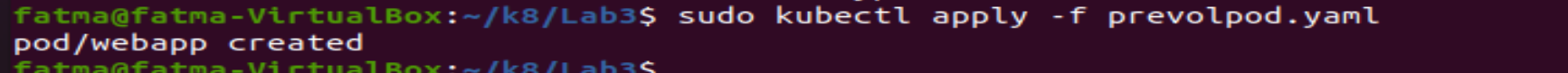
* Volume Name: claim-log-1
* Storage Request: 50Mi
* Access Modes: ReadWriteMany



19- Create a webapp pod to use the persistent volume claim as its storage.

Name: webapp

* Image Name: nginx
* Volume: PersistentVolumeClaim=claim-log-1
* Volume Mount: /var/log/nginx



20- Create a pod named volume-share-datacenter.

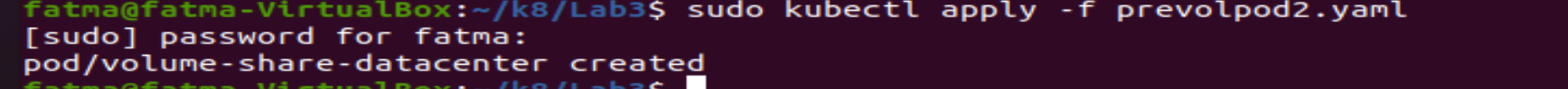
For first container, use image centos:latest, container should be named as volume-container-datacenter-1, and run a command '/bin/bash', '-c' and 'sleep 10000'. Volume volume-share should be mounted at path /tmp/news.

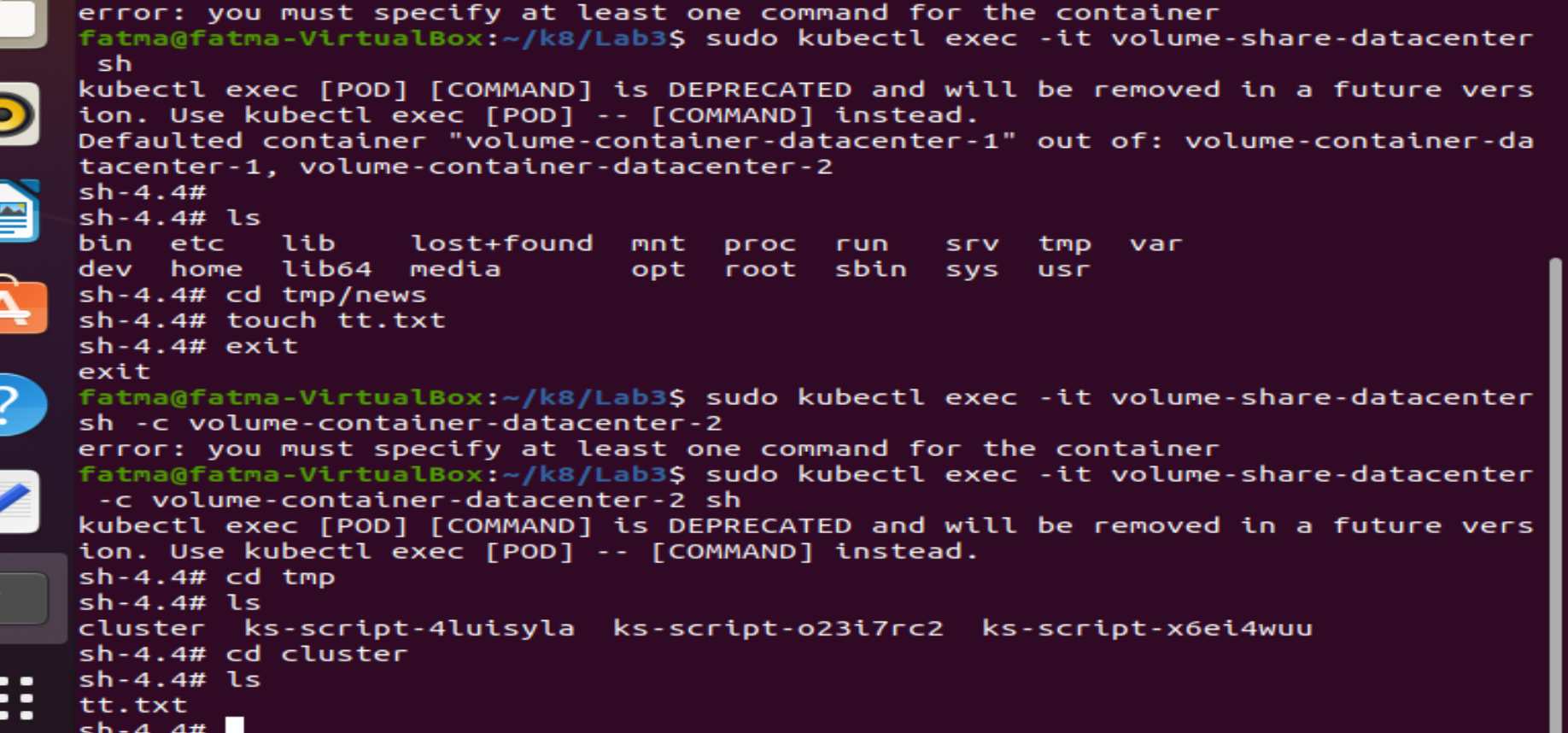
For second container, use image centos:latest, container should be named as volume-container-datacenter-2, and run a command '/bin/bash', '-c' and 'sleep 10000'. Volume volume-share should be mounted at path /tmp/cluster.

Volumes to be named as volume-share and use emptyDir: { }.

After creating the pod, exec into the first container volume-container-datacenter-1, and create a file news.txt with content Welcome from datacenter-1! under the mount path of first container /tmp/news.

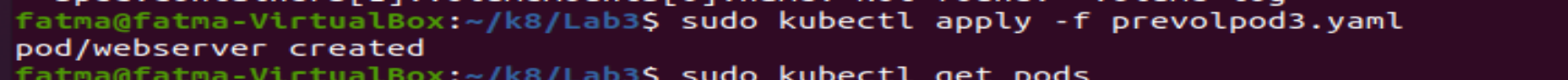
The file news.txt should be present under the mounted path /tmp/cluster of second container volume-container-datacenter-2 as they are using shared volumes.

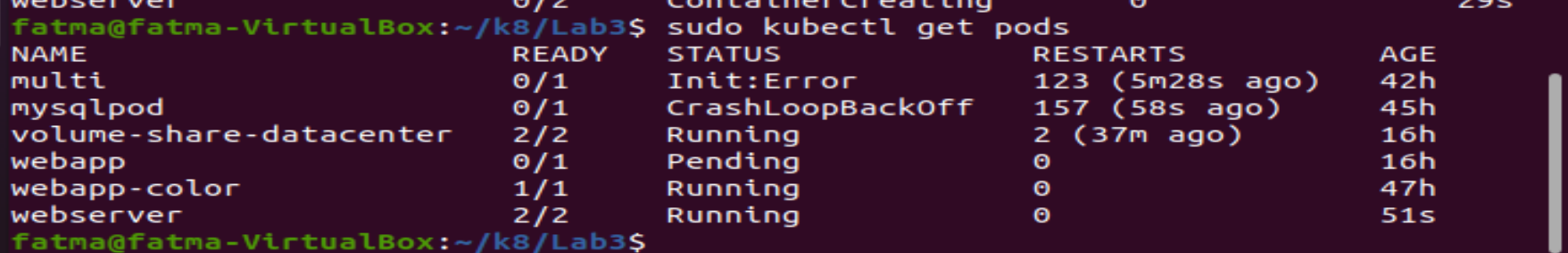




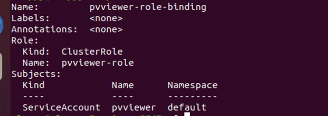
21- Create a pod named webserver. Create an emptyDir volume name: shared-logs. Create two containers from nginx and ubuntu images with latest tag only and remember to mention tag i.e. nginx:latest, nginx container name should be nginx-container and ubuntu container name should be sidecar-container on webserver pod. Add command on sidecar-container "sh","-c","while true; do cat /var/log/nginx/access.log /var/log/nginx/error.log; sleep 30; done"

Mount volume /var/log/nginx on both containers, all containers should be up and running.



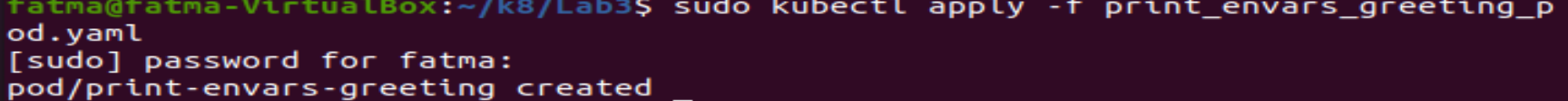


22- Create a new service account with the name pvviewer. Grant this Service account access to list all PersistentVolumes in the cluster by creating an appropriate cluster role called pvviewer-role and ClusterRoleBinding called pvviewer-role-binding.



23- Create a pod named print-envars-greeting.

1. Configure spec as, the container name should be print-env-container and use bash image.



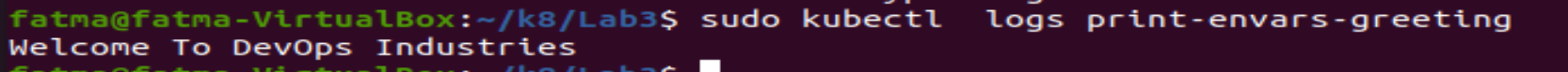
1. Create three environment variables:

a. GREETING and its value should be Welcome to

b. COMPANY and its value should be DevOps

c. GROUP and its value should be Industries

1. Use command to echo ["$(GREETING) $(COMPANY) $(GROUP)"] message.



1. You can check the output using <kubctl logs -f [ pod-name ]> command.

