1- How many ConfigMaps exist in the environment?

kubectl get configmaps --all-namespaces

2- Create a new ConfigMap.

kubectl create configmap webapp-config-map --from-literal=APP\_COLOR=darkblue

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

apiVersion: v1

kind: Pod

metadata:

name: webapp-color

spec:

containers:

- name: nginx

image: nginx

envFrom:

- configMapRef:

name: webapp-config-map

4- How many Secrets exist on the system?

kubectl get secrets --all-namespaces

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

kubectl describe secret default-token-xxxxx -n default

6- Create a POD called db-pod with the image mysql:5.7 then check the POD status.

kubectl run db-pod --image=mysql:5.7

kubectl get pods

7- Why the db-pod status is not ready?

Because MySQL requires environment variables like MYSQL\_ROOT\_PASSWORD which were not provided.

8- Create a new secret named db-secret with the data given.

kubectl create secret generic db-secret \

--from-literal=MYSQL\_DATABASE=sql01 \

--from-literal=MYSQL\_USER=user1 \

--from-literal=MYSQL\_PASSWORD=password \

--from-literal=MYSQL\_ROOT\_PASSWORD=password123

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

kubectl delete pod db-pod

# YAML:

apiVersion: v1

kind: Pod

metadata:

name: db-pod

spec:

containers:

- name: mysql

image: mysql:5.7

envFrom:

- secretRef:

name: db-secret

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

apiVersion: v1

kind: Pod

metadata:

name: yellow

spec:

containers:

- name: lemon

image: busybox

command: ["sleep", "3600"]

- name: gold

image: redis

11- Create a pod red with redis image and an initContainer.

apiVersion: v1

kind: Pod

metadata:

name: red

spec:

initContainers:

- name: init-sleep

image: busybox

command: ['sh', '-c', 'sleep 20']

containers:

- name: redis

image: redis

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

apiVersion: v1

kind: Pod

metadata:

name: print-envars-greeting

spec:

containers:

- name: print-env-container

image: bash

env:

- name: GREETING

value: "Welcome to"

- name: COMPANY

value: "DevOps"

- name: GROUP

value: "Industries"

command: ["sh", "-c", "echo "$(GREETING) $(COMPANY) $(GROUP)" && sleep 3600"]

13- Where is the default kubeconfig file located?

~/.kube/config

14- How many clusters are defined in the default kubeconfig file?

kubectl config view (check clusters section)

15- What is the user configured in the current context?

kubectl config current-context

16- Create a Persistent Volume (pv-log).

apiVersion: v1

kind: PersistentVolume

metadata:

name: pv-log

spec:

capacity:

storage: 100Mi

accessModes:

- ReadWriteMany

hostPath:

path: /pv/log

17- Create a Persistent Volume Claim (claim-log-1).

apiVersion: v1

kind: PersistentVolumeClaim

metadata:

name: claim-log-1

spec:

accessModes:

- ReadWriteMany

resources:

requests:

storage: 50Mi

18- Create a webapp pod to use the Persistent Volume Claim.

apiVersion: v1

kind: Pod

metadata:

name: webapp

spec:

containers:

- name: nginx

image: nginx

volumeMounts:

- mountPath: /var/log/nginx

name: nginx-logs

volumes:

- name: nginx-logs

persistentVolumeClaim:

claimName: claim-log-1