1- How many `ConfigMaps` exist in the environment?

```
controlplane:~$ kubectl get configmaps --all-namespaces
NAMESPACE           NAME
          DATA   AGE
default             kube-root-ca.crt
          1      36d
kube-node-lease     kube-root-ca.crt
          1      36d
kube-public         cluster-info
          2      36d
kube-public         kube-root-ca.crt
          1      36d
kube-system         canal-config
          6      36d
kube-system         coredns
          1      36d
kube-system         extension-apiserver-authentication
          6      36d
kube-system         kube-apiserver-legacy-service-account-token-
tracking   1      36d
kube-system         kube-proxy
          2      36d
kube-system         kube-root-ca.crt
          1      36d
kube-system         kubeadm-config
          1      36d
kube-system         kubelet-config
          1      36d
local-path-storage   kube-root-ca.crt
          1      36d
local-path-storage   local-path-config
          4      36d
```

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

ConfigName Name: webapp-config-map

Data: APP_COLOR=darkblue

```
controlplane:~$ kubectl create configmap webapp-config-map --from
-literal=APP_COLOR=darkblue
configmap/webapp-config-map created
```

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

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

4- How many `Secrets` exist on the system?

```
controlplane:~$ kubectl get secrets --all-namespaces
NAMESPACE     NAME                  TYPE
      DATA   AGE
kube-system   bootstrap-token-fa18uz   bootstrap.kubernetes.io/to
ken   5      36d
```

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

```
Error from server (NotFound): secrets "default-token" not found
controlplane:~$ kubectl run db-pod --image=mysql:5.7 --restart=Ne
ver
pod/db-pod created
```

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

7- why the db-pod status not ready

8- 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

```
pod/db-pod created
controlplane:~$ 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
secret/db-secret created
```

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

Delete and recreate the pod if required.

```
kubectl: command not found
controlplane:~$ kubectl delete pod db-pod
pod "db-pod" deleted
controlplane:~$
```

10- 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

```
pod "db-pod" deleted
controlplane:~$ vim yellow-pod.yaml
controlplane:~$ kubectl apply -f yellow-pod.yaml
pod/yellow created
```

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

```
controlplane:~$ vim print-envars-greeting.yaml
controlplane:~$ kubectl apply -f print-envars-greeting.yaml
pod/print-envars-greeting created
```

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

1. Configure spec as, the container name should be `print-env-container` and use `bash` image.
2. Create three environment variables:
   a. `GREETING` and its value should be "`Welcome to`"
   b. `COMPANY` and its value should be "`DevOps`"

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

c. `GROUP` and its value should be "`Industries`"

4. Use command to echo `["$(GREETING) $(COMPANY) $(GROUP)"]` message.
5. You can check the output using `<kubctl logs -f [ pod-name ]>` command.

13- Where is the default kubeconfig file located in the current environment?
14- How many clusters are defined in the default kubeconfig file?
15- What is the user configured in the current context?

16- Create a `Persistent Volume` with the given specification.

Volume Name: pv-log
Storage: 100Mi
Access Modes: ReadWriteMany
Host Path: /pv/log

17- Create a `Persistent Volume Claim` with the given specification.

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

18- 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