

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

1 pod

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

On the master node

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

-conf

/etc/coredns/Corefile

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

```
lab3 > ! static-busybox.yaml
1  apiVersion: v1
2  kind: Pod
3  metadata:
4    name: static-busybox
5    labels:
6      app: myapp
7      tier: backend
8  spec:
9    containers:
10   - name: busybox
11     image: busybox
12     command: ['sleep 1000']
```

A screenshot of a terminal window titled "lab3 — zsh — 138x5". The window shows the command "static-busybox.yaml" being run. The output lists several pods: static-busybox (0/1, RunContainerError, 0 (7s ago), 12s), test (1/1, Running, 4 (45h ago), 4d), web-app-67579dd966-pj2f7 (1/1, Running, 4 (45h ago), 3d23h), and web-app-67579dd966-svdr8 (1/1, Running, 4 (45h ago), 3d23h). The prompt "bayaderalomari@Bayaders-MacBook-Pro lab3 %" is visible at the bottom.

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

```
lab3 > ! static-busybox.yaml
1  apiVersion: v1
2  kind: Pod
3  metadata:
4    name: static-busybox
5    labels:
6      app: myapp
7      tier: backend
8  spec:
9    containers:
10   - name: busybox
11     image: busybox:1.28.4
12     command: ['sleep 1000']
```

A screenshot of a terminal window titled "lab3 — zsh — 138x5". The window shows the command "static-busybox.yaml" being run. The output lists several pods: static-busybox (0/1, CrashLoopBackOff, 4 (70s ago), 2m53s), test (1/1, Running, 4 (45h ago), 4d), web-app-67579dd966-pj2f7 (1/1, Running, 4 (45h ago), 3d23h), and web-app-67579dd966-svdr8 (1/1, Running, 4 (45h ago), 3d23h). The prompt "bayaderalomari@Bayaders-MacBook-Pro lab3 %" is visible at the bottom.

6- How many ConfigMaps exist in the environment?

```
[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl get cm --all-namespaces
NAMESPACE      NAME          DATA   AGE
default        kube-root-ca.crt 1      6d1h
finance         kube-root-ca.crt 1      5d23h
kube-node-lease kube-root-ca.crt 1      6d1h
kube-public     cluster-info    1      6d1h
kube-public     kube-root-ca.crt 1      6d1h
kube-system    coredns       1      6d1h
kube-system    extension-apiserver-authentication 6      6d1h
kube-system    kube-proxy     2      6d1h
kube-system    kube-root-ca.crt 1      6d1h
kube-system    kubeadm-config 1      6d1h
kube-system    kubelet-config-1.23 1      6d1h
bayaderalomari@Bayaders-MacBook-Pro lab3 % ]
```

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

ConfigName Name: webapp-config-map
Data: APP_COLOR=darkblue

```
[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl create cm \
webapp-config-map --from-literal=APP_COLOR=darkblue

configmap/webapp-config-map created
[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl get cm
NAME          DATA   AGE
kube-root-ca.crt 1      6d1h
webapp-config-map 1      24s
bayaderalomari@Bayaders-MacBook-Pro lab3 % ]
```

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

```
1  apiVersion: v1
2  kind: Pod
3  metadata:
4    name: webapp-color
5    labels:
6      app: myapp
7      tier: backend
8  spec:
9    containers:
10   - name: nginx
11     image: nginx:alpine
12
13  envFrom:
14  - configMapRef:
15    name: webapp-config-map
16
[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl get pods
test                1/1    Running   4 (45h ago)   4d1h
web-app-67579dd966-pj2f7 1/1    Running   4 (45h ago)   4d
web-app-67579dd966-svdr8 1/1    Running   4 (45h ago)   4d
webapp-color          1/1    Running   0           11s
bayaderalomari@Bayaders-MacBook-Pro lab3 % ]
```

9- How many Secrets exist on the system?

```
[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl get secrets
NAME          TYPE           DATA   AGE
default-token-q4v4g  kubernetes.io/service-account-token 3      6d2h
[bayaderalomari@Bayaders-MacBook-Pro lab3 % ]
```

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

1 secret

```
bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl describe secrets
Name:           default-token-q4v4g
Namespace:      default
Labels:          <none>
Annotations:    kubernetes.io/service-account.name: default
                 kubernetes.io/service-account.uid: e850b9a5-6edd-48ec-a668-a4da95bd5e35

Type:  kubernetes.io/service-account-token

Data
====

namespace:  7 bytes
token:      eyJhbGciOiJSUzI1NiIsImtpZCI6IldDUGZrWUFqV3RJWHB1b0F4WmxBTEp5ZlQtcXV5VVZrNG1PWldEQTc0ajAifQ.eyJpc
```

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

busybox-psttk8	0/1	CrashLoopBackOff	163 (2m8s ago)	6d1h
db-pod	0/1	ImagePullBackOff	0	20s
deployment-1-9456bbbf9-f2chn	1/1	Running	6 (4h36m ago)	6d1h

12- why the db-pod status not ready

Failed to pull image "mysql:5.7": rpc error: code = Unknown desc = no matching manifest for linux/arm64/v8 in the manifest list entries

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

Secret Name: db-secret
Secret 1: MYSQL_DATABASE=mysql01
Secret 2: MYSQL_USER=user1
Secret3: MYSQL_PASSWORD=password
Secret 4: MYSQL_ROOT_PASSWORD=password123

```
lab3 % cat db-secret.yaml
1  apiVersion: v1
2  kind: Secret
3  metadata:
4    name: db-secret
5
6  data:
7    MYSQL_DATABASE: c3FsMDE=
8    MYSQL_USER: dXNlcjE=
9    MYSQL_PASSWORD: cGFzc3dvcmQ=
10   MYSQL_ROOT_PASSWORD: cGFzc3dvcmQxMjM=
```

lab3 — zsh — 119x5

```
Normal  BackOff    2s (x6 over 2m1s)  kubelet      Back-off pulling image "mysql:5.7"
Warning  Failed     2s (x6 over 2m1s)  kubelet      Error: ImagePullBackOff
bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl apply -f db-secret.yaml
secret/db-secret created
bayaderalomari@Bayaders-MacBook-Pro lab3 %
```

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

Delete and recreate the pod if required.

```

lab3 % cat db-pod.yaml
 1  apiVersion: v1
 2  kind: Pod
 3  metadata:
 4    name: db-pod
 5    labels:
 6      app: myapp
 7      tier: backend
 8  spec:
 9    containers:
10      - name: mysql
11        image: mysql:5.7
12
13    envFrom:
14      - secretRef:
15        - name: db-secret
16
lab3 % kubectl apply -f db-pod.yaml --validate=false
error: error validating "db-pod.yaml": error validating data: ValidationError(Pod.spec): unknown field
k8s.api.core.v1.PodSpec; if you choose to ignore these errors, turn validation off with --validate=false
[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl apply -f db-pod.yaml --validate=false
pod/db-pod created

```

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

```

lab3 % cat yellow.yaml
 1  apiVersion: v1
 2  kind: Pod
 3  metadata:
 4    name: yellow
 5    labels:
 6      app: myapp
 7      tier: backend
 8  spec:
 9    containers:
10      - name: lemon
11        image: busybox
12      - name: gold
13        image: redis
lab3 % kubectl apply -f yellow.yaml --validate=false
web-app-67579dd966-svdr8      1/1     Running      4 (47h ago)   4d1h
webapp-color                   1/1     Running      0           86m
[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl apply -f yellow.yaml --validate=false
pod/yellow created
bayaderalomari@Bayaders-MacBook-Pro lab3 %

```

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

```

lab3 > . yellow.yaml
1  apiVersion: v1
2  kind: Pod
3  metadata:
4    name: red
5    labels:
6      app: myapp
7      tier: backend
8  spec:
9    containers:
10   - name: redis
11     image: redis
12   initContainers:
13   - name: init-myservice
14     image: busybox:1.28
15     command: ['sh', '-c', 'sleep 20']

```

```

[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl apply -f yellow.yaml --validate=false
pod/yellow created
[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl apply -f red.yaml --validate=false
pod/red created
[bayaderalomari@Bayaders-MacBook-Pro lab3 %

```

17- Create a Persistent Volume with the given specification.

Volume Name: pv-log

Storage: 100Mi

Access Modes: ReadWriteMany

Host Path: /pv/log

```

lab3 > . pv-log.yaml
1  apiVersion: v1
2  kind: Pod
3  metadata:
4    name: pv-log
5  spec:
6    accessModes:
7      - ReadWriteOnce
8    capacity:
9      storage: 100Mi
10   volumes:
11   - name: vol
12     hostPath:
13       path: /pv/log
14   containers:
15   - name: redis
16     image: redis

```

```

[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl apply -f pv-log.yaml --validate=false
The Pod "pv-log" is invalid: spec.containers: Required value
[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl apply -f pv-log.yaml --validate=false
pod/pv-log created

```

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

Volume Name: claim-log-1

Storage Request: 50Mi

Access Modes: ReadWriteMany

```
lab3 % . pvc.yaml
1  apiVersion: v1
2  kind: PersistentVolumeClaim
3  metadata:
4    name: claim-log-1
5  spec:
6    accessModes:
7      - ReadWriteMany
8
9  resources:
10   requests:
11     storage: 50Mi
```

```
[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl apply -f pv-log.yaml --validate=false
The PersistentVolume "pv-log" is invalid: spec: Required value: must specify a volume type
[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl apply -f pvc.yaml --validate=false
persistentvolumeclaim/claim-log-1 created
[bayaderalomari@Bayaders-MacBook-Pro lab3 %
```

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

```
lab3 % . webapp-pvc.yaml
1  apiVersion: v1
2  kind: Pod
3  metadata:
4    name: webapp
5
6  spec:
7    restartPolicy: Never
8    volumes:
9      - name: my-vol
10     PersistentVolumeClaim:
11       claimName: claim-log-1
12     containers:
13       - name: nginx
14         image: nginx
15
16     volumeMounts:
17       - name: my-vol
18         mountPath: /var/log/nginx
```

```
[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl apply -f webapp-pvc.yaml --validate=false
The Pod "webapp" is invalid: spec.containers[0].volumeMounts[0].name: Not found: "vol"
[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl apply -f webapp-pvc.yaml --validate=false
pod/webapp created
```

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

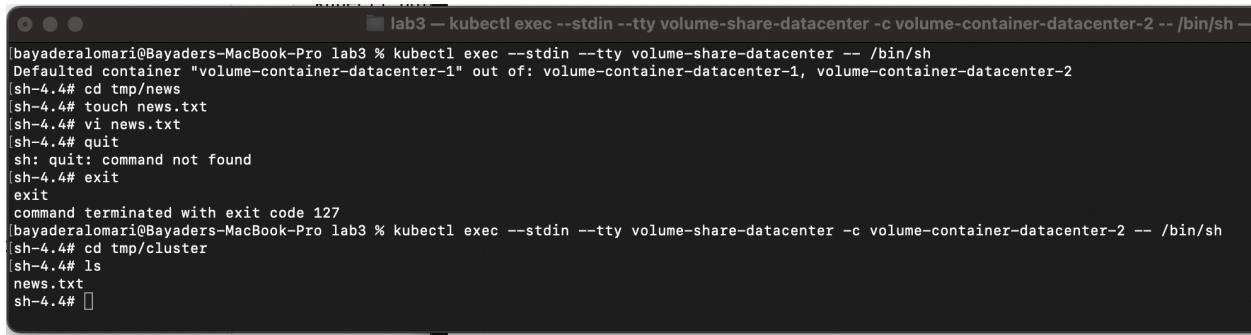
For first container, use image centos:latest, container should be named as volume-container-datacenter-1, and run a command '/bin/bash', '-c' and 'sleep 10000'. Volume volume-share should be mounted at path /tmp/news.

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.



```
lab3 — kubectl exec --stdin --tty volume-share-datacenter -c volume-container-datacenter-2 -- /bin/sh —
|bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl exec --stdin --tty volume-share-datacenter -- /bin/sh
Defaulted container "volume-container-datacenter-1" out of: volume-container-datacenter-1, volume-container-datacenter-2
|sh-4.4# cd tmp/news
|sh-4.4# touch news.txt
|sh-4.4# vi news.txt
|sh-4.4# quit
sh: quit: command not found
|sh-4.4# exit
exit
command terminated with exit code 127
|bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl exec --stdin --tty volume-share-datacenter -c volume-container-datacenter-2 -- /bin/sh
|sh-4.4# cd tmp/cluster
|sh-4.4# ls
news.txt
sh-4.4#
```

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.

```
b3 > ! webserver.yaml
1 apiVersion: v1
2 kind: Pod
3 metadata:
4   name: webserver
5   labels:
6     app: myapp
7     tier: backend
8 spec:
9   containers:
10    - name: nginx-container
11      image: nginx:latest
12      volumeMounts:
13        - name: shared-logs
14          mountPath: /var/log/nginx
15
16    - name: sidecar-container
17      image: ubuntu:latest
18      command: ['sh', '-c', 'while true; do cat /var/log/nginx/access.log /var/log/nginx/error.log; sleep 30; done']
19      volumeMounts:
20        - name: shared-logs
21          mountPath: /var/log/nginx
22 volumes:
23   - name: shared-logs
24     emptyDir: {}
```

NAME	STATUS	CONTAINERS	AGE	LAST SYNCED
webapp	Running	1/1	0	6h44m
webapp-color	Running	1/1	0	9h
webserver	Running	2/2	0	18s
yellow	NotReady	1/2	30 (5m20s ago)	7h43m

bayaderalomari@Bayaders-MacBook-Pro lab3 %

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`.

```
lab3 > ! pvviewer.yaml
1 apiVersion: v1
2 kind: ServiceAccount
3 metadata:
4   name: pvviewer
5   automountServiceAccountToken: false
```

[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl apply -f pvviewer.yaml --validate=false
serviceaccount/pvviewer created
bayaderalomari@Bayaders-MacBook-Pro lab3 %]

```
lab3 > ! pvviewer-role.yaml
1 apiVersion: rbac.authorization.k8s.io/v1
2 kind: Role
3 metadata:
4   name: pvviewer-role
5 rules:
6   - apiGroups: []
7     resources: ["PersistentVolumes"]
8     verbs: ["list"]
```

[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl apply -f pvviewer-role.yaml --validate=false
role.rbac.authorization.k8s.io/pvviewer-role created
bayaderalomari@Bayaders-MacBook-Pro lab3 %]

```

lab3 > ! pvviewer-role-binding.yaml
 1  apiVersion: rbac.authorization.k8s.io/v1
 2  kind: RoleBinding
 3  metadata:
 4    name: pvviewer-role-binding
 5  subjects:
 6  - kind: ServiceAccount
 7    name: "pvviewer"
 8  roleRef:
 9    apiGroup: rbac.authorization.k8s.io
10    kind: ClusterRole
11    name: role-grantor
12
[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl apply -f pvviewer-role-binding.yaml --validate=false
rolebinding.rbac.authorization.k8s.io/pvviewer-role-binding created]

```

23- Create a pod named print-envvars-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`
 - c. `GROUP` and its value should be `Industries`
4. Use command to echo `["$(GREETING) $(COMPANY) $(GROUP)"]` message.
5. You can check the output using `<kubectl logs -f [pod-name]>` command.

```

lab3 > ! print-envvars-greeting.yaml
 1  apiVersion: v1
 2  kind: Pod
 3  metadata:
 4    name: print-envvars-greeting
 5
 6  spec:
 7    containers:
 8    - name: print-env-container
 9      image: bash
10      command: ["echo"]
11      args: ["$(GREETING) $(COMPANY) $(GROUP)"]
12      env:
13        - name: GREETING
14          value: "Welcome to"
15        - name: COMPANY
16          value: "DevOps"
17        - name: GROUP
18          value: "Industries"
[bayaderalomari@Bayaders-MacBook-Pro lab3 % kubectl logs -f print-envvars-greeting
Welcome to DevOps Industries
Failed to create /var/run/secrets/kubernetes.io/serviceaccount/token file.]

```