1-How many DaemonSets are created in the cluster in all namespaces?

```
eman@ubuntu:~$ kubectl get  daemonsets.apps --all-namespaces
NAMESPACE     NAME         DESIRED   CURRENT   READY   UP-TO-DATE   AVAILABLE   NODE SELECTOR            AGE
kube-system   kube-proxy   1         1         1       1            1           kubernetes.io/os=linux   3d13h
eman@ubuntu:~$ kubectl get  daemonsets.apps --all-namespaces --no-headers | wc -l
1
```

2- what DaemonSets exist on the kube-system namespace?

```
eman@ubuntu:~$ kubectl get daemonsets.apps -n kube-system
NAME         DESIRED   CURRENT   READY   UP-TO-DATE   AVAILABLE   NODE SELECTOR            AGE
kube-proxy   1         1         1       1            1           kubernetes.io/os=linux   3d13h
eman@ubuntu:~$
```

3- What is the image used by the POD deployed by the kube-proxy DaemonSet

```
eman@ubuntu:~$ kubectl describe daemonsets.apps kube-proxy -n kube-system | grep "Image"
    Image:      registry.k8s.io/kube-proxy:v1.32.0
eman@ubuntu:~$
```

4- Deploy a DaemonSet for FluentD Logging. Use the given specifications.
Name: elasticsearch
Namespace: kube-system
Image: k8s.gcr.io/fluentd-elasticsearch:1.20

```
eman@ubuntu:~$ kubectl create deployment elasticsearch -n kube-system --image
=k8s.gcr.io/fluentd-elasticsearch:1.20 --dry-run=client -o yaml > q4.yaml
eman@ubuntu:~$ vi q4.yaml
eman@ubuntu:~$ kubectl apply -f q4.yaml
daemonset.apps/elasticsearch created
eman@ubuntu:~$ kubectl get daemonsets.apps
No resources found in default namespace.
eman@ubuntu:~$ kubectl get daemonsets.apps -n kube-system
NAME            DESIRED   CURRENT   READY   UP-TO-DATE   AVAILABLE   NODE SEL
ECTOR           AGE
elasticsearch   1         1         0       1            0           <none>
                47s
kube-proxy      1         1         1       1            1           kubernet
es.io/os=linux  3d14h
eman@ubuntu:~$
```

```
eman@ubuntu:~$ cat q4.yaml
apiVersion: apps/v1
kind: DaemonSet
metadata:
  creationTimestamp: null
  labels:
    app: elasticsearch
  name: elasticsearch
  namespace: kube-system
spec:
  selector:
    matchLabels:
      app: elasticsearch
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: elasticsearch
    spec:
      containers:
      - image: k8s.gcr.io/fluentd-elasticsearch:1.20
        name: fluentd-elasticsearch
        resources: {}
eman@ubuntu:~$
```

5- Deploy a pod named nginx-pod using the nginx:alpine image with the labels set to tier=backend.

```
eman@ubuntu:~$ vi q5.yaml
eman@ubuntu:~$ kubectl apply -f q5.yaml
pod/nginx-pod created
eman@ubuntu:~$ kubectl get pod
NAME                                READY   STATUS    RESTARTS      AGE
deployment-1-77496499c4-75dk9       1/1     Running   3 (41m ago)   45h
deployment-1-77496499c4-rdckk       1/1     Running   3 (41m ago)   45h
deployment-1-77496499c4-vgch9       1/1     Running   3 (41m ago)   45h
nginx                               1/1     Running   3 (41m ago)   2d5h
nginx-deployment-69b966d577-7fl5t   1/1     Running   3 (41m ago)   45h
nginx-deployment-69b966d577-f2b4g   1/1     Running   3 (41m ago)   45h
nginx-deployment-69b966d577-tg4wn   1/1     Running   3 (41m ago)   45h
nginx-pod                           1/1     Running   0             20s
redis                               1/1     Running   3 (41m ago)   2d5h
replica-set-1-f8mk2                 1/1     Running   3 (41m ago)   46h
replica-set-1-jnfnb                 1/1     Running   3 (41m ago)   47h
replica-set-1-kmx24                 1/1     Running   3 (41m ago)   46h
```

```
eman@ubuntu:~$ kubectl get pod | grep "nginx-pod"
nginx-pod                           1/1     Running   0             55s
eman@ubuntu:~$
eman@ubuntu:~$ cat q5.yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx-pod
  labels:
   tier: backend
spec:
  containers:
  - name: nginx-pod
    image: nginx:alpine
```

6- Deploy a test pod using the nginx:alpine image.

```
eman@ubuntu:~$ kubectl run test --image=nginx:alpine  --labels app=backend

pod/test created
eman@ubuntu:~$ kubectl  get pod | grep "test"

test                                1/1     Running   0             20s
eman@ubuntu:~$
```

7- Create a service backend-service to expose the backend application within the cluster on port 80.

```
eman@ubuntu:~$ vi q7.yaml
eman@ubuntu:~$ kubectl apply -f q7.yaml
service/backend-service created
eman@ubuntu:~$ kubectl get service
NAME              TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)   AGE
backend-service   ClusterIP   10.109.83.255   <none>        80/TCP    5s
kubernetes        ClusterIP   10.96.0.1       <none>        443/TCP   3d14h
```

```
eman@ubuntu:~$ cat q7.yaml
apiVersion: v1
kind: Service
metadata:
  name: backend-service
spec:
  selector:
    app: backend
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
  type: ClusterIP
```

8- try to curl the backend-service from the test pod. What is the response?

```
eman@ubuntu:~$ kubectl exec -it test  -- sh
/ # curl backend-service
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
/ # =
```

9- Create a deployment named web-app using the image nginx with 2 Replicas

```
eman@ubuntu:~$ vi q9.yaml
eman@ubuntu:~$ kubectl apply -f q9.yaml
deployment.apps/web-app created
eman@ubuntu:~$ kubectl get deployments.apps
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
deployment-1       3/3     3            3           46h
nginx-deployment   3/3     3            3           45h
web-app            2/2     2            2           19s
eman@ubuntu:~$ kubectl get deployments.apps | grep "web-app"
web-app            2/2     2            2           43s
eman@ubuntu:~$ cat q9.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: web-app
  labels:
    app: nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx
```

10- Expose the web-app as service web-app-service application on port 80 and nodeport 30082 on the nodes on the cluster

```
eman@ubuntu:~$ vi q10.yaml
eman@ubuntu:~$ kubectl apply -f q10.yaml
service/web-app-service created
eman@ubuntu:~$ kubectl get service
NAME              TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)        AGE
backend-service   ClusterIP   10.109.83.255   <none>        80/TCP         27m
kubernetes        ClusterIP   10.96.0.1       <none>        443/TCP        3d15h
web-app-service   NodePort    10.108.66.148   <none>        80:30082/TCP   5s
eman@ubuntu:~$
```

```
eman@ubuntu:~$ cat q10.yaml
apiVersion: v1
kind: Service
metadata:
  name: web-app-service
spec:
  selector:
    app: web-app
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
      nodePort: 30082
  type: NodePort
```

11- access the web app from the node

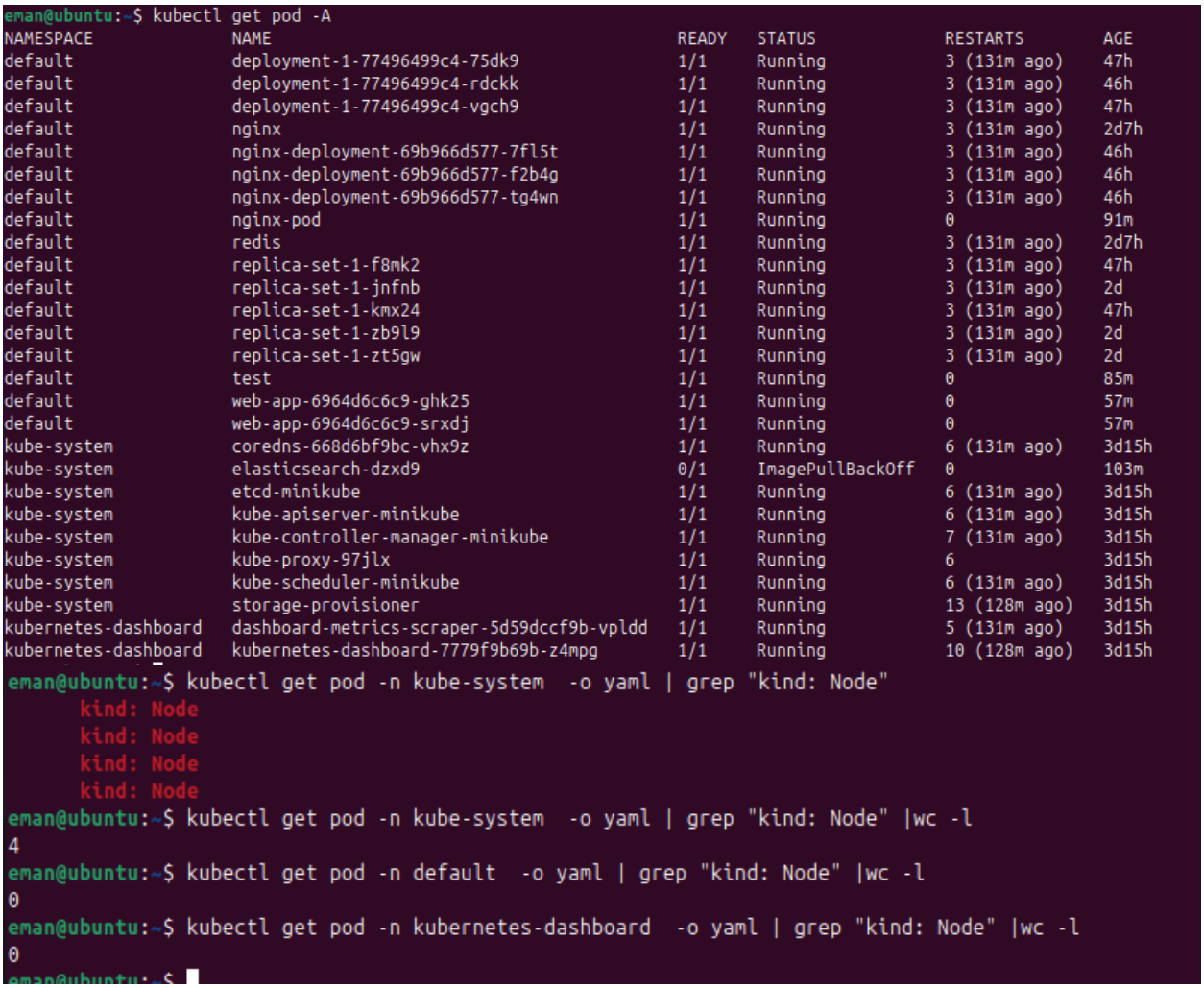
```
eman@ubuntu:~$ kubectl get nodes -o wide
NAME       STATUS   ROLES           AGE     VERSION   INTERNAL-IP    EXTERNAL-IP   OS-IMAGE             KERN
EL-VERSION      CONTAINER-RUNTIME
minikube   Ready    control-plane   3d16h   v1.32.0   192.168.49.2   <none>        Ubuntu 22.04.5 LTS   6.11
.0-17-generic   docker://27.4.1
eman@ubuntu:~$ kubectl get service web-app-service
NAME              TYPE       CLUSTER-IP      EXTERNAL-IP   PORT(S)        AGE
web-app-service   NodePort   10.108.66.148   <none>        80:30082/TCP   80m
eman@ubuntu:~$
```

```
eman@ubuntu:~$ curl http://192.168.49.2:30082
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
```

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



```
eman@ubuntu:~$ kubectl get pod -A
NAMESPACE              NAME                                         READY   STATUS             RESTARTS         AGE
default                deployment-1-77496499c4-75dk9                1/1     Running            3 (131m ago)     47h
default                deployment-1-77496499c4-rdckk                1/1     Running            3 (131m ago)     46h
default                deployment-1-77496499c4-vgch9                1/1     Running            3 (131m ago)     47h
default                nginx                                        1/1     Running            3 (131m ago)     2d7h
default                nginx-deployment-69b966d577-7fl5t            1/1     Running            3 (131m ago)     46h
default                nginx-deployment-69b966d577-f2b4g            1/1     Running            3 (131m ago)     46h
default                nginx-deployment-69b966d577-tg4wn            1/1     Running            3 (131m ago)     46h
default                nginx-pod                                    1/1     Running            0                91m
default                redis                                        1/1     Running            3 (131m ago)     2d7h
default                replica-set-1-f8mk2                          1/1     Running            3 (131m ago)     47h
default                replica-set-1-jnfnb                          1/1     Running            3 (131m ago)     2d
default                replica-set-1-kmx24                          1/1     Running            3 (131m ago)     47h
default                replica-set-1-zb9l9                          1/1     Running            3 (131m ago)     2d
default                replica-set-1-zt5gw                          1/1     Running            3 (131m ago)     2d
default                test                                         1/1     Running            0                85m
default                web-app-6964d6c6c9-ghk25                     1/1     Running            0                57m
default                web-app-6964d6c6c9-srxdj                     1/1     Running            0                57m
kube-system            coredns-668d6bf9bc-vhx9z                     1/1     Running            6 (131m ago)     3d15h
kube-system            elasticsearch-dzxd9                          0/1     ImagePullBackOff   0                103m
kube-system            etcd-minikube                                1/1     Running            6 (131m ago)     3d15h
kube-system            kube-apiserver-minikube                      1/1     Running            6 (131m ago)     3d15h
kube-system            kube-controller-manager-minikube             1/1     Running            7 (131m ago)     3d15h
kube-system            kube-proxy-97jlx                             1/1     Running            6                3d15h
kube-system            kube-scheduler-minikube                      1/1     Running            6 (131m ago)     3d15h
kube-system            storage-provisioner                          1/1     Running            13 (128m ago)    3d15h
kubernetes-dashboard   dashboard-metrics-scraper-5d59dccf9b-vpldd   1/1     Running            5 (131m ago)     3d15h
kubernetes-dashboard   kubernetes-dashboard-7779f9b69b-z4mpg        1/1     Running            10 (128m ago)    3d15h
eman@ubuntu:~$ kubectl get pod -n kube-system  -o yaml | grep "kind: Node"
      kind: Node
      kind: Node
      kind: Node
      kind: Node
eman@ubuntu:~$ kubectl get pod -n kube-system  -o yaml | grep "kind: Node" |wc -l
4
eman@ubuntu:~$ kubectl get pod -n default  -o yaml | grep "kind: Node" |wc -l
0
eman@ubuntu:~$ kubectl get pod -n kubernetes-dashboard  -o yaml | grep "kind: Node" |wc -l
0
eman@ubuntu:~$
```

13-On which nodes are the static pods created currently? (minikube)

```
eman@ubuntu:~$ kubectl get pod -n kube-system  -o yaml | grep "nodeName"
    nodeName: minikube
    nodeName: minikube
    nodeName: minikube
    nodeName: minikube
    nodeName: minikube
            fieldPath: spec.nodeName
    nodeName: minikube
    nodeName: minikube
    nodeName: minikube
```

```
eman@ubuntu:~$ kubectl get pod -n kube-system  -o yaml | grep "kind: Node" |wc -l
4
```