Create a StatefulSet named web-statefulset with 2 replicas using the nginx image.
The StatefulSet should have a Headless Service named web-service :

eb-statefulset.yml > {} spec > {} template > {} spe
io.k8s.api.apps.v1.StatefulSet (v1@statefulset.json)

```yaml
apiVersion: apps/v1
kind: StatefulSet
metadata:
  name: web-statefulset
spec:
  serviceName: web-service
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx
```

eb-service.yml > {} spec > {} selector
io.k8s.api.core.v1.Service (v1@service.json)

```yaml
apiVersion: v1
kind: Service
metadata:
  name: web-service
spec:
  clusterIP: None
  selector:
    app: nginx
```

```
mohamedharoon@mohamedharoon:~/Desktop/ITI/18-k8s/labs/lab2$ sudo kubectl apply -f ./web-statefulset.yml
statefulset.apps/web-statefulset unchanged
mohamedharoon@mohamedharoon:~/Desktop/ITI/18-k8s/labs/lab2$ sudo kubectl apply -f ./web-service.yml
service/web-service created
mohamedharoon@mohamedharoon:~/Desktop/ITI/18-k8s/labs/lab2$ sudo kubectl get svc
NAME          TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes    ClusterIP   10.96.0.1    <none>        443/TCP   5d1h
web-service   ClusterIP   None         <none>        7s
mohamedharoon@mohamedharoon:~/Desktop/ITI/18-k8s/labs/lab2$ sudo kubectl get pods
NAME                READY   STATUS    RESTARTS   AGE
web-statefulset-0   1/1     Running   0          76s
web-statefulset-1   1/1     Running   0          74s
```

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

```
mohamedharoon@mohamedharoon:~/Desktop/ITI/18-k8s/labs/lab2$ sudo kubectl get daemonsets --all-namespaces
NAMESPACE      NAME              DESIRED   CURRENT   READY   UP-TO-DATE   AVAILABLE   NODE SELECTOR            AGE
kube-flannel   kube-flannel-ds   2         2         2       2            2           <none>                   5d1h
kube-system    kube-proxy        2         2         2       2            2           kubernetes.io/os=linux   5d1h
mohamedharoon@mohamedharoon:~/Desktop/ITI/18-k8s/labs/lab2$
```

Create a DaemonSet named “nginx” with image “nginx”.

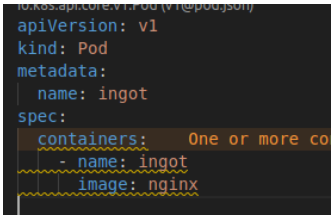
io.k8s.api.apps.v1.DaemonSet (v1@daemonset.j

```yaml
apiVersion: apps/v1
kind: DaemonSet
metadata:
  name: nginx
spec:
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx
```

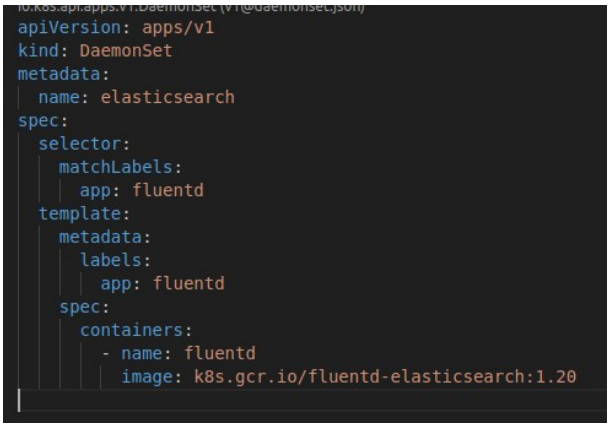
How many pods have been created within the nginx DaemonSet and why?

The number of pods created by a DaemonSet equals the number of nodes in the cluster because a DaemonSet ensures that a copy of a pod runs on each node .

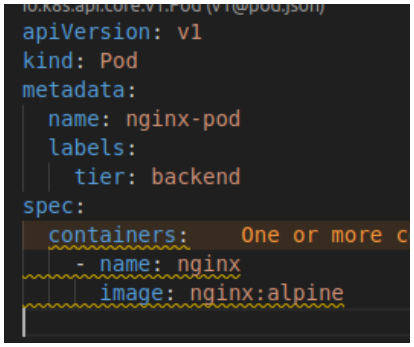
Create a pod named “ingot”



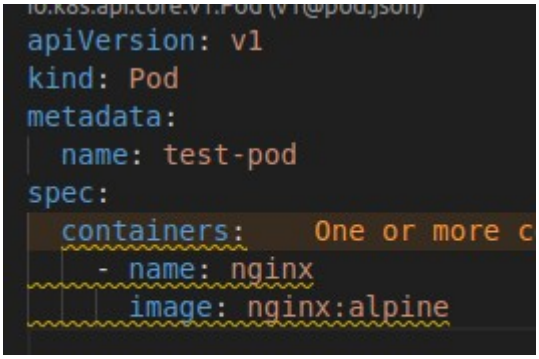
Deploy a DaemonSet for FluentD Logging. Use the given specifications.
➔ Name: elasticsearch
➔ Image: k8s.gcr.io/fluentd-elasticsearch:1.20



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



Deploy a test pod using the nginx:alpine image.



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

```yaml
apiVersion: v1
kind: Service
metadata:
  name: backend-service
spec:
  selector:
    tier: backend
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
```

```
mohamedharoon@mohamedharoon:~/Desktop/ITI/18-k8s/labs/lab2$ sudo kubectl apply -f ./backend-service.yml
service/backend-service created
mohamedharoon@mohamedharoon:~/Desktop/ITI/18-k8s/labs/lab2$
```

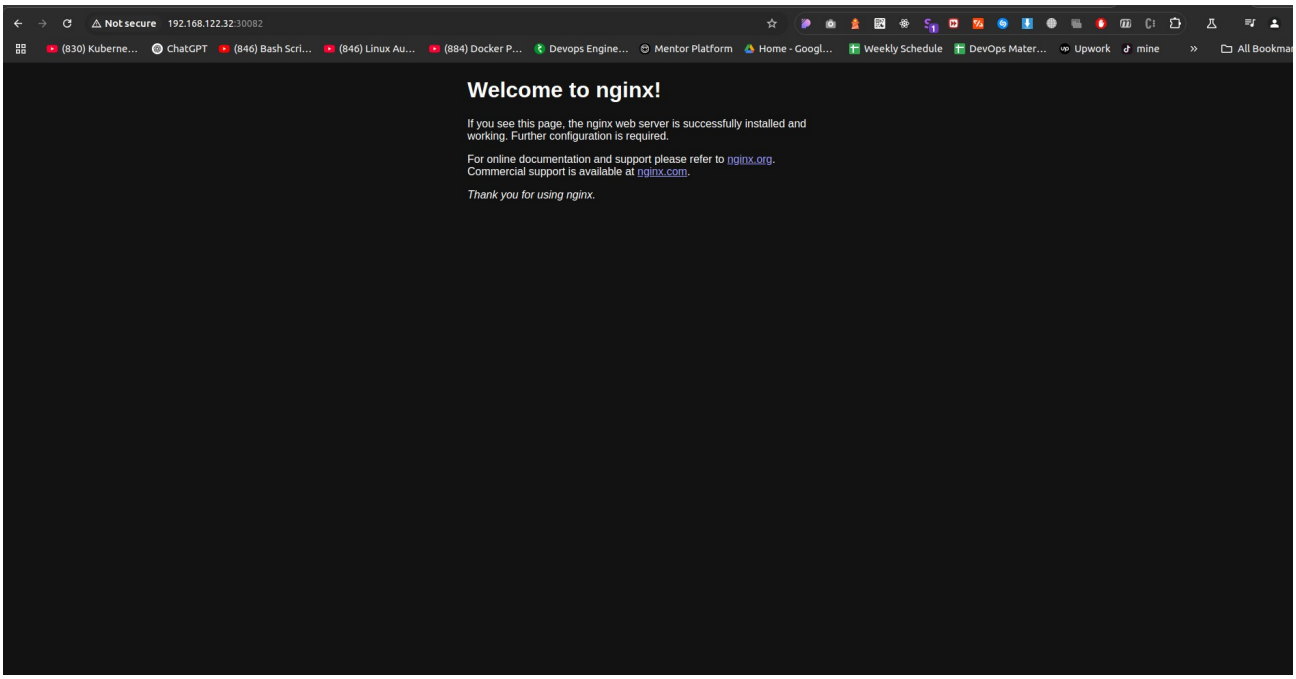
Create a deployment named web-app using the image nginx with 2 replicas What is the response?

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: web-app
spec:
  replicas: 2
  selector:
    matchLabels:
      app: web-app
  template:
    metadata:
      labels:
        app: web-app
    spec:
      containers:    One or more
        - name: nginx
          image: nginx
```

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

```yaml
apiVersion: v1
kind: Service
metadata:
  name: web-app-service
spec:
  type: NodePort
  selector:
    app: web-app
  ports:
    - port: 80
      targetPort: 80
      nodePort: 30082
```

13. Access the web app from the node



14. Cread a deployment nginx with pod labels
➔ app:nginx
➔ tier:frontend

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx
spec:
  replicas: 1
  selector:
    matchLabels:
      app: nginx
      tier: frontend
  template:
    metadata:
      labels:
        app: nginx
        tier: frontend
    spec:
      containers:    One or more
        - name: nginx
          image: nginx
```

15. When can we use the Loadbalancer service?

A LoadBalancer service can be used when you want to expose your service to external clients. It requires a cloud provider that supports external load balancers .