1- How many Namespaces exist on the system?

```
controlplane:~$ kubectl get namespaces
NAME                 STATUS   AGE
default              Active   34d
kube-node-lease      Active   34d
kube-public          Active   34d
kube-system          Active   34d
local-path-storage   Active   34d
```

2- How many pods exist in the `kube-system` namespace?

```
controlplane:~$ kubectl get pods --namespace=kube-system
NAME                                       READY   STATUS    RESTARTS      AGE
calico-kube-controllers-fdf5f5495-dgc76    1/1     Running   2 (81m ago)   34d
canal-9hc7x                                2/2     Running   2 (81m ago)   34d
canal-b5cnm                                2/2     Running   2 (82m ago)   34d
coredns-7695687499-2vdd4                   1/1     Running   1 (82m ago)   34d
coredns-7695687499-ltw2v                   1/1     Running   1 (82m ago)   34d
etcd-controlplane                          1/1     Running   3 (81m ago)   34d
kube-apiserver-controlplane                1/1     Running   2 (81m ago)   34d
kube-controller-manager-controlplane       1/1     Running   2 (81m ago)   34d
kube-proxy-f7jnk                           1/1     Running   2 (81m ago)   34d
kube-proxy-fbkjh                           1/1     Running   1 (82m ago)   34d
kube-scheduler-controlplane                1/1     Running   2 (81m ago)   34d
```

3- Create a deployment with

Name: beta

Image: redis

Replicas: 2

Namespace: finance

Resources Requests:

CPU: .5 vcpu

Mem: 1G

Resources Limits:

CPU: 1 vcpu

Mem: 2G

```
controlplane:~$ kubectl create namespace finance
namespace/finance created
```

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: beta
  namespace: finance
spec:
  replicas: 2
  selector:
    matchLabels:
      app: beta
  template:
    metadata:
      labels:
        app: beta
    spec:
      containers:
      - name: redis
        image: redis
        resources:
          requests:
            cpu: "500m"
            memory: "1Gi"
          limits:
            cpu: "1"
            memory: "2Gi"
```

4- How many Nodes exist on the system?

```
controlplane:~$ kubectl get node
NAME           STATUS   ROLES           AGE   VERSION
controlplane   Ready    control-plane   34d   v1.32.1
node01         Ready    <none>          34d   v1.32.1
```

5- Do you see any taints on `master?`

```
controlplane:~$ kubectl describe node controlplane
Name:               controlplane
Roles:              control-plane
Labels:             beta.kubernetes.io/arch=amd64
                    beta.kubernetes.io/os=linux
                    kubernetes.io/arch=amd64
                    kubernetes.io/hostname=controlplane
                    kubernetes.io/os=linux
                    node-role.kubernetes.io/control-plane=
                    node.kubernetes.io/exclude-from-external-load-balancers=
Annotations:        flannel.alpha.coreos.com/backend-data: {"VNI":1,"VtepMAC":"c2:d0:bd:79:ad:b4"}
                    flannel.alpha.coreos.com/backend-type: vxlan
                    flannel.alpha.coreos.com/kube-subnet-manager: true
                    flannel.alpha.coreos.com/public-ip: 172.30.1.2
                    kubeadm.alpha.kubernetes.io/cri-socket: unix:///var/run/containerd/containerd.sock
                    node.alpha.kubernetes.io/ttl: 0
                    projectcalico.org/IPv4Address: 172.30.1.2/24
                    projectcalico.org/IPv4IPIPTunnelAddr: 192.168.0.1
                    volumes.kubernetes.io/controller-managed-attach-detach: true
CreationTimestamp:  Sat, 22 Mar 2025 19:51:46 +0000
Taints:             <none>
```

None

6- Apply a label `color=blue` to the master node

```
controlplane:~$ kubectl label node controlplane color=blue
node/controlplane labeled
```

7- Create a new deployment named `blue` with the `nginx` image and 3 replicas

Set Node Affinity to the deployment to place the pods on `master` only

NodeAffinity: requiredDuringSchedulingIgnoredDuringExecution
Key: color values: blue

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: blue
spec:
  replicas: 3
  selector:
    matchLabels:
      app: blue
  template:
    metadata:
      labels:
        app: blue
    spec:
      affinity:
        nodeAffinity:
          requiredDuringSchedulingIgnoredDuringExecution:
            nodeSelectorTerms:
            - matchExpressions:
              - key: color
                operator: In
                values:
                - blue
      containers:
      - name: nginx
        image: nginx
```

```
controlplane:~$ kubectl apply -f blue-deployment.yaml
deployment.apps/blue created
controlplane:~$ kubectl get pods -o wide
NAME                    READY   STATUS    RESTARTS   AGE   IP           NODE           NOMINATED NODE   READINESS GATES
blue-7bd99994c-5ppw6    1/1     Running   0          33s   192.168.0.7  controlplane   <none>           <none>
blue-7bd99994c-bmgzl    1/1     Running   0          33s   192.168.0.5  controlplane   <none>           <none>
blue-7bd99994c-d4npt    1/1     Running   0          33s   192.168.0.6  controlplane   <none>           <none>
controlplane:~$ vim blue-deployment.yaml
```