1- How many Namespaces exist on the system?

```
controlplane $ kubectl get namespaces
NAME                   STATUS   AGE
default                Active   13d
kube-node-lease        Active   13d
kube-public            Active   13d
kube-system            Active   13d
local-path-storage     Active   13d
controlplane $
```

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

```
controlplane $ kubectl get pods -n kube-system
NAME                                       READY   STATUS    RESTARTS      AGE
calico-kube-controllers-75bdb5b75d-2b6mr   1/1     Running   2 (23m ago)   13d
canal-q652m                                2/2     Running   2 (23m ago)   13d
canal-wzjz6                                2/2     Running   2 (23m ago)   13d
coredns-5c69dbb7bd-6xvhl                   1/1     Running   1 (23m ago)   13d
coredns-5c69dbb7bd-xfk7l                   1/1     Running   1 (23m ago)   13d
etcd-controlplane                          1/1     Running   2 (23m ago)   13d
kube-apiserver-controlplane                1/1     Running   2 (23m ago)   13d
kube-controller-manager-controlplane       1/1     Running   2 (23m ago)   13d
kube-proxy-dp5fn                           1/1     Running   2 (23m ago)   13d
kube-proxy-nhmtq                           1/1     Running   1 (23m ago)   13d
kube-scheduler-controlplane                1/1     Running   2 (23m ago)   13d
controlplane $
```

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

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: beta
  namespace: finance
spec:
  replicas: 2
  selector:
    matchLabels:
      app: redis
  template:
    metadata:
      labels:
        app: redis
    spec:
      containers:
      - name: redis-container
        image: redis
        resources:
          requests:
            cpu: "0.5"
            memory: "1G"
          limits:
            cpu: "1"
```

4- How many Nodes exist on the system?

```
controlplane $ kubectl get nodes
NAME           STATUS   ROLES           AGE   VERSION
controlplane   Ready    control-plane   14d   v1.30.0
node01         Ready    <none>          14d   v1.30.0
controlplane $
```

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
```

```
tach: true
CreationTimestamp:  Wed, 03 Jul 2024 18:34:54 +0000
Taints:             <none>
Unschedulable:      false
Lease:
```

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

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

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: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx-container
        image: nginx
      affinity:
        nodeAffinity:
          requiredDuringSchedulingIgnoredDuringExecution:
            nodeSelectorTerms:
            - matchExpressions:
              - key: color
                operator: In
                values:
                - blue
```

```
controlplane $ kubectl apply -f blue-deployment.yaml
deployment.apps/blue created
controlplane $
```