1- create minikube cluster

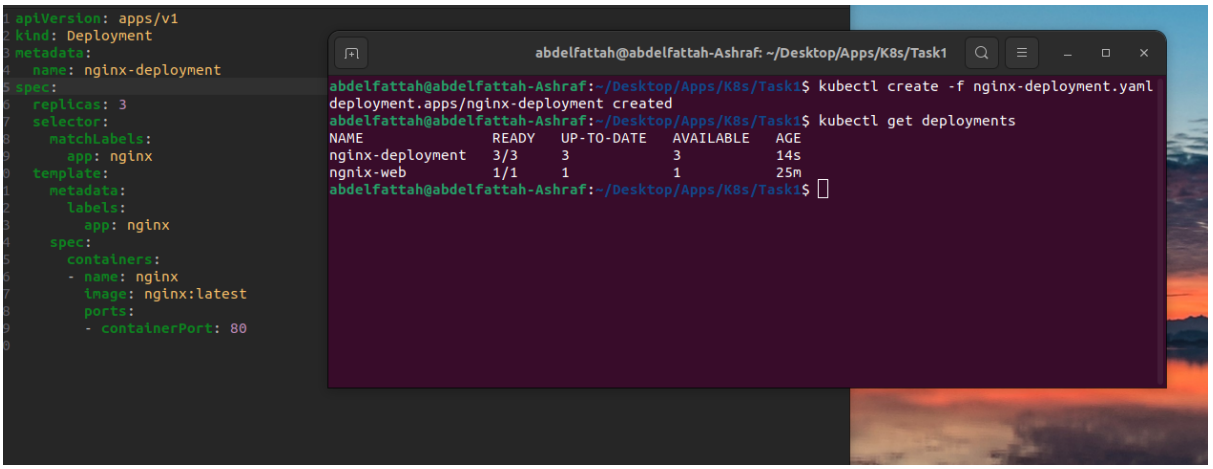
```
abdelfattah@abdelfattah-Ashraf:~$  minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured

abdelfattah@abdelfattah-Ashraf:~$ kubectl version -o yaml
clientVersion:
  buildDate: "2023-05-17T14:20:07Z"
  compiler: gc
  gitCommit: 7f6f68fdabc4df88cfea2dcf9a19b2b830f1e647
  gitTreeState: clean
  gitVersion: v1.27.2
  goVersion: go1.20.4
  major: "1"
  minor: "27"
  platform: linux/amd64
kustomizeVersion: v5.0.1
serverVersion:
  buildDate: "2023-03-15T13:33:12Z"
  compiler: gc
  gitCommit: 9e644106593f3f4aa98f8a84b23db5fa378900bd
  gitTreeState: clean
  gitVersion: v1.26.3
  goVersion: go1.19.7
  major: "1"
  minor: "26"
  platform: linux/amd64
```

```
abdelfattah@abdelfattah-Ashraf:~$ kubectl cluster-info
Kubernetes control plane is running at https://192.168.49.2:8443
CoreDNS is running at https://192.168.49.2:8443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
abdelfattah@abdelfattah-Ashraf:~$ kubectl get nodes
NAME       STATUS   ROLES           AGE   VERSION
minikube   Ready    control-plane   9h    v1.26.3
abdelfattah@abdelfattah-Ashraf:~$
```

2- create nginx deployment with 3 replicas



3- create service to point to this deployment , type cluster IP

```yaml
apiVersion: v1
kind: Service
metadata:
  creationTimestamp: null
  labels:
    app: nginx-service
  name: nginx-service
spec:
  ports:
  - name: 80-80
    port: 80
    protocol: TCP
    targetPort: 80
  selector:
    app: nginx-service
  type: ClusterIP
status:
  loadBalancer: {}
```

```
abdelfattah@abdelfattah-Ashraf: ~/Desktop/Apps/K8...
abdelfattah@abdelfattah-Ashraf:~/Desktop/Apps/K8s/Task1$ kubectl get servic
e
NAME            TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)   AGE
kubernetes      ClusterIP   10.96.0.1        <none>        443/TCP   9h
nginx-service   ClusterIP   10.103.248.175   <none>        80/TCP    52s
abdelfattah@abdelfattah-Ashraf:~/Desktop/Apps/K8s/Task1$
```

4- create debug pod to test the service

```
abdelfattah@abdelfattah-Ashraf:~/Desktop/Apps/K8s/Task1$ kubectl get svc
NAME            TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)   AGE
kubernetes      ClusterIP   10.96.0.1        <none>        443/TCP   9h
nginx-service   ClusterIP   10.103.248.175   <none>        80/TCP    9m6s
abdelfattah@abdelfattah-Ashraf:~/Desktop/Apps/K8s/Task1$ kubectl run -it testpod --image=alpine bin/bash --restart=Never --rm
pod "testpod" deleted
pod default/testpod terminated (ContainerCannotRun)
failed to create shim task: OCI runtime create failed: runc create failed: unable to start container process: exec: "bin/bash": stat bin/bash: no such file o
r directory: unknown
abdelfattah@abdelfattah-Ashraf:~/Desktop/Apps/K8s/Task1$ kubectl run -it testpod --image=alpine bin/ash --restart=Never --rm
If you don't see a command prompt, try pressing enter.
/ #
/ # wget nginx-service:80
Connecting to nginx-service:80 (10.103.248.175:80)
wget: can't connect to remote host (10.103.248.175): Connection refused
/ # wget kubernetes:443
Connecting to kubernetes:443 (10.96.0.1:443)
```