# Document on k8-cluster deployment commands

1. Get into the killercoda site and perform the commands.
2. Start with sudo su command to become as a root use

   Cmd: sudo su

```
controlplane:~$ sudo su
controlplane:~$ ls
filesystem
```

3. Then create a namespace t20-as

   Cmd: kubectl create ns t20-ns (this cmd is used to create the name-space )

```
controlplane:~$ kubectl create ns t20-ns
namespace/t20-ns created
```

4. Then create a directory name it as t20-dep-yaml-files

   Cmd: mkdir t20-dep.yaml (this cmd is used to create a new directory)

```
controlplane:~$ mkdir deployment-yaml-files
controlplane:~$ ls
```

5. After creating the directory go into the directory

   Cmd : cd deployment-yaml-files (this cmd is used to enter into the selected directory)

```
controlplane:~$ cd deployment-yaml-files/
controlplane:~/deployment-yaml-files$ vi t20-dep.yaml
```

6. In that directory create a file name it as t20-dip.yaml

   Cmd: vi t20-dep.yaml(this cmd is used to enter into the file so that we can add the yaml file in it)

```
controlplane:~/deployment-yaml-files$ vi t20-dep.yaml
```

7. Then paste the deployment code in the file

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: ipl-dep
  namespace: t20-ns
spec:
  replicas: 3
  selector:
    matchLabels:
      app: ipl-app
  template:
    metadata:
      labels:
        app: ipl-app
    spec:
      containers:
      - name: ipl-container
        image: nginx:latest
        ports:
        - containerPort: 80
```

8. Now apply the deployment file to run other process

Cmd: kubectl apply -f t20-dep.yaml(this cmd is used to run what all the data presented in the yaml file)

```
controlplane:~/deployment-yaml-files$ vi t20-dep.yaml
controlplane:~/deployment-yaml-files$ kubectl apply -f t20-dep.yaml
deployment.apps/ipl-dep created
```

9. To check all the pods

Cmd: kubectl get all -n t20-ns (this cmd is used to check all the running pods.)

```
No resources found in murali-ns namespace.
controlplane:~/deployment-yaml-files$ kubectl get all -n t20-ns
NAME                           READY   STATUS    RESTARTS   AGE
pod/ipl-dep-6f4df56686-pkrnx   1/1     Running   0          56s
pod/ipl-dep-6f4df56686-ssfbt   1/1     Running   0          56s
pod/ipl-dep-6f4df56686-w997p   1/1     Running   0          56s

NAME                       READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/ipl-dep    3/3     3            3           56s

NAME                                 DESIRED   CURRENT   READY   AGE
replicaset.apps/ipl-dep-6f4df56686   3         3         3       56s
```

10. Now It shows **complete details** about the deployment.

Cmd : kubectl describe deployment ipl-deployment -n t20-ns

```
controlplane:~/deployment-yaml-files$ kubectl describe deployment ipl-deployment -n t20-ns
Error from server (NotFound): deployments.apps "ipl-deployment" not found
controlplane:~/deployment-yaml-files$ kubectl describe deployment ipl-dep -n t20-ns
Name:                   ipl-dep
Namespace:              t20-ns
CreationTimestamp:      Thu, 19 Feb 2026 13:06:14 +0000
Labels:                 <none>
Annotations:            deployment.kubernetes.io/revision: 1
Selector:               app=ipl-app
Replicas:               3 desired | 3 updated | 3 total | 3 available | 0 unavailable
StrategyType:           RollingUpdate
MinReadySeconds:        0
RollingUpdateStrategy:  25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=ipl-app
  Containers:
   ipl-container:
    Image:         nginx:latest
    Port:          80/TCP
    Host Port:     0/TCP
    Environment:   <none>
    Mounts:        <none>
  Volumes:         <none>
  Node-Selectors:  <none>
  Tolerations:     <none>
Conditions:
  Type           Status  Reason
  ----           ------  ------
  Available      True    MinimumReplicasAvailable
  Progressing    True    NewReplicaSetAvailable
OldReplicaSets:  <none>
NewReplicaSet:   ipl-dep-6f4df56686 (3/3 replicas created)
Events:
  Type    Reason             Age    From                   Message
  ----    ------             ----   ----                   -------
  Normal  ScalingReplicaSet  2m45s  deployment-controller  Scaled up replica set ipl-dep-6f4df56686 from 0 to 3
```

11. After completing the cmd now just copy the pod number and check with the command

Cmd: kubectl describe pod ipl-dep-6f4df56686-pkrnx -n t20-ns

```
controlplane:~/deployment-yaml-files$ kubectl describe pod ipl-dep-6f4df56686-pkrnx -n t20-ns
Name:             ipl-dep-6f4df56686-pkrnx
Namespace:        t20-ns
Priority:         0
Service Account:  default
Node:             controlplane/172.30.1.2
Start Time:       Thu, 19 Feb 2026 13:06:14 +0000
Labels:           app=ipl-app
                  pod-template-hash=6f4df56686
Annotations:      cni.projectcalico.org/containerID: 3ca847f28b673ae4ba213998e6e4526f5319d84912cffadf94fe2d43
                  cni.projectcalico.org/podIP: 192.168.0.4/32
                  cni.projectcalico.org/podIPs: 192.168.0.4/32
Status:           Running
IP:               192.168.0.4
IPs:
  IP:           192.168.0.4
Controlled By:  ReplicaSet/ipl-dep-6f4df56686
Containers:
  ipl-container:
    Container ID:   containerd://e057f48724f283adb3a2745352caa67555fb1df9a431ae71708abdfa7910f66c
    Image:          nginx:latest
    Image ID:       docker.io/library/nginx@sha256:341bf0f3ce6c5277d6002cf6e1fb0319fa4252add24ab6a0e262e0056c
    Port:           80/TCP
    Host Port:      0/TCP
    State:          Running
      Started:      Thu, 19 Feb 2026 13:06:22 +0000
    Ready:          True
    Restart Count:  0
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-2v4hf (ro)
Conditions:
  Type                        Status
  PodReadyToStartContainers   True
  Initialized                 True
```

12. It checks and shows the real-time status of the deployment update (whether the rollout is successfully completed or still in progress).

Cmd: kubectl rollout status deployment ipl-dep -n t20-ns

```
controlplane:~/deployment-yaml-files$ kubectl rollout status deployment ipl-dep -n t20-ns
deployment "ipl-dep" successfully rolled out
controlplane:~/deployment-yaml-files$ kubectl get all -n t20-ns
NAME                            READY   STATUS    RESTARTS   AGE
pod/ipl-dep-6f4df56686-pkrnx   1/1     Running   0          7m38s
pod/ipl-dep-6f4df56686-ssfbt   1/1     Running   0          7m38s
pod/ipl-dep-6f4df56686-w997p   1/1     Running   0          7m38s

NAME                       READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/ipl-dep   3/3     3            3           7m38s

NAME                                  DESIRED   CURRENT   READY   AGE
replicaset.apps/ipl-dep-6f4df56686   3         3         3       7m38s
controlplane:~/deployment-yaml-files$ cat t20-dep.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: ipl-dep
  namespace: t20-ns
spec:
  replicas: 3
  selector:
    matchLabels:
      app: ipl-app
  template:
    metadata:
      labels:
        app: ipl-app
    spec:
      containers:
      - name: ipl-container
        image: nginx:latest
        ports:
        - containerPort: 80
```

13. It rolls back the deployment to the previous working version if the current update has issues.

Cmd : kubectl rollout undo deployment ipl-dep -n <namespace>