

1. Install a multi-node kind cluster on your local machine using the below details:

Cluster Name :- cka-cluster2

Nodes:- 1 Control plane and 3 worker nodes

Kubernetes Version :- 1.30

```
! cluster-config.yaml U X
Day-6 > ! cluster-config.yaml > [ ] nodes > { } 3
1 kind: Cluster
2 apiVersion: kind.x-k8s.io/v1alpha4
3 name: cka-cluster2
4 nodes:
5   - role: control-plane
6   - role: worker
7   - role: worker
8   - role: worker
~ |
```

```
Bhakti@LAPTOP-DNC3NQI0 MINGW64 /d/Github/CKA2024 (main)
$ kind create cluster --name cka-cluster2 --config D:/Github/CKA2024/Day-6/cluster-config.yaml
Creating cluster "cka-cluster2" ...
  • Ensuring node image (kindest/node:v1.30.0) 🗄 ...
  ✓ Ensuring node image (kindest/node:v1.30.0) 🗄
  • Preparing nodes 📦 📦 📦 ...
  ✓ Preparing nodes 📦 📦 📦
  • Writing configuration 📜 ...
  ✓ Writing configuration 📜
  ✓ Ensuring node image (kindest/node:v1.30.0) 🗄
  • Preparing nodes 📦 📦 📦 ...
  ✓ Preparing nodes 📦 📦 📦
  • Writing configuration 📜 ...
  ✓ Writing configuration 📜
  • Starting control-plane 🚦 ...
  ✓ Starting control-plane 🚦
  • Installing CNI 🛠 ...
  ✓ Installing CNI 🛠
  • Installing StorageClass 🗄 ...
  ✓ Installing StorageClass 🗄
  ✓ Ensuring node image (kindest/node:v1.30.0) 🗄
  • Preparing nodes 📦 📦 📦 ...
  ✓ Preparing nodes 📦 📦 📦
  • Writing configuration 📜 ...
  ✓ Writing configuration 📜
  • Starting control-plane 🚦 ...
  ✓ Writing configuration 📜
  • Starting control-plane 🚦 ...
  ✓ Starting control-plane 🚦
  • Installing CNI 🛠 ...
  ✓ Installing CNI 🛠
  • Installing StorageClass 🗄 ...
  ✓ Installing StorageClass 🗄
  • Joining worker nodes 🚶 ...
  ✓ Joining worker nodes 🚶
Set kubectl context to "kind-cka-cluster2"
You can now use your cluster with:
```

```
kubectl cluster-info --context kind-cka-cluster2
```

Have a question, bug, or feature request? Let us know! <https://kind.sigs.k8s.io/#community> 😊

2. Set the current context to the newly created cluster.

```
Bhakti@LAPTOP-DNC3NQI0 MINGW64 /d/Github/CKA2024 (main)
● $ kubectl cluster-info --context kind-cka-cluster2
Kubernetes control plane is running at https://127.0.0.1:61265
CoreDNS is running at https://127.0.0.1:61265/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
```

3. Run commands such as `kubectl get nodes` and ensure you have all the nodes ready.

```
Bhakti@LAPTOP-DNC3NQI0 MINGW64 /d/Github/CKA2024 (main)
● $ kubectl get nodes
```

NAME	STATUS	ROLES	AGE	VERSION
cka-cluster2-control-plane	Ready	control-plane	5m22s	v1.30.0
cka-cluster2-worker	Ready	<none>	4m57s	v1.30.0
cka-cluster2-worker2	Ready	<none>	4m57s	v1.30.0
cka-cluster2-worker3	Ready	<none>	4m55s	v1.30.0

These nodes are nothing but containers as KIND means Kubernetes IN Docker, which creates containers and uses them as nodes.

4. Run `docker ps` command to verify that all nodes are running as containers.

```
Bhakti@LAPTOP-DNC3NQI0 MINGW64 /d/Github/CKA2024 (main)
● $ docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
497242d4a780	kindest/node:v1.30.0	"/usr/local/bin/entr..."	7 minutes ago	Up 7 minutes		cka-cluster2-worker2
4e97646dc96b	kindest/node:v1.30.0	"/usr/local/bin/entr..."	7 minutes ago	Up 7 minutes		cka-cluster2-worker
3dfb3b89ac3a	kindest/node:v1.30.0	"/usr/local/bin/entr..."	7 minutes ago	Up 7 minutes	127.0.0.1:61265->6443/tcp	cka-cluster2-control-plane
345ae782e9ef	kindest/node:v1.30.0	"/usr/local/bin/entr..."	7 minutes ago	Up 7 minutes		cka-cluster2-worker3
c8dbfc3b8f26	kindest/node:v1.30.0	"/usr/local/bin/entr..."	12 days ago	Up 2 days	127.0.0.1:56038->6443/tcp	cka-cluster1-control-plane