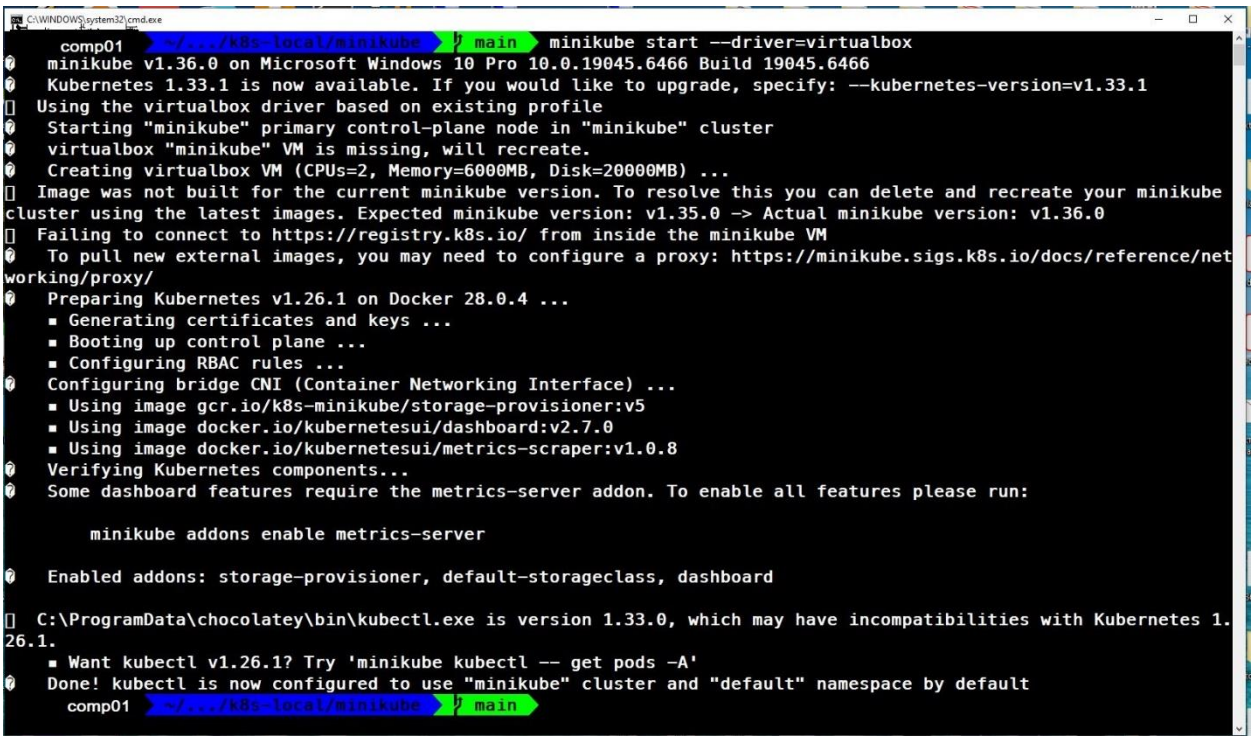
Start minikube on Windows – Virtual Box / HyperV environment

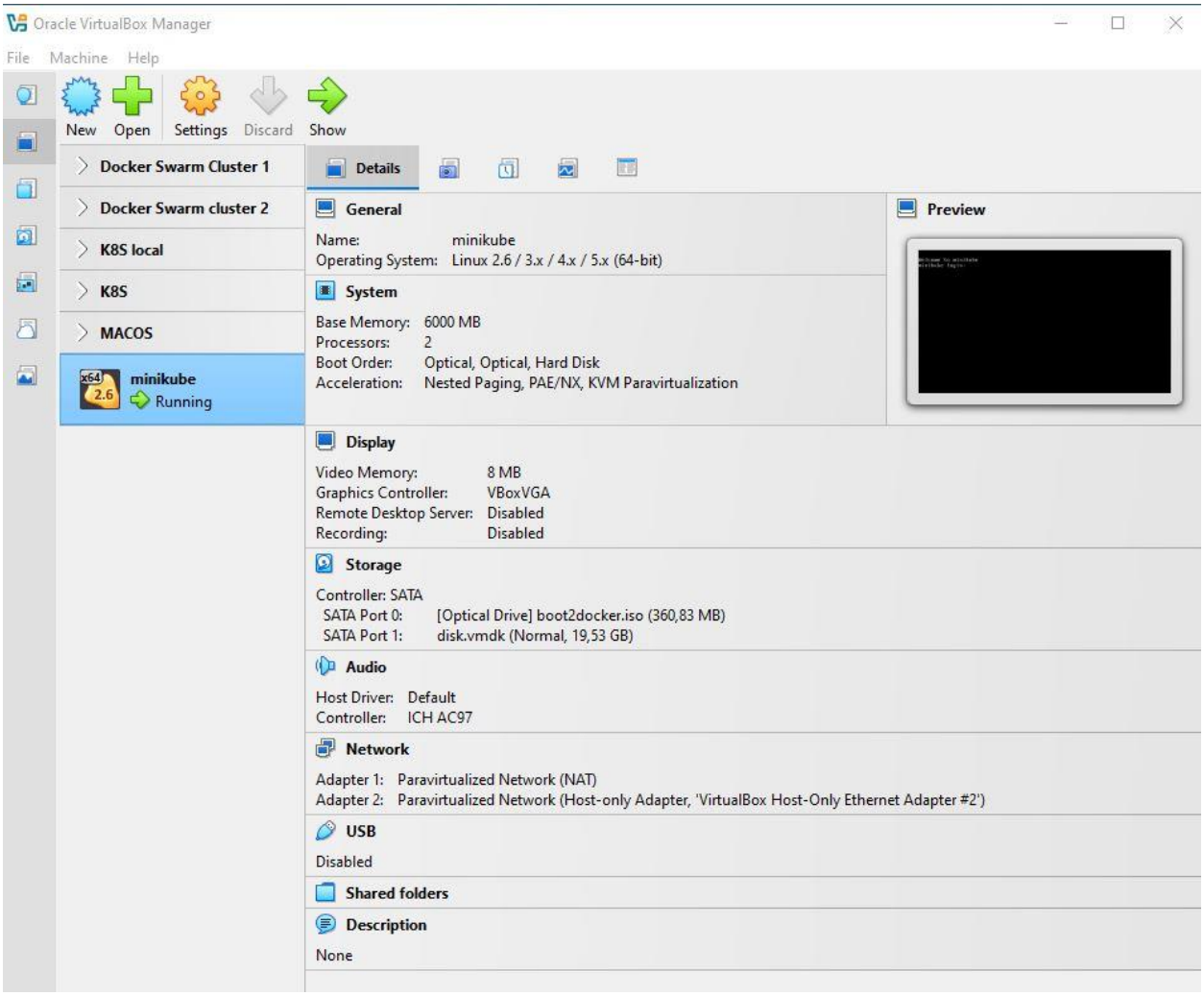
Run as administrator cmd.exe or PowerShell.

Virtual Box environment

To start single node minikube cluster, inside opened terminal run:

minikube start --driver=virtualbox





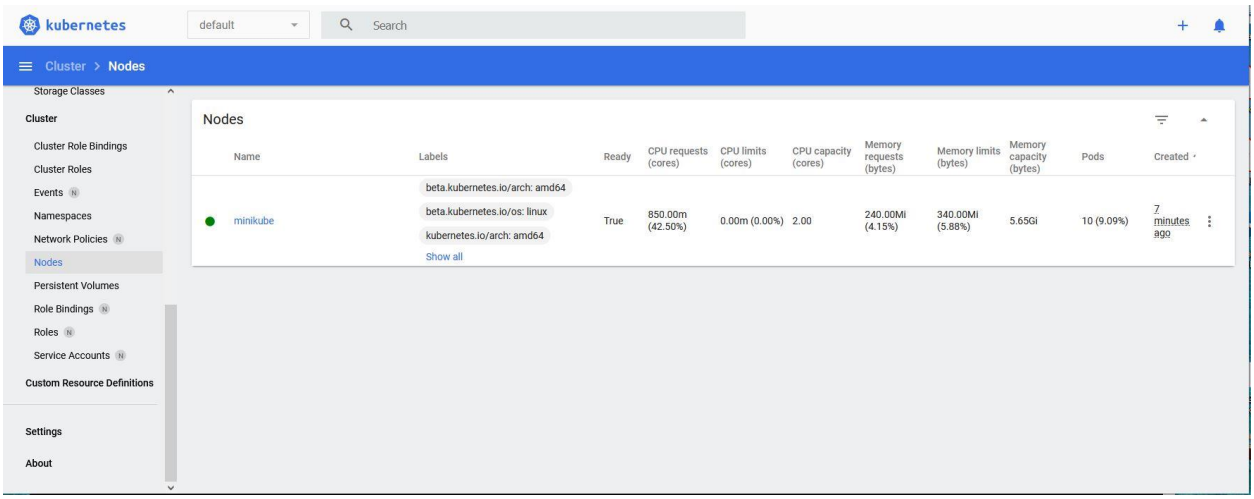
To test if minikube working fine:

kubectl get namespace
or
kubectl get ns

```
comp01 ~/.../k8s-local/minikube main kubectl get namespace
NAME                   STATUS   AGE
default                Active   2m7s
kube-node-lease        Active   2m16s
kube-public            Active   2m16s
kube-system            Active   2m16s
kubernetes-dashboard   Active   2m
```

To access Kubernetes console, inside terminal type:
minikube dashboard

this will open inside you default browser Kubernetes default console:



Create a test app (Deployment)

Let's deploy the classic **nginx** container.
Open second termoinal and type:

kubectl create deployment test-nginx --image=nginx

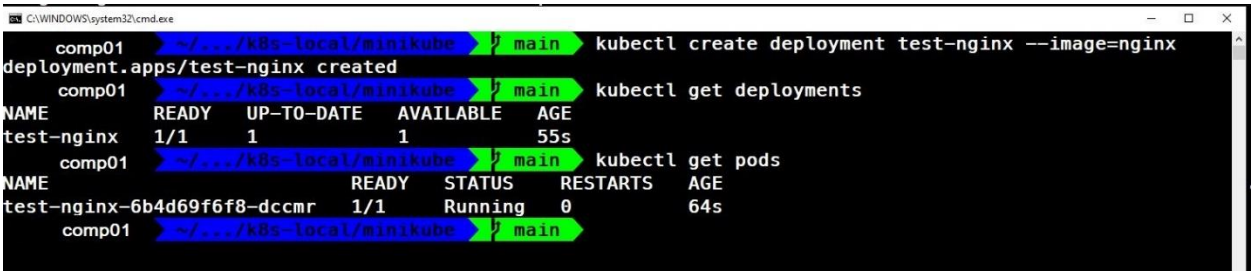
What this does:

- Creates a **Deployment**
- With **1 Pod**
- Using the official nginx image

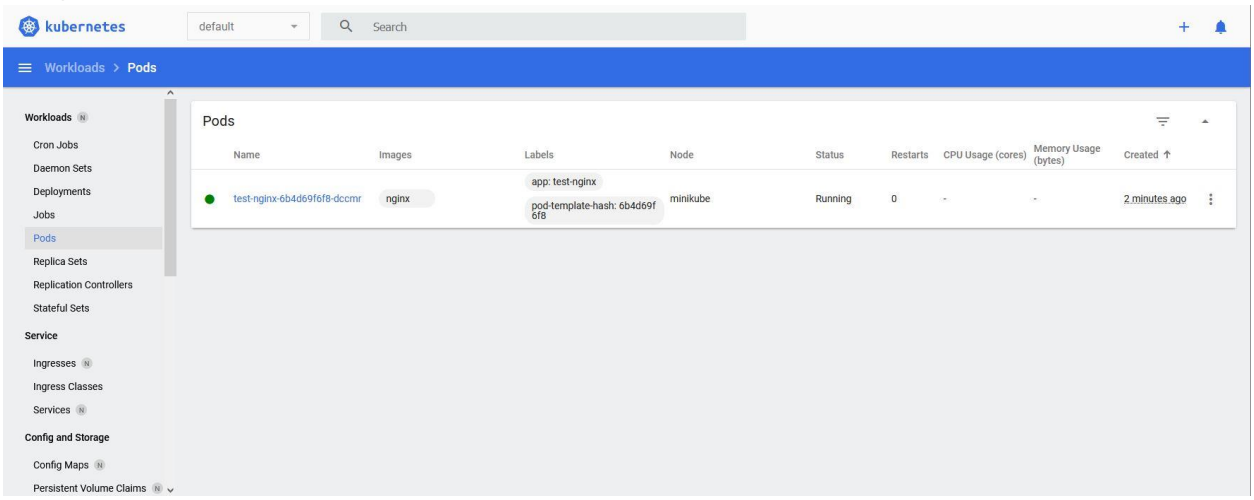
Verify from terminal:

kubectl get deployments
kubectl get pods

You should see a Pod in Running state.



Verify from kubernetese Web UI



---

2 Expose the app (Service)

Expose it inside the cluster:

*kubectl expose deployment test-nginx *
*--type=ClusterIP *
--port=80
kubectl expose deployment test-nginx --type=NodePort --port=80
service/test-nginx exposed
kubectl get svc

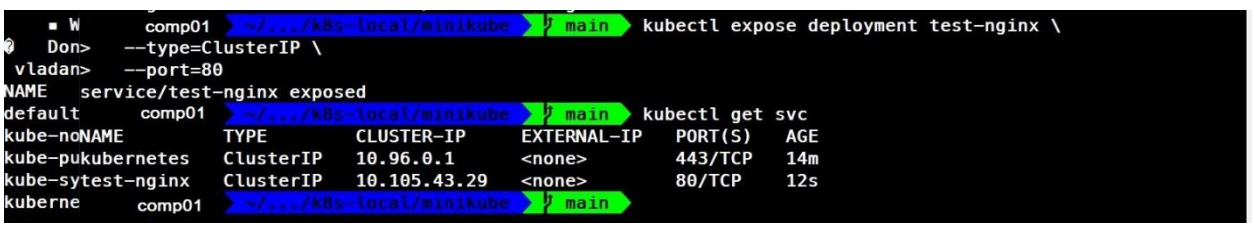
NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
kubernetes ClusterIP 10.96.0.1 <none> 443/TCP 29m
test-nginx NodePort 10.100.219.92 <none> 80:31249/TCP 11s
minikube ip
192.168.59.106

Verify:

kubectl get svc

At this point:

- The app is reachable **inside the cluster**
- Not yet accessible from your browser



2 Use the NodePort

From your previous kubectl get svc output:

test-nginx NodePort 10.100.219.92 <none> 80:31249/TCP

- NodePort = 31249
- ClusterIP = internal only (ignore)

So the URL to open in your browser is:

http://<minikube-ip>:<node-port>

With the example above:

http://192.168.59.106:31249

Open that in your browser — you should see **the nginx default page**.

---

4 Scale the app (still imperative)

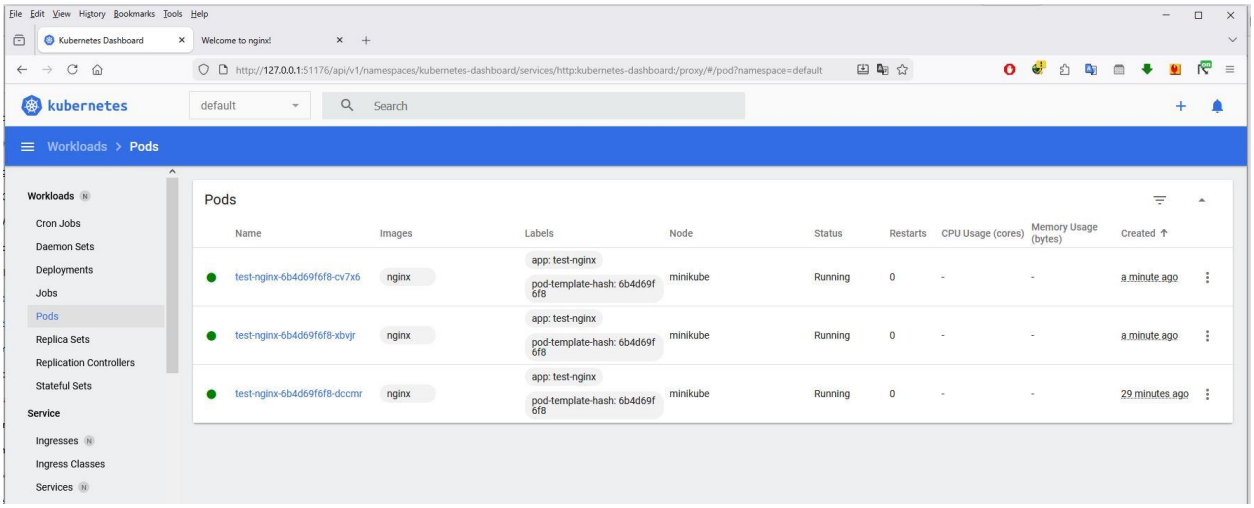
kubectl scale deployment test-nginx --replicas=3
deployment.apps/test-nginx scaled

Verify:

kubectl get pods

NAME READY STATUS RESTARTS AGE
test-nginx-6b4d69f6f8-cv7x6 1/1 Running 0 30s
test-nginx-6b4d69f6f8-dccmr 1/1 Running 0 29m
test-nginx-6b4d69f6f8-xbvjr 1/1 Running 0 30s

You'll now see **3 Pods**.



---

5 Clean up (important habit)

kubectl delete deployment test-nginx
kubectl delete svc test-nginx

🧠 Mental model recap

Imperative commands:

- Are **great for learning and testing**
- Are **not reproducible**
- Should not be used for production

Think of them as:

"Typing directly into the cluster"

📌 One-command quick smoke test

If you want the shortest possible test:

*kubectl create deployment hello --image=nginx && *
*kubectl expose deployment hello --type=NodePort --port=80 && *
minikube service hello

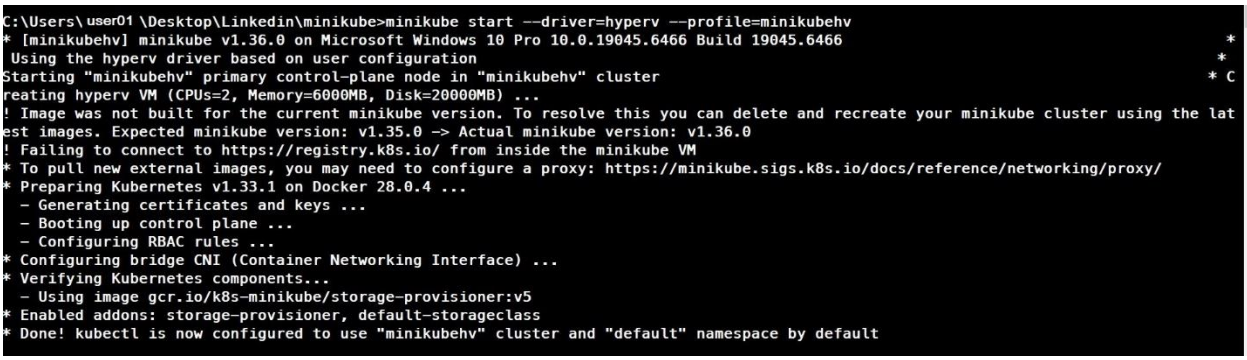
To stop minikube:
minikube stop

* Stopping node "minikube" ...

* 1 node stopped.

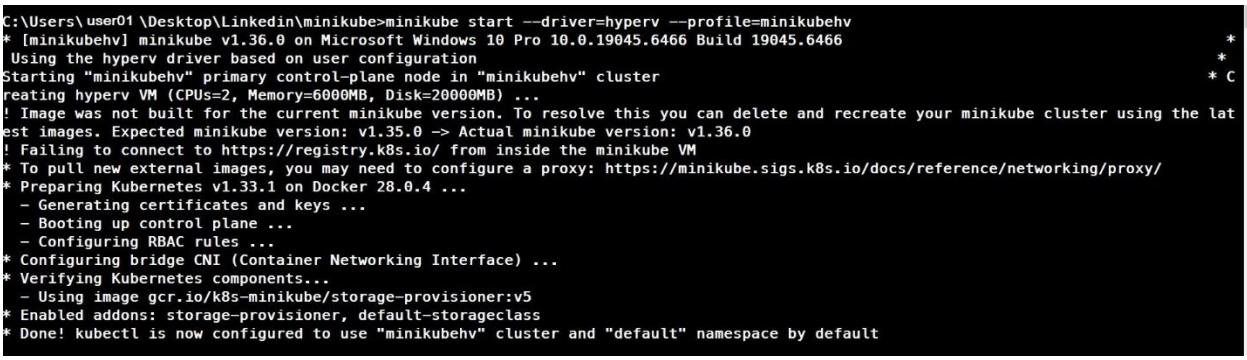
To delete minikube:
minikube delete

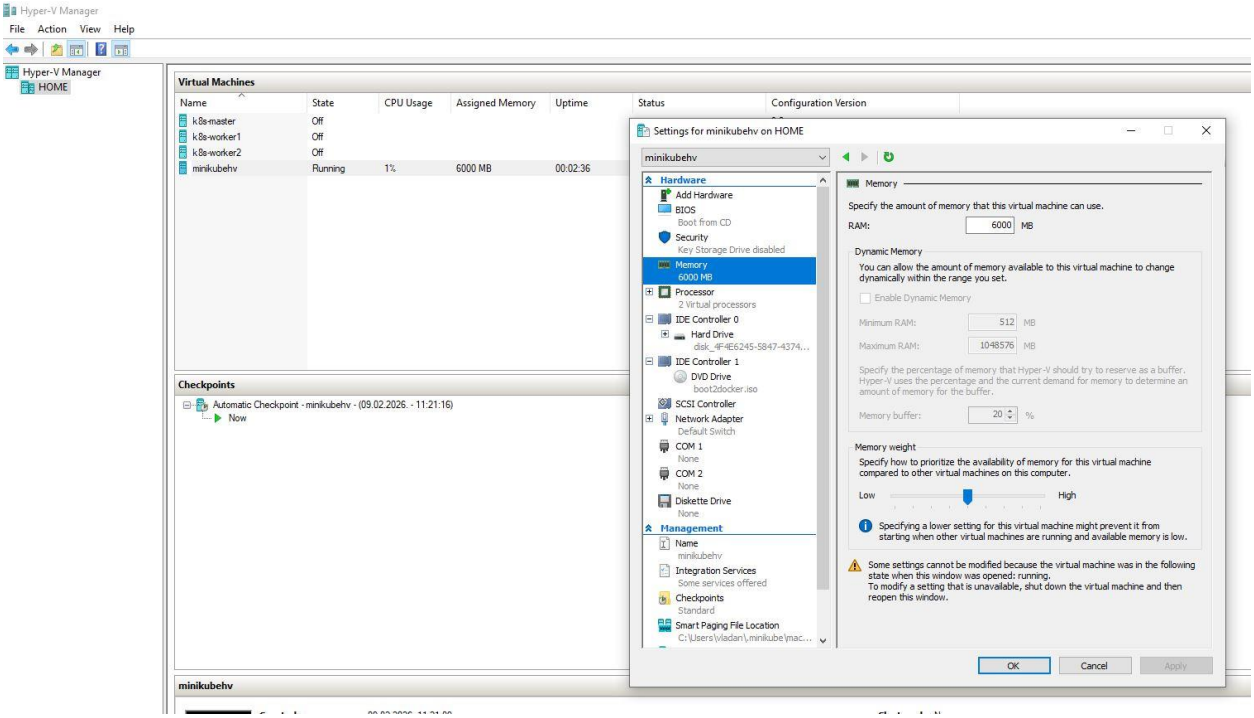
To start minikube 3 node cluster:
minikube start --nodes 3 -p multimode -–driver=virtualbox --profile=mk-vb-3nodes



Hyper-V environment

Run cmd.exe or PowerShell as Administrator:
minikube start --driver=hyperv --profile=minikubehv



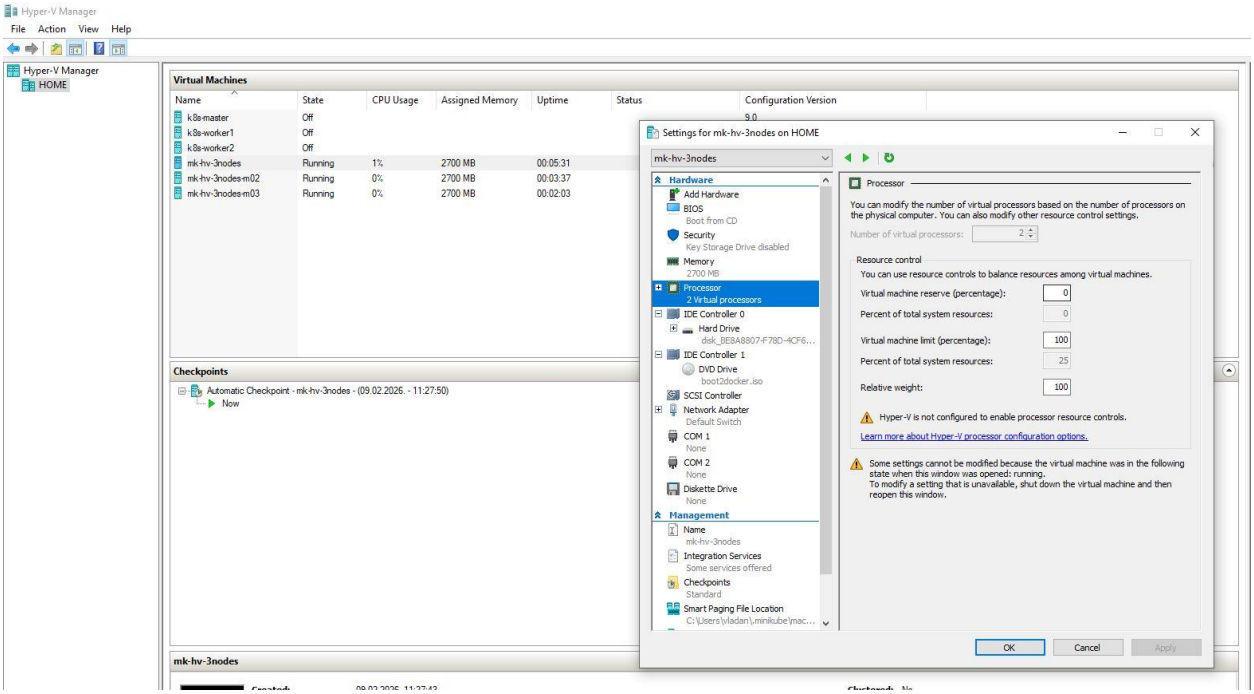


To delete:

```
C:\Users\ user01 \Desktop\Linkedin\minikube>minikube delete --profile=minikubehv
* Stopping node "minikubehv"  ...
* Powering off "minikubehv" via SSH ...
* Deleting "minikubehv" in hyperv ...
* Removed all traces of the "minikubehv" cluster.
```

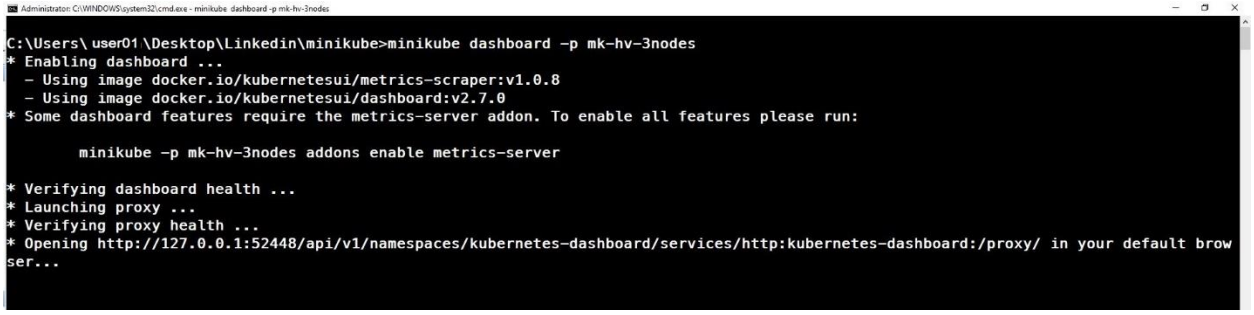
To start minikube 3 node cluster:

minikube start --nodes 3 -p multimode -–driver=hyperv --profile=mk-hv-3nodes



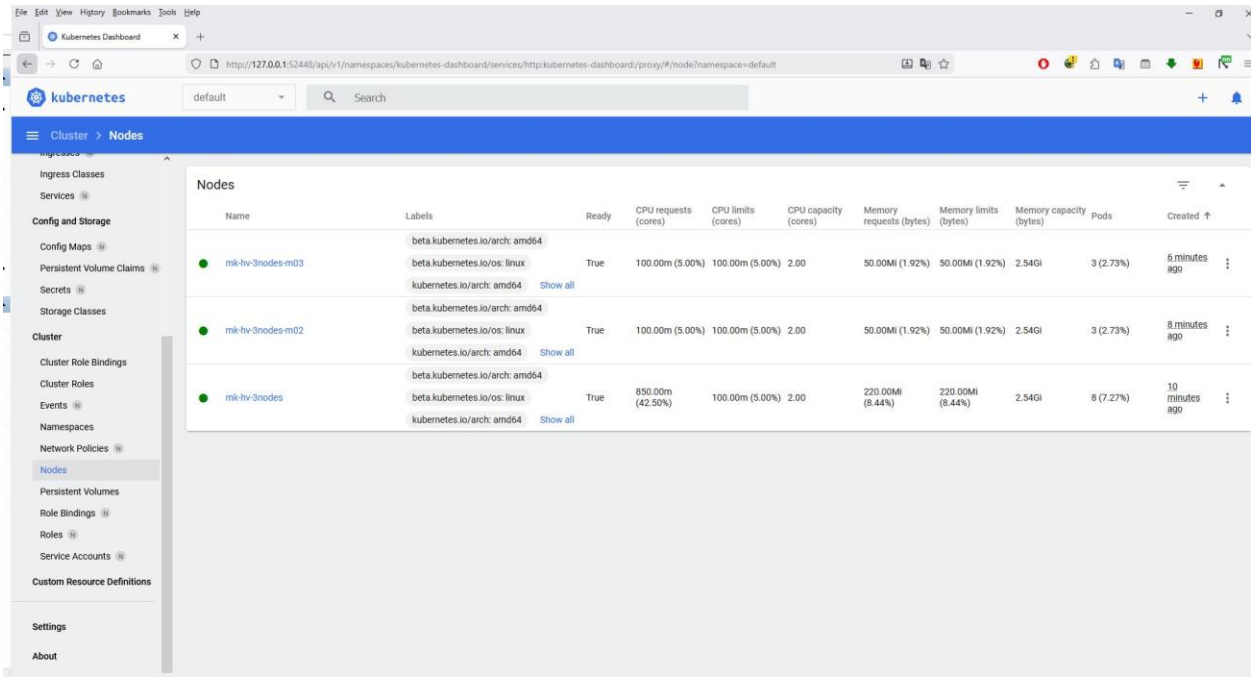
To open cubernetes console:

minikube dashboard -p mk-hv-3nodes

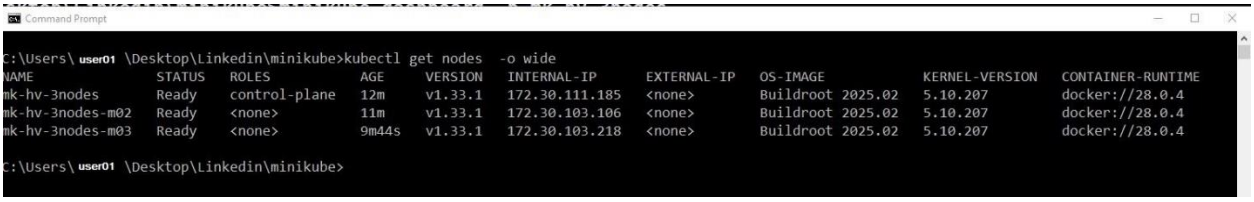


Kubernetes Web UI

http://127.0.0.1:52448/api/v1/namespaces/kubernetes-dashboard/services/http:kubernetes-dashboard:/proxy/#/node?namespace=default



kubectl get nodes -o wide



To delete:
minikube delete --profile=mk-hv-3nodes

1 Stop the cluster (if running)

minikube stop -p mk-hv-3nodes

- Stops all nodes in the cluster
- Safe first step before deletion

---

2 Delete the Minikube cluster

minikube delete -p mk-hv-3nodes

- Deletes **all VMs** created for that profile:
    - mk-hv-3nodes (control-plane)

- mk-hv-3nodes-m02 (worker)
- mk-hv-3nodes-m03 (worker)

- Removes **Minikube state** (cluster metadata, configs) for that profile

---

3 Verify deletion

minikube profile list

- The mk-hv-3nodes profile should no longer exist
- If there's still a profile listed, you can remove it manually:

minikube delete -p <profile-name>

---

4 Remove leftover Hyper-V VMs manually (optional)

Sometimes minikube delete **doesn't remove VMs** completely due to Hyper-V quirks:

1. Open **Hyper-V Manager**
2. Look for VMs named:
   - mk-hv-3nodes
   - mk-hv-3nodes-m02
   - mk-hv-3nodes-m03
3. Right-click → **Delete**

---

5 Remove leftover Minikube config/data (optional)

Minikube keeps data in your **user directory**:

C:\Users\<YourUser>\.minikube

You can delete this folder entirely to **reset Minikube state**:

rmdir /s /q C:\Users\<YourUser>\.minikube

- Deletes all cached ISO images, configs, logs
- Will require re-download when starting a new cluster