

## ▮ EKS Cluster Autoscaling Test + Cleanup Cheat Sheet (MT Project)

---

### ▮ Step 1: Deploy CPU Stress Test

Triggers autoscaler by creating many pods needing CPU.

```
kubectl apply -f
https://raw.githubusercontent.com/kubernetes/autoscaler/master/cluster-
autoscaler/cloudprovider/aws/examples/cluster-autoscaler-test.yaml
```

Check pods:

```
kubectl get pods -w
```

Look for many pods in `Pending` ▮ new nodes will be added.

Check nodes:

```
kubectl get nodes
```

---

### ▮ Step 2: Cleanup (Stop Stress Test & Scale Down)

#### ▮ 2.1 Delete CPU Stress Pods

```
kubectl delete deployment cpu-stress
```

Confirm pods are gone:

```
kubectl get pods
```

#### ▮ 2.2 Watch Nodes Auto-Scale Down (10-15 mins)

```
kubectl get nodes -w
```

Nodes will reduce automatically when they're empty.

#### ▮ 2.3 (Optional) Force Immediate Scale Down to 2 Nodes

Run this ready command for **MT Project**:

```
aws eks update-nodegroup-config --cluster-name mt-cluster --nodegroup-name mt-node-
group --scaling-config minSize=2,maxSize=4,desiredSize=2
```

Wait 2-5 mins and check:

```
kubectl get nodes
```

▮ You should now see only **2 nodes** (clean state).

---

### ▮ Notes

- Cluster Autoscaler auto-adds and removes nodes based on pod demand.

- Stress test is safe: no real workload, just forces scaling.
- Always clean up (Step 2) to avoid AWS charges!

---

▯ **Use this anytime to test scaling! (MT Project safe steps)**