**POD AUTOSCALING**

**VERTICAL POD AUTOSCALING**

* **We increase load of existing pod.**
* **But we cant increase memory beyond a limit,so not used much.**

**HORIZONTAL POD AUTOSCALER**

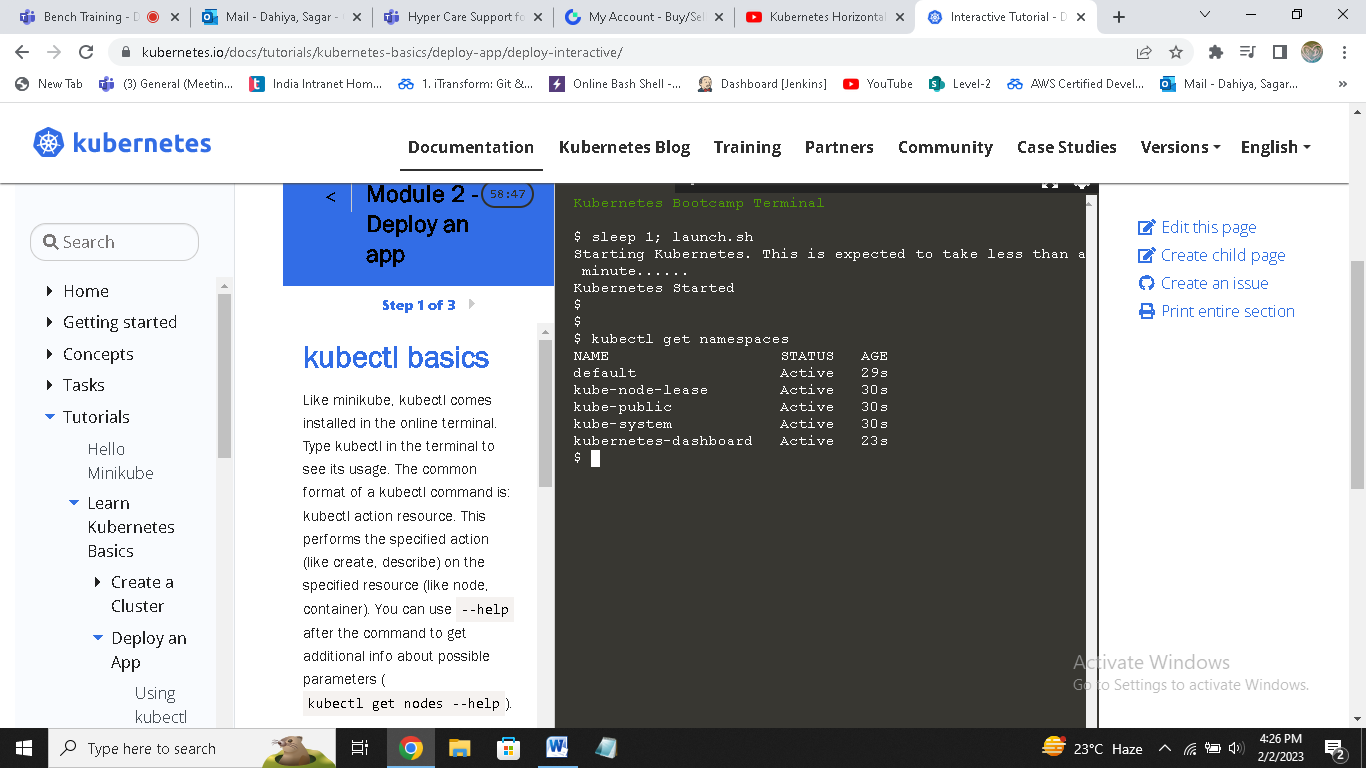
**• It will autoscale automatically acc to loads.**

**• It work on metricserver.**

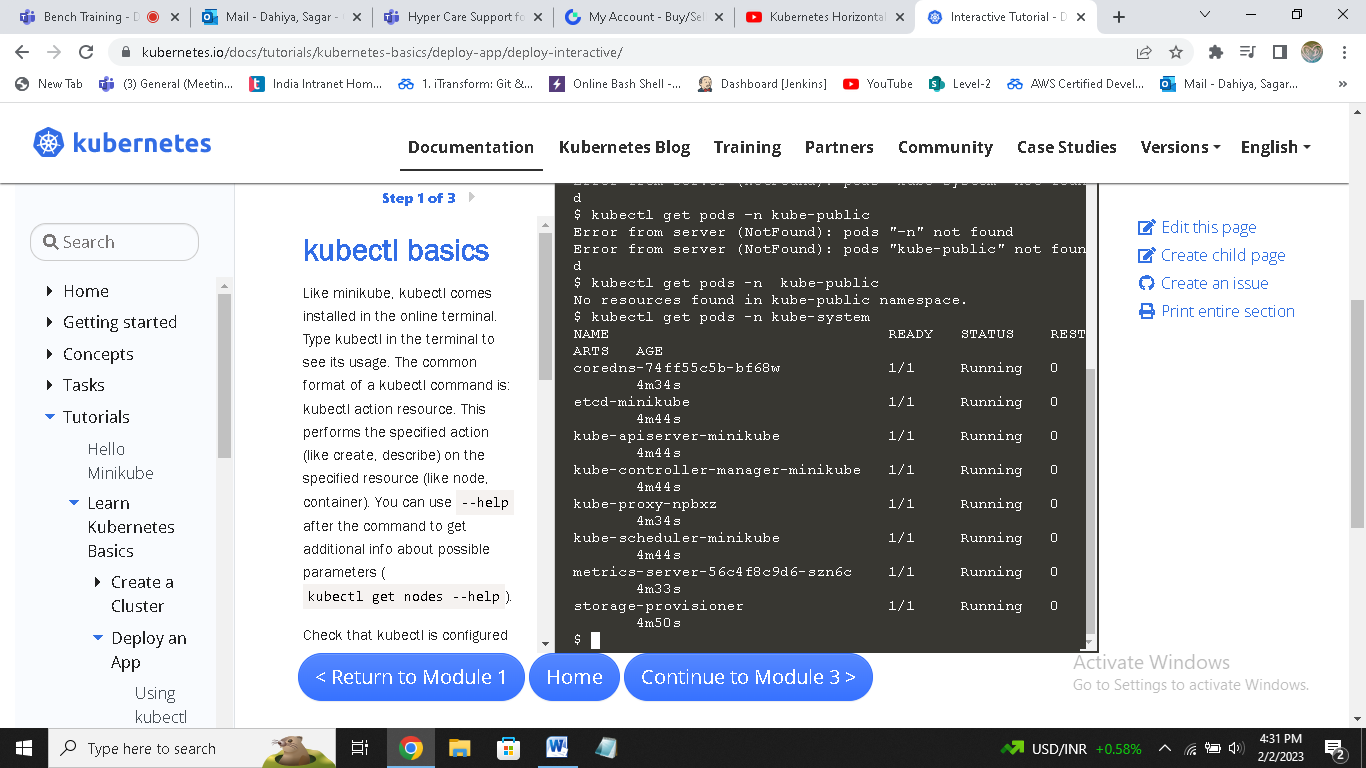
**• Its default time to check load is 30 sec.**

* **It will create new pod as soon as load increase beyond limit.**

**Kubectl get namespaces**



**Kubectl get pods –n kube-system**



**It will show metric-server is running or not.**

**Now run a deployment with below code :**

**kind: Deployment**

**apiVersion: apps/v1**

**metadata:**

**name: mydeploy**

**spec:**

**replicas: 1**

**selector:**

**matchLabels:**

**name: deployment**

**template:**

**metadata:**

**name: testpod8**

**labels:**

**name: deployment**

**spec:**

**containers:**

**- name: c00**

**image: httpd**

**ports:**

**- containerPort: 80**

**resources:**

**limits:**

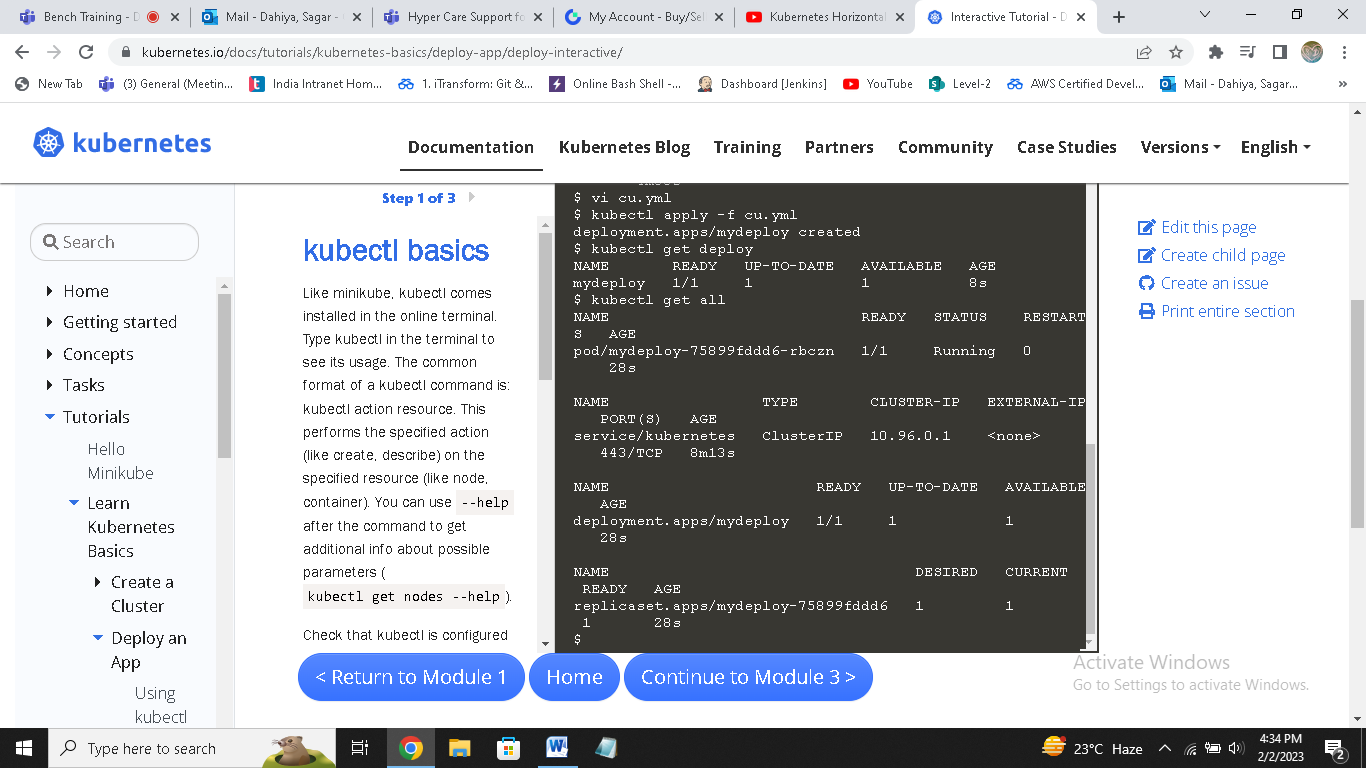
**cpu: 500m**

**requests:**

**cpu: 200m**

**now create deployment**

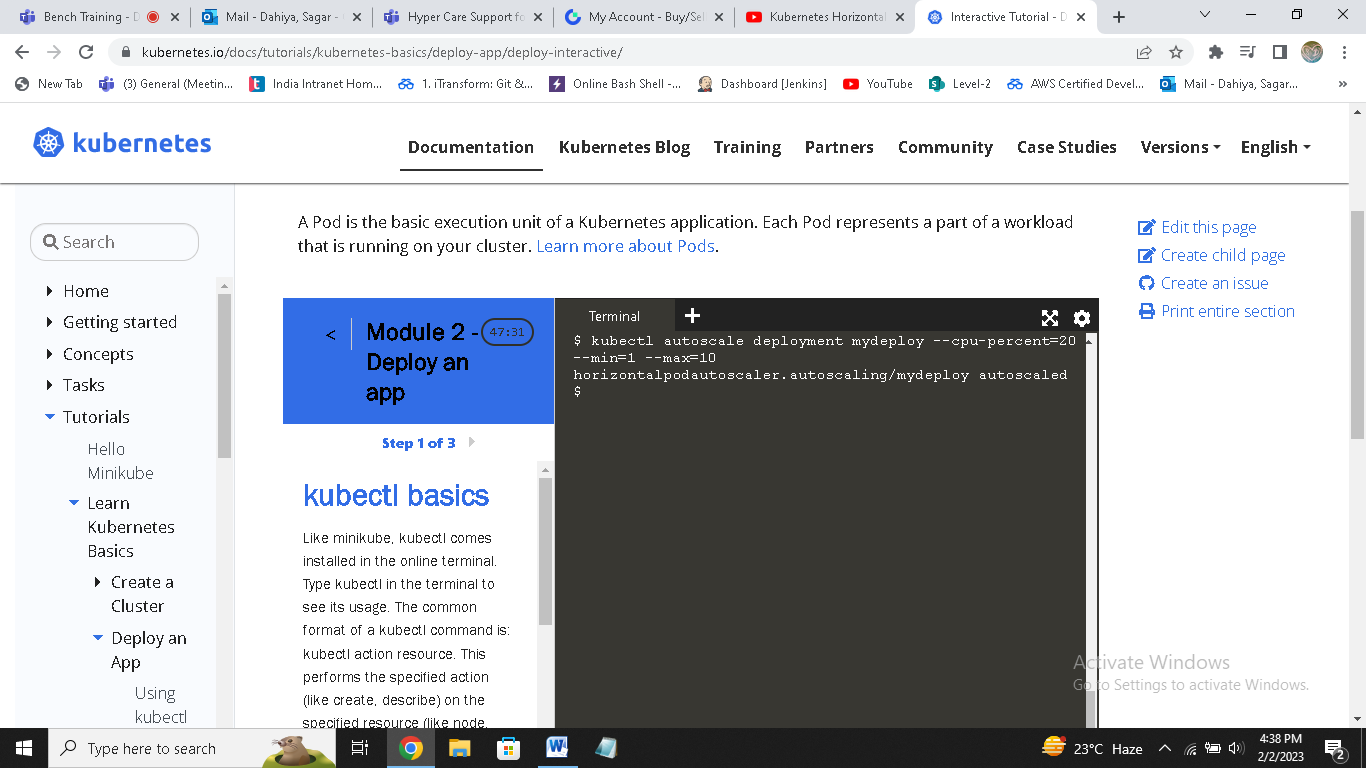
**kubectl get all**



**It will show deployment and inside it,1 container.**

**Now to create horizontal autoscaling,run below command**

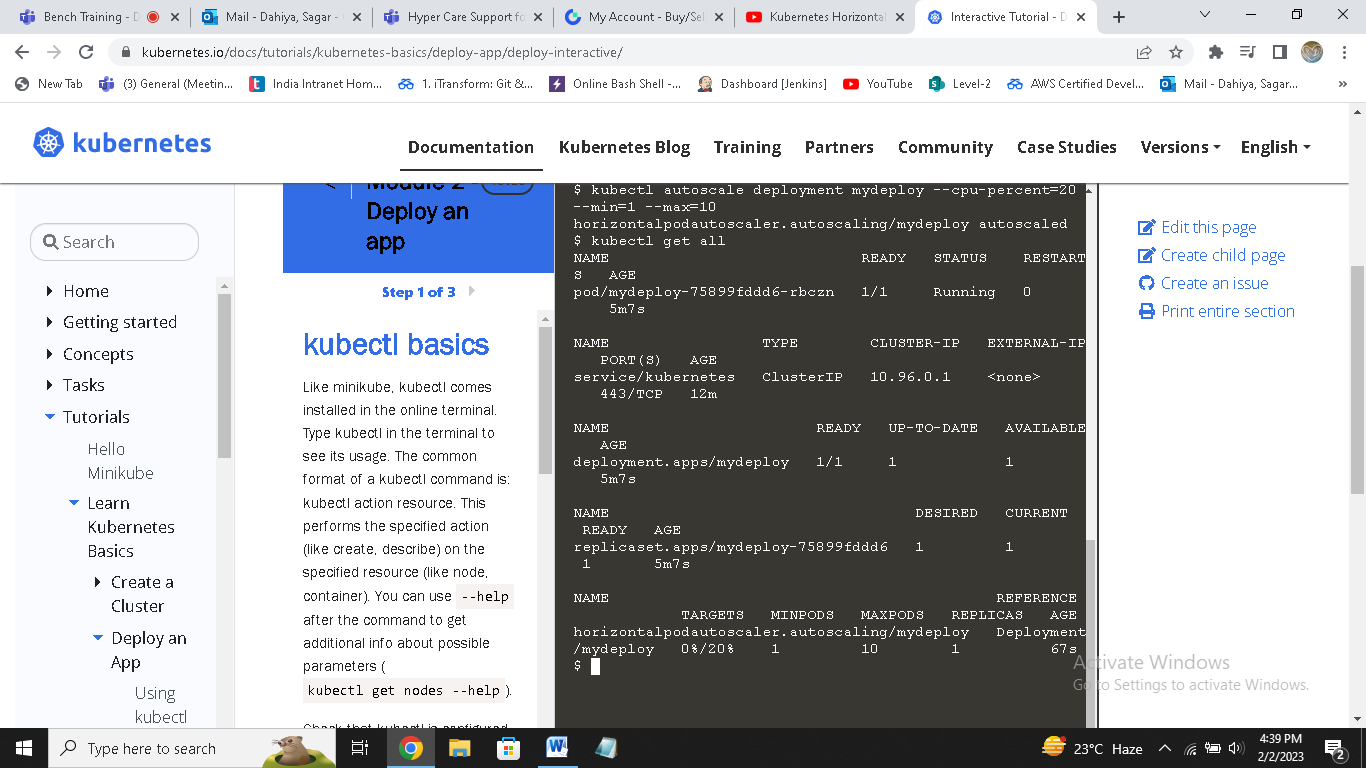
**kubectl autoscale deployment mydeploy --cpu-percent=20 --min=1 --max=10**



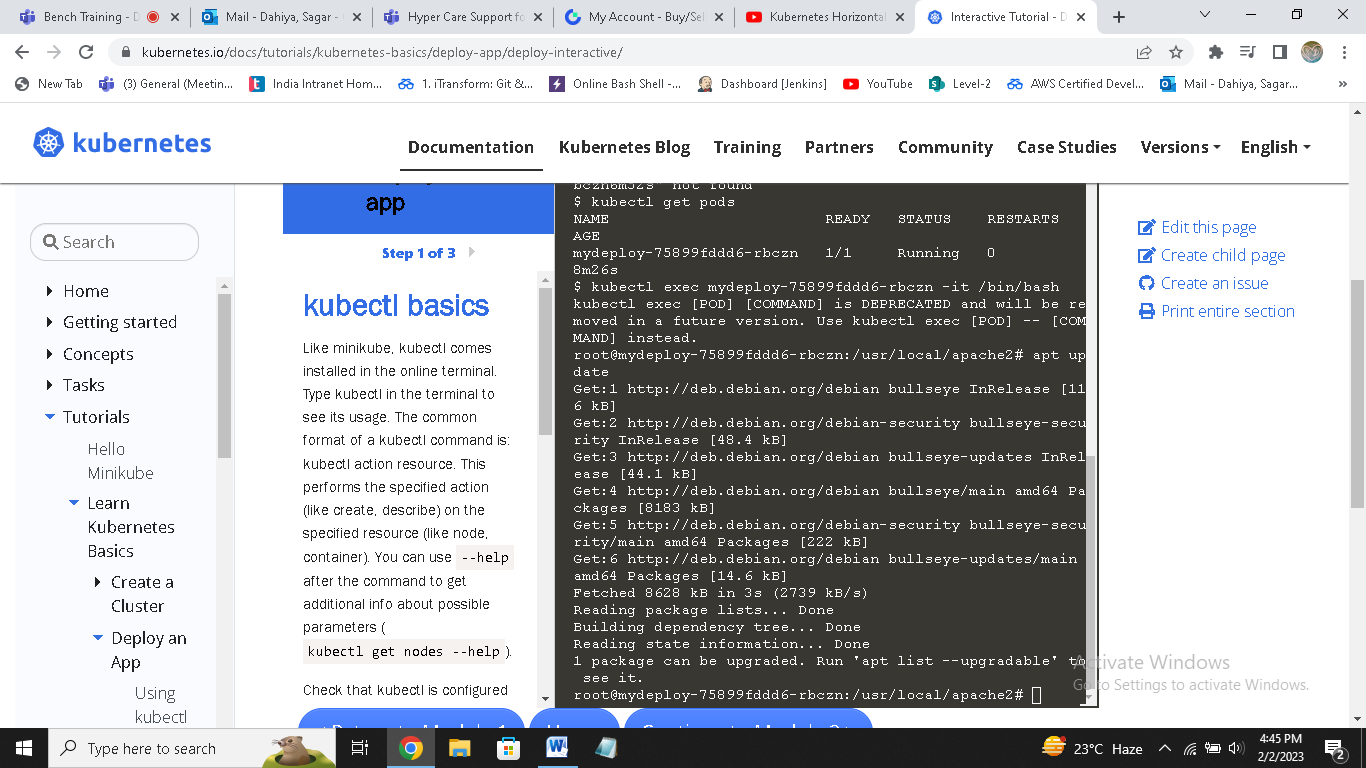
**You can see horizontal auto scaler created.**

**Now run**

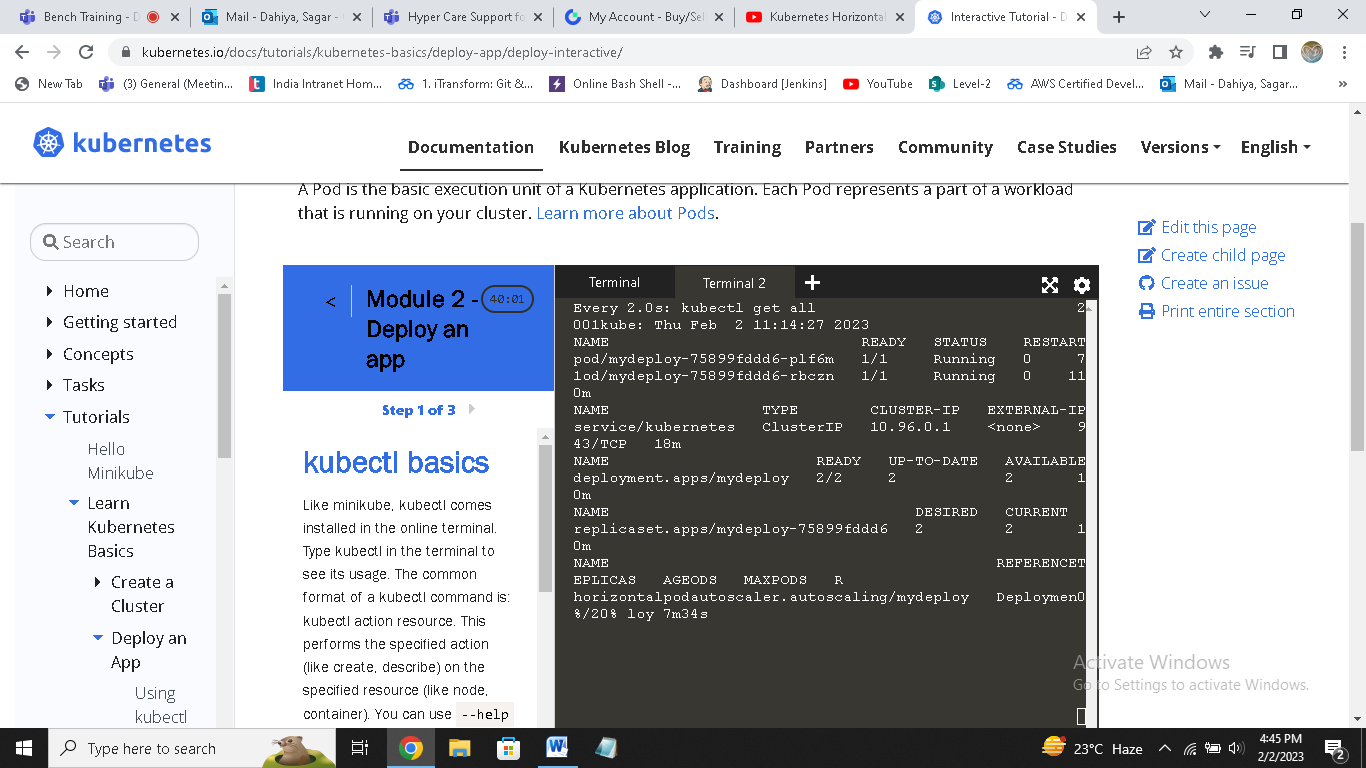
**Kubectl get all**



**Now go into pod and increase load**



**Now check again total pods**



**You can see 2 pods are running now,so it will scale automatically.**