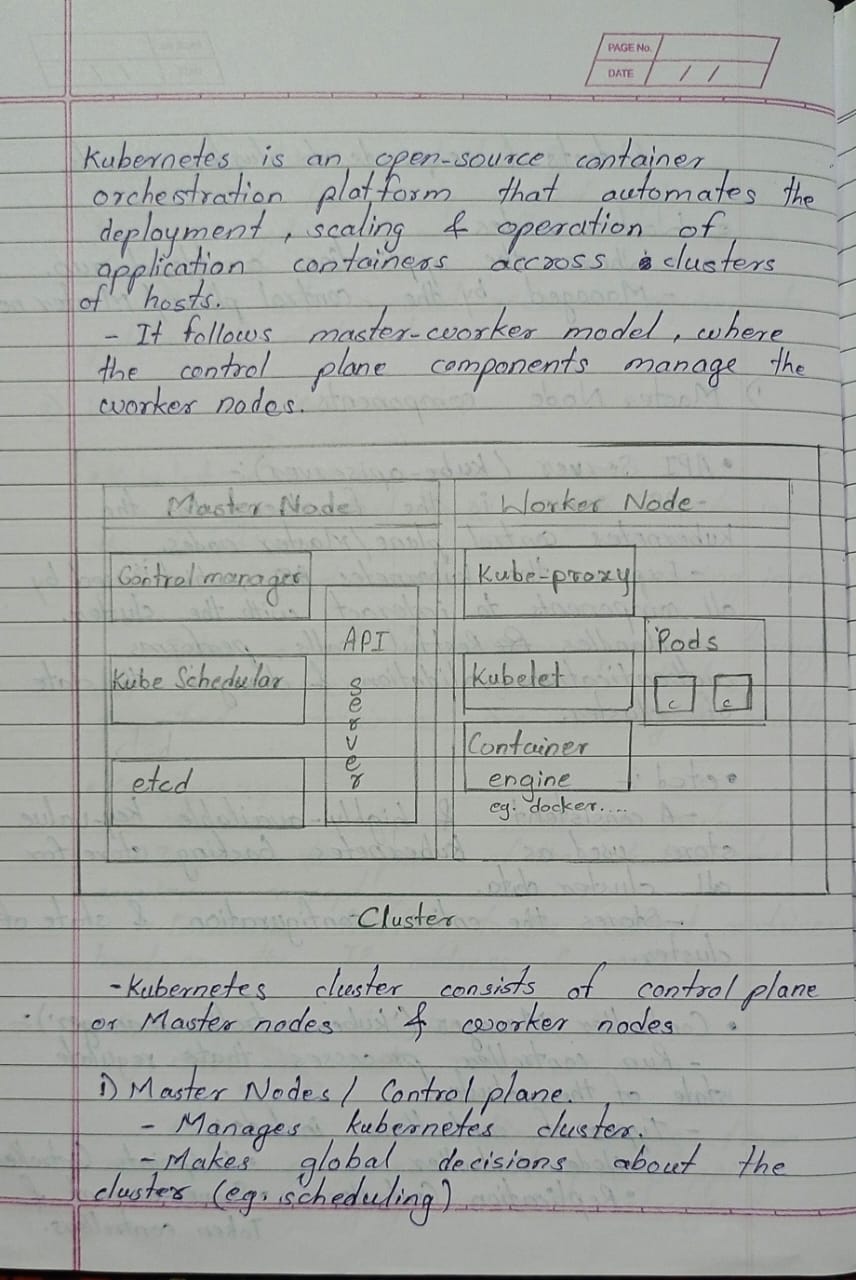
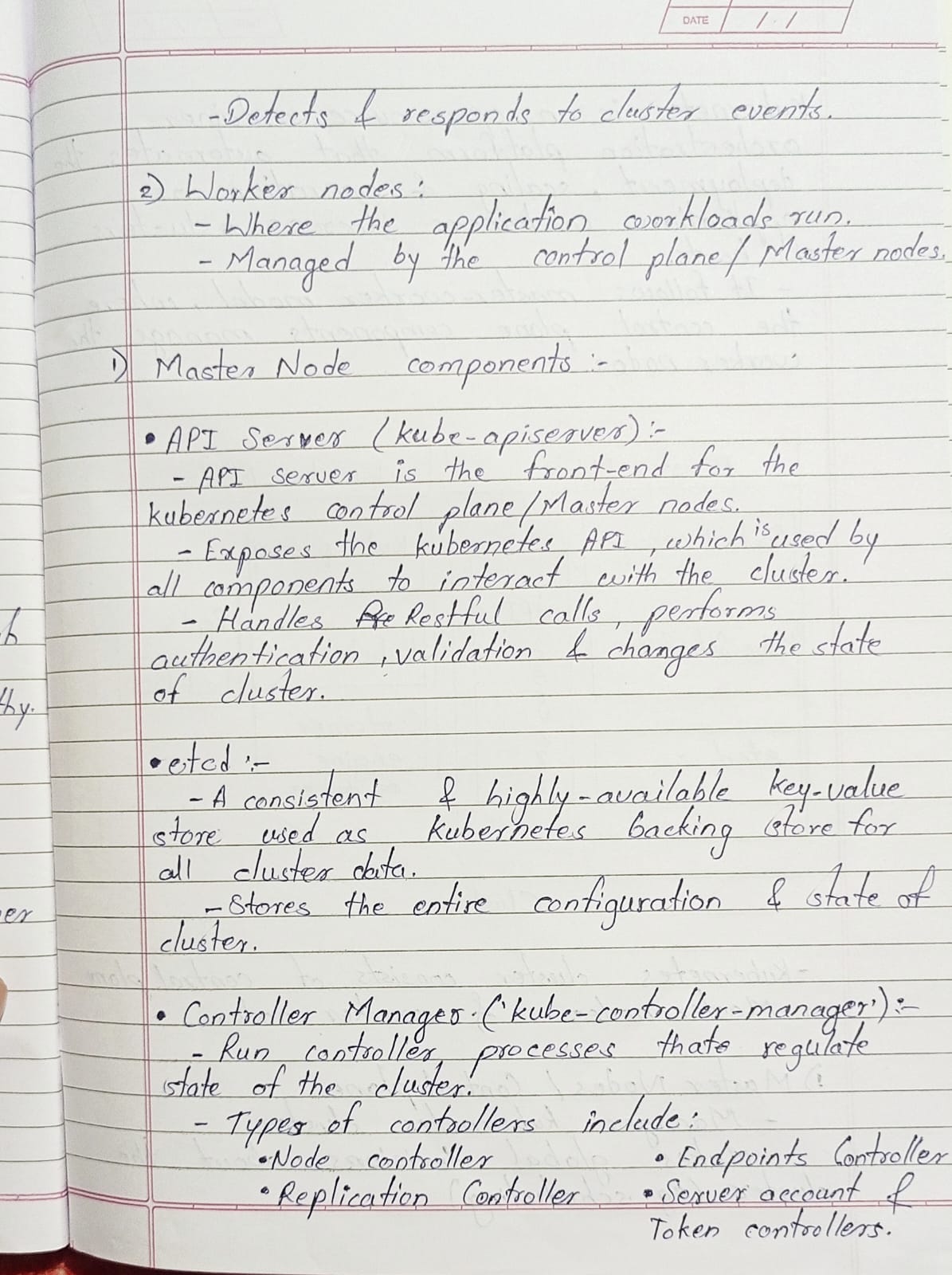
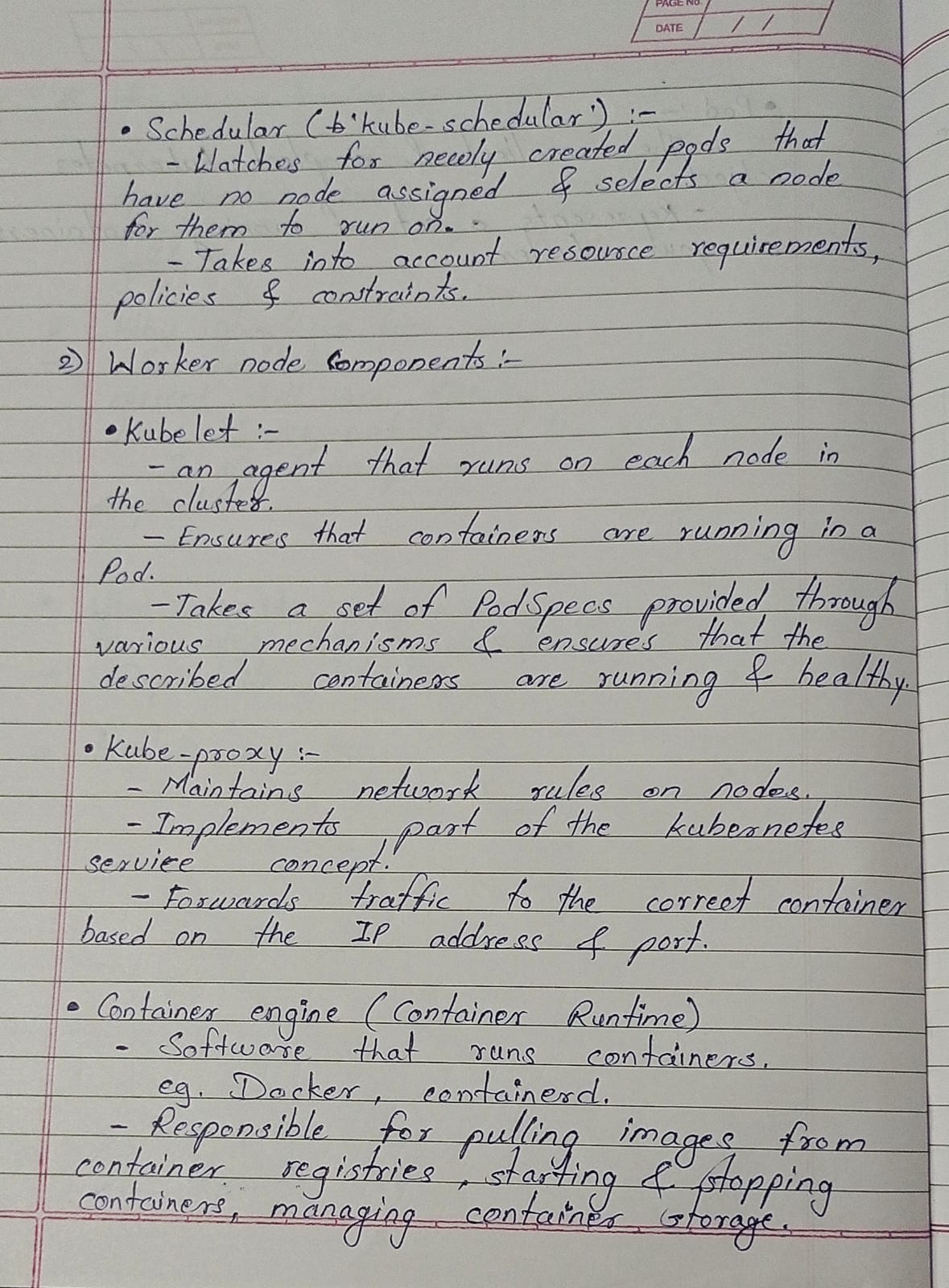
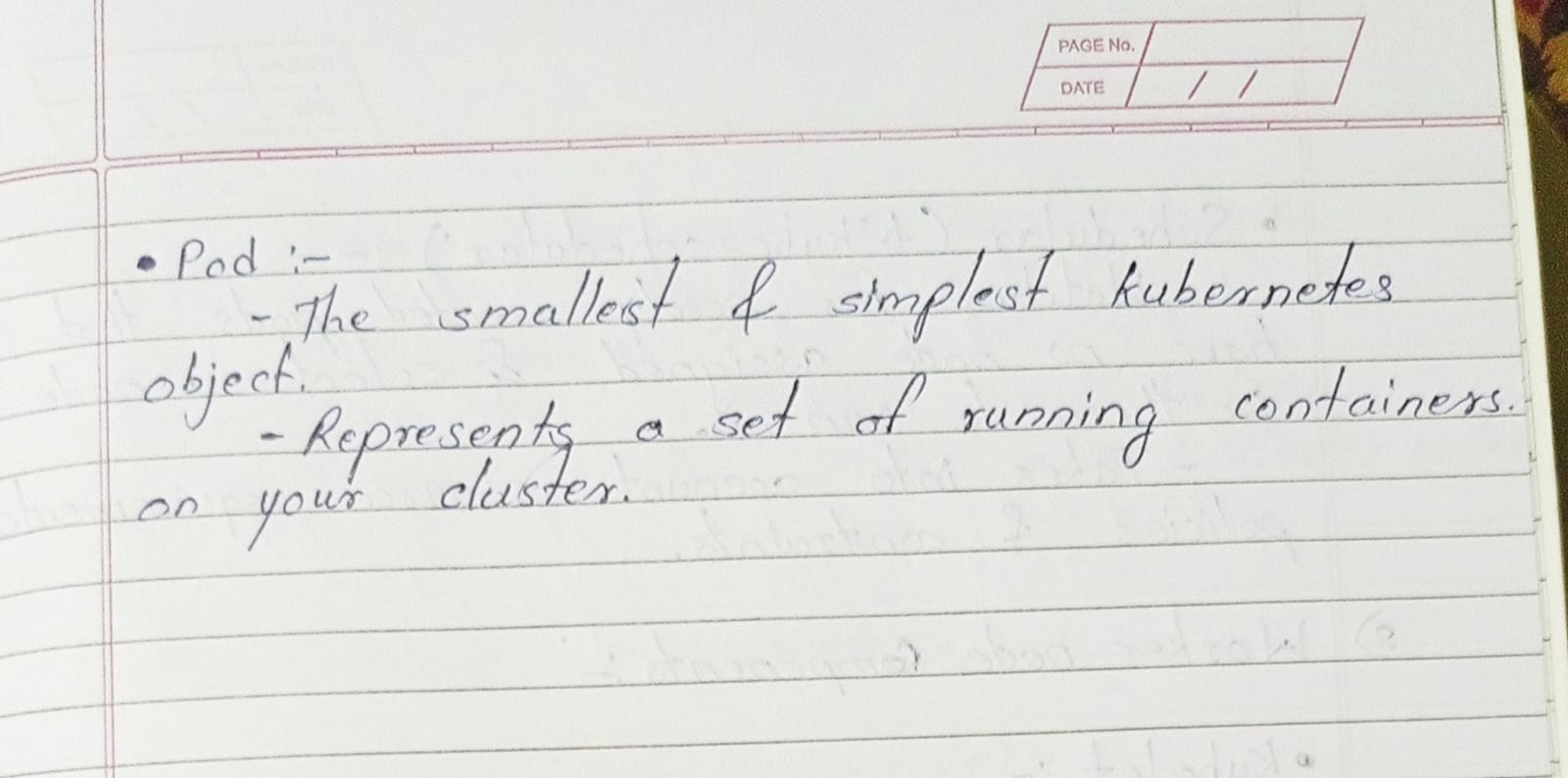
**Kubernetes Task1**

1. **Write a note on Kubernetes Architecture. Explain about each component of Kubernetes cluster.**





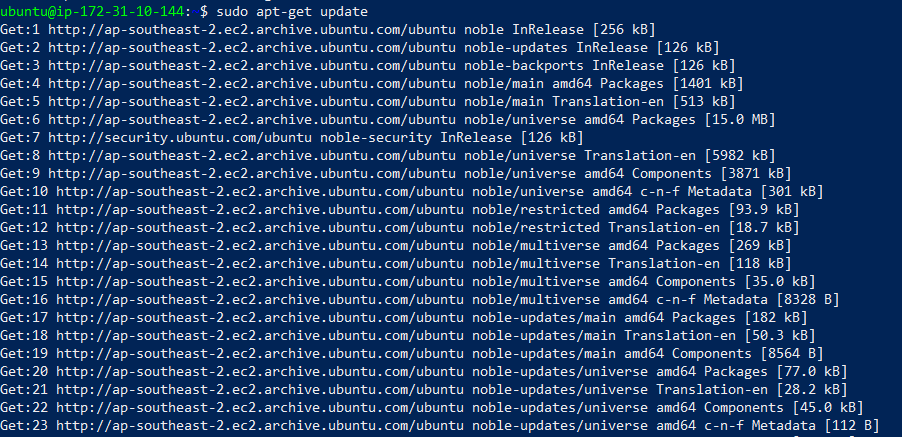


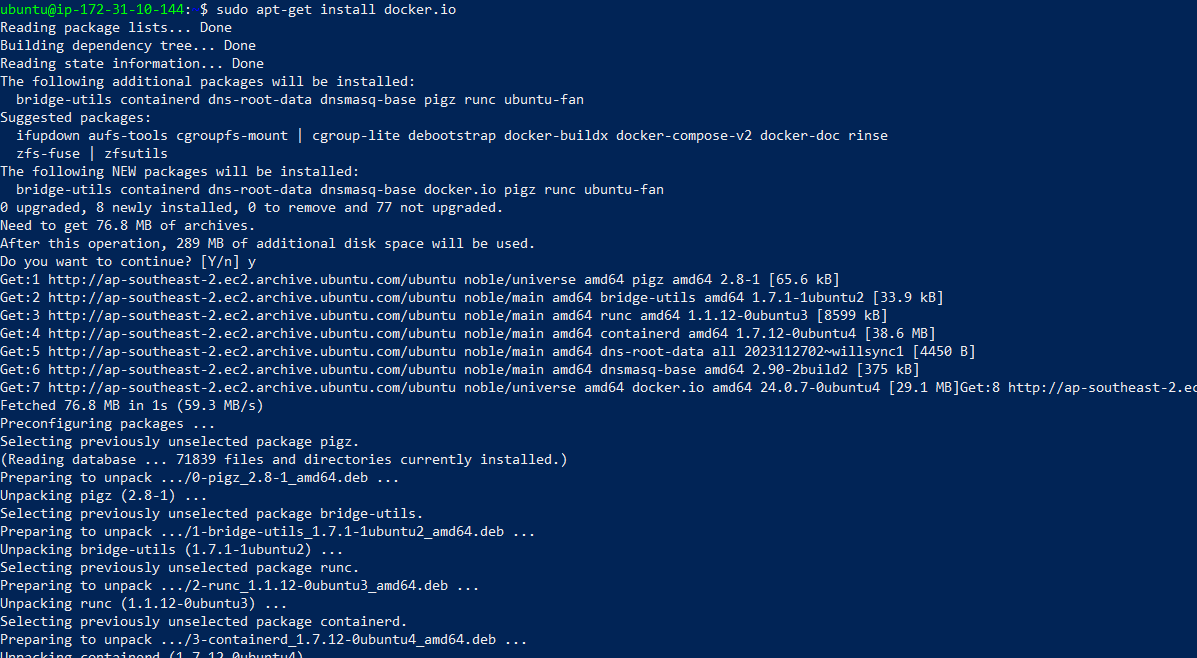


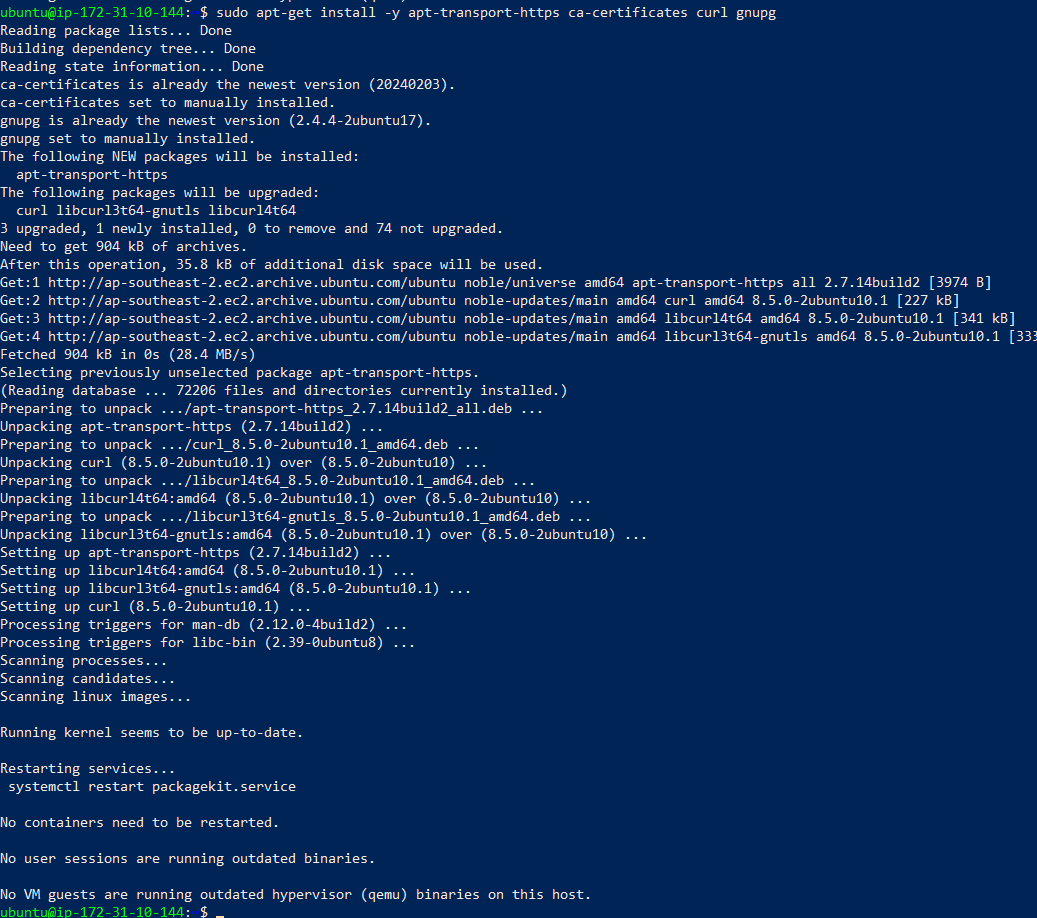
1. **Prepare a documentation on Kubernetes setup on Ubuntu. With screenshot of each command.**

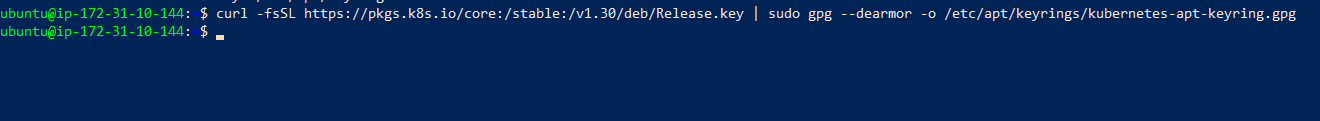
**Kubernetes cluster setup on Ubuntu:**

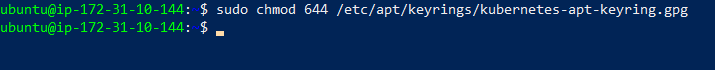
1. **Launch two instances, first one will be master node : for that select instance type as t2.small, Second one will be worker node: for that select instance type as t2.micro.**
2. **Run below commands on both instance.**

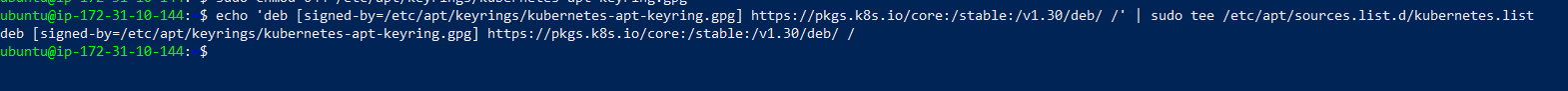




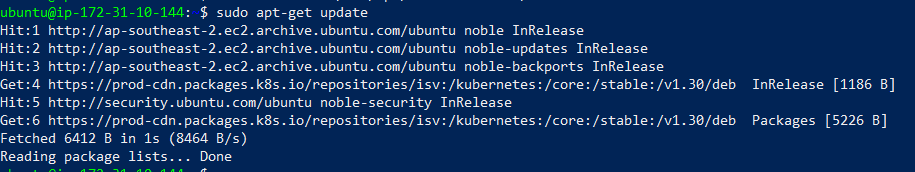


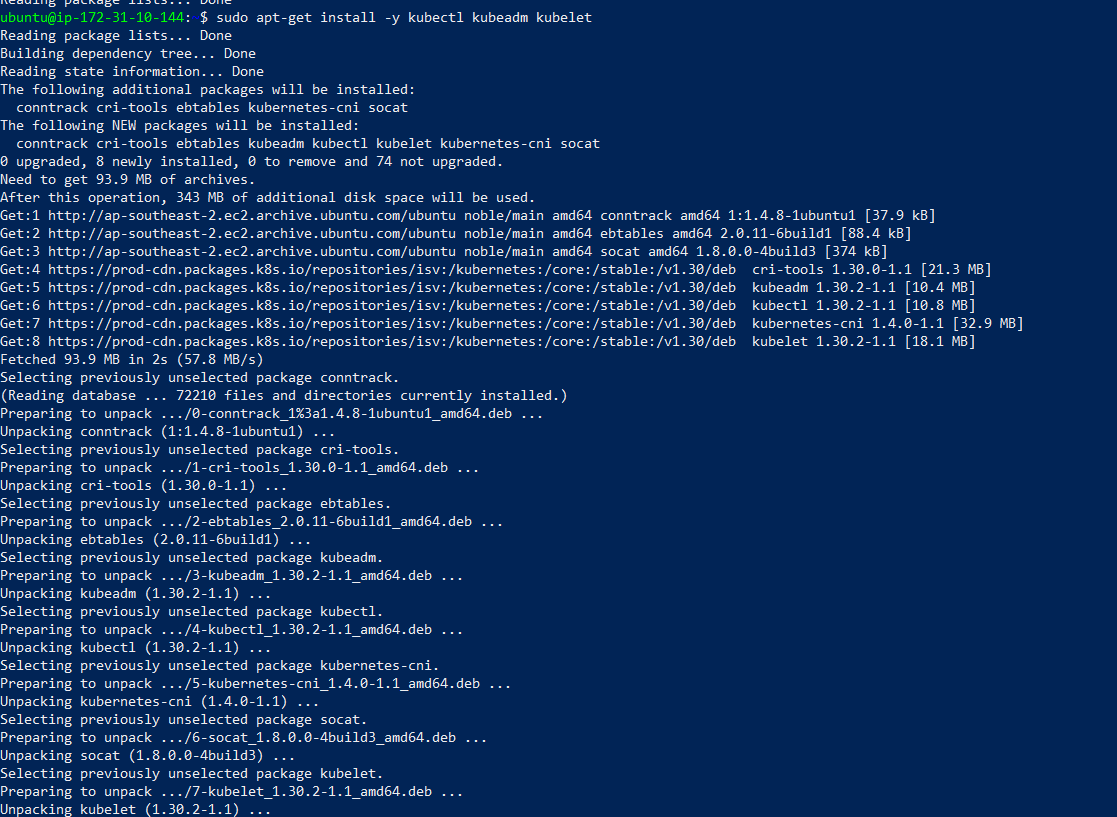


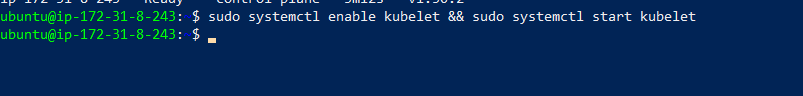












1. **Run below commands only on master instance.**

**Initialize kubeadm**

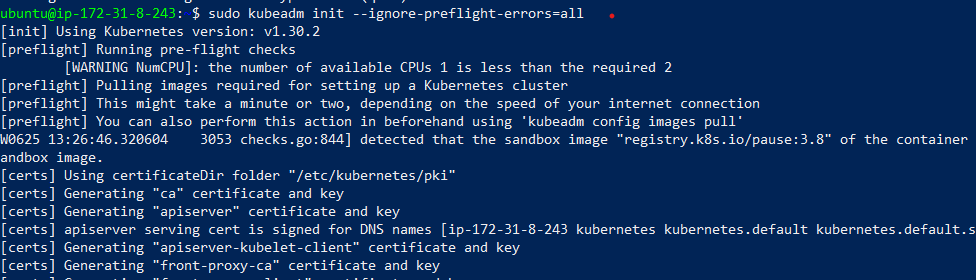
sudo kubeadm init --ignore-preflight-errors=all

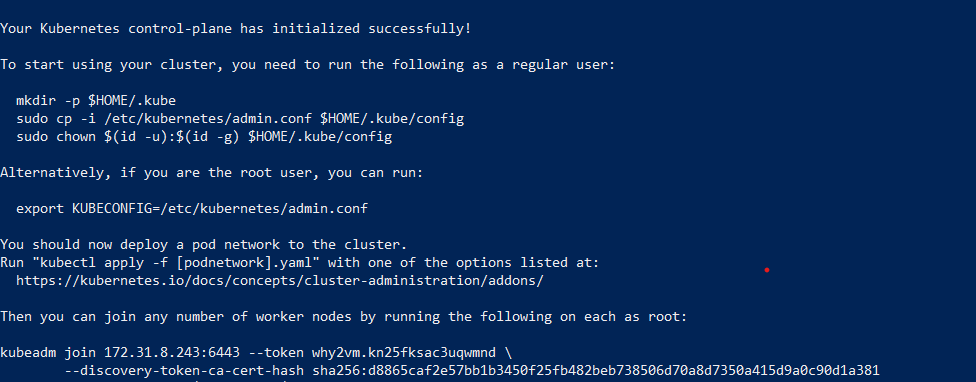
**After running this command we get the kube api token which shows below:**

kubeadm join 172.31.8.243:6443 --token why2vm.kn25fksac3uqwmnd \

--discovery-token-ca-cert-hash sha256:d8865caf2e57bb1b3450f25fb482beb738506d70a8d7350a415d9a0c90d1a381

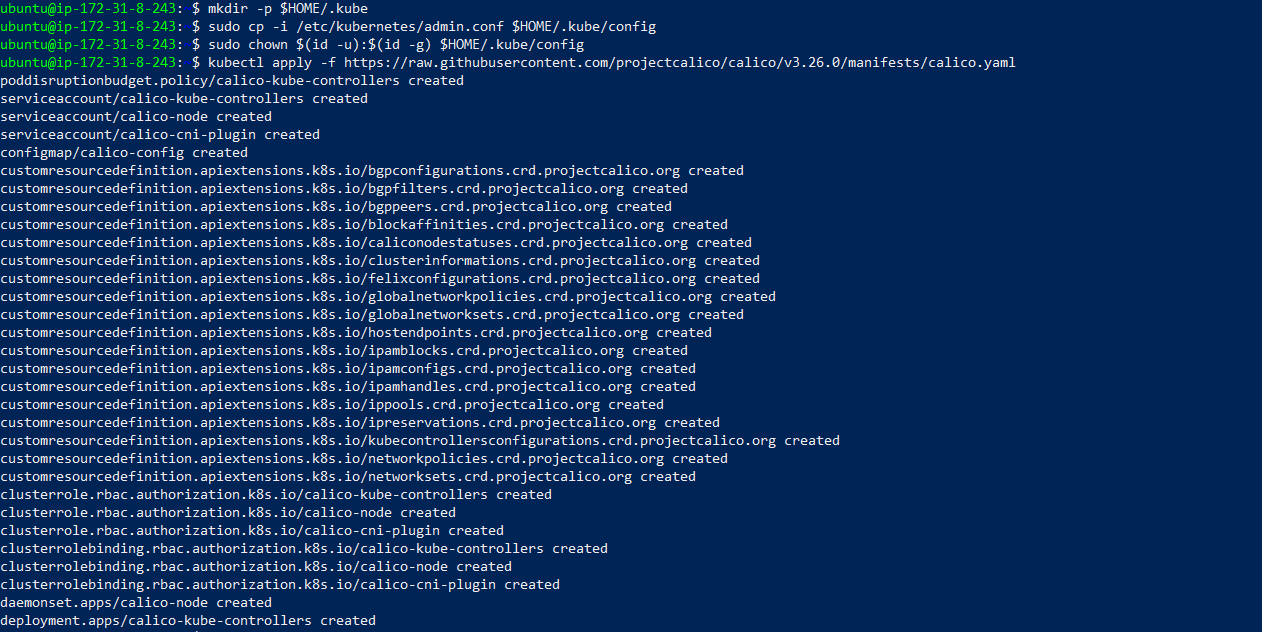
(THIS COMMAND IS GOING TO USED IN WORKER NODE)

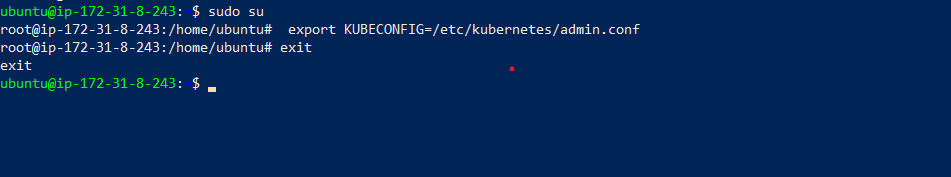




Master node:-

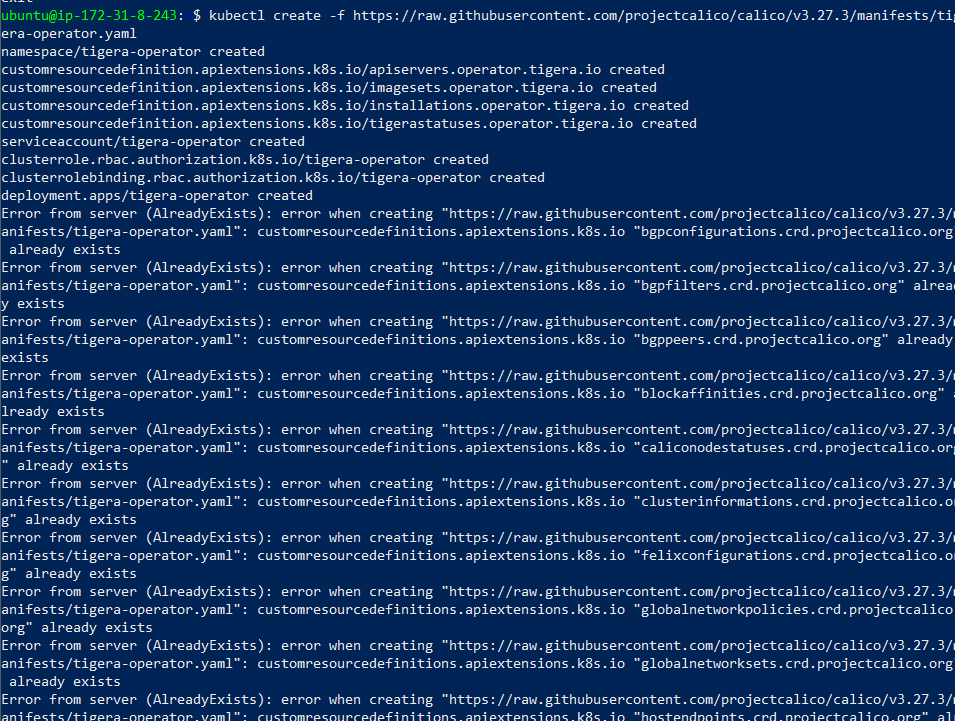
* mkdir -p $HOME/.kube
* sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
* sudo chown $(id -u):$(id -g) $HOME/.kube/config
* sudo su
* export KUBECONFIG=/etc/kubernetes/admin.conf
* exit



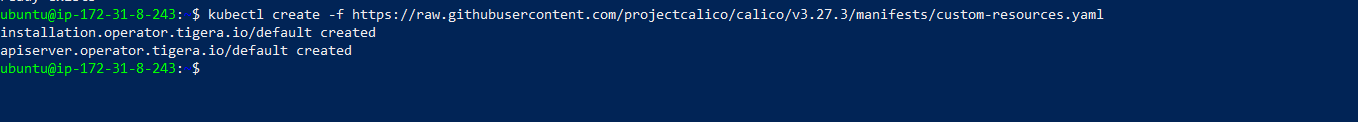


**D) To install Calico on Master**

kubectl create -f <https://raw.githubusercontent.com/projectcalico/calico/v3.27.3/manifests/tigera-operator.yaml>

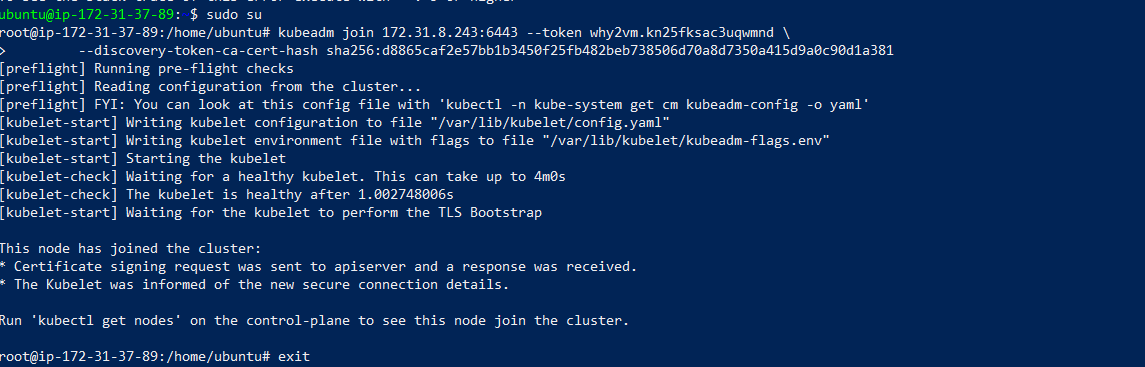


kubectl create -f <https://raw.githubusercontent.com/projectcalico/calico/v3.27.3/manifests/custom-resources.yaml>

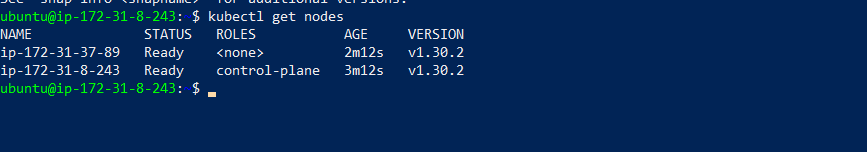


**E)** **Run below commands only on worker instance.**

Copy and paste the kubeadm command which we got at step no. **c)** in worker node**.**

****

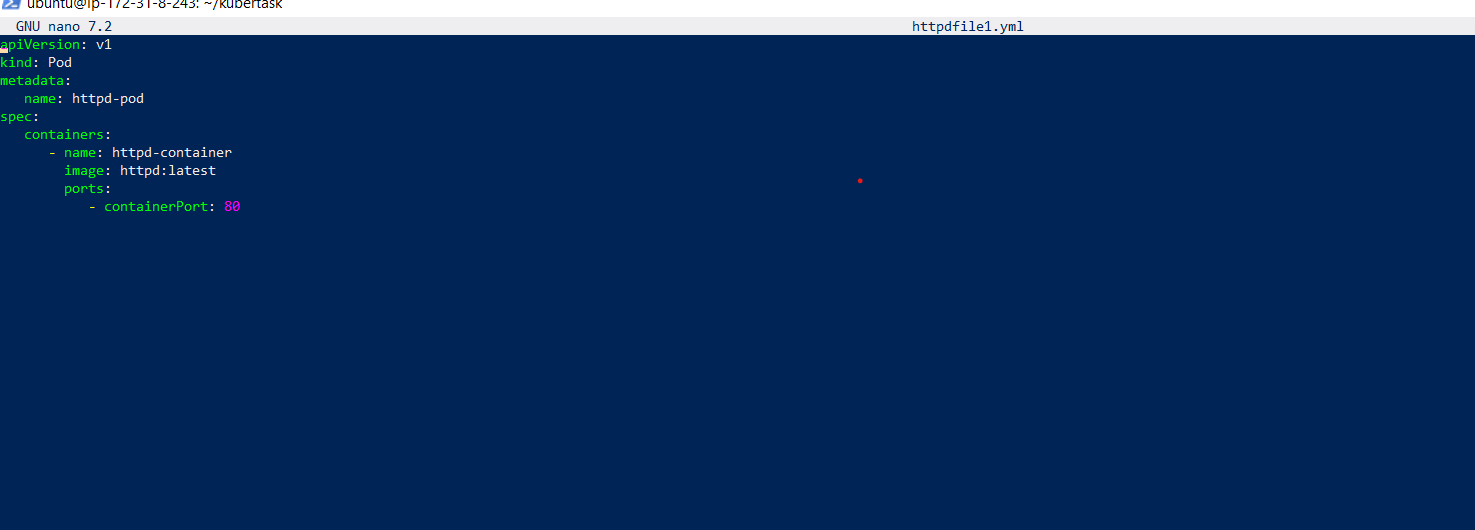
**F) run below command on master node to check connection.**

****

1. **Write a manifest file to create an httpd container in Pod, create pod using that manifest file. Also go inside that httpd container and create own pages and try to access those pages from inside the container.**

**Create a file ,**

* **nano httpdfile.yml**

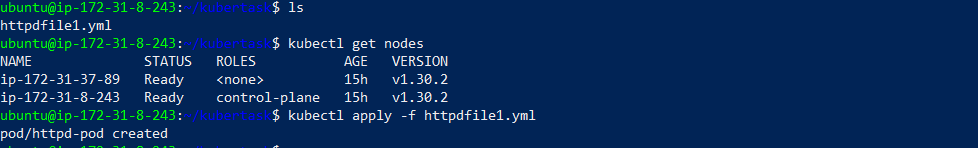
****

**# To create the Pod using this manifest file, we can run the following command:**

* kubectl apply –f httpdfile.yml

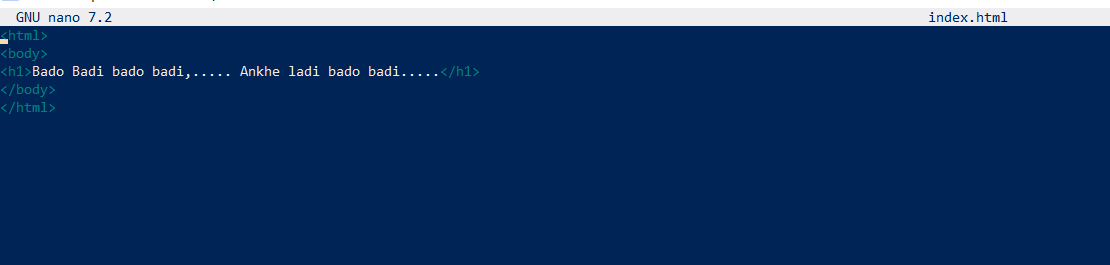
**# Command to check the pods**

* kubectl get nodes

****

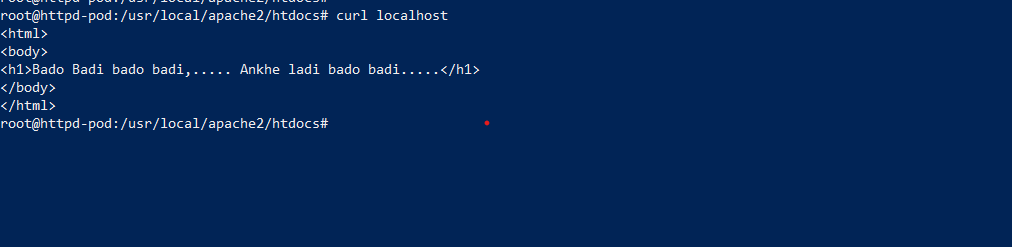
**# Once the Pod is running, We can enter in container & create HTML page.**

* kubectl exec -it httpd-pod /bin/bash
* cd /usr/local/apache2/htdocs
* apt update
* apt install nano
* nano index.html

****

**# To access these pages from inside the container, we can use following command:**

* apt install curl
* curl loalhost



1. **Prepare a documentation in word file with proper screenshots and push that file to your Git hub repository. (send git hub repo link on WhatsApp)**

My github repository link:

<https://github.com/omigit123/Kubernetes-1>