Kubernetes Task

Task Description:

Setup minikube at your local and explore creating namespaces (Go through official documentation).

➔ Step 1: Install dependencies, Docker and Minikube binary

```
pranit@LAPTOP-I9TPEN7J:~$ sudo apt-get install -y curl wget apt-transport-https
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
curl is already the newest version (8.5.0-2ubuntu10.6).
curl set to manually installed.
wget is already the newest version (1.21.4-1ubuntu4.1).
wget set to manually installed.
The following additional packages will be installed:
  apt apt-utils libapt-pkg6.0t64
Suggested packages:
  apt-doc aptitude | synaptic | wajig dpkg-dev powermgmt-base
The following NEW packages will be installed:
  apt-transport-https
The following packages will be upgraded:
  apt apt-utils libapt-pkg6.0t64
3 upgraded, 1 newly installed, 0 to remove and 156 not upgraded.
Need to get 2580 kB of archives.
After this operation, 47.1 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 libapt-pkg6.0t64 amd64 2.8.3 [985 kB]
Get:2 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 apt amd64 2.8.3 [1376 kB]
Get:3 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 apt-utils amd64 2.8.3 [216 kB]
Get:4 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 apt-transport-https all 2.8.3 [3970 B]
Fetched 2580 kB in 2s (1089 kB/s)
(Reading database ... 40875 files and directories currently installed.)
Preparing to unpack .../libapt-pkg6.0t64_2.8.3_amd64.deb ...
Unpacking libapt-pkg6.0t64:amd64 (2.8.3) over (2.7.14build2) ...
Setting up libapt-pkg6.0t64:amd64 (2.8.3) ...
(Reading database ... 40875 files and directories currently installed.)
Preparing to unpack .../archives/apt_2.8.3_amd64.deb ...
```

➔ Installing required dependencies

```
pranit@LAPTOP-I9TPEN7J:~$ sudo apt-get install -y ca-certificates curl gnupg lsb-release
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20240203).
ca-certificates set to manually installed.
curl is already the newest version (8.5.0-2ubuntu10.6).
lsb-release is already the newest version (12.0-2).
lsb-release set to manually installed.
```

➔ Adding Docker’s official GPG Key and setting up repo

```
pranit@LAPTOP-I9TPEN7J:~$ sudo mkdir -p /etc/apt/keyrings
pranit@LAPTOP-I9TPEN7J:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
pranit@LAPTOP-I9TPEN7J:~$ echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \
  $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

➔ Installing Docker Engine

```
pranit@LAPTOP-I9TPEN7J:~$ sudo apt-get install -y docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  docker-ce-rootless-extras iptables libip4tc2 libip6tc2 libnetfilter-conntrack3 libnfnetlink0 libnftables1 libnftnl11 libslirp0 nftables
  slirp4netns
Suggested packages:
  cgroupfs-mount | cgroup-lite docker-model-plugin firewalld
The following NEW packages will be installed:
  containerd.io docker-buildx-plugin docker-ce docker-ce-cli docker-ce-rootless-extras docker-compose-plugin iptables libip4tc2 libip6tc2
  libnetfilter-conntrack3 libnfnetlink0 libnftables1 libnftnl11 libslirp0 nftables pigz slirp4netns
0 upgraded, 17 newly installed, 0 to remove and 145 not upgraded.
```

➔ Starting Minikube with Docker Driver

```
pranit@LAPTOP-I9TPEN7J:~$ minikube start --driver=docker
😄  minikube v1.36.0 on Ubuntu 24.04 (amd64)
✨  Using the docker driver based on user configuration
📌  Using Docker driver with root privileges
👍  Starting "minikube" primary control-plane node in "minikube" cluster
🚜  Pulling base image v0.0.47 ...
💾  Downloading Kubernetes v1.33.1 preload ...
    > preloaded-images-k8s-v18-v1...:  347.04 MiB / 347.04 MiB  100.00% 6.02 Mi
    > gcr.io/k8s-minikube/kicbase...:  502.26 MiB / 502.26 MiB  100.00% 5.97 Mi
🔥  Creating docker container (CPUs=2, Memory=2200MB) ...
🐳  Preparing Kubernetes v1.33.1 on Docker 28.1.1 ...
    ▪ Generating certificates and keys ...
    ▪ Booting up control plane ...
    ▪ Configuring RBAC rules ...
🔗  Configuring bridge CNI (Container Networking Interface) ...
🔎  Verifying Kubernetes components...
    ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🌟  Enabled addons: storage-provisioner, default-storageclass
💡  kubectl not found. If you need it, try: 'minikube kubectl -- get pods -A'
🏄  Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
pranit@LAPTOP-I9TPEN7J:~$ minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured
```

➔ Step 2: Interacting and exploring Namespaces

```
pranit@LAPTOP-I9TPEN7J:~$ kubectl get nodes
Command 'kubectl' not found, but can be installed with:
sudo snap install kubectl
pranit@LAPTOP-I9TPEN7J:~$ minikube kubectl -- get nodes
    > kubectl.sha256:  64 B / 64 B [-------------------------] 100.00% ? p/s 0s
    > kubectl:  57.34 MiB / 57.34 MiB [-------------] 100.00% 9.26 MiB p/s 6.4s
NAME       STATUS   ROLES           AGE    VERSION
minikube   Ready    control-plane   112s   v1.33.1
pranit@LAPTOP-I9TPEN7J:~$ kubectl get nodes
Command 'kubectl' not found, but can be installed with:
sudo snap install kubectl
pranit@LAPTOP-I9TPEN7J:~$ alias kubectl="minikube kubectl --"
pranit@LAPTOP-I9TPEN7J:~$ kubectl get nodes
NAME       STATUS   ROLES           AGE     VERSION
minikube   Ready    control-plane   2m14s   v1.33.1
pranit@LAPTOP-I9TPEN7J:~$ kubectl get nodes -A
NAME       STATUS   ROLES           AGE     VERSION
minikube   Ready    control-plane   2m22s   v1.33.1
pranit@LAPTOP-I9TPEN7J:~$ kubectl create namespace my-namespace
namespace/my-namespace created
pranit@LAPTOP-I9TPEN7J:~$ kubectl get namespaces
NAME              STATUS   AGE
default           Active   3m1s
kube-node-lease   Active   3m
kube-public       Active   3m1s
kube-system       Active   3m1s
my-namespace      Active   7s
pranit@LAPTOP-I9TPEN7J:~$ kubectl run nginx-pod --image=nginx --namespace=my-namespace
pod/nginx-pod created
pranit@LAPTOP-I9TPEN7J:~$ kubectl get pods -n my-namespace
NAME        READY   STATUS              RESTARTS   AGE
nginx-pod   0/1     ContainerCreating   0          6s
pranit@LAPTOP-I9TPEN7J:~$
```