Task

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

1. **Install Minikube (Ubuntu)**

```
ubuntu@ip-172-31-36-238:~$ sudo apt 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 NEW packages will be installed:
  apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 78 not upgraded.
Need to get 3970 B of archives.
After this operation, 36.9 kB of additional disk space will be used.
Get:1 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd
64 apt-transport-https all 2.8.3 [3970 B]
Fetched 3970 B in 0s (252 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 71752 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.8.3_all.deb ...
Unpacking apt-transport-https (2.8.3) ...
Setting up apt-transport-https (2.8.3) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-36-238:~$ curl -LO https://storage.googleapis.com/minikube/rele
ases/latest/minikube-linux-amd64
sudo install minikube-linux-amd64 /usr/local/bin/minikube
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  128M  100  128M    0     0  28.6M      0  0:00:04  0:00:04 --:--:-- 28.6M
ubuntu@ip-172-31-36-238:~$ minikube version
minikube version: v1.38.0
commit: de81223c61ab1bd97dcfcfa6d9d5c59e5da4a0cf
ubuntu@ip-172-31-36-238:~$
```

2. **Install kubectl (Kubernetes command tool)**

```
ubuntu@ip-172-31-36-238:~$ sudo apt install -y ca-certificates curl
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).
0 upgraded, 0 newly installed, 0 to remove and 78 not upgraded.
ubuntu@ip-172-31-36-238:~$ curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   138  100   138    0     0    777      0 --:--:-- --:--:-- --:--:--   775
100 55.8M  100 55.8M    0     0   115M      0 --:--:-- --:--:-- --:--:--  115M
ubuntu@ip-172-31-36-238:~$ chmod +x kubectl
ubuntu@ip-172-31-36-238:~$ sudo mv kubectl /usr/local/bin/
ubuntu@ip-172-31-36-238:~$ kubectl version --client
Client Version: v1.35.0
Kustomize Version: v5.7.1
ubuntu@ip-172-31-36-238:~$
```

3. Start minikube

```
ubuntu@ip-172-31-2-233:~$ minikube start --driver=docker
* minikube v1.38.0 on Ubuntu 24.04
* Using the docker driver based on user configuration
! Starting v1.39.0, minikube will default to "containerd" container runtime. See #21973 for more info.

X The requested memory allocation of 1910MiB does not leave room for system overhead (total system memory: 1910MiB). You may face stability issues.
* Suggestion: Start minikube with less memory allocated: 'minikube start --memory=1910mb'

* Using Docker driver with root privileges
* Starting "minikube" primary control-plane node in "minikube" cluster
* Pulling base image v0.0.49 ...
* Downloading Kubernetes v1.35.0 preload ...
    > gcr.io/k8s-minikube/kicbase...:  514.15 MiB / 514.16 MiB  100.00% 97.01 M
    > preloaded-images-k8s-v18-v1...:  271.45 MiB / 271.45 MiB  100.00% 31.46 M
* Creating docker container (CPUs=2, Memory=1910MB) ...
* Preparing Kubernetes v1.35.0 on Docker 29.2.0 ...
* Configuring bridge CNI (Container Networking Interface) ...
* Verifying Kubernetes components...
  - Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Enabled addons: storage-provisioner, default-storageclass
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
ubuntu@ip-172-31-2-233:~$ kubectl get nodes
NAME       STATUS   ROLES           AGE   VERSION
minikube   Ready    control-plane   14s   v1.35.0
ubuntu@ip-172-31-2-233:~$
```

4. Create a New Namespace

```
ubuntu@ip-172-31-2-233:~$ kubectl create namespace dev
namespace/dev created
ubuntu@ip-172-31-2-233:~$ kubectl get namespaces
NAME              STATUS   AGE
default           Active   7m21s
dev               Active   30s
kube-node-lease   Active   7m21s
kube-public       Active   7m21s
kube-system       Active   7m21s
ubuntu@ip-172-31-2-233:~$
```

5. Use Namespace in Commands

Create a pod in `dev` namespace:

```
ubuntu@ip-172-31-2-233:~$ kubectl run mypod --image=nginx --namespace=dev
pod/mypod created
ubuntu@ip-172-31-2-233:~$ kubectl get pods -n dev
NAME    READY   STATUS    RESTARTS   AGE
mypod   1/1     Running   0          2m11s
```