Practice LAB

To change the user name in Linux:

1.check your user name
>whoami

2. Create a temporary user
>sudo adduser tempadmin

3. Add the new user to sudoers
> sudo usermod -aG sudo tempadmin

4. Logout from the current user via GUI and login to tempadmin

5. In the terminal of tempadmin,
>sudo usermod -l 1RV24MC025_CHANDAN_K chandan

6. Now type the below command in original user after logging again via GUI:
>whoami

```
1RV24MC025_CHANDAN_K@chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC:~$ whoami
1RV24MC025_CHANDAN_K
1RV24MC025_CHANDAN_K@chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC:~$ sudo deluser tempadmin
[sudo] password for 1RV24MC025_CHANDAN_K:
info: Removing crontab ...
info: Removing user `tempadmin' ...
```

7. Remove the temp user,delete it
>sudo deluser tempadmin
>sudo rm -r /home/tempadmin

Install DOCKER using the `apt` repository

Before you install Docker Engine for the first time on a new host machine, you need to set up the Docker `apt` repository. Afterward, you can install and update Docker from the repository.

1. Set up Docker's `apt`repository.

```
# Add Docker's official GPG key:
sudo apt-get update
sudo apt-get install ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o
/etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc

# Add the repository to Apt sources:
echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc]
https://download.docker.com/linux/ubuntu \
  $(. /etc/os-release && echo "${UBUNTU_CODENAME:-$VERSION_CODENAME}")
stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update
```

```
1RV24MC025_CHANDAN_K@chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC:~$ sudo apt-get install ca-certificates curl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20240203).
curl is already the newest version (8.5.0-2ubuntu10.6).
The following packages were automatically installed and are no longer required:
  libllvm17t64 python3-netifaces
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 476 not upgraded.
1RV24MC025_CHANDAN_K@chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC:~$ sudo install -m 0755 -d /etc/apt/keyrings
1RV24MC025_CHANDAN_K@chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC:~$ sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
1RV24MC025_CHANDAN_K@chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC:~$ sudo chmod a+r /etc/apt/keyrings/docker.asc
1RV24MC025_CHANDAN_K@chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC:~$ echo \
>  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \
  $(. /etc/os-release && echo "${UBUNTU_CODENAME:-$VERSION_CODENAME}") stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
1RV24MC025_CHANDAN_K@chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC:~$ sudo apt-get update
Hit:1 https://download.docker.com/linux/ubuntu noble InRelease
Hit:2 https://packages.microsoft.com/repos/code stable InRelease
Hit:3 http://in.archive.ubuntu.com/ubuntu noble InRelease
```

2.Install the Docker packages.

> sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin Docker-compose-plugin

```
1RV24MC025_CHANDAN_K@chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC:~$ sudo apt-get install 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 packages were automatically installed and are no longer required:
  libllvm17t64 python3-netifaces
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  docker-ce-rootless-extras
Suggested packages:
  cgroupfs-mount | cgroup-lite docker-model-plugin
The following packages will be upgraded:
  containerd.io docker-buildx-plugin docker-ce docker-ce-cli docker-ce-rootless-extras docker-compose-plugin
6 upgraded, 0 newly installed, 0 to remove and 470 not upgraded.
Need to get 105 MB of archives.
After this operation, 5,332 kB disk space will be freed.
Do you want to continue? [Y/n] Y
Get:1 https://download.docker.com/linux/ubuntu noble/stable amd64 docker-ce-cli amd64 5:28.5.1-1~ubuntu.24.04~noble [16.5 MB]
Get:2 https://download.docker.com/linux/ubuntu noble/stable amd64 containerd.io amd64 1.7.28-1~ubuntu.24.04~noble [31.9 MB]
Get:3 https://download.docker.com/linux/ubuntu noble/stable amd64 docker-ce amd64 5:28.5.1-1~ubuntu.24.04~noble [19.7 MB]
Get:4 https://download.docker.com/linux/ubuntu noble/stable amd64 docker-buildx-plugin amd64 0.29.1-1~ubuntu.24.04~noble [15.9 MB]
Get:5 https://download.docker.com/linux/ubuntu noble/stable amd64 docker-ce-rootless-extras amd64 5:28.5.1-1~ubuntu.24.04~noble [6,481 kB]
Get:6 https://download.docker.com/linux/ubuntu noble/stable amd64 docker-compose-plugin amd64 2.40.0-1~ubuntu.24.04~noble [14.2 MB]
Fetched 105 MB in 12s (8,880 kB/s)
(Reading database ... 232379 files and directories currently installed.)
```

3. Check docker status

> sudo systemctl status docker

The Docker service starts automatically after installation. To verify that Docker is running, use:

```
1RV24MC025_CHANDAN_K@chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
     Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: enabled)
     Active: active (running) since Fri 2025-10-17 09:51:34 IST; 8min ago
TriggeredBy: ● docker.socket
       Docs: https://docs.docker.com
   Main PID: 17255 (dockerd)
      Tasks: 17
     Memory: 21.1M (peak: 24.9M)
        CPU: 997ms
     CGroup: /system.slice/docker.service
             └─17255 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Oct 17 09:51:33 chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC dockerd[17255]: time="2025-10-17T09:51:33.349875445+05:30" level=info msg="Creating a containerd client" address=/run/conta>
Oct 17 09:51:33 chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC dockerd[17255]: time="2025-10-17T09:51:33.406455857+05:30" level=info msg="[graphdriver] using prior storage driver: overla>
Oct 17 09:51:33 chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC dockerd[17255]: time="2025-10-17T09:51:33.406683289+05:30" level=info msg="Loading containers: start."
Oct 17 09:51:34 chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC dockerd[17255]: time="2025-10-17T09:51:34.706472196+05:30" level=info msg="Loading containers: done."
Oct 17 09:51:34 chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC dockerd[17255]: time="2025-10-17T09:51:34.745704007+05:30" level=info msg="Docker daemon" commit=f8215cc containerd-snapsho>
Oct 17 09:51:34 chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC dockerd[17255]: time="2025-10-17T09:51:34.745817833+05:30" level=info msg="Initializing buildkit"
Oct 17 09:51:34 chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC dockerd[17255]: time="2025-10-17T09:51:34.796470326+05:30" level=info msg="Completed buildkit initialization"
Oct 17 09:51:34 chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC dockerd[17255]: time="2025-10-17T09:51:34.807308340+05:30" level=info msg="Daemon has completed initialization"
Oct 17 09:51:34 chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC dockerd[17255]: time="2025-10-17T09:51:34.807502306+05:30" level=info msg="API listen on /run/docker.sock"
Oct 17 09:51:34 chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC systemd[1]: Started docker.service - Docker Application Container Engine.
```

Some systems may have this behavior disabled and will require a manual start:
sudo systemctl start docker

4. Run a sample container

>sudo docker run hello-world

```
1RV24MC025_CHANDAN_K@chandan-Vivobook-ASUSLaptop-M6500QC-M6500QC:~$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
17eec7bbc9d7: Pull complete
Digest: sha256:6dc565aa630927052111f823c30948cf83670a3903ffa3849f1488ab517f891
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/
```