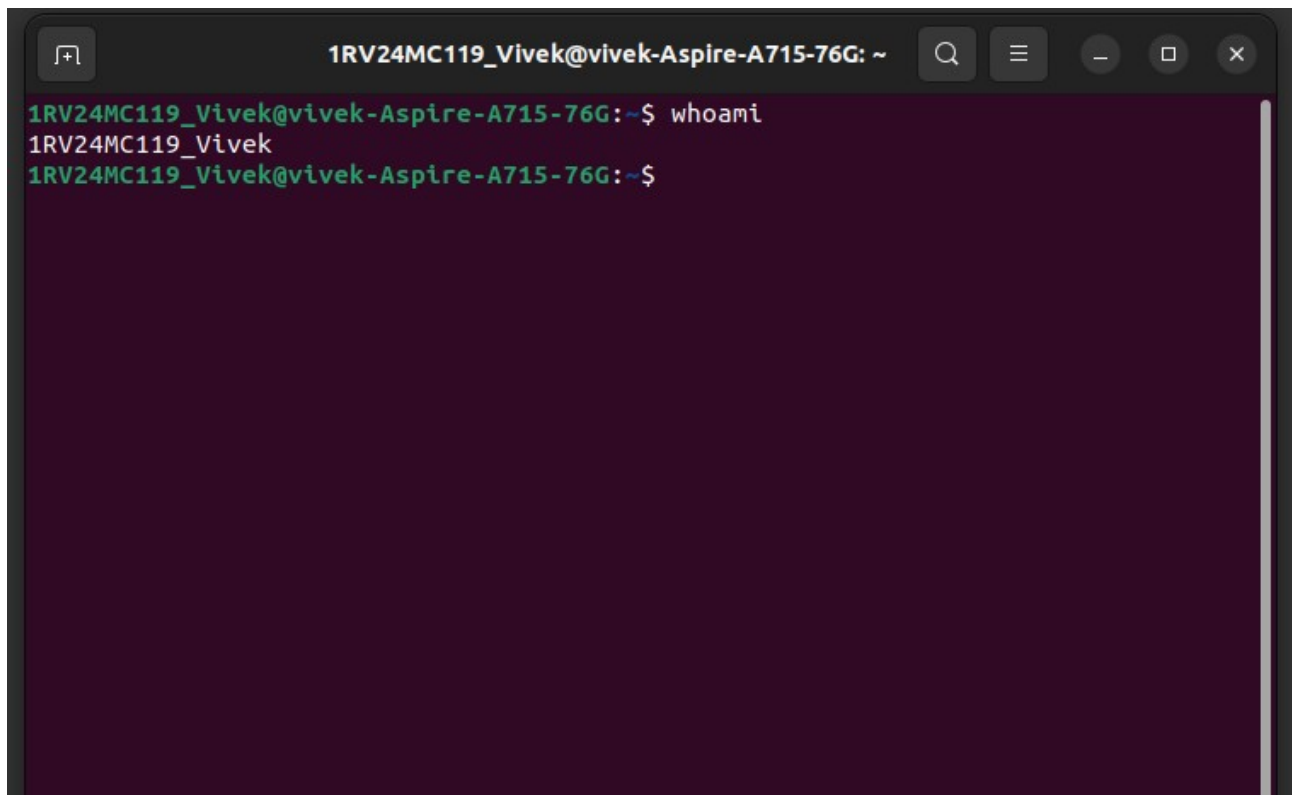


LAB 1 – Docker Installation

1. Changed USER NAME to USN_NAME format



```
1RV24MC119_Vivek@vivek-Aspire-A715-76G: ~  
1RV24MC119_Vivek@vivek-Aspire-A715-76G:~$ whoami  
1RV24MC119_Vivek  
1RV24MC119_Vivek@vivek-Aspire-A715-76G:~$
```

2. Docker Installation

- 2.1 Add Docker's official GPG key



```
1RV24MC119_Vivek@vivek-Aspire-A715-76G: ~  
1RV24MC119_Vivek@vivek-Aspire-A715-76G:~$ curl -fsSL https://download.docker.com  
/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg  
1RV24MC119_Vivek@vivek-Aspire-A715-76G:~$
```

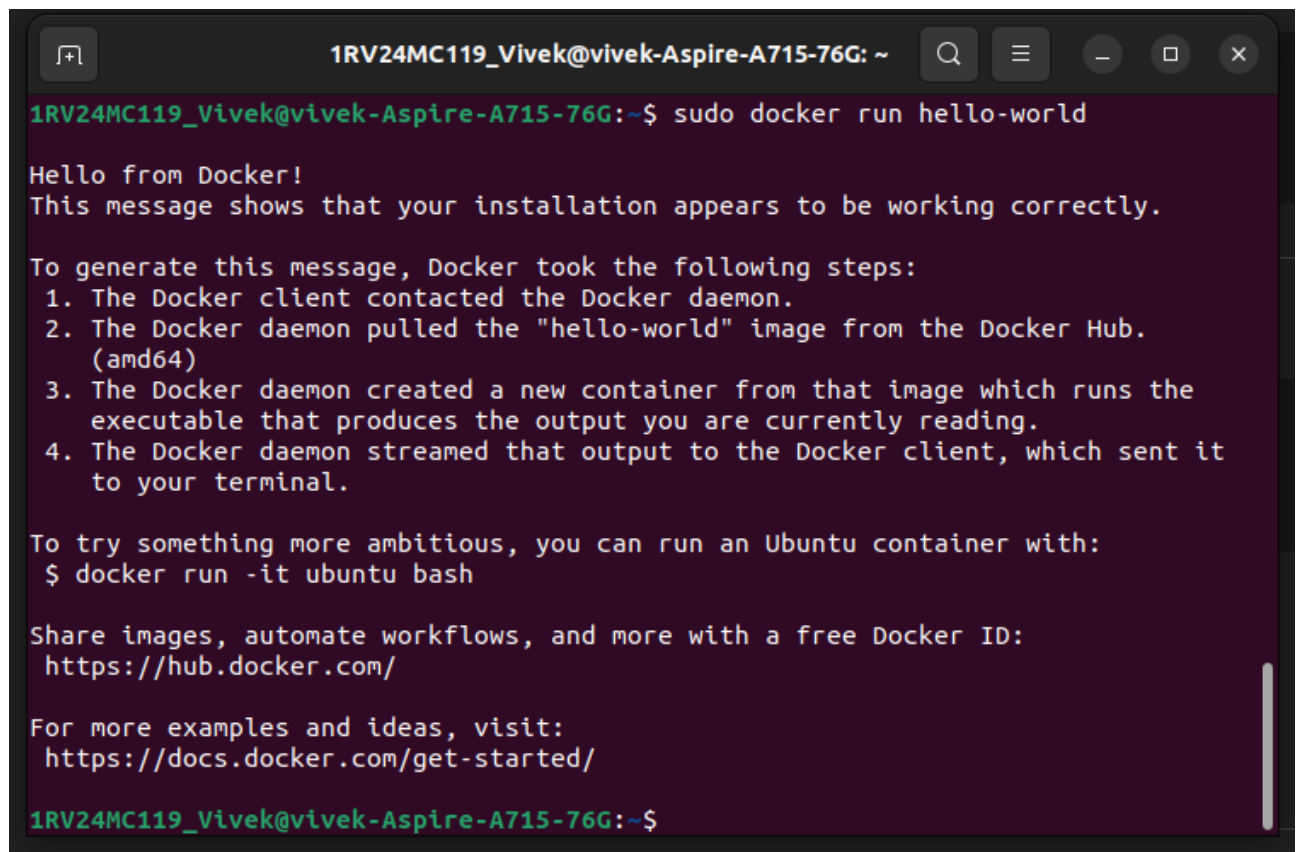
2.2 Install Docker Engine + CLI + compose

```
1RV24MC119_Vivek@vivek-Aspire-A715-76G: ~  
Hit:8 http://archive.ubuntu.com/ubuntu jammy-backports InRelease  
Hit:9 https://ppa.launchpadcontent.net/gns3/ppa/ubuntu jammy InRelease  
Hit:10 http://archive.ubuntu.com/ubuntu jammy-security InRelease  
Fetched 106 kB in 2s (67.3 kB/s)  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
159 packages can be upgraded. Run 'apt list --upgradable' to see them.  
1RV24MC119_Vivek@vivek-Aspire-A715-76G:~$ sudo apt install docker-ce docker-ce-cl  
li containerd.io docker-buildx-plugin docker-compose-plugin -y  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
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-compose-plugin  
5 upgraded, 0 newly installed, 0 to remove and 154 not upgraded.  
Need to get 98.2 MB of archives.  
After this operation, 7,468 kB of additional disk space will be used.  
Get:1 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-ce-cli  
amd64 5:28.5.1-1~ubuntu.22.04~jammy [16.5 MB]  
8% [1 docker-ce-cli 10.4 MB/16.5 MB 63%]  
Docker
```

2.3 Check Docker Version

```
1RV24MC119_Vivek@vivek-Aspire-A715-76G: ~  
Unpacking docker-ce-cli (5:28.5.1-1~ubuntu.22.04~jammy) over (5:28.3.0-1~ubuntu.  
22.04~jammy) ...  
Preparing to unpack .../containerd.io_1.7.28-1~ubuntu.22.04~jammy_amd64.deb ...  
Unpacking containerd.io (1.7.28-1~ubuntu.22.04~jammy) over (1.7.27-1) ...  
Preparing to unpack .../docker-ce_5%3a28.5.1-1~ubuntu.22.04~jammy_amd64.deb ...  
Unpacking docker-ce (5:28.5.1-1~ubuntu.22.04~jammy) over (5:28.3.0-1~ubuntu.22.0  
4~jammy) ...  
Preparing to unpack .../docker-buildx-plugin_0.29.1-1~ubuntu.22.04~jammy_amd64.d  
eb ...  
Unpacking docker-buildx-plugin (0.29.1-1~ubuntu.22.04~jammy) over (0.25.0-1~ubun  
tu.22.04~jammy) ...  
Preparing to unpack .../docker-compose-plugin_2.40.0-1~ubuntu.22.04~jammy_amd64.  
deb ...  
Unpacking docker-compose-plugin (2.40.0-1~ubuntu.22.04~jammy) over (2.37.3-1~ubu  
ntu.22.04~jammy) ...  
Setting up docker-buildx-plugin (0.29.1-1~ubuntu.22.04~jammy) ...  
Setting up containerd.io (1.7.28-1~ubuntu.22.04~jammy) ...  
Setting up docker-compose-plugin (2.40.0-1~ubuntu.22.04~jammy) ...  
Setting up docker-ce-cli (5:28.5.1-1~ubuntu.22.04~jammy) ...  
Setting up docker-ce (5:28.5.1-1~ubuntu.22.04~jammy) ...  
Processing triggers for man-db (2.10.2-1) ...  
1RV24MC119_Vivek@vivek-Aspire-A715-76G:~$ docker --version  
Docker version 28.5.1, build e180ab8  
1RV24MC119_Vivek@vivek-Aspire-A715-76G:~$
```

2.4 Testing Docker

A terminal window with a dark background and light green text. The window title bar shows the user '1RV24MC119_Vivek' on a machine named 'vivek-Aspire-A715-76G'. The terminal displays the command 'sudo docker run hello-world' and its output, which includes a welcome message, a list of steps Docker took to run the container, and links to Docker documentation and the Docker Hub. The prompt returns to the shell after the output.

```
1RV24MC119_Vivek@vivek-Aspire-A715-76G: ~  
1RV24MC119_Vivek@vivek-Aspire-A715-76G:~$ sudo docker run hello-world  
  
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/  
  
1RV24MC119_Vivek@vivek-Aspire-A715-76G:~$
```

