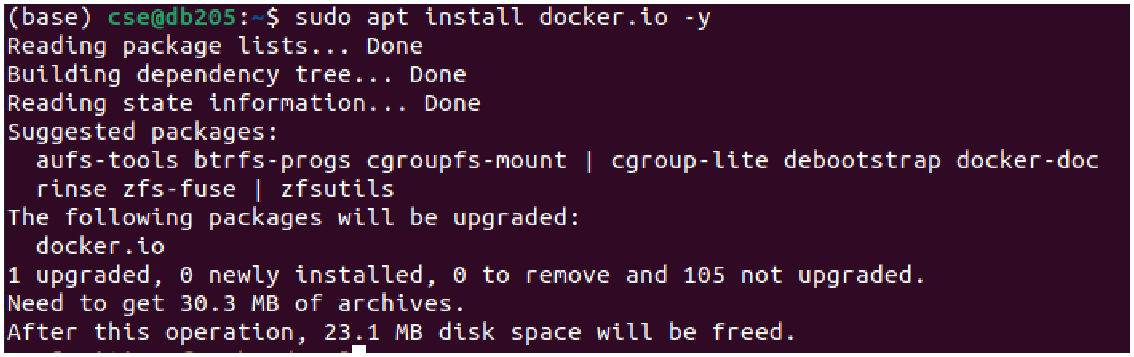
Program 6 : Explore Docker commands for content management.

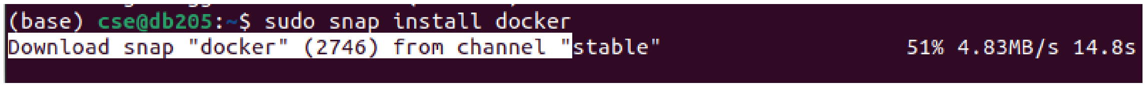
1. Update the package list:
   sudo apt update

2. Next, install Docker with the command:
   sudo apt install docker.io -y
   You'll then get a prompt asking you to choose between y/n - choose y



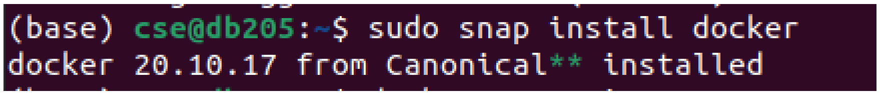
```
(base) cse@db205:~$ sudo apt install docker.io -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  aufs-tools btrfs-progs cgroupfs-mount | cgroup-lite debootstrap docker-doc
  rinse zfs-fuse | zfsutils
The following packages will be upgraded:
  docker.io
1 upgraded, 0 newly installed, 0 to remove and 105 not upgraded.
Need to get 30.3 MB of archives.
After this operation, 23.1 MB disk space will be freed.
```

3. Install all the dependency packages using the following command:
   sudo snap install docker



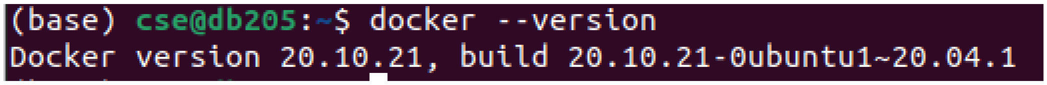
```
(base) cse@db205:~$ sudo snap install docker
Download snap "docker" (2746) from channel "stable"                51% 4.83MB/s 14.8s
```

After downloading,



```
(base) cse@db205:~$ sudo snap install docker
docker 20.10.17 from Canonical** installed
```

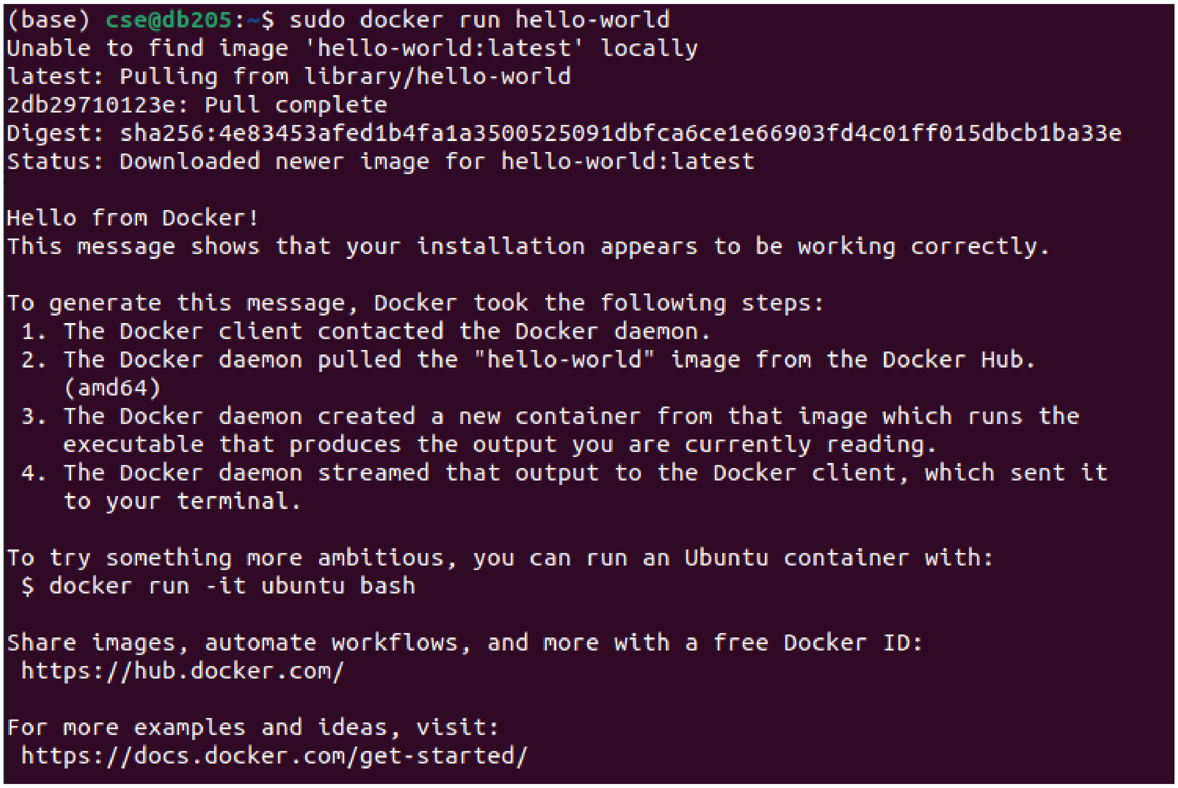
4. Before testing Docker, check the version installed using the following command:
   docker --version



```
(base) cse@db205:~$ docker --version
Docker version 20.10.21, build 20.10.21-0ubuntu1~20.04.1
```

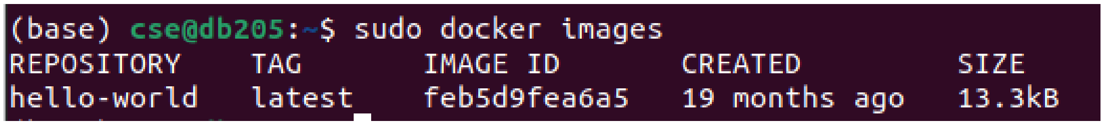
5. Pull an image from the Docker hub using the following command:
   sudo docker run hello-world

   Here, hello-world is the docker image present on the Docker hub.



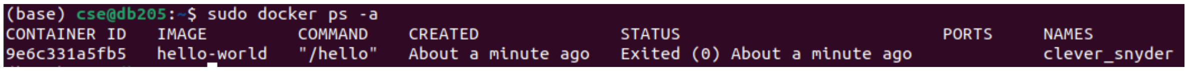
```
(base) cse@db205:~$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:4e83453afed1b4fa1a3500525091dbfca6ce1e66903fd4c01ff015dbcb1ba33e
Status: Downloaded newer image for hello-world:latest

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/
```

6. Check if the docker image has been pulled and is present in your system using the following command:
   sudo docker images



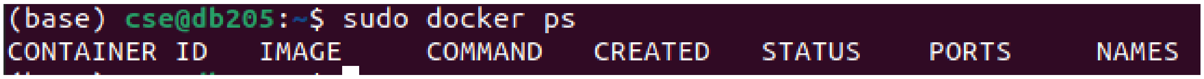
```
(base) cse@db205:~$ sudo docker images
REPOSITORY     TAG       IMAGE ID       CREATED         SIZE
hello-world    latest    feb5d9fea6a5   19 months ago   13.3kB
```

7. To display all the containers pulled, use the following command:
   sudo docker ps -a



```
(base) cse@db205:~$ sudo docker ps -a
CONTAINER ID   IMAGE         COMMAND    CREATED              STATUS                          PORTS     NAMES
9e6c331a5fb5   hello-world   "/hello"   About a minute ago   Exited (0) About a minute ago             clever_snyder
```

8. To check for containers in a running state, use the following command:
   sudo docker ps



```
(base) cse@db205:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
```

9. To pull images from the docker repository(hub.docker.com)
   docker pull <image name>

10. To access the running container

    ```
    docker exec -it <container id> bash
    ```

11. To stop a running container

    ```
    docker stop <container id>
    ```

12. This command kills the container by stopping its execution immediately. The difference between ‘docker kill’ and ‘docker stop’ is that ‘docker stop’ gives the container time to shutdown gracefully, in situations when it is taking too much time for getting the container to stop, one can opt to kill it.

    ```
    docker kill <container id>
    ```

13. Creates a new image of an edited container on the local system

    ```
    docker commit <conatainer id> <username/imagename>
    ```

14. To delete a stopped container

    ```
    docker rm <container id>
    ```

15. To delete an image from local storage

    ```
    docker rmi <image-id>
    ```