Lab 11

CS 5-1 - BSCS – Operating System Lab

Name: Sharaiz Ahmed **SAP:** 57288

Task 1:

```
C:\Users\user>docker pull busybox
Using default tag: latest
latest: Pulling from library/busybox
e59838ecfec5: Pull complete
Digest: sha256:e3652a00a2fabd16ce889f0aa32c38eec347b997e73bd09e69c962ec7f8732ee
Status: Downloaded newer image for busybox:latest
docker.io/library/busybox:latest

C:\Users\user>docker run busybox echo "Docker is working!"
Docker is working!

C:\Users\user>docker run -it busybox
/ # ls
bin    dev    etc    home   lib    lib64  proc   root   sys    tmp    usr    var
/ # pwd
/
/ # date
Sat Nov 22 15:10:59 UTC 2025
/ # exit

C:\Users\user>docker ps -a
CONTAINER ID   IMAGE           COMMAND                  CREATED          STATUS                      PORTS                    NAMES
0d6737f80e95   busybox         "sh"                     33 seconds ago   Exited (0) 8 seconds ago                             competent_tharp
833137fe2008   busybox         "echo 'Docker is wor…"   55 seconds ago   Exited (0) 52 seconds ago                            nifty_mccarthy
69009ab102d5   node-docker-app "docker-entrypoint.s…"   3 days ago       Exited (255) 3 minutes ago  0.0.0.0:3000->3000/tcp   wizardly_yalow

C:\Users\user>docker stop <833137fe2008>
The syntax of the command is incorrect.

C:\Users\user>docker stop 833137fe2008
833137fe2008

C:\Users\user>docker start 833137fe2008
833137fe2008

C:\Users\user>docker stop 833137fe2008
833137fe2008

C:\Users\user>docker rm 833137fe2008
833137fe2008

C:\Users\user>docker ps -a
CONTAINER ID   IMAGE           COMMAND                  CREATED          STATUS                      PORTS                    NAMES
0d6737f80e95   busybox         "sh"                     2 minutes ago    Exited (0) 2 minutes ago                             competent_tharp
```

TASK - 1

Running a busybox container shows docker isolates commands from the host system.

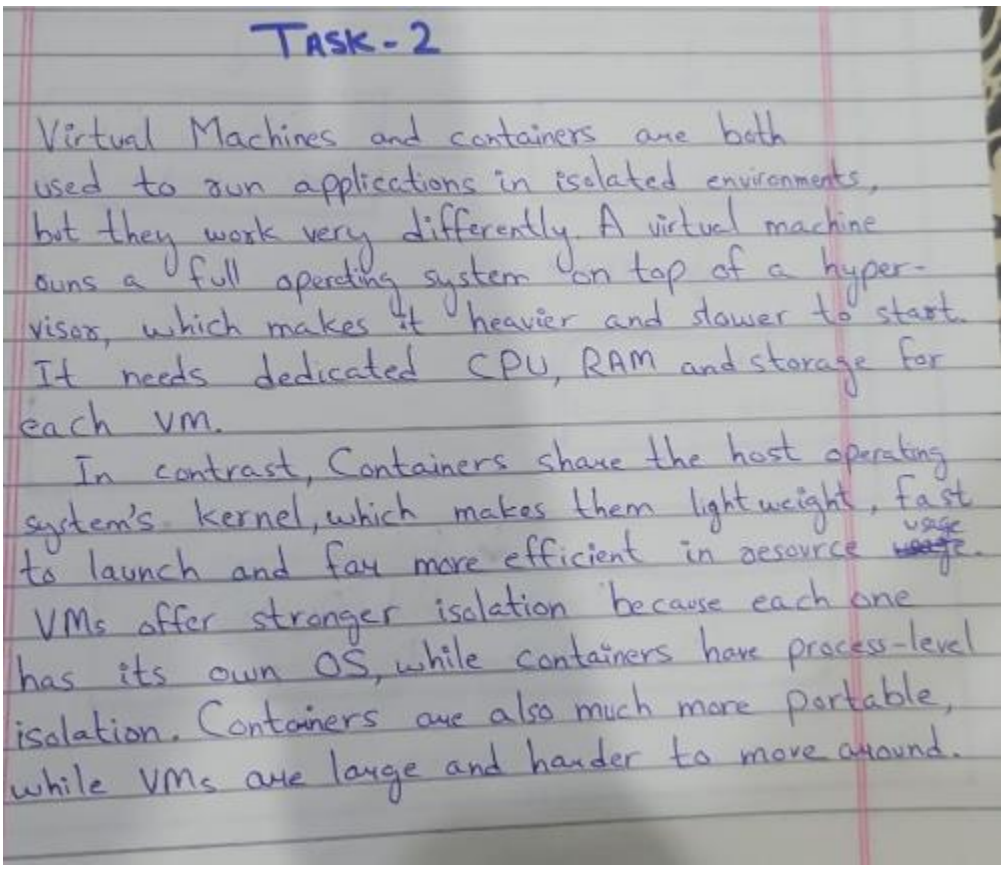
The program executes inside the container, not on windows.

Stopping halts the running process, but the container stays.

Starting resumes it instantly.

Removing deletes the container instance but the image stays.

Task 2:

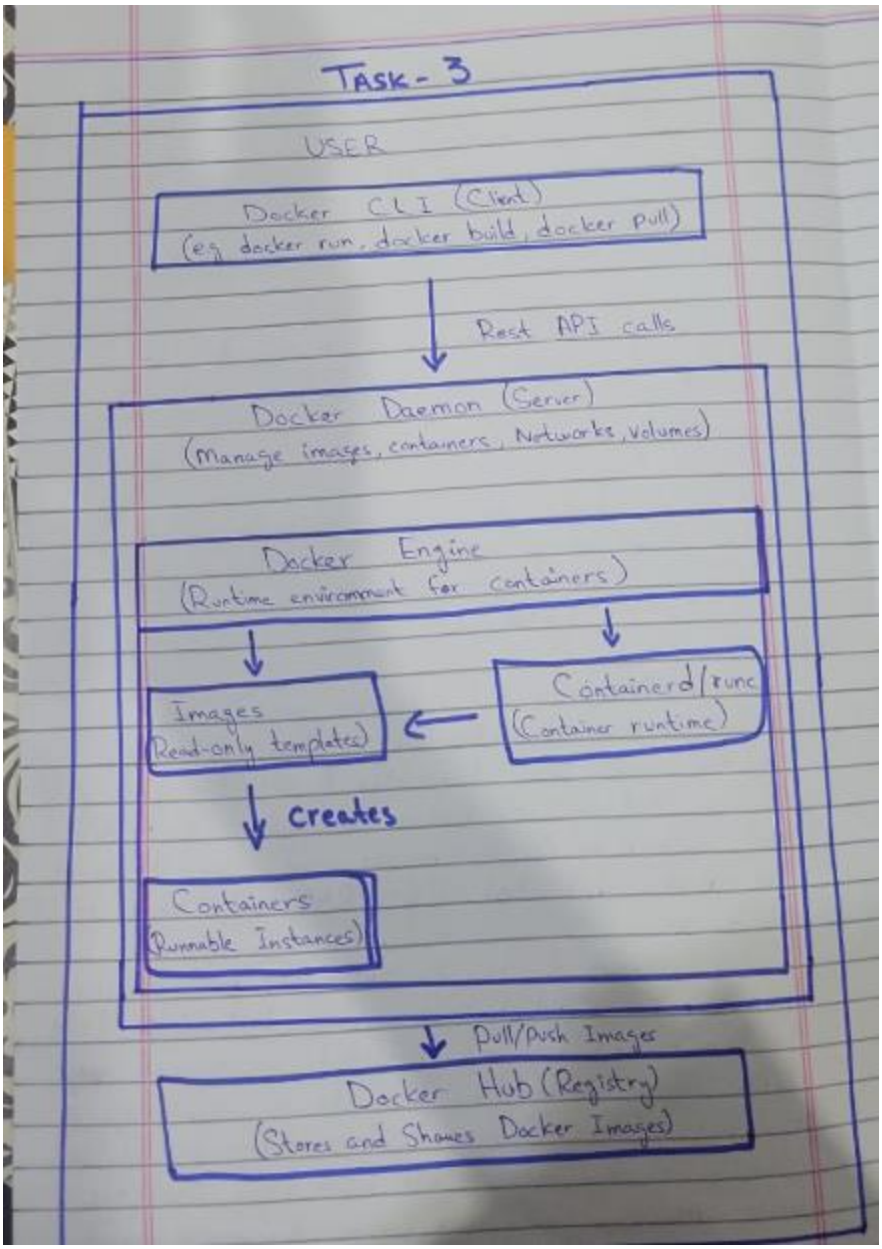


TASK - 2

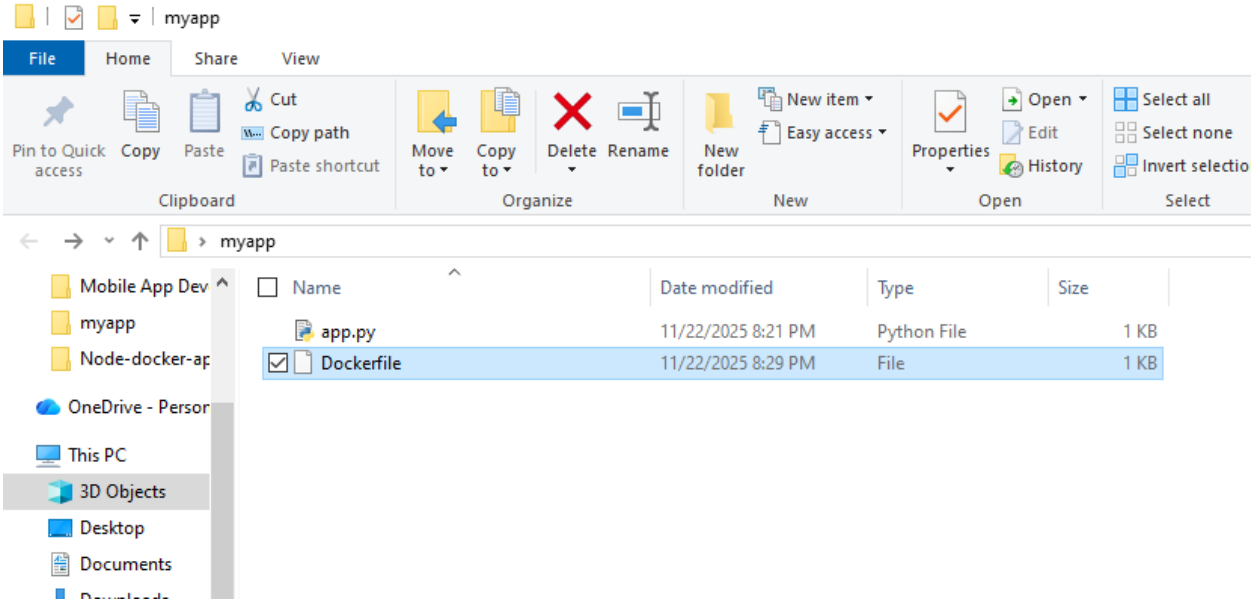
Virtual Machines and containers are both used to run applications in isolated environments, but they work very differently. A virtual machine runs a full operating system on top of a hypervisor, which makes it heavier and slower to start. It needs dedicated CPU, RAM and storage for each VM.

In contrast, Containers share the host operating system's kernel, which makes them lightweight, fast to launch and far more efficient in resource usage. VMs offer stronger isolation because each one has its own OS, while containers have process-level isolation. Containers are also much more portable, while VMs are large and harder to move around.

Task 3:


TASK - 3
USER
Docker CLI (Client)
(eg docker run, docker build, docker pull)
Rest API calls
Docker Daemon (Server)
(Manage images, containers, Networks, Volumes)
Docker Engine
(Runtime environment for containers)
Images
(Read-only templates)
Containerd/runc
(Container runtime)
creates
Containers
(Runnable Instances)
Pull/Push Images
Docker Hub (Registry)
(Stores and Shares Docker Images)

Task 4:



Dockerfile - Notepad

File Edit Format View Help

```
FROM python:3.14

WORKDIR /app

COPY app.py /app/app.py

CMD ["python", "app.py"]
```

Task 5:

```
C:\Users\user\Desktop\myapp>docker buildx build -t myapp:1.0 .
[+] Building 194.4s (9/9) FINISHED                                                                  docker:desktop-linux
 => [internal] load build definition from Dockerfile                                                                0.1s
 => => transferring dockerfile: 126B                                                                                0.0s
 => [internal] load metadata for docker.io/library/python:3.14                                                      4.7s
 => [auth] library/python:pull token for registry-1.docker.io                                                       0.0s
 => [internal] load .dockerignore                                                                                   0.1s
 => => transferring context: 2B                                                                                     0.0s
 => [1/3] FROM docker.io/library/python:3.14@sha256:6942ebef735aad5f708ef9c5e750cbe37dbc7751cee35c140e33764e34843ab9  182.4s
 => => resolve docker.io/library/python:3.14@sha256:6942ebef735aad5f708ef9c5e750cbe37dbc7751cee35c140e33764e34843ab9  0.1s
 => => sha256:50475ada0cb12c5af6c0c7c3188df0354108a5096b87a1ba0ca5a34e4e1150d3 250B / 250B                          0.8s
 => => sha256:d7d82d76b698080549e693e4b53493dc29835ae4ac836fb2aa2d6ef513e9008c 29.22MB / 29.22MB                   39.1s
 => => sha256:7c40a3faff76845154c32b7b35d5535b201d3bd04f94a0c408f8e98f9ed98ad6 235.98MB / 235.98MB                154.6s
 => => sha256:85694cfea2c4dfd425b9947a149670940f0625182653d556ed9c541e0cb1f292 6.08MB / 6.08MB                     20.4s
 => => sha256:ff2e6e687b6ce78177a4cac678dd533c8e72b97469f030783b6bb491f681fd4c 67.11MB / 67.78MB                  188.1s
 => => sha256:eae668646f447b181fe300ae6756351b6167aa2578be449b167ba79ed4926798 25.61MB / 25.61MB                   72.8s
 => => sha256:53c88f1dfeb79b2f207f7f1a03a45e0dc5ed208b9f496de16b98f81189dc0392 49.29MB / 49.29MB                   92.7s
 => => extracting sha256:53c88f1dfeb79b2f207f7f1a03a45e0dc5ed208b9f496de16b98f81189dc0392                          11.2s
 => => extracting sha256:eae668646f447b181fe300ae6756351b6167aa2578be449b167ba79ed4926798                           4.7s
 => => extracting sha256:ff2e6e687b6ce78177a4cac678dd533c8e72b97469f030783b6bb491f681fd4c                           9.6s
 => => extracting sha256:7c40a3faff76845154c32b7b35d5535b201d3bd04f94a0c408f8e98f9ed98ad6                          18.2s
 => => extracting sha256:85694cfea2c4dfd425b9947a149670940f0625182653d556ed9c541e0cb1f292                           1.9s
 => => extracting sha256:d7d82d76b698080549e693e4b53493dc29835ae4ac836fb2aa2d6ef513e9008c                           4.3s
 => => extracting sha256:50475ada0cb12c5af6c0c7c3188df0354108a5096b87a1ba0ca5a34e4e1150d3                           0.1s
 => [internal] load build context                                                                                   0.2s
 => => transferring context: 101B                                                                                   0.0s
 => [2/3] WORKDIR /app                                                                                              5.5s
 => [3/3] COPY app.py /app/app.py                                                                                   0.2s
 => exporting to image                                                                                              0.9s
 => => exporting layers                                                                                             0.4s
 => => exporting manifest sha256:8ccc2d52847156cafbefdf6fe2c4ae08806a19c92064c1e20545c76af0cac427                   0.0s
 => => exporting config sha256:1f6e12cfdd93a0f1f3dd85db6b88e1c94df4b97281e76a4eb44aedf227d87a1f                     0.0s
 => => exporting attestation manifest sha256:771b636055c7acebfa728c88bc4845f4158dde938fecaf02b51ddbc1d3e17c41       0.1s
 => => exporting manifest list sha256:ac340e61768c34ca6720cb7a74c6f0925546afa6234e0c036e9c7ec465a43c57              0.0s
 => => naming to docker.io/library/myapp:1.0                                                                        0.0s
 => => unpacking to docker.io/library/myapp:1.0                                                                     0.1s

C:\Users\user\Desktop\myapp>docker run myapp:1.0
I'm Sharaiz Ahmed, a student of BSCS in Riphah University!!
```

Task 6:

TASK - 6

1. Docker makes it easier to create, run and manage applications using containers.
2. Docker offers an alternative to traditional virtualization by using containers.
3. Containers are lightweight because they share the host operating system's kernel.
4. Docker ensures that applications run the same way in development and production.
5. A Dockerfile is a script that automates the creation of Docker image.