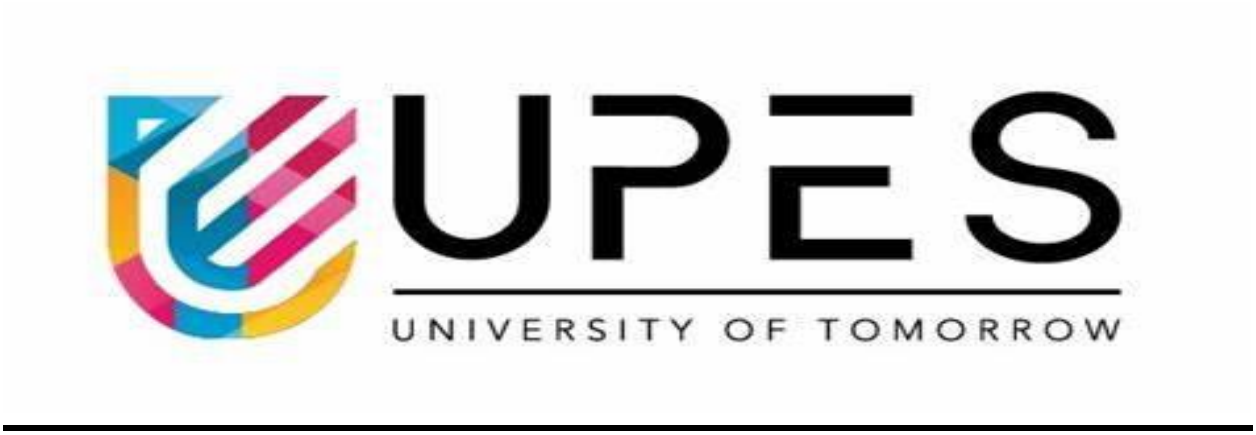
UNIVERSITY OF PETROLEUM AND ENERGY STUDIES (UPES)



Containerization and DevOps

Lab File

Submitted to:

Mr. Prateek Raj Gautam

Submitted by:

Siddharth Negi

R2142231376

SAP ID- 500121910

Experiment-2

Docker Installation, Configuration, and Running Images

Objective

- Pull Docker images
- Run containers
- Manage container lifecycle

Procedure

Step 1: Pull Image

docker pull nginx

```
siddharth@DESKTOP-TJBOHRQ:/mnt/c/WINDOWS/system32$ docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
Digest: sha256:9dd288848f4495869f76676e419ae2d767ca99fece2ec37ec0261f9fdaab5204
Status: Image is up to date for nginx:latest
docker.io/library/nginx:latest
siddharth@DESKTOP-TJBOHRQ:/mnt/c/WINDOWS/system32$
```

Step 2: Run Container with Port Mapping

docker run -d -p 8080:80 nginx

```
siddharth@DESKTOP-TJBOHRQ:/mnt/c/WINDOWS/system32$ docker run -d -p 8080:80 nginx
15e5382c3fb165b16279e2c3de5a6718ef4e4fba61a3e95346a069549957067c
siddharth@DESKTOP-TJBOHRQ:/mnt/c/WINDOWS/system32$
```

Step 3: Verify Running Containers

docker ps

```
siddharth@DESKTOP-TJBOHRQ:/mnt/c/WINDOWS/system32$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS                                     NAMES
15e5382c3fb1   nginx     "/docker-entrypoint.…"   40 seconds ago   Up 39 seconds   0.0.0.0:8080->80/tcp, [::]:8080->80/tcp   naughty_burnell
siddharth@DESKTOP-TJBOHRQ:/mnt/c/WINDOWS/system32$
```

Step 4: Stop and Remove Container

docker stop <container_id>

docker rm <container_id>

```
siddharth@DESKTOP-TJBOHRQ:/mnt/c/WINDOWS/system32$ docker stop 15e5382c3fb1
15e5382c3fb1
siddharth@DESKTOP-TJBOHRQ:/mnt/c/WINDOWS/system32$ docker rm 15e5382c3fb1
15e5382c3fb1
siddharth@DESKTOP-TJBOHRQ:/mnt/c/WINDOWS/system32$
```

Step 5: Remove Image

docker rmi -f nginx

```
siddharth@DESKTOP-TJBOHRQ:/mnt/c/WINDOWS/system32$ docker rmi -f nginx
Untagged: nginx:latest
Untagged: nginx@sha256:9dd288848f4495869f76676e419ae2d767ca99fece2ec37ec0261f9fdaab5204
Deleted: sha256:248d2326f351e7f8dc3dae8e07c24c6b69230a96ecf71ba9d6e282989b972be5
siddharth@DESKTOP-TJBOHRQ:/mnt/c/WINDOWS/system32$
```

Result

Docker images were successfully pulled, containers executed, and lifecycle commands performed.

Overall Conclusion

This lab demonstrated virtualization using **Vagrant + VirtualBox** and containerization using **Docker**, highlighting clear performance and resource efficiency differences. Containers are better suited for rapid deployment and microservices, while VMs provide stronger isolation.