Understand Docker Commands

1. docker –version
2. docker pull nginx
3. docker images
4. docker run -d -p 8080:80 nginx
5. docker ps
6. docker ps -a
7. docker rmi nginx
8. docker file Dockerfile contains instructions like:

    FROM

    RUN

    CMD

    COPY

    EXPOSE

    Example:

    FROM ubuntu
    RUN apt update
    CMD ["echo", "Hello Docker"]
9. docker build -t myapp .
10. docker run -d -p 8080:80 myapp
11. docker run --name c1 nginx
    docker run --name c2 nginx
12. docker stop c1
13. docker start c1
14. docker rm c1
15. docker logs c1
16. docker volume create myvolume
17. docker volume ls
18. docker volume inspect myvolume
19. docker volume rm myvolume
20. docker volume prune

21. docker scout quickview nginx
22. docker scout cves nginx
23. docker scout recommendations nginx

Read: Good Practices for Writing a Dockerfile

https://docs.docker.com/build/building/best-practices/

Refer https://hub.docker.com/

Lab Task

To containerize a simple Python Flask application.

1. Install Docker Desktop
2. Install VS Code
3. (Recommended) Install Docker Extension in VS Code

Step 1:

app.py

```
from flask import Flask

app = Flask(__name__)


@app.route("/")

def home():

    return "Hello from Docker Flask App"


if __name__ == "__main__":

    app.run(host="0.0.0.0", port=5000)
```

Step 2:

requirements.txt is a file that lists all the **Python dependencies (libraries)** required for your application.

Example:

requirements.txt

flask

Step 3: In VS Code:

- Right click → New File
- Name it exactly: Dockerfile
- No extension (not .txt)

Dockerfile

FROM python:3.10

WORKDIR /app

COPY . .

RUN pip install -r requirements.txt

EXPOSE 5000

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

Important Tips in creating docker file

✔ Dockerfile must be in project root
✔ Name must be exactly Dockerfile
✔ Always run build command from the folder containing Dockerfile
✔ Use .dockerignore to exclude unnecessary files-Optimizing Docker image size

Step 4:

docker build -t flaskapp .

docker run -d -p 5000:5000 --name myflask flaskapp

http://localhost:5000

Expected Output:

Hello from Docker Flask App

Step 5: Verify

docker ps

docker logs myflask

Step 6: cleanup

docker stop myflask

docker rm myflask

docker rmi flaskapp