Practical-6

Aim: Deployment of ML project in docker using Flask

Task 1: Ensure that the required libraries are installed:

- Install the Docker Command Line Interface Tool from: https://docs.docker.com/desktop/
- Install Flask library (https://flask.palletsprojects.com/en/2.3.x/installation/)
- Install gunicorn library (https://docs.gunicorn.org/en/latest/install.html)

pip install Flask

pip install gunicorn

Task 2: Create the docker file using the steps described in theory material.

Practical 6 > Dockerfile > ...

```
# Use an official Python runtime as a parent image
FROM python:3.8-slim

# Set the working directory to /app
WORKDIR /app

# Copy the current directory contents into the container at /app
COPY . /app

# Install any needed packages specified in requirements.txt
RUN pip install --no-cache-dir -r requirements.txt

# Make port 80 available to the world outside this container
EXPOSE 80

# Run app.py when the container launches
CMD ["python", "app.py"]

```

Task 3: Create the docker image using docker build command.

```
PS C:\Users\Nakul\Downloads\MLOPs\Practicals\Practical 6> docker build -t pr6 .
[+] Building 26.6s (10/10) FINISHED                                                        docker:default
 => [internal] load .dockerignore                                                                    0.0s
 => => transferring context: 2B                                                                      0.0s
 => [internal] load build definition from Dockerfile                                                 0.0s
 => => transferring dockerfile: 507B                                                                 0.0s
 => [internal] load metadata for docker.io/library/python:3.8-slim                                   2.3s
 => [auth] library/python:pull token for registry-1.docker.io                                        0.0s
 => [1/4] FROM docker.io/library/python:3.8-slim@sha256:3cb3ea0dfa00f89921c9e780618c515a7cbb5f0e0c531dc9b657cf9f155f3a66  0.0s
 => [internal] load build context                                                                    0.0s
 => => transferring context: 3.87kB                                                                  0.0s
 => CACHED [2/4] WORKDIR /app                                                                        0.0s
 => [3/4] COPY . /app                                                                                0.0s
 => [4/4] RUN pip install --no-cache-dir -r requirements.txt                                        23.2s
 => exporting to image                                                                               1.1s
 => => exporting layers                                                                              1.1s
 => => writing image sha256:229eeb5573892a648250973d55f8ec1e0c446ebf1634896caa6d2959a790dc2a         0.0s
 => => naming to docker.io/library/pr6                                                               0.0s

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview
```

```
PS C:\Users\Nakul\Downloads\MLOPs\Practicals\Practical 6> docker images
REPOSITORY                                  TAG       IMAGE ID       CREATED         SIZE
pr6                                         latest    229eeb557389   3 minutes ago   372MB
```

Task 4: Run the docker container to execute the docker image and host the machine learning model using gunicorn wsgi server.

```
PS C:\Users\Nakul\Downloads\MLOPs\Practicals\Practical 6> docker run -p 4000:80 pr6
 * Serving Flask app 'app'
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:5000
 * Running on http://172.17.0.2:5000
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 340-094-520
```

Task 5: Compare the performance of the model in docker container and flask script deployment.