Program 1: Build a Docker Container from a Custom Dockerfile
Project Structure:

Program-1/
-------> Dockerfile
-------> requirements.txt
-------> app.py

STEP 1: Create a Dockerfile and add the following content.

STEP 2: Create a Python file -app.py

```
1rv24mc031_darshan_hegde@darshu-Inspiron-15-3520:~/Desktop/LabPrograms$ mkdir Program1
1rv24mc031_darshan_hegde@darshu-Inspiron-15-3520:~/Desktop/LabPrograms$ cd Program1/
1rv24mc031_darshan_hegde@darshu-Inspiron-15-3520:~/Desktop/LabPrograms/Program1$ touch app.py
1rv24mc031_darshan_hegde@darshu-Inspiron-15-3520:~/Desktop/LabPrograms/Program1$ touch requirements.txt
1rv24mc031_darshan_hegde@darshu-Inspiron-15-3520:~/Desktop/LabPrograms/Program1$ touch Dockerfile
1rv24mc031_darshan_hegde@darshu-Inspiron-15-3520:~/Desktop/LabPrograms/Program1$ open app.py
1rv24mc031_darshan_hegde@darshu-Inspiron-15-3520:~/Desktop/LabPrograms/Program1$ cat app.py
from flask import Flask

app=Flask(__name__)

@app.route("/")
def hello():
    return "Hello, Docker!"

if __name__=="__main__":
    app.run(host="0.0.0.0",port=5000)
```

```
1rv24mc031_darshan_hegde@darshu-Inspiron-15-3520:~/Desktop/LabPrograms/Program1$ open Dockerfile
1rv24mc031_darshan_hegde@darshu-Inspiron-15-3520:~/Desktop/LabPrograms/Program1$ cat Dockerfile
### Dockerfile


# Use slim python base image
FROM python:3.9-slim

# Set the working directory
WORKDIR /app

# Copy the requirements file and install dependencies
COPY requirements.txt .
RUN pip install --no-cache-dir -r requirements.txt

#Copy the application code
COPY . .

#Set flask environment variable
ENV FLASK_APP=app.py
ENV FLASK_RUN_HOST=0.0.0.0
ENV FLASK_RUN_PORT=5000

#Expose the port the flask app will run on
EXPOSE 5000

#command to run the flask application
CMD ["flask","run"]
1rv24mc031_darshan_hegde@darshu-Inspiron-15-3520:~/Desktop/LabPrograms/Program1$ nano requirements.txt
1rv24mc031_darshan_hegde@darshu-Inspiron-15-3520:~/Desktop/LabPrograms/Program1$ cat requirements.txt
flask
```

STEP 3: Create a simple text file – requirements.txt with following content:
flask

STEP 4: Execute the Docker Build command on the Terminal to build the Docker image and then execute the Docker run command specifying the Port Number for the container.

```
1rv24mc031_darshan_hegde@darshu-Inspiron-15-3520:~/Desktop/LabPrograms/Program1$ docker build -t pro1 .
[+] Building 2.4s (10/10) FINISHED                                  docker:default
=> [internal] load build definition from Dockerfile                          0.0s
=> => transferring dockerfile: 537B                                          0.0s
=> [internal] load metadata for docker.io/library/python:3.9-slim            2.3s
=> [internal] load .dockerignore                                             0.0s
=> => transferring context: 2B                                               0.0s
=> [1/5] FROM docker.io/library/python:3.9-slim@sha256:545badebace9a958b     0.0s
=> [internal] load build context                                             0.0s
=> => transferring context: 792B                                             0.0s
=> CACHED [2/5] WORKDIR /app                                                 0.0s
=> CACHED [3/5] COPY requirements.txt .                                      0.0s
=> CACHED [4/5] RUN pip install --no-cache-dir -r requirements.txt           0.0s
=> [5/5] COPY . .                                                            0.0s
=> exporting to image                                                        0.0s
=> => exporting layers                                                       0.0s
=> => writing image sha256:a7d752e7118e9d047275717bb339fffad93ae4f992792     0.0s
=> => naming to docker.io/library/pro1                                       0.0s
```

```
1rv24mc031_darshan_hegde@darshu-Inspiron-15-3520:~/Desktop/LabPrograms/Program1$ docker run -d -p 5000:5000 --name pro1-cont pro1
a41a518cae7b43e5f421cd3ef41d50017eb95822f98b38371358107e01505f48
1rv24mc031_darshan_hegde@darshu-Inspiron-15-3520:~/Desktop/LabPrograms/Program1$ docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS                     PORTS                                       NAMES
a41a518cae7b   pro1      "flask run"              6 seconds ago   Up 6 seconds               0.0.0.0:5000->5000/tcp, [::]:5000->5000/tcp   pro1-cont
f027167c99cc   nginx     "/docker-entrypoint.…"   2 days ago      Exited (0) 2 days ago                                                  loving_vaughan
1rv24mc031_darshan_hegde@darshu-Inspiron-15-3520:~/Desktop/LabPrograms/Program1$ docker images
REPOSITORY   TAG       IMAGE ID       CREATED          SIZE
pro1         latest    a7d752e7118e   2 minutes ago    132MB
nginx        latest    657fdcd1c365   3 weeks ago      152MB
```

STEP 5: Test the Container by verifying the localhost details on web browser, the text Hello Docker! Will be displayed.