

Program 1

Step 1: Create a directory

`mkdir program1`

Step 2: requirements.txt

`nano requirements.txt`

```
1rv24mc040-haripriya@haripriya-ASUS: ~/Devops/program1
GNU nano 7.2 requirements.txt
Flask
```

Step 3: Dockerfile

`nano Dockerfile`

```
1rv24mc040-haripriya@haripriya-ASUS: ~/Devops/program1
GNU nano 7.2 Dockerfile
FROM python:3.9-slim
WORKDIR /app
COPY requirements.txt .
RUN pip install --no-cache-dir -r requirements.txt
COPY . .
ENV FLASK_APP=app.py
ENV FLASK_RUN_HOST=0.0.0.0
ENV FLASK_RUN_PORT=5000
EXPOSE 5000
CMD ["flask","run"]
```

Step 4: app.py

`nano app.py`

```
1rv24mc040-haripriya@haripriya-ASUS: ~/Devops/program1
GNU nano 7.2 app.py
from flask import Flask
app=Flask(__name__)
@app.route("/")
def hello():
    return "Hello, World!"
if __name__=="__main__":
    app.run(host="0.0.0.0",port=5000)
```

Step 5: Build the image

sudo docker build -t program-one .

```
1rv24mc040-haripriya@haripriya-ASUS:~/Devops/program1$ sudo docker build -t program-one .
[+] Building 9.6s (10/10) FINISHED                                docker:default
=> [internal] load build definition from Dockerfile              0.0s
=> => transferring dockerfile: 261B                               0.0s
=> [internal] load metadata for docker.io/library/python:3.9-slim 1.8s
=> [internal] load .dockerignore                                  0.0s
=> => transferring context: 2B                                     0.0s
=> [internal] load build context                                  0.0s
=> => transferring context: 109B                                   0.0s
=> [1/5] FROM docker.io/library/python:3.9-slim@sha256:545badebace9a958b 0.0s
=> CACHED [2/5] WORKDIR /app                                     0.0s
=> [3/5] COPY requirements.txt .                                  0.0s
=> [4/5] RUN pip install --no-cache-dir -r requirements.txt      7.6s
=> [5/5] COPY . .                                                0.0s
=> exporting to image                                             0.2s
=> => exporting layers                                           0.2s
=> => writing image sha256:bb9617fdd90b16464d9d09c9052d313f09b57be20e9da 0.0s
=> => naming to docker.io/library/program-one                    0.0s
```

Step 6: Run the container

sudo docker run -d -p 5000:5000 program-one

Also check for the running containers

docker ps

```
1rv24mc040-haripriya@haripriya-ASUS:~/Devops/program1$ sudo docker run -d -p 5000:5000 program-one
826da1d5b7c74fc6e0616fa8271912dd68e0951b0c9b8300bf47e1631f8a2d8e
1rv24mc040-haripriya@haripriya-ASUS:~/Devops/program1$ docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                               NAMES
826da1d5b7c7   program-one    "flask run"             4 minutes ago   Up 3 minutes   0.0.0.0:5000->5000/tcp, [::]:5000->5000/tcp   jolly_nobel
```

Step 7: Get the output in the browser using the URL

<http://localhost:5000/>

OR

Use CURL to get the output in the terminal

curl <http://localhost:5000/>

