

## Program 1

Step 1: Create a directory

```
mkdir program1
```

Step 2: requirements.txt

```
nano requirements.txt
```

Step 3: Dockerfile

```
nano Dockerfile
```

```
FROM python:3.9-slim
WORKDIR /app
COPY requirements.txt .
RUN pip install --no-cache-dir -r requirements.txt
COPY .
EXPOSE 5000
CMD ["python", "app.py"]
```

Step 4: app.py

```
nano 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)
```

## Step 5: Build the image

```
sudo docker build -t program1 .
```

```
Mon Nov 3 22:01

1rv24mc046_januna@januna-HP-Pavilion-Laptop-14-dv2xxx:~/Documents/Deveops/Program15 sudo docker build -t program1 .
[+] Building 20.5s (10/10) FINISHED
[+] ⚡ Internal load build definition from Dockerfile
=> => transferring dockerfile: 194B
[+] ⚡ Internal load metadata for docker.io/library/python:3.9-slim
[+] ⚡ Internal load .dockerignore
[?] => => transferring context: 2B
[+] [1/5] FROM docker.io/library/python:3.9-slim@sha256:2d97f9180bd3b3866f261f53f144965f755599aabiacade13cf1731b1b
=> =>  removing intermediate layer sha256:2d97f9180bd3b3866f261f53f144965f755599aabiacade13cf1731b1b
=> =>  copying layers sha256:77f591b165d035e05a227762f153f144965f755599aabiacade13cf1731b1b
=> =>  extracting sha256:77f591b165d035e05a227762f153f144965f755599aabiacade13cf1731b1b 10.389MB / 10.389MB, 36kB
=> =>  sha256:dd5b529e1506c35e0f4d227762f153f144965f755599aabiacade13cf1731b1b 7.48kB / 7.48kB
=> =>  sha256:8853de38a1ba449514c4f2d1b19accd73f7b8a1f575996ca8fd7 5.40kB / 5.40kB
=> =>  sha256:3e3c30b35a6c03a3e09855adeed4aa08381dfed849d9aa0758a08c7e3df3881d 1.29MB / 1.29MB
=> =>  sha256:74438849922d13b8d44b8969a953f842f59c6e9d1a0c283d710affa286c88 13.88MB / 13.88MB
=> => sha256:ea56f685404adff81680322f152d2fec62115b30dda481c2c5087315eb508 251B / 251B
=> => extracting sha256:b2ec39836aebc83a0e98540d4ceaa08381dfed849d9aa0758a08c7e3df3881d
=> => extracting sha256:fc74438084902d13b0d44b8969a953f842f59c6e9d1a0c283d710affa286c88
=> => sha256:ea56f685404adff81680322f152d2fec62115b30dda481c2c5087315eb508
[+] ⚡ Internal load build context
=> => transferring context: 454B
[2/5] WORKDIR /app
[3/5] COPY requirements.txt .
[4/5] RUN pip install --no-cache-dir -r requirements.txt
[5/5] COPY .
=> => exporting to image
=> => exporting layers
=> => creating image sha256:588be2cf7fd8245955f6d928f47ca21f49bcac37ce080d4c15cd74e7673783b6
=> => non-existing docker.io/library/python1
1rv24mc046_januna@januna-HP-Pavilion-Laptop-14-dv2xxx:~/Documents/Deveops/Program15 cat Dockerfile
FROM python:3.9-slim
WORKDIR /app
COPY requirements.txt .
RUN pip install --no-cache-dir -r requirements.txt
EXPOSE 5000
CMD ["python", "app.py"]
1rv24mc046_januna@januna-HP-Pavilion-Laptop-14-dv2xxx:~/Documents/Deveops/Program15 sudo docker run -d -p 5000:5000 --name containerflask program1
unable to find image '5000:5000' locally
docker: Error response from daemon: Get "https://registry-1.docker.io/v2/": dial tcp: lookup registry-1.docker.io on 127.0.0.53:53: read udp 127.0.0.1:54907->127.0.0.53:53: i/o timeout

Run `docker run -h` for help for more information
1rv24mc046_januna@januna-HP-Pavilion-Laptop-14-dv2xxx:~/Documents/Deveops/Program15 sudo docker run -d -p 5000:5000 --name containerflask program1
f6f52a899ddefd42c98bb579220af6d87959ae739856e0381cf7ea700d6
docker: Error response from daemon: failed to set up container networking: driver failed programming external connectivity on endpoint containerflask (60f1bf45c06c627c3d5388b47a195cbfd
```

## Step 6: Run the container

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

Also check for the running containers

docker ps

```
[root@januha ~]# 1rv24nc046_januha@januha-HP-Pavilion-Laptop-14-dvxxxx:-/Documents/Deveops/Program1 Mon Nov 3 22:02  
==> extracting sha256:fc74438849822d13be644b6969a531842f59c6e9a16a8c2cd718fa7az86c88  
==> extracting sha256:ea6f685b404adff01680322f5152d2cef62115b30ddaa81c2b450078315beb500  
== [internal] load build context  
==> transferring context: 454B  
== [2/5] WORKDIR /app  
--> RUN pip install --no-cache-dir -r requirements.txt  
== [4/5] COPY . /  
== [5/5] CMD ["python", "app.py"]  
==> exporting to image  
==> exporting layers  
==> writing image sha256:588b82cf7fd8245595f6d928f47ca21f49bcac37ce080d4c15cd74e7673783b6  
> > naming to docker.io/library/program1  
[root@januha ~]# 1rv24nc046_januha@januha-HP-Pavilion-Laptop-14-dvxxxx:-/Documents/Deveops/Program1$ cat Dockerfile  
FROM python:3.9-slim  
WORKDIR /app  
COPY requirements.txt .  
RUN pip install --no-cache-dir -r requirements.txt  
COPY . .  
EXPOSE 5000  
CMD ["python", "app.py"]  
1rv24nc046_januha@januha-HP-Pavilion-Laptop-14-dvxxxx:-/Documents/Deveops/Program1$ sudo docker run -d 5000:5000 --name containerflask program1  
unable to find image '5000:5000' locally  
docker: Error response from daemon: Get "https://registry-1.docker.io/v2/": dial tcp: lookup registry-1.docker.io on 127.0.0.53:53: read udp 127.0.0.1:54907->127.0.0.53:53: l/o timeout  
Run 'docker run --help' for more information  
1rv24nc046_januha@januha-HP-Pavilion-Laptop-14-dvxxxx:-/Documents/Deveops/Program1$ sudo docker run -d -p 5000:5000 --name containerflask program1  
e56725a989defd42c998b6e79228af6d0795a9e7398b3a1301cfef0a70d0  
docker: Error response from daemon: failed to set up container networking: driver failed programming external connectivity on endpoint containerflask (60f1fb45c0c62753d5388b4a7195cbfd  
7c4463ab0d53760778c256768cfe1): Bind for 0.0.0.0:5000 failed: port is already allocated  
  
Run 'docker run --help' for more information  
1rv24nc046_januha@januha-HP-Pavilion-Laptop-14-dvxxxx:-/Documents/Deveops/Program1$ sudo docker ps  
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES  
fc7443884982 /bin/sh -e /etc/... 19 minutes ago Up 19 minutes 0.0.0.0:5000->5000/tcp, [:]:5000->5000/tcp program1  
fc7443884982 /bin/sh -e /etc/... 19 minutes ago Up 19 minutes 0.0.0.0:80->80/tcp, [:]:80->80/tcp apache2  
a4298483b3 /mysql/mysql-server:5.6 19 minutes ago Up 19 minutes (healthy) 0.0.0.0:3306->3306/tcp, [:]:3306->3306/tcp mysql  
47e570356b49 /firstfile:latest 19 minutes ago Up 33 minutes 0.0.0.0:443->443/tcp, [:]:443->443/tcp container2  
3dbff2de6309 /firstfile:latest 19 minutes ago Up 33 minutes 0.0.0.0:8080->8080/tcp, [:]:8080->8080/tcp container1  
  
1rv24nc046_januha@januha-HP-Pavilion-Laptop-14-dvxxxx:-/Documents/Deveops/Program1$ sudo docker container stop cc18  
cc18  
1rv24nc046_januha@januha-HP-Pavilion-Laptop-14-dvxxxx:-/Documents/Deveops/Program1$ sudo docker container rm cc18  
cc18  
1rv24nc046_januha@januha-HP-Pavilion-Laptop-14-dvxxxx:-/Documents/Deveops/Program1$ sudo docker run -d -p 5000:5000 --name containerflask1 program1  
bd66883b42c30f876a316c2b23cf4a8a5de1ecff81ff2d37d  
1rv24nc046_januha@januha-HP-Pavilion-Laptop-14-dvxxxx:-/Documents/Deveops/Program1
```

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/`

