

Build a Docker container from a custom Dockerfile

Step 1 - Create the following directory structure

```
(~/Desktop/devops/lab1)
(09:57:28)→ ls
app.py  Dockerfile  requirements.txt
```

Step 2 - Modify requirements.txt

```
(~/Desktop/devops/lab1)
(09:57:52)→ cat requirements.txt
flask
```

Step 3 - Modify app.py

```
(~/Desktop/devops/lab1)
(09:58:17)→ cat 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 4 - Write the Dockerfile

```
(10:03:25) → cat Dockerfile
#Docker file to build an image for a basic flask server

#Base image
FROM python:3.9-slim

#Set working directory
WORKDIR /app

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

#Copy source code
COPY . .

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

#Expose container port
EXPOSE 5000

#Run command
CMD ["flask", "run"]
```

Step 5 - Build the image by running *sudo docker build . -t flask_image*

```
(~/Desktop/devops/lab1)
(10:03:31) → sudo docker build . -t flask_image
[sudo] password for raksha:
[+] Building 7.4s (10/10) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 472B
=> [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@sha256:545badebace9a958b98d3e272f0f0d46c0a1a389ac77e24c33f2e7b548ce1b6b
=> [internal] load build context
=> => transferring context: 720B
=> CACHED [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
=> => writing image sha256:217e3c609fab913307c1302e4887c05f53092fdc98b2547e4f725cf8c97764aa
=> => naming to docker.io/library/flask_image
```

Step 6 - Create and run a new container of the image by running `sudo docker run -p 5000:5000 flask_image`

```
(~/Desktop/devops/lab1)
(10:06:10) -> sudo docker run -p 5000:5000 flask_image
* Serving Flask app 'app.py'
* Debug mode: off
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://172.18.0.2:5000
Press CTRL+C to quit
172.18.0.1 - - [31/Oct/2025 04:36:26] "GET / HTTP/1.1" 200 -
```

Step 7 - Verify output

