

## VMware Player :

The screenshot displays a VMware Player window with a code editor and a terminal. The code editor shows a Python Flask application named 'app.py'. The terminal shows the command-line execution of the application, including the Flask startup messages and HTTP log entries. A web browser window is open at the bottom, showing the output of the application.

```
Open  [icon]  app.py  [icon] [icon] [icon] [icon] [icon]
```

```
from flask import Flask
app = Flask(__name__)
S
@app.route('/')
def hello():
    return "Hello from inside the Virtual Machine!"

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

```
(venv) haneab@haneab-VirtualBox:~$ nano app.py
(venv) haneab@haneab-VirtualBox:~$ python app.py
* Serving Flask app 'app'
* 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://10.0.2.15:5000
Press CTRL+C to quit
127.0.0.1 - - [04/May/2025 19:08:17] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [04/May/2025 19:08:18] "GET /favicon.ico HTTP/1.1" 404 -
```

127.0.0.1:5000/ x + [icon] [icon] [icon] [icon]

http://127.0.0.1:5000 [icon] [icon] [icon] [icon]

Hello from inside the Virtual Machine!

## Docker Container :

```
haneeb@haneeb-VirtualBox: ~/Desktop/flask_docker_app$ sudo docker run -d -p 5000:5000 flask-docker-app
f1e43d0c02c22a6fa46f99ea54bb3cf08393faab5d74003e07d315d5c83dcf4a
haneeb@haneeb-VirtualBox: ~/Desktop/flask_docker_app$ S
```



The screenshot shows a code editor window titled 'app.py' with the path '~ / Desktop / flask\_docker\_app'. The editor contains the following Python code for a Flask application:

```
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello():
    return "Hello from inside Docker!"

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



The screenshot shows a code editor window titled 'Dockerfile' with the path '~ / Desktop / flask\_docker\_app'. The editor contains the following Dockerfile instructions:

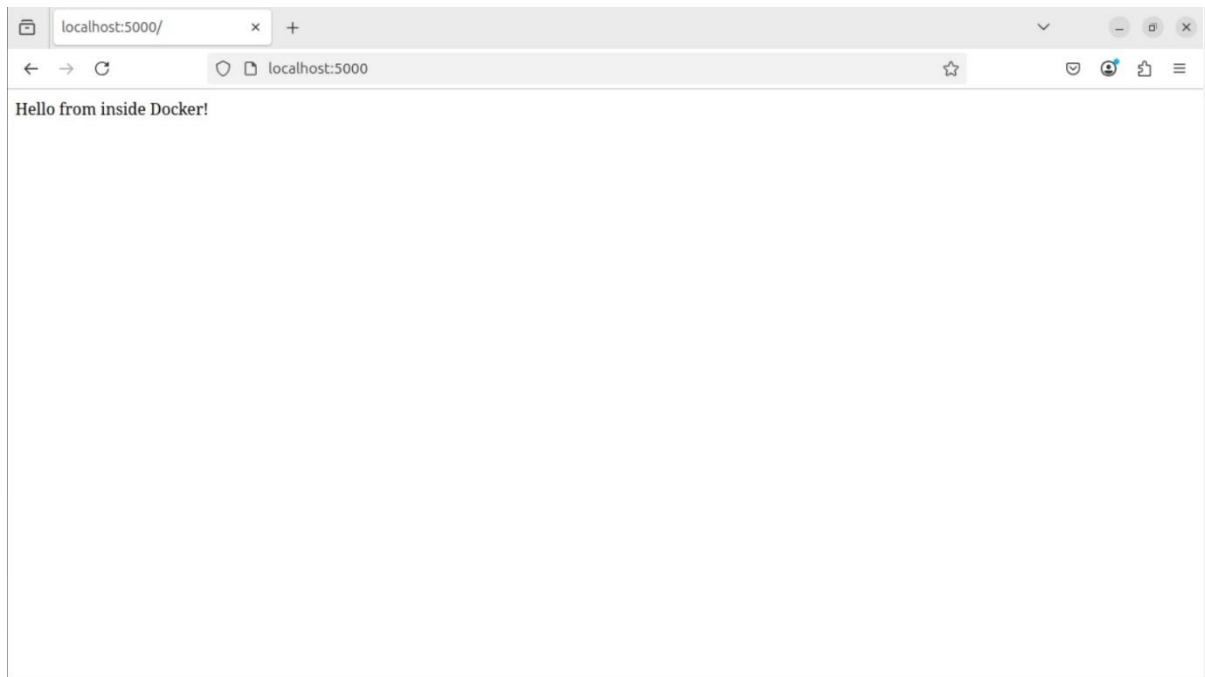
```
# Use official Python base image
FROM python:3.12-slim

# Set working directory
WORKDIR /app

# Copy app to container
COPY . .

# Install Flask
RUN pip install flask

# Run the app
CMD ["python", "app.py"]
```



## AWS EC2 :

```
haneeab@haneeab-VirtualBox:~/Desktop$ ssh -i key.pem ubuntu@13.60.235.140
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1024-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Mon May  5 20:16:59 UTC 2025

System load:  0.0                       Temperature: -273.1 C
Usage of /:   25.2% of 6.71GB           Processes:   106
Memory usage: 23%                       Users logged in: 0
Swap usage:  0%                         IPv4 address for ens5: 172.31.25.188

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.
```



```
(myenv) ubuntu@ip-172-31-25-188:~$ nano app.py
(myenv) ubuntu@ip-172-31-25-188:~$ python app.py
* Serving Flask app 'app'
* 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://172.31.25.188:5000
Press CTRL+C to quit
46.19.85.235 - - [05/May/2025 20:30:40] "GET / HTTP/1.1" 200 -
46.19.85.235 - - [05/May/2025 20:30:41] "GET /favicon.ico HTTP/1.1" 404 -
```