

Make App.py file and requirements.txt file with latest Flask version

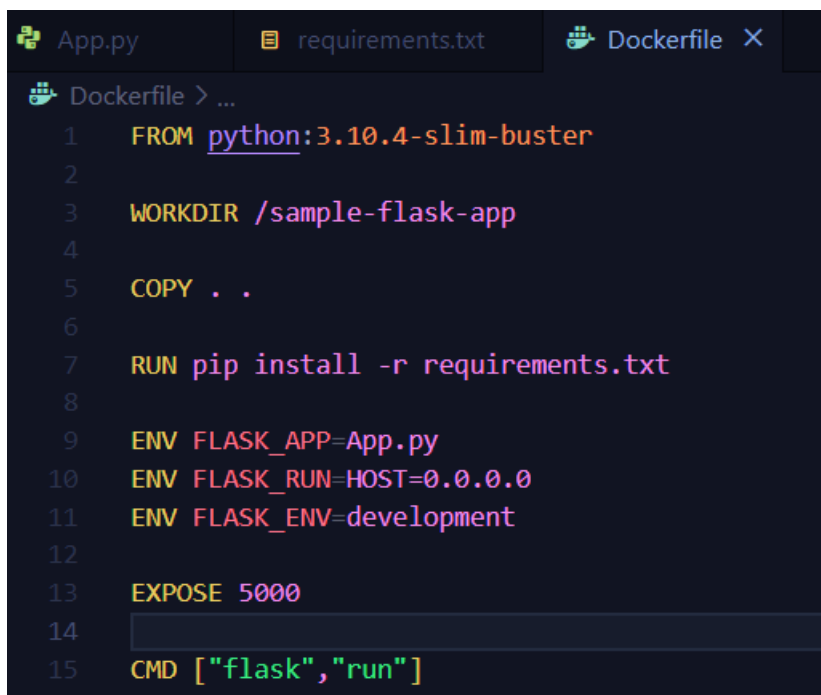


The screenshot shows a code editor with two files open: App.py and requirements.txt. The App.py file contains a Flask application that returns a JSON response "Hello World!". The requirements.txt file specifies the Flask version as 2.2.3.

```
App.py > ...
2
3  app = Flask(__name__)
4
5
6  @app.route('/')
7  def hello_world():
8      return jsonify({
9          "greeting": "Hello World!"
10     })
11
12
13 if __name__ == "__main__":
14     app.run()
15

requirements.txt > ...
requirements.txt
1  Flask==2.2.3
2
```

Create a Dockerfile in the same folder:



The screenshot shows a code editor with a Dockerfile open. The Dockerfile contains instructions to build a Docker image for the Flask application, including setting the base image, working directory, copying files, installing dependencies, setting environment variables, exposing the port, and running the application.

```
App.py requirements.txt Dockerfile > ...
Dockerfile > ...
1  FROM python:3.10.4-slim-buster
2
3  WORKDIR /sample-flask-app
4
5  COPY . .
6
7  RUN pip install -r requirements.txt
8
9  ENV FLASK_APP=App.py
10 ENV FLASK_RUN_HOST=0.0.0.0
11 ENV FLASK_ENV=development
12
13 EXPOSE 5000
14
15 CMD ["flask", "run"]
```

Create Image File for the Flask Application:

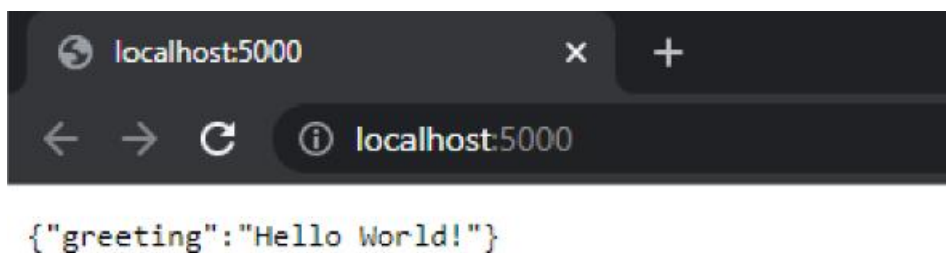
```
C:\Users\pronn\OneDrive\Desktop\SEPM Lab>docker build -t flask-sample .
[+] Building 14.5s (8/8) FINISHED
=> [internal] load build definition from Dockerfile                                0.1s
=> => transferring dockerfile: 258B                                              0.1s
=> [internal] load .dockerignore                                                 0.1s
=> => transferring context: 2B                                                  0.0s
=> [internal] load metadata for docker.io/library/python:3.10.4-slim-buster      4.2s
=> CACHED [1/4] FROM docker.io/library/python:3.10.4-slim-buster@sha256:b59fd1a008ab77d90dca9dffeab9fdcc55475ee7 0.0s
=> [internal] load build context                                                 0.1s
=> => transferring context: 993B                                              0.1s
=> [2/4] COPY . .                                                                0.5s
=> [3/4] RUN pip install -r requirements.txt                                    9.1s
=> exporting to image                                                            0.4s
=> => exporting layers                                                         0.3s
=> writing image sha256:0976eff3da548ada2790afdbd89273b355ef2329045214af9c04757aef191d3c 0.0s
=> naming to docker.io/library/flask-sample                                     0.0s
```

Check if the image has been created & Run the container:

```
C:\Users\pronn\OneDrive\Desktop\SEPM Lab>docker images
REPOSITORY          TAG          IMAGE ID          CREATED           SIZE
flask-sample        latest       98f468824df3     8 seconds ago    142MB
nginx               latest       080ed0ed8312     4 days ago       142MB
docker/getting-started latest       3e4394f6b72f     3 months ago     47MB

C:\Users\pronn\OneDrive\Desktop\SEPM Lab>docker container run -p 5000:5000 -d flask-sample
03f45fd84dc6d9fb3496d6a6fc807eac61dbd058066910bdf695c988f8dedfa1
```

Check the output:



```
localhost:5000
{"greeting": "Hello World!"}
```