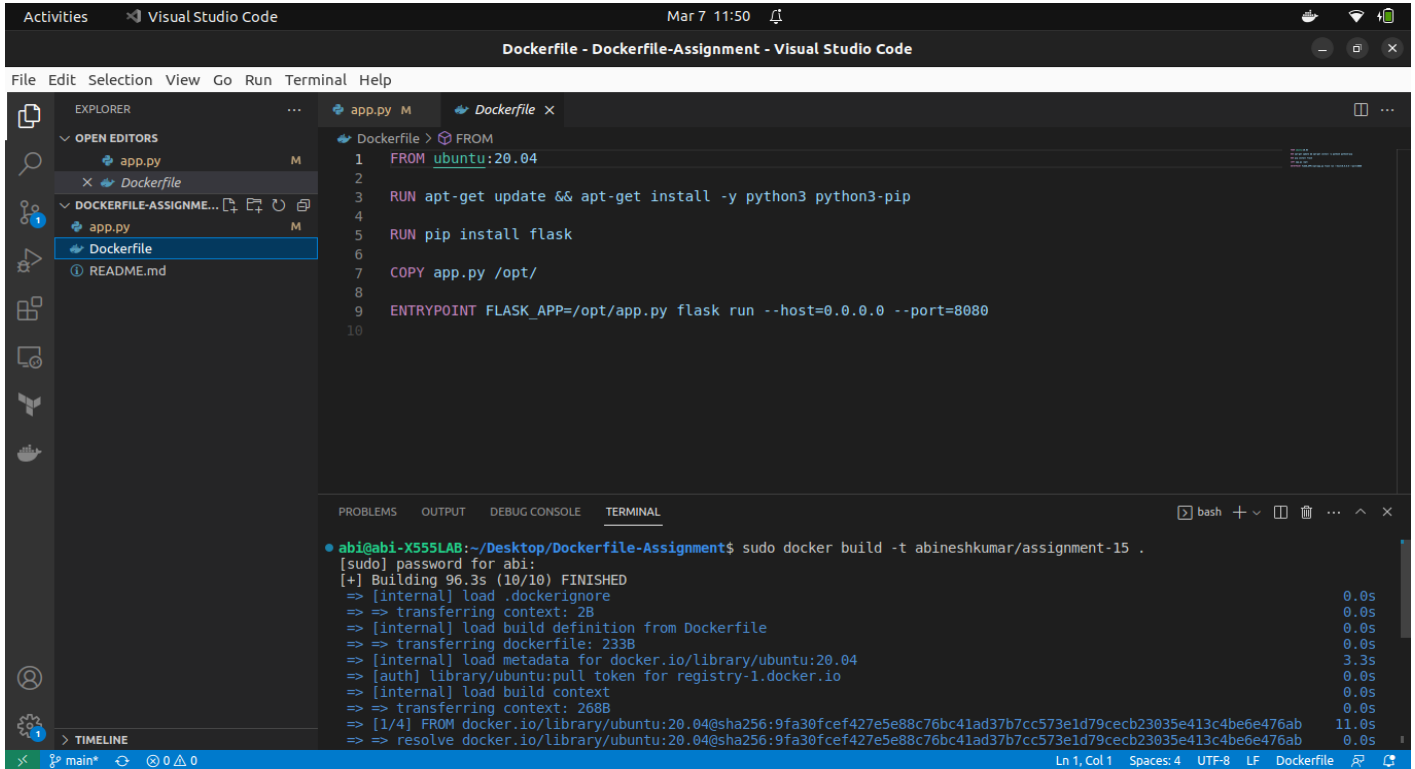


# ASSIGNMENT – 15

## DOCKERIZING THE FLASK APP

### 1. Creating a Docker image



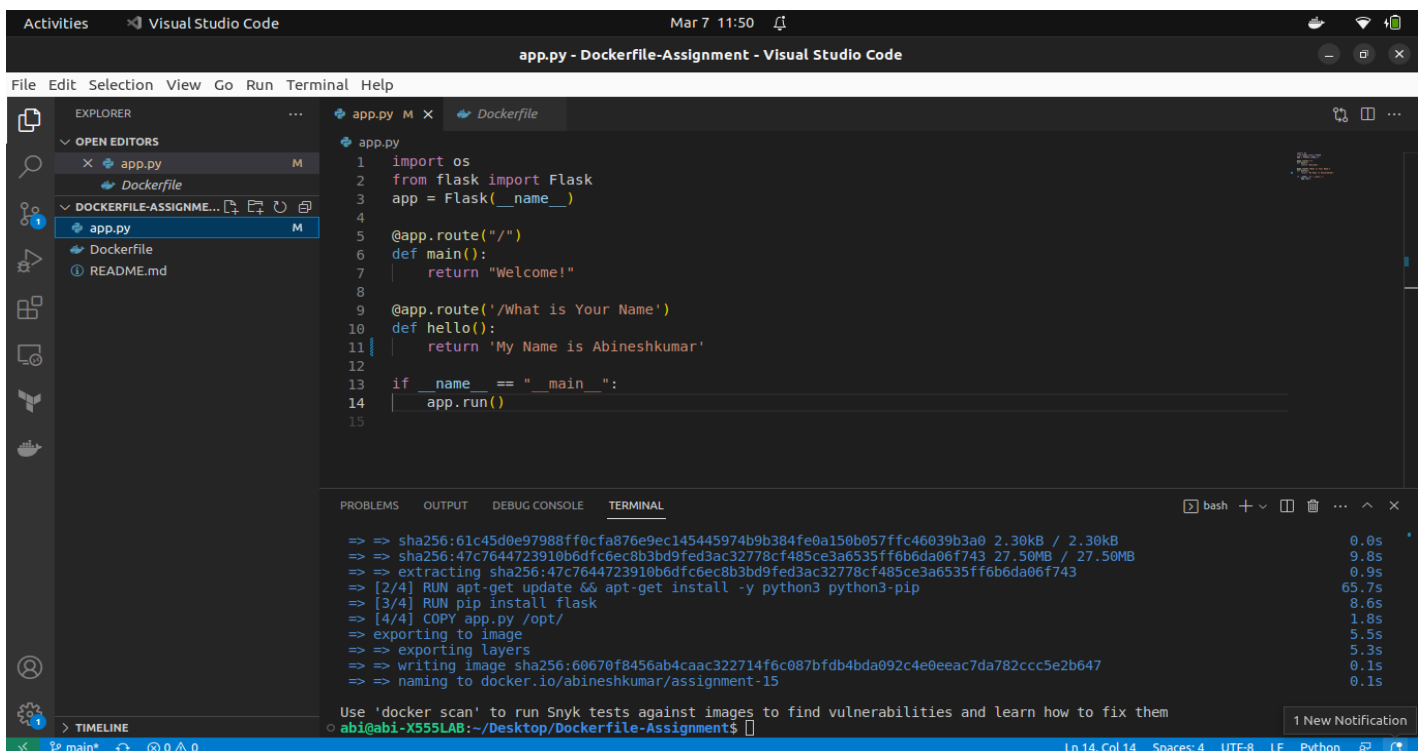
The screenshot shows the Visual Studio Code interface with the Dockerfile editor open. The Dockerfile contains the following instructions:

```
1 FROM ubuntu:20.04
2
3 RUN apt-get update && apt-get install -y python3 python3-pip
4
5 RUN pip install flask
6
7 COPY app.py /opt/
8
9 ENTRYPOINT FLASK_APP=/opt/app.py flask run --host=0.0.0.0 --port=8080
10
```

The terminal at the bottom shows the command to build the Docker image:

```
abi@abi-X555LAB:~/Desktop/Dockerfile-Assignment$ sudo docker build -t abineshkumar/assignment-15 .
[sudo] password for abi:
[+] Building 96.3s (10/10) FINISHED
=> [internal] load .dockerignore 0.0s
=> => transferring context: 2B 0.0s
=> [internal] load build definition from Dockerfile 0.0s
=> => transferring dockerfile: 233B 0.0s
=> [internal] load metadata for docker.io/library/ubuntu:20.04 3.3s
=> [auth] library/ubuntu:pull token for registry-1.docker.io 0.0s
=> [internal] load build context 0.0s
=> => transferring context: 268B 0.0s
=> [1/4] FROM docker.io/library/ubuntu:20.04@sha256:9fa30fcef427e5e88c76bc41ad37b7cc573e1d79cecb23035e413c4be6e476ab 11.0s
=> => resolve docker.io/library/ubuntu:20.04@sha256:9fa30fcef427e5e88c76bc41ad37b7cc573e1d79cecb23035e413c4be6e476ab 0.0s
```

### 2. Change was done in app.py



The screenshot shows the Visual Studio Code interface with the app.py file open. The code is as follows:

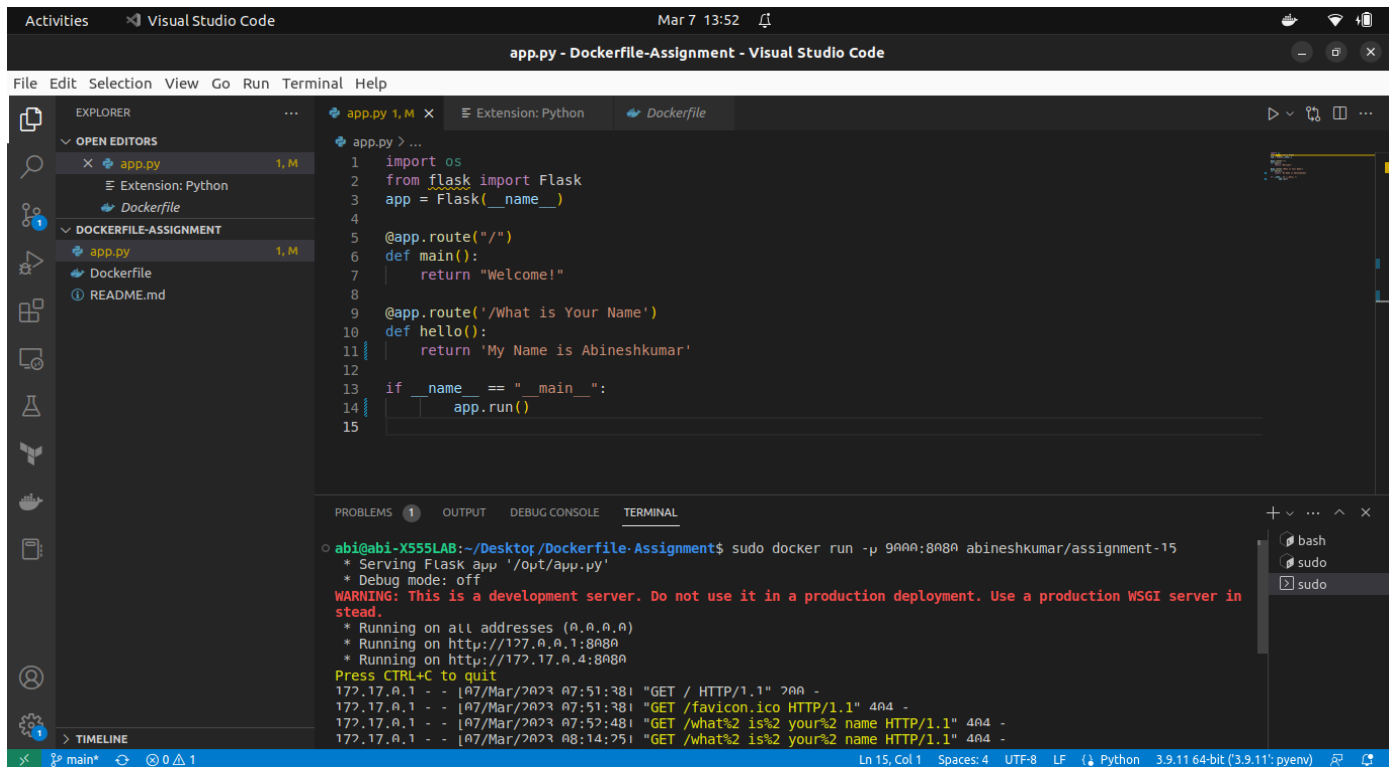
```
1 import os
2 from flask import Flask
3 app = Flask(__name__)
4
5 @app.route("/")
6 def main():
7     return "Welcome!"
8
9 @app.route('/What is Your Name')
10 def hello():
11     return 'My Name is Abineshkumar'
12
13 if __name__ == "__main__":
14     app.run()
15
```

The terminal at the bottom shows the command to build the Docker image:

```
abi@abi-X555LAB:~/Desktop/Dockerfile-Assignment$
=> sha256:61c45d0e97988ff0cfa876e9ec145445974b9b384fe0a150b057ffc46039b3a0 2.30kB / 2.30kB 0.0s
=> sha256:47c7644723910b6dfc6ec8b3bd9fed3ac32778cf485ce3a6535ff6b6da06f743 27.50MB / 27.50MB 9.8s
=> extracting sha256:47c7644723910b6dfc6ec8b3bd9fed3ac32778cf485ce3a6535ff6b6da06f743 0.9s
=> [2/4] RUN apt-get update && apt-get install -y python3 python3-pip 65.7s
=> [3/4] RUN pip install flask 8.6s
=> [4/4] COPY app.py /opt/ 1.8s
=> exporting to image 5.5s
=> exporting layers 5.3s
=> writing image sha256:60670f8456ab4caac322714f6c087bfbdb4bda092c4e0eeac7da782ccc5e2b647 0.1s
=> naming to docker.io/abineshkumar/assignment-15 0.1s
```

A notification at the bottom right says "1 New Notification".

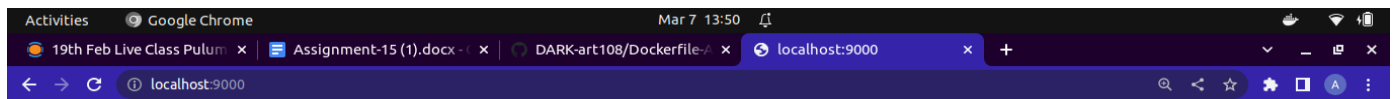
### 3. Running the app in port 9000



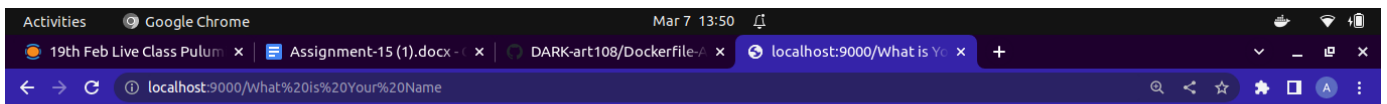
The screenshot shows the Visual Studio Code interface with a project named 'app.py - Dockerfile-Assignment'. The Explorer panel on the left shows the file structure with 'app.py' and 'Dockerfile' under 'DOCKEFILE-ASSIGNMENT'. The main editor displays the contents of 'app.py', which is a Flask application. The code defines two routes: a root route returning 'Welcome!' and a route for '/What is Your Name' returning 'My Name is Abineshkumar'. The application is run using 'app.run()'. The Terminal panel at the bottom shows the command 'sudo docker run -p 9000:8080 abineshkumar/assignment-15' being executed. The output shows the Flask application running on all addresses (0.0.0.0) and listening on port 8080. It also shows the warning message: 'WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.' The terminal output also shows the application running on http://172.17.0.1:8080 and http://172.17.0.4:8080. The status bar at the bottom indicates the file is 'main\*' and the Python version is '3.9.11 64-bit (3.9.11: pyenv)'.

```
app.py > ...
1 import os
2 from flask import Flask
3 app = Flask(__name__)
4
5 @app.route("/")
6 def main():
7     return "Welcome!"
8
9 @app.route('/What is Your Name')
10 def hello():
11     return 'My Name is Abineshkumar'
12
13 if __name__ == "__main__":
14     app.run()
15
```

```
o abi@abi-X555LAB:~/Desktop/Dockerfile-Assignment$ sudo docker run -p 9000:8080 abineshkumar/assignment-15
* Serving Flask app '/opt/app.py'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server in
stead.
* Running on all addresses (0.0.0.0)
* Running on http://172.17.0.1:8080
* Running on http://172.17.0.4:8080
Press CTRL+C to quit
172.17.0.1 - - [07/Mar/2023 07:51:38] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [07/Mar/2023 07:51:38] "GET /favicon.ico HTTP/1.1" 404 -
172.17.0.1 - - [07/Mar/2023 07:52:48] "GET /what%2 is%2 your%2 name HTTP/1.1" 404 -
172.17.0.1 - - [07/Mar/2023 08:14:25] "GET /what%2 is%2 your%2 name HTTP/1.1" 404 -
```

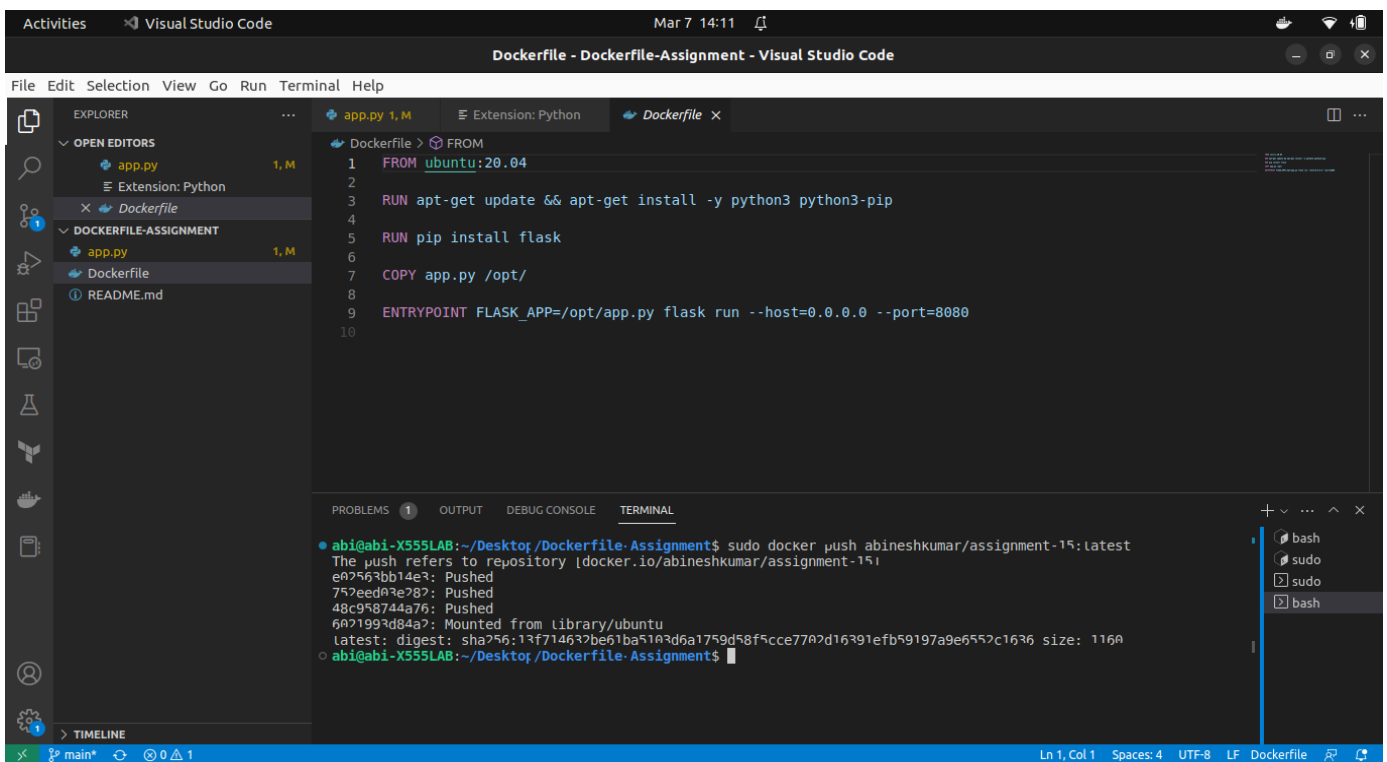


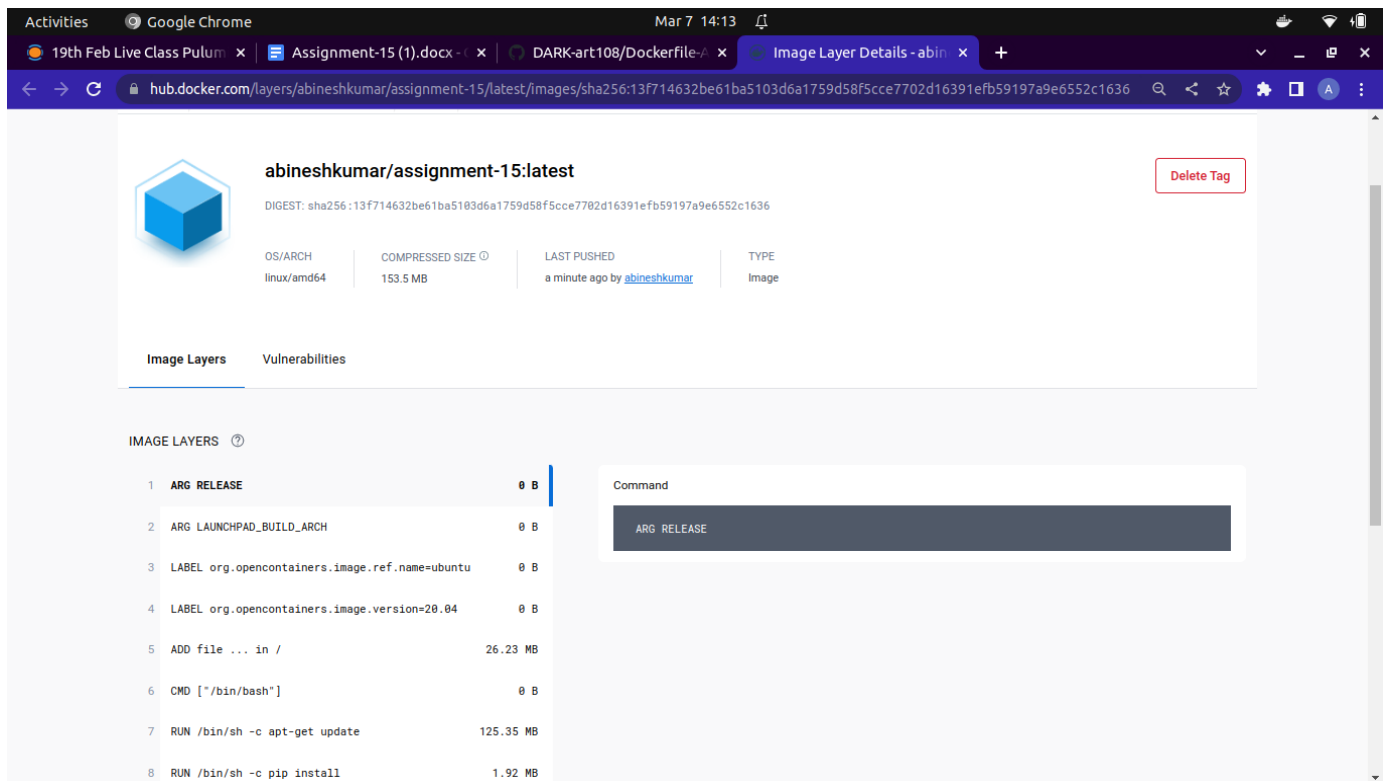
Welcome!



My Name is Abinешkumar

## 4. Pushing the image to Dockerhub





## Dockerhub link

<https://hub.docker.com/repository/docker/abineshkumar/assignment-15/general>

## 5. Dockerfile

### FROM ubuntu:20.04

➔ ubuntu is the base image of this Docker image

### RUN apt-get update && apt-get install -y python3 python3-pip

➔ This line updates the package list and installs python3 and package installer pip3

### RUN pip install flask

➔ This line installs the flask package using package installer

**COPY app.py /opt/**

➔ This line copies the app.py from the local directory to **opt** directory

**ENTRYPOINT FLASK\_APP=/opt/app.py flask run --host=0.0.0.0 --port=8080**

➔ This is command that will be executed when the container is run and this will be accessible on port 8080 on all network