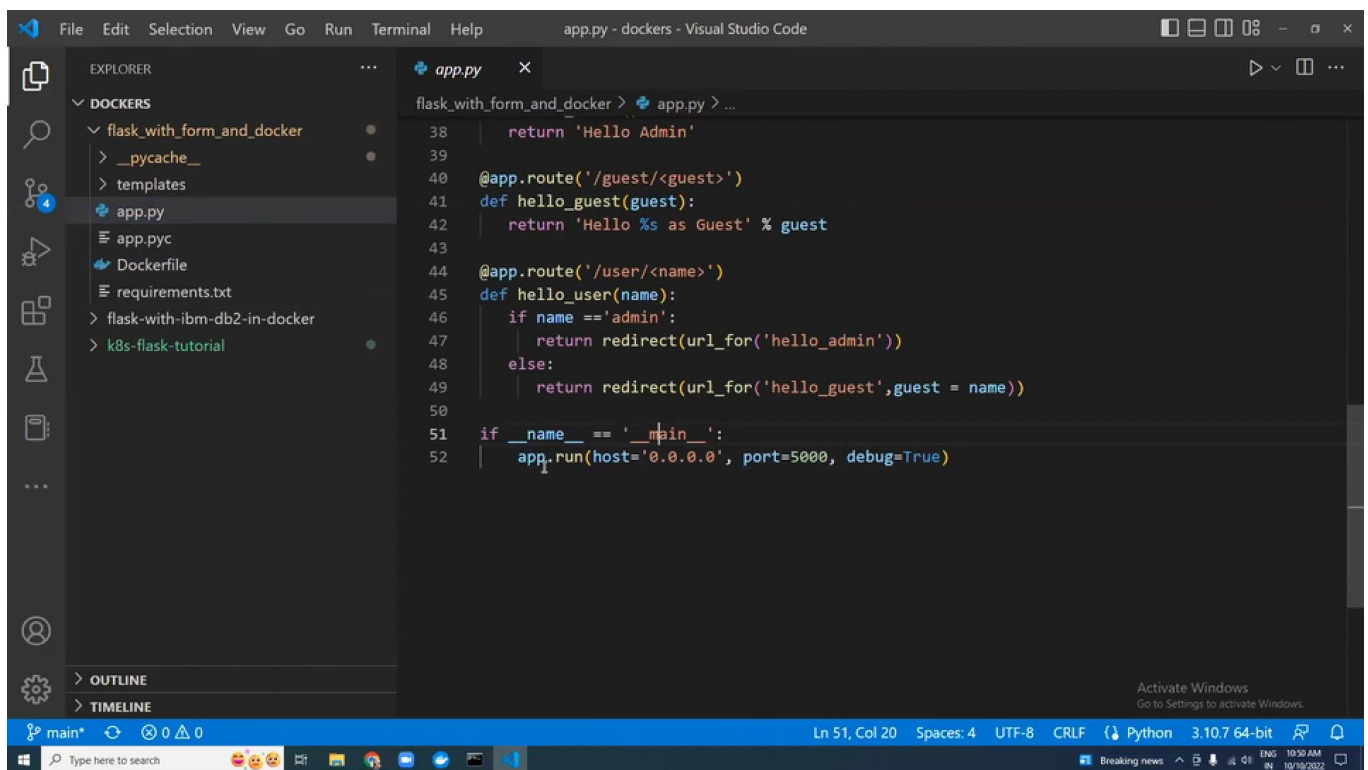


# Containerize Your Flask Application

Student Name	Sanjay S
Student Roll Number	610519104087

## Question:

### 1. Add docker file in flask app

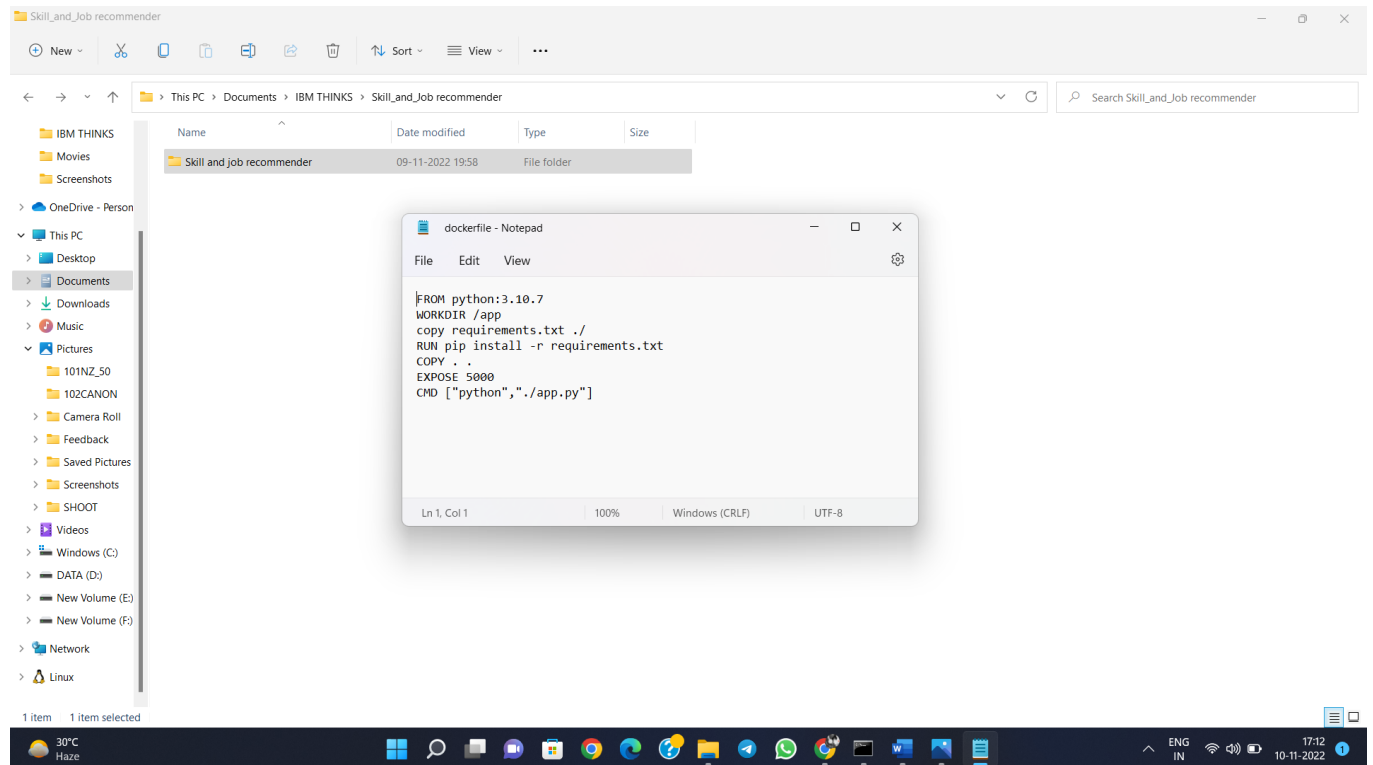


The screenshot shows the Visual Studio Code interface with a Flask application. The Explorer panel on the left shows a project structure with a 'DOCKERS' folder containing a 'Dockerfile'. The main editor displays the 'app.py' file with the following code:

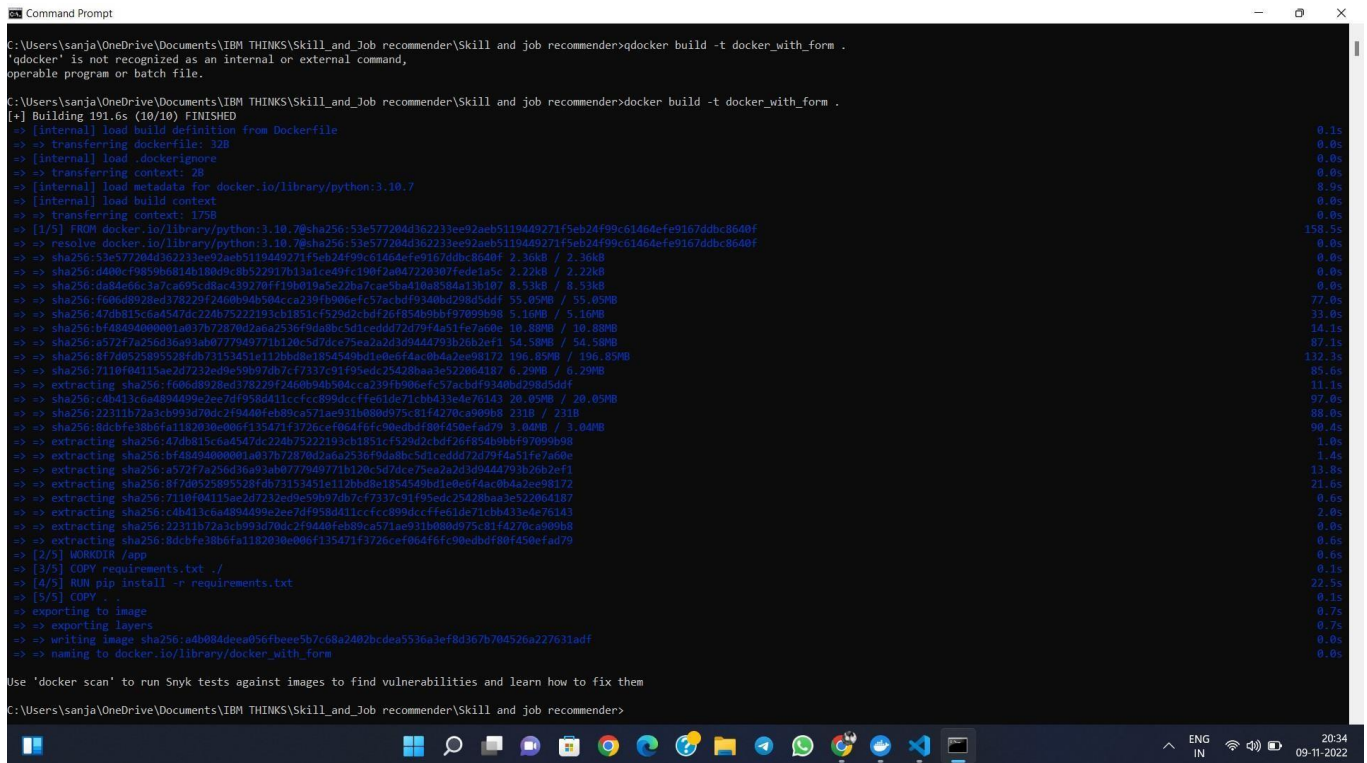
```
38     return 'Hello Admin'
39
40 @app.route('/guest/<guest>')
41 def hello_guest(guest):
42     return 'Hello %s as Guest' % guest
43
44 @app.route('/user/<name>')
45 def hello_user(name):
46     if name == 'admin':
47         return redirect(url_for('hello_admin'))
48     else:
49         return redirect(url_for('hello_guest', guest = name))
50
51 if __name__ == '__main__':
52     app.run(host='0.0.0.0', port=5000, debug=True)
```

The status bar at the bottom indicates the file is at Line 51, Column 20, using UTF-8 encoding and CRLF line endings. The Python version is 3.10.7 64-bit.

## 2. Docker File.



## 3. Build docker image using docker build command.



# Running in docker desktop

```
Command Prompt - docker run -p 5000:5000 docker_with_form

[internal] load metadata for docker.io/library/python:3.10.7
[internal] load build context
=> transferring context: 175B
[1/5] FROM docker.io/library/python:3.10.7@sha256:53e57728dd36223ee92aeb5110449271f5eb24f99c61464efe9167ddbc8640f
=> resolve docker.io/library/python:3.10.7@sha256:53e57728dd36223ee92aeb5110449271f5eb24f99c61464efe9167ddbc8640f
=> sha256:53e57728dd36223ee92aeb5110449271f5eb24f99c61464efe9167ddbc8640f 2.36kB / 2.36kB
=> sha256:a480cf9059b6814b180d9c8b522917b13a1ce49fc190f2a847220307fede1a5c 2.22kB / 2.22kB
=> sha256:a84a66c3a7cae95cd8ac439270ff190d19a5e22ba7cae5ba410a8584a13b197 8.53kB / 8.53kB
=> sha256:f686d8928ed378272460b94b584cca239fb906efc57acbf93a0bd298d5ddf 55.05MB / 55.05MB
=> sha256:47db815c6a4547dc224b7522193cb1851cf529d2cbdf26f854b0bbf97099b98 5.16MB / 5.16MB
=> sha256:b48494000001a037b72870d2a6a2536f9da8bc5d1ceddd72d79f4a51fe7a00e 10.80MB / 10.80MB
=> sha256:a572f7a256d36a93ab0777949771b120c5d7dce75ea2a2d3d9444793b26b2ef1 54.50MB / 54.50MB
=> sha256:8f780525895528f8b73153451e112b0d8e1854540bd1e0eef4acbb4a2ee98172 196.85MB / 196.85MB
=> sha256:7110f04115ae2f722ed0e9907a0747f337c81f99edc2520ba3e520641b7 6.29MB / 6.29MB
=> extracting sha256:f686d8928ed378272460b94b584cca239fb906efc57acbf93a0bd298d5ddf
=> sha256:c4b413c6a4094499e2ee7df958d411ccfcc899dcffef61de71cb433ade76143 20.05MB / 20.05MB
=> sha256:22311b72a3cb993d70dc2f9440feb89ca571ae931b080d975c81f4270ca909b8 231B / 231B
=> sha256:8dc8fe38b6fa1182030e006f135471f3726cef064f6fc90edbdff80f450efad79 3.04MB / 3.04MB
=> extracting sha256:47db815c6a4547dc224b7522193cb1851cf529d2cbdf26f854b0bbf97099b98
=> extracting sha256:b48494000001a037b72870d2a6a2536f9da8bc5d1ceddd72d79f4a51fe7a00e
=> extracting sha256:a572f7a256d36a93ab0777949771b120c5d7dce75ea2a2d3d9444793b26b2ef1
=> extracting sha256:8f780525895528f8b73153451e112b0d8e1854540bd1e0eef4acbb4a2ee98172
=> extracting sha256:7110f04115ae2f722ed0e9907a0747f337c81f99edc2520ba3e520641b7
=> extracting sha256:c4b413c6a4094499e2ee7df958d411ccfcc899dcffef61de71cb433ade76143
=> extracting sha256:22311b72a3cb993d70dc2f9440feb89ca571ae931b080d975c81f4270ca909b8
=> extracting sha256:8dc8fe38b6fa1182030e006f135471f3726cef064f6fc90edbdff80f450efad79
[2/5] WORKDIR /app
[3/5] COPY requirements.txt ./
[4/5] RUN pip install -r requirements.txt
[5/5] COPY . .
=> exporting to image
=> exporting layers
=> writing image sha256:a4b084deea05fbee5b7c68a2402bcdea5536a3ef8d367b704526a227631a0f
=> naming to docker.io/library/docker_with_form

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them

C:\Users\sanja\OneDrive\Documents\IBM THINKS\Skill_and_Job recommender\Skill and job recommender>docker run -p 5000:5000 docker_with_form
* 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://172.0.0.1:5000
* Running on http://172.17.0.2:5000
Press CTRL+C to quit
172.17.0.1 - - [09/Nov/2022 15:06:52] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [09/Nov/2022 15:06:52] "GET /favicon.ico HTTP/1.1" 200 -
```

## 3. Run and test your container.

The screenshot shows the Docker Desktop application window. On the left is a sidebar with navigation options: Containers, Images, Volumes, Dev Environments (BETA), and Extensions (BETA). The main panel displays 'Images on disk' with a table of installed images. Below the table, a 'docker\_with\_form' container is shown as running. At the bottom, a video player for 'CAD-B1-' is visible, showing a 'NOT YET RATED' status and a 'Follow' button for the user 'Nalaiya Thiran'.

NAME	TAG	IMAGE ID	CREATED	SIZE
docker/getting-started	latest	cb90f98fd791	7 months ago	28.78 MB
docker_with_form	latest	a4b084deea05	1 minute ago	932.79 MB

RAM 2.49GB CPU 0.40% Connected to Hub v4.13.1

