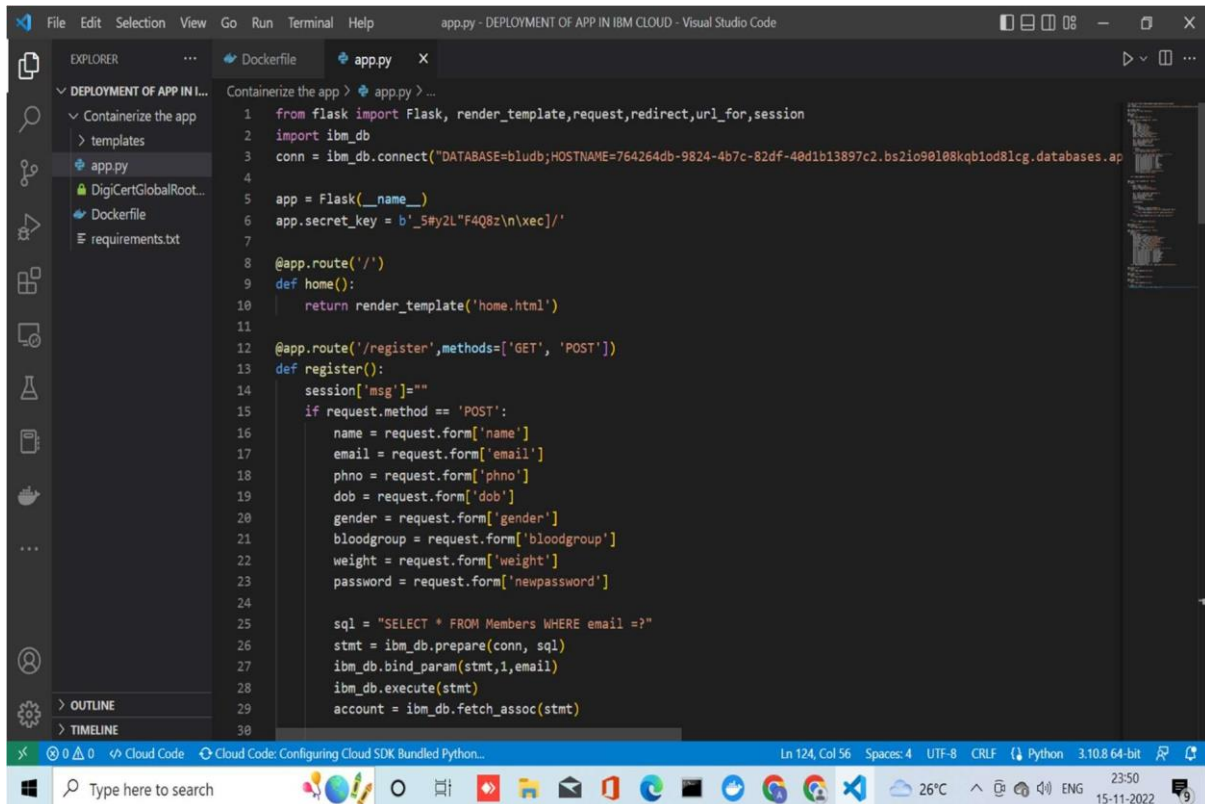


DEPLOYMENT OF APP IN IBM CLOUD

Containerize the App

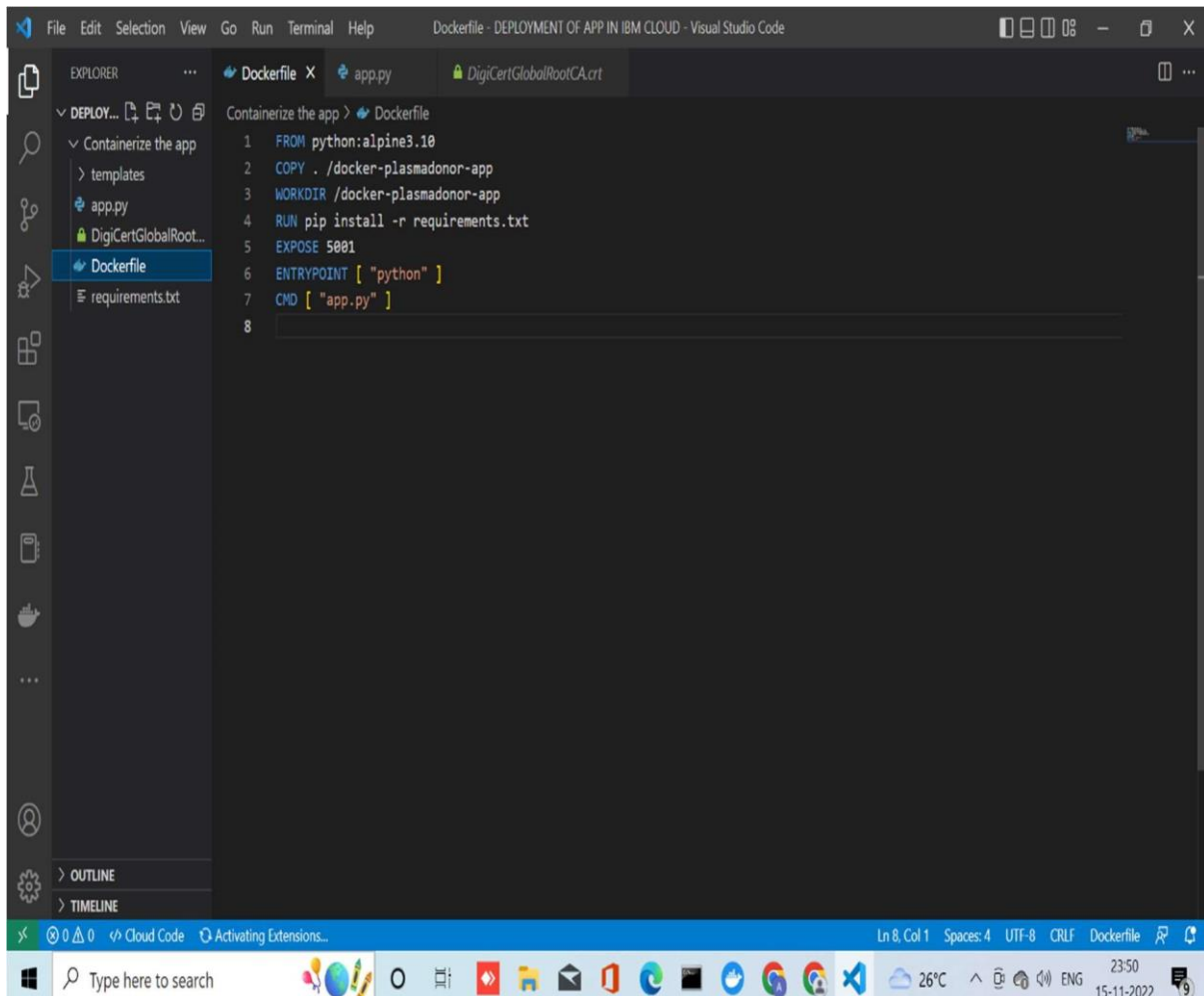
Task: Create a Dockerfile for Plasma donation app and deploy it in docker desktop

- CREATE A FLASK APP FOR PLASMA DONATION AND ADD DOCKER FILE AND NECESSARY REQUIREMENTS



This screenshot shows the Visual Studio Code editor with the 'app.py' file open. The Explorer sidebar on the left shows the project structure: 'DEPLOYMENT OF APP IN I...', 'Containerize the app', 'templates', 'app.py', 'Dockerfile', and 'requirements.txt'. The main editor area displays the Python code for the Flask application, which includes imports for Flask, IBM DB, and session management, a home route, and a register route that interacts with a database. The status bar at the bottom indicates the file is at line 124, column 56, using UTF-8 encoding and CRLF line endings.

```
1 from flask import Flask, render_template, request, redirect, url_for, session
2 import ibm_db
3 conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8l1cg.databases.ap
4
5 app = Flask(__name__)
6 app.secret_key = b'_5#y2L"F4Q8z\n\xec/'
7
8 @app.route('/')
9 def home():
10     return render_template('home.html')
11
12 @app.route('/register', methods=['GET', 'POST'])
13 def register():
14     session['msg']=""
15     if request.method == 'POST':
16         name = request.form['name']
17         email = request.form['email']
18         phno = request.form['phno']
19         dob = request.form['dob']
20         gender = request.form['gender']
21         bloodgroup = request.form['bloodgroup']
22         weight = request.form['weight']
23         password = request.form['newpassword']
24
25         sql = "SELECT * FROM Members WHERE email =?"
26         stmt = ibm_db.prepare(conn, sql)
27         ibm_db.bind_param(stmt, 1, email)
28         ibm_db.execute(stmt)
29         account = ibm_db.fetch_assoc(stmt)
30
```



This screenshot shows the Visual Studio Code editor with the 'Dockerfile' file open. The Explorer sidebar on the left shows the project structure: 'DEPLOY...', 'Containerize the app', 'templates', 'app.py', 'Dockerfile', and 'requirements.txt'. The main editor area displays the Dockerfile instructions for containerizing the application. The status bar at the bottom indicates the file is at line 8, column 1, using UTF-8 encoding and CRLF line endings.

```
1 FROM python:alpine3.10
2 COPY . /docker-plasmadonor-app
3 WORKDIR /docker-plasmadonor-app
4 RUN pip install -r requirements.txt
5 EXPOSE 5001
6 ENTRYPOINT [ "python" ]
7 CMD [ "app.py" ]
8
```

□ BUILD A DOCKER IMAGE IN DOCKER DESKTOP FOR JOB PORTAL APP USING CMD

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19044.2130]
(c) Microsoft Corporation. All rights reserved.

C:\Users\LENOVO\Desktop\PLASMA APP\PRE-DEVELOPMENT\DEPLOYMENT OF APP IN IBM CLOUD\Containerize the app>docker build -t plasma-app
"docker build" requires exactly 1 argument.
See 'docker build --help'.

Usage: docker build [OPTIONS] PATH | URL | -

Build an image from a Dockerfile

C:\Users\LENOVO\Desktop\PLASMA APP\PRE-DEVELOPMENT\DEPLOYMENT OF APP IN IBM CLOUD\Containerize the app>
C:\Users\LENOVO\Desktop\PLASMA APP\PRE-DEVELOPMENT\DEPLOYMENT OF APP IN IBM CLOUD\Containerize the app>docker build -t plasma-app .
[+] Building 31.7s (10/10) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 221B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/python:alpine3.10
=> [auth] library/python:pull token for registry-1.docker.io
=> [internal] load build context
=> => transferring context: 27.82kB
=> CACHED [1/4] FROM docker.io/library/python:alpine3.10@sha256:152b1952d4b42a360f2ef3037d9b645328c0cc0fbc63dec6ff0ff407b96a
=> [2/4] COPY . /docker-plasmadonor-app
=> [3/4] WORKDIR /docker-plasmadonor-app
=> [4/4] RUN pip install -r requirements.txt
=> exporting to image
=> => exporting layers
=> writing image sha256:b94bd65b48eaf5cae9d1a8add60fc138d895406aef6429f3b6faf0f38167f4
=> naming to docker.io/library/plasma-app

C:\Users\LENOVO\Desktop\PLASMA APP\PRE-DEVELOPMENT\DEPLOYMENT OF APP IN IBM CLOUD\Containerize the app>
```

□ PLASMA APP IMAGE IN DOCKER CONTAINER

Docker Desktop interface showing the 'Images' tab. The 'LOCAL' section lists several Docker images, with 'plasma-app' highlighted. The image has a tag of 'latest', is 'Unused', and was created 'less than a minute ago'. Other images listed include 'flask-docker-demo-app', 'docker-jobportal-app', 'shakeelsheriff/firstimage', and 'nginx'.

NAME	TAG	STATUS	CREATED	SIZE	ACTIONS
plasma-app b94bd65b48ea	latest	Unused	less than a minute ago	91.15 MB	▶ ⋮ 🗑️
flask-docker-demo-app eebe14e652de	latest	Unused	about 20 hours ago	92.21 MB	▶ ⋮ 🗑️
docker-jobportal-app 3a9262a84470	latest	Unused	2 days ago	91.13 MB	▶ ⋮ 🗑️
shakeelsheriff/firstimage 3a9262a84470	latest	Unused	2 days ago	91.13 MB	▶ ⋮ 🗑️
nginx 5d58c024174d	latest	Unused	27 days ago	141.77 MB	▶ ⋮ 🗑️

TASK COMPLETED SUCCESSFULLY!