

app.py :

```
from flask import Flask, render_template,request, redirect, url_for, session

import ibm_db

import json

import requests

app = Flask(__name__, template_folder='template')

conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=fbd88901-ebdb-4a4f-a32e-
9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32731;SECURITY
=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=cts72740;PWD=dORKSzIzhw
xBmMvg",",")

@app.route('/registration')

def home():

    return render_template('register.html')

@app.route('/register',methods=['POST'])

def register():

    x = [x for x in request.form.values()]

    print(x)

    name=x[0]

    email=x[1]

    phone=x[2]

    city=x[3]

    infect=x[4]

    blood=x[5]

    password=x[6]

    sql = "SELECT * FROM plasmadonor WHERE email =?"

    stmt = ibm_db.prepare(conn, sql)

    ibm_db.bind_param(stmt,1,email)

    ibm_db.execute(stmt)
```

```

account = ibm_db.fetch_assoc(stmt)

print(account)

if account:

    return render_template('register.html', pred="You are already a member, please login
using your details")

else:

    insert_sql = "INSERT INTO plasmadonor VALUES (?, ?, ?, ?, ?, ?, ?)"

    prep_stmt = ibm_db.prepare(conn, insert_sql)

    ibm_db.bind_param(prepare_stmt, 1, name)

    ibm_db.bind_param(prepare_stmt, 2, email)

    ibm_db.bind_param(prepare_stmt, 3, phone)

    ibm_db.bind_param(prepare_stmt, 4, city) ibm_db.bind_param(prepare_stmt, 5, infect)

    ibm_db.bind_param(prepare_stmt, 6, blood)

    ibm_db.bind_param(prepare_stmt, 7, password)

    ibm_db.execute(prepare_stmt)

    return render_template('register.html', pred="Registration Successful, please login using
your details")

@app.route('/')

@app.route('/login')

def login():

    return render_template('login.html')

@app.route('/loginpage',methods=['POST'])

def loginpage():

    user = request.form['user']

    passw = request.form['passw']

    sql = "SELECT * FROM plasmadonor WHERE email =? AND password=?"

    stmt = ibm_db.prepare(conn, sql)

```

```

    ibm_db.bind_param(stmt,1,user)

    ibm_db.bind_param(stmt,2,passw)

    ibm_db.execute(stmt)

    account = ibm_db.fetch_assoc(stmt)

    print (account)

    print(user,passw)

    if account: return redirect(url_for('stats'))

    else:

        return render_template('login.html', pred="Login unsuccessful. Incorrect username /
password !")

@app.route('/stats')
def stats():

    "sql = "SELECT blood FROM user group by blood"

    stmt = ibm_db.prepare(conn, sql)

    ibm_db.execute(stmt)

    count = ibm_db.fetch_assoc(stmt)

    print(count)"

    return render_template('stats.html',b=5,b1=2,b2=3,b3=4,b4=2,b5=1,b6=2,b7=1,b8=1)


@app.route('/requester')
def requester():

    return render_template('request.html')

@app.route('/requested',methods=['POST'])
def requested():

    bloodgrp = request.form['bloodgrp']

    address = request.form['address']

    print(address)

```

```

sql = "SELECT * FROM plasmadonor WHERE blood=?"

stmt = ibm_db.prepare(conn, sql)

ibm_db.bind_param(stmt,1,bloodgrp)

ibm_db.execute(stmt)

data = ibm_db.fetch_assoc(stmt)

msg = "Need Plasma of your blood group for: "+address

while data != False:

    print ("The Phone is : ", data["PHONE"])
url="https://www.fast2sms.com/dev/bulk?authorization=xCXuwWTzyjOD2ARd1EngbH3a7t
KIq5PklJ8YSf0Lh4FQZecs9iNI1dSvuqprxFwCKYJXA5amQkBE36Rl&sender_id=FSTSM
S&message="+msg+"&language=english&route=p&numbers="+str(data["PHONE"])

    result=requests.request("GET",url)

    print(result)

    data = ibm_db.fetch_assoc(stmt)

    return render_template('request.html', pred="Your request is sent to the concerned
people.")

if __name__ == "__main__":

    app.run(host='0.0.0.0', port=8080)

```

Output :

The screenshot shows a web browser window with the URL `127.0.0.1:8080/registration`. The browser tabs include IBM, CAD-B7-1A3E (Evenin), Docker Hub, IBM Cloud Container Regi, IBM Cloud, and Plasma Donor App. The page title is "Plasma Donor App" and there is a "Home" link in the top right corner. The registration form contains the following fields:

- Name: SIVANANDHINI R
- Email: sivanandhini2210@gmail.com
- Phone: 13568346
- City: chennai
- Blood Group: Infected
- Address: B Positive
- Register button

The Windows taskbar at the bottom shows the system clock as 21:36 on 18-11-2022, with a temperature of 25°C and various system icons.

Enter Your Name

Enter Email

Enter 10-digit mobile number

Enter Your City Name

Select COVID infection status

Choose your blood group

Enter Password

Register

Registration Successful, please login
using your details

Load Data Load History **Tables** Views Indexes Aliases MQTs Sequences Application objects

CTS72740.PLASMADONOR Back

Export to CSV

NAME	EMAIL	PHONE	CITY	INFECT	BLOOD	PASSWORD
SIVANANDHINI.R	sivanandhini2210@gmail.com	13568346	chennai	infected	B Positive	12344
viji	viji@gmail.com	1234567893	tiruchy	infected	O Positive	12345

Type here to search 31°C 13:01 18-11-2022