

ASSIGNMENT 2

ROLL NUMBER	2019503506
NAME	AMALA DEJOE NITHIN J
TEAM ID	PNT2022TMID35667

1. Create registration page in html with username, email and phone number and by using POST method display it in next html page. Develop a flask program which should contain at least 5 packages used from pypi.org.

SOURCE CODE

main.py

```
from flask import Flask, request, render_template

app = Flask(__name__)

@app.route("/", methods=["GET", "POST"])
def registerHandler():
    if request.method == "GET":
        return render_template("index.html")

    if request.method == "POST":
        data = {}

        data["name"] = request.form.get("name")
        data["email"] = request.form.get("email")
        data["phone-number"] = request.form.get("phone-number")

        return render_template("success.html", data=data)

if __name__ == '__main__':
    app.run(debug=True)
```

templates/index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Register</title>
```

```

</head>
<body>
  <div class="wrapper">
    <form method="POST" action="/">
      <div>
        <label for="name">Name</label>
        <input type="text" id="name" name="name" required>
      </div>

      <div>
        <label for="email">Email</label>
        <input type="email" id="email" name="email" required>
      </div>

      <div>
        <label for="phone-number">Phone Number</label>
        <input type="tel" id="phone-number" name="phone-
number" required>
      </div>

      <div>
        <label for="password">Password</label>
        <input type="password" id="password" name="password" required>
      </div>

      <button type="submit">Register</button>
    </form>
  </div>
</body>
</html>

```

templates/success.html

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Success</title>
</head>
<body>
  <div class="wrapper">
    <h1>Registered Successfully</h1>
    <p>Name: {{data['name']}}</p>
    <p>Email: {{data['email']}}</p>
    <p>Phone Number: {{data['phone-number']}}</p>
  </div>
</body>

```

```
</html>
```

OUTPUT

Name

Email

Phone Number

Password

**Registered
Successfully**

Name : AMALA DEJOE NITHIN J

Email : amaladejoenithin@gmail.com

Phone Number : 6383106929

- Create User table with user with email, username, roll-number, password. Perform UPDATE, DELETE Queries with user table. Connect python code to DB2. Create a flask app with registration page, login page and welcome page. By default, load the registration page once the user enters all the fields store the data in database and navigate to login page authenticate user username and password. If the user is valid show the welcome page.**

SOURCE CODE

CREATE STATEMENT

```
CREATE TABLE QDP46216.user (  
    username VARCHAR(50) NOT NULL,  
    email VARCHAR(50) NOT NULL,  
    roll_number VARCHAR(50) NOT NULL,  
    password VARCHAR(50) NOT NULL  
);
```

INSERT STATEMENT

```
INSERT INTO QDP46217.user
```

```
VALUES ('KARTHIKEYAN C', 'karthi@gmail.com',
'2019503020','Default@IBM');
INSERT INTO QDP46217.user
VALUES ('VAISHNAVANAMBI SV', 'nambi@gmail.com', '2019503058',
'Default@IBM');
INSERT INTO QDP46217.user VALUES ('AMALA DEJOE NITHIN J',
'dejoe@gmail.com', '2019503506', 'Default@IBM');
INSERT INTO QDP46217.user VALUES ('EZHIL SARAVANAN T',
'ezhil@gmail.com', '2019503516', 'Default@IBM');
```

UPDATE STATEMENT

```
UPDATE QDP46217.user
SET email = 'dejoenithin@gmail.com'
WHERE username = 'AMALA DEJOE NITHIN J';
```

DELETE STATEMENT

```
DELETE FROM QDP46217.user
WHERE email = 'dejoenithin@gmail.com';
```

CONNECTING PYTHON TO DB2

```
import ibm_db

DATABASE = "bludb"
HOSTNAME = "fbd88901-ebdb-4a4f-a32e-
9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud"
PORT = 32731
UID = "qdp46216"
PWD = "MGhHNGxutNYPFPfE"

connection = ibm_db.connect(

f"DATABASE={DATABASE};HOSTNAME={HOSTNAME};PORT={PORT};SECURITY=SSL;SSLServe
rCertificate=DigitCertGlobalRootCA.crt;UID={UID};PWD={PWD}", "", ""
)
```

LOGIN & REGISTER

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

app = Flask(__name__)
app.secret_key = 'Zenik'

DATABASE = "bludb"
```

```

HOSTNAME ="fbd88901-ebdb-4a4f-a32e-
9822b9fb237b.clogj3sd0tgtu0lqde00.databases.appdomain.cloud"
PORT =32731
UID ="qdp46216"
PWD ="MGhHNGxutNYPFPfE"

connection =ibm_db.connect(

f"DATABASE={DATABASE};HOSTNAME={HOSTNAME};PORT={PORT};SECURITY=SSL;SSLServe
rCertificate=DigitCertGlobalRootCA.crt;UID={UID};PWD={PWD}", "", ""
)

@app.route('/')
@app.route('/home')
defhome():
    returnrender_template('home.html', title='Home', msg=" ")

@app.route('/dashboard')
defdashboard():
    SQL ="SELECT * FROM USERS WHERE username = ?"
    statement =ibm_db.prepare(connection, SQL)

    ibm_db.bind_param(statement, 1, session['username'])
    ibm_db.execute(statement)

    account =ibm_db.fetch_assoc(statement)

    returnrender_template('dashboard.html', title='Dashboard',
account=account)

@app.route('/logout')
deflogout():
    session.pop('Loggedin', None)
    session.pop('id', None)
    session.pop('username', None)

    returnredirect('/')

@app.route('/login', methods=['GET', 'POST'])
deflogin():
    message =""
    globaluser_id
    ifrequest.method=="POST":
        username =request.form['username']
        password =request.form['password']

        SQL ="SELECT * FROM USERS WHERE username =? AND password =?"
        statement =ibm_db.prepare(connection, SQL)

```

```

        ibm_db.bind_param(statement, 1, username)
        ibm_db.bind_param(statement, 2, password)
        ibm_db.execute(statement)

        account = ibm_db.fetch_assoc(statement)

        if account:
            session['Loggedin'] = True
            session['id'] = account['USERNAME']
            session['username'] = account['USERNAME']

            user_id = account['USERNAME']

            return redirect('/dashboard')
        else:
            message = "Incorrect login credentials"
            return render_template('login.html', title='Login',
message=message)

@app.route('/register', methods=['GET', 'POST'])
def register():
    message = ""
    if request.method == "POST":
        username = request.form['username']
        email = request.form['email']
        password = request.form['password']
        roll_number = request.form['roll-number']

        SQL = "SELECT * FROM USERS WHERE username = ? or email = ?"
        statement = ibm_db.prepare(connection, SQL)

        ibm_db.bind_param(statement, 1, username)
        ibm_db.bind_param(statement, 2, email)

        ibm_db.execute(statement)

        account = ibm_db.fetch_assoc(statement)

        if account:
            message = "Account already exists"
        elif not re.match(r'[A-Za-z0-9]+', username):
            message = "Username should be only alphabets and numbers"
        else:
            SQL = "INSERT INTO USERS VALUES (?, ?, ?, ?)"
            statement = ibm_db.prepare(connection, SQL)

            ibm_db.bind_param(statement, 1, username)

```

```

        ibm_db.bind_param(statement, 2, email)
        ibm_db.bind_param(statement, 3, roll_number)
        ibm_db.bind_param(statement, 4, password)

        ibm_db.execute(statement)

        return redirect('/login')

    return render_template('register.html', message=message,
title="Register")

if __name__ == '__main__':
    app.run(debug=True)

```

OUTPUT

USER TABLE

Name	Data type	Nullable	Length	Scale
USERNAME	VARCHAR	N	50	0
EMAIL	VARCHAR	N	50	0
ROLL_NUMBER	VARCHAR	N	50	0
PASSWORD	VARCHAR	N	50	0

RESULT OF INSERT QUERY

USERNAME	EMAIL	ROLL_NUMBER	PASSWORD
KARTHIKEYAN C	karthi@gmail.com	2019503020	Default@IBM
VAISHNAVANAMBI SV	nambi@gmail.com	2019503058	Default@IBM
AMALA DEJOE NITHIN J	dejoe@gmail.com	2019503506	Default@IBM
EZHIL SARAVANAN T	ezhil@gmail.com	2019503516	Default@IBM

RESULT OF UPDATE QUERY

USERNAME	EMAIL	ROLL_NUMBE R	PASSWORD
KARTHIKEYAN C	karthi@gmail.com	2019503020	Default@IB M
VAISHNAVANAM BI SV	nambi@gmail.com	2019503058	Default@IB M
AMALA DEJOE NITHIN J	dejoenithin@gmail.co m	2019503506	Default@IB M
EZHIL SARAVANAN T	ezhil@gmail.com	2019503516	Default@IB M

RESULT OF DELETE QUERY

USERNAME	EMAIL	ROLL_NUMBER	PASSWORD
KARTHIKEYAN C	karthi@gmail.com	2019503020	Default@IBM
VAISHNAVANAMBI S V	nambi@gmail.com	2019503058	Default@IBM
EZHIL SARAVANAN T	ezhil@gmail.com	2019503516	Default@IBM

LOGIN & REGISTER

Name

Email

Roll Number

Password

Email

Password

Dashboard

Username : Dejoe

Email : amaladejoenithin@gmail.com

Roll Number : 2019503506