

Name	Kavinraja R.
Assignment Number	02
Roll No	CITC1905023

Question:

1. Create User table with user with email, username, roll number, password.
2. Perform UPDATE, DELETE Queries with user table
3. Connect python code to db2.
4. 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

- **login.html**

```

1  <html>
2  <head>
3      <title>Flask Intro - login page</title>
4      <meta name="viewport" content="width=device-width, initial-scale=1.0">
5      <link href="static/bootstrap.min.css" rel="stylesheet" media="screen">
6  </head>
7  <body>
8      <div class="container">
9          <h1>Please login</h1>
10         <br>
11         <form action="" method="post">
12             <input type="text" placeholder="Username" name="username" value="{{
13                 request.form.username }}">
14             <input type="password" placeholder="Password" name="password" value="{{
15                 request.form.password }}">
16             <input class="btn btn-default" type="submit" value="Login">
17         </form>
18         {% if error %}
19         <p class="error"><strong>Error:</strong> {{ error }}
20         {% endif %}
21     </div>
22 </body>
23 </html>

```

- **Index.html**

```

1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Document</title>
8  </head>
9  <body style="margin:0;">
10     <div style="height: 100vh; width: 100vw; display:flex; justify-content:center; align-items: center; flex-direction: column;">
11         <h1>Registration</h1>
12         <form action="", method = "post" style="display: flex; flex-direction:column">
13             <input style="margin: 0.5rem 0" name="name" placeholder="Name">
14             <input style="margin: 0.5rem 0" type="email" name="email" placeholder="Email">
15             <input style="margin: 0.5rem 0" name="r_no" placeholder="Roll Number">
16             <input style="margin: 0.5rem 0" type="password" placeholder="Password" name="password" >
17             <button style="margin: 0.5rem 0" type="submit">Submit</button>
18         </form>
19         {% if error %}
20         <p class="error"><strong>Error:</strong> {{ error }}
21         {% endif %}
22     </div>
23 </body>
24 </html>

```

- welcome.html

```

1  <!DOCTYPE html>
2  <html>
3  <body>
4
5  <h1>WELCOME WELCOME WELCOME WELCOME </h1>
6  {% if error %}
7  <p class="error"><strong>AUTHENTICATED:</strong> {{ error }}
8  {% endif %}
9
10 </body>
11 </html>
12

```

- MAIN.py

```

File Edit Format Run Options Window Help
from flask import Flask, render_template, redirect, url_for, request
from flask_db2 import DB2

app = Flask(__name__)
app.config['DB2_DATABASE'] = 'sample'
app.config['DB2_HOSTNAME'] = 'localhost'
app.config['DB2_PORT'] = 25000
app.config['DB2_PROTOCOL'] = 'TCPIP'
app.config['DB2_USER'] = 'db2admin'
app.config['DB2_PASSWORD'] = 'kavinraja'

db = DB2(app)

```

```

@app.route('/', methods=['GET', 'POST'])
def register():
    error=None
    check=False
    cheins=False
    cheex=False
    if request.method == 'POST':
        error=None
        check=False
        cheins=False
        cheex=False
        uname=request.form['name']
        passw=request.form['password']
        r_no=request.form['r_no']
        email=request.form['email']
        print(uname+' '+passw+' '+r_no+' '+email)
        if (len(uname)>0 and len(passw)>0 and len(r_no)>0 and len(email)>0):
            check=True
            cur = db.connection.cursor()
            cur.execute('select * from user where username=\''+uname+'\' or email=\''+email+'\' or roll_number=\''+r_no+'\'')
            s=cur.fetchall()
            print(s)
            print(' cdddddddddddddddddddd')
            if(len(s)==0):
                cheex=True
                cheins=cur.execute('INSERT INTO user(email, username, roll_number, password) VALUES (\''+email+'\' , \''+uname+
            if check and cheins and cheex:
                error=None
                return redirect(url_for('login'))
            else:
                if not check:
                    error = 'ENTER ALL DATA.'
                elif not cheex:
                    error = 'DATA ALREADY exist.'
                else:
                    error = 'TRY AFTER SOME TIME'
            return render_template('index.html', error=error)

```

```

@app.route('/login', methods=['GET', 'POST'])
def login():
    error = None
    print("fff")
    if request.method == 'POST':
        cur = db.connection.cursor()
        cur.execute('select * from user;');
        #print(cur.execute('INSERT INTO user(email, username, roll_number, password) VALUES (\''kavin@gmail.com\' , \''kavin\' ,
        s=cur.fetchall()

        cur.close()

        username=request.form['username']
        password=request.form['password']
        print(s)
        check=False
        for st in s:

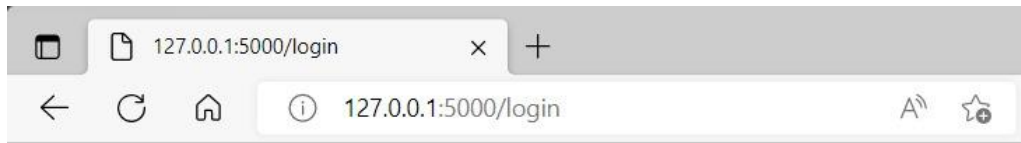
            if(len(st)>3 and st[0]==username and st[3]==password):
                # 0 email 1 username 2 roll no 3 password 4 index
                check=True
                print('done')
                error='WELCOME '+st[1]+' you are LOGINED'
                break

        if check:
            error=None
            return render_template('welcome.html',error=error)
        else:
            error = 'Invalid Credentials. Please try again.'

    return render_template('login.html', error=error)
if __name__ == "__main__":
    app.run()

```

OUTPUT:



**WELCOME WELCOME WELCOME
WELCOME**

Error: WELCOME kavin you are LOGINED

Please login

Error: Invalid Credentials. Please try again.

Please login

127.0.0.1:5000

Registration

kavin

kavin123@gmail.com

1905023

••••••••

Submit