

## Assignment -2

Assignment Date	2-11-2022
Student Name	Maheshwaransriram M
Student Roll Number	711719104050
Team ID	PNT2022TMID31573

### Question-1:

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.

### Solution:

#### App.py

```
from flask import Flask, render_template, request, redirect
import sqlite3 as sql
import models as dbHandler

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

@app.route('/')
def home():
    return render_template('home.html')

@app.route('/adduser')
def new_user():
    return render_template('add_user.html')

@app.route('/addrec', methods=['POST', 'GET'])
def addrec():
    if request.method == 'POST':
        try:
            email = request.form['email']
            un = request.form['username']
            rn = request.form['rollnumber']
            pin = request.form['pin']
            with sql.connect("User_database.db") as con:
                cur = con.cursor()
                cur.execute("INSERT INTO users(email, username, rollnumber, pin) VALUES(?, ?, ?, ?)", (email, un, rn, pin))
```

```

        con.commit()
        msg = "Record successfully added!"
    except:
        con.rollback()
        msg = "error in insert operation"
    finally:
        return render_template("list.html", msg=msg)
    con.close()

@app.route('/list')
def list():
    con = sql.connect("User_database.db")
    con.row_factory = sql.Row
    cur = con.cursor()
    cur.execute("select * from users")
    users = cur.fetchall()
    return render_template("list.html", users=users)

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

@app.route("/ delete")
def delete():
    return render_template("delete.html")

@app.route('/deleterecord', methods=["POST"])
def deleterecord():
    un = request.form['username']
    with sql.connect("User_database.db") as con:
        try:
            cur = con.cursor()
            cur.execute("DELETE FROM users WHERE username = ?", [un])
            con.commit()
            msg = "Record successfully deleted"
        except:
            msg = "can't be deleted"
        finally:
            return render_template("home1.html", msg=msg)

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

@app.route('/delddb', methods=["POST"])
def delddb():
    con = sql.connect('User_database.db')
    cur = con.cursor()
    cur.execute('DELETE FROM users')
    con.commit()
    con.close()

```

```

        msg = 'All the data has been deleted'
        return render_template("home1.html", msg=msg)
@app.route("/log")
def log():
    return render_template("login.html")
@app.route('/login', methods=['GET', 'POST'])
def login():
    un = request.form['username']
    if request.method == 'POST':
        users = dbHandler.retrieveUsers()
        msg = 'Logged in successfully!'
        return render_template('welcome.html', users=un, msg=msg)
    else:
        msg = 'You are not registered, would you like to be registered'
        return render_template('home1.html', msg=msg)
if __name__ == '__main__':
    app.run(debug=False, host='0.0.0.0')

```

## **Models.py**

```

import sqlite3 as sql
def retrieveUsers():
    con = sql.connect("User_database.db")
    cur = con.cursor()
    cur.execute("SELECT username, pin FROM users")
    users = cur.fetchone()
    con.close()
    return users

```

## **sqlite db setup.py**

```

import sqlite3
conn = sqlite3.connect('User_database.db')
print("Opened database successfully")
conn.execute('CREATE TABLE users (email TEXT, username TEXT, rollnumber INTEGER, pin INTEGER)')
print("Table created successfully")
conn.close()

```

## **Home.html**

```

<h1>Welcome to User DB APP</h1><br><br>
<a href="/">HOME</a><br><br>
<a href="/adduser">User Registration</a><br><br>
<a href="/list">List User</a><br><br>
<a href="/log">Log in</a><br><br>
<a href="/delete">Remove a User</a>

```

## Add\_user.html

```
<form action="{{ url_for('addrec') }}" method="POST">
  <h3>User
    Information</h3> Email< br>
    <input type="email" name="email" /></br>
    Username<br>
    <input type="text" name="username" /></br>
    Rollnumber<br>
    <input type="text" name="rollnumber" /><br>
    PIN<br>
    <input type="password" name="pin" min="4" max="8" /><br><br>
    <input type="submit" value="submit" />
  <p> </p>
  <input type="reset" />
</form>
```

## List.html

```
<!doctype html>
<html>

<body>
  <a href="/">HOME</a><br><br>
  <a href="/adduser">Add New Student</a><br><br>
  <a href="/list">List Student</a><br><br>
  <br>
  <hr>
  {{ msg }}
  <table border=1>
    <thead>
      <td> Email </td>
      <td> Username </td>
      <td> Roll Number </td>
      <td> Pin </td>
    </thead>
    {% for row in users %}
    <tr>
      <td>{{row["email"]}}</td>
      <td>{{row["username"]}}</td>
      <td> {{ row["rollnumber"]}}</td>
      <td>{{row['pin']}}</td>
    </tr>
    {% endfor %}
  </table>
</body>

</html>
```

