

File Edit Selection View Go Run Terminal Help

D:\Projects\IBM\IBM-Project-1050-1658337039\Assignment 2\Team Leader-Ruban.M\app.py

app.py x

app.py > ...

1 from flask import Flask,render\_template, request,jsonify

2 import sqlite3 as sql

3 app = Flask(\_\_name\_\_)

4

5 @app.route('/')

6 def index():

7 return render\_template('login.html')

8

9 @app.route('/signup',methods = ['POST', 'GET'])

10 def signup():


PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER COMMENTS


pwsh + - ^ x

PS D:\Projects\IBM\IBM-Project-1050-1658337039\Assignment 2\Team Leader-Ruban.M> pip install flask


Ln 1, Col 1 Spaces: 3 UTF-8 CRLF Python 3.10.7 64-bit Go Live Prettier

main tabnine starter





9:44 AM  
09-11-2022



File Edit Selection View Go Run Terminal Help D:\Projects\IBM\IBM-Project-1050-1658337039\Assignment 2\Team Leader-Ruban.M\app.py

EXPLO... app.py x

static  
css  
styles.css  
images  
15276013\_5570863.jpg  
templates  
home.html  
login.html  
signup.html  
app.py  
README.md  
ruban\_output.pdf  
sqlite\_db\_setup.py  
user\_database.db

```
1 from flask import Flask,render_template, request,jsonify
2 import sqlite3 as sql
3 app = Flask(__name__)
4
5 @app.route('/')
6 def index():
7     return render_template('login.html')
8
9 @app.route('/signup',methods = ['POST', 'GET'])
10 def signup():
11     return render_template('signup.html')
12
13 @app.route('/login',methods = ['POST', 'GET'])
14 def login():
15     if request.method == 'POST':
16         email = request.form['email']
17         password = request.form['password']
18
19         con=sql.connect("user_database.db")
20         cur = con.cursor()
21         statement=f"SELECT * FROM users WHERE email = '{email}' AND passw
22         cur.execute(statement)
```

Ln 6, Col 13 Spaces: 3 UTF-8 CRLF Python 3.10.7 64-bit Go Live Prettier



Search



9:47 AM  
09-11-2022

```

from flask import Flask,render_template, request,jsonify
import sqlite3 as sql
app = Flask(__name__)

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

@app.route('/signup',methods = ['POST', 'GET'])
def signup():
    return render_template('signup.html')

@app.route('/login',methods = ['POST', 'GET'])
def login():
    if request.method == 'POST':
        email = request.form['email']
        password = request.form['password']

        con=sql.connect("user_database.db")
        cur = con.cursor()
        statement=f"SELECT * FROM users WHERE email = '{email}' AND password = '{password}';"
        cur.execute(statement)
        if not cur.fetchone():
            msg = "invalid email or password",
        else:
            msg = "login successfully"
        return render_template('home.html',msg=msg)
    con.close()

@app.route('/adduser',methods = ['POST', 'GET'])
def adduser():
    if request.method == 'POST':
        try:
            name = request.form['name']
            email = request.form['email']
            password = request.form['password']
            confirm_password = request.form['confirm_password']

            with sql.connect("user_database.db") as con:
                cur = con.cursor()
                cur.execute("INSERT INTO users (name,email,password,confirm_password) VALUES (?,?,,?)",(name,email,password,confirm_password) )
                con.commit()
                msg = "Record successfully added!"
        except:
            con.rollback()
            msg = "error in insert operation"

        finally:
            return msg
            con.close()

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

```