IMPLEMENTING WEB APPLICATION

CREATE IBM DB2 AND CONNECT WITH PYTHON

Date	14 NOV 2022
Team id	PNT2022TMID49595
Project name	Smart Fashion Recommender Application
Team members	Srivathsava.M
	Balamurugan.A
	Mathivanan.M
	Vinod kumar.L

Create IBM DB2 and connect with python

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

import ibm_db

import re

app = Flask(__name__)

app.secret_key = 'a'

conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32731;Security=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=smy44614;PWD=5Ix4dXwbnT7a1Fbe;","","")

@app.route('/')

def home():

   return render_template('homepage.html')

@app.route('/login')

def log():

   return render_template('index.html')

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

def login():

   msg = ''

   if request.method == 'POST' and 'username' in request.form and 'password' in request.form:

```python
        username = request.form['username']

        password = request.form['password']

        stmt = ibm_db.prepare(conn,'SELECT * FROM users WHERE username = ? AND
password = ?')

        ibm_db.bind_param(stmt,1,username)

        ibm_db.bind_param(stmt,2,password)

        ibm_db.execute(stmt)

        account = ibm_db.fetch_assoc(stmt)

        if account:

            session['loggedin'] = True

            session['username'] = account['USERNAME']

            msg = 'Logged in successfully !'

            return render_template('home1.html', msg = msg)

        else:

            msg = 'Incorrect username / password !'

    return render_template('index.html', msg = msg)


@app.route('/profile',methods=["POST","GET"])

def profile():

    if 'username' in session:

        uid = session['username']

        stmt = ibm_db.prepare(conn, 'SELECT * FROM users WHERE username = ?')

        ibm_db.bind_param(stmt, 1, uid)

        ibm_db.execute(stmt)

        acc = ibm_db.fetch_tuple(stmt)

        return render_template('profile.html',username=acc[0],email=acc[1])

    return render_template('profile.html')


@app.route('/edit')

def edit():
```

```python
    return render_template('editprofile.html')


@app.route('/editprofile', methods=["POST","GET"])
def editprofile():
    if request.method == 'POST' :
        msg = ''
        if 'username' in session:
            uid = session['username']
            username = request.form['username']
            email = request.form['email']
            stmt = ibm_db.prepare(conn,"UPDATE users SET username = ?, email = ? WHERE username = ?")
            ibm_db.bind_param(stmt, 1, username)
            ibm_db.bind_param(stmt, 2, email)
            ibm_db.bind_param(stmt, 3, uid)
            ibm_db.execute(stmt)
            msg = 'Profile Updated Successfully!'
            return render_template('editprofile.html',a = msg)
    return render_template('editprofile.html')


@app.route('/logout')
def logout():
    session.pop('loggedin', None)
    session.pop('id', None)
    session.pop('username', None)
    return redirect(url_for('login'))


@app.route('/reg', methods =['GET', 'POST'])
def register():
    msg = ''
```

```python
    if request.method == 'POST' and 'username' in request.form and 'password' in request.form
and 'email' in request.form :
        username = request.form['username']
        password = request.form['password']
        email = request.form['email']
        stmt = ibm_db.prepare(conn,'SELECT * FROM users WHERE username = ?')
        ibm_db.bind_param(stmt,1,username)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        if account:
            msg = 'Account already exists !'
        elif not re.match(r'[^@]+@[^@]+\.[^@]+', email):
            msg = 'Invalid email address !'
        elif not re.match(r'[A-Za-z0-9]+', username):
            msg = 'Username must contain only characters and numbers !'
        elif not username or not password or not email:
            msg = 'Please fill out the form !'
        else:
            prep_stmt = ibm_db.prepare(conn,"INSERT INTO users VALUES(?, ?, ?)")
            ibm_db.bind_param(prep_stmt, 1, username)
            ibm_db.bind_param(prep_stmt, 2, email)
            ibm_db.bind_param(prep_stmt, 3, password)
            ibm_db.execute(prep_stmt)
            msg = 'You have successfully registered !'
    elif request.method == 'POST':
        msg = 'Please fill out the form !'
    return render_template('reg.html', msg = msg)


@app.route('/admin')
def admin():
```

```python
    return render_template('admin.html')


@app.route('/page2')
def page2():
    return render_template('page2.html')



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


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


@app.route('/single1')
def single1():
    return render_template('single1.html')


@app.route('/single2')
def single2():
    return render_template('single2.html')


@app.route('/single3')
def single3():
    return render_template('single3.html')


@app.route('/single4')
def single4():
    return render_template('single4.html')
```

```python
@app.route('/single5')
def single5():
    return render_template('single5.html')


@app.route('/single6')
def single6():
    return render_template('single6.html')


@app.route('/single7')
def single7():
    return render_template('single7.html')


@app.route('/single8')
def single8():
    return render_template('single8.html')


@app.route('/single9')
def single9():
    return render_template('single9.html')


@app.route('/single10')
def single10():
    return render_template('single10.html')


@app.route('/single11')
def single11():
    return render_template('single11.html')


@app.route('/single12')
```

```python
def single12():
    return render_template('single12.html')


@app.route('/single13')
def single13():
    return render_template('single13.html')


@app.route('/single14')
def single14():
    return render_template('single14.html')


@app.route('/single15')
def single15():
    return render_template('single15.html')



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


@app.route('/home1')
def home1():
    return render_template('home1.html')


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


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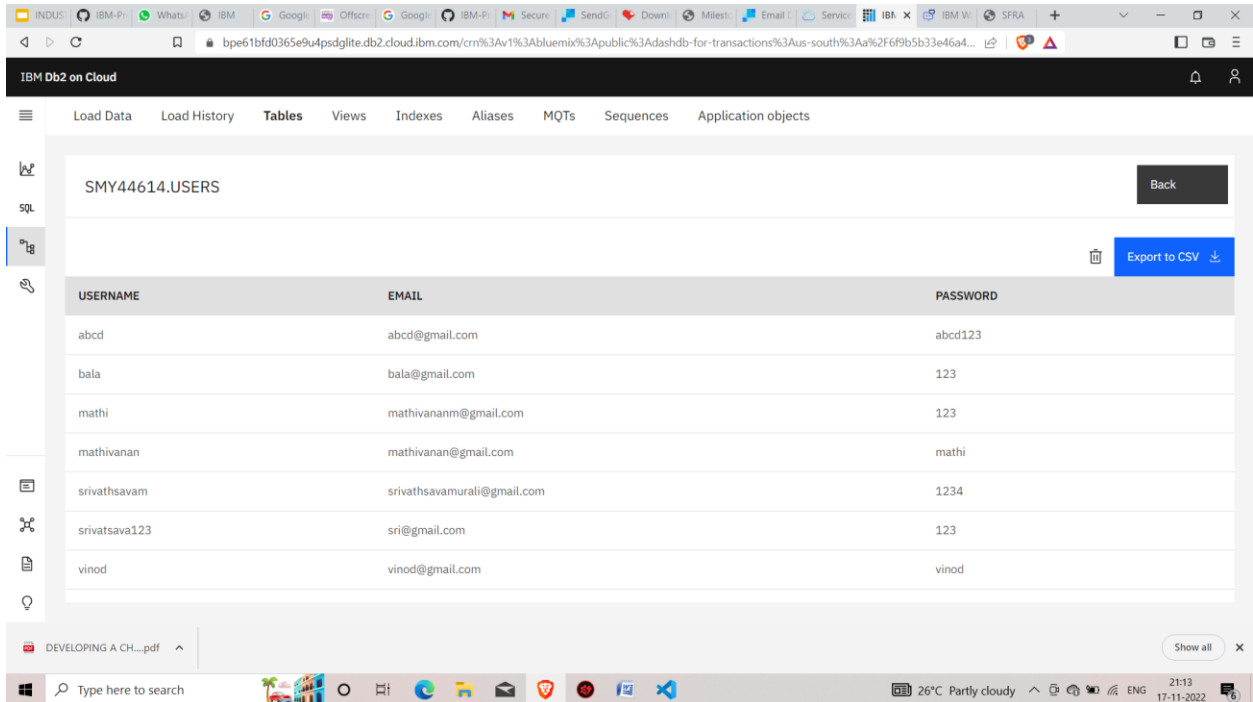
```
    return render_template('add.html')


if __name__ == '__main__':
    app.debug = True
    app.run(host='0.0.0.0',port=8080)
```