from flask import Flask, render\_template, request, redirect, url\_for, session

import ibm\_db

import bcrypt

conn = ibm\_db.connect("DATABASE=bludb;HOSTNAME=;PORT=;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=;PWD=",'','')

# url\_for('static', filename='style.css')

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

app.secret\_key = b'\_5#y2L"F4Q8z\n\xec]/'

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

def home():

if 'email' not in session:

return redirect(url\_for('login'))

return render\_template('home.html',name='Home')

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

def register():

if request.method == 'POST':

email = request.form['email']

username = request.form['username']

rollNo = request.form['rollNo']

password = request.form['password']

if not email or not username or not rollNo or not password:

return render\_template('register.html',error='Please fill all fields')

hash=bcrypt.hashpw(password.encode('utf-8'),bcrypt.gensalt())

query = "SELECT \* FROM USER WHERE email=? OR rollNo=?"

stmt = ibm\_db.prepare(conn, query)

ibm\_db.bind\_param(stmt,1,email)

ibm\_db.bind\_param(stmt,2,rollNo)

ibm\_db.execute(stmt)

isUser = ibm\_db.fetch\_assoc(stmt)

if not isUser:

insert\_sql = "INSERT INTO User(username,email,PASSWORD,rollNo) VALUES (?,?,?,?)"

prep\_stmt = ibm\_db.prepare(conn, insert\_sql)

ibm\_db.bind\_param(prep\_stmt, 1, username)

ibm\_db.bind\_param(prep\_stmt, 2, email)

ibm\_db.bind\_param(prep\_stmt, 3, hash)

ibm\_db.bind\_param(prep\_stmt, 4, rollNo)

ibm\_db.execute(prep\_stmt)

return render\_template('register.html',success="You can login")

else:

return render\_template('register.html',error='Invalid Credentials')

return render\_template('register.html',name='Home')

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

def login():

if request.method == 'POST':

email = request.form['email']

password = request.form['password']

if not email or not password:

return render\_template('login.html',error='Please fill all fields')

query = "SELECT \* FROM USER WHERE email=?"

stmt = ibm\_db.prepare(conn, query)

ibm\_db.bind\_param(stmt,1,email)

ibm\_db.execute(stmt)

isUser = ibm\_db.fetch\_assoc(stmt)

print(isUser,password)

if not isUser:

return render\_template('login.html',error='Invalid Credentials')

isPasswordMatch = bcrypt.checkpw(password.encode('utf-8'),isUser['PASSWORD'].encode('utf-8'))

if not isPasswordMatch:

return render\_template('login.html',error='Invalid Credentials')

session['email'] = isUser['EMAIL']

return redirect(url\_for('home'))

return render\_template('login.html',name='Home')

@app.route('/logout')

def logout():

session.pop('email', None)

return redirect(url\_for('login'))