# -\*- coding: utf-8 -\*-

"""

Spyder Editor

This is a temporary script file.

"""

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

# from flask\_mysqldb import MySQL

# import MySQLdb.cursors

import re

from flask\_db2 import DB2

import ibm\_db

import ibm\_db\_dbi

from sendemail import sendgridmail,sendmail

# from gevent.pywsgi import WSGIServer

import os

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

app.secret\_key = 'a'

# app.config['MYSQL\_HOST'] = 'remotemysql.com'

# app.config['MYSQL\_USER'] = 'D2DxDUPBii'

# app.config['MYSQL\_PASSWORD'] = 'r8XBO4GsMz'

# app.config['MYSQL\_DB'] = 'D2DxDUPBii'

"""

dsn\_hostname = "3883e7e4-18f5-4afe-be8c-fa31c41761d2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud"

dsn\_uid = "sbb93800"

dsn\_pwd = "wobsVLm6ccFxcNLe"

dsn\_driver = "{IBM DB2 ODBC DRIVER}"

dsn\_database = "bludb"

dsn\_port = "31498"

dsn\_protocol = "tcpip"

dsn = (

"DRIVER={0};"

"DATABASE={1};"

"HOSTNAME={2};"

"PORT={3};"

"PROTOCOL={4};"

"UID={5};"

"PWD={6};"

).format(dsn\_driver, dsn\_database, dsn\_hostname, dsn\_port, dsn\_protocol, dsn\_uid, dsn\_pwd)

"""

# app.config['DB2\_DRIVER'] = '{IBM DB2 ODBC DRIVER}'

app.config['database'] = 'bludb'

app.config['hostname'] = '3883e7e4-18f5-4afe-be8c-fa31c41761d2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud'

app.config['port'] = '31498'

app.config['protocol'] = 'tcpip'

app.config['uid'] = 'sbb93800'

app.config['pwd'] = 'wobsVLm6ccFxcNLe'

app.config['security'] = 'SSL'

try:

mysql = DB2(app)

conn\_str='database=bludb;hostname=3883e7e4-18f5-4afe-be8c-fa31c41761d2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;port=31498;protocol=tcpip;\

uid=sbb93800;pwd=wobsVLm6ccFxcNLe;security=SSL'

ibm\_db\_conn = ibm\_db.connect(conn\_str,'','')

print("Database connected without any error !!")

except:

print("IBM DB Connection error : " + DB2.conn\_errormsg())

# app.config['']

# mysql = MySQL(app)

#HOME--PAGE

@app.route("/home")

def home():

return render\_template("homepage.html")

@app.route("/")

def add():

return render\_template("home.html")

#SIGN--UP--OR--REGISTER

@app.route("/signup")

def signup():

return render\_template("signup.html")

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

def register():

msg = ''

print("Break point1")

if request.method == 'POST' :

username = request.form['username']

email = request.form['email']

password = request.form['password']

print("Break point2" + "name: " + username + "------" + email + "------" + password)

try:

print("Break point3")

connectionID = ibm\_db\_dbi.connect(conn\_str, '', '')

cursor = connectionID.cursor()

print("Break point4")

except:

print("No connection Established")

# cursor = mysql.connection.cursor()

# with app.app\_context():

# print("Break point3")

# cursor = ibm\_db\_conn.cursor()

# print("Break point4")

print("Break point5")

sql = "SELECT \* FROM register WHERE username = ?"

stmt = ibm\_db.prepare(ibm\_db\_conn, sql)

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

ibm\_db.execute(stmt)

result = ibm\_db.execute(stmt)

print(result)

account = ibm\_db.fetch\_row(stmt)

print(account)

param = "SELECT \* FROM register WHERE username = " + "\'" + username + "\'"

res = ibm\_db.exec\_immediate(ibm\_db\_conn, param)

print("---- ")

dictionary = ibm\_db.fetch\_assoc(res)

while dictionary != False:

print("The ID is : ", dictionary["USERNAME"])

dictionary = ibm\_db.fetch\_assoc(res)

# dictionary = ibm\_db.fetch\_assoc(result)

# cursor.execute(stmt)

# account = cursor.fetchone()

# print(account)

# while ibm\_db.fetch\_row(result) != False:

# # account = ibm\_db.result(stmt)

# print(ibm\_db.result(result, "username"))

# print(dictionary["username"])

print("break point 6")

if account:

msg = 'Username already exists !'

elif not re.match(r'[^@]+@[^@]+\.[^@]+', email):

msg = 'Invalid email address !'

elif not re.match(r'[A-Za-z0-9]+', username):

msg = 'name must contain only characters and numbers !'

else:

sql2 = "INSERT INTO register (username, email,password) VALUES (?, ?, ?)"

stmt2 = ibm\_db.prepare(ibm\_db\_conn, sql2)

ibm\_db.bind\_param(stmt2, 1, username)

ibm\_db.bind\_param(stmt2, 2, email)

ibm\_db.bind\_param(stmt2, 3, password)

ibm\_db.execute(stmt2)

# cursor.execute('INSERT INTO register VALUES (NULL, % s, % s, % s)', (username, email,password))

# mysql.connection.commit()

msg = 'You have successfully registered !'

return render\_template('signup.html', msg = msg)

#LOGIN--PAGE

@app.route("/signin")

def signin():

return render\_template("login.html")

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

def login():

global userid

msg = ''

if request.method == 'POST' :

username = request.form['username']

password = request.form['password']

# cursor = mysql.connection.cursor()

# cursor.execute('SELECT \* FROM register WHERE username = % s AND password = % s', (username, password ),)

# account = cursor.fetchone()

# print (account)

sql = "SELECT \* FROM register WHERE username = ? and password = ?"

stmt = ibm\_db.prepare(ibm\_db\_conn, sql)

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

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

result = ibm\_db.execute(stmt)

print(result)

account = ibm\_db.fetch\_row(stmt)

print(account)

param = "SELECT \* FROM register WHERE username = " + "\'" + username + "\'" + " and password = " + "\'" + password + "\'"

res = ibm\_db.exec\_immediate(ibm\_db\_conn, param)

dictionary = ibm\_db.fetch\_assoc(res)

# sendmail("hello sakthi","sivasakthisairam@gmail.com")

if account:

session['loggedin'] = True

session['id'] = dictionary["ID"]

userid = dictionary["ID"]

session['username'] = dictionary["USERNAME"]

session['email'] = dictionary["EMAIL"]

return redirect('/home')

else:

msg = 'Incorrect username / password !'

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

#ADDING----DATA

@app.route("/add")

def adding():

return render\_template('add.html')

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

def addexpense():

date = request.form['date']

expensename = request.form['expensename']

amount = request.form['amount']

paymode = request.form['paymode']

category = request.form['category']

print(date)

p1 = date[0:10]

p2 = date[11:13]

p3 = date[14:]

p4 = p1 + "-" + p2 + "." + p3 + ".00"

print(p4)

# cursor = mysql.connection.cursor()

# cursor.execute('INSERT INTO expenses VALUES (NULL, % s, % s, % s, % s, % s, % s)', (session['id'] ,date, expensename, amount, paymode, category))

# mysql.connection.commit()

# print(date + " " + expensename + " " + amount + " " + paymode + " " + category)

sql = "INSERT INTO expenses (userid, date, expensename, amount, paymode, category) VALUES (?, ?, ?, ?, ?, ?)"

stmt = ibm\_db.prepare(ibm\_db\_conn, sql)

ibm\_db.bind\_param(stmt, 1, session['id'])

ibm\_db.bind\_param(stmt, 2, p4)

ibm\_db.bind\_param(stmt, 3, expensename)

ibm\_db.bind\_param(stmt, 4, amount)

ibm\_db.bind\_param(stmt, 5, paymode)

ibm\_db.bind\_param(stmt, 6, category)

ibm\_db.execute(stmt)

print("Expenses added")

# email part

param = "SELECT \* FROM expenses WHERE userid = " + str(session['id']) + " AND MONTH(date) = MONTH(current timestamp) AND YEAR(date) = YEAR(current timestamp) ORDER BY date DESC"

res = ibm\_db.exec\_immediate(ibm\_db\_conn, param)

dictionary = ibm\_db.fetch\_assoc(res)

expense = []

while dictionary != False:

temp = []

temp.append(dictionary["ID"])

temp.append(dictionary["USERID"])

temp.append(dictionary["DATE"])

temp.append(dictionary["EXPENSENAME"])

temp.append(dictionary["AMOUNT"])

temp.append(dictionary["PAYMODE"])

temp.append(dictionary["CATEGORY"])

expense.append(temp)

print(temp)

dictionary = ibm\_db.fetch\_assoc(res)

total=0

for x in expense:

total += x[4]

param = "SELECT id, limitss FROM limits WHERE userid = " + str(session['id']) + " ORDER BY id DESC LIMIT 1"

res = ibm\_db.exec\_immediate(ibm\_db\_conn, param)

dictionary = ibm\_db.fetch\_assoc(res)

row = []

s = 0

while dictionary != False:

temp = []

temp.append(dictionary["LIMITSS"])

row.append(temp)

dictionary = ibm\_db.fetch\_assoc(res)

s = temp[0]

if total > int(s):

msg = "Hello " + session['username'] + " , " + "you have crossed the monthly limit of Rs. " + s + "/- !!!" + "\n" + "Thank you, " + "\n" + "Team Personal Expense Tracker."

sendmail(msg,session['email'])

return redirect("/display")

#DISPLAY---graph

@app.route("/display")

def display():

print(session["username"],session['id'])

# cursor = mysql.connection.cursor()

# cursor.execute('SELECT \* FROM expenses WHERE userid = % s AND date ORDER BY `expenses`.`date` DESC',(str(session['id'])))

# expense = cursor.fetchall()

param = "SELECT \* FROM expenses WHERE userid = " + str(session['id']) + " ORDER BY date DESC"

res = ibm\_db.exec\_immediate(ibm\_db\_conn, param)

dictionary = ibm\_db.fetch\_assoc(res)

expense = []

while dictionary != False:

temp = []

temp.append(dictionary["ID"])

temp.append(dictionary["USERID"])

temp.append(dictionary["DATE"])

temp.append(dictionary["EXPENSENAME"])

temp.append(dictionary["AMOUNT"])

temp.append(dictionary["PAYMODE"])

temp.append(dictionary["CATEGORY"])

expense.append(temp)

print(temp)

dictionary = ibm\_db.fetch\_assoc(res)

return render\_template('display.html' ,expense = expense)

#delete---the--data

@app.route('/delete/<string:id>', methods = ['POST', 'GET' ])

def delete(id):

# cursor = mysql.connection.cursor()

# cursor.execute('DELETE FROM expenses WHERE id = {0}'.format(id))

# mysql.connection.commit()

param = "DELETE FROM expenses WHERE id = " + id

res = ibm\_db.exec\_immediate(ibm\_db\_conn, param)

print('deleted successfully')

return redirect("/display")

#UPDATE---DATA

@app.route('/edit/<id>', methods = ['POST', 'GET' ])

def edit(id):

# cursor = mysql.connection.cursor()

# cursor.execute('SELECT \* FROM expenses WHERE id = %s', (id,))

# row = cursor.fetchall()

param = "SELECT \* FROM expenses WHERE id = " + id

res = ibm\_db.exec\_immediate(ibm\_db\_conn, param)

dictionary = ibm\_db.fetch\_assoc(res)

row = []

while dictionary != False:

temp = []

temp.append(dictionary["ID"])

temp.append(dictionary["USERID"])

temp.append(dictionary["DATE"])

temp.append(dictionary["EXPENSENAME"])

temp.append(dictionary["AMOUNT"])

temp.append(dictionary["PAYMODE"])

temp.append(dictionary["CATEGORY"])

row.append(temp)

print(temp)

dictionary = ibm\_db.fetch\_assoc(res)

print(row[0])

return render\_template('edit.html', expenses = row[0])

@app.route('/update/<id>', methods = ['POST'])

def update(id):

if request.method == 'POST' :

date = request.form['date']

expensename = request.form['expensename']

amount = request.form['amount']

paymode = request.form['paymode']

category = request.form['category']

# cursor = mysql.connection.cursor()

# cursor.execute("UPDATE `expenses` SET `date` = % s , `expensename` = % s , `amount` = % s, `paymode` = % s, `category` = % s WHERE `expenses`.`id` = % s ",(date, expensename, amount, str(paymode), str(category),id))

# mysql.connection.commit()

p1 = date[0:10]

p2 = date[11:13]

p3 = date[14:]

p4 = p1 + "-" + p2 + "." + p3 + ".00"

sql = "UPDATE expenses SET date = ? , expensename = ? , amount = ?, paymode = ?, category = ? WHERE id = ?"

stmt = ibm\_db.prepare(ibm\_db\_conn, sql)

ibm\_db.bind\_param(stmt, 1, p4)

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

ibm\_db.bind\_param(stmt, 3, amount)

ibm\_db.bind\_param(stmt, 4, paymode)

ibm\_db.bind\_param(stmt, 5, category)

ibm\_db.bind\_param(stmt, 6, id)

ibm\_db.execute(stmt)

print('successfully updated')

return redirect("/display")

#limit

@app.route("/limit" )

def limit():

return redirect('/limitn')

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

def limitnum():

if request.method == "POST":

number= request.form['number']

# cursor = mysql.connection.cursor()

# cursor.execute('INSERT INTO limits VALUES (NULL, % s, % s) ',(session['id'], number))

# mysql.connection.commit()

sql = "INSERT INTO limits (userid, limitss) VALUES (?, ?)"

stmt = ibm\_db.prepare(ibm\_db\_conn, sql)

ibm\_db.bind\_param(stmt, 1, session['id'])

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

ibm\_db.execute(stmt)

return redirect('/limitn')

@app.route("/limitn")

def limitn():

# cursor = mysql.connection.cursor()

# cursor.execute('SELECT limitss FROM `limits` ORDER BY `limits`.`id` DESC LIMIT 1')

# x= cursor.fetchone()

# s = x[0]

param = "SELECT id, limitss FROM limits WHERE userid = " + str(session['id']) + " ORDER BY id DESC LIMIT 1"

res = ibm\_db.exec\_immediate(ibm\_db\_conn, param)

dictionary = ibm\_db.fetch\_assoc(res)

row = []

s = " /-"

while dictionary != False:

temp = []

temp.append(dictionary["LIMITSS"])

row.append(temp)

dictionary = ibm\_db.fetch\_assoc(res)

s = temp[0]

return render\_template("limit.html" , y= s)

#REPORT

@app.route("/today")

def today():

# cursor = mysql.connection.cursor()

# cursor.execute('SELECT TIME(date) , amount FROM expenses WHERE userid = %s AND DATE(date) = DATE(NOW()) ',(str(session['id'])))

# texpense = cursor.fetchall()

# print(texpense)

param1 = "SELECT TIME(date) as tn, amount FROM expenses WHERE userid = " + str(session['id']) + " AND DATE(date) = DATE(current timestamp) ORDER BY date DESC"

res1 = ibm\_db.exec\_immediate(ibm\_db\_conn, param1)

dictionary1 = ibm\_db.fetch\_assoc(res1)

texpense = []

while dictionary1 != False:

temp = []

temp.append(dictionary1["TN"])

temp.append(dictionary1["AMOUNT"])

texpense.append(temp)

print(temp)

dictionary1 = ibm\_db.fetch\_assoc(res1)

# cursor = mysql.connection.cursor()

# cursor.execute('SELECT \* FROM expenses WHERE userid = % s AND DATE(date) = DATE(NOW()) AND date ORDER BY `expenses`.`date` DESC',(str(session['id'])))

# expense = cursor.fetchall()

param = "SELECT \* FROM expenses WHERE userid = " + str(session['id']) + " AND DATE(date) = DATE(current timestamp) ORDER BY date DESC"

res = ibm\_db.exec\_immediate(ibm\_db\_conn, param)

dictionary = ibm\_db.fetch\_assoc(res)

expense = []

while dictionary != False:

temp = []

temp.append(dictionary["ID"])

temp.append(dictionary["USERID"])

temp.append(dictionary["DATE"])

temp.append(dictionary["EXPENSENAME"])

temp.append(dictionary["AMOUNT"])

temp.append(dictionary["PAYMODE"])

temp.append(dictionary["CATEGORY"])

expense.append(temp)

print(temp)

dictionary = ibm\_db.fetch\_assoc(res)

total=0

t\_food=0

t\_entertainment=0

t\_business=0

t\_rent=0

t\_EMI=0

t\_other=0

for x in expense:

total += x[4]

if x[6] == "food":

t\_food += x[4]

elif x[6] == "entertainment":

t\_entertainment += x[4]

elif x[6] == "business":

t\_business += x[4]

elif x[6] == "rent":

t\_rent += x[4]

elif x[6] == "EMI":

t\_EMI += x[4]

elif x[6] == "other":

t\_other += x[4]

print(total)

print(t\_food)

print(t\_entertainment)

print(t\_business)

print(t\_rent)

print(t\_EMI)

print(t\_other)

return render\_template("today.html", texpense = texpense, expense = expense, total = total ,

t\_food = t\_food,t\_entertainment = t\_entertainment,

t\_business = t\_business, t\_rent = t\_rent,

t\_EMI = t\_EMI, t\_other = t\_other )

@app.route("/month")

def month():

# cursor = mysql.connection.cursor()

# cursor.execute('SELECT DATE(date), SUM(amount) FROM expenses WHERE userid= %s AND MONTH(DATE(date))= MONTH(now()) GROUP BY DATE(date) ORDER BY DATE(date) ',(str(session['id'])))

# texpense = cursor.fetchall()

# print(texpense)

param1 = "SELECT DATE(date) as dt, SUM(amount) as tot FROM expenses WHERE userid = " + str(session['id']) + " AND MONTH(date) = MONTH(current timestamp) AND YEAR(date) = YEAR(current timestamp) GROUP BY DATE(date) ORDER BY DATE(date)"

res1 = ibm\_db.exec\_immediate(ibm\_db\_conn, param1)

dictionary1 = ibm\_db.fetch\_assoc(res1)

texpense = []

while dictionary1 != False:

temp = []

temp.append(dictionary1["DT"])

temp.append(dictionary1["TOT"])

texpense.append(temp)

print(temp)

dictionary1 = ibm\_db.fetch\_assoc(res1)

# cursor = mysql.connection.cursor()

# cursor.execute('SELECT \* FROM expenses WHERE userid = % s AND MONTH(DATE(date))= MONTH(now()) AND date ORDER BY `expenses`.`date` DESC',(str(session['id'])))

# expense = cursor.fetchall()

param = "SELECT \* FROM expenses WHERE userid = " + str(session['id']) + " AND MONTH(date) = MONTH(current timestamp) AND YEAR(date) = YEAR(current timestamp) ORDER BY date DESC"

res = ibm\_db.exec\_immediate(ibm\_db\_conn, param)

dictionary = ibm\_db.fetch\_assoc(res)

expense = []

while dictionary != False:

temp = []

temp.append(dictionary["ID"])

temp.append(dictionary["USERID"])

temp.append(dictionary["DATE"])

temp.append(dictionary["EXPENSENAME"])

temp.append(dictionary["AMOUNT"])

temp.append(dictionary["PAYMODE"])

temp.append(dictionary["CATEGORY"])

expense.append(temp)

print(temp)

dictionary = ibm\_db.fetch\_assoc(res)

total=0

t\_food=0

t\_entertainment=0

t\_business=0

t\_rent=0

t\_EMI=0

t\_other=0

for x in expense:

total += x[4]

if x[6] == "food":

t\_food += x[4]

elif x[6] == "entertainment":

t\_entertainment += x[4]

elif x[6] == "business":

t\_business += x[4]

elif x[6] == "rent":

t\_rent += x[4]

elif x[6] == "EMI":

t\_EMI += x[4]

elif x[6] == "other":

t\_other += x[4]

print(total)

print(t\_food)

print(t\_entertainment)

print(t\_business)

print(t\_rent)

print(t\_EMI)

print(t\_other)

return render\_template("today.html", texpense = texpense, expense = expense, total = total ,

t\_food = t\_food,t\_entertainment = t\_entertainment,

t\_business = t\_business, t\_rent = t\_rent,

t\_EMI = t\_EMI, t\_other = t\_other )

@app.route("/year")

def year():

# cursor = mysql.connection.cursor()

# cursor.execute('SELECT MONTH(date), SUM(amount) FROM expenses WHERE userid= %s AND YEAR(DATE(date))= YEAR(now()) GROUP BY MONTH(date) ORDER BY MONTH(date) ',(str(session['id'])))

# texpense = cursor.fetchall()

# print(texpense)

param1 = "SELECT MONTH(date) as mn, SUM(amount) as tot FROM expenses WHERE userid = " + str(session['id']) + " AND YEAR(date) = YEAR(current timestamp) GROUP BY MONTH(date) ORDER BY MONTH(date)"

res1 = ibm\_db.exec\_immediate(ibm\_db\_conn, param1)

dictionary1 = ibm\_db.fetch\_assoc(res1)

texpense = []

while dictionary1 != False:

temp = []

temp.append(dictionary1["MN"])

temp.append(dictionary1["TOT"])

texpense.append(temp)

print(temp)

dictionary1 = ibm\_db.fetch\_assoc(res1)

# cursor = mysql.connection.cursor()

# cursor.execute('SELECT \* FROM expenses WHERE userid = % s AND YEAR(DATE(date))= YEAR(now()) AND date ORDER BY `expenses`.`date` DESC',(str(session['id'])))

# expense = cursor.fetchall()

param = "SELECT \* FROM expenses WHERE userid = " + str(session['id']) + " AND YEAR(date) = YEAR(current timestamp) ORDER BY date DESC"

res = ibm\_db.exec\_immediate(ibm\_db\_conn, param)

dictionary = ibm\_db.fetch\_assoc(res)

expense = []

while dictionary != False:

temp = []

temp.append(dictionary["ID"])

temp.append(dictionary["USERID"])

temp.append(dictionary["DATE"])

temp.append(dictionary["EXPENSENAME"])

temp.append(dictionary["AMOUNT"])

temp.append(dictionary["PAYMODE"])

temp.append(dictionary["CATEGORY"])

expense.append(temp)

print(temp)

dictionary = ibm\_db.fetch\_assoc(res)

total=0

t\_food=0

t\_entertainment=0

t\_business=0

t\_rent=0

t\_EMI=0

t\_other=0

for x in expense:

total += x[4]

if x[6] == "food":

t\_food += x[4]

elif x[6] == "entertainment":

t\_entertainment += x[4]

elif x[6] == "business":

t\_business += x[4]

elif x[6] == "rent":

t\_rent += x[4]

elif x[6] == "EMI":

t\_EMI += x[4]

elif x[6] == "other":

t\_other += x[4]

print(total)

print(t\_food)

print(t\_entertainment)

print(t\_business)

print(t\_rent)

print(t\_EMI)

print(t\_other)

return render\_template("today.html", texpense = texpense, expense = expense, total = total ,

t\_food = t\_food,t\_entertainment = t\_entertainment,

t\_business = t\_business, t\_rent = t\_rent,

t\_EMI = t\_EMI, t\_other = t\_other )

#log-out

@app.route('/logout')

def logout():

session.pop('loggedin', None)

session.pop('id', None)

session.pop('username', None)

session.pop('email', None)

return render\_template('home.html')

port = os.getenv('VCAP\_APP\_PORT', '8080')

if \_\_name\_\_ == "\_\_main\_\_":

app.secret\_key = os.urandom(12)

app.run(debug=True, host='0.0.0.0', port=port)