Assignment -2

Assignment Date	14 OCTOBER 2022
Student Name	G.SATHIYA
Student Roll Number	422119104024
Maximum Marks	2 Marks

Question 1.

Create User table with email, username, roll number, password.

CREATE TABLE usertable(

roll_number int,

username varchar(300),

mail varchar(300),

password varchar(300));

	Field	Type	Null	Key	Default	Extra
▶	roll_number	int	YES		NULL	
	username	varchar(300)	YES		NULL	
	mail	varchar(300)	YES		NULL	
	password	varchar(300)	YES		NULL	

Question 2.

Perform UPDATE, DELETE Queries with user table

INSERT STATEMENT:

INSERT INTO usertable

(roll_number, username, mail,password) VALUES

(1,salethreena,reena001@gmail.com,reena12),

(2,'karthika','karthika11@gmail.com','karthika123'),

(3,'samshath',samshath123@gmail.com','begam34'),

(4,'sathiya',sathiya78@gmail.com','sathiya345');

roll_number	username	mail	password
1	salethre...	reena001@gmail.com	reena12
2	karthika	karthika11@gmail.com	karthika123
3	samshath	samshath123@gmail.com	begam34
4	sathiya	sathiya78@gmail.com	sathiya345

UPDATE STATEMENT:

UPDATE usertable

SET username='sathiya'

WHERE roll_number=3;

roll_number	username	mail	password
1	salethre...	reena001@gmail.com	reena12
2	karthika	karthika11@gmail.com	karthika123
3	sathiya	samshath123@gmail.com	begam34
4	sathiya	sathiya78@gmail.com	sathiya345

DELETE STATEMENT:

DELETE FROM usertable where roll_number=4;

roll_number	username	mail	password
1	salethre...	reena001@gmail.com	reena12
2	karthika	karthika11@gmail.com	karthika123
3	samshath	samshath123@gmail.com	begam34

Question 3.

Connect python code to db2.

```
dsn_hostname = "2d46b6b4-cbf6-40eb-bbce-
6251e6ba0300.bs2io90l08kqb1od8lcg.databases.appdomain.cloud"
dsn_uid = "avengers005"
dsn_pwd = "multiverse"
dsn_driver = "{IBM DB2 ODBC DRIVER}"
dsn_database = "DB_ADMIN"
dsn_port = "32328"
dsn_protocol = "TCPIP"
dsn_security = "SSL"
dsn = (
  "DRIVER={0};"
  "DATABASE={1};"
  "HOSTNAME={2};"
  "PORT={3};"
  "PROTOCOL={4};"
  "UID={5};"
  "PWD={6};"
  "SECURITY={7};").format(dsn_driver, dsn_database, dsn_hostname, dsn_port,
dsn_protocol, dsn_uid, dsn_pwd,dsn_security)
print(dsn)
try:
  conn = ibm_db.connect(dsn, "", "")
  print ("Connected to database: ", dsn_database, "as user: ", dsn_uid, "on host: ",
dsn_hostname)

except:
  print ("Unable to connect: ", ibm_db.conn_errormsg() )
```

Question 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 with username and password. If the user is valid show the welcome page

Flask app:

App.py

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

from flask_mysqldb import MySQL

import MySQLdb.cursors

import re

app = Flask(__name__)

app.secret_key = 'your secret key'

app.config['MYSQL_HOST'] = 'localhost'

app.config['MYSQL_USER'] = 'avengers005'

app.config['MYSQL_PASSWORD'] = 'multiverse'

app.config['MYSQL_DB'] = 'login'

mysql = MySQL(app)

@app.route('/')

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

def login():

    msg = ''

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

        username = request.form['username']

        password = request.form['password']

        cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
```

```python
        cursor.execute('SELECT * FROM accounts WHERE username = % s AND password = % s',
(username, password, ))

        account = cursor.fetchone()

        if account:

            session['loggedin'] = True

            session['id'] = account['id']

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

            msg = 'Logged in successfully !'

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

        else:

            msg = 'Incorrect username / password !'

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


@app.route('/logout')

def logout():

    session.pop('loggedin', None)

    session.pop('id', None)

    session.pop('username', None)

    return redirect(url_for('login'))


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

def register():

    msg = ''

    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']
```

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

        cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)

        cursor.execute('SELECT * FROM accounts WHERE username = % s', (username, ))

        account = cursor.fetchone()

        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:

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

            mysql.connection.commit()

            msg = 'You have successfully registered !'

    elif request.method == 'POST':

        msg = 'Please fill out the form !'

    return render_template('register.html', msg = msg)
```

```
(env) C:\Users\GREAT\Desktop\login>flask run
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
```

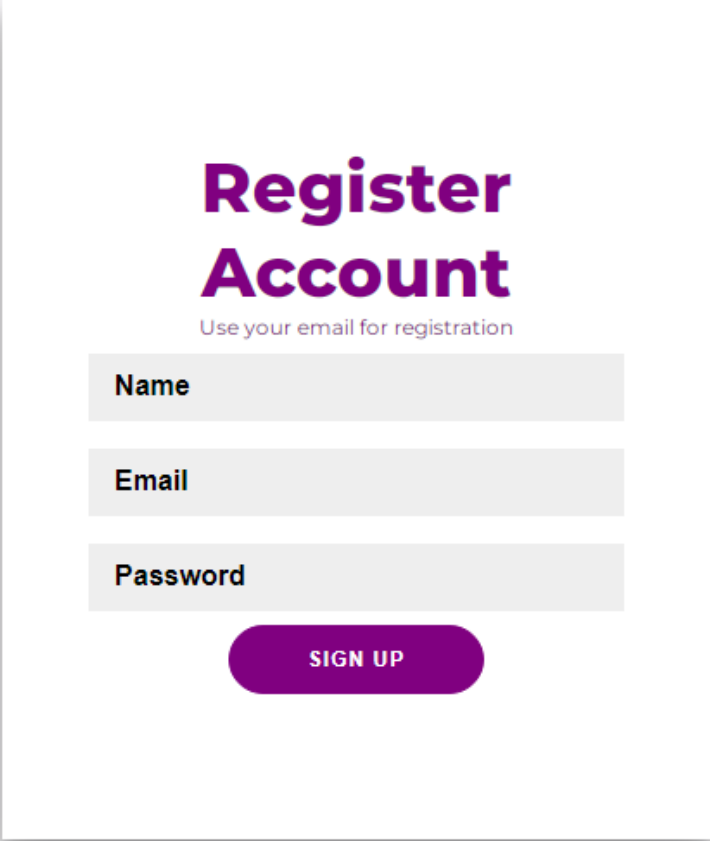**LOGIN PAGE:**

Login

Use your account to sign in

Name

Email

Password

SIGN IN

Don't have an account? Sign Up here

REGISTER PAGE:

WELCOME PAGE: