Assignment - 2

Assignment Date	16 October 2022
Student Name	Amrish D
Student Roll Number	14229106008
Maximum Marks	2 Marks

1.Create User Table with user email username,roll number,password.

```
CREATE TABLE USER
    (EMAIL    VARCHAR(30)      NOT NULL,
     USERNAME VARCHAR(24)  NOT NULL,
     ROLLNO  CHAR(3)      NOT NULL,
     PASSWORD VARCHAR(6)      NOT NULL)
    IN DATABASE DSN8D11A
    VALIDPROC DSN8EAPR;
```

2.Perform UPDATE,DELETE Queries with user table

Delete:

```
EXEC SQL DELETE FROM USER
   WHERE USERNAME = 'E11' OR USERNAME = 'D21';
```

Update:

```
UPDATE USER
    SET ROLLNO='3565'
    WHERE USERNAME='000190';
```

3. Connect python code to db2.

```
import ibm_db
conn = ibm_db.connect("database","username","password")
```

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 the

database and navigate to the login page to authenticate user username and password. If the user is valid show the welcome page

```
# Store this code in 'app.py' file

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'] = 'root'
app.config['MYSQL_PASSWORD'] = 'your password'
app.config['MYSQL_DB'] = 'geeklogin'

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)
                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']
                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)
```