

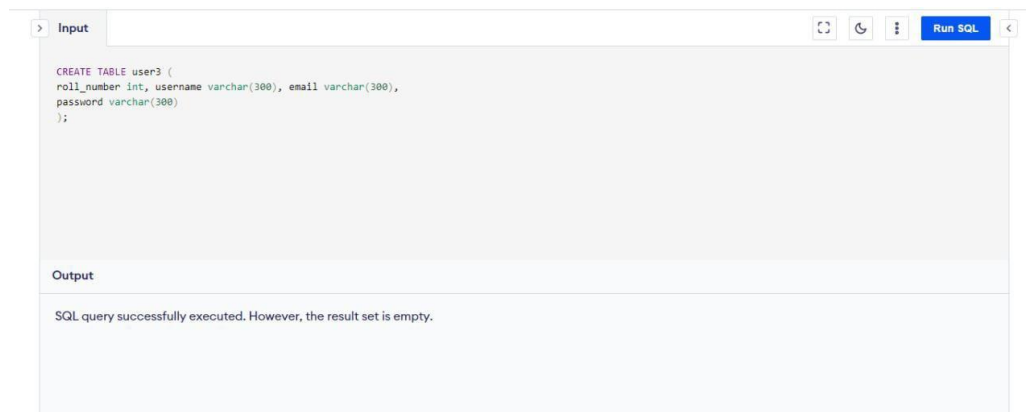
Assignment -2

Assignment Date	19 September 2022
Student Name	Sridevi.M
Student Roll Number	310119104077
Maximum Marks	2 Marks

Question-1:

Create user table with user with email, username, roll number, password

```
CREATE TABLE user3 (  
    roll_number int,  
    username varchar(300),  
    email varchar(300),  
    password varchar(300)  
);
```



The screenshot shows a web-based SQL editor interface. At the top, there is a tab labeled 'Input'. Below the tab, the SQL query is entered in a text area: `CREATE TABLE user3 (
 roll_number int, username varchar(300), email varchar(300),
 password varchar(300)
);`. To the right of the text area are several icons: a refresh icon, a save icon, a settings icon, and a blue button labeled 'Run SQL'. Below the text area is a section labeled 'Output'. The output text reads: 'SQL query successfully executed. However, the result set is empty.'

1. Perform UPDATE, DELETE Queries with user table

INSERT Statement:

INSERT INTO users3(roll_number, username ,email, password) VALUES
(1, 'Vinothkumar', 'vinothkuma415a@gmail.com','vinoth88'),
(2, 'Sasirajan', 'sasirajan@gmail.com','sasi71'),
(3, 'Sutharshan', 'sutharshan@gmail.com','sutharshan81'),
(4, 'Yuvaraj', 'yuvi@gmail.com','yuvaraj94');

Online SQL Editor

Input

```
CREATE TABLE users5 (  
  roll_number int, username varchar(300), email varchar(300),  
  password varchar(300)  
);  
INSERT INTO users5( roll_number, username ,email, password) VALUES  
(1, 'Sneha', 'snegaosuran@gmail.com','sneha74'),  
(2, 'Sridevi', 'sridevi123@gmail.com','sridevi77'),  
(3, 'Sneha', 'sneha@gmail.com','snehaa75'),  
(4, 'thenmozhi', 'thenmozhi@gmail.com','thenmozhi84'),  
(5, 'vijayalaxmi', 'viji@gmail.com','viji87');
```

Run SQL

Output

SQL query successfully executed. However, the result set is empty.

Users5

roll_number	username	email
1	Sneha	snegaosuran@gmail.com
2	Sridevi	sridevi123@gmail.com
3	Sneha	sneha@gmail.com
4	thenmozhi	thenmozhi@gmail.com
5	vijayalaxmi	viji@gmail.com

UPDATE Statement:

UPDATE users4 SET username = 'Sutharshan' WHERE roll_number = '2';

Online SQL Editor

> Input

Run SQL

>

```
CREATE TABLE users7 (  
  roll_number int, username varchar(300), email varchar(300),  
  password varchar(300)  
);  
INSERT INTO users7( roll_number, username ,email, password) VALUES  
(1, 'Sneha', 'snegaosuran@gmail.com','sneha74'),  
(2, 'Sridevi', 'sridevi123@gmail.com','sridevi77'),  
(3, 'Sneha', 'sneha@gmail.com','snehaa75'),  
(4, 'thenmozhi', 'thenmozhi@gmail.com','thenmozhi84'),  
(5, 'vijayalaxmi', 'viji@gmail.com','viji87');  
UPDATE users7 SET username = 'Sneha' WHERE roll_number = '4';
```

Output

SQL query successfully executed. However, the result set is empty.

Users7

roll_number	username	email
1	Sneha	snegaosuran@gmail.com
2	Sridevi	sridevi123@gmail.com
3	Sneha	sneha@gmail.com
4	Sneha	thenmozhi@gmail.com
5	vijayalaxmi	viji@gmail.com

Insert Statement:

```
INSERT INTO users5 values(5,'aa','aaa@gmail.com','aasdfg2');
```

Online SQL Editor

> Input

Run SQL

```
INSERT INTO users8 values(5,'aa','aaa@gmail.com','aasdfg2');
```

Output

SQL query successfully executed. However, the result set is empty.

Users8

roll_number	username	email
1	Sneha	snegaosuran@gmail.com
2	Sridevi	sridevi123@gmail.com
3	Sneha	sneha@gmail.com
4	Sneha	thenmozhi@gmail.com
5	vijayalaxmi	viji@gmail.com
5	aa	aaa@gmail.com

DELETE Statement:

DELETE FROM users6 where roll_number='5'



Users9

roll_number	username	email
1	Sneha	snegaosuran@gmail.com
2	Sridevi	sridevi123@gmail.com
3	Sneha	sneha@gmail.com
4	Sneha	thenmozhi@gmail.com

Connect python with db2

```
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=824dfd4d-99de-440d-9991-629c01b3832d.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;PORT=30119;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=lvq43963;PWD=BsnsG112sBgIRhVN",'')
```

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

```
from flask_mysqlldb import MySQL

import MySQLdb.cursors

import reapp = 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)

```


Register

Enter Your Username

Enter Your Password

Enter Your Email ID

Sign Up

Already have an account? [Sign In here](#)

Login

Enter Your Username

Enter Your Password

Sign In

Don't have an account? [Sign Up here](#)

Index

Hi user!!

Welcome to the index page...

Logout

3) Write a flask program which should display resume details and also have upload resume option by using file uploading.

UPLOAD.PY:

```
from flask import *
app = Flask(__name__)

@app.route('/')
def upload():
    return render_template("file_upload_form.html")

@app.route('/success', methods = ['POST'])
def success():
    if request.method == 'POST':
        f = request.files['file']
        f.save(f.filename)
        return render_template("success.html", name = f.filename)

if __name__ == '__main__':
    app.run(debug = True)
```

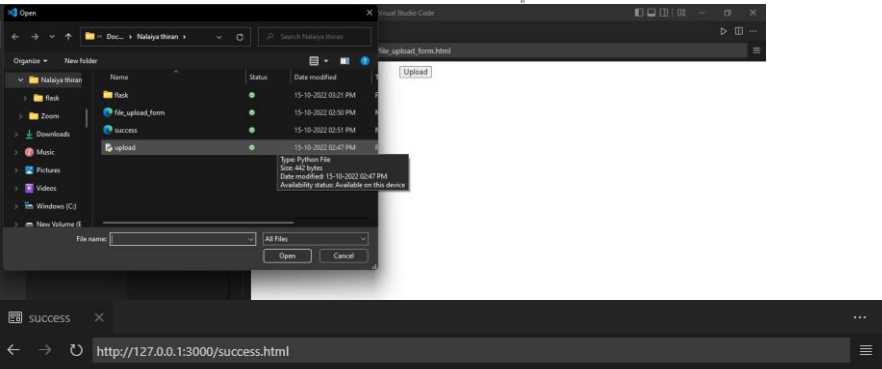
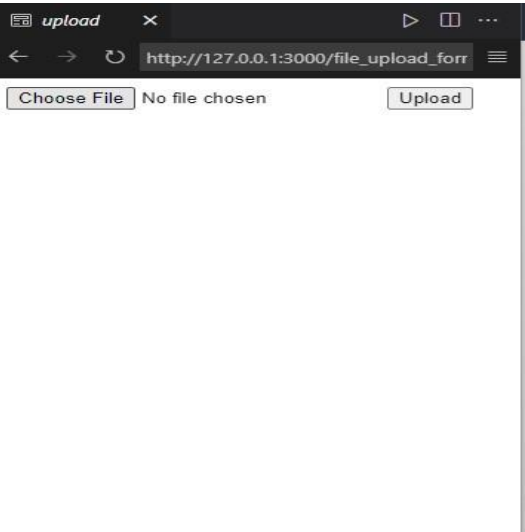
FILE_UPLOAD_FORM.HTML:

```
<html>
<head>
    <title>upload</title>
</head>
<body>
    <form action = "/success" method = "post" enctype="multipart/form-data">
        <input type="file" name="file" />
        <input type = "submit" value="Upload">
    </form>
</body>
</html>
```

SUCCESS.HTML:

```
<html>
<head>
<title>success</title>
</head>
<body>
<p>File uploaded successfully</p>
<p>File Name: {{ name }}</p>
</body>
</html>
```

OUTPUT:



File uploaded successfully
File Name: {{file_upload_form}}