

## Assignment -2

Assignment Date	19 September 2022
Student Name	BHUVANESHKUMAR S
Student Roll Number	310119104020
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 an online SQL editor interface. At the top, there is a tab labeled 'Input'. To the right of the tab are several icons: a full-screen icon, a moon icon for dark mode, a vertical ellipsis menu icon, and a blue 'Run SQL' button. Below the tab, the editor area contains the following text:

```
-- Online SQL Editor to Run SQL Online.  
-- Use the editor to create new tables, insert data and all other SQL operations.  
  
CREATE TABLE users(roll_number int, username varchar(300), email varchar(300),  
password varchar(300));
```

## 1. Perform UPDATE, DELETE Queries with user table

### INSERT Statement:

```
INSERT INTO users3( roll_number, username ,email, password) VALUES
(1, 'Harishvaran', 'romanharish465@gmail.com','Harishvaran 36'),
(2, 'Guru pradeep', 'gurupratheep98@gmail.com','Guru pradeep 33'),
(3, 'Hari prasad', 'parasad57harish@gmail.com','Hari prasad 35'),
(4, 'Bhuvaneshkumar', 'buvaneshbs999@gmail.com','Bhuvaneshkumar 20');
(5, 'Gnanesh', 'gnaneshg1002@gmail.com','Gnanesh 31');
```

< Input

Run SQL

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

INSERT INTO users( roll_number, username ,email, password) VALUES (1,
'Harishvaran', 'romanharish465@gmail.com','Harishvaran 36'),
(2, 'Guru pradeep', 'gurupratheep98@gmail.com','Guru pradeep 33'),
(3, 'Hari prasad', 'parasad57harish@gmail.com','Hari prasad 35'),
(4, 'Bhuvaneshkumar', 'buvaneshbs999@gmail.com','Bhuvaneshkumar 20'),
(5, 'Gnanesh', 'gnaneshg1002@gmail.com','Gnanesh 31');
```

Output

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

### Users

roll_number	username	email
1	Harishvaran	romanharish465@gmail.com
2	Guru pradeep	gurupratheep98@gmail.com
3	Hari prasad	parasad57harish@gmail.com
4	Bhuvaneshkumar	buvaneshbs999@gmail.com
5	Gnanesh	gnaneshg1002@gmail.com

**UPDATE Statement:**

UPDATE users4 SET username = 'Harishvaran' WHERE roll\_number = '2';

<

Input

Run SQL

```
CREATE TABLE users (  
  roll_number int, username varchar(300), email varchar(300),  
  password varchar(300)  
);  
INSERT INTO users( roll_number, username ,email, password) VALUES (1,  
'Harishvaran', 'romanharish465@gmail.com','Harishvaran 36'),  
(2, 'Guru pradeep', 'gurupratheep98@gmail.com','Guru pradeep 33'),  
(3, 'Hari prasad', 'parasad57harish@gmail.com','Hari prasad 35'),  
(4, 'Bhuvaneshkumar', 'buvaneshbs999@gmail.com','Bhuvaneshkumar 20'),  
(5, 'Gnanesh', 'gnaneshg1002@gmail.com','Gnanesh 31');  
  
UPDATE users SET username = 'Harishvaran' WHERE roll_number = '2';
```

Output

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

Users

roll_number	username	email
1	Harishvaran	romanharish465@gmail.com
2	Harishvaran	gurupratheep98@gmail.com
3	Hari prasad	parasad57harish@gmail.com
4	Bhuvaneshkumar	buvaneshbs999@gmail.com
5	Gnanesh	gnaneshg1002@gmail.com

## Insert Statement:

```
INSERT INTO users values(6,'ggggg','jarvis@gmail.com','jarvis47');
```

< Input

Run SQL

>

```
CREATE TABLE users (  
  roll_number int, username varchar(300), email varchar(300),  
  password varchar(300)  
);  
INSERT INTO users( roll_number, username ,email, password) VALUES (1,  
'Harishvaran', 'romanharish465@gmail.com', 'Harishvaran 36'),  
(2, 'Guru pradeep', 'gurupratheep98@gmail.com', 'Guru pradeep 33'),  
(3, 'Hari prasad', 'parasad57harish@gmail.com', 'Hari prasad 35'),  
(4, 'Bhuvaneshkumar', 'buvaneshbs999@gmail.com', 'Bhuvaneshkumar 20'),  
(5, 'Gnanesh', 'gnaneshg1002@gmail.com', 'Gnanesh 31');  
  
UPDATE users SET username = 'Harishvaran' WHERE roll_number = '2';  
  
INSERT INTO users values(6,'ggggg','jarvis@gmail.com','jarvis47');
```

Output

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

### Users

roll_number	username	email
1	Harishvaran	romanharish465@gmail.
2	Harishvaran	gurupratheep98@gmail.
3	Hari prasad	parasad57harish@gmail.
4	Bhuvaneshkumar	buvaneshbs999@gmail.c
5	Gnanesh	gnaneshg1002@gmail.cc
6	ggggg	jarvis@gmail.com

### DELETE Statement:

DELETE FROM users6 where roll\_number='6'

<

Input

Run SQL

>

```
INSERT INTO users (roll_number, username, email, password) VALUES (1,
'Harishvaran', 'romanharish465@gmail.com', 'Harishvaran 36'),
(2, 'Guru pradeep', 'gurupratheep98@gmail.com', 'Guru pradeep 33'),
(3, 'Hari prasad', 'parasad57harish@gmail.com', 'Hari prasad 35'),
(4, 'Bhuvaneshkumar', 'buvaneshbs999@gmail.com', 'Bhuvaneshkumar 20'),
(5, 'Gnanesh', 'gnaneshg1002@gmail.com', 'Gnanesh 31');

UPDATE users SET username = 'Harishvaran' WHERE roll_number = '2';

INSERT INTO users values(6, 'ggggg', 'jarvis@gmail.com', 'jarvis47') ;

DELETE FROM users where roll_number='6'
```

Output

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

### Users

roll_number	username	email
1	Harishvaran	romanharish465@gmail.
2	Harishvaran	gurupratheep98@gmail.
3	Hari prasad	parasad57harish@gmail.
4	Bhuvaneshkumar	buvaneshbs999@gmail.c
5	Gnanesh	gnaneshg1002@gmail.cc

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_mysql 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)
```

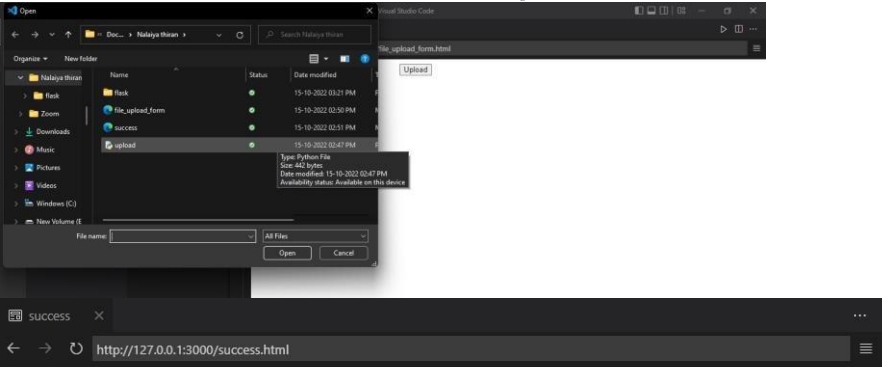
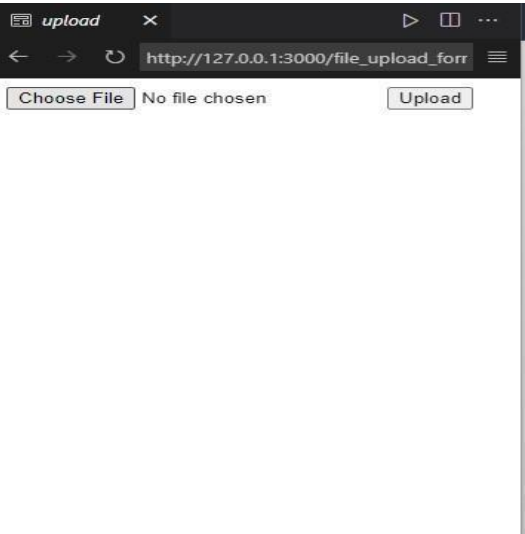
**FILE\_UPLOAD\_FORM.HTML:**

```
<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}}