

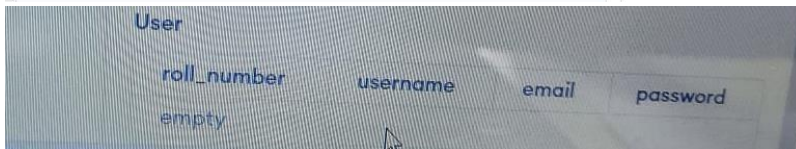
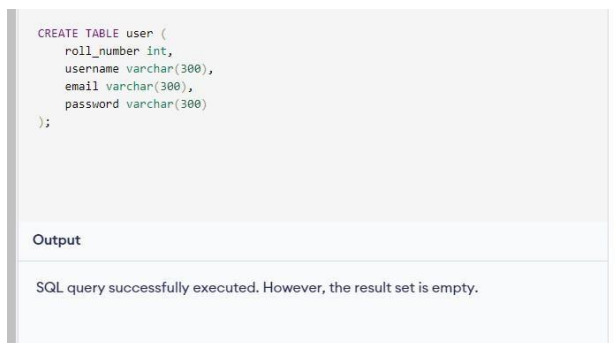
Assignment -2

Assignment Date	19 September 2022
Student Name	J.V. Shijonida
Student Register Number	961819104080
Maximum Marks	2 Marks

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

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

OUTPUT:



2. Perform UPDATE, DELETE Queries with user table

INSERT Statement: INSERT

INTO user

(roll_number, username, email, password) VALUES

(1, 'akshya', 'akshya@gmail.com', 'akshya123'),

(2, 'ashwini', 'ashwini@gmail.com', 'ashwini123'),

(3, 'durga', 'durga@gmail.com', 'durga123'),

(4, 'deekshitha', 'deekshi@gmail.com', 'deekshi123');

OUTPUT:

```
INSERT INTO user
(roll_number, username, email, password) VALUES
(1, 'akshya', 'akshya@gmail.com', 'akshya123'),
(2, 'ashwini', 'ashwini@gmail.com', 'ashwini123'),
(3, 'durga', 'durga@gmail.com', 'durga123'),
(4, 'deekshitha', 'deekshi@gmail.com', 'deekshi123');
```

Output

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

Output		Available Tables	
roll_number	username	email	password
1	akshya	akshya@gmail.com	akshya123
2	ashwini	ashwini@gmail.com	ashwini123
3	durga	durga@gmail.com	durga123
4	deekshi	deekshi@gmail.com	deekshi123

UPDATE Statement:

UPDATE user

SET username = 'deekshi'

WHERE roll_number = '4'

OUTPUT:

```
UPDATE user
SET username = 'deekshi'
WHERE roll_number = '4'
```

Output

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

```
SELECT * FROM Customers;
```

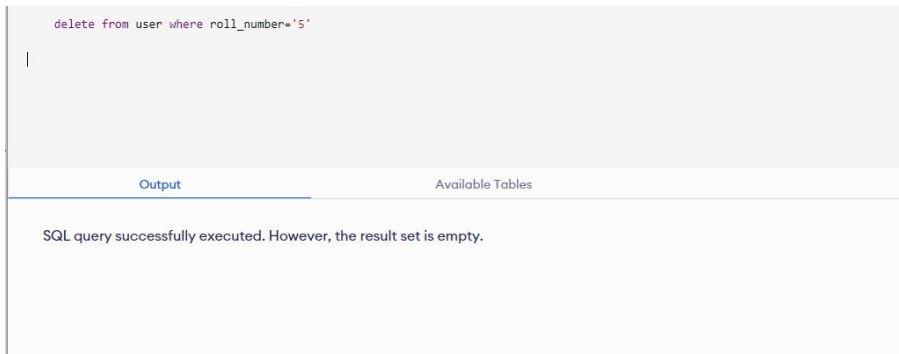
Output		Available Tables		
customer_id	first_name	last_name	age	country
1	John	Doe	31	USA
2	Robert	Luna	22	USA
3	David	Robinson	22	UK
4	John	Reinhardt	25	UK
5	Betty	Doe	28	UAE

DELETE Statemnet:

insert into user values(5,'aa','aaa@gmail.com','aasdfg2') ;

delete from users where roll_number='5'

OUTPUT:



The screenshot shows a SQL query execution interface. At the top, the query `delete from user where roll_number='5'` is entered. Below the query, there are two tabs: "Output" and "Available Tables". The "Available Tables" tab is selected, and it displays a table with 4 rows and 4 columns: roll_number, username, email, and password.

roll_number	username	email	password
1	akshya	akshya@gmail.com	akshya123
2	ashwini	ashwini@gmail.com	ashwini123
3	durga	durga@gmail.com	durga123
4	deekshi	deekshi@gmail.com	deekshi123

3. 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=BsnsG1l2sBgIRhVN",',',')
```

```
from flask import Flask, render_template, request, redirect, url_for, session
from flask_mysql import MySQL
import MySQLdb.cursors
import reapp

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 !'

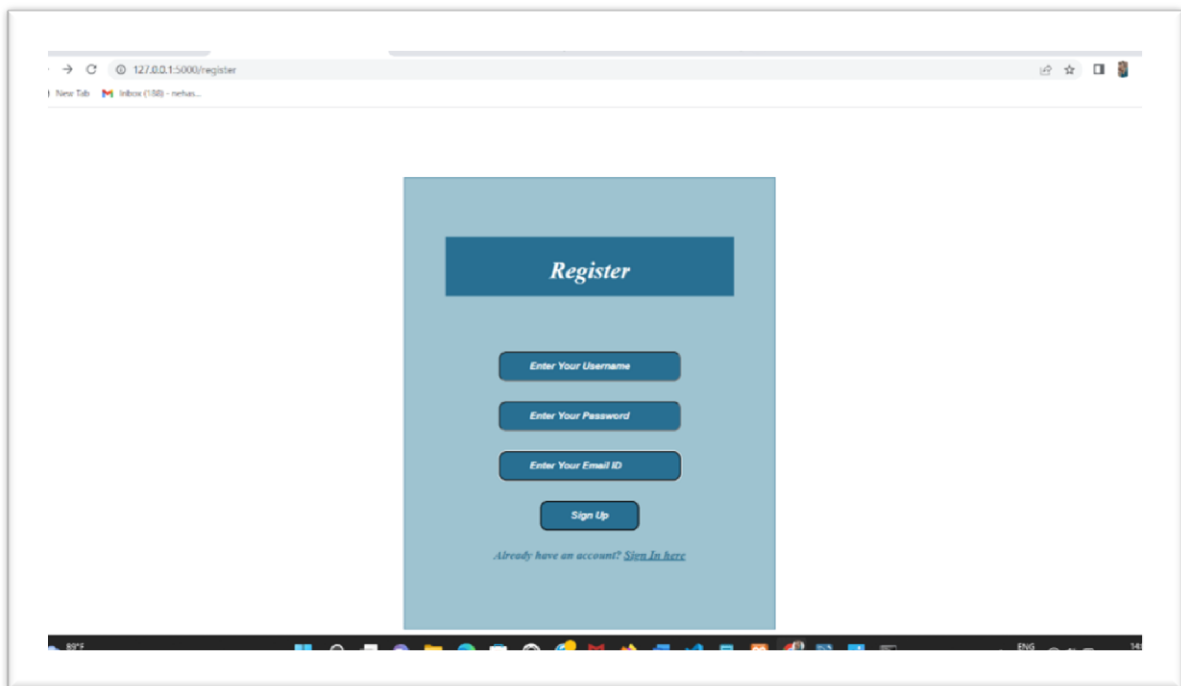
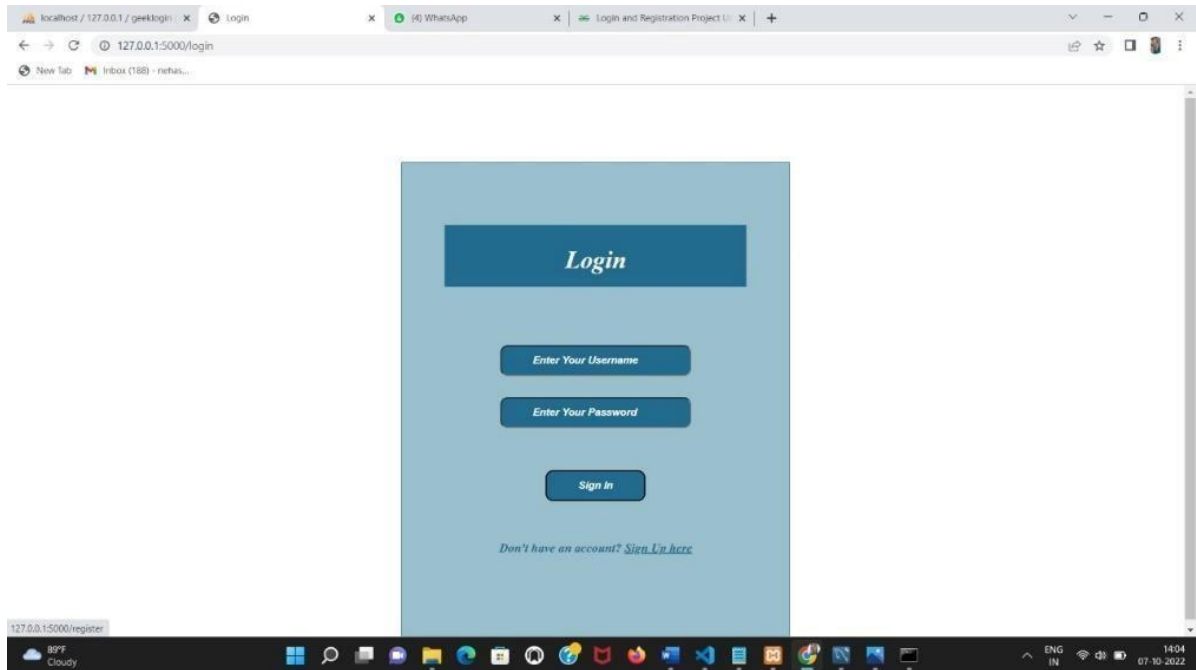
elif request.method == 'POST':

    msg = 'Please fill out the form !'

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

```

OUTPUT:



Index

Hi user!!

Welcome to the index page...

Logout