

## ASSINGMENT -2

Assingment date	19 SEPTEMBER 2022
Student name	POORANI.S
Student roll no	AC19UIT033

- 1.Create user table with user with email,username,password,roll no
2. perform update ,delete queries with user table
- 3.connect python code to database2
- 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 username amnd password.if the user is valid show the welcome page

- 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)  
);
```

User			
roll_number	username	email	password
empty			

2. Perform UPDATE, DELETE Queries with user table

**INSERT Statement:**

INSERT INTO user

( roll\_number, username ,email, password) VALUES

(1, 'Divya', 'divyaraja@gmail.com','Divya15'),

(2, 'Giftlinpriyanka', 'giftlin@gmail.com','gift123'),

(3, 'poorani', 'poorani@gmail.com','poorani123'),

(4, 'rohini', 'rohini@gmail.com', 'rohini123'),

(5, 'Vaishnavi', 'vaishu@gmail.com ', ' vaishnavi123'

User

roll_number	username	email
1	Divya	divyaraja@gmail.com
2	Giftlinpriyanka	giftlin@gmail.com
3	poorani	poorani@gmail.com
4	rohini	rohini@gmail.com
5	Vaishnavi	vaishu@gmail.com

### UPDATE Statement:

UPDATE users

SET username = 'divya'

WHERE roll\_number = '5'

User

roll_number	username	email
1	Divya	divyaraja@gmail.com
2	Giftlinpriyanka	giftlin@gmail.com
3	poorani	poorani@gmail.com
4	rohini	rohini@gmail.com
5	divya	vaishu@gmail.com
5	aa	aaa@gmail.com

### DELETE Statemnet:

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

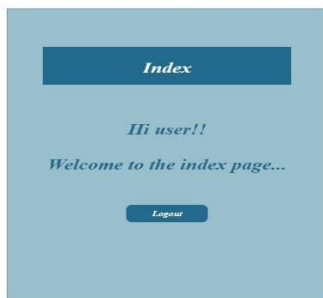
delete from users where roll\_number='5'

#### User

roll_number	username	email
1	Divya	divyaraja@gmail.com
2	Giftlinpriyanka	giftlin@gmail.com
3	poorani	poorani@gmail.com
4	rohini	rohini@gmail.com

### 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",' ');
```



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 username and password. if the user is valid show the welcome page

```
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)
```

