

Assignment -2

Assignment Date	19 September 2022
Student Name	Gopinath K
Student Roll Number	AC19UIT013
Maximum Marks	2 Marks

Question-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)  
);
```

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

Output

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

1. Perform UPDATE, DELETE Queries with user table

INSERT Statement:

INSERT INTO user

(roll_number, username ,email, password) VALUES

(1, 'Raja lingam', 'rajalingam@gmail.com','raju987'),

(2, 'Ajay', 'ajay@gmail.com','ajay654'),

(3, 'Anton', 'anton@gmail.com','anton321'),

(4, 'Prasanth', 'prasanth@gmail.com','prasanth123');

```
INSERT INTO user
( roll_number, username ,email, password) VALUES
(1, 'Raja lingam', 'rajalingam@gmail.com','raju987'),
(2, 'Ajay', 'ajay@gmail.com','ajay654'),
(3, 'Anton', 'anton@gmail.com','anton321'),
(4, 'Prasanth', 'prasanth@gmail.com','prasanth123');
```

Output

Available Tables

5

Delivered

1

User

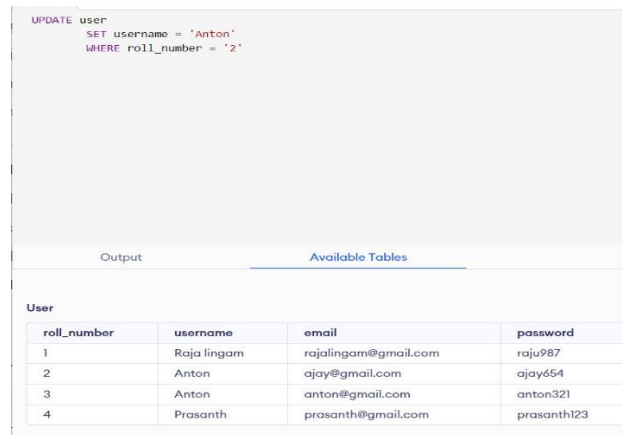
roll_number	username	email	password
1	Raja lingam	rajalingam@gmail.com	raju987
2	Ajay	ajay@gmail.com	ajay654
3	Anton	anton@gmail.com	anton321
4	Prasanth	prasanth@gmail.com	prasanth123

UPDATE Statement:

UPDATE users

SET username = 'Anton'

WHERE roll_number = '2'



The screenshot shows a database interface. At the top, there is a text area containing the SQL statement: `UPDATE user
SET username = 'Anton'
WHERE roll_number = '2'`. Below this, there are two tabs: 'Output' and 'Available Tables'. The 'Available Tables' tab is selected, showing a table named 'User'. The table has four columns: 'roll_number', 'username', 'email', and 'password'. The data in the table is as follows:

roll_number	username	email	password
1	Raja lingam	rajalingham@gmail.com	raju987
2	Anton	ajay@gmail.com	ajay654
3	Anton	anton@gmail.com	anton321
4	Prasanth	prasanth@gmail.com	prasanth123

Insert Statement:

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



The screenshot shows a database interface with a table named 'User'. The table has four columns: 'roll_number', 'username', 'email', and 'password'. The data in the table is as follows:

roll_number	username	email	password
1	Raja lingam	rajalingham@gmail.com	raju987
2	Anton	ajay@gmail.com	ajay654
3	Anton	anton@gmail.com	anton321
4	Prasanth	prasanth@gmail.com	prasanth123
5	aa	aaa@gmail.com	aasdfg2

DELETE Statement:

delete from users where roll_number='5'

User

roll_number	username	email	password
1	Raja lingam	rajalingam@gmail.com	raju987
2	Anton	ajay@gmail.com	ajay654
3	Anton	anton@gmail.com	anton321
4	Prasanth	prasanth@gmail.com	prasanth123

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