Assignment -2

Assignment Date	23 September 2022
Student Name	Rajagnanaguru R
Student Roll no	2019103565
Maximum marks	2 Marks

Question 1:

1) Create User table with email, username, roll number, password.

2) Perform UPDATE, DELETE Queries with user table

Insert command

Table after insertion

```
6  );
7
8  select * from student;
```

History | **Results**

Result set 1 | Details

Filter table   Total:2

EMAILID	USERNAME	ROLLNO	PASSWD
user1@gmail.com	user1	201910xxxx	pass1
user2@gmail.com	user2	201910xxxx	pass2

Update command

```
12
13  update student set passwd='passuser2' where username='user2';
14  select * from student;
```

History | **Results**

Result set 1 | Details

Filter table   Total:2

EMAILID	USERNAME	ROLLNO	PASSWD
user1@gmail.com	user1	201910xxxx	pass1
user2@gmail.com	user2	201910xxxx	passuser2

Delete command

```
15
16  delete from student where emailid='user2@gmail.com';
17  select * from student;
```

History | **Results**

Result set 1 | Details

Filter table   Total:1

EMAILID	USERNAME	ROLLNO	PASSWD
user1@gmail.com	user1	201910xxxx	pass1

3) Connect python code to db2. [connecting via ODBC (open database connection) by sending dsn (data source name)]

```
#Create the dsn connection string
dsn = (
    "DRIVER={0};"
    "DATABASE={1};"
    "HOSTNAME={2};"
    "PORT={3};"
    "PROTOCOL={4};"
    "UID={5};"
    "PWD={6};"
    "SECURITY={7};").format(dsn_driver, dsn_database, dsn_hostname, dsn_port, dsn_protocol, dsn_uid, dsn_pwd,dsn_security)

print(dsn)

## Creating db connection
try:
    conn = ibm_db.connect(dsn, "", "")
    print ("Connected to database: ", dsn_database, "as user: ", dsn_uid, "on host: ", dsn_hostname)

except:
    print ("Unable to connect: ", ibm_db.conn_errormsg())
```

```
DRIVER={IBM DB2 ODBC DRIVER};DATABASE=BLUDB;HOSTNAME=815fa4db-dc03-4c70-869a-a9cc13f33084.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;PORT=30367;PROTOCOL=TCPIP
Connected to database:  BLUDB as user:  rxz06099 on host:  815fa4db-dc03-4c70-869a-a9cc13f33084.bs2io90l08kqb1od8lcg.databases.appdomain.cloud
```

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 with username and password. If the user is valid show the welcome page.

Before insertion:

```
15
16  delete from student where emailid='None';
17  select * from student;
```

History | **Results**

Result set 1 | Details

🔍 Filter table | Total:3 ▽ ⤷

EMAILID	USERNAME	ROLLNO	PASSWD
user1@gmail.com	user1	201910xxxx	pass1
user1@gmail.com	user1	201910xxxx	pass1
raja@gmail.com	user1	2019103565	password987

Form:

Emailid: bharath@gmail.com
Username: Bharath
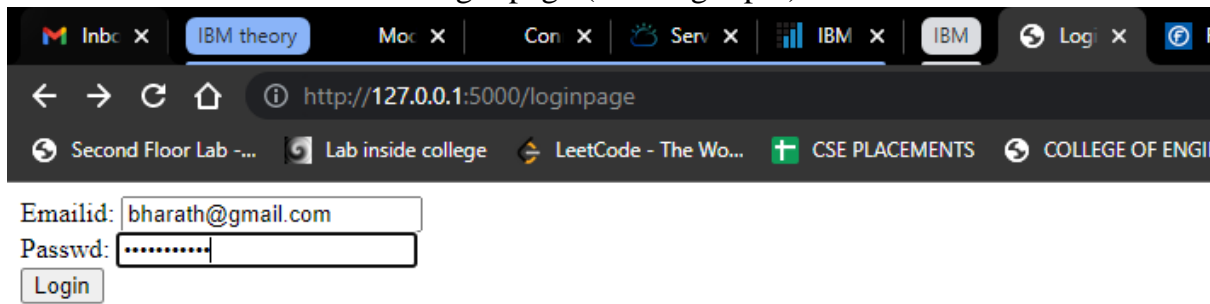Rollno: 2019103010
Passwd: ••••••••••
Submit

Inserted

After data insertion from form (registration)

```
14  select * from student;
15
16  delete from student where emailid='None';
17  select * from student;
```

History	Results	
Result set 1	Details	

Q Filter table   Total:4

EMAILID	USERNAME	ROLLNO	PASSWD
user1@gmail.com	user1	201910xxxx	pass1
bharath@gmail.com	Bharath	2019103010	bharathpass
user1@gmail.com	user1	201910xxxx	pass1
raja@gmail.com	user1	2019103565	password987
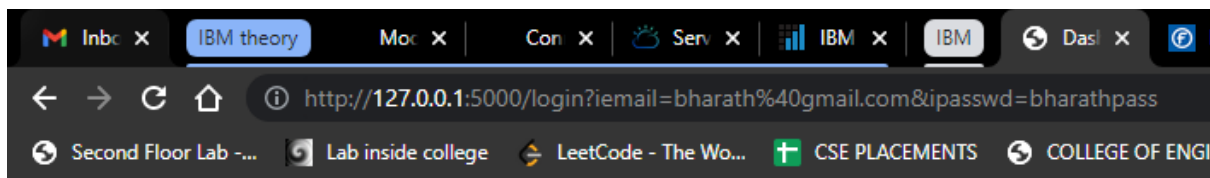
Login page (entering input)

Emailid: bharath@gmail.com
Passwd: ••••••••••
Login

When credentials match

http://127.0.0.1:5000/login?iemail=bharath%40gmail.com&ipasswd=bharathpass
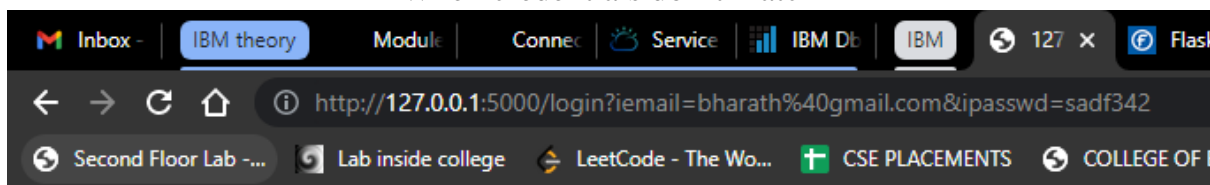
Hello there, this is the dashboard!

When credentials don't match

http://127.0.0.1:5000/login?iemail=bharath%40gmail.com&ipasswd=sadf342

Incorrect credentials!