

# ASSIGNMENT-02

**SANJANA. A**  
**195001097**

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

```
CREATE TABLE User(email VARCHAR(40),  
username VARCHAR(40),  
roll_number BIGINT,  
password VARCHAR(40)  
);
```

The screenshot displays a database management interface. At the top, there are tabs for 'Load Data', 'Load History', 'Tables', 'Views', 'Indexes', 'Aliases', 'MQTs', 'Sequences', and 'Application objects'. The 'Tables' tab is active. Below the tabs is a search bar labeled 'Find schemas or tables' and a 'Refresh' button. The main area is divided into two panels. The left panel, titled 'Tables', shows a table with columns 'Name', 'Schema', and 'Properties'. It contains one entry: 'USER' in schema 'HCQ89801'. The right panel, titled 'Table definition', shows the definition for the 'USER' table. It includes a table with columns 'Name', 'Data type', 'Nullable', 'Length', and 'Scale'. The table definition is as follows:

Name	Data type	Nullable	Length	Scale
EMAIL	VARCHAR	Y	40	0
USERNAME	VARCHAR	Y	40	0
ROLL_NUMBER	BIGINT	Y		0
PASSWORD	VARCHAR	Y	40	0

At the bottom of the right panel, there is a 'View data' button.

```
INSERT INTO User (email, username, roll_number, password)
VALUES ('ranj@gmail.com','Ranji02', 195001089, 'Ranji@123');
```

```
INSERT INTO User (email, username, roll_number, password)
VALUES ('pooj@gmail.com','Pooj01', 195001076, 'Pooj@123');
```

```
INSERT INTO User (email, username, roll_number, password)
VALUES ('sanj@gmail.com','Sanj01', 195001097, 'Sanj@123');
```

```
INSERT INTO User (email, username, roll_number, password)
VALUES ('varsh@gmail.com','Varsh01', 19500112, 'Varsh@123');
```

1SELECT \* FROM User;

History

Results

Result set 1

Details

Filter table

Total:4

EMAIL	USERNAME	ROLL_NUMBER	PASSWORD
ranj@gmail.com	Ranji02	195001089	Ranji@123
pooj@gmail.com	Pooj01	195001076	Pooj@123
sanj@gmail.com	Sanj01	195001097	Sanj@123
varsh@gmail.com	Varsh01	19500112	Varsh@123

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

```
1
2 UPDATE User
3 SET roll_number = 195001115
4 WHERE username = 'Varsh01';
5
6 SELECT * FROM User;
```

History			
Results			
Result set 1			
Details			
Filter table			
Total:4			
EMAIL	USERNAME	ROLL_NUMBER	PASSWORD
ranj@gmail.com	Ranji02	195001089	Ranji@123
pooj@gmail.com	Pooj01	195001076	Pooj@123
sanj@gmail.com	Sanj01	195001097	Sanj@123
varsh@gmail.com	Varsh01	195001115	Varsh@123

```
1
2 UPDATE User
3 SET password = 'Qwerty@123'
4 WHERE email = 'pooj@gmail.com';
5
6 SELECT * FROM User;
```

History			
Results			
Result set 1			
Details			
Filter table			
Total:4			
EMAIL	USERNAME	ROLL_NUMBER	PASSWORD
ranj@gmail.com	Ranji02	195001089	Ranji@123
pooj@gmail.com	Pooj01	195001076	Qwerty@123
sanj@gmail.com	Sanj01	195001097	Sanj@123
varsh@gmail.com	Varsh01	195001115	Varsh@123

The screenshot shows a SQL IDE interface. The top toolbar includes icons for file operations, undo/redo, code formatting, and a 'Run all' button. The SQL editor contains the following code:

```
1 DELETE FROM User
2 WHERE roll_number = 195001089;
3
4
5 SELECT * FROM User;
```

Below the editor, the 'Results' tab is active, displaying 'Result set 1'. The results are shown in a table with 4 columns: EMAIL, USERNAME, ROLL\_NUMBER, and PASSWORD. There are 3 rows of data.

EMAIL	USERNAME	ROLL_NUMBER	PASSWORD
pooj@gmail.com	Pooj01	195001076	Qwerty@123
sanj@gmail.com	Sanj01	195001097	Sanj@123
varsh@gmail.com	Varsh01	195001115	Varsh@123

### 3. Connect python code to db2.

**Main.py:**

```
import ibm_db

try:
    ibm_db.connect("DATABASE=bludb;HOSTNAME=19af6446-6171-4641-8aba-
9dcff8e1b6ff.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30699;SECURIT
Y=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=hcq89801;PWD=usAygURNq
a4m1FSR", '', '')
    print("Connected")
except:
    print("Not Connected")
```

**OUTPUT:**

The screenshot shows a Windows command prompt window. The user has executed the command `python main.py` in the directory `C:\Users\SSN\OneDrive\Documents\123\IBM\Assignment 2`. The output shows the command being executed and the response 'Connected'.

```
Microsoft Windows [Version 10.0.22000.856]
(c) Microsoft Corporation. All rights reserved.

C:\Users\SSN\OneDrive\Documents\123\IBM\Assignment 2>python main.py
Connected

C:\Users\SSN\OneDrive\Documents\123\IBM\Assignment 2>
```

**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.**

## Register Page

Email

Username

RollNumber

Password

Register

[Already have an account? Log In](#)

## Register Page

**Register**

[Already have an account? Log In](#)

## Login Page

**Login**

[Do not have an account? Sign Up?](#)

# Login Page

***Login***

[Do not have an account? Sign Up?](#)

© 127.0.0.1:5000/welcome

>>>Welcome Page<<<