

ASSIGNMENT -2

FLASK APPLICATION INTEGRATION WITH DB2

Assignment Date	26 September 2022
Student Name	Samreen Ayesha. S
Student Roll Number	2019103055
Maximum Marks	2 Marks

Question-1:

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

The screenshot shows the IBM Db2 on Cloud console. On the left, the 'Data objects' pane shows a database named 'DDB22023'. The main area is titled 'USER TABLE' and contains a SQL script to create a table named 'user'. The script is as follows:

```
1 CREATE TABLE user(  
2     email varchar(255) NOT NULL,  
3     username varchar(255) NOT NULL UNIQUE,  
4     rollno varchar(255) NOT NULL PRIMARY KEY,  
5     passwd varchar(255) NOT NULL  
6 );
```

Below the script, the 'History' tab is active, showing a table with columns: Script, Date, Status, and Runtime. It lists two successful execution records for the 'CREATE TABLE user' script.

Script	Date	Status	Runtime
USER TABLE	Sep 26, 2022 5:23:31 PM	✓ 1	0.125 s
CREATE TABLE user(email varchar(255) NOT NULL, username...		✓	0.125 s

The screenshot shows the IBM Db2 on Cloud console with the 'Tables' view selected. The 'Schemas' pane on the left shows a schema named 'DDB22023' of type 'User' with 1 table. The 'Tables' pane on the right shows a table named 'USER' in the 'DDB22023' schema. The 'Properties' column for the 'USER' table is empty.

Name	Type	Tables
✓ DDB22023	User	1

Name	Schema	Properties
USER	DDB22023	...

Question-2:

Perform UPDATE, DELETE queries with user table.

INSERT:

The screenshot shows the IBM Db2 on Cloud interface. On the left, the 'Data objects' pane shows a database named 'DDB22023'. The main editor shows the following SQL script:

```
1 CREATE TABLE user(  
2   email varchar(255) NOT NULL,  
3   username varchar(255) NOT NULL UNIQUE,  
4   rollno varchar(255) NOT NULL PRIMARY KEY,  
5   passwd varchar(255) NOT NULL  
6 );  
7  
8 INSERT INTO user VALUES('samreen123@gmail.com','Samreen','2019103055','sunshine');  
9 SELECT * FROM user;
```

The 'Results' pane shows 'Result set 1' with the following data:

EMAIL	USERNAME	ROLLNO	PASSWRD
samreen123@gmail.com	Samreen	2019103055	sunshine

The screenshot shows the IBM Db2 on Cloud interface with the following SQL script:

```
7  
8 INSERT INTO user VALUES('samreen123@gmail.com','Samreen','2019103055','sunshine');  
9 SELECT * FROM user;  
10  
11 INSERT INTO user VALUES('sumaiya456@gmail.com','Sumaiya','2019103059','harrypotter');  
12 SELECT * FROM user;
```

The 'Results' pane shows 'Result set 1' with the following data:

EMAIL	USERNAME	ROLLNO	PASSWRD
samreen123@gmail.com	Samreen	2019103055	sunshine
sumaiya456@gmail.com	Sumaiya	2019103059	harrypotter

UPDATE:

IBM Db2 on Cloud

* USER TABLE

BetaClassic

SQL

10

11

12

13

14

15

INSERT INTO user VALUES('sumaiya456@gmail.com','Sumaiya','2019103059','harrypotter');

SELECT * FROM user;

UPDATE user SET PASSWRD='book' WHERE USERNAME='Sumaiya';

SELECT * FROM user;

HistoryResults

Result set 1Details

Filter table

Total:2

EMAIL	USERNAME	ROLLNO	PASSWRD
samreen123@gmail.com	Samreen	2019103055	sunshine
sumaiya456@gmail.com	Sumaiya	2019103059	book

DELETE:

IBM Db2 on Cloud

* USER TABLE

BetaClassic

SQL

13

14

15

16

17

18

UPDATE user SET PASSWRD='book' WHERE USERNAME='Sumaiya';

SELECT * FROM user;

DELETE FROM user WHERE ROLLNO='2019103059';

SELECT * FROM user;

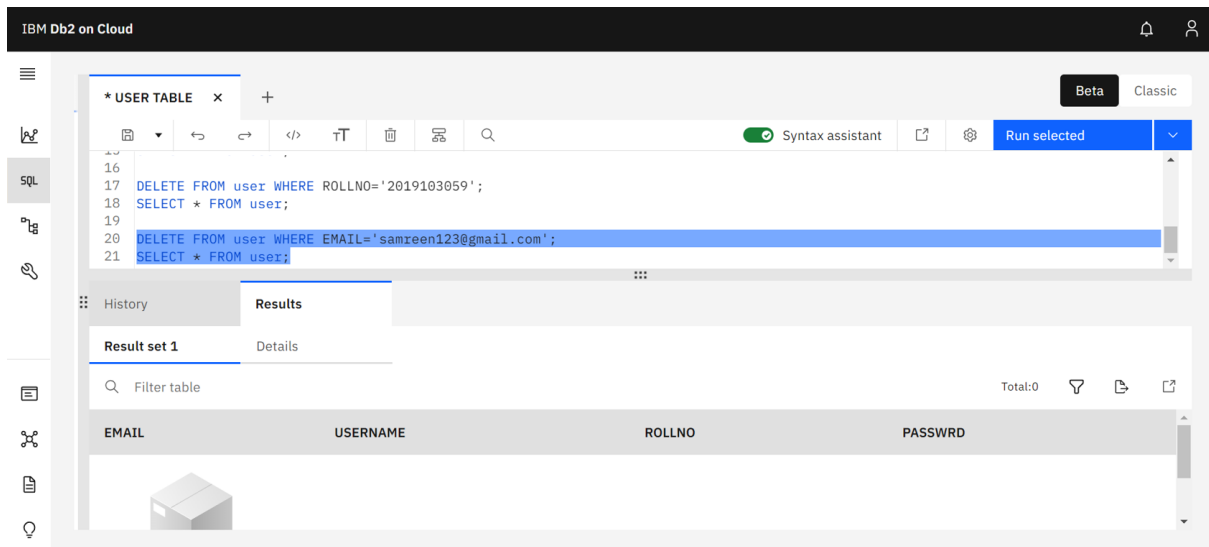
HistoryResults

Result set 1Details

Filter table

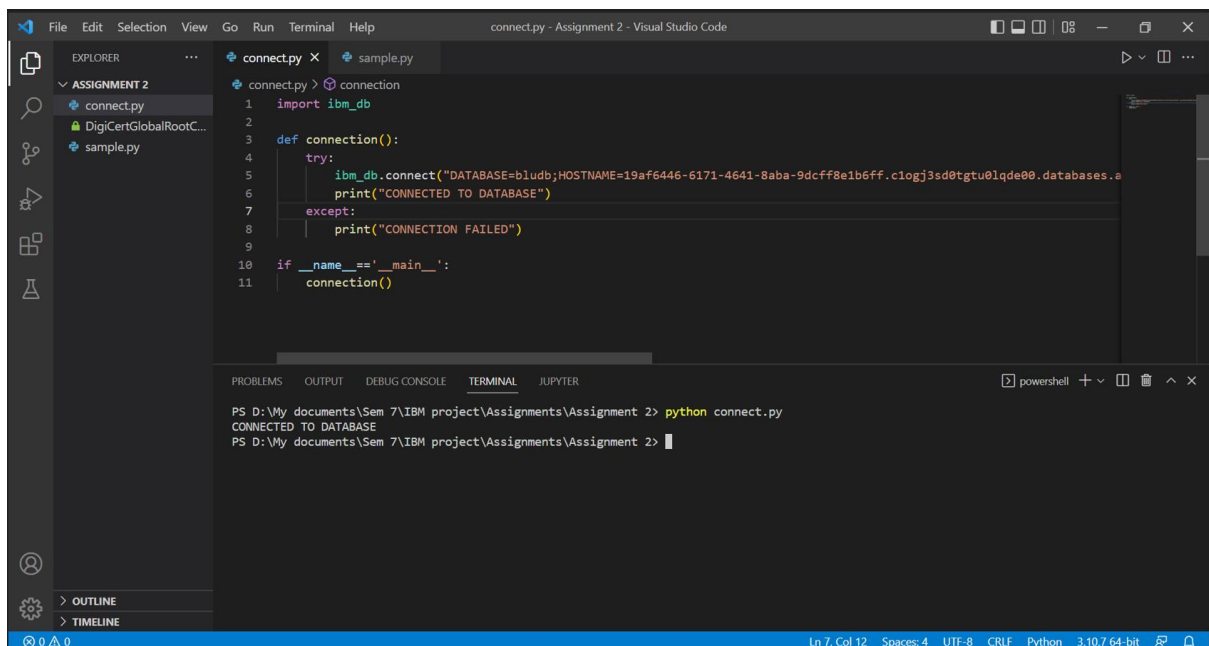
Total:1

EMAIL	USERNAME	ROLLNO	PASSWRD
samreen123@gmail.com	Samreen	2019103055	sunshine



Question-3:

Connect python code to db2.



Syntax:

```

ibm_db.connect("DATABASE=bludb;HOSTNAME=19af6446-6171-4641-8aba-
9dcff8e1b6ff.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30699;PROTOCOL=TCPIP;UI
D=ddb22023;PWD=pm6CGacdO7nzi3pU;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.c
rt;", "", "")

```

Question-4:

Create a flask app with registration page, login page and welcome page. By default, the registration page loads. Once the user enters all the fields store the data in database and navigate to login page to authenticate the user. If user is valid show welcome page

Program:

App.py:

```
app.py > ...
1  from flask import Flask,render_template,request,flash
2  import ibm_db
3
4  app = Flask(__name__)
5
6  def connection():
7      try:
8          conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=19af6446-6171-4641-8aba-9dcff8e1b6ff.clogj3sd0tgtu0lqde00.database
9              print("CONNECTED TO DATABASE")
10             return conn
11         except:
12             print("CONNECTION FAILED")
13
14 @app.route("/",methods=["GET","POST"])
15 def registerPage():
16     if request.method=="POST":
17         conn=connection()
18         try:
19             sql="INSERT INTO USER VALUES('{}','{}','{}','{}').format(request.form["uemail"],request.form["uname"],re
20                 ibm_db.exec_immediate(conn,sql)
21                 flash("Successfully Registered")
22                 return render_template('login.html')
23             except:
24                 flash("Account already exists")
25                 return render_template('register.html')
26         else:
27             return render_template('register.html')
28
29 @app.route("/login",methods=["GET","POST"])
30 def loginPage():
31     if request.method=="POST":
32         conn=connection()
33         username=request.form["luname"]
34         password=request.form["lupass"]
35         sql="SELECT COUNT(*) FROM USER WHERE USERNAME=? AND PASSWRD=?"
36         stmt=ibm_db.prepare(conn,sql)
37         ibm_db.bind_param(stmt,1,username)
38         ibm_db.bind_param(stmt,2,password)
39         ibm_db.execute(stmt)
40         res=ibm_db.fetch_assoc(stmt)
41         if res['1']==1:
42             return render_template("Welcome.html")
43         else:
44             flash("Wrong Username or Password")
45             return render_template('login.html')
46     else:
47         return render_template('login.html')
48
49
50 if __name__=='__main__':
51     app.config['SECRET_KEY']='super secret key'
52     app.config['SESSION_TYPE']='filesystem'
53     app.run(debug=True)
```

TEMPLATES FOLDER:

Register.html:

```
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <title>Register Page</title>
5      <link rel="stylesheet" href="../../static/styles.css">
6      <script>
7        function loadLoginPage(){
8          window.location.assign("http://localhost:5000/login");
9        }
10     </script>
11   </head>
12   <body>
13     <div id="Header"><h2>REGISTRATION PORTAL</h2></div>
14     <form action="http://localhost:5000/" method="post">
15       <div id="frm">
16         <label for="ele1">Email ID:</label>
17         <input type="email" name="uemail" id="ele1"><br><br>
18
19         <label for="ele2">User Name:</label>
20         <input type="text" name="uname" id="ele2"><br><br>
21
22         <label for="ele3">Password:</label>
23         <input type="password" name="upass" id="ele3"><br><br>
24
25         <label for="ele4">Roll Number:</label>
26         <input type="text" name="urollno" id="ele4"><br><br>
27
28         <div id="button"><input type="submit" value="Submit"></div>
29         <button type="button" onclick="loadLoginPage();">LOGIN</button>
30       </div>
31     </form>
32   </body>
33 </html>
```

Login.html:

```
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <title>Login Page</title>
5      <link rel="stylesheet" href="../../static/styles.css">
6    </head>
7    <body>
8      <div id="Header"><h2>LOGIN PORTAL</h2></div>
9      <form action="http://localhost:5000/login" method="post">
10        <div id="frm">
11          <label for="ele1">User Name:</label>
12          <input type="text" name="luname" id="ele1"><br><br>
13
14          <label for="ele2">Password:</label>
15          <input type="password" name="lupass" id="ele2"><br><br>
16          <div id="button"><input type="submit" value="Submit"></div>
17        </div>
18      </form>
19    </body>
20 </html>
```

Welcome.html:

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Welcome Page</title>
5    </head>
6    <body>
7      <p>Authentication Successful</p>
8      
9    </body>
10 </html>
```

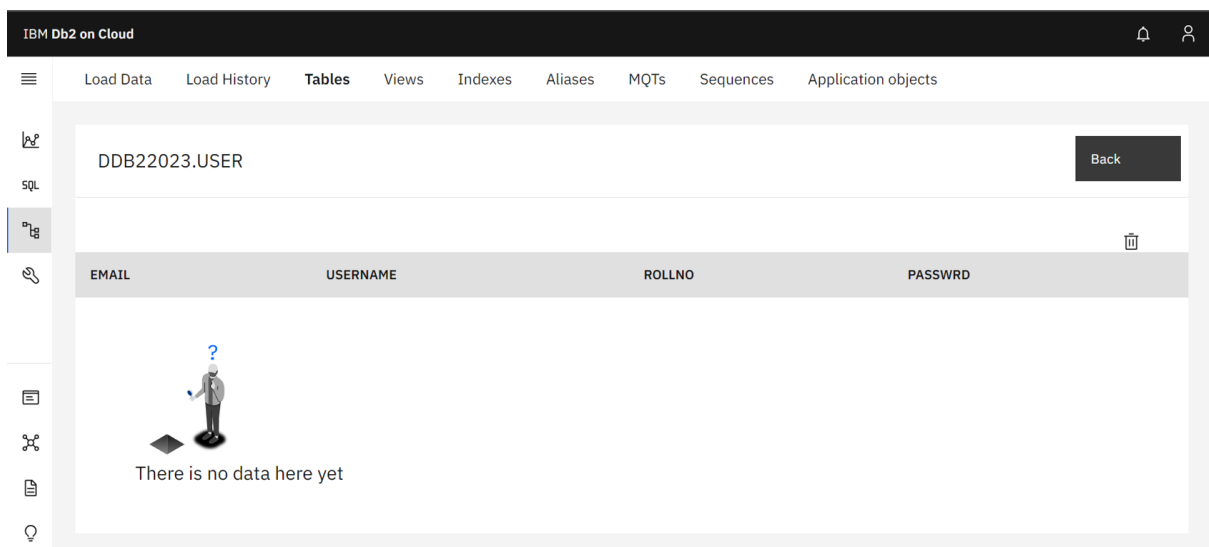
STATIC FOLDER:

Styles.css:

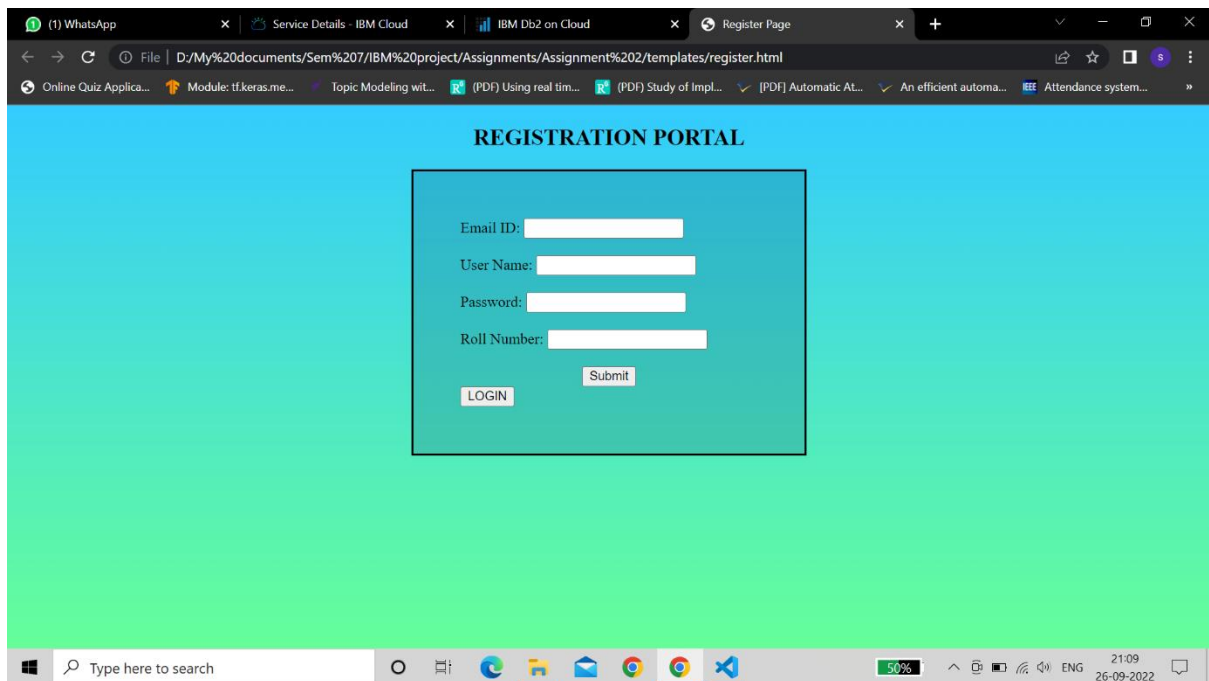
```
1  html{
2      min-height: 100%;
3      min-width: 100%;
4      background: linear-gradient(to bottom, #33ccff 0%, #66ff99 100%);
5  }
6  #frm{
7      background: rgba(5, 87, 120, 0.24);
8      color: black;
9      max-width: 25%;
10     margin: auto;
11     padding: 50px;
12     border-style: solid;
13     border-width: 2px;
14     border-color: black;
15 }
16 #Header{
17     text-align: center;
18 }
19 #button{
20     display: flex;
21     justify-content: center;
22     align-items: center;
23 }
```

EXECUTION:

The user table is initially empty



The registration page



The screenshot shows a web browser window with the title "Register Page". The address bar displays the file path: "D:/My%20documents/Sem%207/IBM%20project/Assignments/Assignment%202/templates/register.html". The browser's tab bar shows several open tabs, including "WhatsApp", "Service Details - IBM Cloud", "IBM Db2 on Cloud", and "Register Page". The main content area has a light blue background with the heading "REGISTRATION PORTAL" in bold. Below the heading is a white rectangular form with a black border. Inside the form, there are four input fields: "Email ID:", "User Name:", "Password:", and "Roll Number:". Below these fields are two buttons: "LOGIN" and "Submit". The Windows taskbar at the bottom shows the search bar, task view icon, and several application icons. The system tray on the right indicates the time as 21:09 and the date as 26-09-2022.

REGISTRATION PORTAL

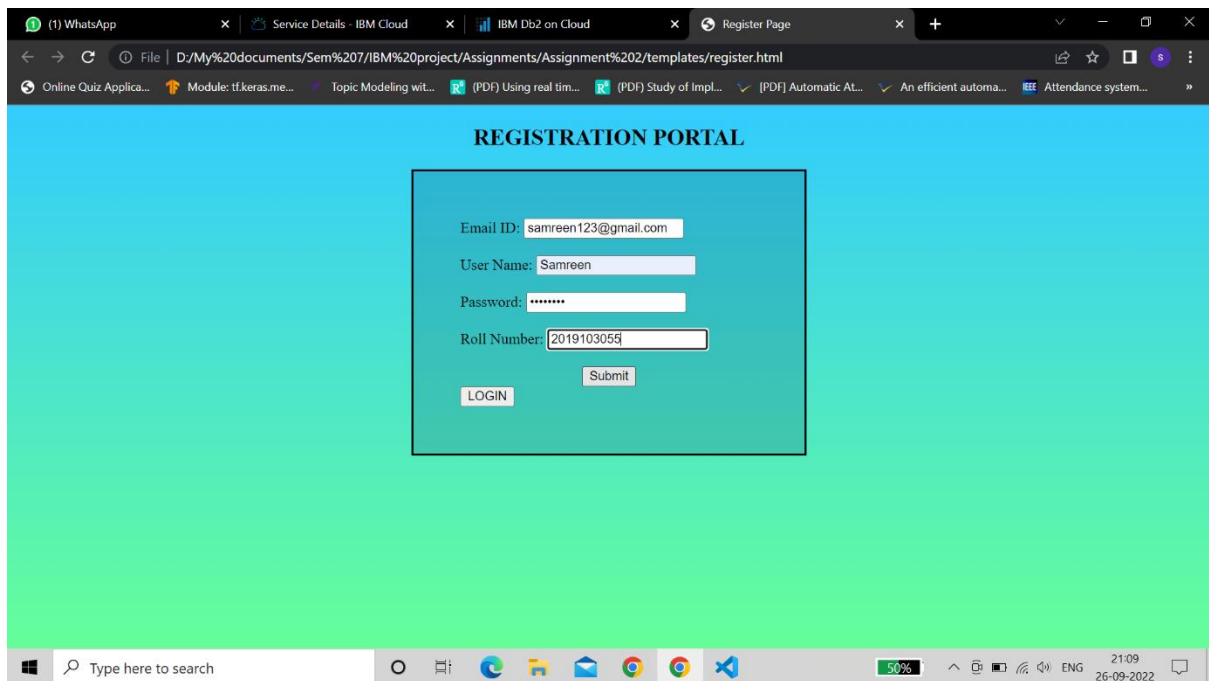
Email ID:

User Name:

Password:

Roll Number:

The registration details are filled and on submitting, the data is entered into the database and it navigates to the login page for authentication.



This screenshot shows the same "REGISTRATION PORTAL" form as the previous one, but with the registration details filled in. The "Email ID" field contains "samreen123@gmail.com", the "User Name" field contains "Samreen", the "Password" field contains "*****", and the "Roll Number" field contains "2019103055". The "LOGIN" and "Submit" buttons remain at the bottom of the form. The browser window and Windows taskbar are identical to the previous screenshot.

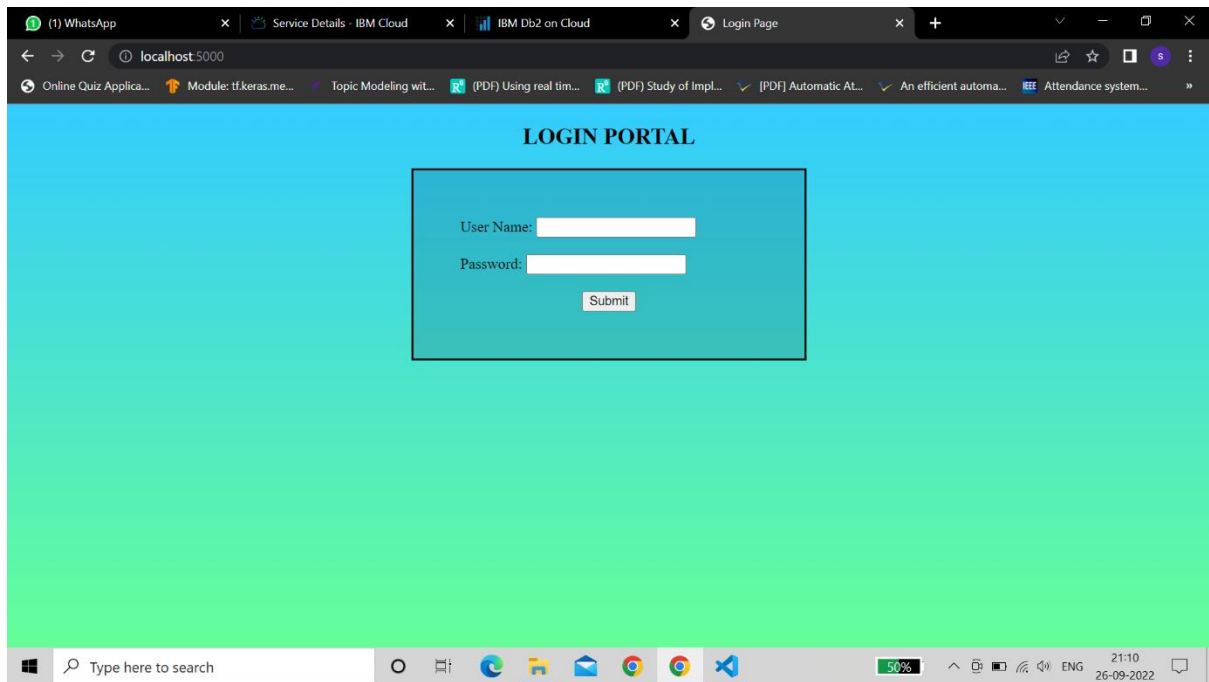
REGISTRATION PORTAL

Email ID:

User Name:

Password:

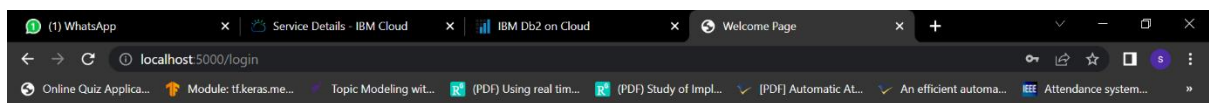
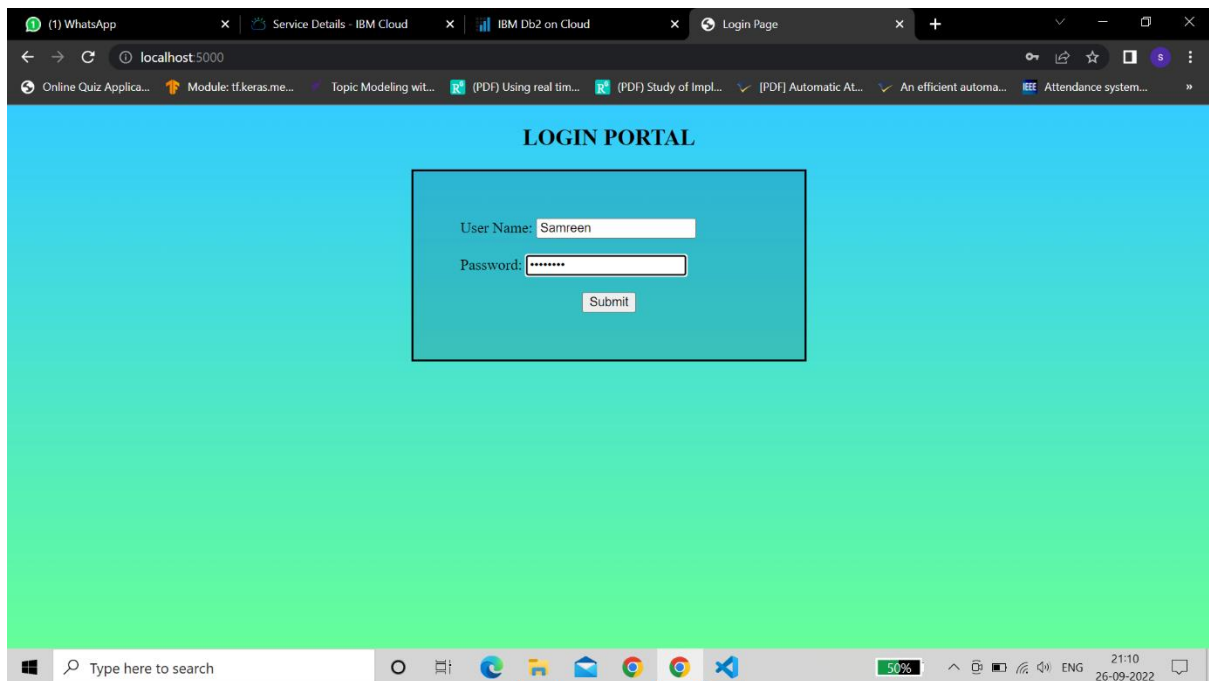
Roll Number:



The inserted data in the database

IBM Db2 on Cloud					
Load Data Load History Tables Views Indexes Aliases MQTs Sequences Application objects					
DDB22023.USER					Back
					Export to CSV
EMAIL	USERNAME	ROLLNO	PASSWRD		
samreen123@gmail.com	Samreen	2019103055	sunshine		

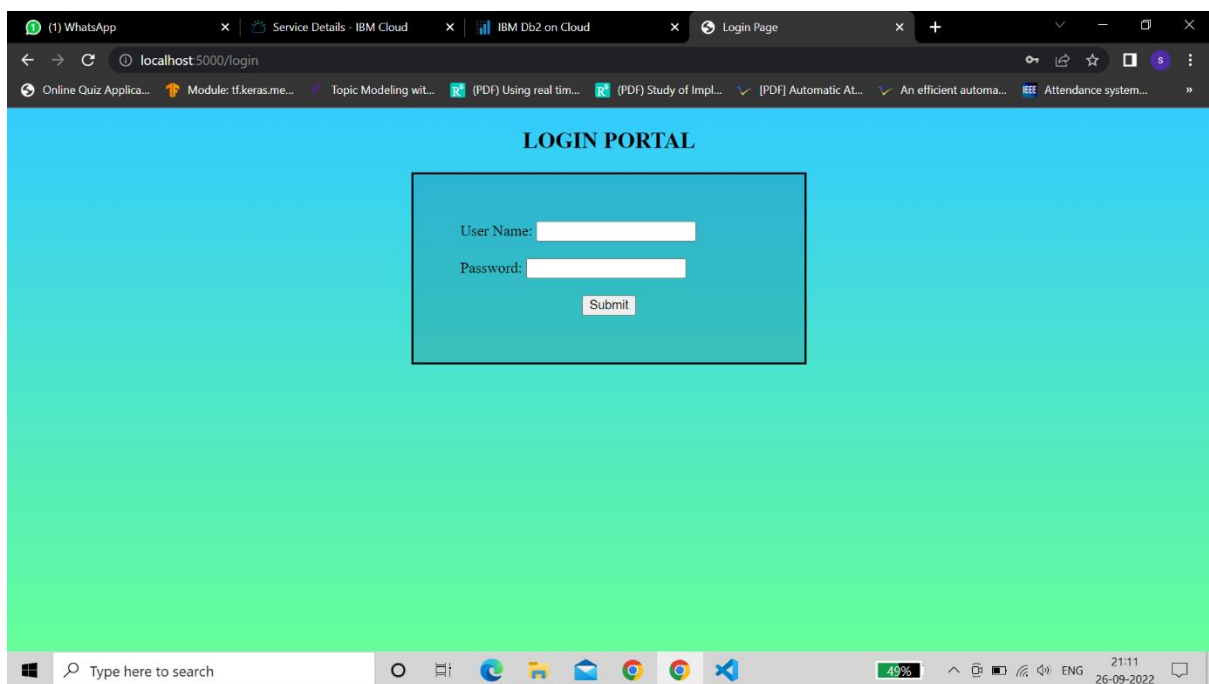
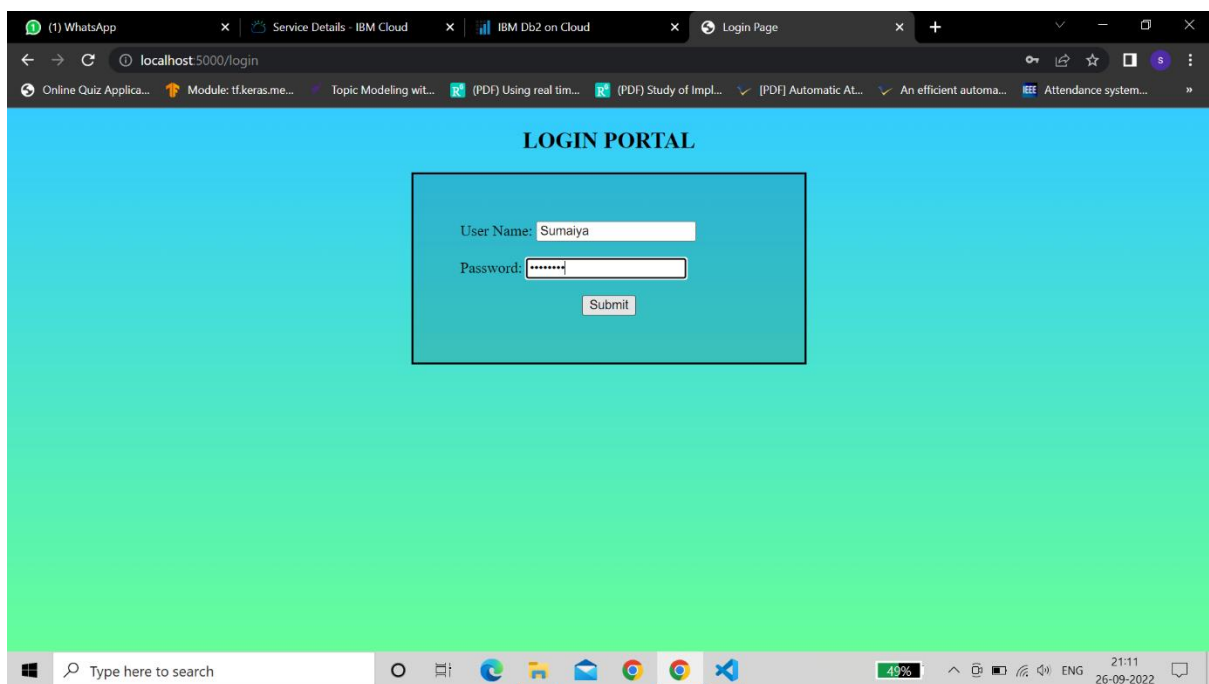
On entering the correct login details in the login page it is redirected to the welcome page



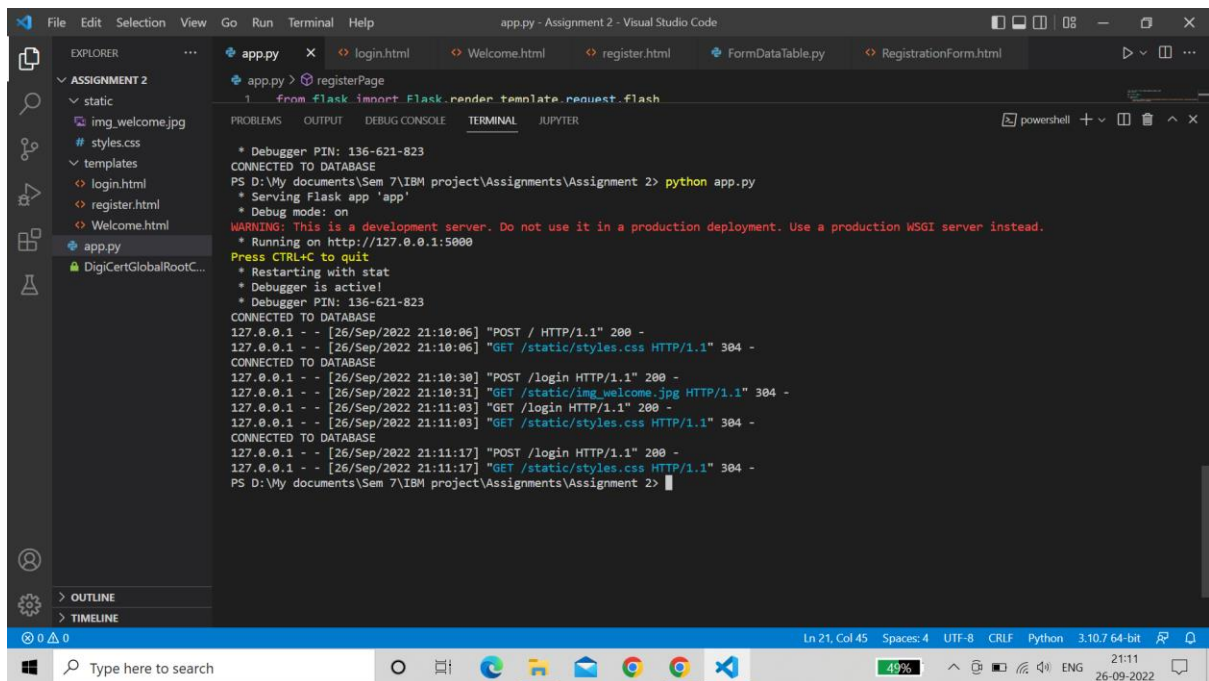
Authentication Successful



Incase wrong login details are entered it stays in the login page



The server terminal:



The screenshot shows a Visual Studio Code window with a terminal running a Flask application. The terminal output includes the following text:

```
app.py x login.html Welcome.html register.html FormDataTable.py RegistrationForm.html
1 from flask import Flask, render_template, request, flash

* Debugger PIN: 136-621-823
CONNECTED TO DATABASE
PS D:\My documents\Sem 7\IBM project\Assignments\Assignment 2> python app.py
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 136-621-823
CONNECTED TO DATABASE
127.0.0.1 - - [26/Sep/2022 21:10:06] "POST / HTTP/1.1" 200 -
127.0.0.1 - - [26/Sep/2022 21:10:06] "GET /static/styles.css HTTP/1.1" 304 -
CONNECTED TO DATABASE
127.0.0.1 - - [26/Sep/2022 21:10:30] "POST /login HTTP/1.1" 200 -
127.0.0.1 - - [26/Sep/2022 21:10:31] "GET /static/img_welcome.jpg HTTP/1.1" 304 -
127.0.0.1 - - [26/Sep/2022 21:11:03] "GET /login HTTP/1.1" 200 -
127.0.0.1 - - [26/Sep/2022 21:11:03] "GET /static/styles.css HTTP/1.1" 304 -
CONNECTED TO DATABASE
127.0.0.1 - - [26/Sep/2022 21:11:17] "POST /login HTTP/1.1" 200 -
127.0.0.1 - - [26/Sep/2022 21:11:17] "GET /static/styles.css HTTP/1.1" 304 -
PS D:\My documents\Sem 7\IBM project\Assignments\Assignment 2>
```

The Visual Studio Code interface shows the Explorer sidebar with a project named 'ASSIGNMENT 2' containing files like 'static', 'img_welcome.jpg', 'styles.css', 'templates', 'login.html', 'register.html', 'Welcome.html', and 'app.py'. The bottom status bar indicates the current file is 'app.py' at line 21, column 45, with a Python 3.10.7 64-bit interpreter.