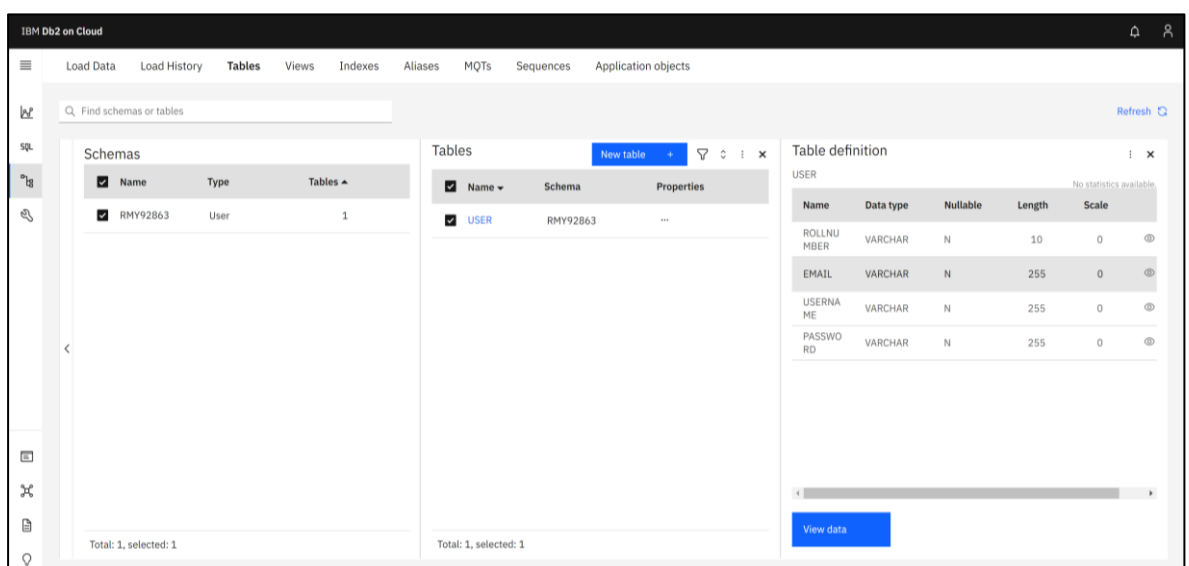
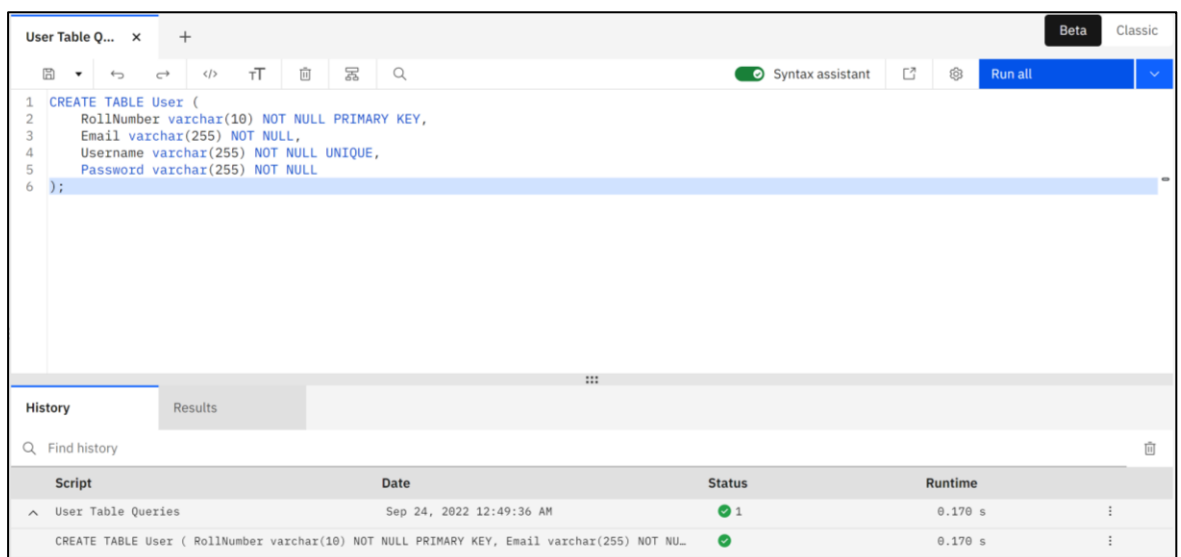


ASSIGNMENT - 2

FLASK APPLICATION INTEGRATION WITH DB2

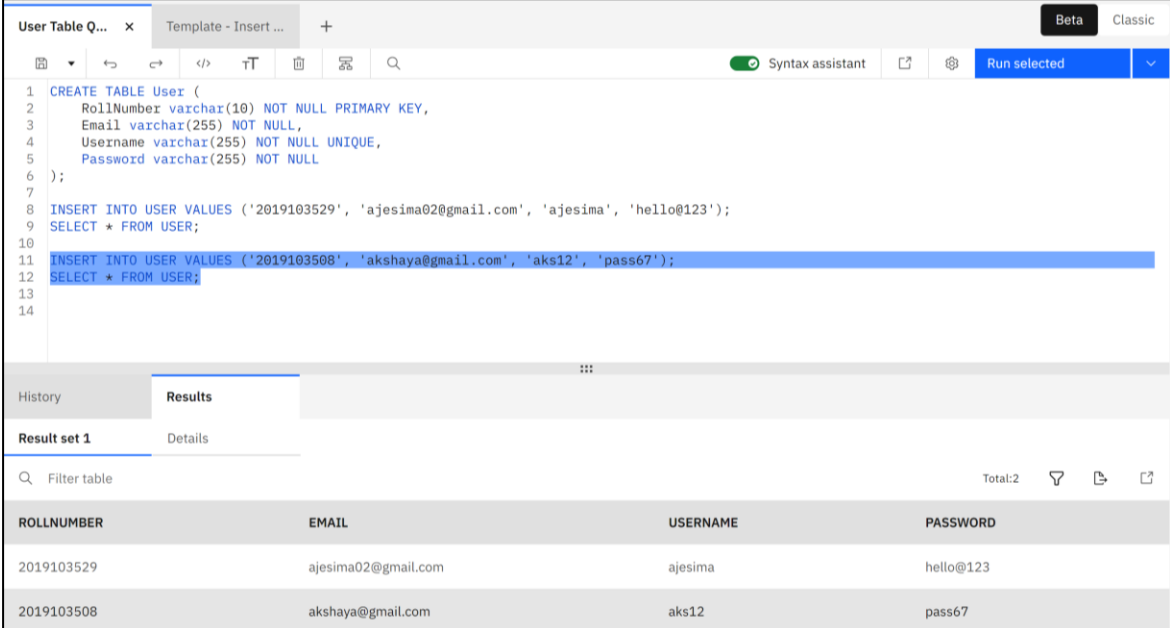
Assignment Date	24 September 2022
Student Name	Jesima A
Student Roll Number	2019103529
Maximum Marks	2 Marks

1. Create a user table with email, username, password and roll number.



2. Perform UPDATE, and DELETE queries with the User table.

INSERT QUERY



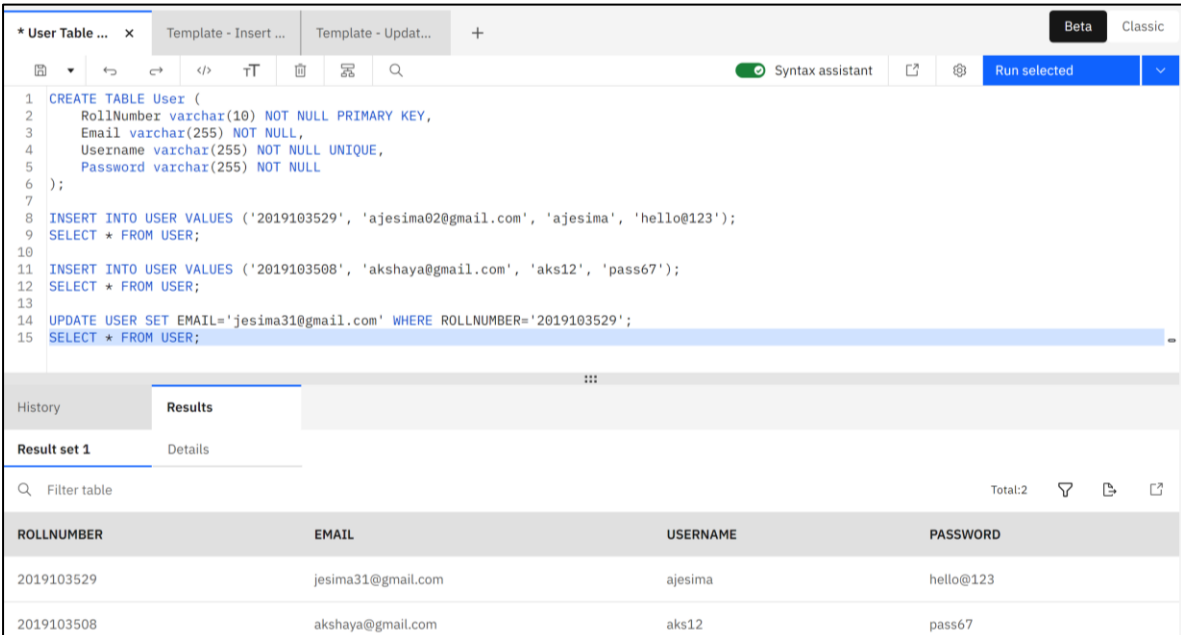
The screenshot shows a SQL IDE interface with a query editor and a results pane. The query editor contains the following SQL code:

```
1 CREATE TABLE User (  
2     RollNumber varchar(10) NOT NULL PRIMARY KEY,  
3     Email varchar(255) NOT NULL,  
4     Username varchar(255) NOT NULL UNIQUE,  
5     Password varchar(255) NOT NULL  
6 );  
7  
8 INSERT INTO USER VALUES ('2019103529', 'ajesima02@gmail.com', 'ajesima', 'hello@123');  
9 SELECT * FROM USER;  
10  
11 INSERT INTO USER VALUES ('2019103508', 'akshaya@gmail.com', 'aks12', 'pass67');  
12 SELECT * FROM USER;  
13  
14
```

The results pane shows the output of the query, displaying two rows of data:

ROLLNUMBER	EMAIL	USERNAME	PASSWORD
2019103529	ajesima02@gmail.com	ajesima	hello@123
2019103508	akshaya@gmail.com	aks12	pass67

UPDATE QUERY



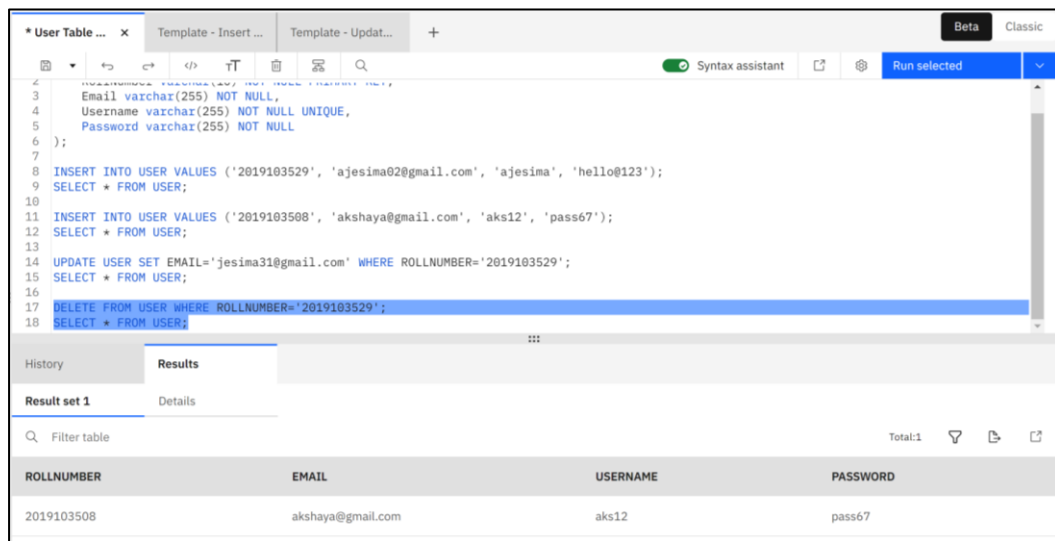
The screenshot shows a SQL IDE interface with a query editor and a results pane. The query editor contains the following SQL code:

```
1 CREATE TABLE User (  
2     RollNumber varchar(10) NOT NULL PRIMARY KEY,  
3     Email varchar(255) NOT NULL,  
4     Username varchar(255) NOT NULL UNIQUE,  
5     Password varchar(255) NOT NULL  
6 );  
7  
8 INSERT INTO USER VALUES ('2019103529', 'ajesima02@gmail.com', 'ajesima', 'hello@123');  
9 SELECT * FROM USER;  
10  
11 INSERT INTO USER VALUES ('2019103508', 'akshaya@gmail.com', 'aks12', 'pass67');  
12 SELECT * FROM USER;  
13  
14 UPDATE USER SET EMAIL='jesima31@gmail.com' WHERE ROLLNUMBER='2019103529';  
15 SELECT * FROM USER;
```

The results pane shows the output of the query, displaying two rows of data:

ROLLNUMBER	EMAIL	USERNAME	PASSWORD
2019103529	jesima31@gmail.com	ajesima	hello@123
2019103508	akshaya@gmail.com	aks12	pass67

DELETE QUERY

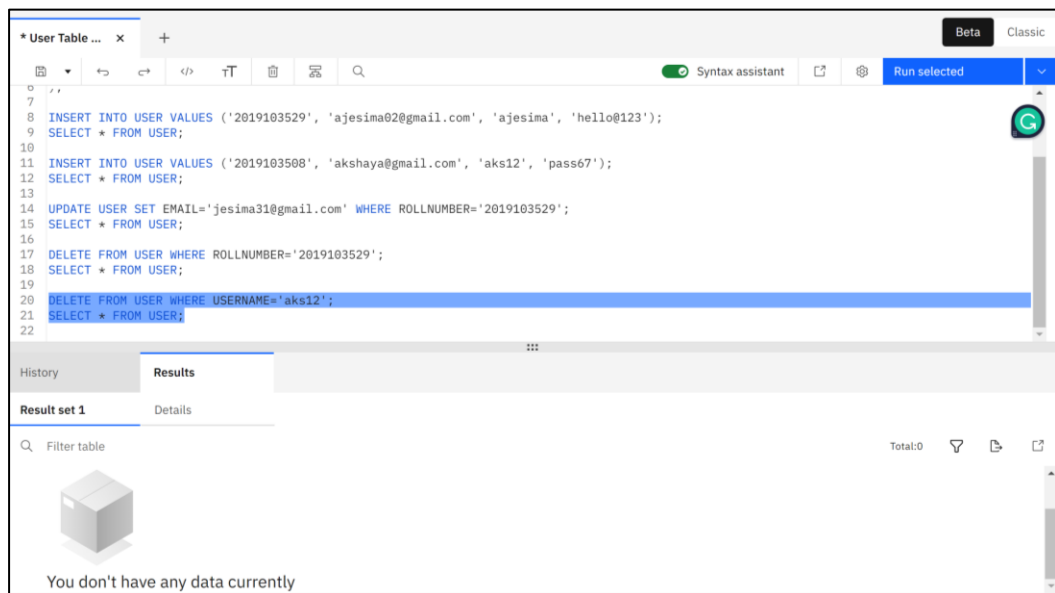


The screenshot shows a database IDE with a SQL query editor and a results pane. The query is as follows:

```
CREATE TABLE USER (
  ROLLNUMBER VARCHAR(255) NOT NULL,
  Email varchar(255) NOT NULL,
  Username varchar(255) NOT NULL UNIQUE,
  Password varchar(255) NOT NULL
);
INSERT INTO USER VALUES ('2019103529', 'ajesima02@gmail.com', 'ajesima', 'hello@123');
SELECT * FROM USER;
INSERT INTO USER VALUES ('2019103508', 'akshaya@gmail.com', 'aks12', 'pass67');
SELECT * FROM USER;
UPDATE USER SET EMAIL='jesima31@gmail.com' WHERE ROLLNUMBER='2019103529';
SELECT * FROM USER;
DELETE FROM USER WHERE ROLLNUMBER='2019103529';
SELECT * FROM USER;
```

The results pane shows a table with the following data:

ROLLNUMBER	EMAIL	USERNAME	PASSWORD
2019103508	akshaya@gmail.com	aks12	pass67



The screenshot shows the same database IDE with the same SQL query, but the results pane is empty, indicating that the data has been deleted.

You don't have any data currently

3. Connect python code to DB2.

```
app.py  U X
app.py > connection
1  import ibm_db
2
3  def connection():
4      try:
5          ibm_db.connect("DATABASE=bludb;HOSTNAME=b70af05b-76e4-4bca-a1f5-23dbb4c6a74e.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;\
6              PORT=32716;PROTOCOL=TCPIP;UID=rmy92863;PWD=goEFGk1TDT55aQbj;SECURITY=SSL;SSLSERVICECERTIFICATE=DigiCertGlobalRootCA.crt", "", "")
7          print("Connected to Database")
8      except:
9          print("Not Connected to Database")
10
11 if __name__ == '__main__':
12     connection()
```



```
app.py U X
app.py > defaultpage
42     res = ibm_db.fetch_assoc(stmt)
43     if res['1'] == 1:
44         session['loggedin']= True
45         session['username'] = username
46         return render_template('welcome.html')
47     else:
48         flash("Username/ Password is incorrect!")
49         return render_template('login.html')
50 else:
51     return render_template('login.html')
52
53 @app.route('/logout')
54 def logout():
55     session.pop('loggedin', None)
56     session.pop('username', None)
57     return redirect(url_for('login'))
58
59 if __name__ == '__main__':
60     app.config['SECRET_KEY'] = 'super secret key'
61     app.config['SESSION_TYPE'] = 'filesystem'
62     app.run()
```

templates/register.html

```
register.html U X
templates > register.html > html > body
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset="UTF-8">
5      <title>Registration</title>
6      <link rel="stylesheet" type="text/css" href= "../static/styles.css" >
7      <script>
8          function loadLoginPage() {
9              window.location.assign("http://localhost:5000/login");
10         }
11     </script>
12 </head> <body>
13     <div class="overallDiv">
14         <h2 class="heading">Registration Page</h2>
15         {% for message in get_flashed_messages() %}
16             <div class="message-flash">
17                 <strong>{{message}}</strong>
18             </div>
19         {% endfor %}
20         <form action="http://localhost:5000/" method="post" id="studentform">
21             <div class="form-control">
22                 <label for="rollno" class="labels">Roll Number</label><br>
23                 <input type="text" id="rollno" name="rollno" maxlength="10" required><br><br>
24             </div>
25             <div class="form-control">
26                 <label for="email" class="labels">E-mail ID</label><br>
27                 <input type="email" id="email" name="email" required><br><br>
28             </div>
29             <div class="form-control">
30                 <label for="username" class="labels">Username</label><br>
31                 <input type="text" id="username" name="username" required><br><br>
32             </div>
33             <div class="form-control">
34                 <label for="password" class="labels">Password</label><br>
35                 <input type="password" id="password" name="password" required><br><br>
36             </div>
37             <input type="submit" value="Sign Up!">
38             <button type="button" onclick="loadLoginPage();" id="loginbutton">Sign In!</button>
39         </form>
40     </div>
41 </body>
42 </html>
```

templates/login.html

```
login.html U X
templates > login.html > html
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta charset="UTF-8">
5 <title>Login Page</title>
6 <link rel="stylesheet" type="text/css" href= "../static/styles.css" >
7 <script>
8     function loadRegisterPage() {
9         window.location.assign("http://localhost:5000/");
10    }
11 </script>
12 </head>
13
14 <body>
15 <div class="overallDiv">
16 <h2 class="heading">Login</h2>
17
18 {% for message in get_flashed_messages() %}
19 <div class="message-flash">
20 <strong>{{message}}</strong>
21 </div>
22 {% endfor %}
23
24 <form action="http://localhost:5000/login" method="post" id="studentform">
25
26 <div class="form-control">
27 <label for="username" class="labels">Username</label><br>
28 <input type="text" id="username" name="username" required><br><br>
29 </div>
30
31 <div class="form-control">
32 <label for="password" class="labels">Password</label><br>
33 <input type="password" id="password" name="password" required><br><br>
34 </div>
35
36 <input type="submit" value="Sign In!">
37 <button type="button" onclick="loadRegisterPage();" id="registerbutton">Sign Up!</button>
38 </form>
39 </div>
40 </body>
41 </html>
```

templates/welcome.html

```
welcome.html U X
templates > welcome.html > html
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta charset="UTF-8">
5 <title>Welcome Page</title>
6 <link rel="stylesheet" type="text/css" href= "../static/styles.css" >
7 <script>
8     function loadLoginPage() {
9         window.location.assign("http://localhost:5000/logout");
10    }
11 </script>
12 </head>
13 <body>
14 <div class="overallDiv">
15 <h2>Welcome, {{session.username}}!</h2>
16 <button type="button" onclick="loadLoginPage();" id="logoutbutton">Logout</button>
17 </div>
18 </body>
19 </html>
```


static/styles.css

```
# styles.css U X
static > # styles.css > ⚡ input[type=submit]
1  body {
2      background-color: #201a55;
3      font-family: 'Tahoma';
4      text-align: center;
5  }
6  h2{
7      font-size: 30px;
8      color: #201a55;
9  }
10 form{
11     max-width: 200px;
12     padding-top: 50px;
13     font-size: 20px;
14     text-align: center;
15 }
16 .form-control{
17     margin-bottom: 5px;
18     text-align: left;
19 }
20 .overallDiv{
21     background-color: #ffffff;
22     border: 5px solid #000;
23     margin-left: 400px;
24     margin-right: 400px;
25     margin-top: 80px;
26     margin-bottom: 5px;
27     padding-left: 50px;
28     padding-right: 50px;
29 }
30 input[type=text], input[type=password], input[type=email] {
31     width: 50%;
32     padding: 12px 20px;
33     margin: 8px 0;
34     box-sizing: border-box;
35     font-size: large;
36 }
```

```
# styles.css U X
static > # styles.css > ⚡ input[type=submit]
37 input[type=submit], #registerbutton, #loginbutton, #logoutbutton{
38     background-color: #201a55;
39     color: white;
40     padding: 15px 32px;
41     text-align: center;
42     width: 40%;
43     font-size: 20px;
44     font-weight: bold;
45     margin-bottom: 20px;
46     display: inline-block;
47     margin-right: 10px;
48 }
49 .labels{
50     font-weight: bold;
51     color: #201a55;
52 }
53 .message-flash{
54     color: #31708F;
55     background-color: #D9EDF7;
56     border-color: #d6e9c6;
57     padding: 15px;
58     margin-top: 20px;
59     margin-bottom: 10px;
60     border: 1px solid transparent;
61     border-radius: 4px;
62 }
```

OUTPUT

User Table

RMY92863.USER				Back
ROLLNUMBER	EMAIL	USERNAME	PASSWORD	
				There is no data here yet

Registration Page Rendered by Default

Registration

localhost:5000

Registration Page

Roll Number

2019103529

E-mail ID

ajesima02@gmail.com

Username

jesima31

Password

.....

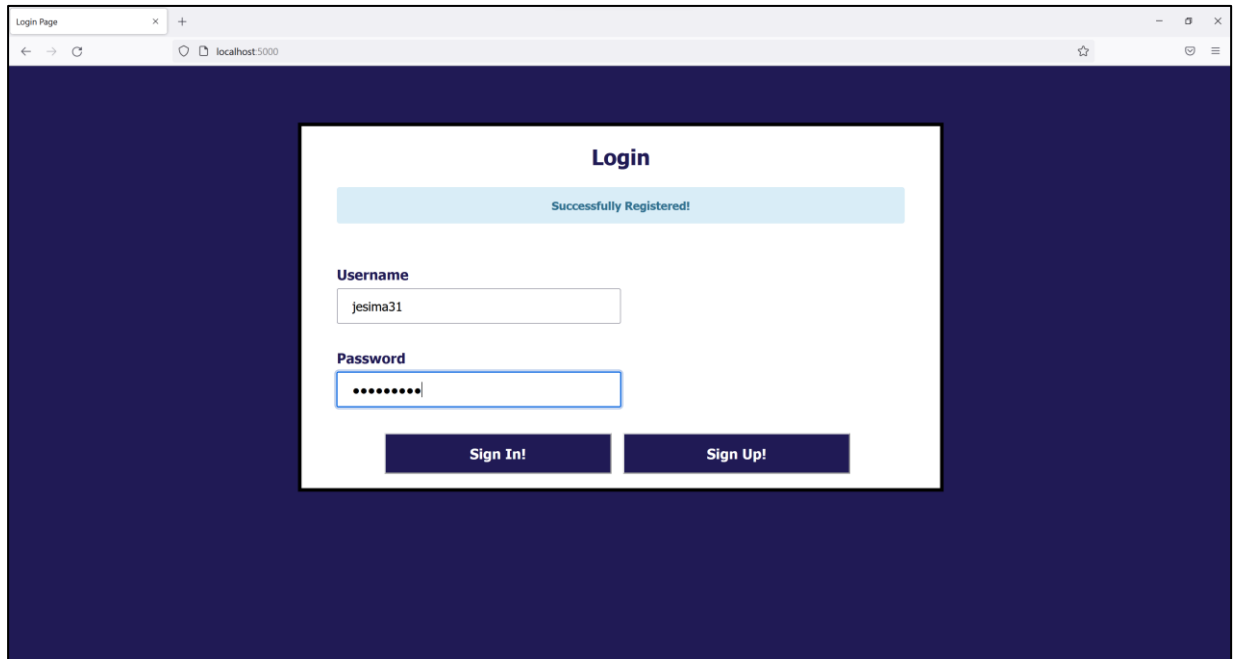
Sign Up!

Sign In!

Database Updated upon Registration

RMY92863.USER				Back
				Export to CSV
ROLLNUMBER	EMAIL	USERNAME	PASSWORD	
2019103529	ajesima02@gmail.com	jesima31	hello@123	

Login Page Rendered after Successful Registration



Login Page

localhost:5000

Login

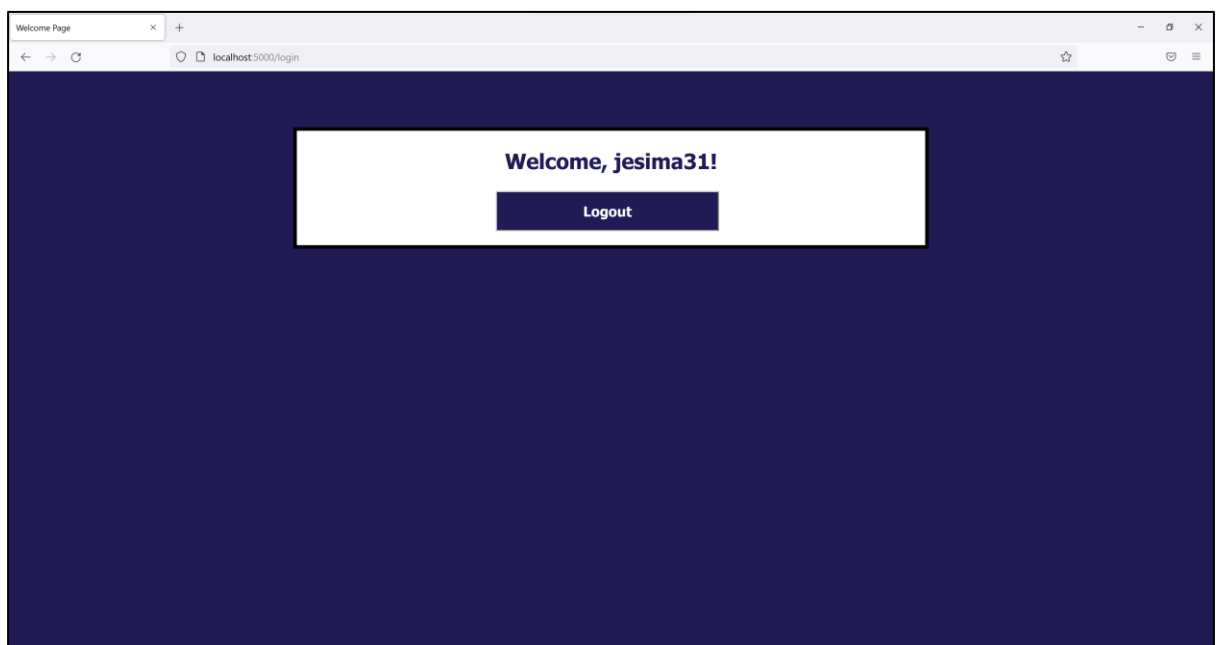
Successfully Registered!

Username

Password

[Sign In!](#) [Sign Up!](#)

Welcome Page Rendered on Successful Login



Welcome Page

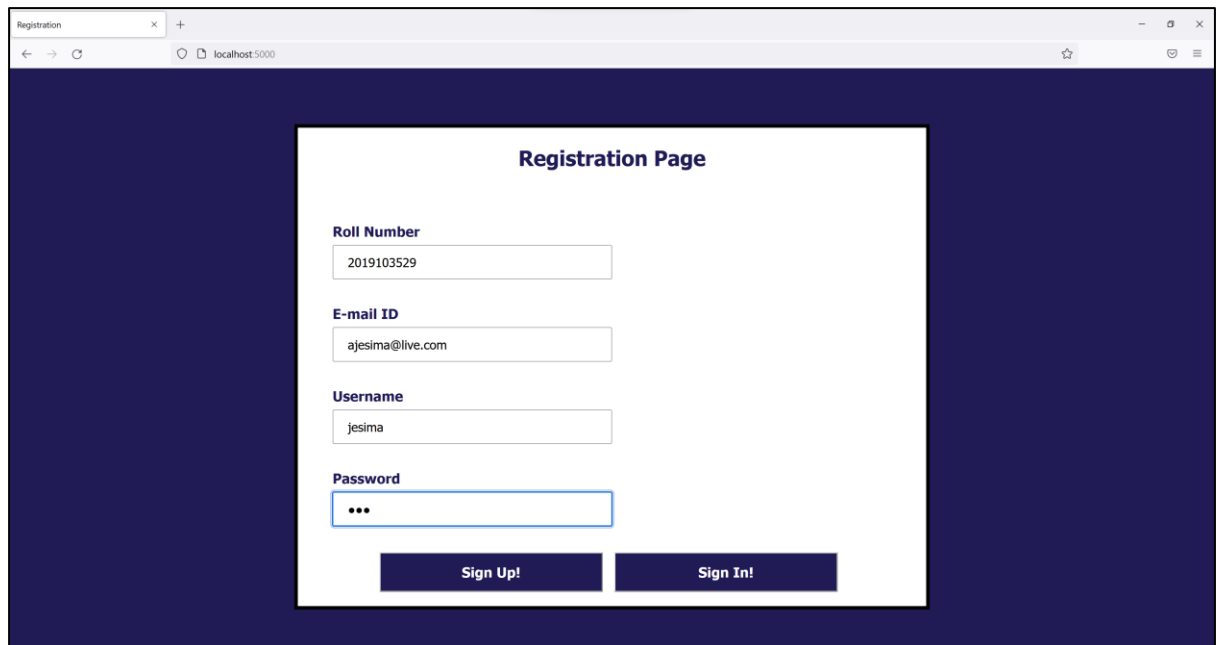
localhost:5000/login

Welcome, jesima31!

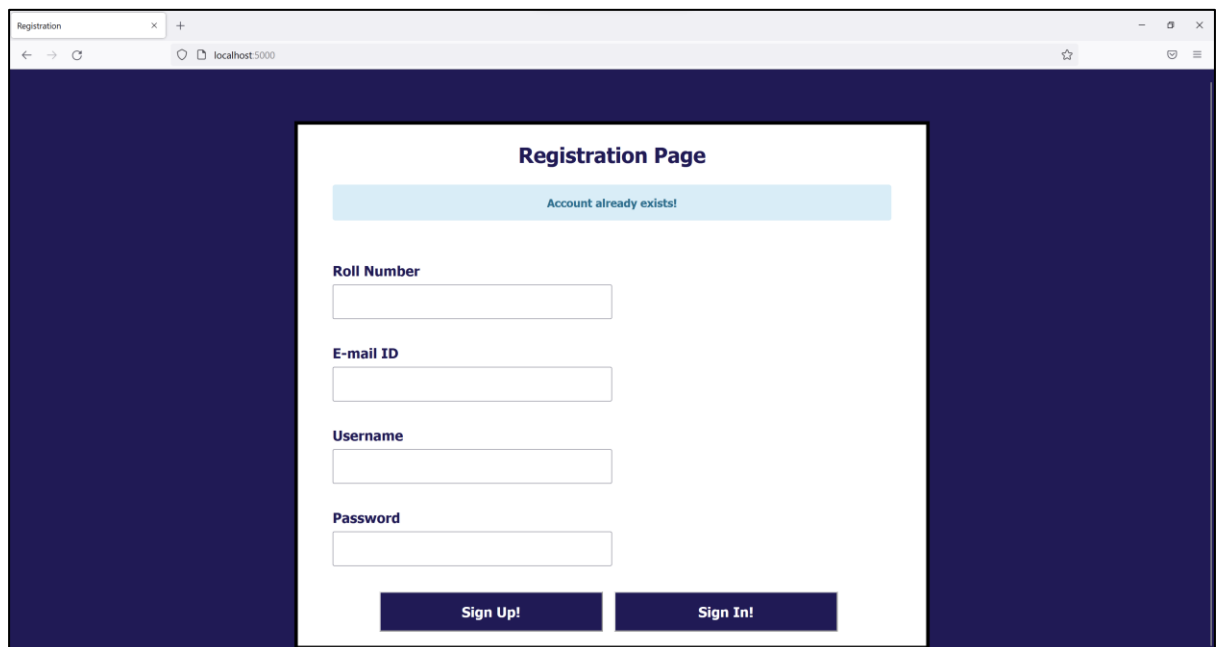
[Logout](#)

Secondary Cases

Case1: Roll Number Already Exists/ Username not unique

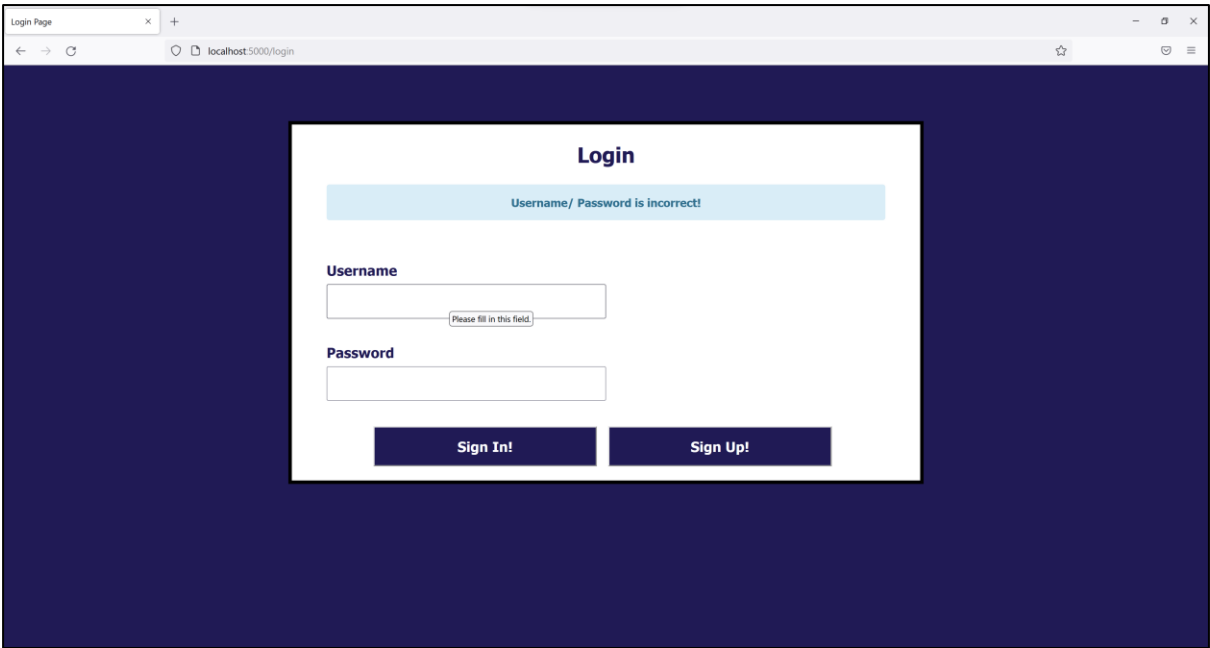
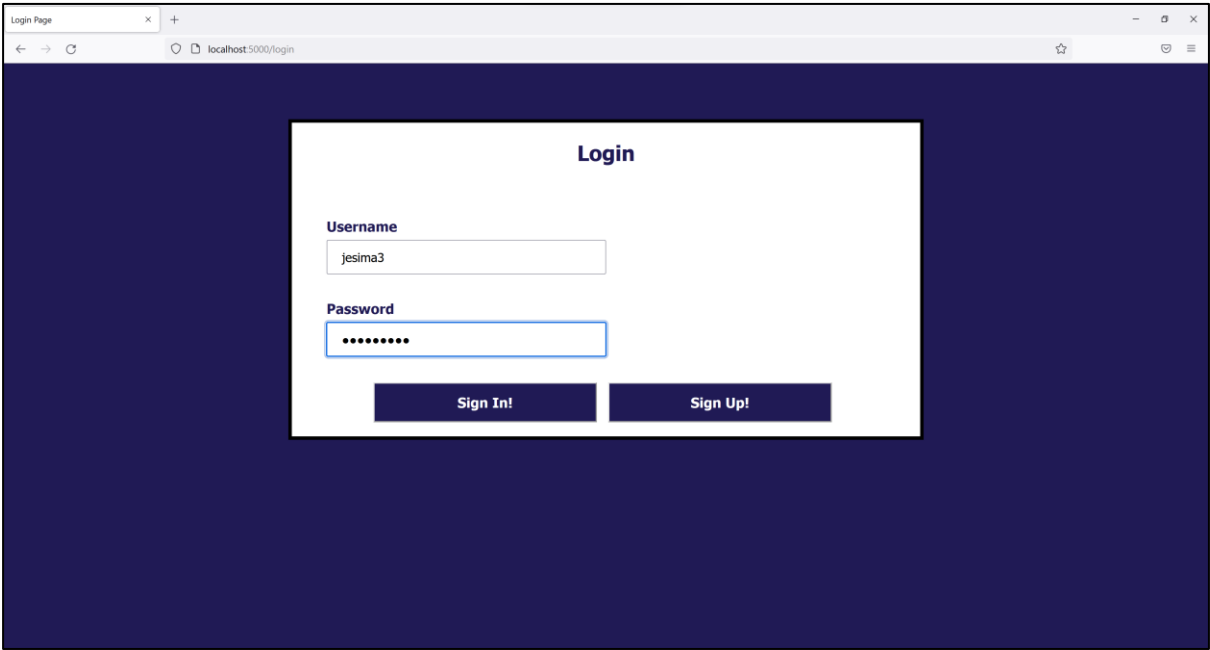


A screenshot of a web browser window titled "Registration" showing a registration page. The page has a dark blue background and a white central form. The form is titled "Registration Page" and contains four input fields: "Roll Number" (with value "2019103529"), "E-mail ID" (with value "ajesima@live.com"), "Username" (with value "jesima"), and "Password" (with three dots). Below the fields are two buttons: "Sign Up!" and "Sign In!".



A screenshot of a web browser window titled "Registration" showing the same registration page as above. A light blue error message banner at the top of the form reads "Account already exists!". The input fields for "Roll Number", "E-mail ID", "Username", and "Password" are now empty. The "Sign Up!" and "Sign In!" buttons remain at the bottom.

Case 2: Incorrect Login Credentials



Terminal

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  JUPYTER
Windows PowerShell
Copyright (c) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\Jes1\CEG\SEM 7\IBM\IBM-Project-23530-1659885342\ASSIGNMENTS\JESIMA A\ASSIGNMENT 2> & "C:/Program Files/Python/Python310/python.exe" "d:/Jes1/CEG/SEM 7/IBM/IBM-Project-23530-1659885342/ASSIGNMENTS/JESIMA A/ASSIGNMENT 2/app.py"
* Serving Flask app 'app'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5800
Press CTRL+C to quit
127.0.0.1 - - [25/Sep/2022 17:02:42] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [25/Sep/2022 17:02:42] "GET /static/styles.css HTTP/1.1" 304 -
Connected to Database
127.0.0.1 - - [25/Sep/2022 17:04:32] "POST / HTTP/1.1" 200 -
127.0.0.1 - - [25/Sep/2022 17:04:32] "GET /static/styles.css HTTP/1.1" 304 -
Connected to Database
127.0.0.1 - - [25/Sep/2022 17:04:41] "POST /login HTTP/1.1" 200 -
127.0.0.1 - - [25/Sep/2022 17:04:41] "GET /static/styles.css HTTP/1.1" 304 -
127.0.0.1 - - [25/Sep/2022 17:04:44] "GET /logout HTTP/1.1" 302 -
127.0.0.1 - - [25/Sep/2022 17:04:44] "GET /login HTTP/1.1" 200 -
127.0.0.1 - - [25/Sep/2022 17:04:44] "GET /static/styles.css HTTP/1.1" 304 -
Connected to Database
127.0.0.1 - - [25/Sep/2022 17:05:01] "POST /login HTTP/1.1" 200 -
127.0.0.1 - - [25/Sep/2022 17:05:01] "GET /static/styles.css HTTP/1.1" 304 -
127.0.0.1 - - [25/Sep/2022 17:05:21] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [25/Sep/2022 17:05:21] "GET /static/styles.css HTTP/1.1" 304 -
Connected to Database
127.0.0.1 - - [25/Sep/2022 17:05:42] "POST / HTTP/1.1" 200 -
127.0.0.1 - - [25/Sep/2022 17:05:42] "GET /static/styles.css HTTP/1.1" 304 -
PS D:\Jes1\CEG\SEM 7\IBM\IBM-Project-23530-1659885342\ASSIGNMENTS\JESIMA A\ASSIGNMENT 2>
```