

# **ASSIGNMENT-2**

**S. DIVYA THARSHINI**

**VV COLLEGE OF ENGINEERING**

**III-YEAR CSE**

**CLOUD APP DEVELOPMENT**

In the last Assignment you have created index.html and profile.html. So, now in this assignment you have create two html pages and style it using CSS.

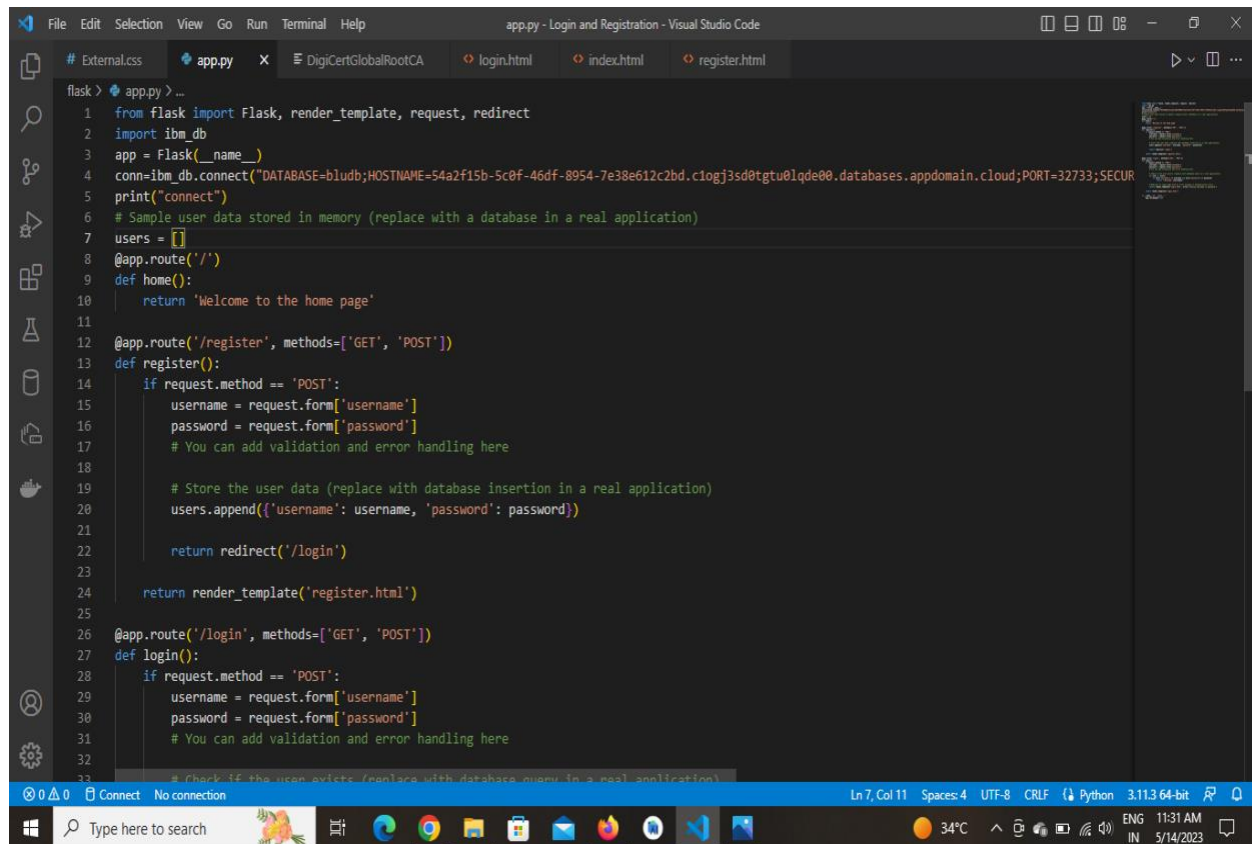
### 1. Login and Register html page

**Task 1:** You need to run the html pages in server by using Flask web framework

**Task 2:** Store the user registration details in IBM DB2

**Task 3:** while user logging in do the login validation (checking the login credentials is correct or not)

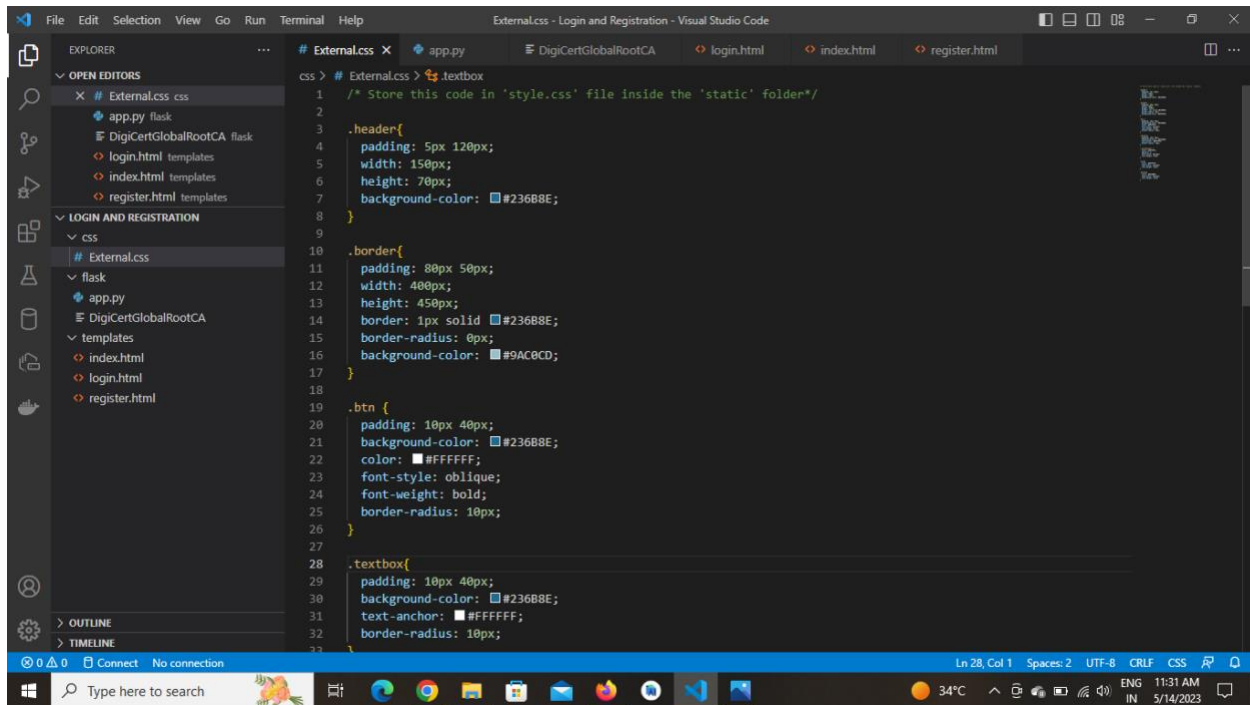
# App.py:



```
flask > app.py > ...
1 from flask import Flask, render_template, request, redirect
2 import ibm_db
3 app = Flask(__name__)
4 conn=ibm_db.connect("DATABASE=b1udb;HOSTNAME=54a2f15b-5c0f-46df-8954-7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32733;SECUR
5 print("connect")
6 # Sample user data stored in memory (replace with a database in a real application)
7 users = []
8 @app.route('/')
9 def home():
10     return 'Welcome to the home page'
11
12 @app.route('/register', methods=['GET', 'POST'])
13 def register():
14     if request.method == 'POST':
15         username = request.form['username']
16         password = request.form['password']
17         # You can add validation and error handling here
18
19         # Store the user data (replace with database insertion in a real application)
20         users.append({'username': username, 'password': password})
21
22         return redirect('/login')
23
24     return render_template('register.html')
25
26 @app.route('/login', methods=['GET', 'POST'])
27 def login():
28     if request.method == 'POST':
29         username = request.form['username']
30         password = request.form['password']
31         # You can add validation and error handling here
32
33         # Check if the user exists (replace with database query in a real application)
```

```
File Edit Selection View Go Run Terminal Help app.py - Login and Registration - Visual Studio Code
# External.css app.py x DigiCertGlobalRootCA login.html index.html register.html
flask > app.py > ...
19 # Store the user data (replace with database insertion in a real application)
20 users.append({'username': username, 'password': password})
21
22 return redirect('/login')
23
24 return render_template('register.html')
25
26 @app.route('/login', methods=['GET', 'POST'])
27 def login():
28     if request.method == 'POST':
29         username = request.form['username']
30         password = request.form['password']
31         # You can add validation and error handling here
32
33         # Check if the user exists (replace with database query in a real application)
34         for user in users:
35             if user['username'] == username and user['password'] == password:
36                 return f'Welcome, {username}!'
37
38         # Redirect to login page with an error message if authentication fails
39         return render_template('login.html', error='Invalid username or password')
40
41     return render_template('login.html')
42
43 if __name__ == '__main__':
44     app.run(debug=True)
45
```

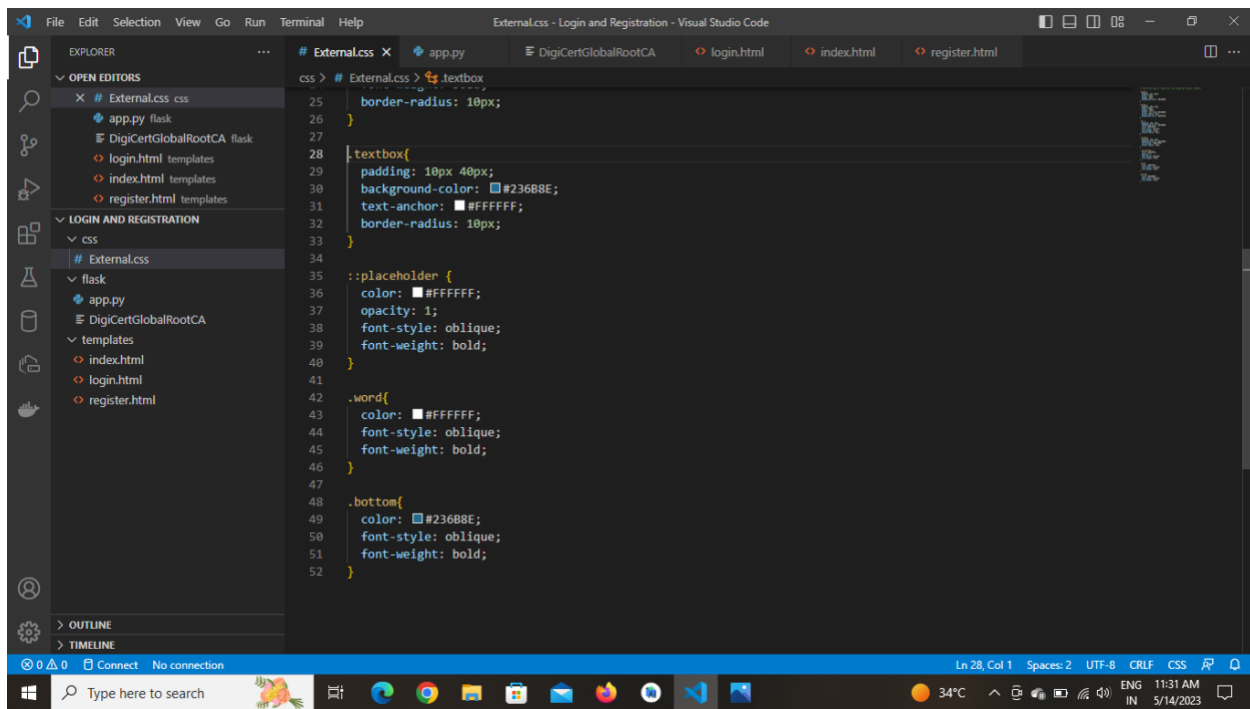
# External.css



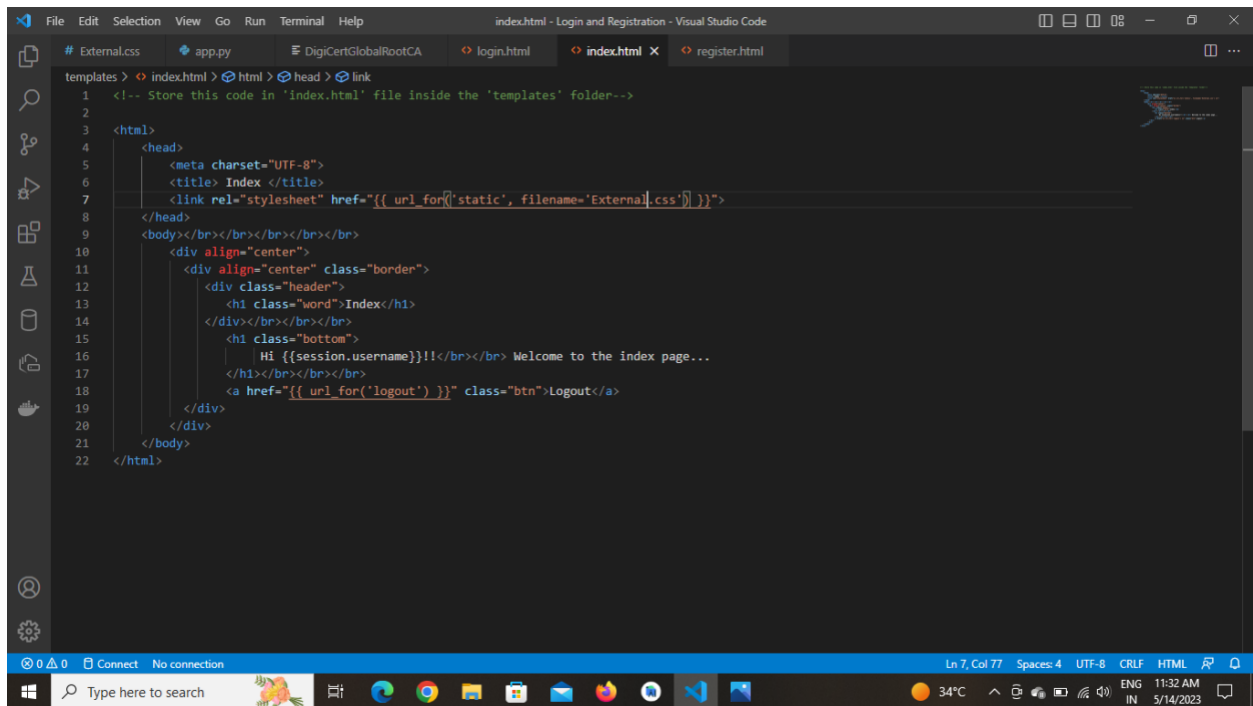
The screenshot shows the Visual Studio Code interface with the 'External.css' file open in the editor. The file is located in the 'css' folder of the 'Login and Registration' project. The code defines styles for a header, a border, a button, and a text box.

```
css > # External.css > .textbox
1  /* Store this code in 'style.css' file inside the 'static' folder*/
2
3  .header{
4      padding: 5px 120px;
5      width: 150px;
6      height: 70px;
7      background-color: #236B8E;
8  }
9
10 .border{
11     padding: 80px 50px;
12     width: 400px;
13     height: 450px;
14     border: 1px solid #236B8E;
15     border-radius: 0px;
16     background-color: #9AC8CD;
17 }
18
19 .btn {
20     padding: 10px 40px;
21     background-color: #236B8E;
22     color: #FFFFFF;
23     font-style: oblique;
24     font-weight: bold;
25     border-radius: 10px;
26 }
27
28 .textbox{
29     padding: 10px 40px;
30     background-color: #236B8E;
31     text-align: center;
32     border-radius: 10px;
33 }
```

The status bar at the bottom indicates the current position is Line 28, Column 1, with 2 spaces, in UTF-8 encoding, CRLF line endings, and CSS syntax highlighting. The system tray shows a temperature of 34°C and the date 5/14/2023.

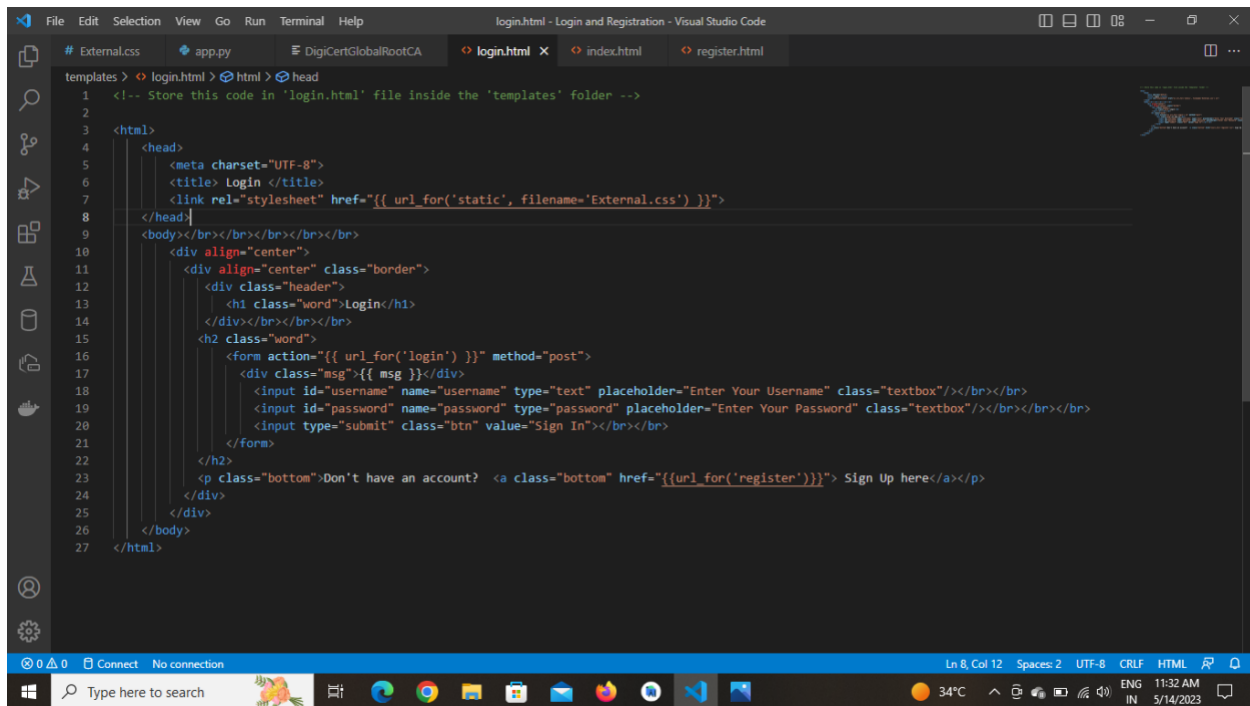


# Index.html



```
1  <!-- Store this code in 'index.html' file inside the 'templates' folder-->
2
3  <html>
4      <head>
5          <meta charset="UTF-8">
6          <title> Index </title>
7          <link rel="stylesheet" href="{{ url_for('static', filename='External.css')}}">
8      </head>
9      <body><br></br></br></br></br>
10         <div align="center">
11             <div align="center" class="border">
12                 <div class="header">
13                     <h1 class="word">Index</h1>
14                 </div><br></br></br></br>
15                 <div class="bottom">
16                     Hi {{session.username}}!!<br></br> Welcome to the index page...
17                 </div><br></br></br></br>
18                 <a href="{{ url_for('logout') }}" class="btn">Logout</a>
19             </div>
20         </div>
21     </body>
22 </html>
```

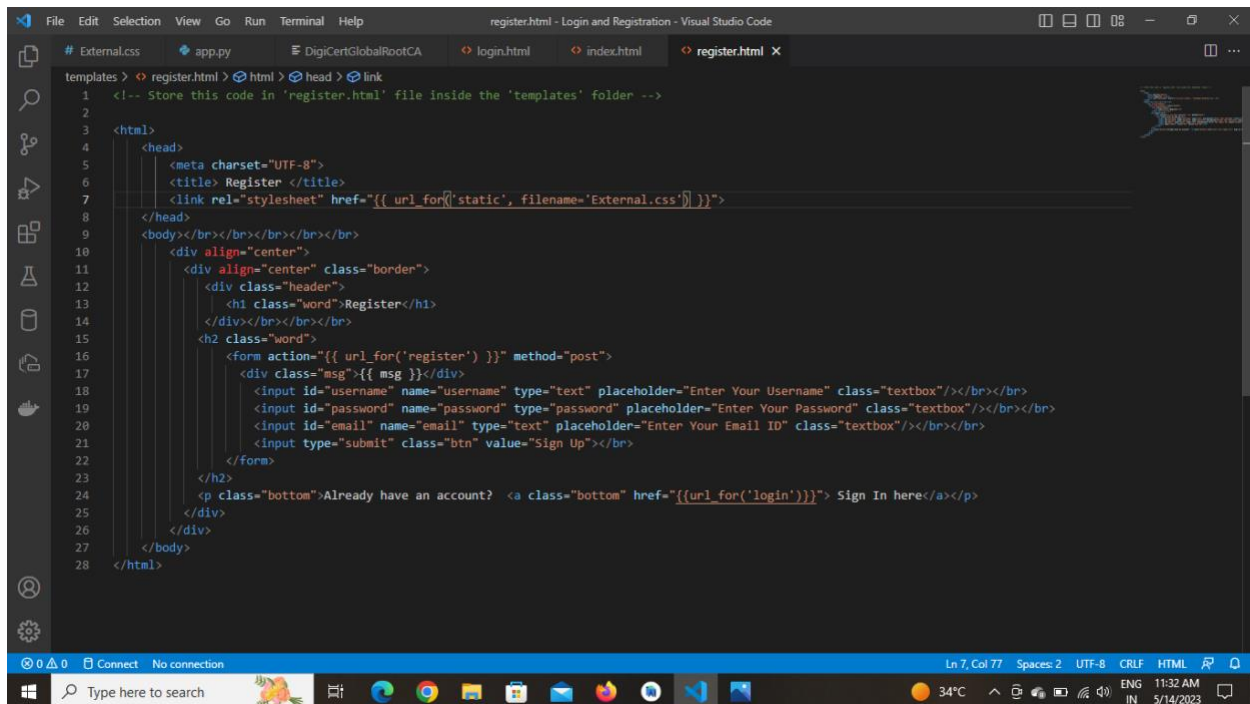
# Login.html



```
1 <!-- Store this code in 'login.html' file inside the 'templates' folder -->
2
3 <html>
4   <head>
5     <meta charset="UTF-8">
6     <title> Login </title>
7     <link rel="stylesheet" href="{{ url_for('static', filename='External.css') }}">
8   </head>
9   <body><br></br></br></br></br>
10   <div align="center">
11     <div align="center" class="border">
12       <div class="header">
13         <h1 class="word">Login</h1>
14       </div><br></br></br></br>
15       <h2 class="word">
16         <form action="{{ url_for('login') }}" method="post">
17           <div class="msg">{{ msg }}</div>
18           <input id="username" name="username" type="text" placeholder="Enter Your Username" class="textbox"/><br></br>
19           <input id="password" name="password" type="password" placeholder="Enter Your Password" class="textbox"/><br></br></br>
20           <input type="submit" class="btn" value="Sign In"/></br></br>
21         </form>
22       </h2>
23       <p class="bottom">Don't have an account? <a class="bottom" href="{{url_for('register')}}"> Sign Up here</a></p>
24     </div>
25   </div>
26 </body>
27 </html>
```



# Register.html



```
1 <!-- Store this code in 'register.html' file inside the 'templates' folder -->
2
3
4 <html>
5   <head>
6     <meta charset="UTF-8">
7     <title> Register </title>
8     <link rel="stylesheet" href="{{ url_for('static', filename='External.css')}}">
9   </head>
10  <body><br></br></br></br></br></br>
11    <div align="center">
12      <div align="center" class="border">
13        <div class="header">
14          <h1 class="word">Register</h1>
15        </div><br></br></br></br>
16        <h2 class="word">
17          <form action="{{ url_for('register') }}" method="post">
18            <div class="msg">{{ msg }}</div>
19            <input id="username" name="username" type="text" placeholder="Enter Your Username" class="textbox"/><br></br>
20            <input id="password" name="password" type="password" placeholder="Enter Your Password" class="textbox"/><br></br>
21            <input id="email" name="email" type="text" placeholder="Enter Your Email ID" class="textbox"/><br></br>
22            <input type="submit" class="btn" value="Sign Up"></br>
23          </form>
24        </h2>
25        <p class="bottom">Already have an account? <a class="bottom" href="{{url_for('login')}}"> Sign In here</a></p>
26      </div>
27    </div>
28  </body>
29 </html>
```

Login page:



A login page with a light blue background. At the top, there is a dark blue rectangular box containing the word "Login" in white, italicized font. Below this box, there are three input fields, each with a dark blue header and a light blue body. The first input field has a dark blue header with the text "Enter Your Username". The second input field has a dark blue header with the text "Enter Your Password". Below these two input fields is a dark blue button with the text "Sign In" in white. At the bottom of the form, there is a line of text: "Don't have an account? [Sign Up here](#)".

Registration page:



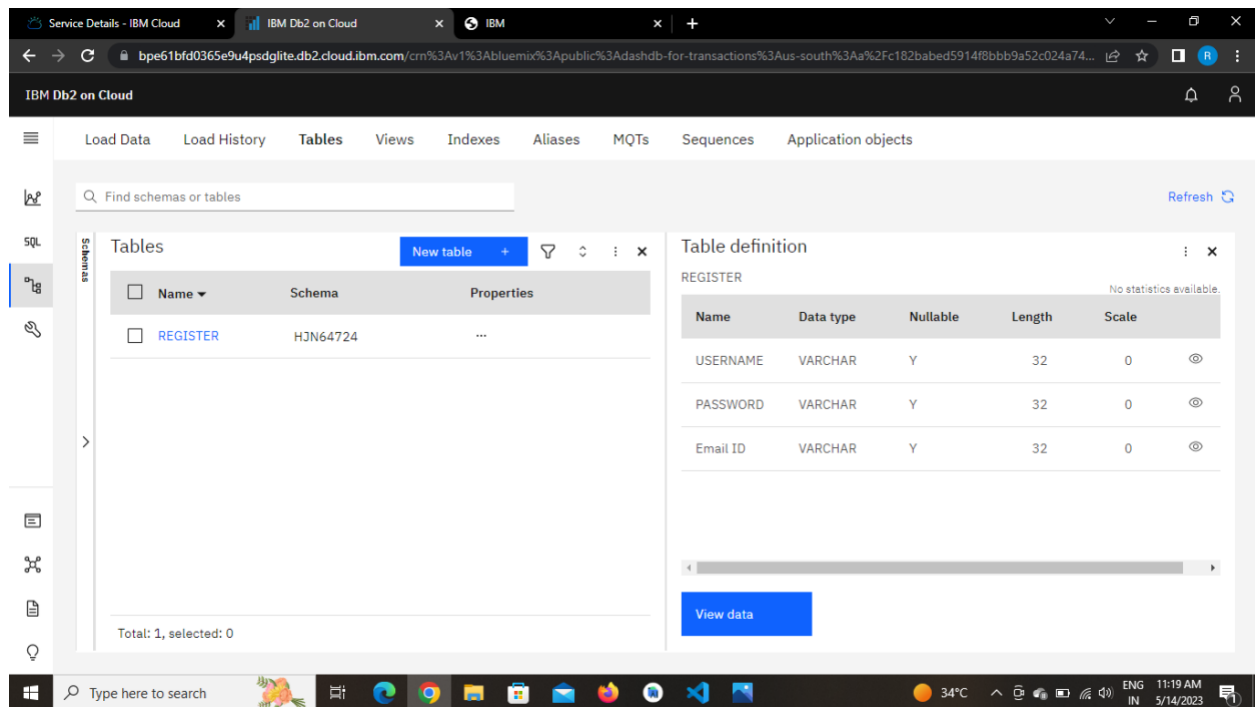
A registration page with a light blue background. At the top, there is a dark blue rectangular box containing the word "Register" in white, italicized font. Below this box, there are three input fields, each with a dark blue header and a light blue body. The first input field has a dark blue header with the text "Enter Your Username". The second input field has a dark blue header with the text "Enter Your Password". The third input field has a dark blue header with the text "Enter Your Email ID". Below these three input fields is a dark blue button with the text "Sign Up" in white. At the bottom of the form, there is a line of text: "Already have an account? [Sign In here](#)".

If registration successful:



A registration success screen with a light blue background. At the top, a dark blue button labeled "Register" is centered. Below it, the text "You have successfully registered!" is displayed in a bold, italicized font. Underneath this text are three dark blue buttons stacked vertically, labeled "Enter Your Username", "Enter Your Password", and "Enter Your Email ID". Below these buttons is another dark blue button labeled "Sign Up". At the bottom, the text "Already have an account? [Sign In here](#)" is displayed, with "Sign In here" as a link.

Database Table:

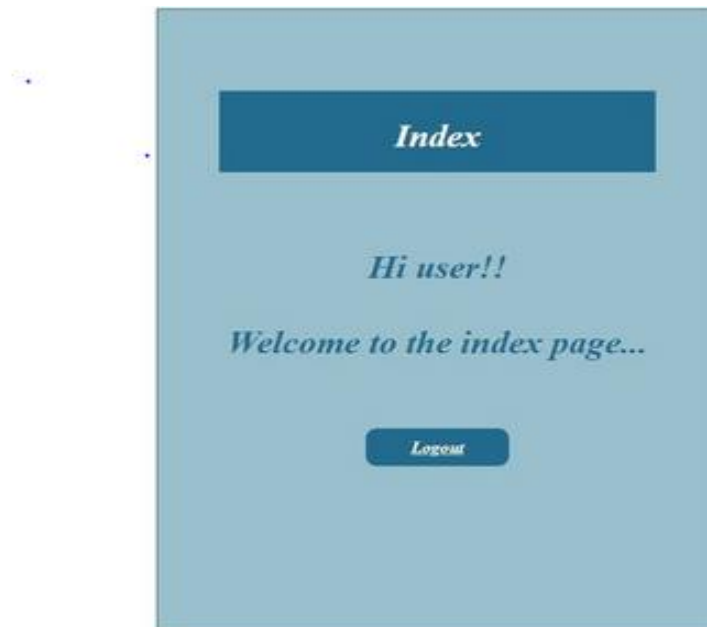


The screenshot shows the IBM Db2 on Cloud web interface. The top navigation bar includes tabs for "Service Details - IBM Cloud", "IBM Db2 on Cloud", and "IBM". The main content area has a sidebar on the left with a search bar and a "Tables" tab selected. The "Tables" tab displays a table with columns "Name", "Schema", and "Properties". The table contains one entry: "REGISTER" in the "Name" column, "HJN64724" in the "Schema" column, and "..." in the "Properties" column. To the right of the table is a "Table definition" panel for the "REGISTER" table. This panel shows the table's structure with columns: "Name", "Data type", "Nullable", "Length", and "Scale". The columns are: "USERNAME" (VARCHAR, Y, 32, 0), "PASSWORD" (VARCHAR, Y, 32, 0), and "Email ID" (VARCHAR, Y, 32, 0). A "View data" button is located at the bottom of the table definition panel. The bottom of the screenshot shows a Windows taskbar with various application icons and system information.

Name	Schema	Properties
REGISTER	HJN64724	...

Name	Data type	Nullable	Length	Scale
USERNAME	VARCHAR	Y	32	0
PASSWORD	VARCHAR	Y	32	0
Email ID	VARCHAR	Y	32	0

If login successful:



Indexpage is displayed:

