

ASSIGNMENT-2

IBM CLOUD DBS

Assignment Date	29 September 2022
Student Name	VIGNESH B
Student Roll Number	951919CS111

Questions:

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

The screenshot displays the IBM Db2 on Cloud web interface. The top navigation bar includes tabs for 'Load Data', 'Load History', 'Tables', 'Views', 'Indexes', 'Aliases', 'MQTs', 'Sequences', and 'Application objects'. The 'Tables' tab is active, showing a search bar and a 'New table' button. Below the search bar, a table lists the existing tables: 'STUDENTS' in the 'XYT88079' schema. The 'Table definition' panel on the right shows the structure of the 'STUDENTS' table:

Name	Data type	Nullable	Length	Scale
USERNAME	CHAR	Y	6	0
ROLLNO	INTEGER	Y		0
EMAIL ID	VARCHAR	Y	32	0
PASSWORD	VARCHAR	Y	32	0

At the bottom of the 'Table definition' panel, there is a 'View data' button. The bottom status bar shows the system temperature (23°C Cloudy), time (22:50), and date (19-10-2022).

2. Perform UPDATE, DELETE Queries with user table.

INSERT TABLE:-

The screenshot shows the IBM Db2 on Cloud SQL editor interface. The left sidebar displays the 'Data objects' tree with a filter 'ZZM98810'. The main editor area shows a script with the following SQL statements:

```
1 insert into user values('vigneshhackerstyle@gmail.com','Vignesh_B',190111,123);
2 UPDATE user SET password = 21052002 WHERE email='vigneshhackerstyle@gmail.com';
3
4
```

The 'History' tab is active, showing a list of executed scripts with their status and runtime. The first two scripts are highlighted:

Script	Date	Status	Runtime
Untitled - 1	Nov 8, 2022 2:54:53 PM	✓ 1	0.009 s
UPDATE user SET password = 21052002 WHERE email='vigneshhacke...		✓	0.009 s

The screenshot shows the IBM Db2 on Cloud SQL editor interface, specifically the 'Tables' view for the 'ZZM98810.USER' table. The table structure is displayed with the following columns:

EMAIL	USERNAME	roll number	PASSWORD
vigneshhackerstyle@gmail.com	Vignesh_B	190111	21052002

The 'Export to CSV' button is visible in the top right corner of the table view.

DELETE TABLE:-

The screenshot shows the IBM Db2 on Cloud console interface. On the left, the 'Data objects' tab is active, displaying a search bar and a list of objects including 'ZZM98810'. The main area shows a SQL script in a text editor with the following code:

```
1 insert into user values('vigneshhackerstyle@gmail.com', 'Vignesh_B', 190111, 123);
2 UPDATE user SET password = 21052002 WHERE email='vigneshhackerstyle@gmail.com';
3 DELETE FROM user WHERE email='vigneshhackerstyle@gmail.com';
4
5
6
```

Below the script, the 'History' tab is selected, showing a table of executed scripts:

Script	Date	Status	Runtime
Untitled - 1	Nov 8, 2022 2:58:13 PM	✓ 1	0.010 s
DELETE FROM user WHERE email='vigneshhackerstyle@gmail.com'		✓	0.010 s
Untitled - 1	Nov 8, 2022 2:54:53 PM	✓ 1	0.009 s
UPDATE user SET password = 21052002 WHERE email='vigneshhacke...		✓	0.009 s
Untitled - 1	Nov 8, 2022 2:52:50 PM	✓ 1	0.009 s
insert into user values('vigneshhackerstyle@gmail.com', 'Vigne...		✓	0.009 s

The screenshot shows the IBM Db2 on Cloud console interface for the 'ZZM98810.USER' table. The 'Tables' tab is active, and the table structure is displayed:

EMAIL	USERNAME	roll number	PASSWORD
		roll number	

Below the table structure, a message states: "There is no data here yet".

3. CONNECT PYTHON CODE TO DB2.

NOTE:- Question 4 contains Question 3 answer

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

Answers App.py

```
from flask import Flask, render_template, request, redirect, url_for, session
import ibm_db
import re
app = Flask(__name__) app.secret_key

=

conn = ibm_db.connect(
    "DATABASE=Db2New;HOSTNAME=19af6446-6171-4641-8aba-
9dcff8e1b6ff.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30699;USERNAM
E=kgb92696;PASSWORD=3IlWyGYU3bMIvwnX;SECURITY=SSL;SSLSERVERCERTIFICATE=DigiCer
tGlobalRootCA.crt;", "", "")

@app.route("/", methods=['GET', 'POST']) def
register():
    msg = '' if request.method
    == 'POST':
        username = request.form['username'] email =
        request.form['email'] password =
        request.form['password'] sql = "SELECT * FROM
        users WHERE username =?" stmt =
        ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt, 1, username)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print(account) if account:
            msg = 'Account already exists !'
        elif not re.match(r'^@+@[@]+\.[^@]+', email): msg
            = 'Invalid email address !'
        elif not re.match(r'[A-Za-z0-9]+', username):
            msg = 'name must contain only characters and numbers !'
```

```

        else:
            insert_sql = "INSERT INTO users VALUES (?, ?, ?)"
            prep_stmt = ibm_db.prepare(conn, insert_sql)
            ibm_db.bind_param(prepare_stmt, 1, username)
            ibm_db.bind_param(prepare_stmt, 2, email)
            ibm_db.bind_param(prepare_stmt, 3, password)
            ibm_db.execute(prepare_stmt) msg = 'You have
            successfully registered !'
    elif request.method == 'POST':
        msg = 'Please fill out the form !'
    return render_template('register.html', msg=msg)

@app.route('/login', methods=['GET', 'POST'])
def login(): global userid msg = ''

    if request.method == 'POST':
        username = request.form['username'] password =
        request.form['password'] sql = "SELECT * FROM users WHERE
        username=? AND password=?" stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt, 1, username) ibm_db.bind_param(stmt,
        2, password) ibm_db.execute(stmt) account =
        ibm_db.fetch_assoc(stmt) print(account) if account:
            session['loggedin'] = True session['id']
            = account['USERNAME'] userid =
            account['USERNAME'] session['username'] =
            account['USERNAME'] msg = 'Logged in
            successfully !'

            msg = 'Logged in successfully !'
            return render_template('dashboard.html', msg=msg)
        else:
            msg = 'Incorrect username / password !'
    return render_template('login.html', msg=msg)

if __name__ == '__main__':
    app.run(host='0.0.0.0')
    # app.run(debug=True)

```

```

<html Lang="en">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Register Form</title>

  <!-- bootstrap css cdn -->
  <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
  integrity="sha384-
JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z"
crossorigin="anonymous">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.css">
  <!-- css stylesheet -->
  <link rel="stylesheet" href="Style.css">
  <!-- font styles cdn -->
  <link rel="preconnect" href="https://fonts.gstatic.com">
  <link
href="https://fonts.googleapis.com/css2?family=Alegreya&display=swap"
rel="stylesheet">
  <link
href="https://fonts.googleapis.com/css2?family=Alegreya:wght@600&display=swa
p" rel="stylesheet"> </head>

<body>
  <!-- bootstrap navbar -->
  <!-- navbar ends -->
  <!-- Login form -->
  <div class="login text-center mt-5">

    <form action="/" method="post">
      <h2 class="form-text-h2"> Register Form </h2>
      <div class="msg">{{ msg }}</div>
      <input type="text" class="form-control" name="username"
placeholder="Enter Your Username" id="username" required>
      <input type="email" class="form-control" name="email"
placeholder="Enter Your Email ID" id="email" required>
      <input type="password" class="form-control" name="password"
placeholder="Enter Your Password" id="password" required>
      <button type="submit" id="button" class="btn btn-primary">
Register </button>

```

```

        <div class="note mt-3 text-center">
            <!--Register form -->
            <p> already have an account ? please login <a
href="/login">login! </a> </p>
        </div>
    </form>

</div>

</body>

</html>

```

Login.html

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Login Form</title>

    <link rel="icon" type="image/png" sizes="16x16" href="/assets/img/favicon-32x32.png">

    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
" integrity="sha384-
JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z"
crossorigin="anonymous">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.css">

    <link rel="stylesheet" href="styleless.css">

    <link rel="preconnect" href="https://fonts.gstatic.com">
    <link
href="https://fonts.googleapis.com/css2?family=Alegreya&display=swap"
rel="stylesheet">

```

```

    <link
href="https://fonts.googleapis.com/css2?family=Alegreya:wght@600&display=swap"
rel="stylesheet">
</head>
<body>

    <div class="login text-center mt-5">

        <form action="/login" method="post">
            <h2 class="form-text-h2"> Login Form </h2>
            <div class="msg">{{ msg }}</div>
            <div class="form-group mb-3">
                <label class="label" for="username">Username</label>
                <input type="text" class="form-control" placeholder="Username"
name="username" required>
            </div>
            <div class="form-group mb-3">
                <label class="label" for="password">Password</label>
                <input type="password" class="form-control"
placeholder="Password" name="password" required>
            </div>

            <button type="submit" id="button" class="btn btn-primary"> Login
</button>

            <div class="note mt-3 text-center">
                <!--Register form -->
                <p> Don't have an account yet? Click here to <a
href="/">register! </a> </p>
            </div>
        </form>
    </div>

</body>
</html>

```

Style.css

```

* { margin: 0;
padding: 0;
} body { color: #444444; background:linear-gradient(-45deg, #EE7752,
#E73C7E, #23A6D5, #23D5AB); background-size: 400% 400%; position:
relative; display: flex; flex-direction: column;

```



```

        align-items: center; justify-
content: center; animation: anime 10s
ease-in-out infinite; }
@keyframes anime{
    0%{ background-position: 0
        50%;
    }
    50%{ background-position:
        100% 50%;
    }
    100%{ background-position:
0 50%; } } form { display:
flex; flex-direction: column;
align-items: center; justify-
content: center; background-
color: white; width: 40%;
height: 450px; border-radius:
50px; margin: 100px; gap:
25px;
}

h2 { font-size:
    large; font-
weight:
    bolder;
}

input { width:
    80% !important;
}

div { display: flex;
align-items: center;
justify-content:
center; flex-
direction: column;
width: 90%; }

.form-text-vk
{ font-size:
    small;

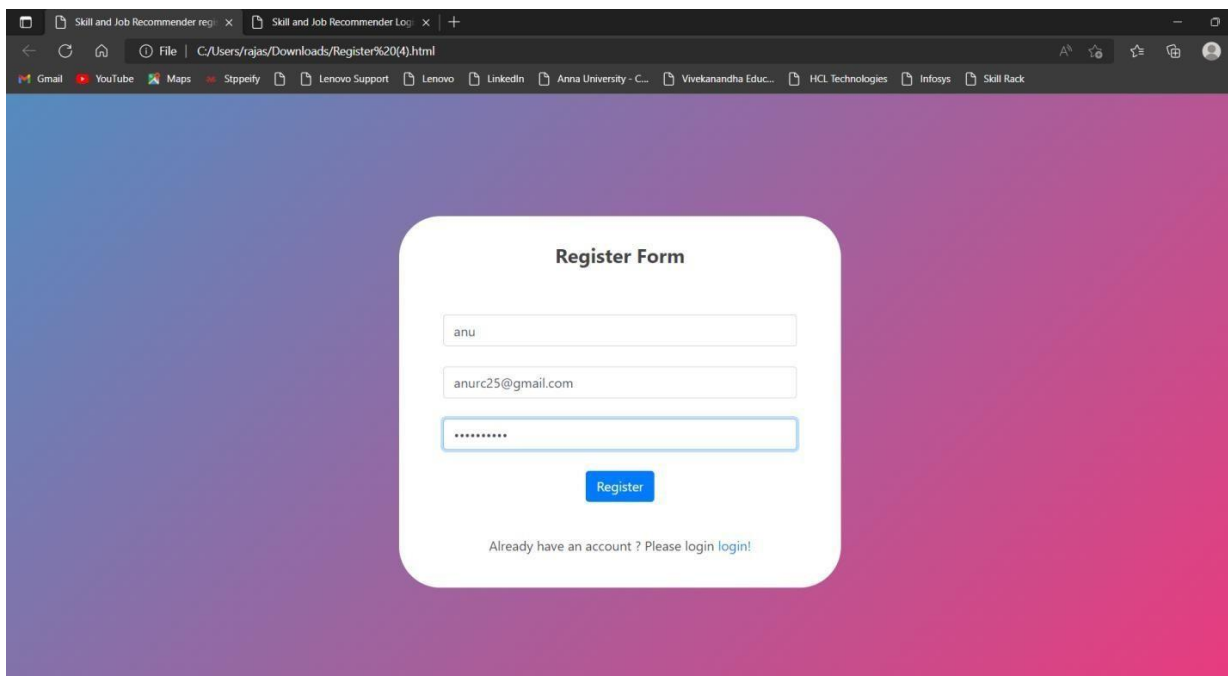
```

```
padding-bottom: 10px;
}

.form-text-h2 { font-size: x-large;
padding-top: 10px;
}
```

OUTPUT:-

REGISTRATION PAGE:-



The screenshot shows a web browser window with two tabs. The active tab is titled 'Skill and Job Recommender Logi...'. The address bar shows the file path 'C:/Users/rajas/Downloads/Register%20(4).html'. The browser's bookmark bar contains links to Gmail, YouTube, Maps, Stppety, Lenovo Support, Lenovo, LinkedIn, Anna University - C..., Vivekanandha Educ..., HCL Technologies, Infosys, and Skill Rack. The main content area features a registration form with a pink-to-purple gradient background. The form is titled 'Register Form' and contains three input fields: a username field with 'anu', an email field with 'anurc25@gmail.com', and a password field with masked characters. Below the fields is a blue 'Register' button. At the bottom of the form, there is a link: 'Already have an account ? Please login [login!](#)'.

LOGIN PAGE:-

Browser tabs: Skill and Job Recommender reg..., Skill and Job Recommender Log...

Address bar: C:/Users/rajas/Downloads/Login%20(2).html

Browser bookmarks: Gmail, YouTube, Maps, Stppeify, Lenovo Support, Lenovo, LinkedIn, Anna University - C..., Vivekanandha Educ..., HCL Technologies, Infosys, Skill Rack

Login Form

Username:

Password:

[Login](#)

Don't have an account yet? Click here to [register!](#)

WELCOME PAGE:-

