

Assignment -2

IBM CLOUD DB2

| | |
|---------------------|-------------------|
| Assignment Date | 29 September 2022 |
| Student Name | MOHANA SAKTHI G S |
| Student Roll Number | 613019104046 |
| Maximum Marks | 2 Marks |

Questions:

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

The screenshot displays the IBM Db2 on Cloud web interface. The browser address bar shows a URL starting with 'bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com'. The interface includes a top navigation bar with 'Load Data', 'Load History', 'Tables', 'Views', 'Indexes', 'Aliases', 'MQTs', 'Sequences', and 'Application objects'. The 'Tables' tab is active, showing a list of tables with columns 'Name', 'Schema', and 'Properties'. A table named 'STUDENT' is listed under schema 'JRZ46343'. To the right, the 'Table definition' panel for the 'STUDENT' table is shown, detailing the following columns:

| Name | Data type | Nullable | Length | Scale |
|-------------|-----------|----------|--------|-------|
| USERNAME | CHAR | Y | 5 | 0 |
| ROLL NUMBER | INTEGER | Y | | 0 |
| EMAIL ID | VARCHAR | Y | 32 | 0 |
| PASSWORD | VARCHAR | Y | 32 | 0 |

At the bottom of the interface, a Windows taskbar is visible with the date '19-10-2022' and time '10:23'.

2. Perform UPDATE,DELETE Queries with user table.

UPDATE TABLE:-

The screenshot shows the IBM Db2 on Cloud console interface. On the left, the 'Data objects' panel lists 'JRZ46343'. The main editor area, titled '*Untitled - 1', contains the SQL query: `INSERT INTO student values ('mona','46','mona@gmail.com','hardik');`. The 'History' panel at the bottom displays the execution details:

| Script | Date | Status | Runtime |
|--------------------------------------------------------------------|--------------------------|--------|---------|
| Untitled - 1 | Oct 19, 2022 10:19:12 AM | ✓ 1 | 0.000 s |
| INSERT INTO student values ('mona','46','mona@gmail.com','hardik') | | ✓ | 0.000 s |

The screenshot shows the IBM Db2 on Cloud console displaying the 'JRZ46343.STUDENT' table. The table has four columns: USERNAME, ROLL NUMBER, EMAIL ID, and PASSWORD. The data is as follows:

| USERNAME | ROLL NUMBER | EMAIL ID | PASSWORD |
|----------|-------------|----------------|----------|
| mona | 46 | mona@gmail.com | hardik |

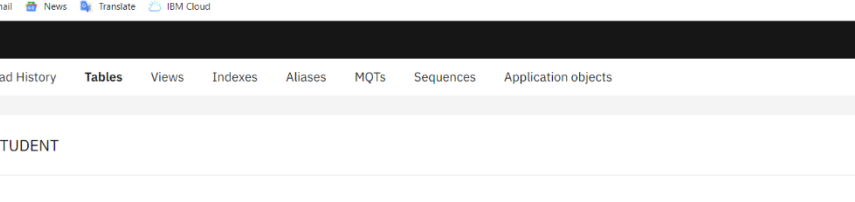
DELETE TABLE:-

The screenshot shows the IBM Db2 on Cloud console. On the left, there's a sidebar with 'Data objects' and 'My script'. The main area displays a SQL script in a text editor. The script contains three statements:

```
1 INSERT INTO STUDENT values ('mona','46','mona@gmail.com','hardik');
2 INSERT INTO STUDENT values ('ram','47','ram@gmail.com','hardik');
3 DELETE FROM STUDENT WHERE USERNAME = 'mona';
```

Below the script, there's a 'History' tab showing the execution results. The table has columns: Script, Date, Status, and Runtime.

| Script | Date | Status | Runtime |
|--------------------------------------------------------------------|--------------------------|--------|---------|
| Untitled - 1 | Oct 19, 2022 11:55:31 AM | ✓ 3 | 0.020 s |
| INSERT INTO STUDENT values ('mona','46','mona@gmail.com','hardik') | | ✓ | 0.006 s |
| INSERT INTO STUDENT values ('ram','47','ram@gmail.com','hardik') | | ✓ | 0.006 s |
| DELETE FROM STUDENT WHERE USERNAME = 'mona' | | ✓ | 0.008 s |
| Untitled - 1 | Oct 19, 2022 11:53:16 AM | ✓ 2 | 0.015 s |



Service Details - IBM Cloud x IBM Db2 on Cloud x +

bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crm%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2F0b73012b1065427ba4f18d2302516f99%3A03590afc-7750-47a... ☆

YouTube Maps Gmail News Translate IBM Cloud

IBM Db2 on Cloud

Load Data Load History **Tables** Views Indexes Aliases MQTs Sequences Application objects

JRZ46343.STUDENT Back

Export to CSV

| USERNAME | ROLLNUMBER | EMAIL ID | PASSWORD |
|----------|------------|---------------|----------|
| ram | 47 | ram@gmail.com | hardik |
| ram | 47 | ram@gmail.com | hardik |
| ram | 47 | ram@gmail.com | hardik |

31°C Mostly sunny Search the web 11:58 19-10-2022

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 =
'a'
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)

```

Register.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>Skill and Job Recommender register</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/fontawesome/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=swap"
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>Skill and Job Recommender Login</title>
    <!-- favicon -->
    <!-- <link rel="shortcut icon" href="/assets/img/favicon.ico"
type="image/x-icon"> -->
    <!-- <link rel="icon" href="/assets/img/favicon.ico" type="image/x-icon">
-->
    <link rel="icon" type="image/png" sizes="16x16"
href="/assets/img/favicon32x32.png">
    <!-- 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/fontawesome/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=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;
```

```

        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;

```

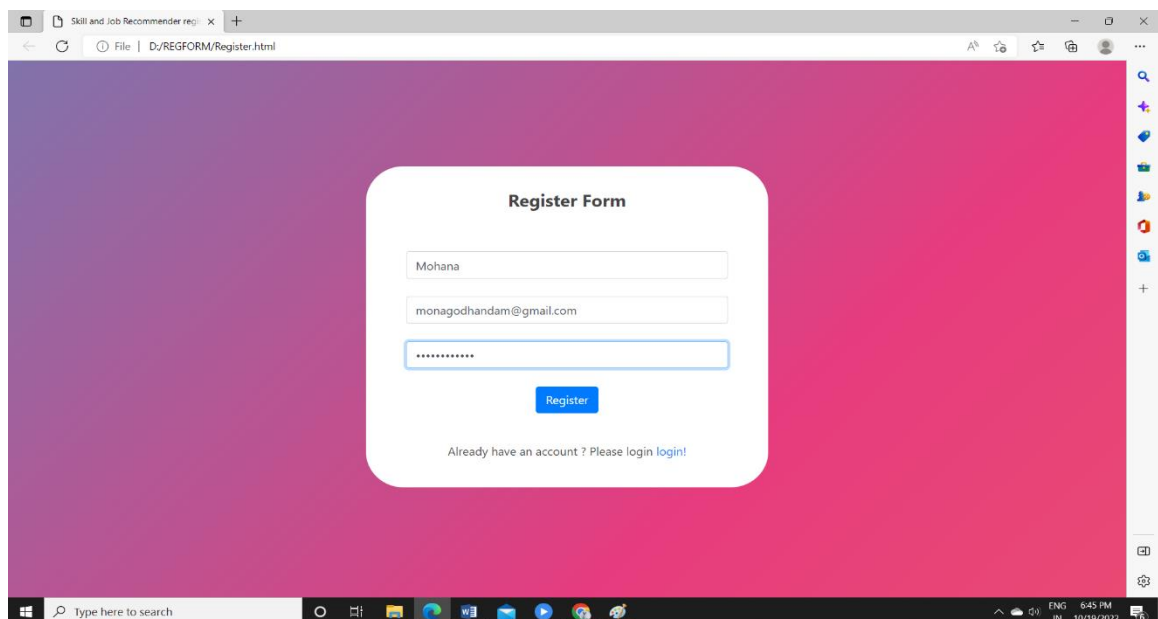
```
} body
{
    color: #444444;    background:linear-gradient(-45deg, #EE7752, #E73C7E,
#23A6D5, #23D5AB);    background-size: 400% 400%;    position: relative;
display: flex;    flex-direction: column;
```

```
padding-bottom: 10px;
}

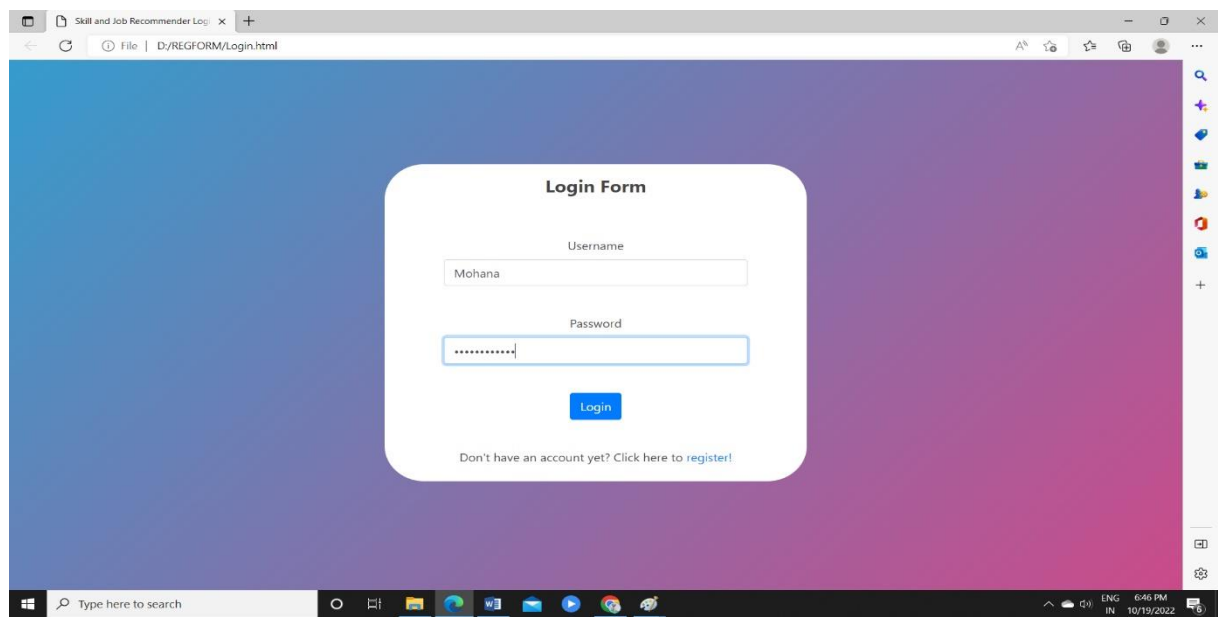
.form-text-h2 {    font-size: x-large;    padding-top: 10px;
}
```

OUTPUT:-

REGISTRATION PAGE:-



LOGIN PAGE:-



The screenshot shows a web browser window with the address bar displaying "D:/REGFORM/Login.html". The page has a blue and purple gradient background. In the center is a white rounded rectangle containing the "Login Form". The form has two input fields: "Username" with the text "Mohana" and "Password" with masked characters "*****". Below the fields is a blue "Login" button. At the bottom of the form, it says "Don't have an account yet? Click here to [register!](#)". The Windows taskbar is visible at the bottom with the search bar and various application icons.

WELCOME PAGE:-

