

ASSIGNMENT- 2

IBM CLOUD DB2

Date	29 September 2022
Student Name	AGILA P
Student Roll Number	613019104001
Maximum Marks	2 Marks

Questions:

- 1) Create the 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 the URL: `bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2Fd5abf...`. The interface includes a top navigation bar with tabs for 'Load Data', 'Load History', 'Tables', 'Views', 'Indexes', 'Aliases', 'MQTs', 'Sequences', and 'Application objects'. The 'Tables' tab is active. On the left, a sidebar shows a search bar 'Find schemas or tables' and a list of schemas, with 'KPH64374' selected. The main area is divided into two panels: 'Tables' on the left and 'Table definition' on the right. The 'Tables' panel shows a table named 'STUDENTS' under the schema 'KPH64374'. The 'Table definition' panel shows the table's structure with the following columns:

Name	Data type	Nullable	Length	Scale
ROLL NO	INTEGER	Y		0
USERNAME	CHAR	Y	5	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 of the screenshot shows a Windows taskbar with the search bar, task icons, and system tray information indicating 28°C Cloudy, 20:30, and 18-10-2022.

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 a database 'KPH64374'. The main area is titled 'Untitled - 1' and contains a SQL script: `INSERT INTO STUDENTS values('01','Agila','agila@gmail.com','Charmy');`. Below the script, the 'History' tab shows a table of executed queries.

Script	Date	Status	Runtime
Untitled - 1	Oct 18, 2022 8:42:39 PM	✓ 1	0.005 s
INSERT INTO STUDENTS values('01','Agila','agila@gmail.com','Ch...		✓	0.005 s

The screenshot shows the IBM Db2 on Cloud console interface, specifically the 'Tables' tab. The table 'KPH64374.STUDENTS' is displayed with the following data:

ROLLNO	USERNAME	EMAIL ID	PASSWORD
1	Agila	agila@gmail.com	Charmy
1	Agila	agila@gmail.com	Charmy

DELETE TABLE:

Service Details - IBM Cloud

IBM Db2 on Cloud

New Tab

bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crm%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2Fd5abf...

IBM Db2 on Cloud

Data objects

My script

Filter objects

KPH64374

*Untitled - 1

Beta Classic

Run all

```
1 INSERT INTO STUDENTS values('01','Agila','agila@gmail.com','Chazmy');
2 INSERT INTO STUDENTS values('06','Shiva','sheela@gmail.com','Shivu');
3 DELETE FROM STUDENTS WHERE USERNAME='Agila';
4
```

History

Find history

Script	Date	Status	Runtime
Untitled - 1	Oct 18, 2022 8:57:55 PM	3	0.014 s
INSERT INTO STUDENTS values('01','Agila','agila@gmail.com','Ch...			0.006 s
INSERT INTO STUDENTS values('06','Shiva','sheela@gmail.com','S...			0.004 s
DELETE FROM STUDENTS WHERE USERNAME='Agila'			0.004 s

Type here to search

28°C Cloudy

ENG

20:58

18-10-2022

Service Details - IBM Cloud

IBM Db2 on Cloud

New Tab

bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crm%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2Fd5abf...

IBM Db2 on Cloud

Load Data

Load History

Tables

Views

Indexes

Aliases

MQTs

Sequences

Application objects

KPH64374.STUDENTS

Back

Export to CSV

ROLL NO	USERNAME	EMAIL	PASSWORD
6	Shiva	sheela@gmail.com	Shivu

Type here to search

28°C Cloudy

ENG

21:03

18-10-2022

3) CONNECT PYTHON CODE TO DB2.

NOTE: Question 4 contains Question 3 answer

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>Nutrition Assistant 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/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=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>Nutrition Assistant 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/favicon-
32x32.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/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=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%;
}
```

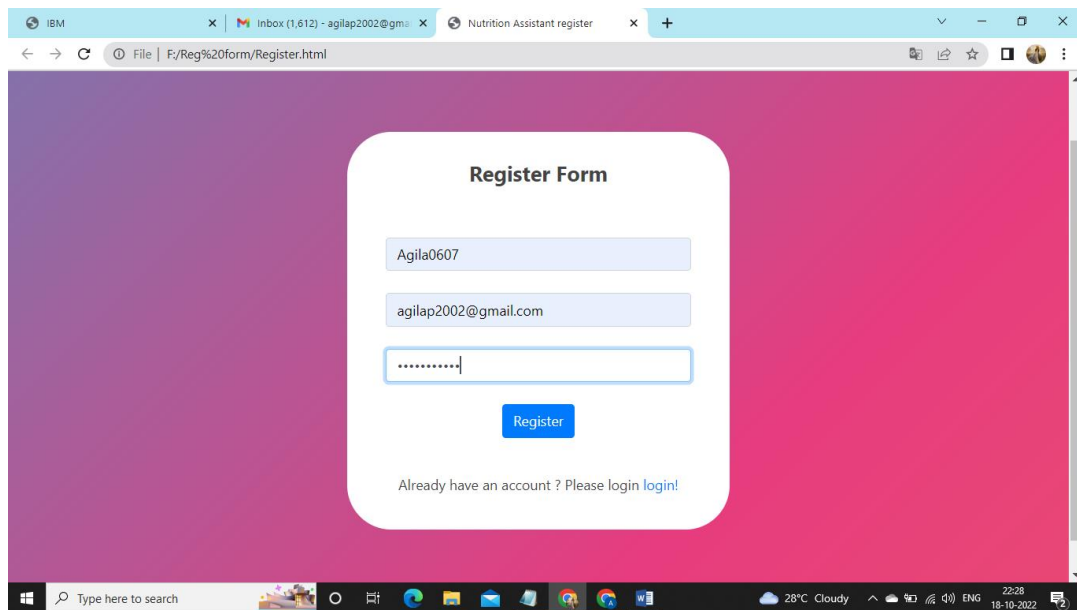
```
.form-text-vk {
```

```
padding-bottom: 10px;
}

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

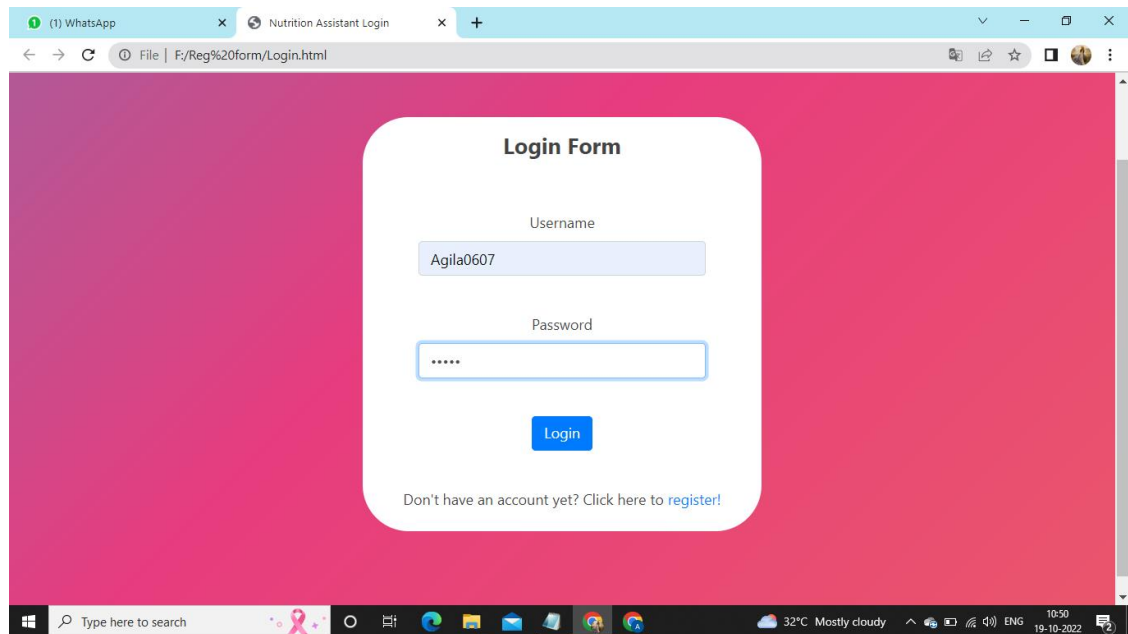
OUTPUT:

REGISTRATION PAGE:



The screenshot displays a web browser window with the address bar showing 'File | F:/Reg%20form/Register.html'. The browser has three tabs: 'IBM', 'Inbox (1,612) - agilap2002@gmail.com', and 'Nutrition Assistant register'. The main content area features a registration form titled 'Register Form' centered on a pink-to-purple gradient background. The form contains three input fields: the first is filled with 'Agila0607', the second with 'agilap2002@gmail.com', and the third is masked with dots. Below the inputs is a blue 'Register' button. At the bottom of the form, it says 'Already have an account ? Please login [login!](#)'. The Windows taskbar at the bottom shows the search bar, task view button, and several application icons, along with system information: '28°C Cloudy', 'ENG', and the date '18-10-2022'.

LOGIN PAGE:



The screenshot shows a web browser window with two tabs: '(1) WhatsApp' and 'Nutrition Assistant Login'. The address bar displays 'File | F:/Reg%20form/Login.html'. The page has a pink gradient background. In the center is a white rounded rectangle containing the 'Login Form'. The form has two input fields: 'Username' with the text 'Agila0607' 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 at the bottom shows the search bar, task view button, and several application icons. The system tray on the right indicates '32°C Mostly cloudy' and the date '19-10-2022'.

Login Form

Username
Agila0607

Password

Login

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

WELCOME PAGE:

