

Assignment - 2 IBM CLOUD DB2

Assignment Date	29 September 2022
Student Name	BEULA B
Student Roll Number	613019104010
Maximum Marks	2 Marks

Questions:

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

The screenshot shows the IBM Db2 on Cloud web interface. On the left, the 'Data objects' tab is active, showing a search bar and a list of objects including 'ZXP19479'. The main area displays a SQL script in a text editor: `INSERT INTO STUDENTS values ('beula','78','beu@gmail.com','Duke');`. Below the editor, the 'History' tab is open, showing a table of executed scripts.

Script	Date	Status	Runtime
Untitled - 1	Oct 19, 2022 10:07:50 PM	✓ 1	0.010 s
INSERT INTO STUDENTS values ('beula','78','beu@gmail.com...		✓	0.010 s

At the bottom of the console, there is a message: 'Activate Windows Go to Settings to activate Windows.'

2. Perform UPDATE,DELETE Queries with user table.

UPDATE TABLE:-

IBM Db2 on Cloud

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

Find schemas or tables Refresh

Tables New table +

Name	Schema	Properties
STUDENTS	ZXP19479	...

Total: 1, selected: 1

Table definition STUDENTS No statistics available.

Name	Data type	Nullable	Length	Scale
USERNAME	CHAR	Y	5	0
ROLLNUMBER	INTEGER	Y		0
EMAILID	VARCHAR	Y	32	0
PASSWORD	VARCHAR	Y	32	0

View data

Activate Windows Go to Settings to activate Windows.

IBM Db2 on Cloud

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

ZXP19479.STUDENTS Back

Export to CSV

USERNAME	ROLLNUMBER	EMAILID	PASSWORD
beula	78	beu@gmail.com	Duke

Activate Windows Go to Settings to activate Windows.

The screenshot displays the IBM Db2 on Cloud console interface. The top navigation bar shows the service details and the database name. The left sidebar contains icons for various database functions, with 'SQL' selected. The main area is divided into two panes: 'Data objects' and 'My script'. The 'My script' pane shows a SQL script with three statements: two INSERT statements and one DELETE statement. The 'History' pane below the script shows the execution history of the script, including the script name, date, status, and runtime.

Script	Date	Status	Runtime
Untitled - 1	Oct 19, 2022 10:16:02 PM	3	0.030 s
INSERT INTO STUDENTS values ('beula', '78', 'beu@gmail.com', 'Duke');			0.011 s
INSERT INTO STUDENTS values ('nila', '08', 'nila@gmail.com', 'Door');			0.008 s
DELETE FROM STUDENTS WHERE USERNAME = 'beula';			0.011 s
Untitled - 1	Oct 19, 2022 10:07:50 PM	1	0.010 s

DELETE TABLE:-

Service Details - IBM Cloud

IBM Db2 on Cloud

+

← → ↻

bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2Fa75048...

Paused

IBM Db2 on Cloud

Load Data

Load History

Tables

Views

Indexes

Aliases

MQTs

Sequences

Application objects

ZXP19479.STUDENTS

Back

Export to CSV

USERNAME	ROLLNUMBER	EMAILID	PASSWORD
nila	8	nila@gmail.com	Door

Activate Windows
Go to Settings to activate Windows.

Search Windows

10:17 PM

19-Oct-22

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=3WYGYU3bMvwnX;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'^[a-zA-Z0-9]+@[a-zA-Z0-9]+\.[a-zA-Z0-9]+$', email):
            msg = 'Invalid email address !'
        elif not re.match(r'^[A-Za-z0-9]+$', 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>Containment Zone Alerting 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>Containment Zone Alerting 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%;
}

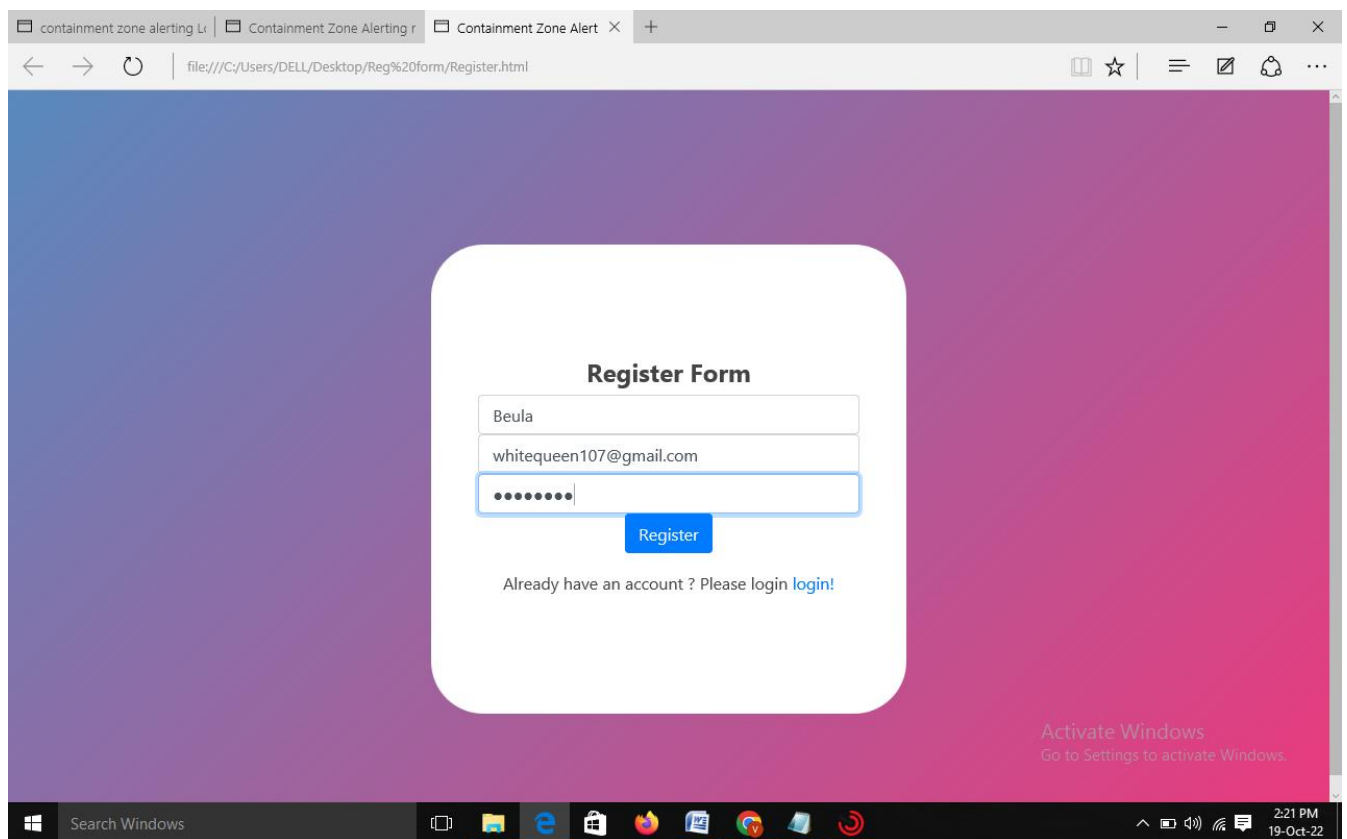
.form-text-vk {
    font-size: small;

```

```
padding-bottom: 10px;
}

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

OUTPUT:- REGISTRATION PAGE:-



The screenshot shows a web browser window with a registration form. The browser's address bar displays the file path: `file:///C:/Users/DELL/Desktop/Reg%20form/Register.html`. The form is centered on a purple-to-pink gradient background. It contains three input fields: a text field with the value 'Beula', an email field with the value 'whitequeen107@gmail.com', and a password field with masked characters. Below the inputs is a blue 'Register' button. A link for existing users is provided at the bottom of the form. An 'Activate Windows' watermark is visible in the bottom right corner of the browser window.

Register Form

Beula

whitequeen107@gmail.com

.....

Register

Already have an account ? Please login [login!](#)

Activate Windows
Go to Settings to activate Windows.

LOGIN PAGE:-

containment zone alerting Lc containment zone alertit

file:///C:/Users/DELL/Desktop/Reg%20form/Login.html

Login Form

Username

Kavya

Password

.....

Login

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

Activate Windows
Go to Settings to activate Windows.

Search Windows

2:27 PM
19-Oct-22

WELCOME PAGE:-

