

Assignment -2

IBM CLOUD DB2

Student Name	Ragavi.R
Student Roll Number	613019104061
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 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 table list. The table list contains one entry: 'STUDENT' in schema 'JCH91892'. To the right, the 'Table definition' panel for 'STUDENT' is shown, detailing the columns: USERNAME (CHAR, 5, nullable), ROLLNUMBER (INTEGER, nullable), EMAIL ID (VARCHAR, 32, nullable), and PASSWORD (VARCHAR, 32, nullable). A 'View data' button is located at the bottom of the definition panel. The bottom of the image shows a Windows taskbar with various application icons and system status information (29°C, Haze, 22:49, 18-10-2022).

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

2. Perform UPDATE,DELETE Queries with user table.

UPDATE TABLE:-

The screenshot shows the IBM Db2 on Cloud console. On the left, under 'Data objects', the database 'KNH78128' is listed. The main area is titled '*Untitled - 1' and contains an SQL script: `1 INSERT INTO STUDENTS values ('Ragav','61','rragavi009@gmail.com','Rose')`. A 'Run all' button is visible. Below the script editor, a 'History' tab shows a table of executed scripts.

Script	Date	Status	Runtime
Untitled - 1	Oct 19, 2022 11:15:17 AM	✓ 1	0.006 s
INSERT INTO STUDENTS values ('Ragav','61','rragavi009@gmail.com...		✓	0.006 s

The screenshot shows the IBM Db2 on Cloud console with the 'Tables' tab selected. The table 'KNH78128.STUDENTS' is displayed. An 'Export to CSV' button is visible. The table data is as follows:

USERNAME	ROLL NUMBER	EMAIL ID	PASSWORD
sri	77	sri007@gmail.com	VCTW
sri	77	sri007@gmail.com	VCTW
sri	77	sri007@gmail.com	VCTW

DELETE TABLE:-

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

bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2F9210e99d13eb46d39e2e511afaf24...

IBM Db2 on Cloud

Data objects My script

Filter objects

KNH78128

*Untitled - 1 x +

Beta Classic

Syntax assistant

Run all

```
1 INSERT INTO STUDENTS values ('Ragav','61','rragavi009@gmail.com','Rose');
2 INSERT INTO STUDENTS values ('sri','77','sri007@gmail.com','VCTW');
3 DELETE FROM STUDENTS WHERE USERNAME = 'Ragav';
```

History

Find history

Script	Date	Status	Runtime
Untitled - 1	Oct 19, 2022 11:19:56 AM	3	0.016 s
INSERT INTO STUDENTS values ('Ragav','61','rragavi009@gmail.c...			0.005 s
INSERT INTO STUDENTS values ('sri','77','sri007@gmail.com','V...			0.004 s
DELETE FROM STUDENTS WHERE USERNAME = 'Ragav'			0.007 s
Untitled - 1	Oct 19, 2022 11:19:21 AM	2 1	0.021 s

30°C Mostly sunny

23:20 18-10-2022

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

bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2F9210e99d13eb46d39e2e511afaf24...

IBM Db2 on Cloud

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

KNH78128.STUDENTS

Back

Export to CSV

USERNAME	ROLL NUMBER	EMAIL ID	PASSWORD
sri	77	sri007@gmail.com	VCTW
sri	77	sri007@gmail.com	VCTW
sri	77	sri007@gmail.com	VCTW

30°C Mostly sunny

23:20 18-10-2022

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>Smart Fashion 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>Smart Fashion 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="style2.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%;
}

```

```

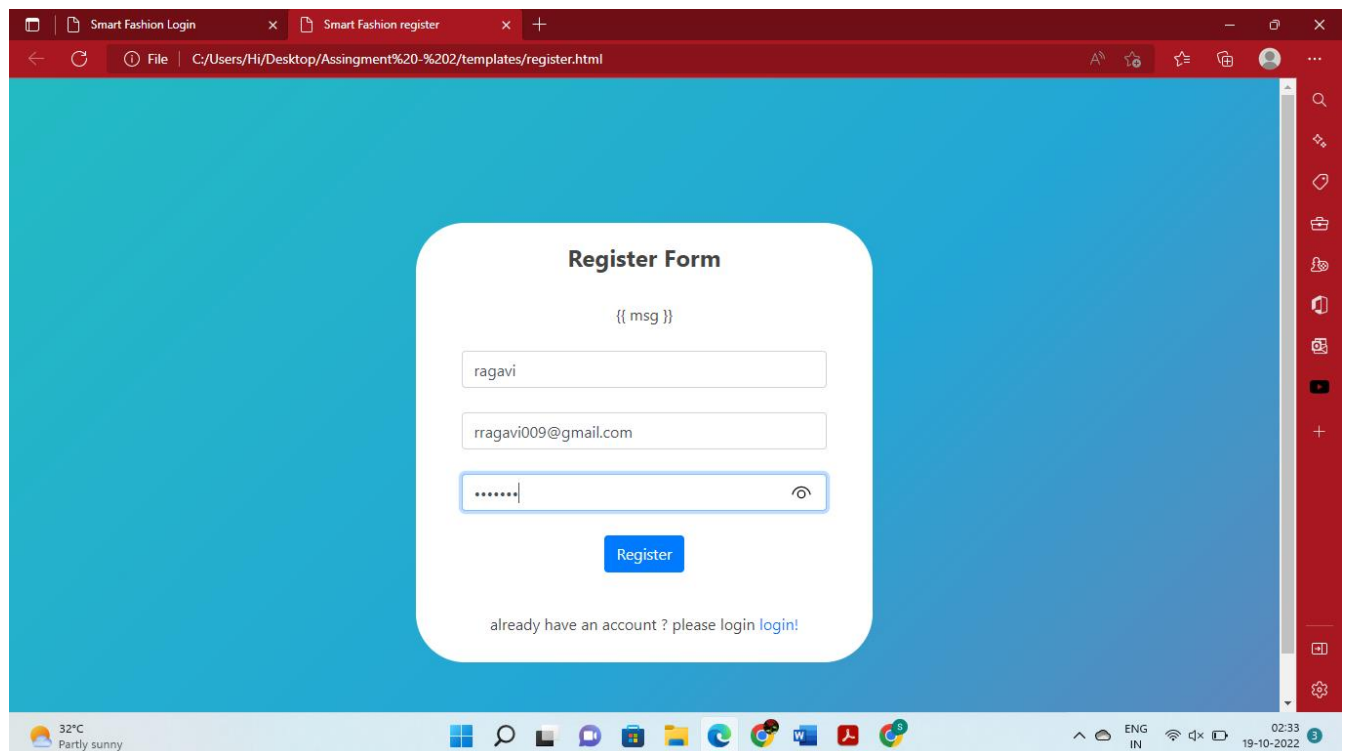
.form-text-vk {
  font-size: small;
}

```

```
padding-bottom: 10px;
}

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

OUTPUT:-
REGISTRATION PAGE:-




The screenshot displays a web browser window with two tabs: 'Smart Fashion Login' and 'Smart Fashion register'. The active tab shows a registration form titled 'Register Form' on a blue gradient background. The form is contained within a white rounded rectangle and includes a placeholder for a message, followed by input fields for 'ragavi', 'rragavi009@gmail.com', and a password field with a toggle icon. A blue 'Register' button is positioned below the inputs. At the bottom of the form, a link reads 'already have an account ? please login login!'. The browser's address bar shows the file path 'C:/Users/Hi/Desktop/Assingment%20-%202/templates/register.html'. The Windows taskbar at the bottom indicates a temperature of 32°C, 'Partly sunny' weather, and the date '19-10-2022' at '02:33'.

Smart Fashion Login x Smart Fashion register x +

File | C:/Users/Hi/Desktop/Assingment%20-%202/templates/register.html

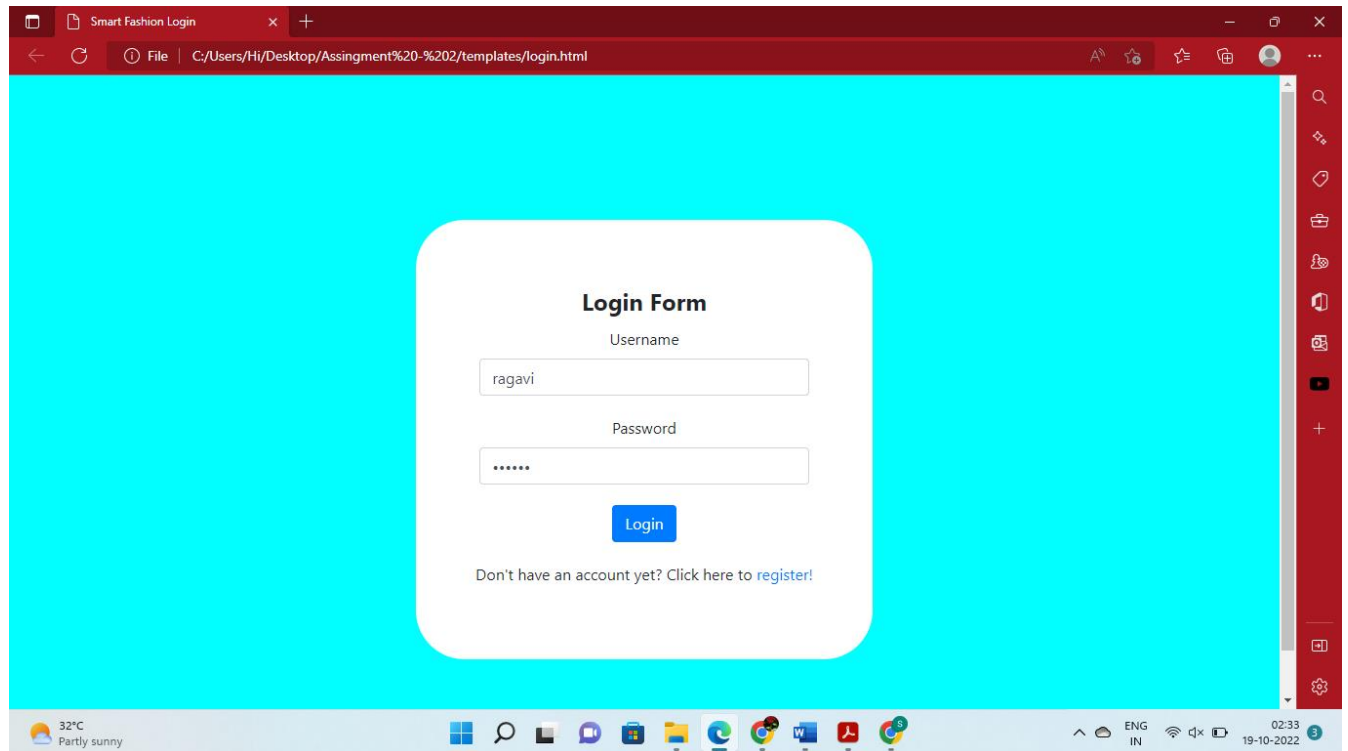
Register Form

{{ msg }}

[already have an account ? please login login!](#)

LOGIN PAGE:-



The screenshot shows a web browser window with the title 'Smart Fashion Login'. The address bar displays the file path 'C:/Users/Hi/Desktop/Assingment%20-%202/templates/login.html'. The page has a solid blue background. In the center, there is a white rounded rectangle containing the 'Login Form'. The form includes a 'Username' label above a text input field containing 'ragavi', a 'Password' label above a password input field with masked characters, and a blue 'Login' button. Below the button, a link reads 'Don't have an account yet? Click here to register!'. The browser's taskbar at the bottom shows the Windows logo, search icon, and various application icons. The system tray on the right indicates a temperature of 32°C, 'Partly sunny' weather, and the date '19-10-2022' at '02:33'.

Login Form

Username

ragavi

Password

Login

Don't have an account yet? Click here to register!

WELCOME PAGE:-

