

Assignment -2

IBM CLOUD DB2

Assignment Date	29 September 2022
Student Name	SWETHA.S
Student Roll Number	613019205050
Maximum Marks	2 Marks

Questions:

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

The screenshot displays the IBM Db2 on Cloud web interface. The 'Tables' tab is selected, showing a list of tables. A table named 'STUDENT' is listed under the schema 'XKT46690'. The 'Table definition' panel on the right shows the structure of the 'STUDENT' table:

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

The interface also includes a search bar for schemas and tables, a 'New table' button, and a 'View data' button at the bottom of the table definition panel.

- 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 'XKT46690' is selected. The main area shows a script editor with the following SQL statement:

```
1 INSERT INTO STUDENT values ('swe','50','swe@gmail.com','swe@');
```

Below the script editor, the 'History' tab is active, displaying a table of executed scripts:

Script	Date	Status	Runtime
Untitled - 1	Oct 19, 2022 9:03:30 AM	1	0.013 s
INSERT INTO STUDENT values ('swe','50','swe@gmail.com','sw...			0.013 s

The screenshot shows the IBM Db2 on Cloud console with the 'Tables' tab selected. The table 'XKT46690.STUDENT' is displayed. The table structure and data are as follows:

USERNAME	ROLL NUMBER	EMAIL ID	PASSWORD
swe	50	swe@gmail.com	swe@

DELETE TABLE:-

The screenshot shows the IBM Db2 on Cloud SQL editor interface. The left sidebar displays the 'Data objects' tab with a search filter and a list of objects, including 'XKT46690'. The main editor area shows a script titled 'Untitled - 1' with the following SQL statements:

```
1 INSERT INTO STUDENT values ('swe', '50', 'swe@gmail.com', 'swe@');
2 INSERT INTO STUDENT values ('kavi', '51', 'kavi@gmail.com', 'kavi@');
3 DELETE FROM STUDENT WHERE USERNAME = 'swe';
```

Below the script editor, the 'History' tab is active, showing a table of script execution history:

Script	Date	Status	Runtime
Untitled - 1	Oct 19, 2022 9:07:56 AM	3	0.024 s
INSERT INTO STUDENT values ('swe', '50', 'swe@gmail.com', 'swe@');		✓	0.011 s
INSERT INTO STUDENT values ('kavi', '51', 'kavi@gmail.com', 'kavi@');		✓	0.006 s
DELETE FROM STUDENT WHERE USERNAME = 'swe';		✓	0.007 s
Untitled - 1	Oct 19, 2022 9:03:30 AM	1	0.013 s

The screenshot shows the IBM Db2 on Cloud SQL editor interface with the 'Tables' tab selected. The table 'XKT46690.STUDENT' is displayed with the following structure and data:

USERNAME	ROLL NUMBER	EMAIL ID	PASSWORD
kavi	51	kavi@gmail.com	kavi@

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>Plasma donor application 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>Plasma Donor Application 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;

```

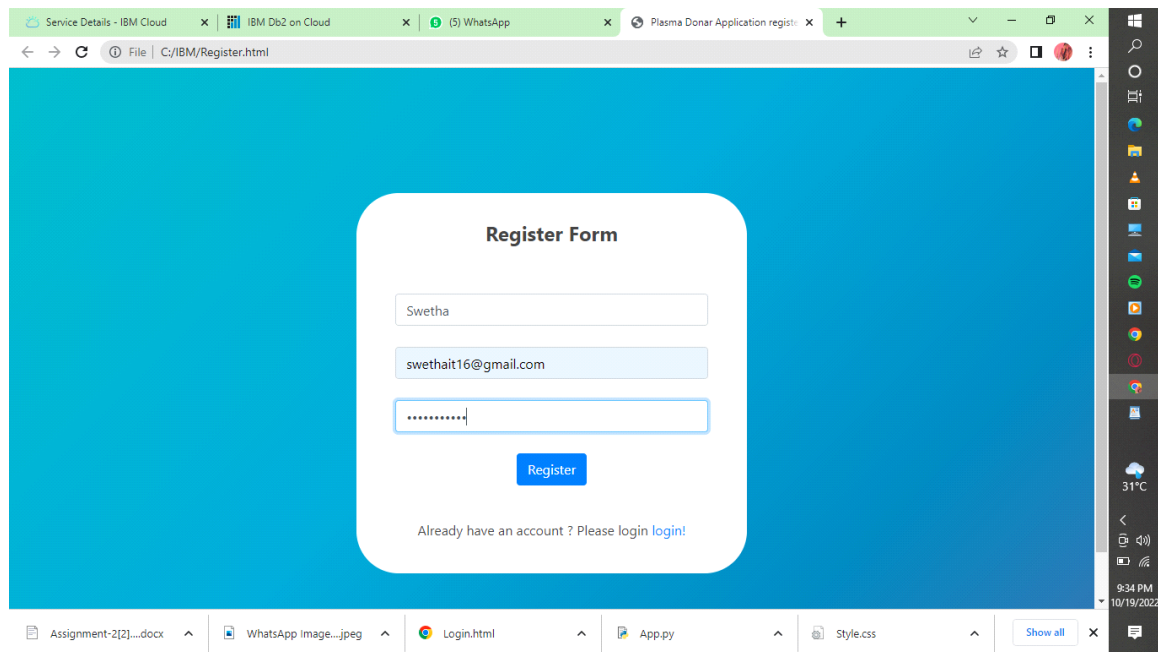
```
padding-bottom: 10px;
```

```
}
```

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

```
}
```

OUTPUT:- REGISTRATION PAGE:-



The screenshot displays a web browser window with multiple tabs. The active tab is titled "Plasma Donar Application regist...". The address bar shows the file path "C:/IBM/Register.html". The main content area features a "Register Form" on a blue gradient background. The form includes three input fields: a text field containing "Swetha", an email field containing "swethait16@gmail.com", and a password field with masked characters "*****". A blue "Register" button is positioned below the password field. At the bottom of the form, a link is provided for existing users: "Already have an account ? Please login [login!](#)". The browser's taskbar at the bottom shows several open files: "Assignment-2[2]....docx", "WhatsApp Image...jpeg", "Login.html", "App.py", and "Style.css". The system tray on the right indicates a temperature of 31°C and the date/time as 9:34 PM on 10/19/2022.

Service Details - IBM Cloud x IBM Db2 on Cloud x (5) WhatsApp x Plasma Donar Application regist... x

File | C:/IBM/Register.html

Register Form

Swetha

swethait16@gmail.com

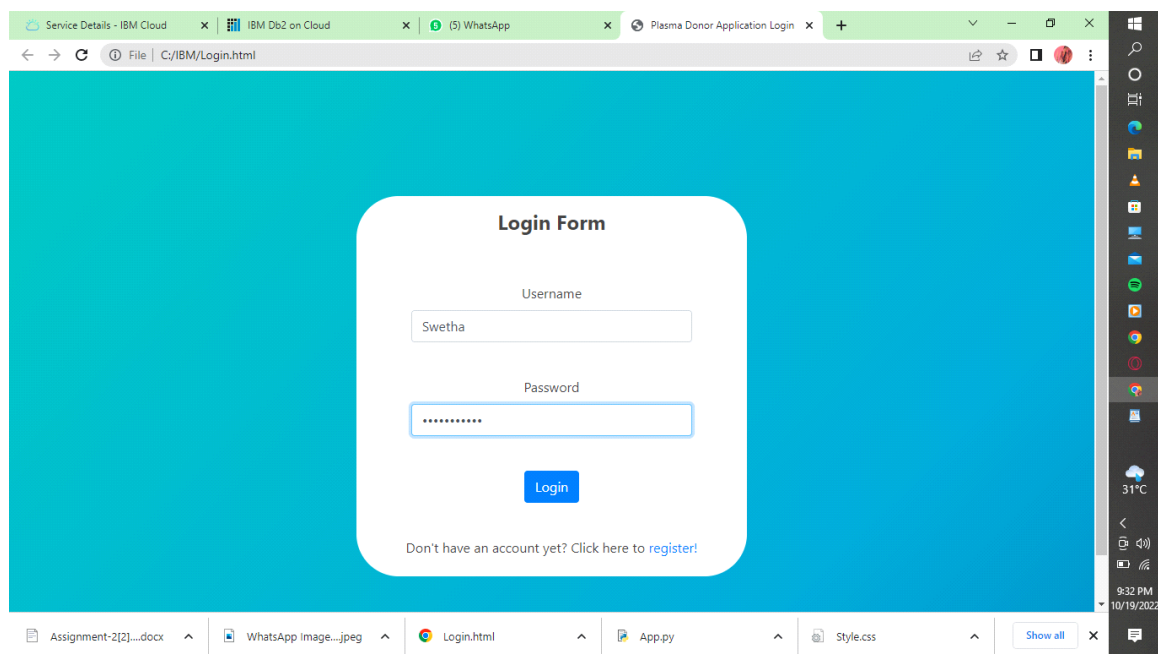
Register

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

Assignment-2[2]....docx WhatsApp Image...jpeg Login.html App.py Style.css Show all x

31°C 9:34 PM 10/19/2022

LOGIN PAGE:-



The screenshot displays a web browser window with multiple tabs. The active tab is titled "Plasma Donar Application Login...". The address bar shows the file path "C:/IBM/Login.html". The main content area features a "Login Form" on a blue gradient background. The form includes two input fields: a text field containing "Swetha" and a password field with masked characters "*****". A blue "Login" button is positioned below the password field. At the bottom of the form, a link is provided for new users: "Don't have an account yet? Click here to [register!](#)". The browser's taskbar at the bottom shows the same set of open files as the registration page: "Assignment-2[2]....docx", "WhatsApp Image...jpeg", "Login.html", "App.py", and "Style.css". The system tray on the right indicates a temperature of 31°C and the date/time as 9:32 PM on 10/19/2022.

Service Details - IBM Cloud x IBM Db2 on Cloud x (5) WhatsApp x Plasma Donar Application Login... x

File | C:/IBM/Login.html

Login Form

Username

Swetha

Password

Login

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

Assignment-2[2]....docx WhatsApp Image...jpeg Login.html App.py Style.css Show all x

31°C 9:32 PM 10/19/2022

WELCOME PAGE:-

