

## Assignment -2

### IBM CLOUD DB2

Assignment Date	29 September 2022
Student Name	Saravanan L
Student Roll Number	511319205025
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 console interface. The 'Tables' tab is selected, displaying a list of tables. A 'New table' button is visible. The 'Table definition' panel on the right shows the structure of the 'USER' table.

Name	Data type	Nullable	Length	Scale
USER NAME	CHAR	Y	21	0
ROLL NUMBER	INTEGER	Y		0
EMAIL	CHAR	Y	18	0
PASSWORD	VARCHAR	Y	32	0

View data

## 2.Perform UPDATE,DELETE Queries with user table

### UPDATETABLE:-

The screenshot shows the IBM Db2 on Cloud console. On the left, the 'Data objects' panel shows a database named 'QYS10826'. The main area displays a script editor with a single SQL statement: `INSERT INTO USER VALUES('SARAN','25','SARAN@gmail.com','5678')`. Below the script editor, the 'History' tab shows a table of executed queries.

Script	Date	Status	Runtime
Untitled - 1	Oct 25, 2022 10:21:44 PM	✓ 1	0.011 s
INSERT INTO USER VALUES('SARAN','25','SARAN@gmail.com','5678')		✓	0.011 s

The screenshot shows the IBM Db2 on Cloud console with the 'Tables' tab selected. The table 'QYS10826.USER' is displayed. The table has four columns: USERNAME, ROLLNUMBER, EMAIL, and PASSWORD. The data is as follows:

USERNAME	ROLLNUMBER	EMAIL	PASSWORD
KAVI	16	KAVI@gmail.com	3478
SARAN	25	SARAN@gmail.com	6789

### DELETETABLE:-

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

bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/cm%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2F89b4e7c161d849b0aec31c59a6535ed0%3Ab64da145-3282-4c...

IBM Db2 on Cloud

Data objects My script

Filter objects

QYS10826

\*Untitled - 1 x +

1 DELETE FROM USER WHERE ROLLNUMBER = 16

Syntax assistant Run all

History

Find history

Script	Date	Status	Runtime
Untitled - 1	Oct 25, 2022 10:55:34 PM	✓ 1	0.009 s
DELETE FROM USER WHERE ROLLNUMBER = 16		✓	0.009 s
Untitled - 1	Oct 25, 2022 10:53:05 PM	✓ 1	0.016 s
INSERT INTO USER VALUES('KAVI','16','KAVI@gmail.com','3478')		✓	0.016 s
Untitled - 1	Oct 25, 2022 10:48:45 PM	✓ 1	0.009 s

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

bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/cm%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2F89b4e7c161d849b0aec31c59a6535ed0%3Ab64da145-3282-4c...

IBM Db2 on Cloud

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

QYS10826.USER

Export

USERNAME	ROLLNUMBER	EMAIL	PASSWORD
SARAN	25	SARAN@gmail.com	6789

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=bludb;HOSTNAME=1bbf73c5-d84a-4bb0-85b9-
ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;PORT=32286;SECURITY=SSL;SSL
ServerCertificate=DigiCertGlobalRootCA.crt;UID=qys10826;PWD=fXqq3tiLqGLZhx8",",")

@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 user 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 user 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 user 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="\static\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="\static\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 {

    background-image:url(https://simpleassignment1234.s3.jp-tok.cloud-object-
storage.appdomain.cloud/image%20bucket.jpeg);

    background-color: #cccccc;

    display: flex;

    flex-direction: column;

    align-items: center;

    justify-content: center;

}

form {

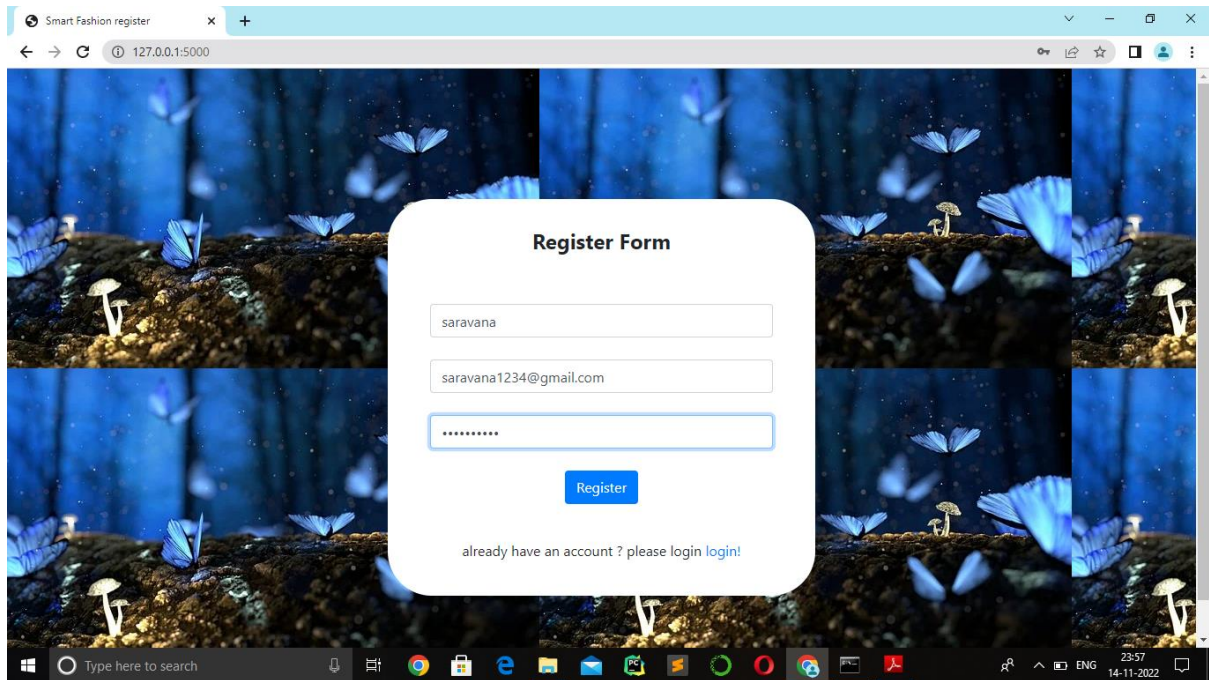
```

```
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;  
}
```

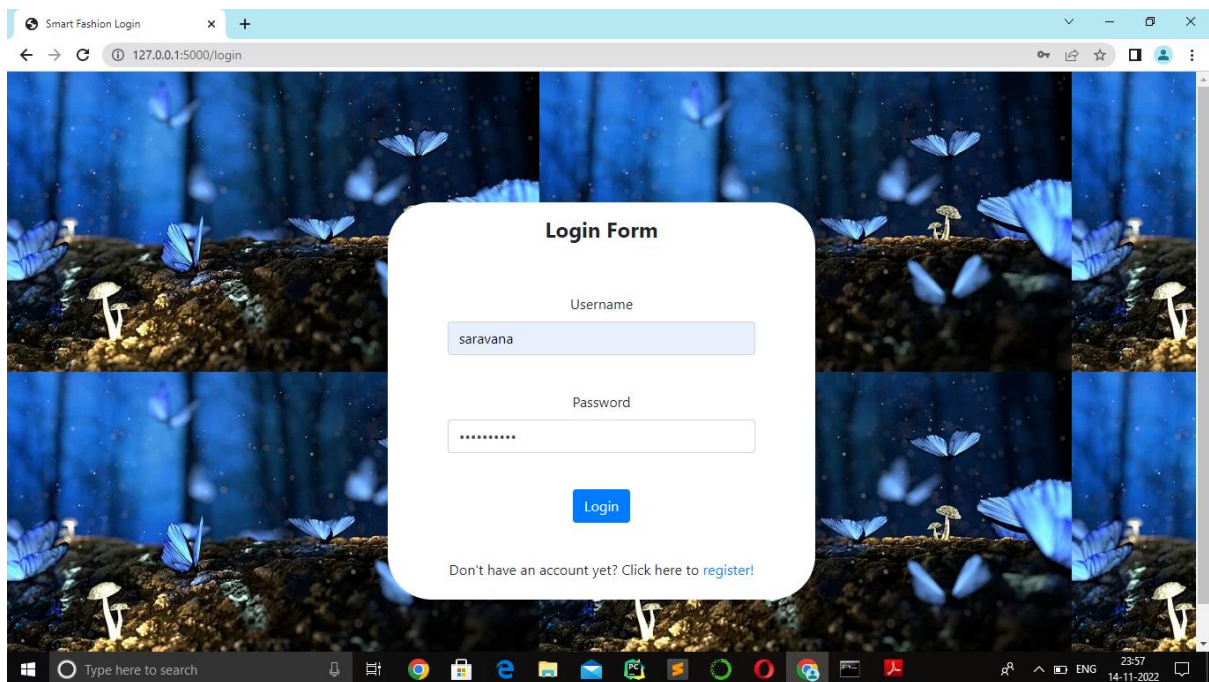
## Output:

### Registration Page:



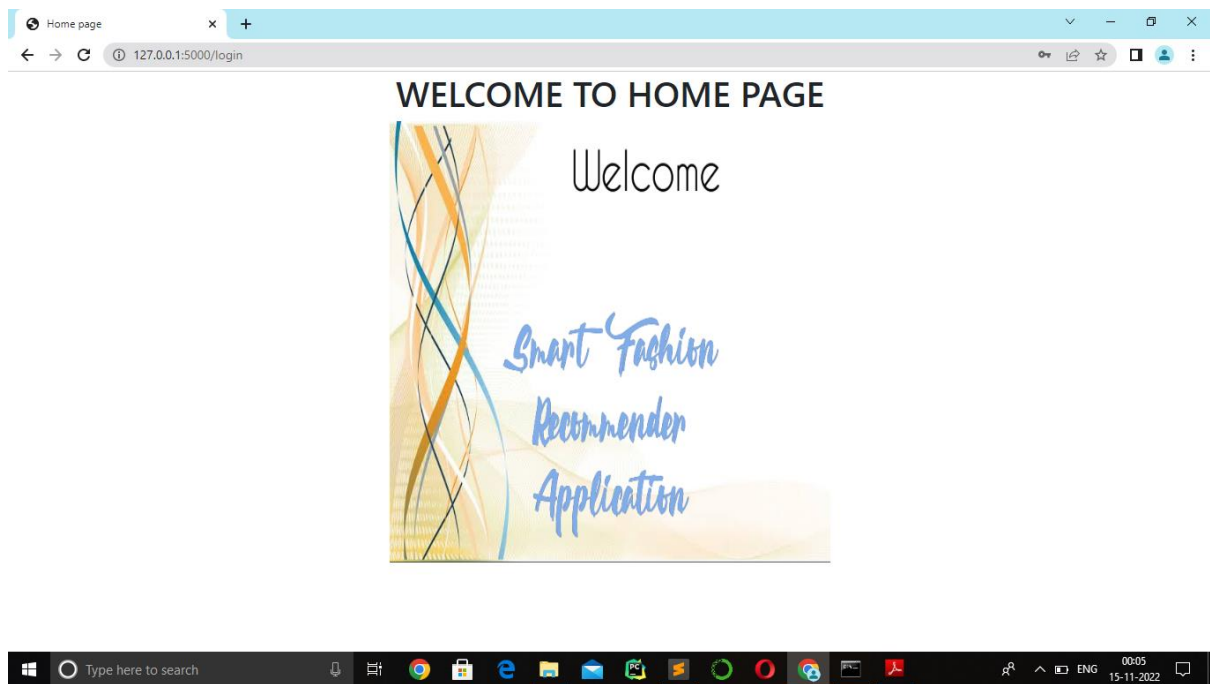
The screenshot shows a web browser window titled "Smart Fashion register" with the URL "127.0.0.1:5000". The background is a dark, blue, forest-like scene with many glowing blue butterflies and small white mushrooms on the ground. In the center, there is a white rounded rectangle titled "Register Form". Inside the form, there are three input fields: the first contains "saravana", the second contains "saravana1234@gmail.com", and the third is a password field with masked characters "\*\*\*\*\*". Below the input fields is a blue button labeled "Register". At the bottom of the form, there is a link: "already have an account ? please login [login!](#)". The Windows taskbar is visible at the bottom of the browser window.

### Login Page:



The screenshot shows a web browser window titled "Smart Fashion Login" with the URL "127.0.0.1:5000/login". The background is the same dark, blue, forest-like scene with glowing blue butterflies and small white mushrooms. In the center, there is a white rounded rectangle titled "Login Form". Inside the form, there are two input fields: the first is labeled "Username" and contains "saravana", the second is labeled "Password" and contains masked characters "\*\*\*\*\*". Below the input fields is a blue button labeled "Login". At the bottom of the form, there is a link: "Don't have an account yet? Click here to [register!](#)". The Windows taskbar is visible at the bottom of the browser window.

## Welcome Page:



## Data Base Table:

