

## ASSIGNMENT- 2

Assignment Date	26 September 2022
Student Name	Ms. Anchana P
Student Roll Number	311519104009
Maximum Marks	2 Marks

### 1) CREATE USER TABLE WITH USER WITH EMAIL, USERNAME, ROLL NUMBER, PASSWORD

OUTPUT:

The screenshot displays the IBM Db2 on Cloud web interface. The 'Tables' tab is selected, showing a list of tables. A table named 'USER' is listed under the schema 'FPN02497'. The 'Table definition' panel on the right shows the table's structure:

Name	Data type	Nullable	Length	Scale
USERNAME	VARCHAR	Y	50	0
ROLLNO	VARCHAR	Y	50	0
EMAIL	VARCHAR	Y	50	0
PASSWORD	VARCHAR	Y	50	0

The interface also shows a search bar, a 'New table' button, and a 'View data' button. The bottom status bar indicates the system is at 31°C Cloudy, with a date of 13-10-2022 and time of 19:46.

IBM Db2 on Cloud

Data objects My script

Filter objects

FPN02497

SQL

Untitled - 1

```
1 INSERT INTO USER VALUES('ANCHANA','311519104009','anchana0521@gmail.com','Anchana21');
2 INSERT INTO USER VALUES('ANCHU','311519105009','anchana0521@gmail.com','Anchana21');
3 INSERT INTO USER VALUES('ANJU','311519105009','anchana0521@gmail.com','Anchana21');
4
5
6
7
8
9
```

History Results

Find history

Script	Date	Status	Runtime
Untitled - 1	Oct 13, 2022 7:52:09 PM	3	0.068 s
INSERT INTO USER VALUES('ANCHANA','311519104009','anchana0521@gmail.com','An...			0.020 s
INSERT INTO USER VALUES('ANCHU','311519105009','anchana0521@gmail.com','Anch...			0.020 s
INSERT INTO USER VALUES('ANJU','311519105009','anchana0521@gmail.com','Ancha...			0.020 s

31°C Cloudy

IBM Db2 on Cloud

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

FPN02497.USER

Back

Export to CSV

USERNAME	ROLLNO	EMAIL	PASSWORD
ANCHANA	311519104009	anchana0521@gmail.com	Anchana21
ANCHU	311519105009	anchana0521@gmail.com	Anchana21
ANJU	311519105009	anchana0521@gmail.com	Anchana21

31°C Cloudy

## 2) PERFORM UPDATE,DELETE QUERIES WITH USER TABLE

### OUTPUT:

### UPDATE:

The screenshot displays the IBM Db2 on Cloud web interface. The top section shows a SQL editor with the following query:

```
UPDATE USER SET ROLLNO=09 WHERE USERNAME = 'ANCHU';
```

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

Script	Date	Status	Runtime
Untitled - 1	Oct 13, 2022 7:53:04 PM	✓ 1	0.005 s
UPDATE USER SET ROLLNO=09 WHERE USERNAME = 'ANCHU'		✓	0.005 s
Untitled - 1	Oct 13, 2022 7:52:09 PM	✓ 3	0.060 s
INSERT INTO USER VALUES('ANCHANA','311519104009','anchana0521@gmail.com','A_		✓	0.020 s
INSERT INTO USER VALUES('ANCHU','311519105009','anchana0521@gmail.com','Anc		✓	0.020 s

The bottom section shows the 'Tables' view for the 'FPN02497.USER' database. It displays a table with the following data:

USERNAME	ROLLNO	EMAIL	PASSWORD
ANCHANA	311519104009	anchana0521@gmail.com	Anchana21
ANCHU	9	anchana0521@gmail.com	Anchana21
ANJU	311519105009	anchana0521@gmail.com	Anchana21

## DELETE:

The screenshot displays the IBM Db2 on Cloud web interface. The top navigation bar includes tabs for 'Post Attendee - Zoom', 'Service Details - IBM', 'IBM Db2 on Cloud', '(32) WhatsApp', 'IBM', 'Fwd: IBM assignment', and '(112 unread) - anchi'. The browser address bar shows the URL: `bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2F2fd65151bd3e43a09c89a9792bdea2b6%3Aa1378646-20f...`. The interface is divided into two main sections: the top section for SQL execution and the bottom section for data results.

**Top Section: SQL Execution**

The left sidebar shows 'Data objects' with a search filter 'Filter objects' and a list of objects including 'FPN02497'. The main area displays a script editor with the following SQL command:

```
1 DELETE FROM USER WHERE USERNAME='ANCHU';
2
3
4
5
```

The 'History' tab is active, showing a table of executed scripts:

Script	Date	Status	Runtime
Untitled - 1	Oct 13, 2022 7:54:19 PM	✓ 1	0.028 s
DELETE FROM USER WHERE USERNAME='ANCHU'		✓	0.028 s
Untitled - 1	Oct 13, 2022 7:53:04 PM	✓ 1	0.005 s
UPDATE USER SET ROLLNO=09 WHERE USERNAME = 'ANCHU'		✓	0.005 s
Untitled - 1	Oct 13, 2022 7:52:09 PM	✓ 3	0.060 s

**Bottom Section: Data Results**

The left sidebar shows 'Tables' with a search filter 'Filter objects' and a list of tables including 'FPN02497.USER'. The main area displays a table of user data:

USERNAME	ROLLNO	EMAIL	PASSWORD
ANCHANA	311519104009	anchana0521@gmail.com	Anchana21
ANJU	311519105009	anchana0521@gmail.com	Anchana21

The interface also includes a 'Back' button and an 'Export to CSV' button. The bottom status bar shows the temperature as 31°C, the time as 19:54, and the date as 13-10-2022.

### **3) CONNECT PYTHON CODE TO DB2**

**AND**

- 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.**

#### **PROGRAM:**

**reg2.html**

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

```
<style>
```

```
body{
```

```
    font-family: Calibri, Helvetica, sans-serif;
```

```
        background-color:#E8DAEF;
```

```
}
```

```
.container {
```

```
    padding: 50px;
```

```
    height:500px;
```

```
    width:500px;
```

```
    background-color: #D2B4DE;
```

```
    align-items: center;
```

```
    margin-left:500px;
```

```
}
```

```
input[type=text], input[type=password], textarea {
```

```
    width: 100%;
```

```
    padding: 15px;
```

```
    margin: 5px 0 22px 0;
```

```
display: inline-block;
border: none;
background: #f1f1f1;
}
input[type=text]:focus, input[type=password]:focus {
background-color: orange;
outline: none;
}
div {
padding: 10px 0;
}
hr {
border: 1px solid #f1f1f1;
margin-bottom: 25px;
}
.registerbtn {
background-color: #5B2C6F;
color: white;
padding: 16px 20px;
margin: 8px 0;
border: none;
cursor: pointer;
width: 100%;
opacity: 0.9;
}
.registerbtn:hover {
opacity: 1;
}
</style>
</head>
```

```
<body>
  <center>
    <h3><b>REGISTRATION</b></h3>
    <form action = "http://localhost:5000/reg2" method = "POST">

      <div class="container">
        <hr>

        <label> <b>Enter Register No.: </b> </label>
        <input type="text" name="rollno" placeholder= "REG. NO" size="15"
required />

        <label> <b>Enter Email ID: </b> </label>
        <input type="text" name="email" placeholder= "EMAIL ID" size="15"
required />

        <label> <b>Enter Username: </b> </label>
        <input type="text" name="uname" placeholder= "USER NAME"
size="15" required />

        <label> <b>Enter Password: </b> </label>
        <input type="password" name="pwd" placeholder= "PASSWORD"
size="15" required />

        <button type="submit" class="registerbtn">Register</button>
      </div>
    </form>
  </center>
</body>
</html>
```

## login.html

```
<!DOCTYPE html>

<html>

<head>

<meta name="viewport" content="width=device-width, initial-scale=1">

<style>

body{

    font-family: Calibri, Helvetica, sans-serif;

        background-color:#E8DAEF;

}

.container {

    padding: 50px;

    height:500px;

    width:500px;

    background-color: #D2B4DE;

    align-items: center;

    margin-left:500px;

}

input[type=text], input[type=password], textarea {

    width: 100%;

    padding: 15px;

    margin: 5px 0 22px 0;

    display: inline-block;

    border: none;

    background: #f1f1f1;

}

input[type=text]:focus, input[type=password]:focus {

    background-color: orange;
```



```
        outline: none;
    }
    div {
        padding: 10px 0;
    }
    hr {
        border: 1px solid #f1f1f1;
        margin-bottom: 25px;
    }
    .registerbtn {
        background-color: #5B2C6F;
        color: white;
        padding: 16px 20px;
        margin: 8px 0;
        border: none;
        cursor: pointer;
        width: 100%;
        opacity: 0.9;
    }
    .registerbtn:hover {
        opacity: 1;
    }
</style>
</head>
<body>
    <center> <h1> Login Form</h1> </center>
    <form action = "http://localhost:5000/login" method = "POST">
    <div class="container">
    <hr>
```

```

<label> <b>Enter Username: </b> </label>
<input type="text" name="uname" placeholder= "Username" size="15" required />

<label><b>Enter Password: </b></label>
<input type="password" placeholder="Password" name="pwd" required>

<button type="submit" class="registerbtn">Login</button>
</div>
</form>
</body>
</html>

```

### **Python file:**

```
# -*- coding: utf-8 -*-
```

```
"""
```

```
Created on Thu Oct 13 20:03:42 2022
```

```
@author: Anchana P
```

```
"""
```

```

from flask import Flask,render_template, redirect, url_for, request, session
import ibm_db
import re

```

```

app=Flask(__name__)
app.secret_key='a'

```

```

conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=fbd88901-ebdb-4a4f-
a32e-

```

```
9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32731;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=fpn02497;PWD=orQlwu8rjdyzyQSd;"",")
```

```
@app.route('/')
```

```
def home():
```

```
    return render_template('reg2.html')
```

```
@app.route('/login',methods=["GET","POST"])
```

```
def login():
```

```
    global userid
```

```
    msg=""
```

```
    if request.method=="POST":
```

```
        username = request.form['uname']
```

```
        password = request.form['pwd']
```

```
        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!'
```

```
            return redirect(url_for('welcome', username=username))
```

```

        else: msg = "Incorrect Username/Password"
    return render_template('login.html', msg = msg)

@app.route('/reg2',methods=["GET","POST"])
def registration():
    msg = " "

    if request.method=="POST":
        username = request.form['uname']
        email = request.form['email']
        password = request.form['pwd']
        rollno = request.form['rollno']
        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 = "Username must contain only alphabets 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,rollno)
            ibm_db.bind_param(prepare_stmt,3,email)

```

```
ibm_db.bind_param(prepare_stmt,4,password)

ibm_db.execute(prepare_stmt)

msg = "You have successfully registered."

return render_template('login.html', msg=msg)
```

```
elif request.method == 'POST': msg="Please fill out the form."

return render_template('reg2.html',msg=msg)
```

```
@app.route('/welcome/<username>')

def welcome(username):

    return "Welcome %s!" %username
```

```
if __name__=="__main__":

    app.run(host='0.0.0.0')
```

## OUTPUT:

The screenshot displays a web browser window with multiple tabs. The active tab shows a registration form titled "REGISTRATION" on a purple background. The form contains the following fields and values:

- Enter Register No.:** 311519104012
- Enter Email ID:** dhana@gmail.com
- Enter Username:** Dhana
- Enter Password:** \*\*\*\*

A purple "Register" button is located at the bottom of the form. The browser's address bar shows "localhost:5000". The Windows taskbar at the bottom indicates a temperature of 32°C and the date 16-10-2022.



