

Assignment -2

Connect with Database Assignment

Assignment Date	09 October 2022
Student Name	Ramesh S
Student Roll Number	621319104043
Maximum Marks	2 Marks

Question-1:

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

The screenshot shows a SQL IDE interface with a toolbar at the top containing icons for file operations, editing, and execution. A 'Run all' button is visible. The main editor area contains a single SQL statement: `CREATE TABLE USER (EMAIL VARCHAR(30), USERNAME CHAR(20), ROLLNO INT, PASSWORD VARCHAR(100));`. Below the editor is a 'History' panel with a search bar and a table listing executed scripts.

Script	Date	Status	Runtime
Untitled - 1	Oct 9, 2022 1:14:27 PM	✓ 1	0.078 s
CREATE TABLE USER (EMAIL VARCHAR(30), USERNAME CHAR(20), ROLLNO INT, PASSWOR...		✓	0.078 s

The screenshot shows a database viewer interface. At the top, it displays 'PHB43134.USER'. Below this, the table structure is shown with four columns: EMAIL, USERNAME, ROLLNO, and PASSWORD. A 'Back' button is located in the top right corner.

2. Perform UPDATE, DELETE Queries with User table

The screenshot shows a SQL IDE interface with two SQL statements in the editor: `INSERT INTO USER VALUES('abc@gmail.com','RAM','6211','Abc@ibm');` and `INSERT INTO USER VALUES('def@gmail.com','sita','6212','Def@ibm');`. The 'Run all' button is visible. Below the editor is a 'History' panel with a search bar and a table listing executed scripts.

Script	Date	Status	Runtime
Untitled - 1	Oct 14, 2022 10:14:42 AM	✓ 2	0.018 s
INSERT INTO USER VALUES('abc@gmail.com','RAM','6211','Abc@ibm')		✓	0.010 s
INSERT INTO USER VALUES('def@gmail.com','sita','6212','Def@ibm')		✓	0.008 s

Table View:

USERNAME	ROLLNUMBER	PASSWORD
ash@gmail.com	6211	Ash@item
shf@gmail.com	6212	Shf@item

UPDATE:

Script	Date	Status	Duration
UPDATE USER SET USERNAME='TRACK' WHERE USERNAME='ash@gmail.com'	04/18/2022 10:22:28 AM	Success	0.002 s
UPDATE USER SET USERNAME='TRACK' WHERE USERNAME='ash@gmail.com'	04/18/2022 10:22:28 AM	Success	0.002 s
UPDATE USER SET USERNAME='TRACK' WHERE USERNAME='ash@gmail.com'	04/18/2022 10:22:28 AM	Success	0.002 s

Table View:

USERNAME	ROLLNUMBER	PASSWORD
ash@gmail.com	6211	Ash@item
shf@gmail.com	6212	Shf@item

DELETE:

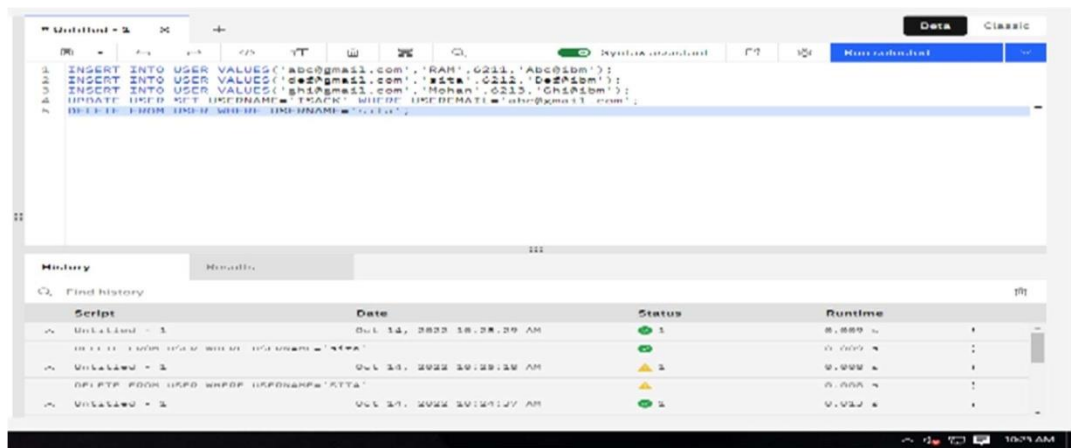
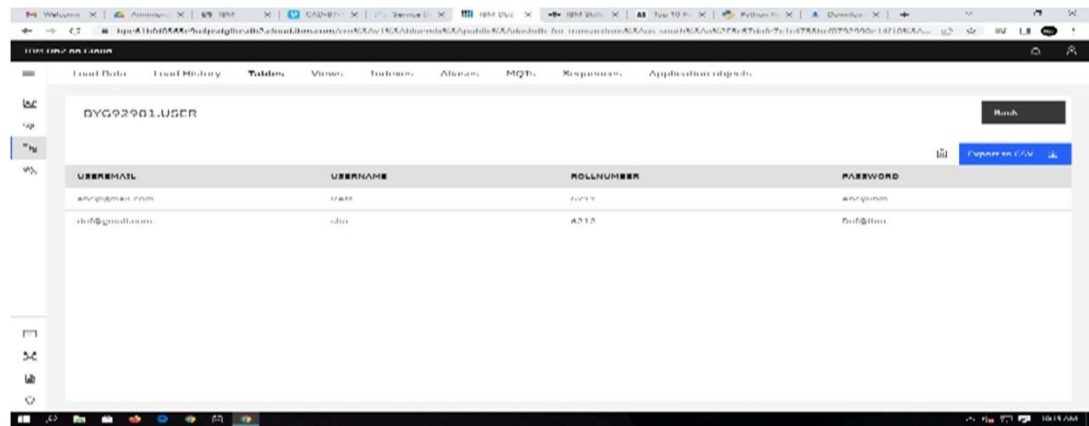


TABLE View:



3. Connect python with db2.

Solution:

```
import ibm_db
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME= 1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0
nqnrk39u98g.databases.appdomain.cloud;PORT=32286;SECURITY=SSL;SSLS     erverCertificate=DigiCertGl
obalRootCA.crt;PROTOCOL=TCPIP;UID= byg92981;PWD=" 9jZpv8EpbeEMaB6i",',')

```

4. Create a flask app with the registration page. Login page and the 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 so the welcome page.

Solution:

app.py

```
from flask import Flask, render_template, request, redirect, url_for, session

```

```
import ibm_db
import bcrypt
conn =
ibm_db.connect("DATABASE=bludb;HOSTNAME=;PORT=;SECURITY=SSL;SSLServerCertificate=DigiCer
tGlobalRootCA.crt;UID=;PWD=",",")

```

```
# url_for('static', filename='style.css')

```

```
app = Flask(__name_)
app.secret_key = 'C21FGSBAPOK43K5VSIDFB2'

```

```
@app.route("/",methods=['GET'])
def home():
    if 'email' not in session:
        return redirect(url_for('login'))
    return render_template('home.html',name='Home')

```

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

```

```

def register():
    if request.method == 'POST':
        email = request.form['email']
        username = request.form['username']
        rollNo = request.form['rollNo']
        password = request.form['password']

        if not email or not username or not rollNo or not password:
            return render_template('register.html',error='Please fill all fields')

        hash=bcrypt.hashpw(password.encode('utf-8'),bcrypt.gensalt())

        query = "SELECT * FROM USER WHERE email=? OR rollNo=?"
        stmt = ibm_db.prepare(conn, query)
        ibm_db.bind_param(stmt,1,email)
        ibm_db.bind_param(stmt,2,rollNo)
        ibm_db.execute(stmt)
        isUser = ibm_db.fetch_assoc(stmt)

        if not isUser:
            insert_sql = "INSERT INTO User(username,email,PASSWORD,rollNo) 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, hash)
            ibm_db.bind_param(prepare_stmt, 4, rollNo)
            ibm_db.execute(prepare_stmt)
            return render_template('register.html',success="You can login")
        else:
            return render_template('register.html',error='Invalid Credentials')

    return render_template('register.html',name='Home')

@app.route("/login",methods=['GET','POST'])
def login():
    if request.method == 'POST':
        email = request.form['email']
        password = request.form['password']

        if not email or not password:
            return render_template('login.html',error='Please fill all fields')
        query = "SELECT * FROM USER WHERE email=?"
        stmt = ibm_db.prepare(conn, query)
        ibm_db.bind_param(stmt,1,email)
        ibm_db.execute(stmt)
        isUser = ibm_db.fetch_assoc(stmt)
        print(isUser,password)

```

```

if not isUser:
    return render_template('login.html',error='Invalid Credentials')

isPasswordMatch = bcrypt.checkpw(password.encode('utf-8'),isUser['PASSWORD'].encode('utf-8'))

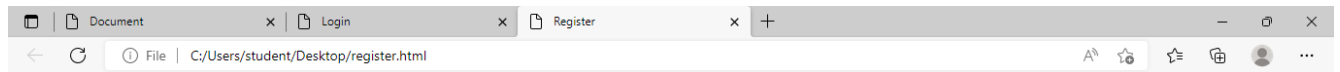
if not isPasswordMatch:
    return render_template('login.html',error='Invalid Credentials')

session['email'] = isUser['EMAIL']
return redirect(url_for('home'))

return render_template('login.html',name='Home')
@app.route('/logout')
def logout():
    session.pop('email', None)
    return redirect(url_for('login'))

```

OUTPUT:



Register

{{success}}

{{error}}

[Already have an account? Login](#)



Login

rameshselvam17033@gmail.com

.....

Login

{{success}}

{{error}}

[Don't have an account? Register](#)



WELCOME

You are logged in...

Logout



Database: