

## Assignment -2

### Python DB2

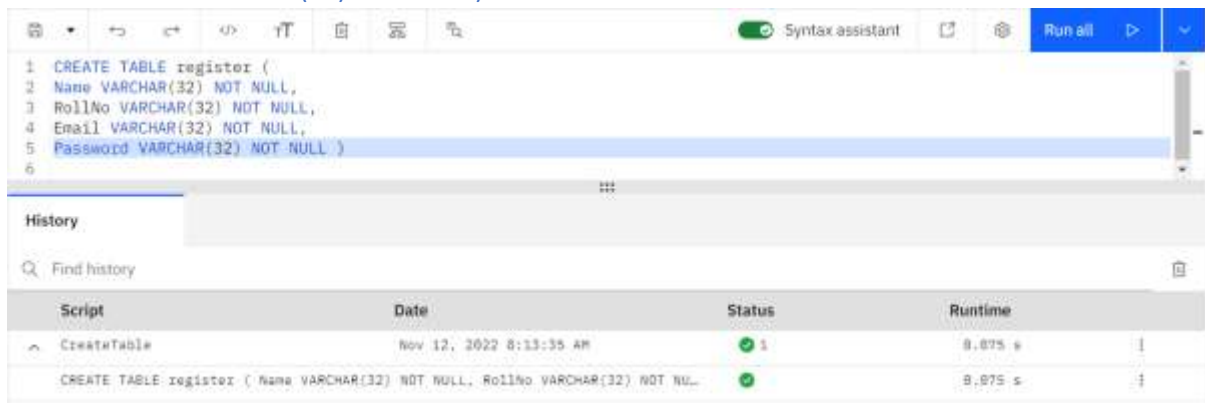
Student Name	Dilip ST
Student Roll Number	1901035
Maximum Marks	2 Marks

#### Question-1:

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

#### Solution:

```
CREATE TABLE register (  
Name VARCHAR (32) NOT NULL,  
RollNo VARCHAR (32) NOT NULL,  
Email VARCHAR (32) NOT NULL,  
Password VARCHAR (32) NOT NULL )
```



#### Question-2:

Perform UPDATE, DELETE Queries with user table.

#### Solution:

```
INSERT INTO register VALUES ('Abishek PS','1901002','abishek@gmail.com','Abi@001');  
INSERT INTO register VALUES ('Aditya Kushwaha','1901004','aditya@gmail.com','Adi@002');  
INSERT INTO register VALUES ('Arjun','1901016','arjun@gmail.com','Arj@003');  
INSERT INTO register VALUES ('Ashwin','1901023','ashwin@gmail.com','Ash@004');  
UPDATE register SET Name='Ashwin Balaji PL' WHERE RollNo='1901023';  
DELETE FROM register WHERE RollNo='1901004';
```

The screenshot displays a database IDE interface with two panels. The top panel shows a script editor with four INSERT statements. The bottom panel shows a history table with details of the executed script.

**Script Editor (Top Panel):**

```

1 INSERT INTO register VALUES('Abishek P5','1901002','abishek@gmail.com','Abi@001');
2 INSERT INTO register VALUES('Aditya Kushmaha','1901004','aditya@gmail.com','Adi@002');
3 INSERT INTO register VALUES('Arjun','1901016','arjun@gmail.com','Arj@003');
4 INSERT INTO register VALUES('Ashwin','1901023','ashwin@gmail.com','Ash@004');

```

**History Table (Bottom Panel):**

Script	Date	Status	Runtime
Insert	Nov 12, 2022 8:15:11 AM	✓ 4	0.091 s
INSERT INTO register VALUES('Abishek P5','1901002','abishek@gmail.com','Abi@001');		✓	0.011 s
INSERT INTO register VALUES('Aditya Kushmaha','1901004','aditya@gmail.com','Adi@002');		✓	0.000 s
INSERT INTO register VALUES('Arjun','1901016','arjun@gmail.com','Arj@003');		✓	0.039 s
INSERT INTO register VALUES('Ashwin','1901023','ashwin@gmail.com','Ash@004');		✓	0.033 s

The bottom panel also shows a second script editor with two statements:

```

1 UPDATE register SET Name='Ashwin Balaji PL' WHERE RollNo='1901023';
2 DELETE FROM register WHERE RollNo='1901004';

```

**History Table (Bottom Panel):**

Script	Date	Status	Runtime
Untitled - 1	Nov 12, 2022 8:20:18 AM	✓ 2	0.019 s
UPDATE register SET Name='Ashwin Balaji PL' WHERE RollNo='1901023';		✓	0.011 s
DELETE FROM register WHERE RollNo='1901004';		✓	0.008 s

### Question-3:

Connect python code to db2.

### Solution:

```

import ibm_db
dsn_hostname = "HOSTNAME"
dsn_uid = "UID"
dsn_pwd = "PASSWORD"
dsn_driver = "{IBM DB2 ODBC DRIVER}"
dsn_database = "DATABASE"
dsn_port = "PORTNUMBER"
dsn_protocol = "TCP/IP"
dsn_security = "SSL"
dsn = (
"DRIVER={0};""DATABASE={1};""HOSTNAME={2};""PORT={3};""PROTOCOL={4};""UID={5};""PWD={6};
""SECURITY={7};").format(dsn_driver,dsn_database,dsn_hostname,dsn_port,dsn_protocol,dsn_uid,dsn_pwd,dsn_security,)
print(dsn)
try:

```

```

conn = ibm_db.connect(dsn, "", "")
print ("Connected to database: ", dsn_database, "as user: ", dsn_uid, "on host: ",
dsn_hostname)
except:
    print ("Unable to connect: ", ibm_db.conn_errormsg() )

```

```

PS C:\Users\Abby\Desktop\NalayaThiran\Assignment_II> & C:/Users/Abby/AppData/Local/Programs/Python/Python310/python.exe c:/Users/Abby/Desktop/NalayaThiran/Assignment_II/db2.py
DRIVER={IBM DB2 ODBC DRIVER};DATABASE=bludb;HOSTNAME=55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kb10d8lcg.databases.appdomain.cloud;PORT=31929;PROTOCOL=TCP;UID=gtm67329;PWD=gPwP7XVrh6tNBWF;SECURITY=SSL;
Connected to database: bludb as user: gtm67329 on host: 55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kb10d8lcg.databases.appdomain.cloud
PS C:\Users\Abby\Desktop\NalayaThiran\Assignment_II>

```

#### Question-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.

#### Solution:

##### Main.py

```

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

app = Flask(__name__)
app.secret_key='a'
dsn_hostname = "*****"
dsn_uid = "*****"
dsn_pwd = "*****"
dsn_driver = "{IBM DB2 ODBC DRIVER}"
dsn_database = "*****"
dsn_port = "*****"
dsn_protocol = "TCP"
dsn_security = "SSL"
dsn =
("DRIVER={0};""DATABASE={1};""HOSTNAME={2};""PORT={3};""PROTOCOL={4};""UID={5};""
PWD={6};""SECURITY={7};").format(dsn_driver,dsn_database,dsn_hostname,dsn_port,dsn_p
rotocol,dsn_uid,dsn_pwd,dsn_security,)
try:
    conn = ibm_db.connect(dsn, "", "")
    print ("Connected to database: ", dsn_database, "as user: ", dsn_uid, "on host: ",
dsn_hostname)
    @app.route("/",methods=['GET','POST'])
    @app.route("/register",methods=['GET','POST'])
    def register():
        msg = None
        if request.method == 'POST':
            name=request.form['name']
            rollno=request.form['rollno']

```

```

email=request.form['email']
password=request.form['password']

select_sql = "Select * from User where email=?"
stmt = ibm_db.prepare(conn,select_sql)
ibm_db.bind_param(stmt,1,email)
ibm_db.execute(stmt)
user = ibm_db.fetch_assoc(stmt)
if user:
    msg = "User with this email already exists!"
    return render_template("register.html",msg = msg)
else:
    insert_sql = "Insert into User Values (?, ?, ?, ?)"
    stmt = ibm_db.prepare(conn,insert_sql)
    ibm_db.bind_param(stmt,1,name)
    ibm_db.bind_param(stmt,2,rollno)
    ibm_db.bind_param(stmt,3,email)
    ibm_db.bind_param(stmt,4,password)
    ibm_db.execute(stmt)
    return redirect("/login")
return render_template("register.html")

```

```

@app.route("/login",methods=['GET','POST'])
def login():
    msg = None
    if request.method == 'POST':
        email=request.form['email']
        password=request.form['password']

        select_sql = "Select * from User where email=?"
        stmt = ibm_db.prepare(conn,select_sql)
        ibm_db.bind_param(stmt,1,email)
        ibm_db.execute(stmt)
        user = ibm_db.fetch_assoc(stmt)
        if not user:
            msg = "User with this email does not exist!"
            return render_template("login.html",msg = msg)
        elif user['PASSWORD'] != password:
            msg = "Incorrect Passowrd!"
            return render_template("login.html",msg = msg)
        else:
            session['loggedin'] = True
            session['id'] = user['EMAIL']
            session['username'] = user['NAME']
            return redirect('/welcome')
    return render_template("login.html")

```

```

@app.route("/logout")

```

```

def logout():
    session['loggedin'] = False
    session['id'] = ""
    session['username'] = ""
    return render_template("login.html",msg="Logged out successfully!")

@app.route("/welcome")
def welcome():
    return render_template("welcome.html",)

except:
    print ("Unable to connect: ", ibm_db.conn_errormsg() )

if __name__ == '__main__':
    app.run(debug=True)

```

### Login.html

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="{{url_for('static', filename='css/style.css')}}">
    <title>Login Form</title>
</head>
<body>
    <div class="center">
        <h1>Login Form</h1>
        {% if msg %}
            <span>{{msg}}</span>
        {% endif %}

        <form action="/login" method="POST">
            <div class="inputbox">
                <label for="email">Email</label>
                <input type="text" name="email" required>
            </div>
            <div class="inputbox">
                <label for="password">Password:</label>
                <input type="password" name="password" required>
            </div>
            <button type="submit">Submit</button>
            <button type="reset">Cancel</button>
        </form>
        <span>Don't have an account? <a href="/register">Sign Up</a></span>
    </div>
</body>
</html>

```

## Register.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="{{url_for('static', filename='css/style.css')}}">
  <title>Registration Form</title>
</head>
<body>
  <div class="center">
    <h1>SignIn Form</h1>
    {% if msg %}
      <span>{{msg}}</span>
    {% endif %}
    <form action="/register" method="POST">
      <div class="inputbox">
        <label for="email">Name</label>
        <input type="text" name="name" required>
      </div>
      <div class="inputbox">
        <label for="rollno">Roll No</label>
        <input type="text" name="rollno" required>
      </div>
      <div class="inputbox">
        <label for="email">Email</label>
        <input type="email" name="email" required>
      </div>
      <div class="inputbox">
        <label for="password">Password</label>
        <input type="password" name="password" required>
      </div>
      <button type="submit">Submit</button>
      <button type="reset">Cancel</button>
    </form>
    <span>Already have an account? <a href="/login">Login</a></span>
  </div>
</body>
</html>
```

## Welcome.html

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Welcome Page</title>
</head>
```

```

<body>
  <h1>Welcome {{session['username']}}!</h1>
  <br>
  <a href="/logout">Logout</a>
</body>
</html>

```

## REGISTRATION FORM

### Sign In Form





Already have an account? [Login](#)

### Login Form



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

## DB

GTM67329.REGISTER				Back
				Export to CSV
NAME	ROLLNO	EMAIL	PASSWORD	
Ananda	1901011	ananda.1901011@gmail.com	ananda@0011	
Karthik	1901062	karthik.1901062@gmail.com	Passwerd	
Dilip	1901035	dilip.1901035@gmail.com	dilip35	
Kupanesan	1901068	kupa.1901038@gmail.com	kupa.088	

## LOGIN

<div><h3>Login Form</h3><div>Email</div><div>Password:</div><div><div>Submit</div><div>Cancel</div></div><div>Don't have an account? <a href="#">Sign Up</a></div></div>	<div><h3>Login Form</h3><div>dilipthambhuraj007@gmail.com</div><div>*****</div><div><div>Submit</div><div>Cancel</div></div><div>Don't have an account? <a href="#">Sign Up</a></div></div>
--	---

## VALID USER

WELCOME DILIP!

[Logout](#)

## INVALID USER DETAILS

<div><h3>Login Form</h3><div>User with this email does not exist!</div><div>Email</div><div>Password:</div><div><div>Submit</div><div>Cancel</div></div><div>Don't have an account? <a href="#">Sign Up</a></div></div>	<div><h3>Login Form</h3><div>Incorrect Passowrd!</div><div>Email</div><div>Password:</div><div><div>Submit</div><div>Cancel</div></div><div>Don't have an account? <a href="#">Sign Up</a></div></div>
---	--