

## Assignment-2

AssignmentDate	19September2022
StudentName	Lokesh V
StudentRollNumber	737819ECR094
MaximumMarks	2Marks

### Question-1:

**Create Usertable with user with email, username, roll number, password.**

```
CREATETABLEUSER(emailvarchar(255),us  
    urname  
    varchar(255),roll_numvarchar(  
    255),passwordvarchar(255)  
    );
```

### Question-2:

**2. Perform UPDATE, DELETE Queries with usertable**

```
UPDATEuser  
SETusername='Logesh'  
WHERE  
email='logu12@gmail.com';DELETEfromuserwhereuse  
rname='Logesh';
```

### Question-3:

**3. Connect python code to db2.**

```
db_connection=ibm_db.connect("DATABASE=bludb;QUERYTIMEOUT=1;CONNECTTIMEOUT=10;HOST  
NAME=764264db-9824-4b7c-82df-  
40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;PORT=32536;SECURITY=SSL;SSLSe  
rverCertificate=./DigiCertGlobalRootCA.crt;PROTOCOL=TCPIP;UID  
=zmn60939;PWD=YjtYgCpA1y4T7AZQ","","")
```

#### Question-4:

**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 to authenticate user username and password. If the user is valid show the welcome page**

#### app.py

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

app = Flask(__name__)
app.config['DEBUG'] = True

db_connection = ibm_db.connect(
    "DATABASE=bludb;QUERYTIMEOUT=1;CONNECTTIMEOUT=10;HOSTNAME=0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lbg.databases.appdomain.cloud;PORT=31198;SECURITY=SSL;SSLServerCertificate=../DigiCertGlobalRootCA.crt;PROTOCOL=TCPIP;UID=gpf02299;PWD=yLEwXmSNoNC AhWGp", "", "")

@app.route("/", methods=["GET", "POST"])
def register():
    if request.method == 'POST':
        data = request.form.to_dict()
        print(data)
        sql_query = f"INSERT INTO User (username, email, roll_no, password) VALUES ('{data['username']}', '{data['email']}', '{data['roll_no']}', '{data['password']}')"
        ibm_db.exec_immediate(db_connection, sql_query)
        return redirect(url_for("login"))
    if request.method == 'GET':
        return render_template("registration.html")
```

```

@app.route("/login", methods=["GET",
"POST"])deflogin():
    ifrequest.method=='POST':
        data=request.form.to_dict()
        sql_query=f"SELECTpasswordfromUserWHEREusername='{data['username']}'"result=ib
        m_db.exec_immediate(db_connection,sql_query)
        value =
        ibm_db.fetch_tuple(result)ifvalue[
        0]==data["password"]:
            return
            redirect(url_for("welcome"))else:
                return "<p style=color:red;>Invalid
Credentials</p>"ifrequest.method=="GET":
                returnrender_template("login.html")

```

```

@app.route("/welcome",
methods=["GET"])defwelcome():
    return"<h1style=color:green;>Welcome!!!!</h1>"

```

```

ifname
_____=='main':app.r
un()

```

## templates/registration.html:

```
<!DOCTYPEhtml>

<htmllang="en">

<head>

    <metacharset="UTF-8">

    <metaname="viewport"content="width=device-width,initial-scale=1.0">

    <metahttp-equiv="X-UA-Compatible"content="ie=edge">

    <title>RegistrationForm</title>

    <linkrel="stylesheet"href="{{url_for('static',filename='registration.css')}}">

</head>

<body>

    <formmethod="POST"action="{{url_for('register')}}">

        <divclass="container">

            <br/>

            <inputname="username"placeholder="Username"type="text"/>

            <br/>

            <inputname="email"placeholder="contact@student1.com"type="email"/>

            <br/>

            <inputname="roll_no"placeholder="Rollnumber"type="text"/>

            <br/>

            <inputname="password"placeholder="Password"type="password"/>

            <br/>

            <buttontype="submit">Register</button>

        </div>

    </form>

</body>

</html>
```

## login.html

```
<!DOCTYPEhtml>

<htmllang="en">

<head>

  <metacharset="UTF-8">

  <metaname="viewport"content="width=device-width,initial-scale=1.0">

  <metahttp-equiv="X-UA-Compatible"content="ie=edge">

  <title>RegistrationForm</title>

  <linkrel="stylesheet"href="{{url_for('static',filename='registration.css')}}">

</head>

<body>

  <formmethod="POST"action="{{url_for('login')}}">

    <divclass="container">

      <inputname="username"placeholder="Username"type="text"/>

      <br/>

      <inputname="password"placeholder="Password"type="password"/>

      <br/>

      <buttontype="submit">Login</button>

    </div>

  </form>

</body>

</html>
```

## Registration.css

```
@import url('https://fonts.googleapis.com/css?family=Montserrat:400,500&display=swap');
```

```
body{  
    font-family: Montserrat,Arial, Helvetica, sans-  
    serif;background-color:#f7f7f7;  
}  
*{box-sizing:border-box}
```

```
.container  
{padding:  
    20px;width:50  
    0px;  
    position:  
    absolute;left:50%;  
    top:50%;  
    transform: translate(-50%, -  
    50%);border:1pxsolid#ccc;  
    border-  
    radius:10px;background:white;  
-webkit-box-shadow:2px1px21px-9pxrgba(0,0,0,0.38);  
-moz-box-shadow:2px1px21px-  
9pxrgba(0,0,0,0.38);box-shadow:2px1px21px-  
9pxrgba(0,0,0,0.38);  
}
```

```
input[type=text], input[type=email], input[type=password]  
{width:100%;  
padding:  
    15px;margin:5px022  
    px0;display: inline-  
    block;border:  
    none;background:#f  
    7f7f7;
```

```
        font-family:Montserrat,Arial,Helvetica,sans-serif;
    }
```

```
input[type=text]:focus, input[type=password]:focus, input[type=phone]:focus, select:focus
{background-color:#ddd;
outline:none;
}
```

```
button{
    background-color:
    #0eb7f4;color:white;
    padding:14px20px;
    margin: 8px
    0;border:
    none;cursor:
    pointer;width:100%
    ;
    opacity:0.9;
        font-size:16px;
        font-family: Montserrat,Arial, Helvetica, sans-
        serif;border-radius:10px;
    }
```

```
button:hover{
    opacity:1;
}
```

```
@mediascreenand(max-width:300px){
    .signupbtn{widt
        h:100%}}
```

