

## Assignment-3

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

```
CREATE TABLE Users (  
    username varchar(255),  
    email varchar(255),  
    roll_number int,  
    pwd varchar(255)
```

```
);
```

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

```
UPDATE Users SET roll_number=10 , pwd="open123" WHERE username="Mala";
```

```
DELETE FROM Users WHERE username="Mala";
```

### 3. Connect python code to db2.

```
conn=ibm_db.connect("DATABASE=bludb;
```

```
HOSTNAME=b0aebb68-94fa-46ec-a1fc-
```

```
1c999edb6187.c3n41cmd0nqnkrk39u98g.databases.appdomain.cloud;
```

```
PORT=31249;
```

```
SECURITY=SSL;
```

```
ServerCertificate=DigiCertGlobalRootCA.crt;
```

```
UID=tbd72327;
```

```
PWD=ftnrC2umf9VxgMW1",",")
```

### 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.

#### Welcome.html

```
<html>
```

```
    <head>
```

```
        <title> Index </title>
```

```
    </head>
```

```
<body></br></br></br></br></br>

    <div align="center">

        <div align="center" class="border">

            <div class="header">

                <h1 class="word">Index</h1>

            </div></br></br></br>

            <h1 class="bottom">

                </br></br> Welcome to the index page...

            </h1></br></br></br>

        </div>

    </div>

</body>

</html>
```

## Login.html

```
<html>

    <head>

        <meta charset="UTF-8">

        <title> Login </title>

    </head>

    <body></br></br></br></br></br>

        <div align="center">

            <div align="center" >

                <div >

                    <h1>Login</h1>

                </div></br></br></br>

                <h2 class="word">
```

```
<form action="/login" method="post">

    <input id="username" name="username" type="text"
placeholder="Enter Your Username"/></br></br>

    <input id="password" name="password" type="password"
placeholder="Enter Your Password" /></br></br></br>

    <input type="submit" value="Sign In"/></br></br>

</form>

</h2>

<p>Don't have an account? <a href="register.html"> Sign Up here</a></p>

</div>

</div>

</body>

</html>
```

## register.html

```
<html>

    <head>

        <meta charset="UTF-8">

        <title> Register </title>

    </head>

    <body></br></br></br></br></br>

        <div align="center">

            <div align="center" class="border">

                <div class="header">

                    <h1 class="word">Register</h1>

                </div></br></br></br>

                <h2 class="word">

                    <form action="/register" method="post">
```

```

<input id="username" name="username" type="text"
placeholder="Enter Your Username"/></br></br>

<input id="email" name="email" type="text"
placeholder="Enter Your Email ID"/></br></br>

<input id="password" name="password" type="password"
placeholder="Enter Your Password" /></br></br>

<input type="submit" class="btn" value="Sign Up"></br>

</form>

</h2>

<p>Already have an account? <a href="login.html"> Sign In here</a></p>

</div>

</div>

</body>

</html>

```

## 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=b0aebb68-94fa-46ec-a1fc-
1c999edb6187.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;PORT=31249;SECURITY=SSL;
ServerCertificate=DigiCertGlobalRootCA.crt;UID=tbd72327;PWD=ftnrC2umf9VxgMW1",",")

@app.route('/')

def home():

    return render_template('home.html')

@app.route('/login',method=['GET','POST'])

def login():

```

```
global userid
```

```
msg = "
```

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

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

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

```
    sql = "SELECT * FROM Users 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 render_template('dashboard.html',msg=msg)
```

```
    else:
```

```
        msg ='incorrect username and password'
```

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

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

```
def register():
```

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

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

```

email= request.form['email']

password= request.form['password']

sql = "SELECT * FROM Users WHERE username =? "

stmt=ibm_db.prepare(conn,sql)

ibm_db.bind_param(stmt,1,username)

ibm_db.bind_param(stmt,2,password)

ibmm_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 characters and numbers"

else:

    insert_sql="INSERT INTO users 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'

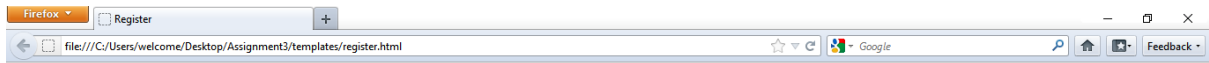
    return render_template('register.html',msg=msg)

if __name__=='__main__':

```

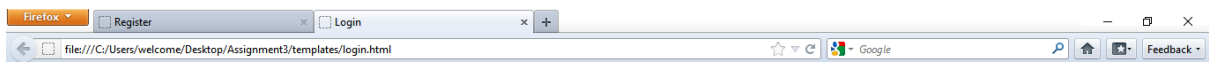
```
app.run(host='0.0.0.0')
```

## OUTPUT:



### Register

Already have an account? [Sign In here](#)



### Login

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

