

## ASSIGNMENT-3

Date	11 October 2022
Team ID	PNT2022TMID49558
Project Name	Project-plasma donor Application
Team Member Name	ANU K

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

```
Run SQL Command Line
Connected.
SQL> create table anu(email varchar(30),username varchar(15), rollnumber int, password varchar(10));
Table created.
SQL> desc anu;
+-----+-----+-----+
Name                               Null?   Type
+-----+-----+-----+
EMAIL                              VARCHAR2(30)
USERNAME                           VARCHAR2(15)
ROLLNUMBER                         NUMBER(38)
PASSWORD                           VARCHAR2(10)
+-----+-----+-----+
SQL> insert into anu values('anu43@gmail.com','anu','03','h53');
1 row created.
SQL> insert into anu values('evanji113@gmail.com','evanji','11','f43');
1 row created.
SQL> insert into anu values('sher37@gmail.com','shermina','18','y67');
1 row created.
SQL> select* from anu;
+-----+-----+-----+-----+
EMAIL                               USERNAME   ROLLNUMBER PASSWORD
+-----+-----+-----+-----+
anu43@gmail.com                     anu        3 h53
evanji113@gmail.com                 evanji     11 f43
sher37@gmail.com                     shermina   18 y67
+-----+-----+-----+-----+
SQL> update anu set rollnumber = '11' where username='evanji';
1 row updated.
SQL> select* from anu;
+-----+-----+-----+-----+
EMAIL                               USERNAME   ROLLNUMBER PASSWORD
+-----+-----+-----+-----+
anu43@gmail.com                     anu        3 h53
evanji113@gmail.com                 evanji     11 f43
+-----+-----+-----+-----+
```

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

```
Run SQL Command Line
1 row updated.
SQL> select* from anu;
EMAIL                USERNAME      ROLLNUMBER  PASSWORD
-----
anu43@gmail.com      anu          3           h53
evanji13@gmail.com   evanji       11          f43
sher37@gmail.com     shermina     18          y67

SQL> update anu set rollnumber='43' where username='evanji';
1 row updated.
SQL> select* from anu;
EMAIL                USERNAME      ROLLNUMBER  PASSWORD
-----
anu43@gmail.com      anu          3           h53
evanji13@gmail.com   evanji       43          f43
sher37@gmail.com     shermina     18          y67

SQL> delete from anu where rollnumber='18';
1 row deleted.
SQL> select* from anu;
EMAIL                USERNAME      ROLLNUMBER  PASSWORD
-----
anu43@gmail.com      anu          3           h53
evanji13@gmail.com   evanji       43          f43

SQL> update anu set username='varsha' where rollnumber='43';
1 row updated.
SQL> select* from anu;
EMAIL                USERNAME      ROLLNUMBER  PASSWORD
-----
anu43@gmail.com      anu          3           h53
evanji13@gmail.com   varsha       43          f43
```

## 3. Connect python code to db2

```
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=125f9f61-
9715-46f9-9399-
c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30
426;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UI
D=thy82883;PWD=DtTOZYv3UCHUfP15",",")
```

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 the database and navigate to the login page to authenticate user username and password. If the user is valid show the welcome page

**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=125f9f61-9715-46f9-9399-
c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30426;SECU
RITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=thy82883;PWD=DtTOZ
Yv3UCHUfP15","")
@app.route('/')
def home():
    return render_template('register.html')
@app.route('/login', methods=['GET', 'POST'])
def login():
    global userid
    msg = ""
    if request.method == 'POST' and 'username' in request.form and 'password' in
request.form:
        username = request.form['username']
        password = request.form['password']
        stmt = ibm_db.prepare(conn,'SELECT * FROM accounts WHERE username = ?
AND password = ?')
        ibm_db.bind_param(stmt,1,username)
        ibm_db.bind_param(stmt,2,password)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        if account:
            session['loggedin'] = True
            session['username'] = account['USERNAME']
            msg = 'Logged in successfully !'
            return render_template('index.html', msg = msg)
        else:
            msg = 'Incorrect username / password !'
            return render_template('login.html', msg = msg)

@app.route('/logout')
def logout():
    session.pop('loggedin', None)
    session.pop('id', None)
    session.pop('username', None)
    return redirect(url_for('login'))

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

```

msg = "
if request.method == 'POST':
    username = request.form['username']
    password = request.form['password']
    email = request.form['email']
    sql = "SELECT * FROM accounts 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 characters and numbers !'
    elif not username or not password or not email:
        msg = 'Please fill out the form !'
    else:
        insert_sql = "INSERT INTO accounts VALUES (?, ?, ?)"
        stmt = ibm_db.prepare(conn,insert_sql)
        ibm_db.bind_param(stmt, 1, username)
        ibm_db.bind_param(stmt, 2, password)
        ibm_db.bind_param(stmt, 3, email)
        ibm_db.execute(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(debug = True)

```

## index.html

```

<html>
<head>
    <meta charset="UTF-8">
    <title> Index </title>
    <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
</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">
    Hi {{ session.username }}!!<br></br> Welcome to the index page...
  </h1><br></br></br>
  <a href="{{ url_for('logout') }}" class="btn">Logout</a>
</div>
</div>
</body>
</html>

```

### register.html

```

<html>
<head>
  <meta charset="UTF-8">
  <title> Register </title>
  <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
</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="{{ url_for('register') }}" method="post">
          <div class="msg">{{ msg }}</div>
          <input id="username" name="username" type="text" placeholder="Enter Your
Username" class="textbox"/></br></br>
          <input id="password" name="password" type="password" placeholder="Enter Your
Password" class="textbox"/></br></br>
          <input id="email" name="email" type="text" placeholder="Enter Your Email ID"
class="textbox"/></br></br>
          <input type="submit" class="btn" value="Sign Up"></br>
        </form>
      </h2>
      <p class="bottom">Already have an account? <a class="bottom"
href="{{ url_for('login') }}"> Sign In here</a></p>

```

```
        </div>
    </div>
</body>
</html>
```

## login.html

```
<html>
<head>
    <meta charset="UTF-8">
    <title> Login </title>
    <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
</head>
<body></br></br></br></br>
    <div align="center">
        <div align="center" class="border">
            <div class="header">
                <h1 class="word">Login</h1>
            </div></br></br></br>
            <h2 class="word">
                <form action="{{ url_for('login') }}" method="post">
                    <div class="msg">{{ msg }}</div>
                    <input id="username" name="username" type="text" placeholder="Enter Your
Username" class="textbox"/></br></br>
                    <input id="password" name="password" type="password" placeholder="Enter Your
Password" class="textbox"/></br></br></br>
                    <input type="submit" class="btn" value="Sign In"></br></br>
                </form>
            </h2>
            <p class="bottom">Don't have an account? <a class="bottom"
href="{{ url_for('register') }}"> Sign Up here</a></p>
        </div>
    </div>
</body>
</html>
```

## style.css

```
.header{
    padding: 5px 120px;
    width: 150px;
```

```
height: 70px;  
background-color: #236B8E;  
}
```

```
.border{  
padding: 80px 50px;  
width: 400px;  
height: 450px;  
border: 1px solid #236B8E;  
border-radius: 0px;  
background-color: #9AC0CD;  
}
```

```
.btn {  
padding: 10px 40px;  
background-color: #236B8E;  
color: #FFFFFF;  
font-style: oblique;  
font-weight: bold;  
border-radius: 10px;  
}
```

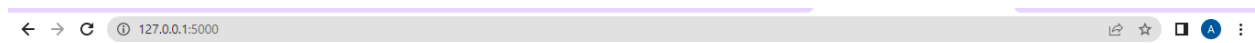
```
.textbox{  
padding: 10px 40px;  
background-color: #236B8E;  
text-color: #FFFFFF;  
border-radius: 10px;  
}
```

```
::placeholder {  
color: #FFFFFF;  
opacity: 1;  
font-style: oblique;  
font-weight: bold;  
}
```

```
.word{  
color: #FFFFFF;  
font-style: oblique;  
font-weight: bold;
```

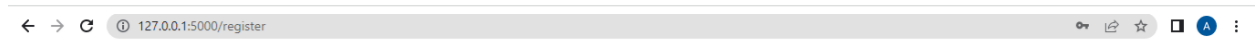
```
}  
  
.bottom{  
  color: #236B8E;  
  font-style: oblique;  
  font-weight: bold;  
}
```

## OUTPUT



### Register

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



### Register

**You have successfully registered !**

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



## Login

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

## Index

Hi Anu !!

Welcome to the index page...

[Logout](#)