

### ASSIGNMENT-3

DATE	01 November 2022
Team ID	PNT2022TMID49597
Project Name	Plasma donar application
Team Member Name	Raja chera kesaree ve.su

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

```
MySQL 5.5 Command Line Cli
mysql> use assign
Database changed
mysql> create table yogan(email varchar(30),username varchar(30), rollnumber int(20),password varchar(10));
Query OK, 0 rows affected (0.02 sec)

mysql> desc yogan;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| email | varchar(30) | YES | | NULL | |
| username | varchar(30) | YES | | NULL | |
| rollnumber | int(20) | YES | | NULL | |
| password | varchar(10) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.01 sec)

mysql> insert into yogan values(yog@gmail.com,yogan,'703',yogan);
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '@gmail.com,yogan,'703',yogan)' at line 1
mysql> insert into yogan values('yog@gmail.com','yogan','703','yogan');
Query OK, 1 row affected (0.00 sec)

mysql> insert into yogan values('remy@gmail.com','remy','73','remy');
Query OK, 1 row affected (0.00 sec)

mysql> insert into yogan values('abc@gmail.com','rabc','73','abc');
Query OK, 1 row affected (0.00 sec)

mysql> select* from yogan;
+-----+-----+-----+-----+
| email | username | rollnumber | password |
+-----+-----+-----+-----+
| yog@gmail.com | yogan | 703 | yogan |
| remy@gmail.com | remy | 73 | remy |
| abc@gmail.com | rabc | 73 | abc |
+-----+-----+-----+-----+
```

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

```
MySQL 5.5 Command Line Cli X + v
+-----+-----+-----+-----+
| email      | username | rollnumber | password |
+-----+-----+-----+-----+
| yog@gmail.com | yogan   | 783       | yogan   |
| remy@gmail.com | remy    | 73        | remy    |
| abc@gmail.com | rabc    | 73        | abc     |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> update yogan set rollnumber='67' where username='yogan';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> select* from yogan;
+-----+-----+-----+-----+
| email      | username | rollnumber | password |
+-----+-----+-----+-----+
| yog@gmail.com | yogan   | 67        | yogan   |
| remy@gmail.com | remy    | 73        | remy    |
| abc@gmail.com | rabc    | 73        | abc     |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> delete from yogan where username='remy';
Query OK, 1 row affected (0.00 sec)

mysql> select* from yogan;
+-----+-----+-----+-----+
| email      | username | rollnumber | password |
+-----+-----+-----+-----+
| yog@gmail.com | yogan   | 67        | yogan   |
| abc@gmail.com | rabc    | 73        | abc     |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

```
MySQL 5.5 Command Line Cli X + v
+-----+-----+-----+-----+
| remy@gmail.com | remy    | 73        | remy    |
| abc@gmail.com | rabc    | 73        | abc     |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> delete from yogan where username='remy';
Query OK, 1 row affected (0.00 sec)

mysql> select* from yogan;
+-----+-----+-----+-----+
| email      | username | rollnumber | password |
+-----+-----+-----+-----+
| yog@gmail.com | yogan   | 67        | yogan   |
| abc@gmail.com | rabc    | 73        | abc     |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> delete from yogan where username='abc';
Query OK, 0 rows affected (0.00 sec)

mysql> select* from yogan;
+-----+-----+-----+-----+
| email      | username | rollnumber | password |
+-----+-----+-----+-----+
| yog@gmail.com | yogan   | 67        | yogan   |
| abc@gmail.com | rabc    | 73        | abc     |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> |
```

3. Connect python code to db2

```
conn =  
ibm_db.connect("DATABASE=bludb;HOSTNAME=6667d8e9-9d4d-4ccb-ba32-21da3bb  
5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30376;SECURITY=SS  
L;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=lxk19443;PWD=Qv2NUIITS3C4  
Ur7iB", "", "")
```

**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=6667d8e9-9d4d-4ccb-ba32-21da3bb  
5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30376;SECURITY=SS  
L;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=lxk19443;PWD=Qv2NUIITS3C4  
Ur7iB", "", "")  
@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>
    <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>
```

## 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></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>

```

## 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>

```

**OUTPUT:**



## Register

Sign Up

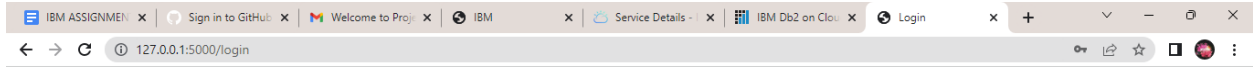
Already have an account? [Sign In here](#)

## Register

**You have successfully registered !**

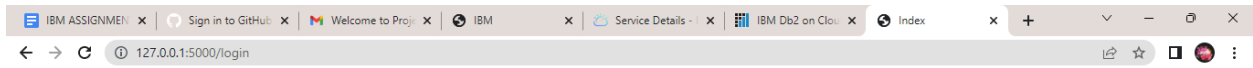
Sign Up

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



## Login

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



## Index

**Hi yogan !!**

**Welcome to the index page...**

[Logout](#)

