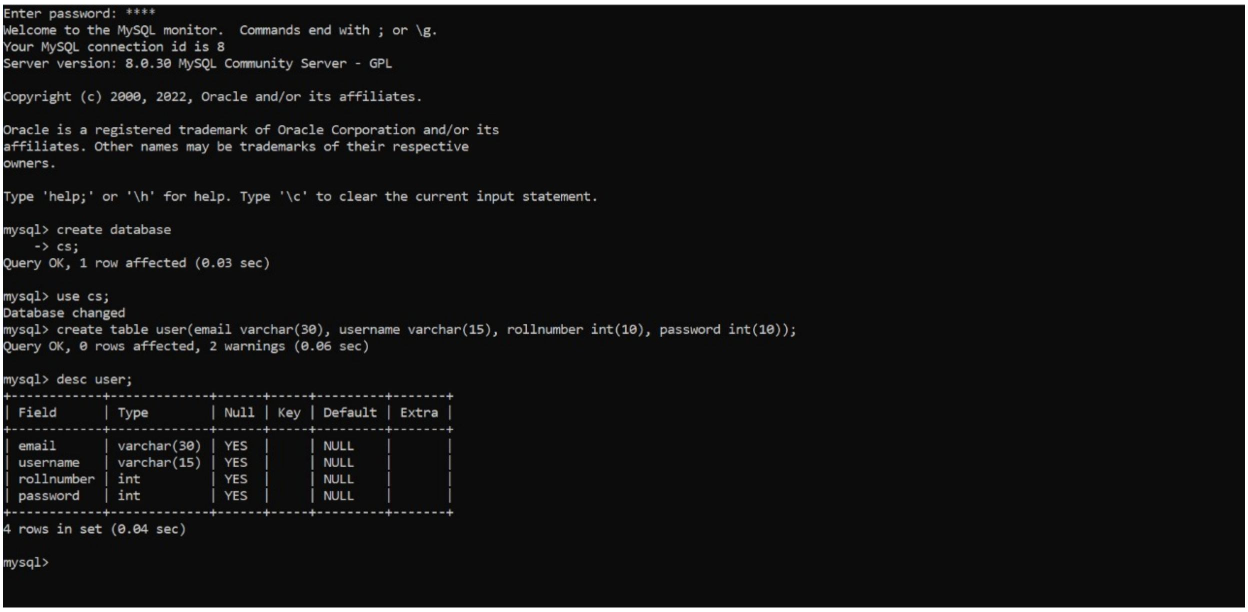
Team ID	PNT2022PMID49597
Project Name	Project – PLASMA DONOR APPLICATION
Name	SARAVANAN S
Roll Number	950019104041

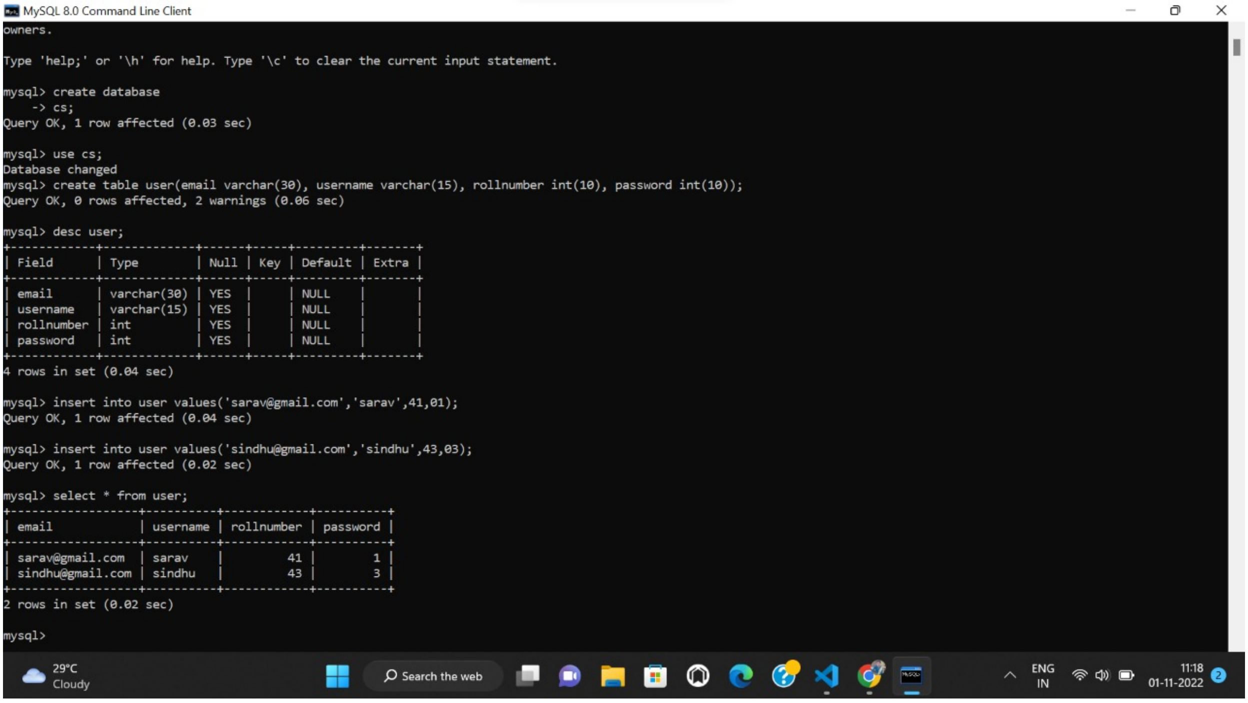
Assignment-3

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



```
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.30 MySQL Community Server - GPL

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database
    -> cs;
Query OK, 1 row affected (0.03 sec)

mysql> use cs;
Database changed
mysql> create table user(email varchar(30), username varchar(15), rollnumber int(10), password int(10));
Query OK, 0 rows affected, 2 warnings (0.06 sec)

mysql> desc user;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| email      | varchar(30) | YES  |     | NULL    |       |
| username   | varchar(15) | YES  |     | NULL    |       |
| rollnumber | int         | YES  |     | NULL    |       |
| password   | int         | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
4 rows in set (0.04 sec)

mysql>
```

2.Perform UPDATE, DELETE Queries with user table.



```
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database
    -> cs;
Query OK, 1 row affected (0.03 sec)

mysql> use cs;
Database changed
mysql> create table user(email varchar(30), username varchar(15), rollnumber int(10), password int(10));
Query OK, 0 rows affected, 2 warnings (0.06 sec)

mysql> desc user;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| email      | varchar(30) | YES  |     | NULL    |       |
| username   | varchar(15) | YES  |     | NULL    |       |
| rollnumber | int         | YES  |     | NULL    |       |
| password   | int         | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
4 rows in set (0.04 sec)

mysql> insert into user values('sarav@gmail.com','sarav',41,01);
Query OK, 1 row affected (0.04 sec)

mysql> insert into user values('sindhu@gmail.com','sindhu',43,03);
Query OK, 1 row affected (0.02 sec)

mysql> select * from user;
+------------------+----------+------------+----------+
| email            | username | rollnumber | password |
+------------------+----------+------------+----------+
| sarav@gmail.com  | sarav    |         41 |        1 |
| sindhu@gmail.com | sindhu   |         43 |        3 |
+------------------+----------+------------+----------+
2 rows in set (0.02 sec)

mysql>
```

3.Connect python code to db2.

App.py

```
conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=31321;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=dxg36717;PWD=miSxjpV7Ybwrrlyf",'','')
```

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.

app.py

```
from flask import Flask, render_template ,request,redirect,session,url_for
import ibm_db
import re
app = Flask(__name__)
app.secret_key='a'
conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=31321;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=dxg36717;PWD=miSxjpV7Ybwrrlyf",'','')
@app.route('/')
@app.route('/login', methods =['GET', 'POST'])
def login():
    global userid
    msg = ''
    if request.method == 'POST' and 'name' in request.form and 'password' in request.form:
        name = request.form['name']
        password = request.form['password']
        stmt = ibm_db.prepare(conn,'SELECT * FROM users WHERE name = ? AND password = ?')
        ibm_db.bind_param(stmt,1,name)
        ibm_db.bind_param(stmt,2,password)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        if account:
            session['loggedin'] = True
            session['name'] = account['NAME']
            msg = 'Logged in successfully !'
            return render_template('home.html', msg = msg)
        else:
            msg = 'Incorrect name / password !'
    return render_template('login.html', msg = msg)
@app.route('/logout')
def logout():
        session.pop('loggedin', None)
        session.pop('id', None)
        session.pop('name', None)
        return redirect(url_for('login'))
@app.route('/register', methods =['GET', 'POST'])
def register():
        msg = ''
        if request.method == 'POST' and 'name' in request.form and 'password' in request.form and 'email' in request.form :
                name = request.form['name']
                password = request.form['password']
                email = request.form['email']
                stmt = ibm_db.prepare(conn,'SELECT * FROM users WHERE name
= ?')
                ibm_db.bind_param(stmt,1,name)
                ibm_db.execute(stmt)
                account = ibm_db.fetch_assoc(stmt)
                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]+', name):
                    msg = 'name must contain only characters and numbers !'
                elif not name or not password or not email:
                    msg = 'Please fill out the form !'
                else:
                    prep_stmt = ibm_db.prepare(conn,"INSERT INTO users
VALUES(?, ?, ?)")
                    ibm_db.bind_param(prep_stmt, 1, name)
                    ibm_db.bind_param(prep_stmt, 2, password)
                    ibm_db.bind_param(prep_stmt, 3, email)
                    ibm_db.execute(prep_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)
```

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="name" name="name" 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="SignIn"></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

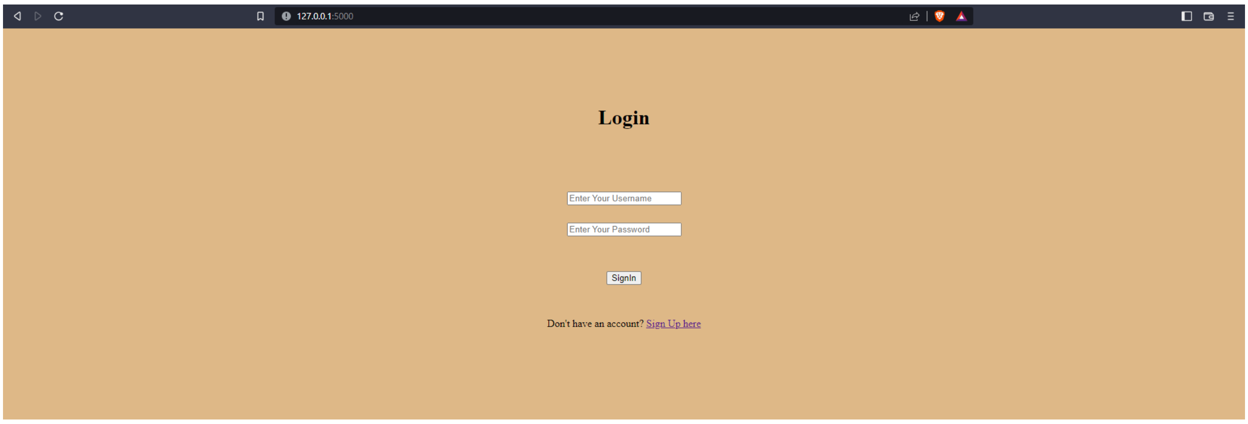
```
<html>

<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="name" name="name" 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="SignUp"></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>
```

Home.html

```html
<html>

<head>
    <meta charset="UTF-8">
    <title> WELCOME PAGE </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">WELCOME PAGE </h1>
            </div></br></br></br>
            <h1 class="bottom">
                Hi {{session.name}}!!</br></br> Welcome to Ourpage...
            </h1></br></br></br>
            <a href="{{ url_for('logout') }}" class="btn">Logout</a>
        </div>
    </div>
</body>

</html>
```

Output:

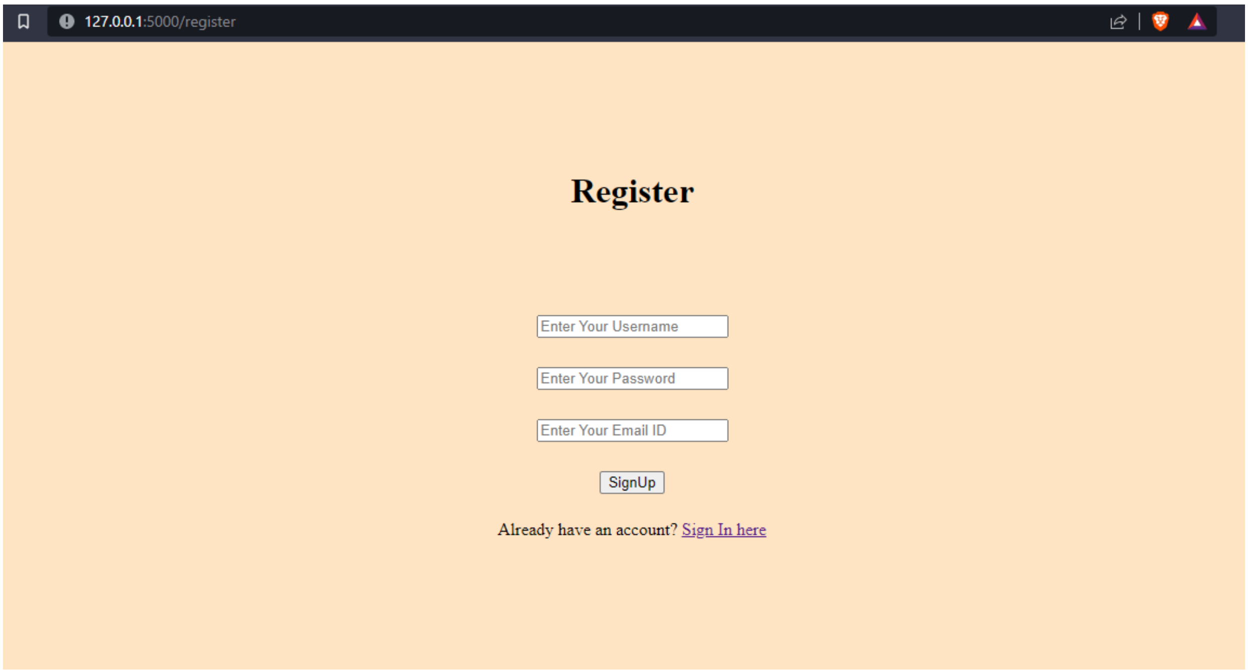
Login.html



Login

Don't have an account? Sign Up here

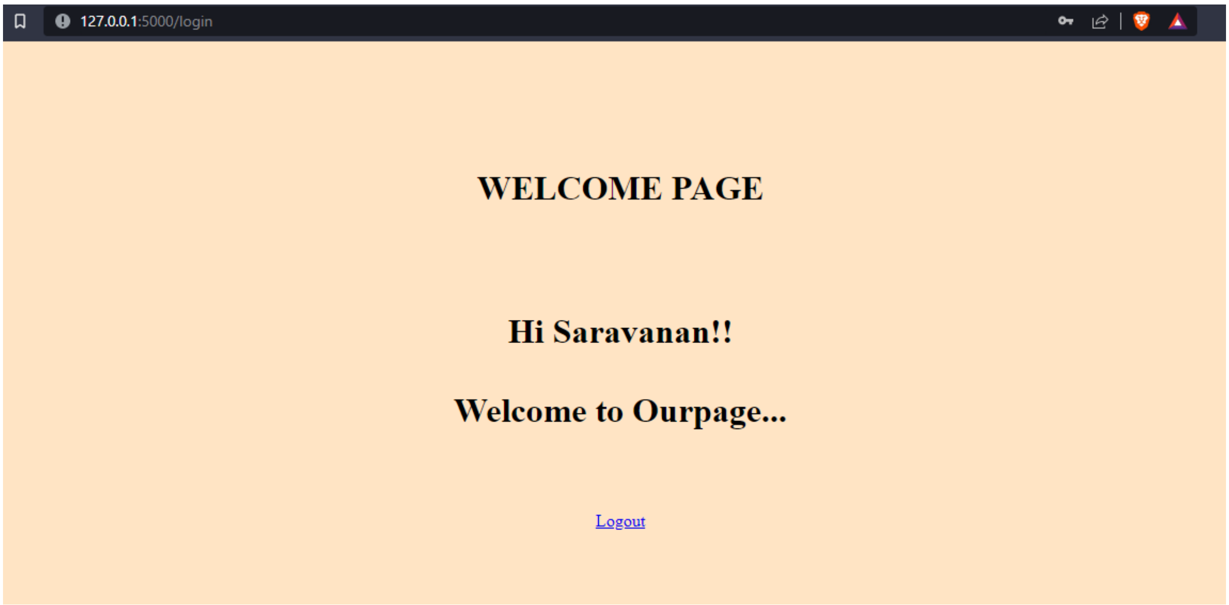
Register.html



Register

Already have an account? Sign In here

Home.html



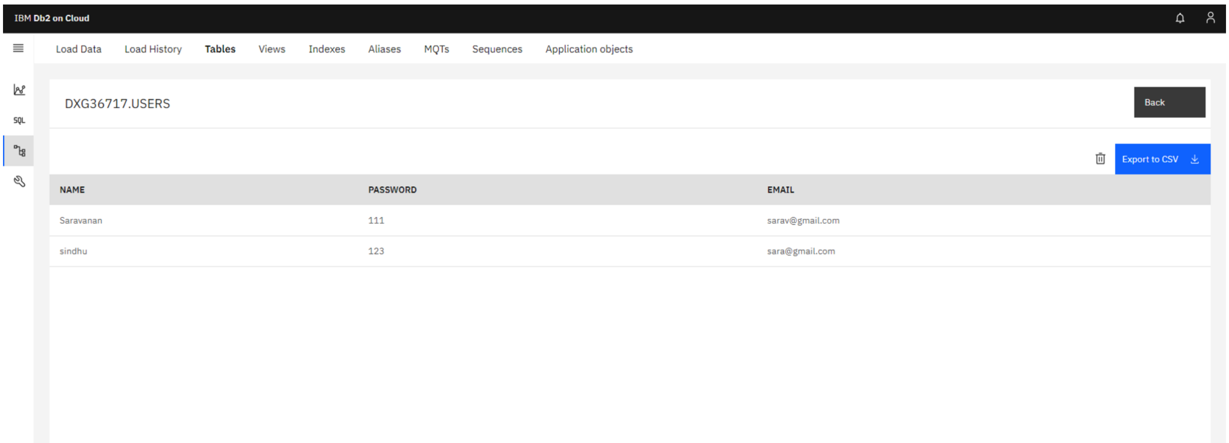
WELCOME PAGE

Hi Saravanan!!

Welcome to Ourpage...

Logout

Ibm Database view



DXG36717.USERS

NAME	PASSWORD	EMAIL
Saravanan	111	sarav@gmail.com
sindhu	123	sara@gmail.com