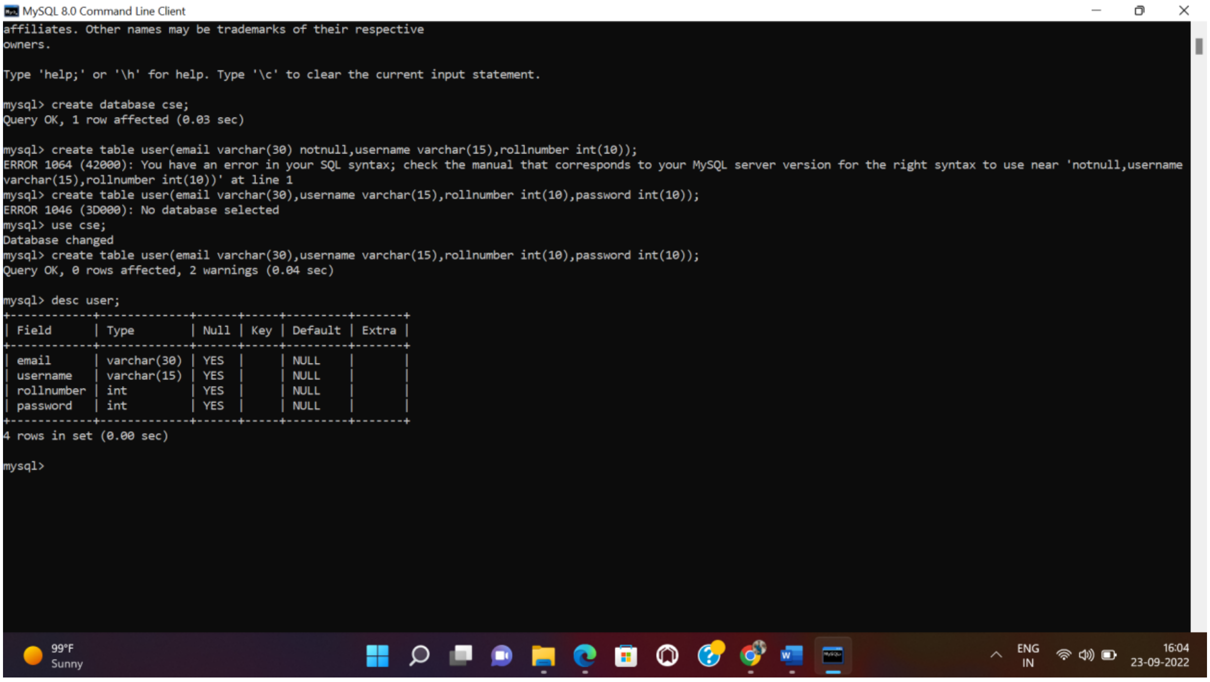
Team ID	PNT2022PMID49593
Project Name	Project – PLASMA DONOR APPLICATION
Name	SINDHU R
Roll Number	950019104043

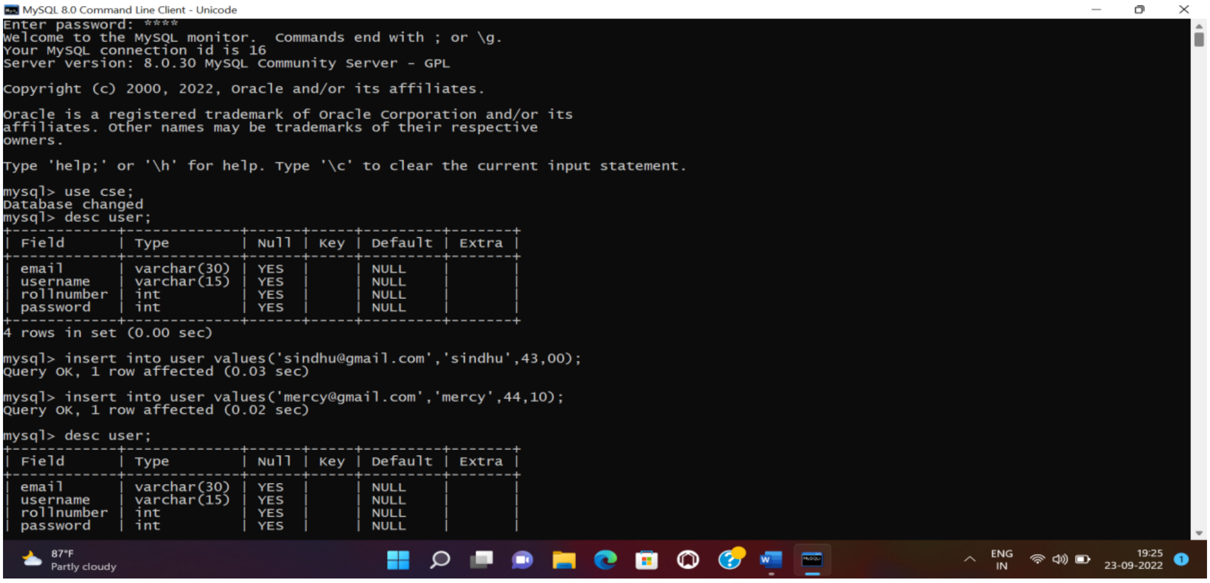
Assignment-2

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

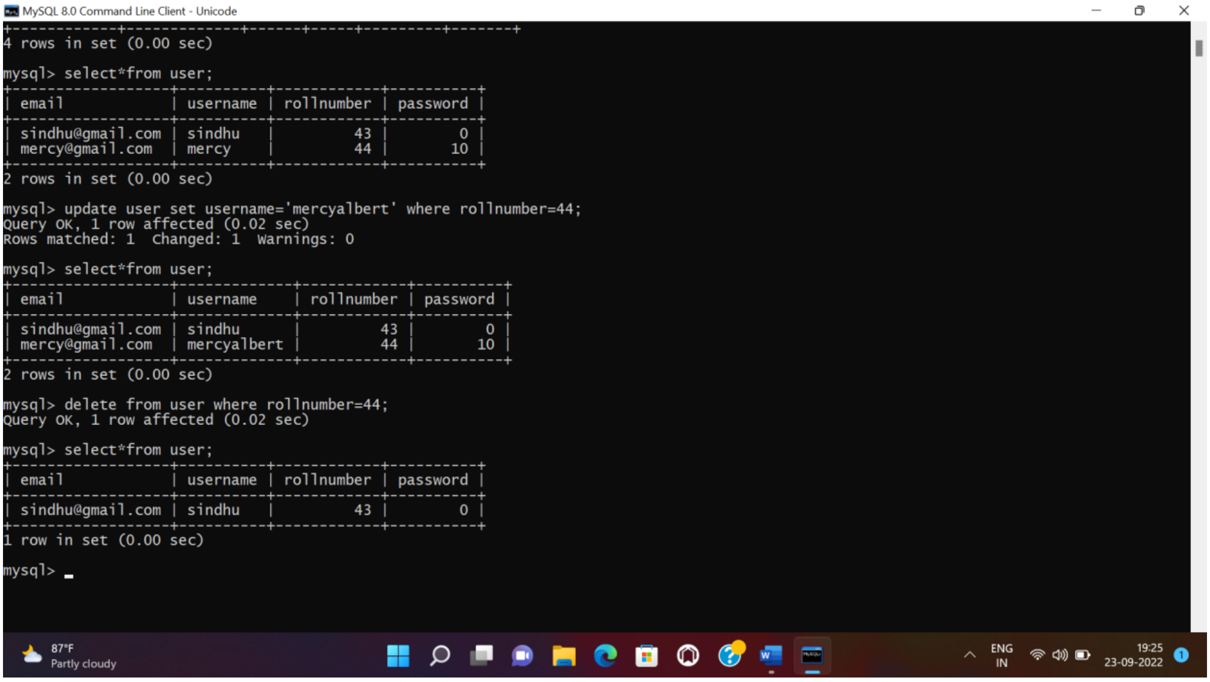


```
affiliates. Other names may be trademarks of their respective
owners.

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

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

mysql> create table user(email varchar(30) notnull,username varchar(15),rollnumber int(10));
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 'notnull,username
varchar(15),rollnumber int(10))' at line 1
mysql> create table user(email varchar(30),username varchar(15),rollnumber int(10),password int(10));
ERROR 1046 (3D000): No database selected
mysql> use cse;
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 (0.04 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 (0.00 sec)

mysql>
```

2.Perform UPDATE, DELETE Queries with user table.



```
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 16
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> use cse;
Database changed
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 (0.00 sec)

mysql> insert into user values('sindhu@gmail.com','sindhu',43,00);
Query OK, 1 row affected (0.03 sec)

mysql> insert into user values('mercy@gmail.com','mercy',44,10);
Query OK, 1 row affected (0.02 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 (0.00 sec)

mysql> select*from user;
+------------------+----------+------------+----------+
| email            | username | rollnumber | password |
+------------------+----------+------------+----------+
| sindhu@gmail.com | sindhu   |         43 |        0 |
| mercy@gmail.com  | mercy    |         44 |       10 |
+------------------+----------+------------+----------+
2 rows in set (0.00 sec)

mysql> update user set username='mercyalbert' where rollnumber=44;
Query OK, 1 row affected (0.02 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select*from user;
+------------------+-------------+------------+----------+
| email            | username    | rollnumber | password |
+------------------+-------------+------------+----------+
| sindhu@gmail.com | sindhu      |         43 |        0 |
| mercy@gmail.com  | mercyalbert |         44 |       10 |
+------------------+-------------+------------+----------+
2 rows in set (0.00 sec)

mysql> delete from user where rollnumber=44;
Query OK, 1 row affected (0.02 sec)

mysql> select*from user;
+------------------+----------+------------+----------+
| email            | username | rollnumber | password |
+------------------+----------+------------+----------+
| sindhu@gmail.com | sindhu   |         43 |        0 |
+------------------+----------+------------+----------+
1 row in set (0.00 sec)

mysql> _
```

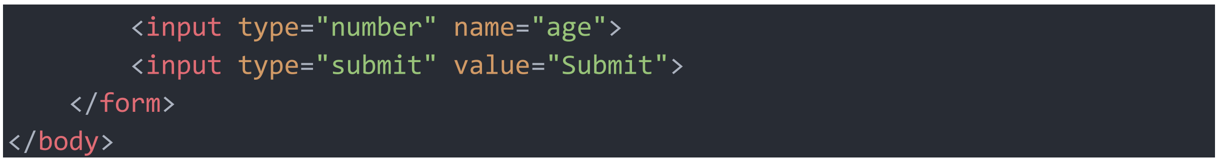
3.Connect python code to db2.

App.py

```
from flask import Flask, render_template ,request,redirect,session,url_for
import ibm_db
import re
app = Flask(__name__)

conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=2f3279a5-73d1-4859-88f0-a6c3e6b4b907.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;PORT=30756;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=mpf67194;PWD=iCG3PvOYvDiHMrq8",'','')
@app.route('/')
def home():
    return render_template('register.html')

@app.route('/login',methods=['GET','POST'])
def login():
    global Userid
    msg= ''

    if request.method=='POST':
        Name =request.form['name']
        Age= request.form['age']
        sql ="SELECT * FROM login WHERE name= ? AND age=?"
        stmt = ibm_db.prepare(conn,sql)
        ibm_db.bind_param(stmt,1,'name')
        ibm_db.bind_param(stmt,2,'age')
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print(account)
        if account:
            session['Loggedin']=True
            session['id']=account['name']
            userid=account['NAME']
            session['name']=account['NAME']
            msg='Registered Successfully'
            return render_template('register.html',msg=msg)

@app.route('/register',methods=['GET','POST'])
def register():
    if request.method =='POST':
        name=request.form['name']
        age=request.form['age']
        sql = "SELECT *FROM login WHERE name=?"
        stmt = ibm_db.prepare(conn,sql)
        ibm_db.bind_param(stmt,1,name)
        ibm_db.execute(stmt)
        account=ibm_db.fetch_assoc(stmt)
        print(account)
        if account:
            msg ="Account already exist"
        else:
            insert_sql="INSERT INTO LOGIN VALUES(?,?)"
            stmt=ibm_db.prepare(conn,insert_sql)
            ibm_db.bind_param(stmt,1,name)
            ibm_db.bind_param(stmt,2,age)
            ibm_db.execute(stmt)
            msg="You have successfully registerd"
    elif request.method=="POST":
            msg = "Please fill the form"
    else:
       return "Error password or user not match"

    return render_template('register.html',msg=msg)
if __name__=='main_':
    app.run(debug=True)
```

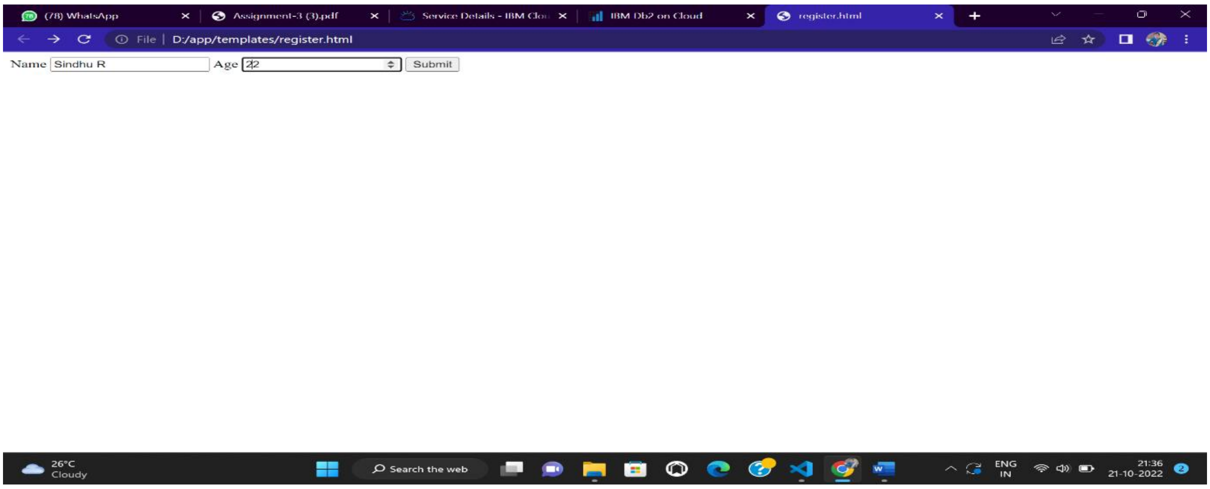
register.html

```
<!DOCTYPE html>
<body>
    <form action="/register" method="post">
        <label for="name">Name</label>
        <input type="text" name="name">
        <label for="age" >Age</label>
```



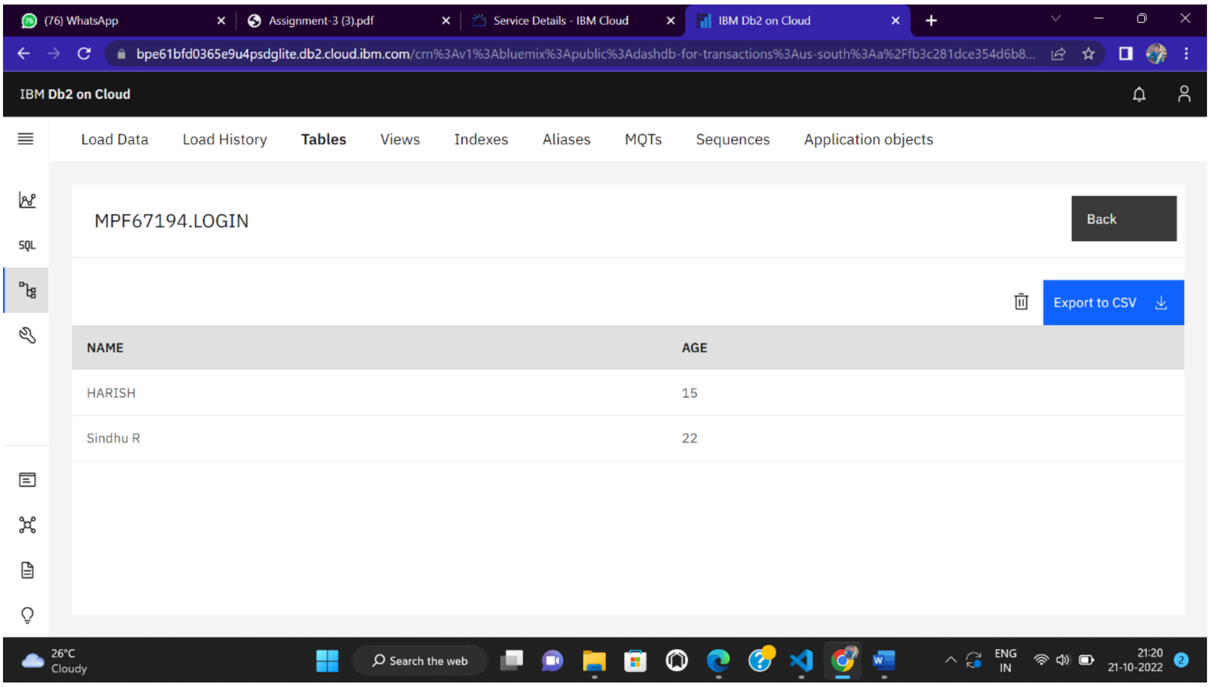
```
        <input type="number" name="age">
        <input type="submit" value="Submit">
    </form>
</body>
```

Output:



D:/app/templates/register.html

Name Sindhu R Age 22 Submit



bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2Ffb3c281dce354d6b8...

IBM **Db2 on Cloud**

MPF67194.LOGIN

NAME	AGE
HARISH	15
Sindhu R	22

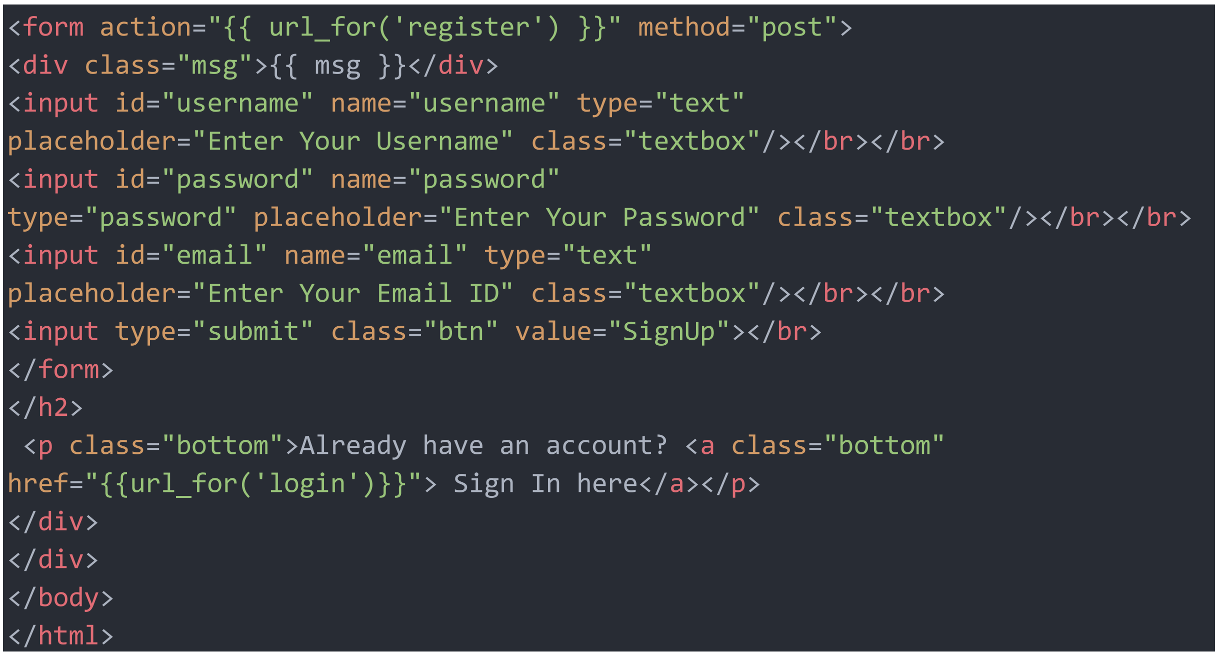
4. Create a flask app with registration page, login page and welcome page. By default load the registration page once the user enters all the fields store the data in database and navigate to login page authenticate user username and password. If the user is valid show
the welcome page.

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=2f3279a5-73d1-4859-88f0-
a6c3e6b4b907.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;PORT=30756;SECURIT
Y=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=mpf67194;PWD=iCG3PvOYv
DiHMrq8",'','')
@app.route('/')
@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 users 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('wel.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' and 'username' in request.form and 'password'
in request.form and 'email' in request.form :
       username = request.form['username']
       password = request.form['password']
       email = request.form['email']
       stmt = ibm_db.prepare(conn,'SELECT * FROM users WHERE username = ?')
       ibm_db.bind_param(stmt,1,username)
       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]+', 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:
           prep_stmt = ibm_db.prepare(conn,"INSERT INTO users VALUES(?, ?,
?)")
           ibm_db.bind_param(prep_stmt, 1, username)
           ibm_db.bind_param(prep_stmt, 2, email)
           ibm_db.bind_param(prep_stmt, 3, password)
           ibm_db.execute(prep_stmt)
           msg = 'You have successfully registered !'
    elif request.method == 'POST':
        msg = 'Please fill out the form !'
    return render_template('reg.html', msg = msg)
if __name__ == '__main__':
 app.debug = True
```

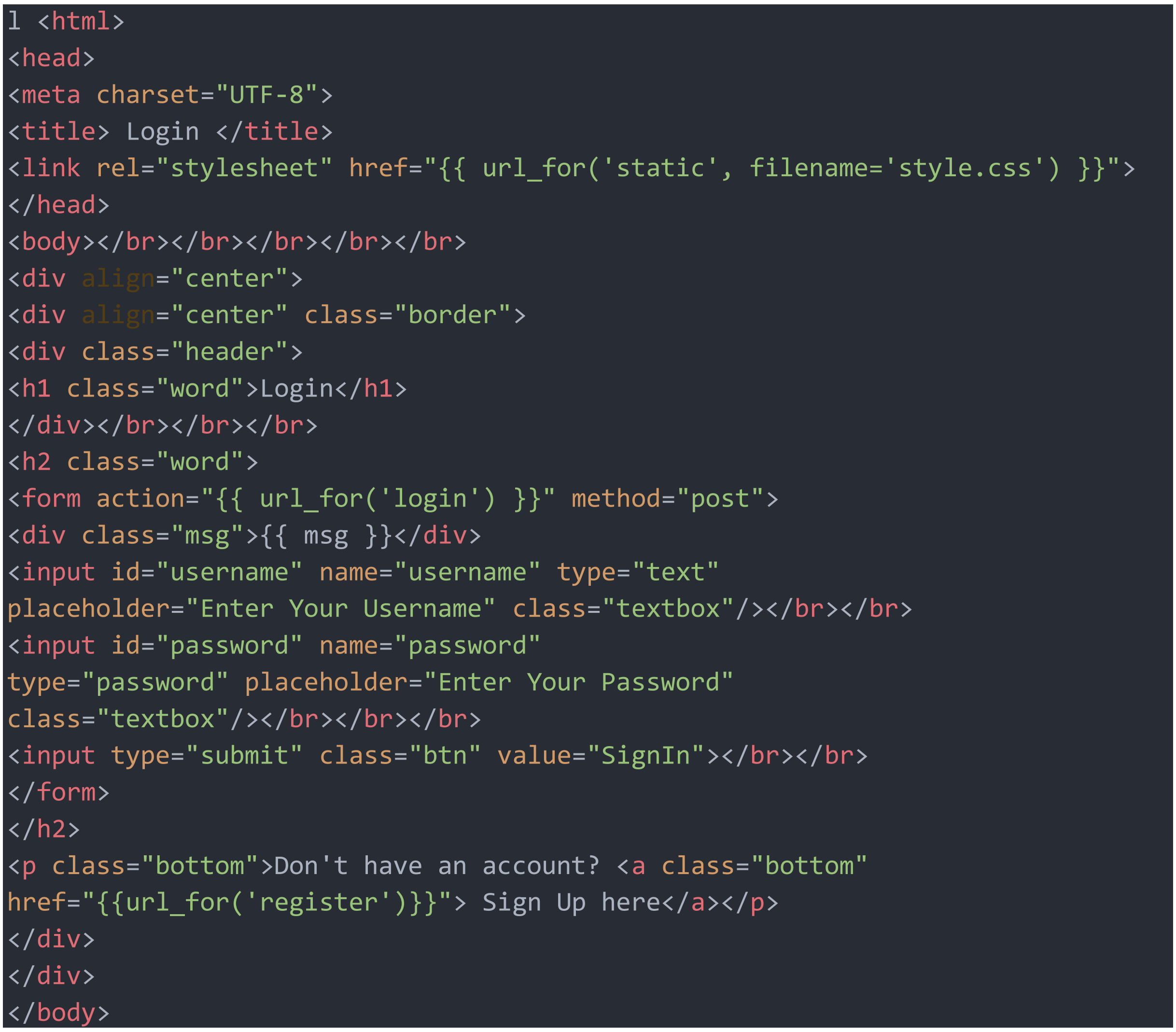
Reg.html

```
<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="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>
```

Login.html



```
l <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="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>
```

Wel.html

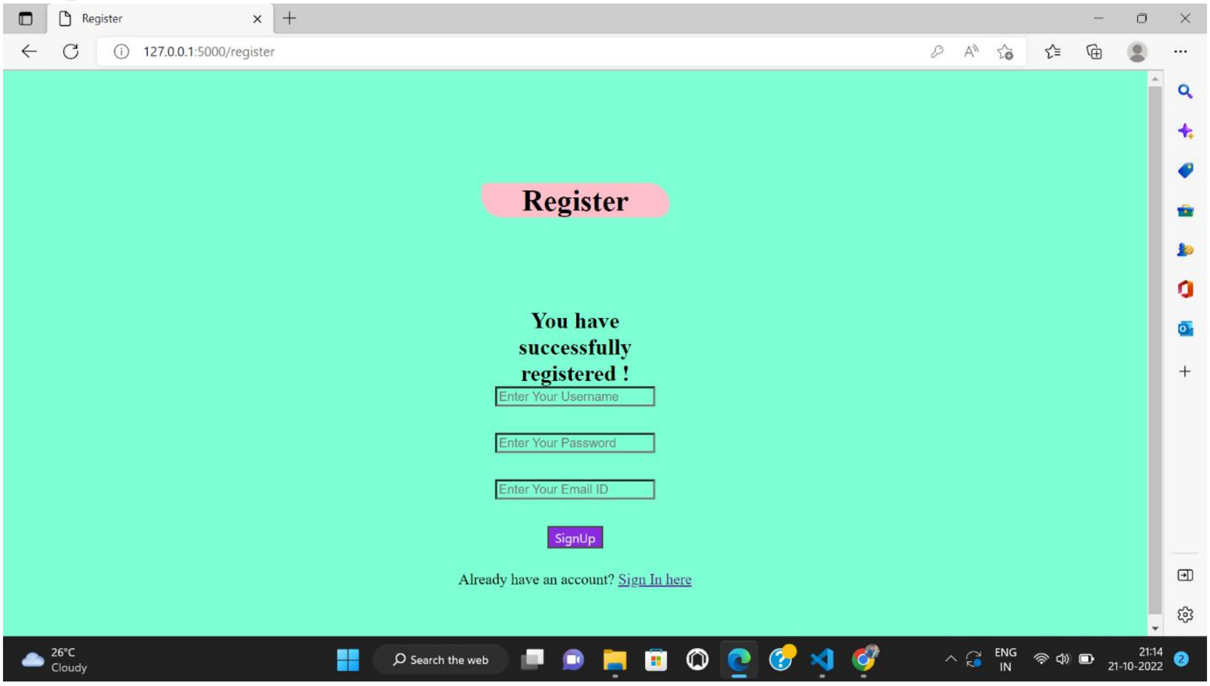
```
<html>
<head>
<meta charset="UTF-8">
<title> WELCOME PAGE </title>
<link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
</head>
<body></br></br></br></br></br>
<div align="center">
<div align="center" class="border">
<div class="header">
<h1 class="word">WELCOME PAGE </h1>
</div></br></br></br>
<h1 class="bottom">
Hi {{session.username}}!!</br></br> Welcome to Our
page...
</h1></br></br></br>
<a href="{{ url_for('logout') }}" class="btn">Logout</a>
</div>
</div>
</body>
</html>
```

Style.css

```
body, html{
    height: 100%;

}
* {
box-sizing: border-box;
background-color: aquamarine;
}

.border {
   margin: auto;
}
.btn{
    background-color: blueviolet;
    color: aliceblue;
    border: 2px solid #4c4c4c;
    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
}
```

```
.word{
    border-radius: 15px 50px 30px;
    background: pink;
    width: 200px;
}
```

Output:



Register

You have successfully registered !

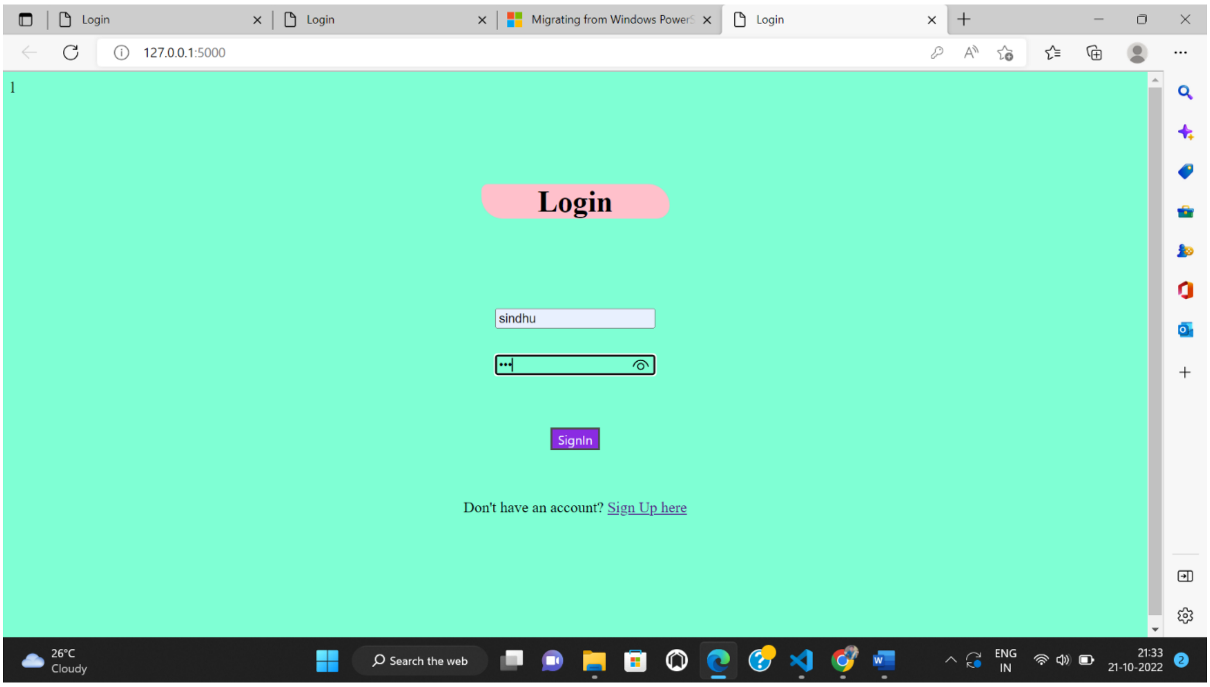
Enter Your Username

Enter Your Password

Enter Your Email ID

SignUp

Already have an account? Sign In here



1

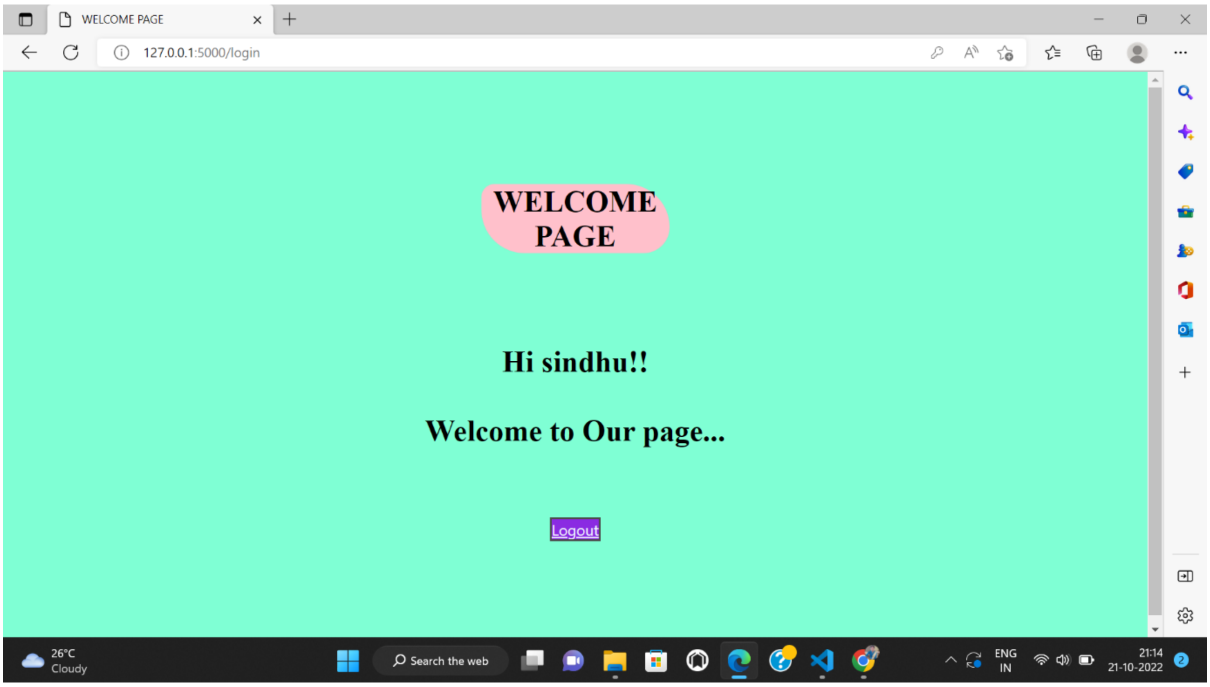
Login

sindhu

•••

SignIn

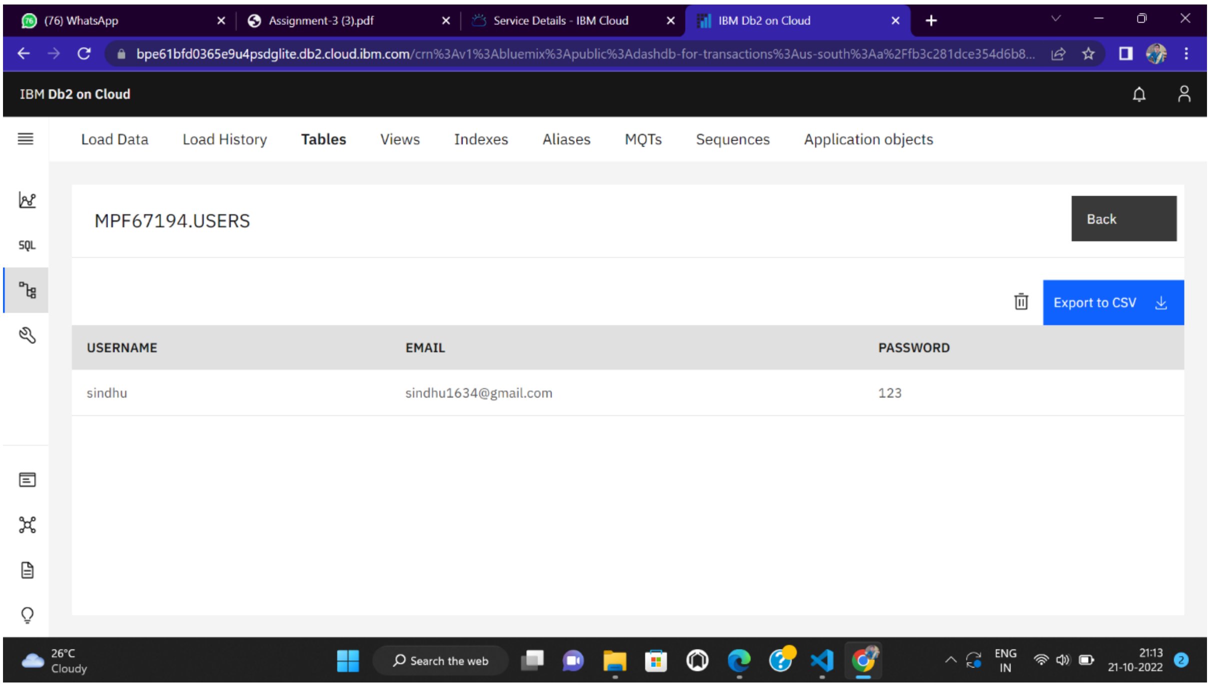
Don't have an account? Sign Up here



WELCOME PAGE

Hi sindhu!!

Welcome to Our page...



MPF67194.USERS

USERNAME	EMAIL	PASSWORD
sindhu	sindhu1634@gmail.com	123