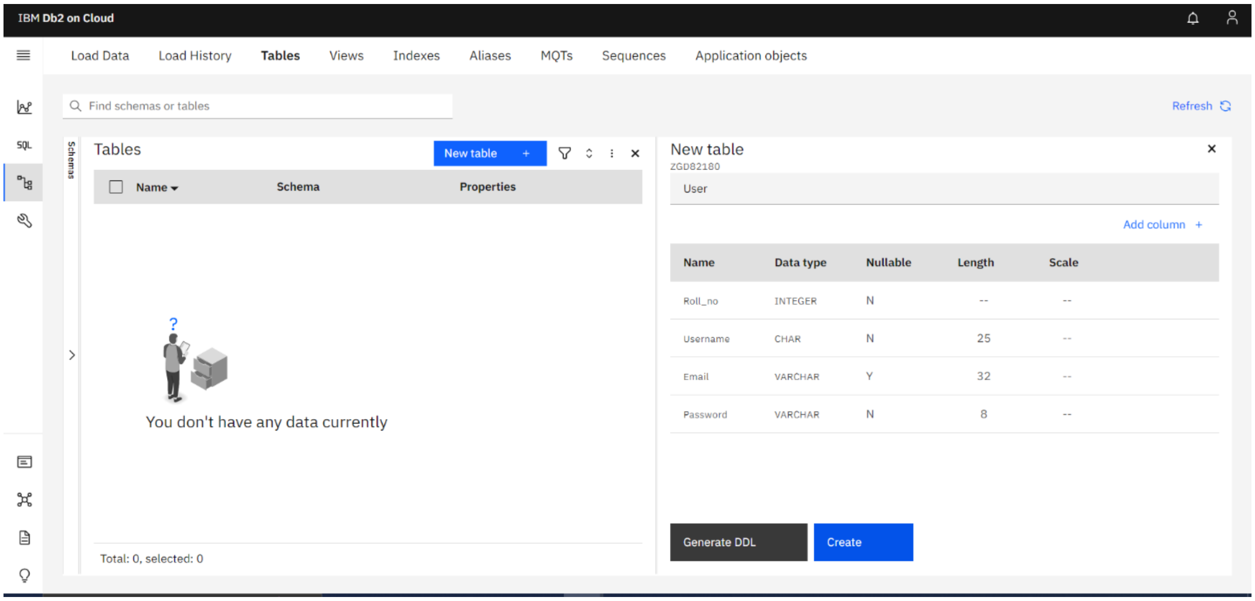
Assignment -2

Assignment Date	26 September 2022
Team ID	PNT2022TMID28027
Project Name	Personal Expense Tracker Application
Maximum Marks	2 Marks

Question-1:

Create User table with email, username, roll number, password.



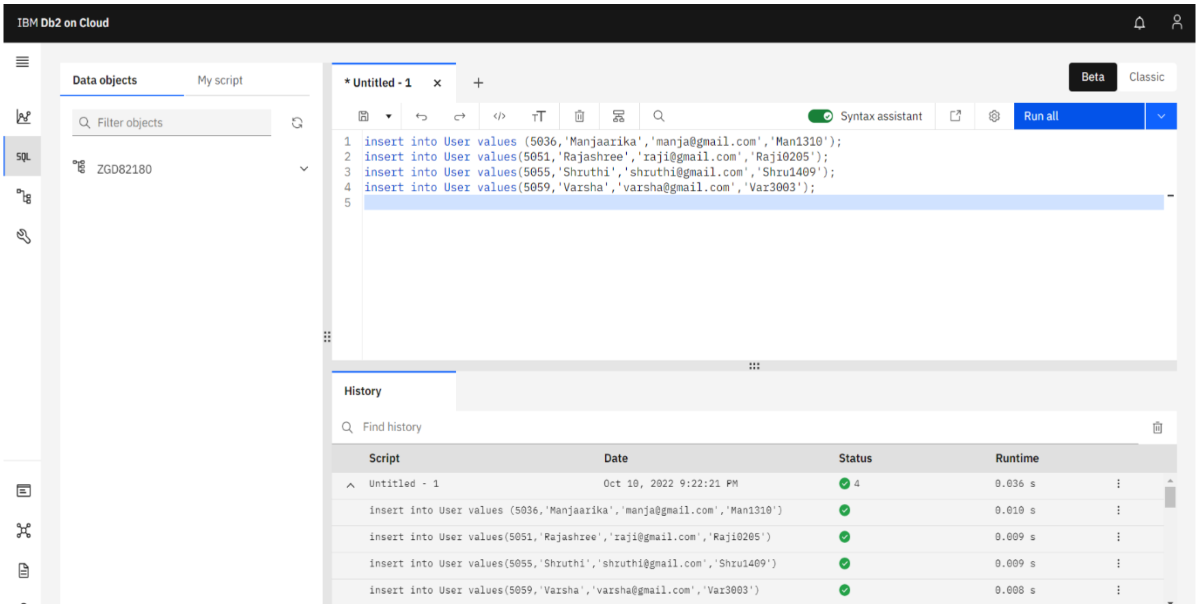
New table

ZGD82180

User

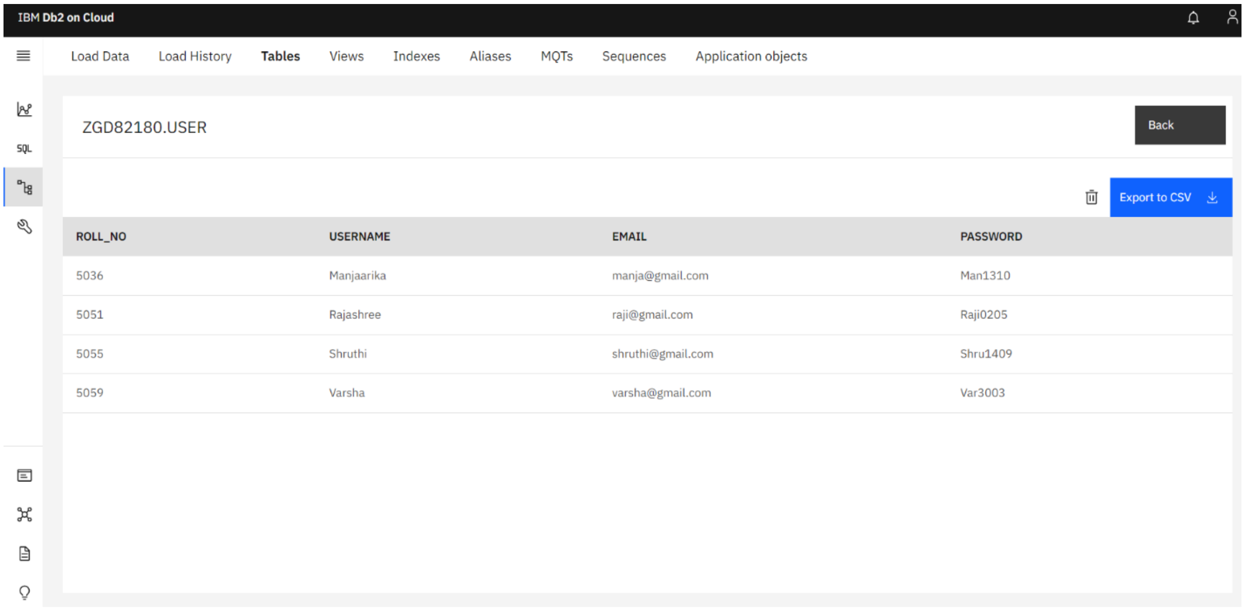
Name	Data type	Nullable	Length	Scale
Roll_no	INTEGER	N	--	--
Username	CHAR	N	25	--
Email	VARCHAR	Y	32	--
Password	VARCHAR	N	8	--

Insert value into table



```
insert into User values (5036,'Manjaarika','manja@gmail.com','Man1310');
insert into User values(5051,'Rajashree','raji@gmail.com','Raji0205');
insert into User values(5055,'Shruthi','shruthi@gmail.com','Shru1409');
insert into User values(5059,'Varsha','varsha@gmail.com','Var3003');
```

Script	Date	Status	Runtime
Untitled - 1	Oct 10, 2022 9:22:21 PM	✓ 4	0.036 s
insert into User values (5036,'Manjaarika','manja@gmail.com','Man1310')		✓	0.010 s
insert into User values(5051,'Rajashree','raji@gmail.com','Raji0205')		✓	0.009 s
insert into User values(5055,'Shruthi','shruthi@gmail.com','Shru1409')		✓	0.009 s
insert into User values(5059,'Varsha','varsha@gmail.com','Var3003')		✓	0.008 s



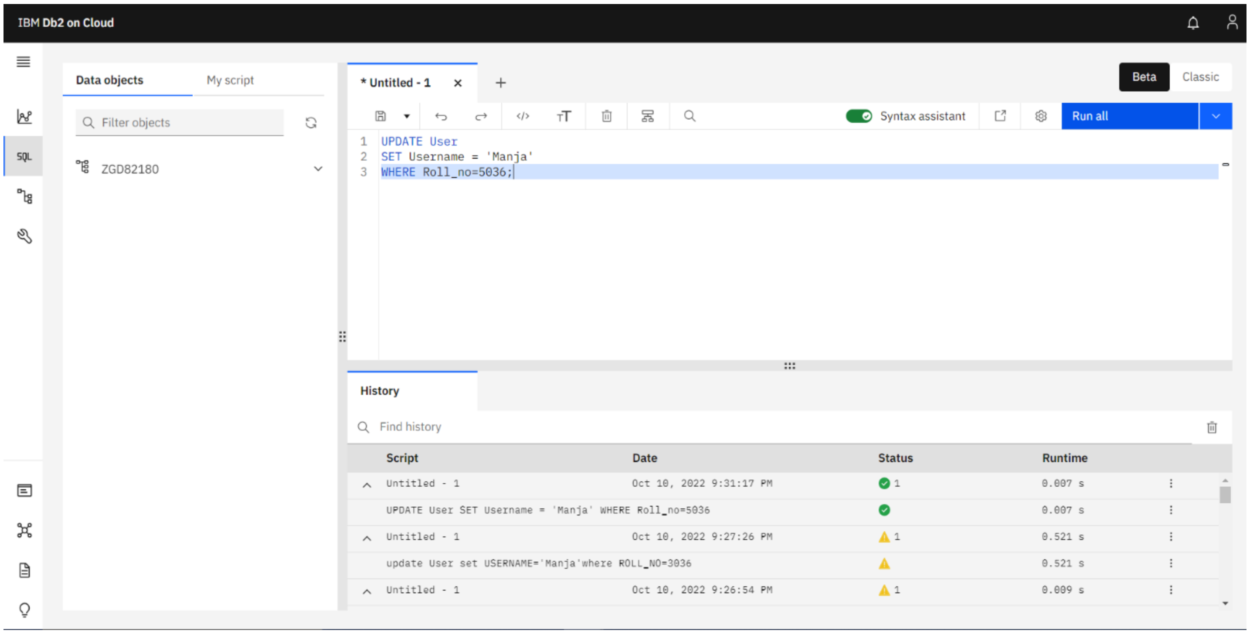
ZGD82180.USER

ROLL_NO	USERNAME	EMAIL	PASSWORD
5036	Manjaarika	manja@gmail.com	Man1310
5051	Rajashree	raji@gmail.com	Raji0205
5055	Shruthi	shruthi@gmail.com	Shru1409
5059	Varsha	varsha@gmail.com	Var3003

Question-2:

Perform UPDATE, DELETE Queries with user table

Update the table

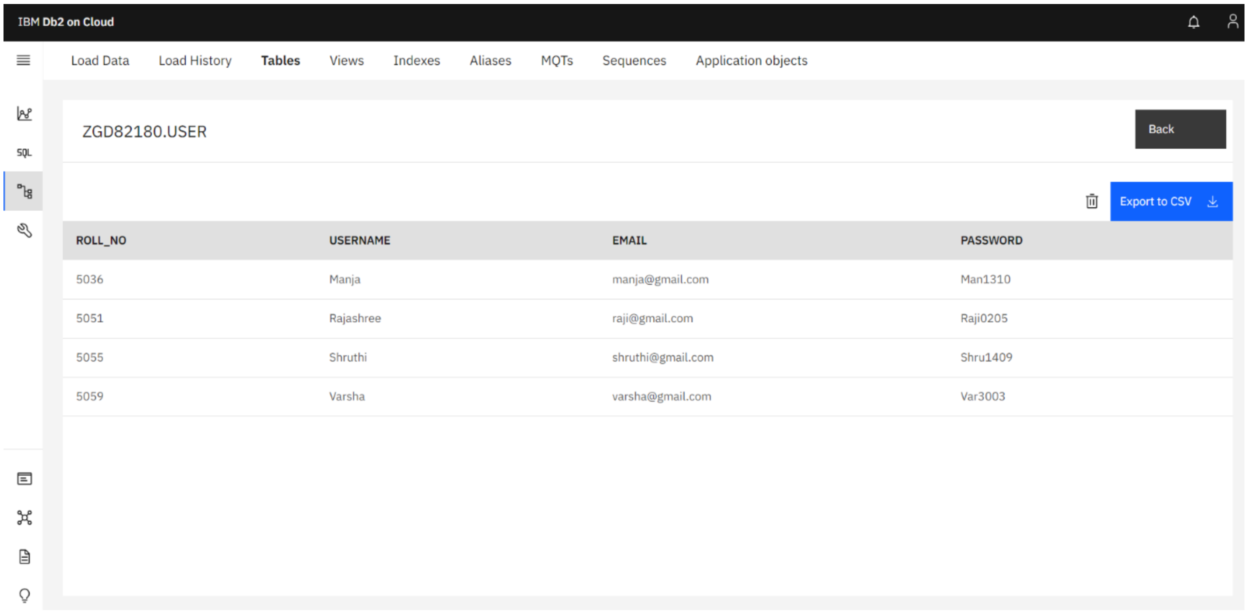


```
UPDATE User
SET Username = 'Manja'
WHERE Roll_no=5036;
```

History

Find history

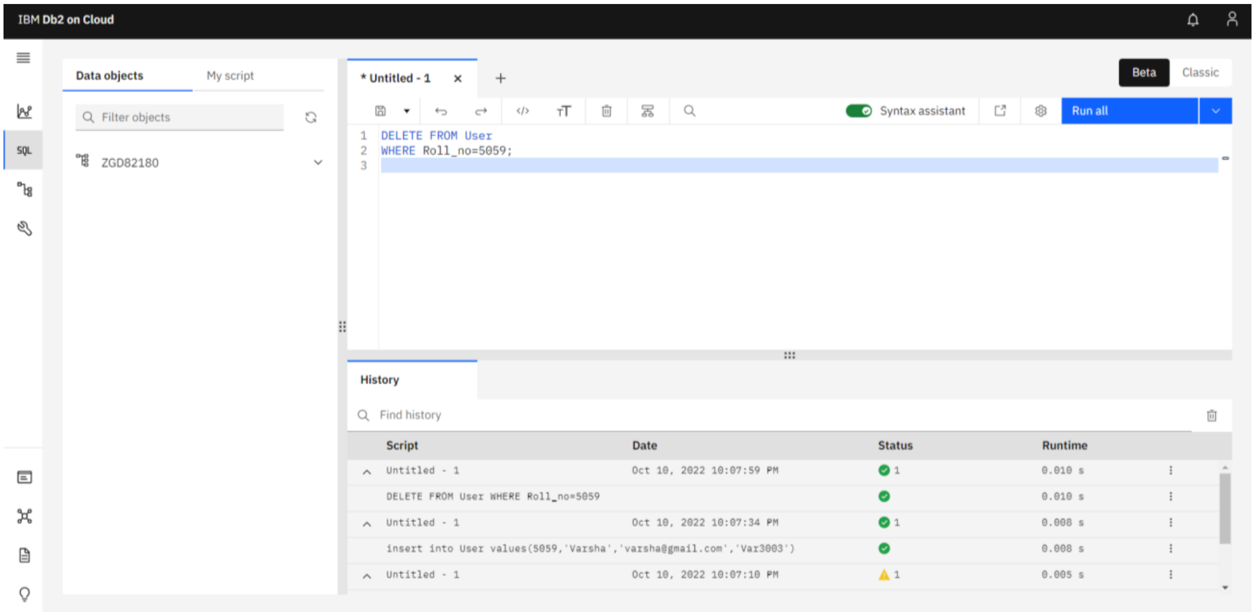
	Script	Date	Status	Runtime
^	Untitled - 1	Oct 10, 2022 9:31:17 PM	✓ 1	0.007 s
	UPDATE User SET Username = 'Manja' WHERE Roll_no=5036		✓	0.007 s
^	Untitled - 1	Oct 10, 2022 9:27:26 PM	⚠ 1	0.521 s
	update User set USERNAME='Manja'where ROLL_NO=3036		⚠	0.521 s
^	Untitled - 1	Oct 10, 2022 9:26:54 PM	⚠ 1	0.009 s



ZGD82180.USER

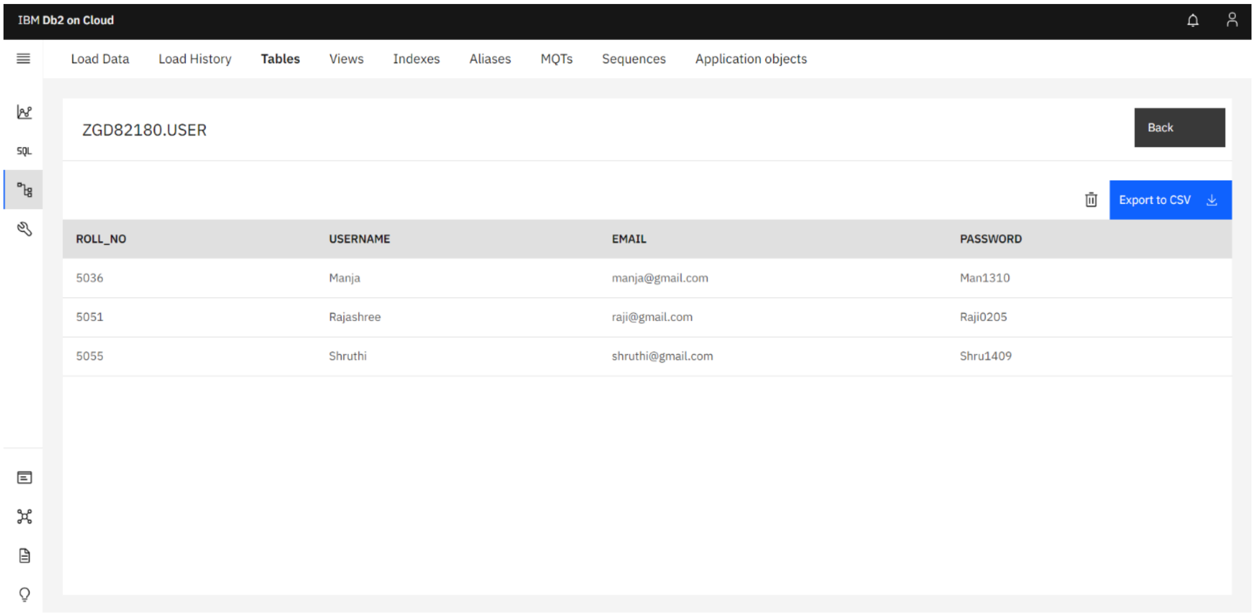
ROLL_NO	USERNAME	EMAIL	PASSWORD
5036	Manja	manja@gmail.com	Man1310
5051	Rajashree	raji@gmail.com	Raji0205
5055	Shruthi	shruthi@gmail.com	Shru1409
5059	Varsha	varsha@gmail.com	Var3003

Delete a row in the table



```
DELETE FROM User
WHERE Roll_no=5059;
```

	Script	Date	Status	Runtime
^	Untitled - 1	Oct 10, 2022 10:07:59 PM	✓ 1	0.010 s
	DELETE FROM User WHERE Roll_no=5059		✓	0.010 s
^	Untitled - 1	Oct 10, 2022 10:07:34 PM	✓ 1	0.008 s
	insert into User values(5059,'Varsha','varsha@gmail.com','Var3003')		✓	0.008 s
^	Untitled - 1	Oct 10, 2022 10:07:10 PM	⚠ 1	0.005 s



ZGD82180.USER

ROLL_NO	USERNAME	EMAIL	PASSWORD
5036	Manja	manja@gmail.com	Man1310
5051	Rajashree	raji@gmail.com	Raji0205
5055	Shruthi	shruthi@gmail.com	Shru1409

Question 3:

Connect python code to db2

Program:

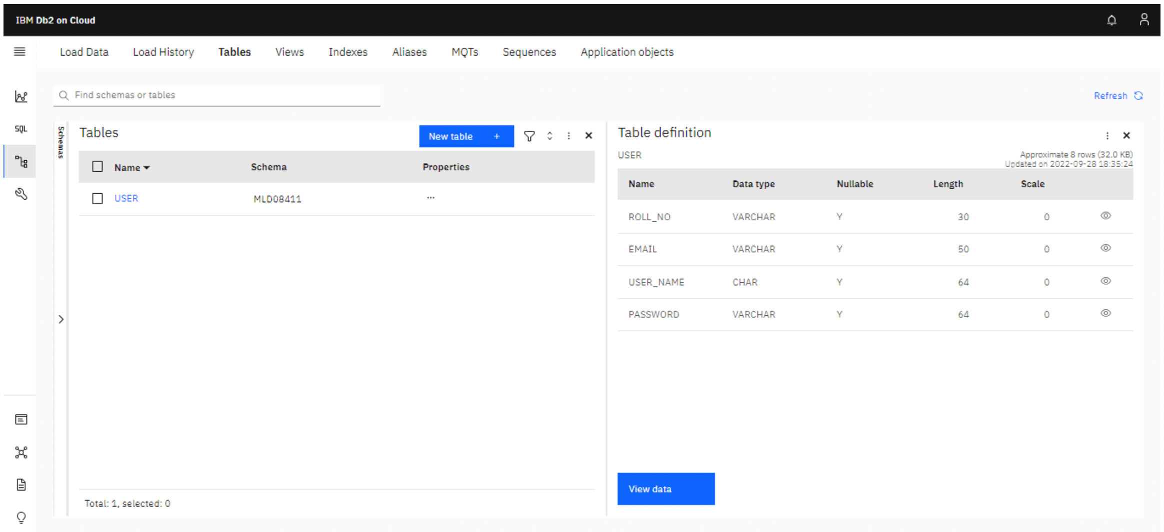
```
from flask import Flask
import ibm_db
app=Flask(__name__)
app.secret_key='a'
try:
   conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=2f3279a5-73d1-4859-88f0-a6c3e6b4b907.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;PORT=30756;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=mld08411;PWD=mCn9EnGnLGGbg2JH","","")

except:
   print("Unable to connect: ",ibm_db.conn_error())
@app.route("/")
def dashboard():
   return “welcome”
if __name__=='__main__':
   app.run(debug=True)
```

Question 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 username and password. If the user is valid show the welcome page.

Database:



Tables

☐ Name	Schema	Properties
☐ USER	MLD08411	...

Total: 1, selected: 0

Table definition

USER

Approximate 8 rows (32.0 KB)
Updated on 2022-09-28 18:35:24

Name	Data type	Nullable	Length	Scale
ROLL_NO	VARCHAR	Y	30	0
EMAIL	VARCHAR	Y	50	0
USER_NAME	CHAR	Y	64	0
PASSWORD	VARCHAR	Y	64	0

Program:

User.py

```
from flask import Flask,flash,render_template,request,redirect,url_for,session
import ibm_db
app=Flask(__name__)
app.secret_key='a'
try:
    conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=2f3279a5-73d1-4859-88f0-a6c3e6b4b907.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;PORT=30756;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=mld08411;PWD=mCn9EnGnLGGbg2JH","","")
except:
    print("Unable to connect: ",ibm_db.conn_error())

@app.route("/")
def dash():
    return render_template('register.html',msg=" ")
```

```
@app.route("/register",methods=['GET','POST'])
def register():
    error = None
    if request.method=='POST':
           username=request.form['username']
           email=request.form['email']
           roll_number=request.form['rollnumber']
           password=request.form['password']
           sql="SELECT * FROM user WHERE roll_no=?"
           prep_stmt=ibm_db.prepare(conn,sql)
           ibm_db.bind_param(prep_stmt,1,roll_number)
           ibm_db.execute(prep_stmt)
           account=ibm_db.fetch_assoc(prep_stmt)
           print(account)
           if account:
               error="Account already exists! Log in to continue !"
           else:
               insert_sql="INSERT INTO user values(?,?,?,?)"
               prep_stmt=ibm_db.prepare(conn,insert_sql)
               ibm_db.bind_param(prep_stmt,1,roll_number)
               ibm_db.bind_param(prep_stmt,2,email)
               ibm_db.bind_param(prep_stmt,3,username)
               ibm_db.bind_param(prep_stmt,4,password)
               ibm_db.execute(prep_stmt)
               flash(" Registration successfull. Log in to continue !")
    else:
        pass
    return render_template('login.html',error=error)

@app.route('/login',methods=['GET','POST'])
def login():
    error = None
    if request.method=='POST':
        username=request.form['username']
        password=request.form['password']
        sql="SELECT * FROM user WHERE user_name=? 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['USER_NAME']
            session["username"]=account["USER_NAME"]
            flash("Logged in successfully!")
            return redirect(url_for("welcome_page"))
        else:
            error="Incorrect username / password"
            return render_template('login.html',error=error)
@app.route('/welcome')
def welcome_page():
    return render_template("welcome.html",user=session['id'])
if __name__=='__main__':
    app.run(debug=True)
```

register.html

```
<html>
    <head>
        <title>Registration page</title>
        <link rel="stylesheet" href="{{url_for('static', filename='style.css')}}">
    </head>
    <body>
        <div class="center"> <br /><br />
        <form action="/register" method="POST">
            <h2>Student Registration Portal</h2> <br /><br />
            <p> Enter Email ID:</p>
            <input type="email" name="email"/>
            <p> Enter Username:</p>
            <input type="text" name= "username" />
            <p> Enter Password:</p>
            <input type="password" name="password"/>
            <p> Enter Roll number:</p>
            <input type="text" name="rollnumber"/>
            </br></br></br>
            <input type="submit" value="submit" />
        </form>
        </div>
    </body>
</html>
```

login.html

```
<html>
    <head>
        <title>Login page</title>
        <link rel="stylesheet" href="{{url_for('static', filename='style.css')}}">
    </head>
    <body>
    {% if error %}
        <p><strong style="color:red">Error</strong>: {{error}}</p>
    {% endif %}
    {% with messages = get_flashed_messages() %}
         {% if messages %}
               {% for message in messages %}
                    <p style="color:green">{{ message }}</p>
               {% endfor %}
         {% endif %}
      {% endwith %}
        <div class="center"> <br /><br />
        <form action="/login" method="POST">
         <h1>Login</h1> <br /><br />
            <p> Enter Username:</p>
            <input type="text" name= "username" />
            <p> Enter Password:</p>
            <input type="password" name="password"/>
            </br></br></br>
            <input type="submit" value="submit" />
        </form>
        </div>
    </body>
</html>
```

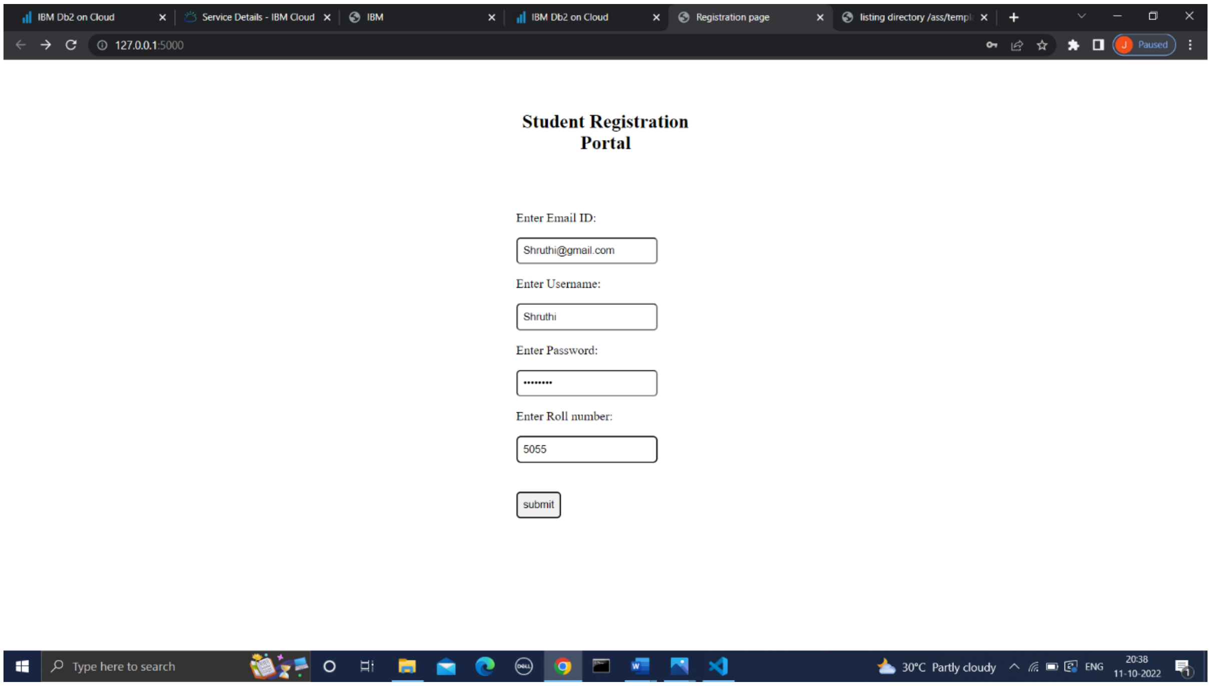
welcome.html

```
<html>
    <head>
        <title>Login page</title>
        <link rel="stylesheet" href="{{url_for('static', filename='style.css')}}">
    </head>
    <body>
    {% with messages = get_flashed_messages() %}
         {% if messages %}
               {% for message in messages %}
                    <p style="color:green">{{ message }}</p>
               {% endfor %}
         {% endif %}
      {% endwith %}
         <br /><br />
        <center>
         <h2 > Welcome {{user}}</h2>
        </center>
    </body>
</html>
```

style.css

```
.center{
margin: auto;
width: 15%;
}
h2{
text-align:center;
}
input{
border-radius:5px;
padding:7px;
}
```

Output:



Student Registration Portal

Enter Email ID:

Shruthi@gmail.com

Enter Username:

Shruthi

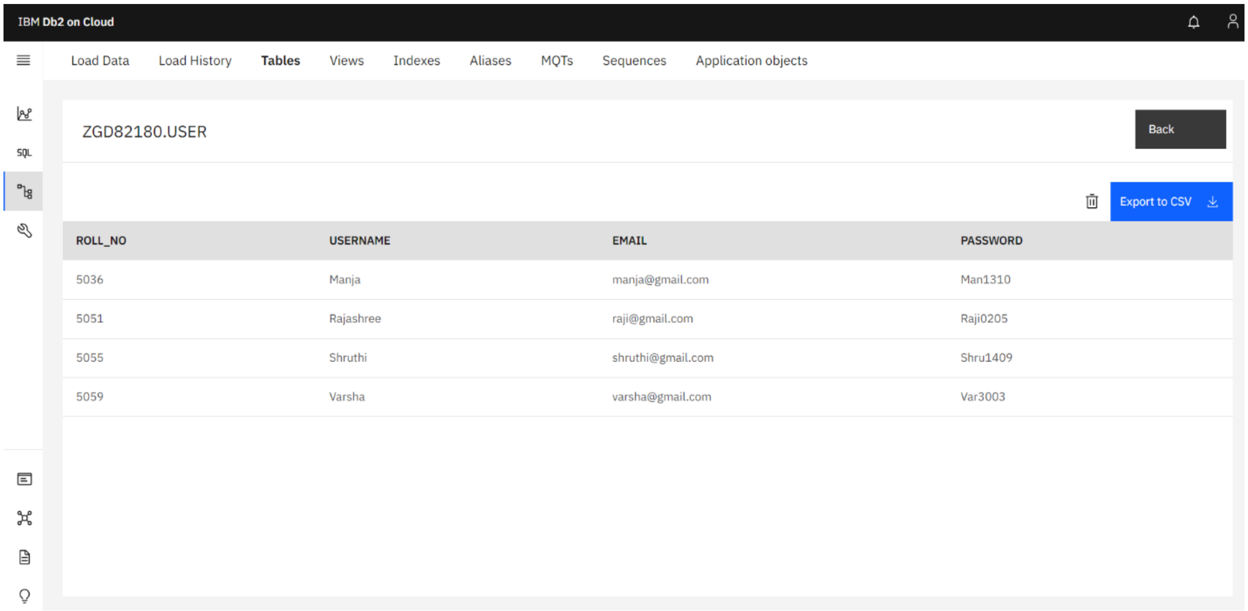
Enter Password:

••••••••

Enter Roll number:

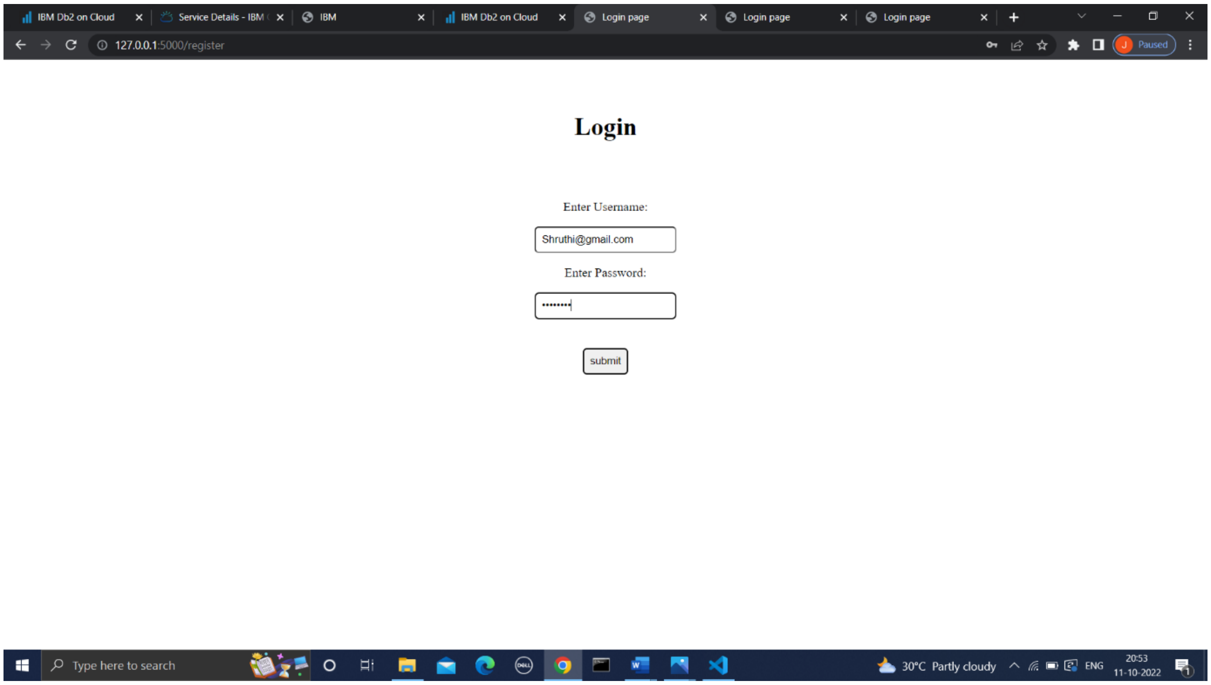
5055

submit



ZGD82180.USER

ROLL_NO	USERNAME	EMAIL	PASSWORD
5036	Manja	manja@gmail.com	Man1310
5051	Rajashree	raji@gmail.com	Raji0205
5055	Shruthi	shruthi@gmail.com	Shru1409
5059	Varsha	varsha@gmail.com	Var3003



Login

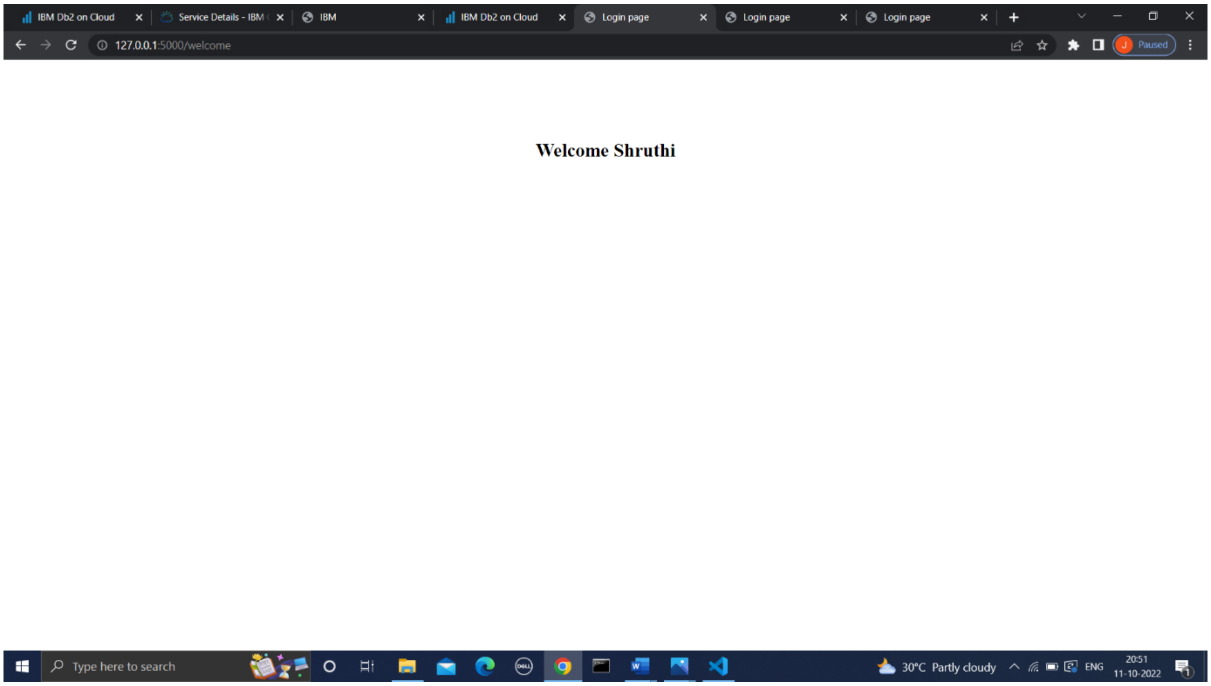
Enter Username:

Shruthi@gmail.com

Enter Password:

••••••••

submit



Welcome Shruthi