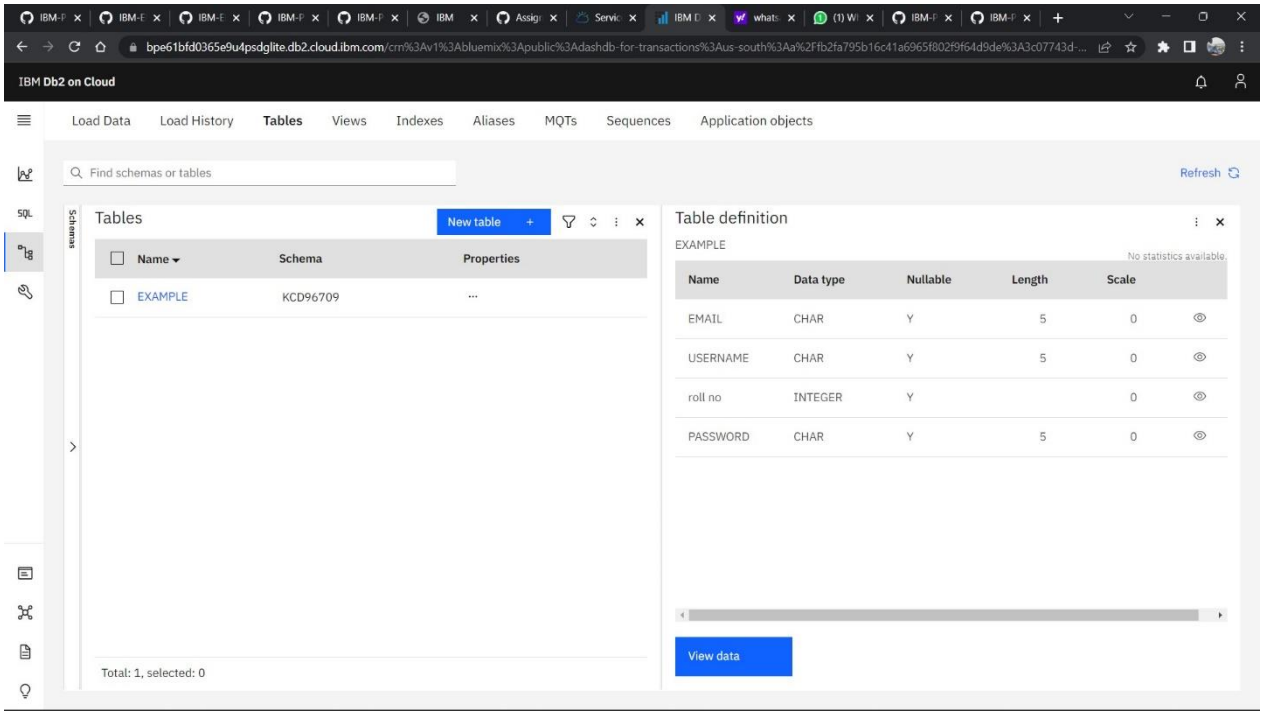
Assignment -2

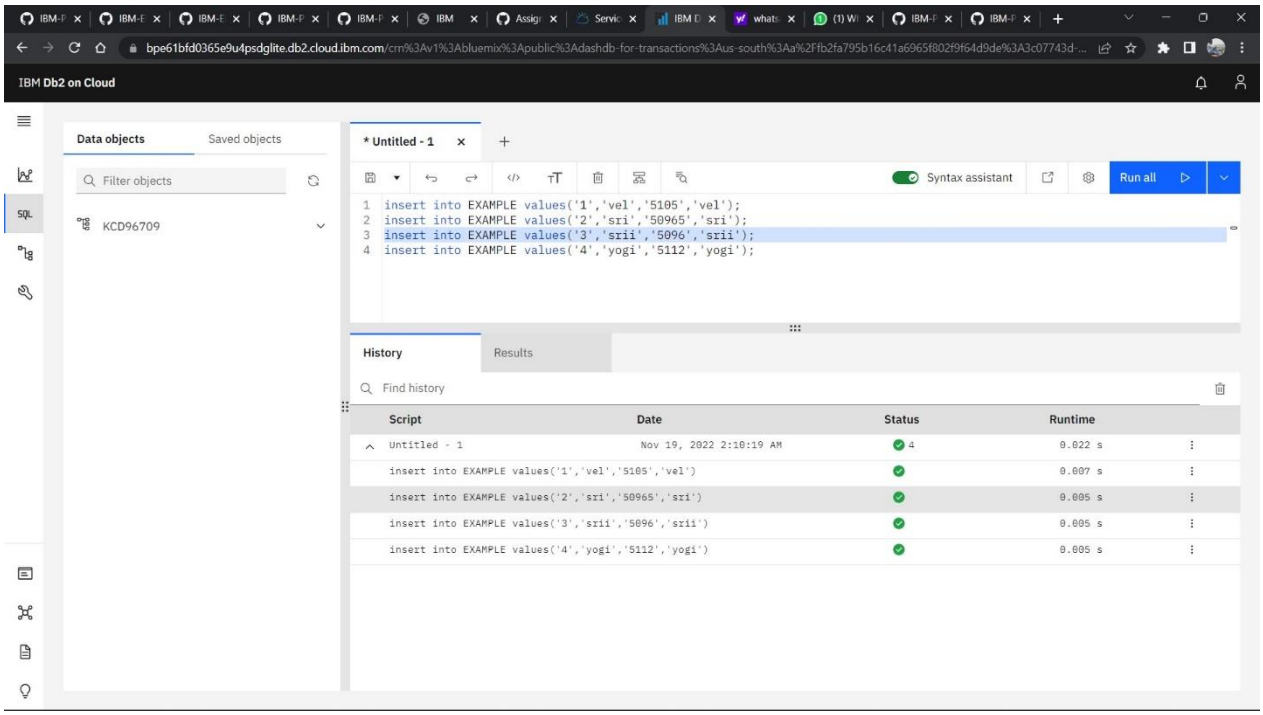
Assignment Date	19 November 2022
Team ID	PNT2022TMID32569
Project Name	Personal Expense Tracker Application
Maximum Marks	2 Marks

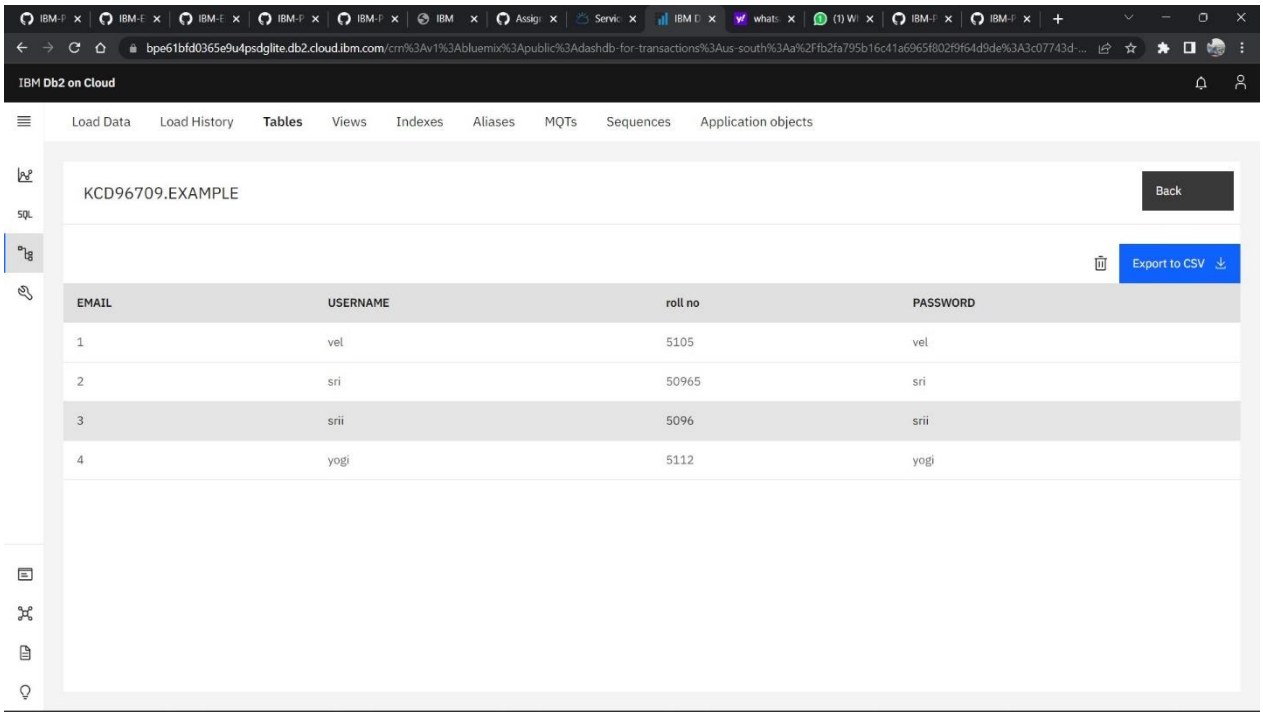
Question-1:

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

Insert value into table



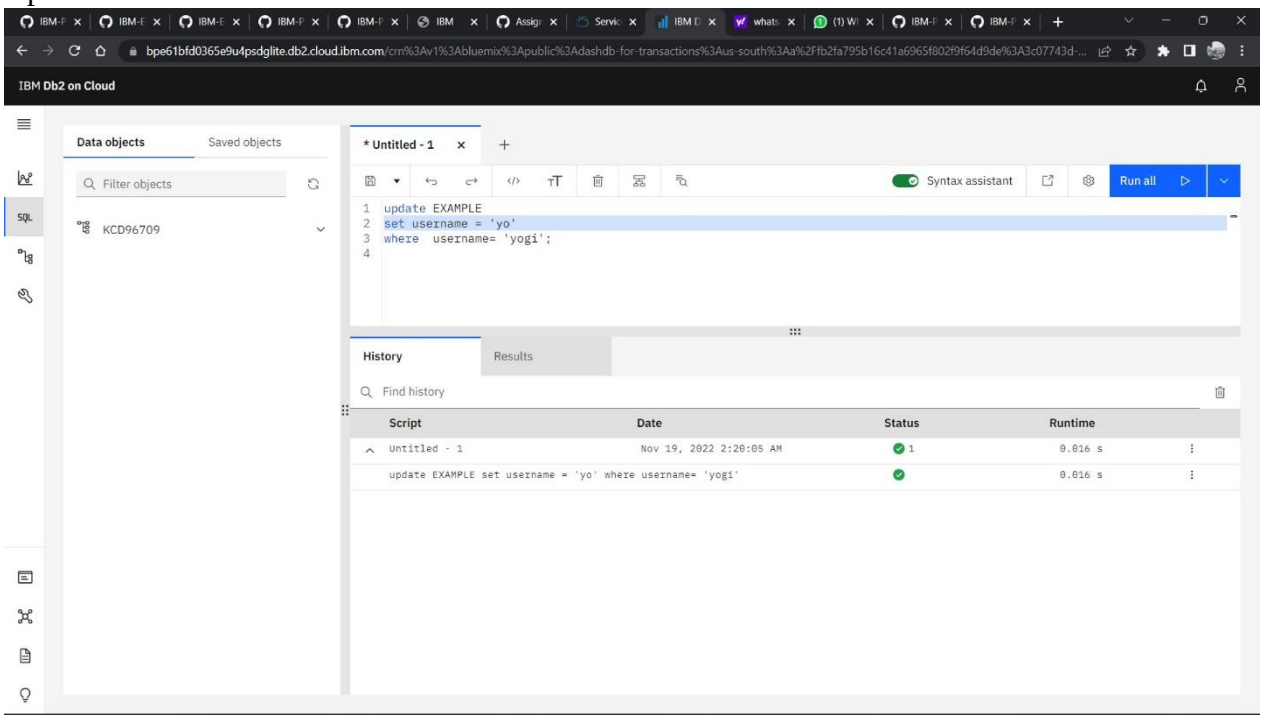

IBM Db2 on Cloud
Data objects
Saved objects
Filter objects
KCD96709
* Untitled - 1
Syntax assistant
Run all
1 insert into EXAMPLE values('1','vel','5105','vel');
2 insert into EXAMPLE values('2','sri','50965','sri');
3 insert into EXAMPLE values('3','srii','5096','srii');
4 insert into EXAMPLE values('4','yogi','5112','yogi');
History
Results
Find history
Script
Date
Status
Runtime
Untitled - 1
Nov 19, 2022 2:10:19 AM
4
0.022 s
insert into EXAMPLE values('1','vel','5105','vel')
0.007 s
insert into EXAMPLE values('2','sri','50965','sri')
0.005 s
insert into EXAMPLE values('3','srii','5096','srii')
0.005 s
insert into EXAMPLE values('4','yogi','5112','yogi')
0.005 s

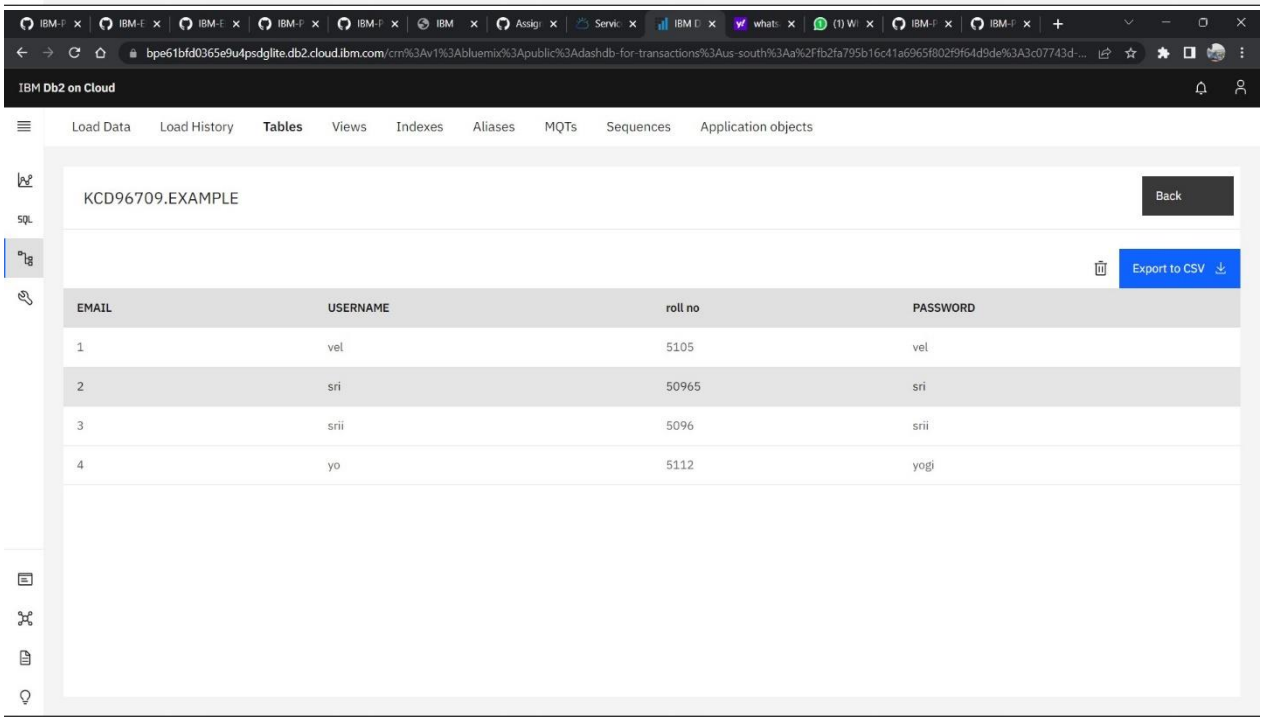

IBM Db2 on Cloud
Load Data
Load History
Tables
Views
Indexes
Aliases
MQTs
Sequences
Application objects
KCD96709.EXAMPLE
Back
Export to CSV
EMAIL
USERNAME
roll no
PASSWORD
1
vel
5105
vel
2
sri
50965
sri
3
srii
5096
srii
4
yogi
5112
yogi

Question-2:

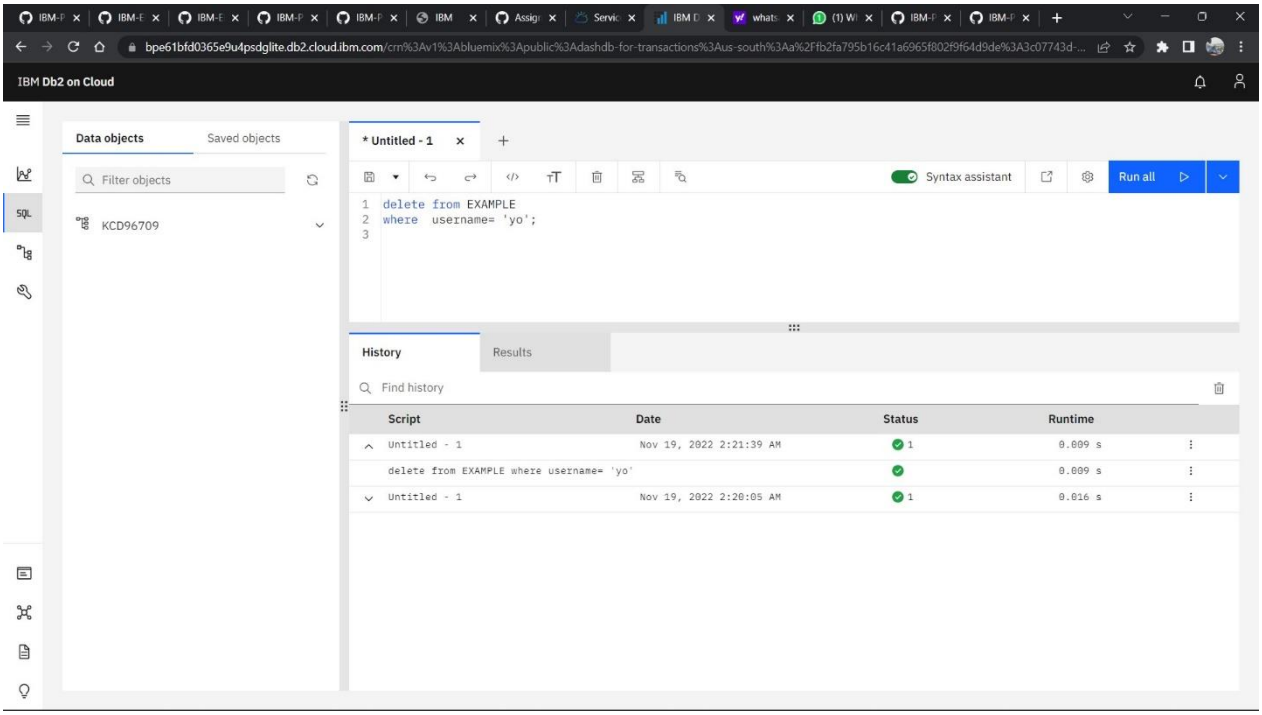
Perform UPDATE, DELETE Queries with user table

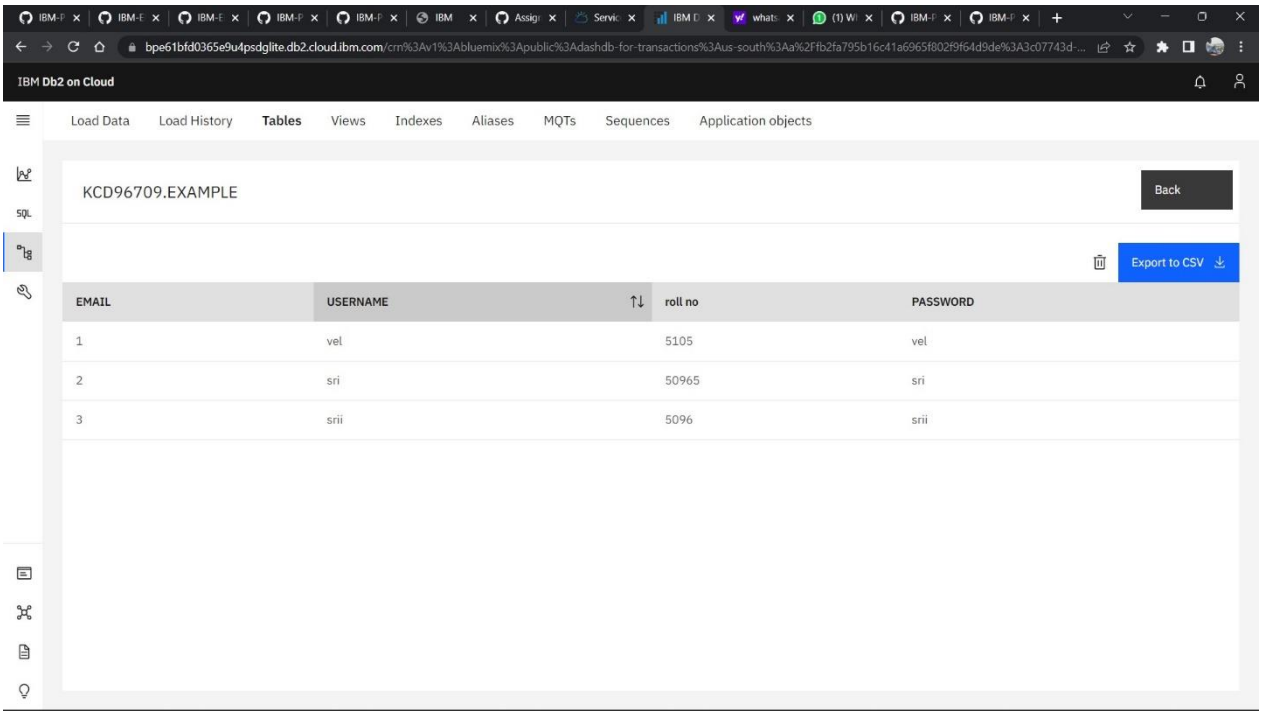
Update the table





Delete a row in the table





Question 3:

Connect python code to db2

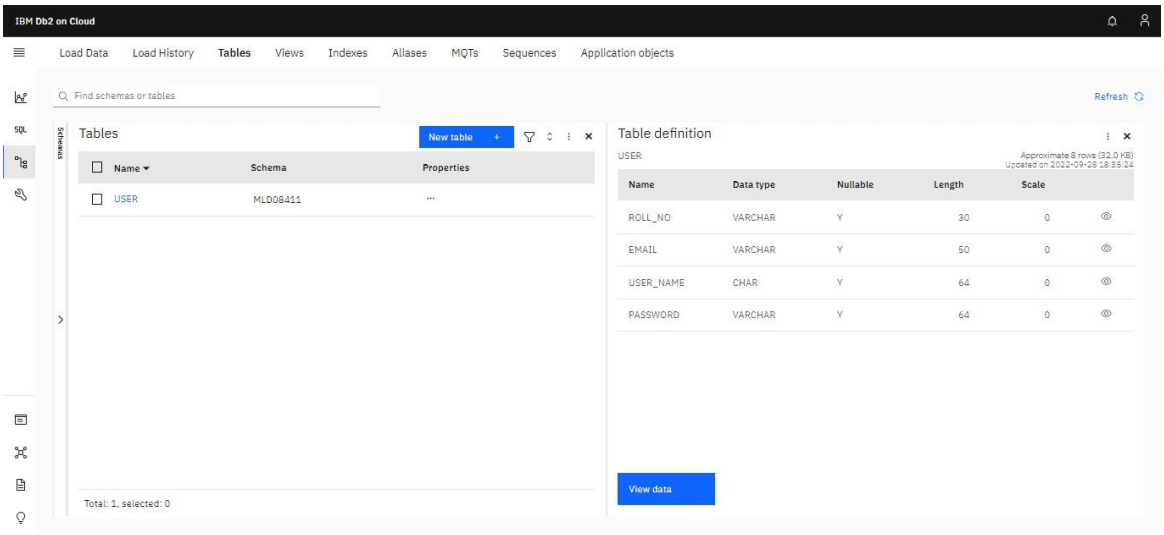
Program:

```
from flask import Flask
import ibm_db
app=Flask(__name__)
app.secret_key='a' try:
   conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=2f3279a5-73d1-4859-88f0a6c3e6b4b907.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:



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=2f3279a5-73d1-4859-88f0a6c3e6b4b907.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;PORT=30756;SECUTY=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':
        email=request.form['email']
        password=request.form['password']
        sql="SELECT * FROM user WHERE email=?"
```

```
 prep_stmt=ibm_db.prepare(conn,sql)
ibm_db.bind_param(prep_stmt,1,email)
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,2,email)
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':
 email=request.form['email']
  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!")
 returnredirect(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

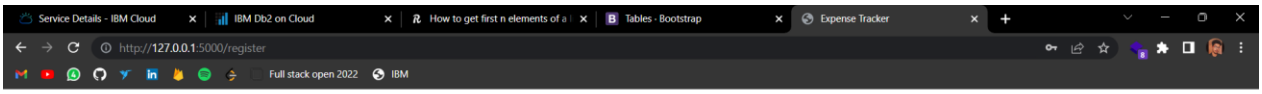
```
<html>
   <head>
      <title>Login page</title>
      <link rel="stylesheet" href="{{url_for('static', filename='style.css')}}">
   </head>
```

```
    <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:



REGISTER

Name

Sridharan S

Email

sri@gmail.com

Password

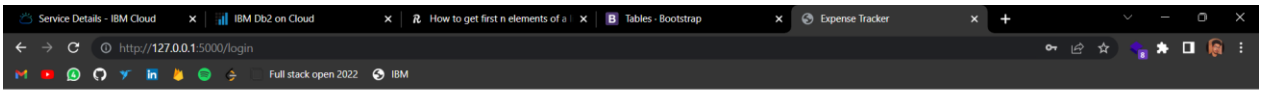
••••••

Confirm Password

••••••

Register

Already have an account? Login Here



LOGIN

Email Address

sri@gmail.com

Password

••••••

Login

Don't have an account? Register Here

Welcome ,Sridharan S !