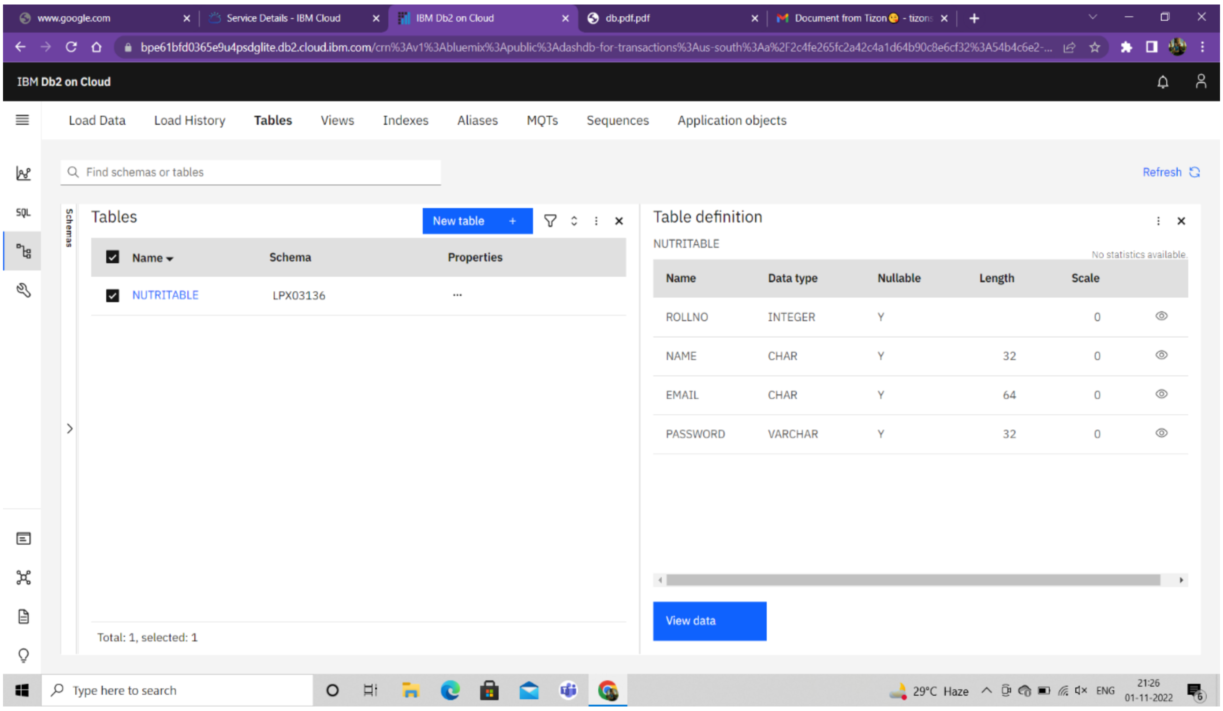
ASSIGNMENT – 2

Date	25 September 2022
Team ID	PNT2022TMID54057
Project Name	Nutrition Assistant Application
Maximum Marks	4 Marks

Questions:

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

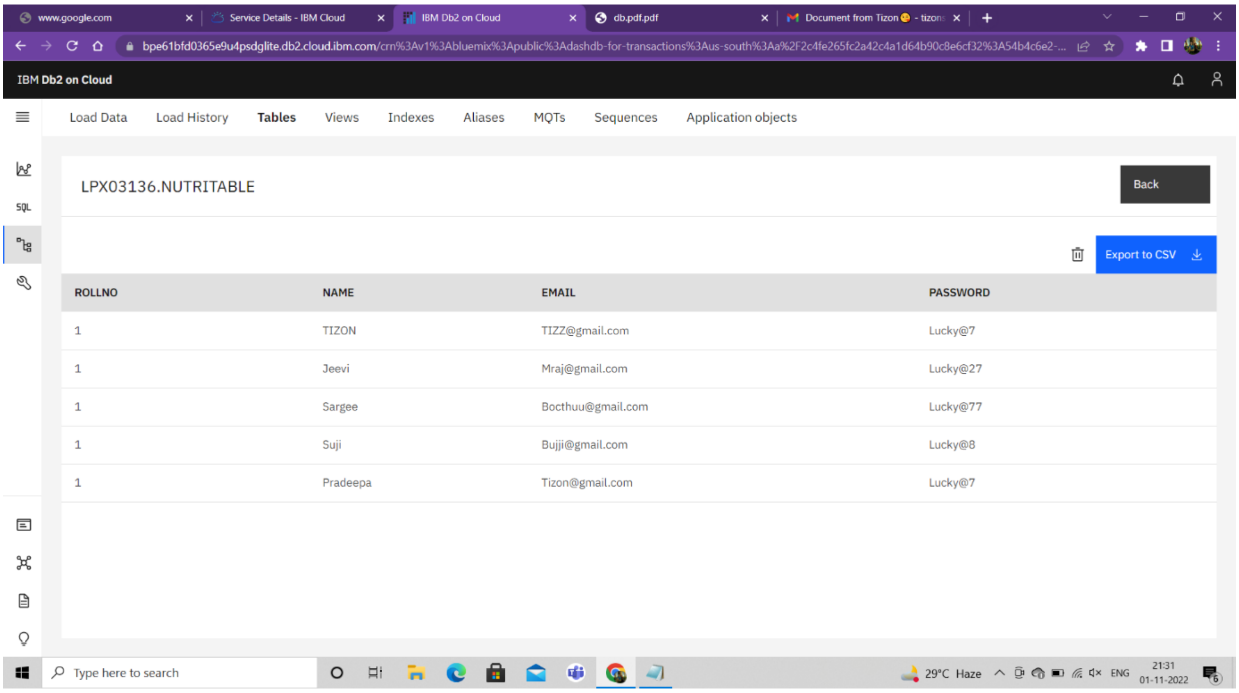


Name	Schema	Properties
NUTRITABLE	LPX03136	···

NUTRITABLE

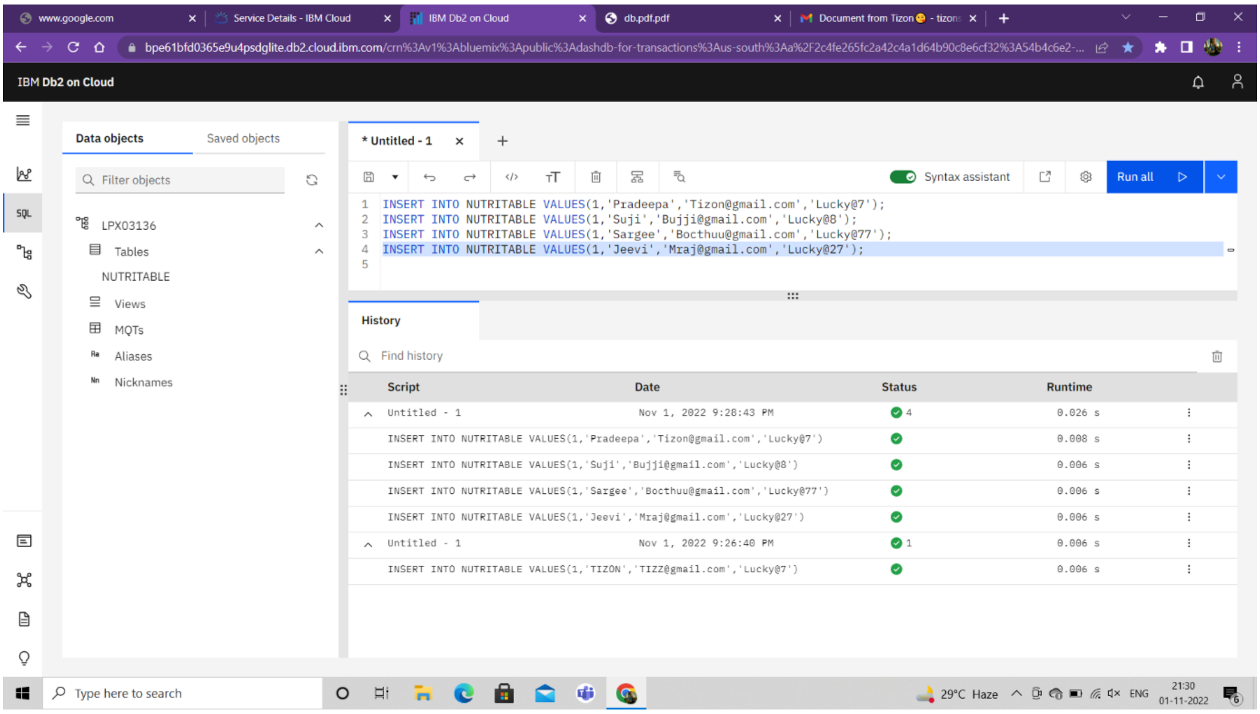
No statistics available.

Name	Data type	Nullable	Length	Scale
ROLLNO	INTEGER	Y		0
NAME	CHAR	Y	32	0
EMAIL	CHAR	Y	64	0
PASSWORD	VARCHAR	Y	32	0



LPX03136.NUTRITABLE

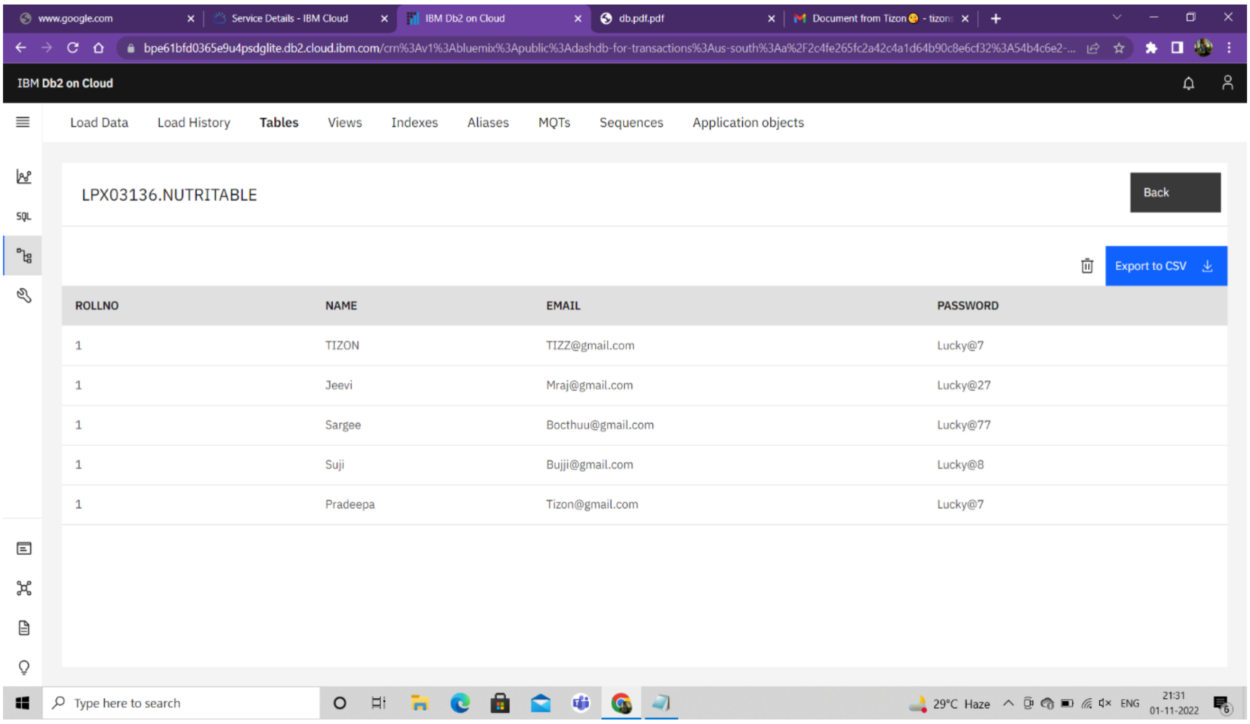
ROLLNO	NAME	EMAIL	PASSWORD
1	TIZON	TIZZ@gmail.com	Lucky@7
1	Jeevi	Mraj@gmail.com	Lucky@27
1	Sargee	Bocthuu@gmail.com	Lucky@77
1	Suji	Bujji@gmail.com	Lucky@8
1	Pradeepa	Tizon@gmail.com	Lucky@7



```
INSERT INTO NUTRITABLE VALUES(1,'Pradeepa','Tizon@gmail.com','Lucky@7');
INSERT INTO NUTRITABLE VALUES(1,'Suji','Bujji@gmail.com','Lucky@8');
INSERT INTO NUTRITABLE VALUES(1,'Sargee','Bocthuu@gmail.com','Lucky@77');
INSERT INTO NUTRITABLE VALUES(1,'Jeevi','Mraj@gmail.com','Lucky@27');
```

History

Script	Date	Status	Runtime
Untitled - 1	Nov 1, 2022 9:28:43 PM	✓ 4	0.026 s
INSERT INTO NUTRITABLE VALUES(1,'Pradeepa','Tizon@gmail.com','Lucky@7')		✓	0.008 s
INSERT INTO NUTRITABLE VALUES(1,'Suji','Bujji@gmail.com','Lucky@8')		✓	0.006 s
INSERT INTO NUTRITABLE VALUES(1,'Sargee','Bocthuu@gmail.com','Lucky@77')		✓	0.006 s
INSERT INTO NUTRITABLE VALUES(1,'Jeevi','Mraj@gmail.com','Lucky@27')		✓	0.006 s
Untitled - 1	Nov 1, 2022 9:26:40 PM	✓ 1	0.006 s
INSERT INTO NUTRITABLE VALUES(1,'TIZON','TIZZ@gmail.com','Lucky@7')		✓	0.006 s

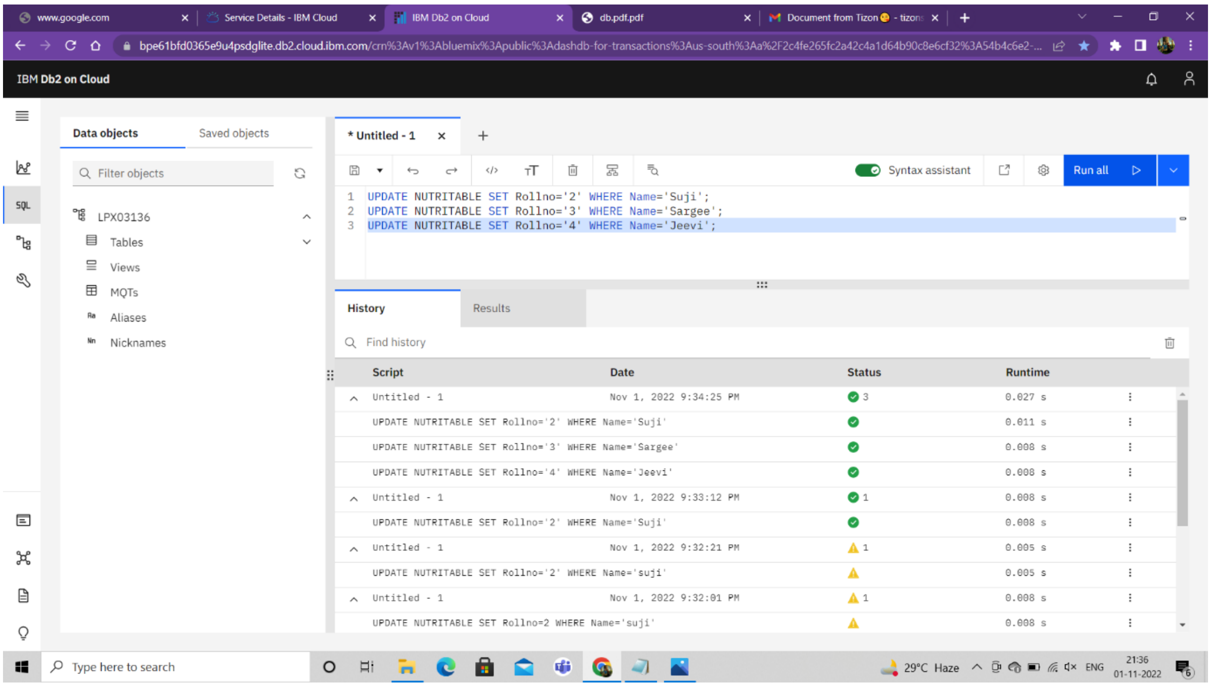


LPX03136.NUTRITABLE

ROLLNO	NAME	EMAIL	PASSWORD
1	TIZON	TIZZ@gmail.com	Lucky@7
1	Jeevi	Mraj@gmail.com	Lucky@27
1	Sargee	Bocthuu@gmail.com	Lucky@77
1	Suji	Bujji@gmail.com	Lucky@8
1	Pradeepa	Tizon@gmail.com	Lucky@7

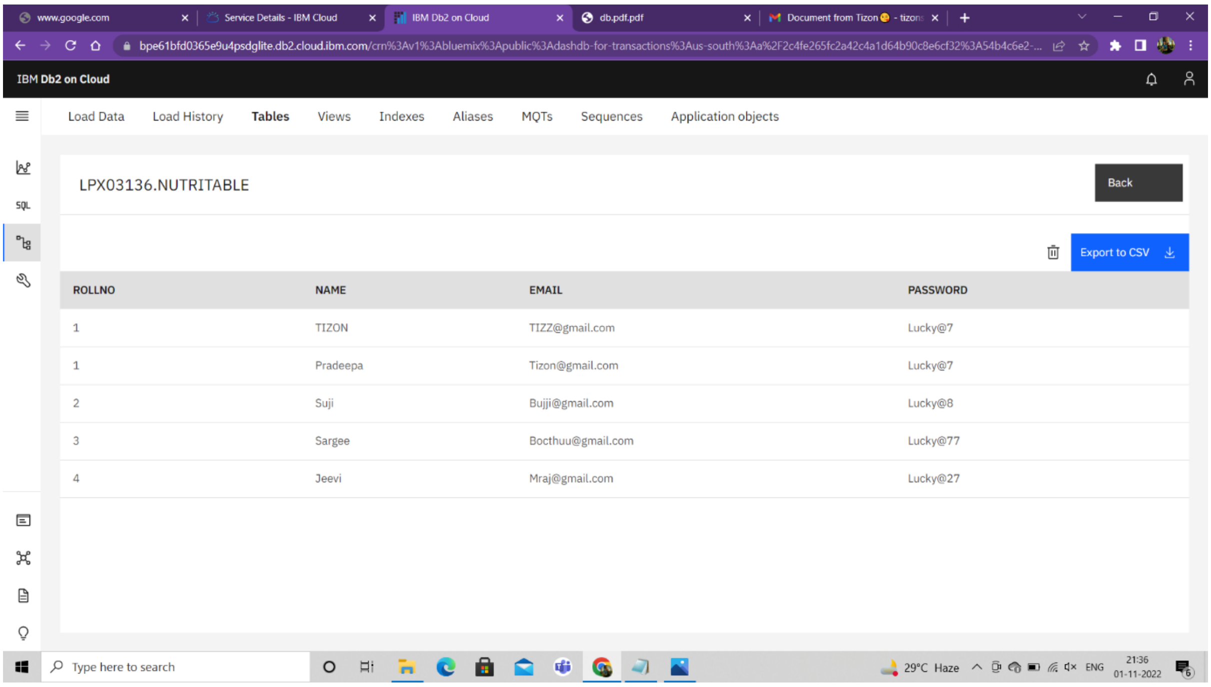
2. Perform UPDATE,DELETE Queries with user table

UPDATE :



```
UPDATE NUTRITABLE SET Rollno='2' WHERE Name='Suji';
UPDATE NUTRITABLE SET Rollno='3' WHERE Name='Sargee';
UPDATE NUTRITABLE SET Rollno='4' WHERE Name='Jeevi';
```

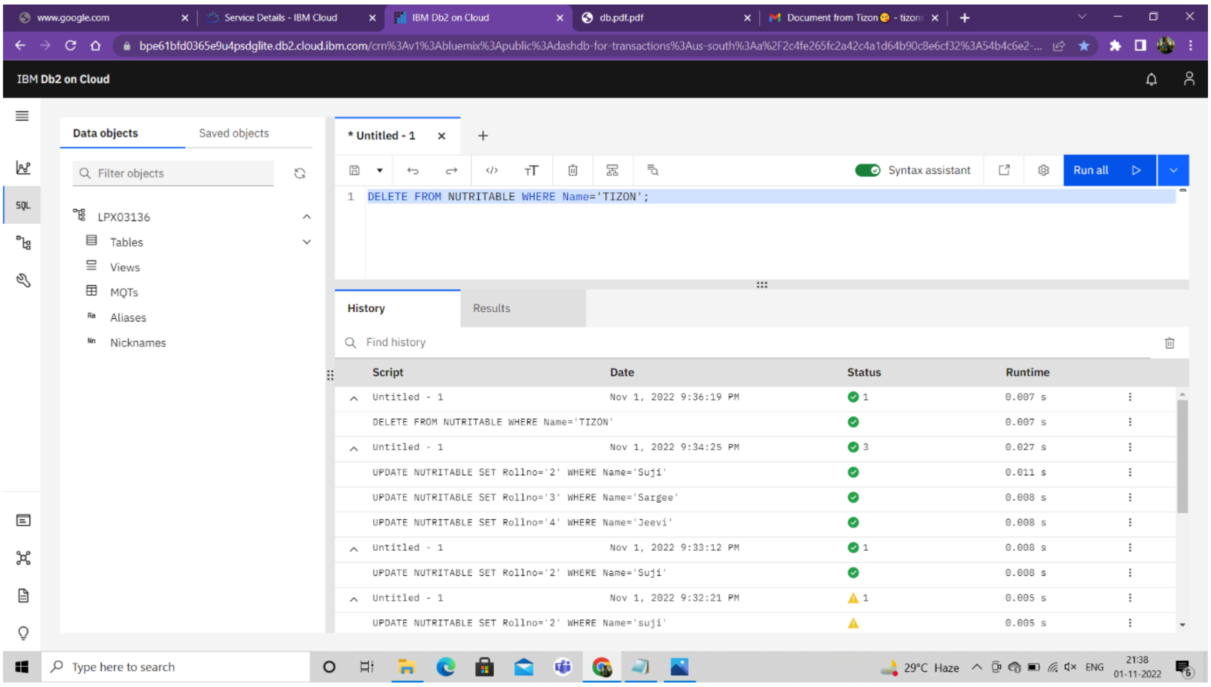
Script	Date	Status	Runtime
Untitled - 1	Nov 1, 2022 9:34:25 PM	✓ 3	0.027 s
UPDATE NUTRITABLE SET Rollno='2' WHERE Name='Suji'		✓	0.011 s
UPDATE NUTRITABLE SET Rollno='3' WHERE Name='Sargee'		✓	0.008 s
UPDATE NUTRITABLE SET Rollno='4' WHERE Name='Jeevi'		✓	0.008 s
Untitled - 1	Nov 1, 2022 9:33:12 PM	✓ 1	0.008 s
UPDATE NUTRITABLE SET Rollno='2' WHERE Name='Suji'		✓	0.008 s
Untitled - 1	Nov 1, 2022 9:32:21 PM	⚠ 1	0.005 s
UPDATE NUTRITABLE SET Rollno='2' WHERE Name='suji'		⚠	0.005 s
Untitled - 1	Nov 1, 2022 9:32:01 PM	⚠ 1	0.008 s
UPDATE NUTRITABLE SET Rollno=2 WHERE Name='suji'		⚠	0.008 s



LPX03136.NUTRITABLE

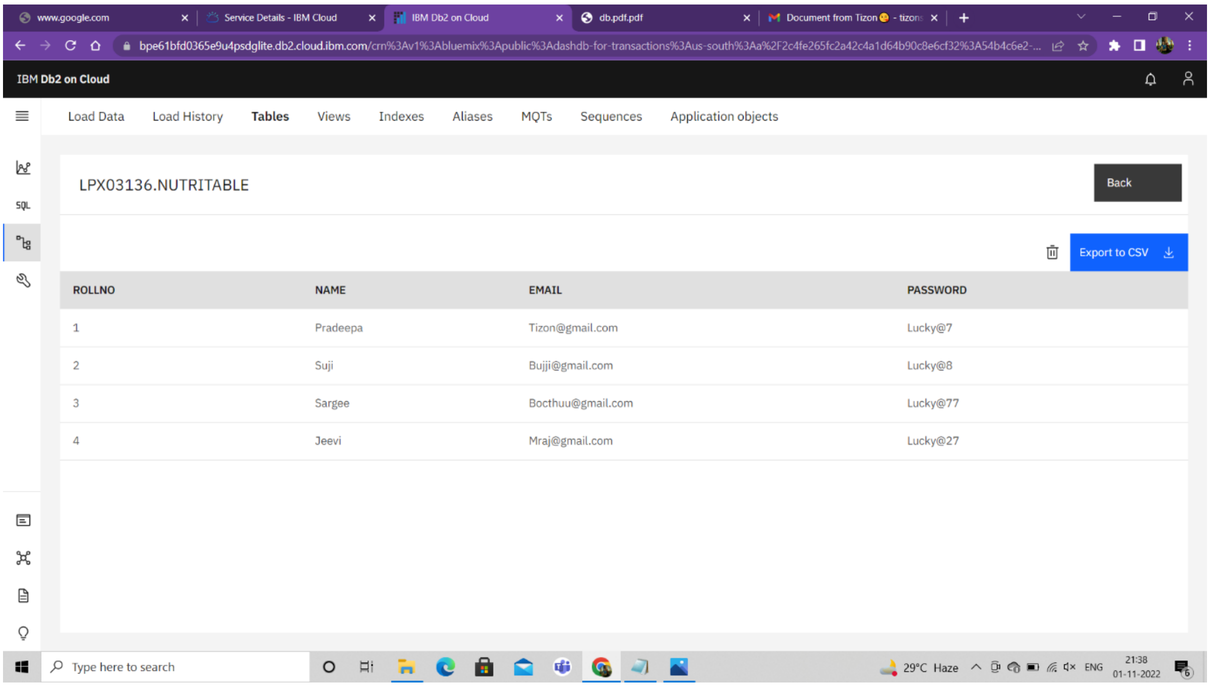
ROLLNO	NAME	EMAIL	PASSWORD
1	TIZON	TIZZ@gmail.com	Lucky@7
1	Pradeepa	Tizon@gmail.com	Lucky@7
2	Suji	Bujji@gmail.com	Lucky@8
3	Sargee	Bocthuu@gmail.com	Lucky@77
4	Jeevi	Mraj@gmail.com	Lucky@27

DELETE :



```
DELETE FROM NUTRITABLE WHERE Name='TIZON';
```

Script	Date	Status	Runtime
Untitled - 1	Nov 1, 2022 9:36:19 PM	✓ 1	0.007 s
DELETE FROM NUTRITABLE WHERE Name='TIZON'		✓	0.007 s
Untitled - 1	Nov 1, 2022 9:34:25 PM	✓ 3	0.027 s
UPDATE NUTRITABLE SET Rollno='2' WHERE Name='Suji'		✓	0.011 s
UPDATE NUTRITABLE SET Rollno='3' WHERE Name='Sargee'		✓	0.008 s
UPDATE NUTRITABLE SET Rollno='4' WHERE Name='Jeevi'		✓	0.008 s
Untitled - 1	Nov 1, 2022 9:33:12 PM	✓ 1	0.008 s
UPDATE NUTRITABLE SET Rollno='2' WHERE Name='Suji'		✓	0.008 s
Untitled - 1	Nov 1, 2022 9:32:21 PM	⚠ 1	0.005 s
UPDATE NUTRITABLE SET Rollno='2' WHERE Name='suji'		⚠	0.005 s

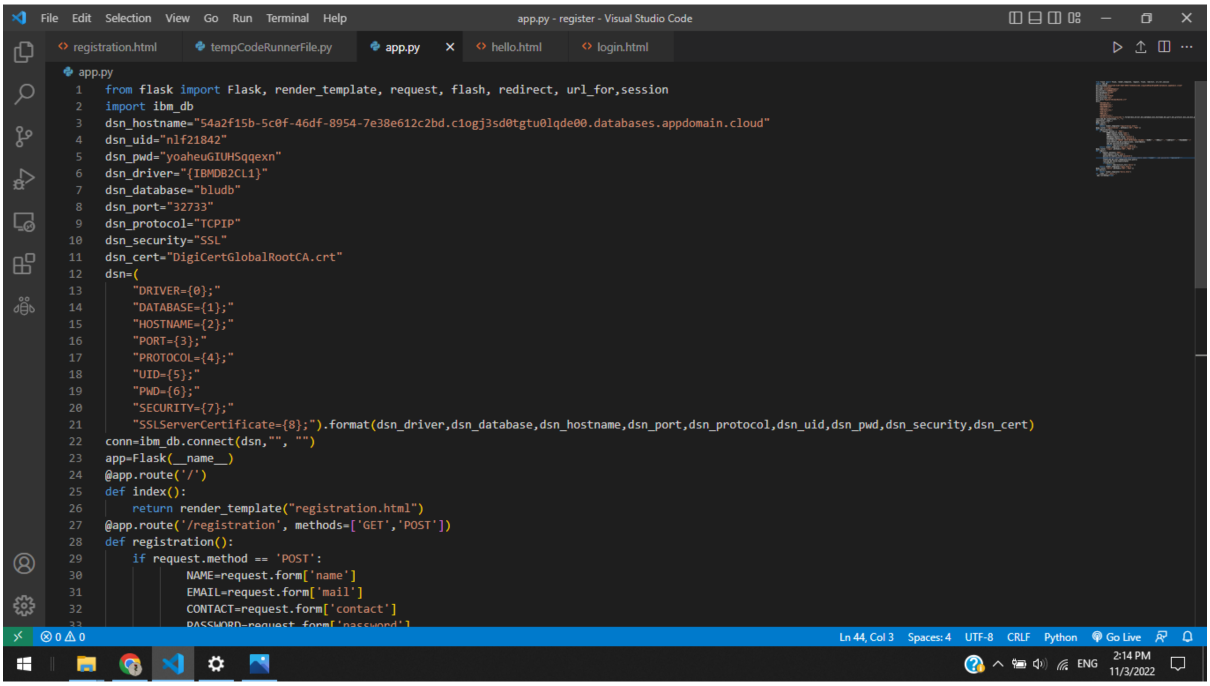


LPX03136.NUTRITABLE

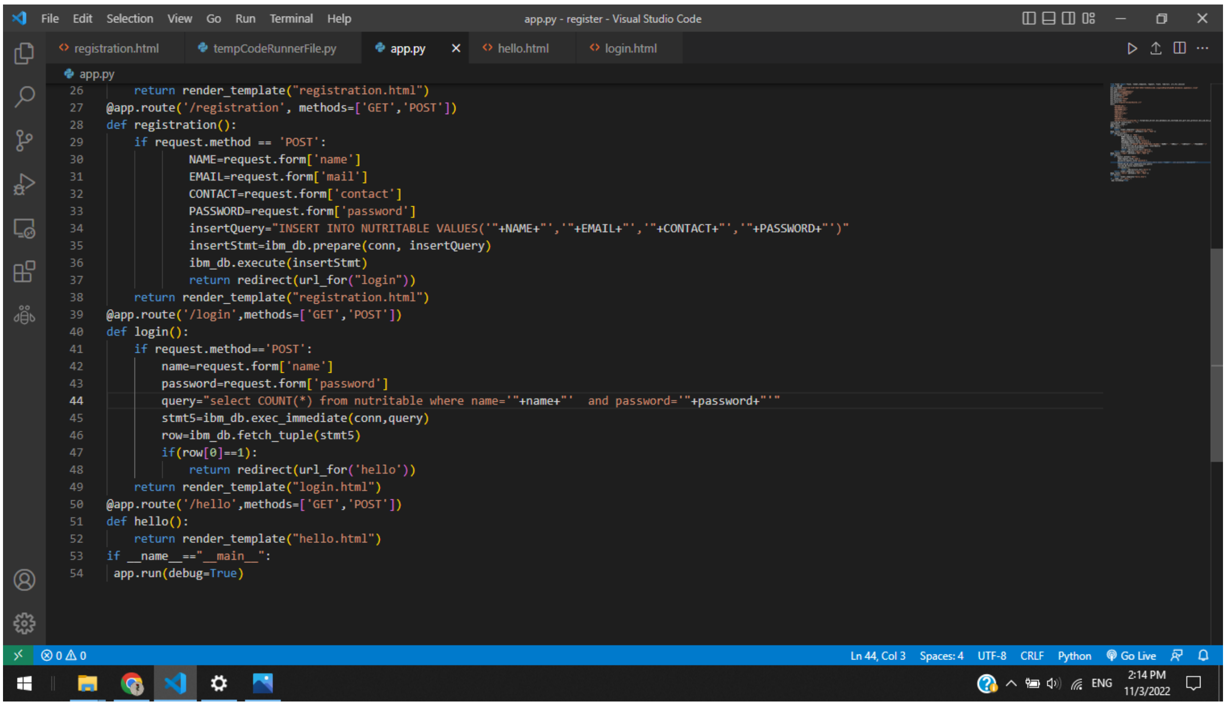
ROLLNO	NAME	EMAIL	PASSWORD
1	Pradeepa	Tizon@gmail.com	Lucky@7
2	Suji	Bujji@gmail.com	Lucky@8
3	Sargee	Bocthuu@gmail.com	Lucky@77
4	Jeevi	Mraj@gmail.com	Lucky@27

3. Connect python code to db2.
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 :

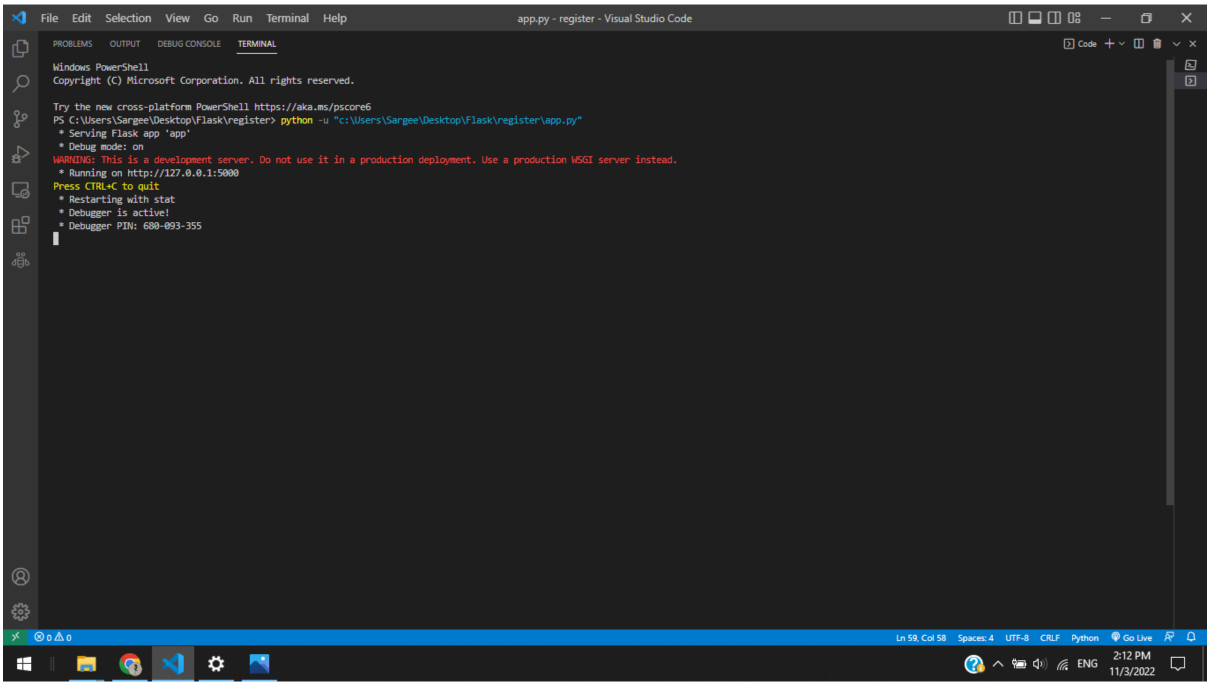


```python
from flask import Flask, render_template, request, flash, redirect, url_for,session
import ibm_db
dsn_hostname="54a2f15b-5c0f-46df-8954-7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud"
dsn_uid="nlf21842"
dsn_pwd="yoaheuGIUHSqqexn"
dsn_driver="{IBMDB2CL1}"
dsn_database="bludb"
dsn_port="32733"
dsn_protocol="TCPIP"
dsn_security="SSL"
dsn_cert="DigiCertGlobalRootCA.crt"
dsn=(
    "DRIVER={0};"
    "DATABASE={1};"
    "HOSTNAME={2};"
    "PORT={3};"
    "PROTOCOL={4};"
    "UID={5};"
    "PWD={6};"
    "SECURITY={7};"
    "SSLServerCertificate={8};").format(dsn_driver,dsn_database,dsn_hostname,dsn_port,dsn_protocol,dsn_uid,dsn_pwd,dsn_security,dsn_cert)
conn=ibm_db.connect(dsn,"", "")
app=Flask(__name__)
@app.route('/')
def index():
    return render_template("registration.html")
@app.route('/registration', methods=['GET','POST'])
def registration():
    if request.method == 'POST':
            NAME=request.form['name']
            EMAIL=request.form['mail']
            CONTACT=request.form['contact']
            PASSWORD=request.form['password']
```



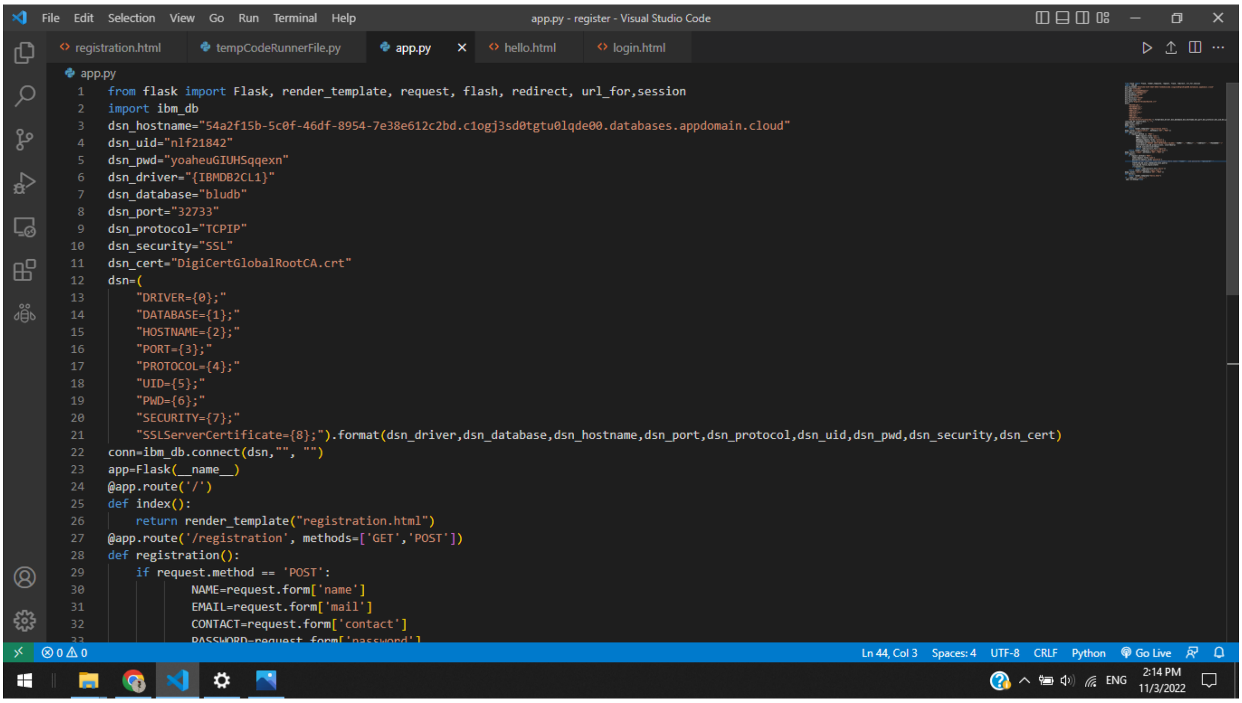
```python
    return render_template("registration.html")
@app.route('/registration', methods=['GET','POST'])
def registration():
    if request.method == 'POST':
            NAME=request.form['name']
            EMAIL=request.form['mail']
            CONTACT=request.form['contact']
            PASSWORD=request.form['password']
            insertQuery="INSERT INTO NUTRITABLE VALUES('"+NAME+"','"+EMAIL+"','"+CONTACT+"','"+PASSWORD+"')"
            insertStmt=ibm_db.prepare(conn, insertQuery)
            ibm_db.execute(insertStmt)
            return redirect(url_for("login"))
    return render_template("registration.html")
@app.route('/login',methods=['GET','POST'])
def login():
    if request.method=='POST':
        name=request.form['name']
        password=request.form['password']
        query="select COUNT(*) from nutritable where name='"+name+"'  and password='"+password+"'"
        stmt5=ibm_db.exec_immediate(conn,query)
        row=ibm_db.fetch_tuple(stmt5)
        if(row[0]==1):
            return redirect(url_for('hello'))
    return render_template("login.html")
@app.route('/hello',methods=['GET','POST'])
def hello():
    return render_template("hello.html")
if __name__=="__main__":
 app.run(debug=True)
```

Terminal :

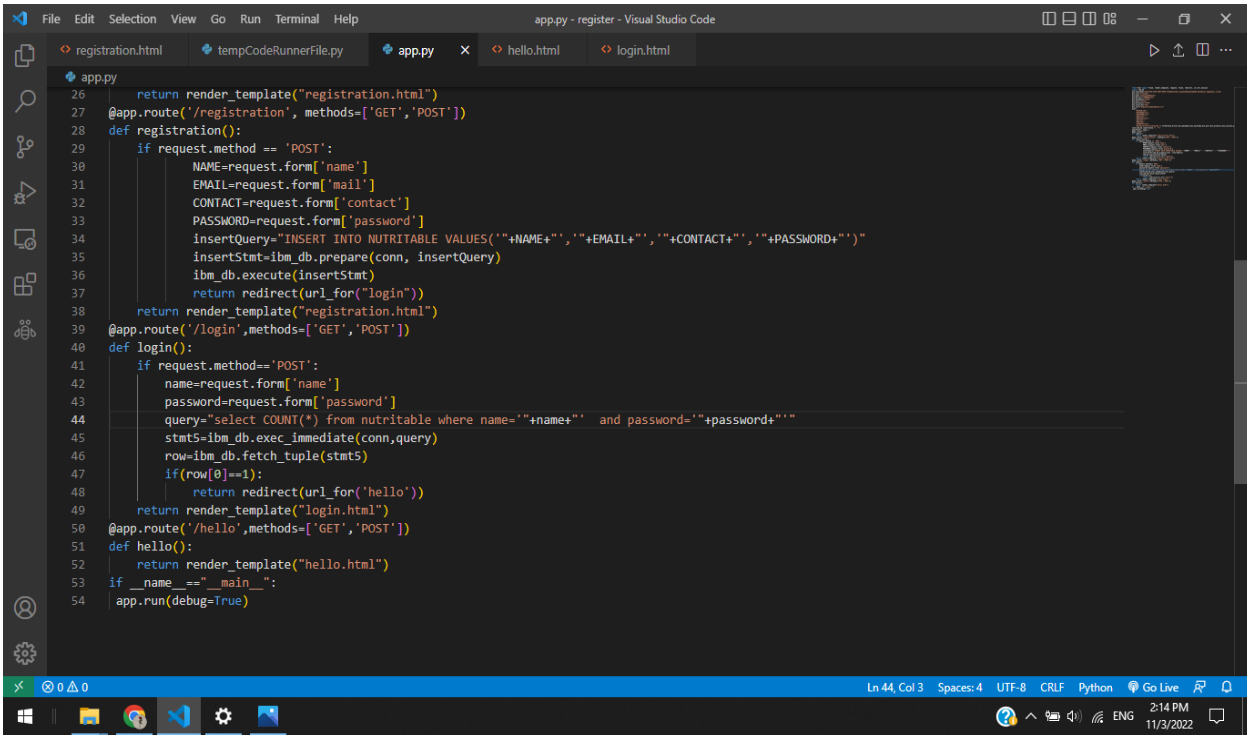


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\Sargee\Desktop\Flask\register> python -u "c:\Users\Sargee\Desktop\Flask\register\app.py"
 * Serving Flask app 'app'
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 680-093-355
```

Registration.html :

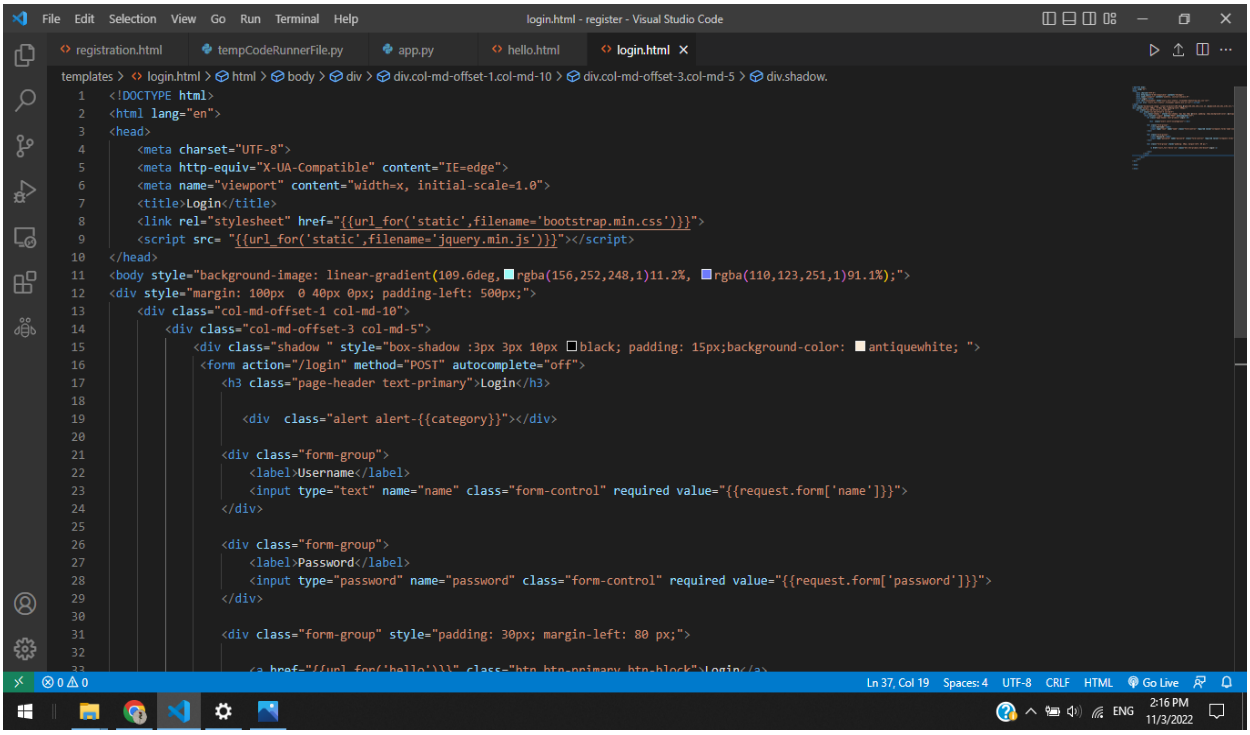


```python
from flask import Flask, render_template, request, flash, redirect, url_for,session
import ibm_db
dsn_hostname="54a2f15b-5c0f-46df-8954-7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud"
dsn_uid="nlf21842"
dsn_pwd="yoaheuGIUHSqqexn"
dsn_driver="{IBMDB2CL1}"
dsn_database="bludb"
dsn_port="32733"
dsn_protocol="TCPIP"
dsn_security="SSL"
dsn_cert="DigiCertGlobalRootCA.crt"
dsn=(
    "DRIVER={0};"
    "DATABASE={1};"
    "HOSTNAME={2};"
    "PORT={3};"
    "PROTOCOL={4};"
    "UID={5};"
    "PWD={6};"
    "SECURITY={7};"
    "SSLServerCertificate={8};").format(dsn_driver,dsn_database,dsn_hostname,dsn_port,dsn_protocol,dsn_uid,dsn_pwd,dsn_security,dsn_cert)
conn=ibm_db.connect(dsn,"", "")
app=Flask(__name__)
@app.route('/')
def index():
    return render_template("registration.html")
@app.route('/registration', methods=['GET','POST'])
def registration():
    if request.method == 'POST':
            NAME=request.form['name']
            EMAIL=request.form['mail']
            CONTACT=request.form['contact']
            PASSWORD=request.form['password']
```

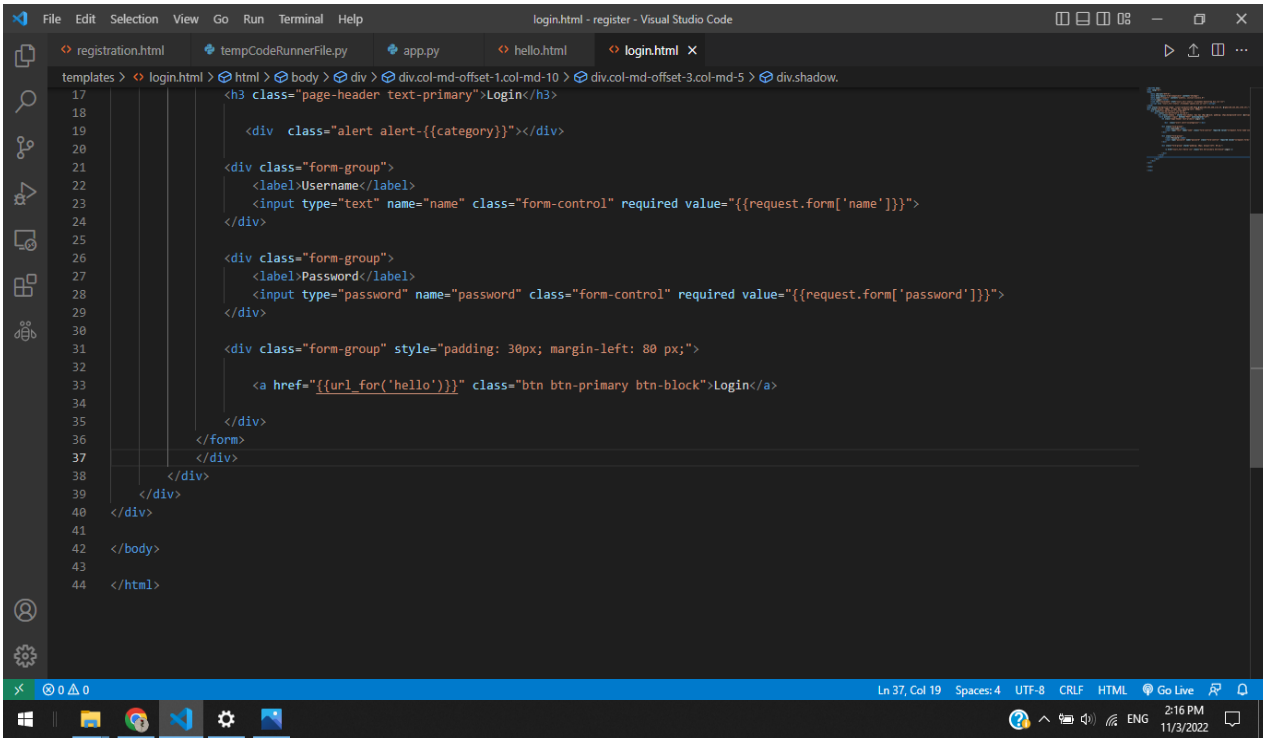


```python
    return render_template("registration.html")
@app.route('/registration', methods=['GET','POST'])
def registration():
    if request.method == 'POST':
            NAME=request.form['name']
            EMAIL=request.form['mail']
            CONTACT=request.form['contact']
            PASSWORD=request.form['password']
            insertQuery="INSERT INTO NUTRITABLE VALUES('"+NAME+"','"+EMAIL+"','"+CONTACT+"','"+PASSWORD+"')"
            insertStmt=ibm_db.prepare(conn, insertQuery)
            ibm_db.execute(insertStmt)
            return redirect(url_for("login"))
    return render_template("registration.html")
@app.route('/login',methods=['GET','POST'])
def login():
    if request.method=='POST':
        name=request.form['name']
        password=request.form['password']
        query="select COUNT(*) from nutritable where name='"+name+"'  and password='"+password+"'"
        stmt5=ibm_db.exec_immediate(conn,query)
        row=ibm_db.fetch_tuple(stmt5)
        if(row[0]==1):
            return redirect(url_for('hello'))
    return render_template("login.html")
@app.route('/hello',methods=['GET','POST'])
def hello():
    return render_template("hello.html")
if __name__=="__main__":
 app.run(debug=True)
```

Login.html:



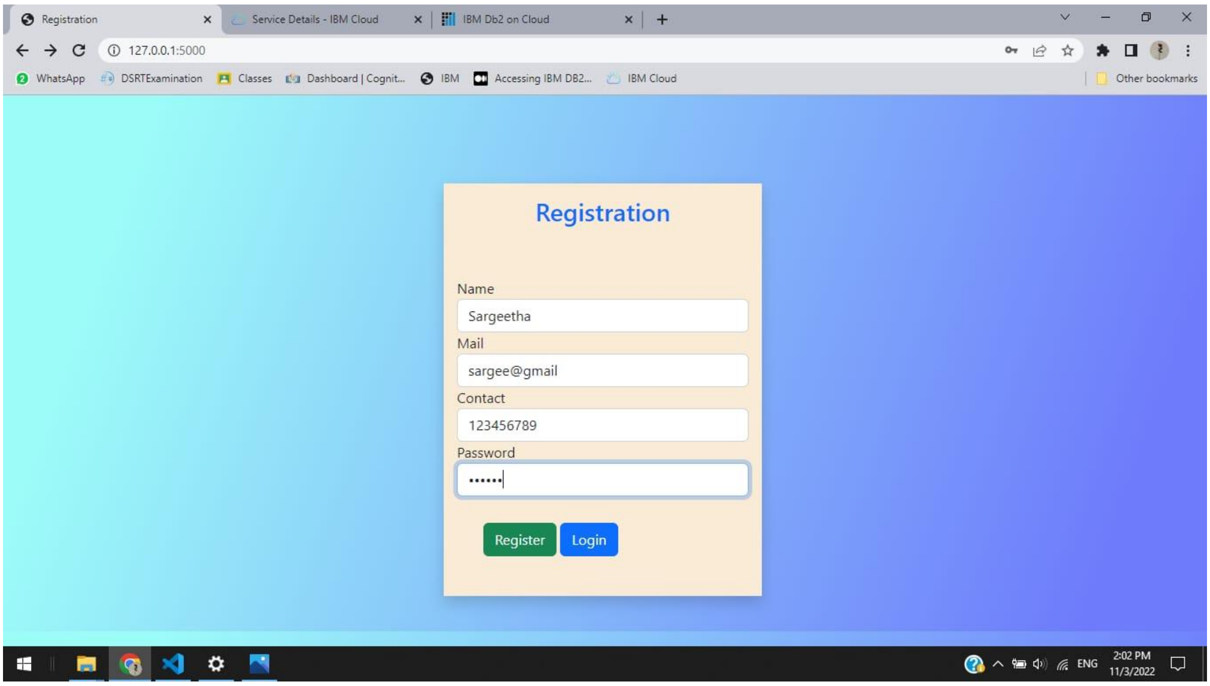
```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=x, initial-scale=1.0">
    <title>Login</title>
    <link rel="stylesheet" href="{{url_for('static',filename='bootstrap.min.css')}}">
    <script src= "{{url_for('static',filename='jquery.min.js')}}"></script>
</head>
<body style="background-image: linear-gradient(109.6deg, rgba(156,252,248,1)11.2%,  rgba(110,123,251,1)91.1%);">
<div style="margin: 100px  0 40px 0px; padding-left: 500px;">
    <div class="col-md-offset-1 col-md-10">
        <div class="col-md-offset-3 col-md-5">
            <div class="shadow " style="box-shadow :3px 3px 10px black; padding: 15px;background-color:  antiquewhite; ">
             <form action="/login" method="POST" autocomplete="off">
                <h3 class="page-header text-primary">Login</h3>

                   <div  class="alert alert-{{category}}"></div>

                <div class="form-group">
                    <label>Username</label>
                    <input type="text" name="name" class="form-control" required value="{{request.form['name']}}">
                </div>

                <div class="form-group">
                    <label>Password</label>
                    <input type="password" name="password" class="form-control" required value="{{request.form['password']}}">
                </div>

                <div class="form-group" style="padding: 30px; margin-left: 80 px;">

```



```html
                <h3 class="page-header text-primary">Login</h3>

                   <div  class="alert alert-{{category}}"></div>

                <div class="form-group">
                    <label>Username</label>
                    <input type="text" name="name" class="form-control" required value="{{request.form['name']}}">
                </div>

                <div class="form-group">
                    <label>Password</label>
                    <input type="password" name="password" class="form-control" required value="{{request.form['password']}}">
                </div>

                <div class="form-group" style="padding: 30px; margin-left: 80 px;">

                    <a href="{{url_for('hello')}}" class="btn btn-primary btn-block">Login</a>

                </div>
            </form>
            </div>
        </div>
    </div>
</div>

</body>

</html>
```

OUTPUT :

REGISTERING :



Registration

Name

Sargeetha

Mail

sargee@gmail

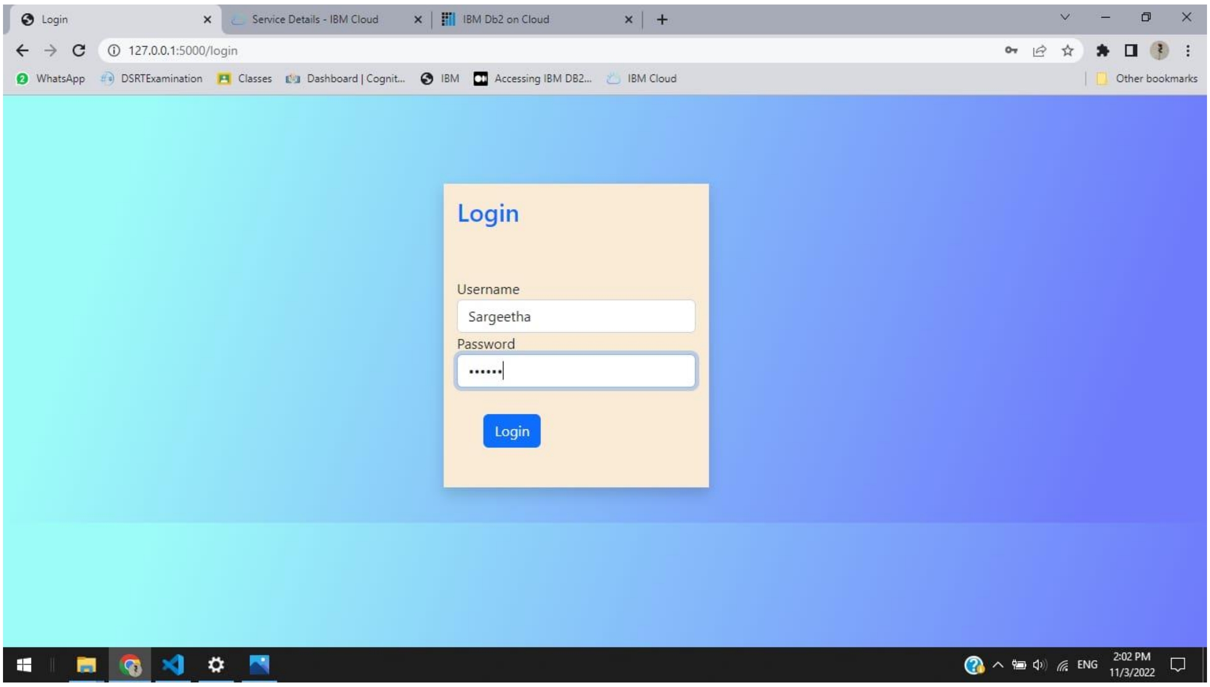
Contact

123456789

Password

••••••

LOGIN :



Login

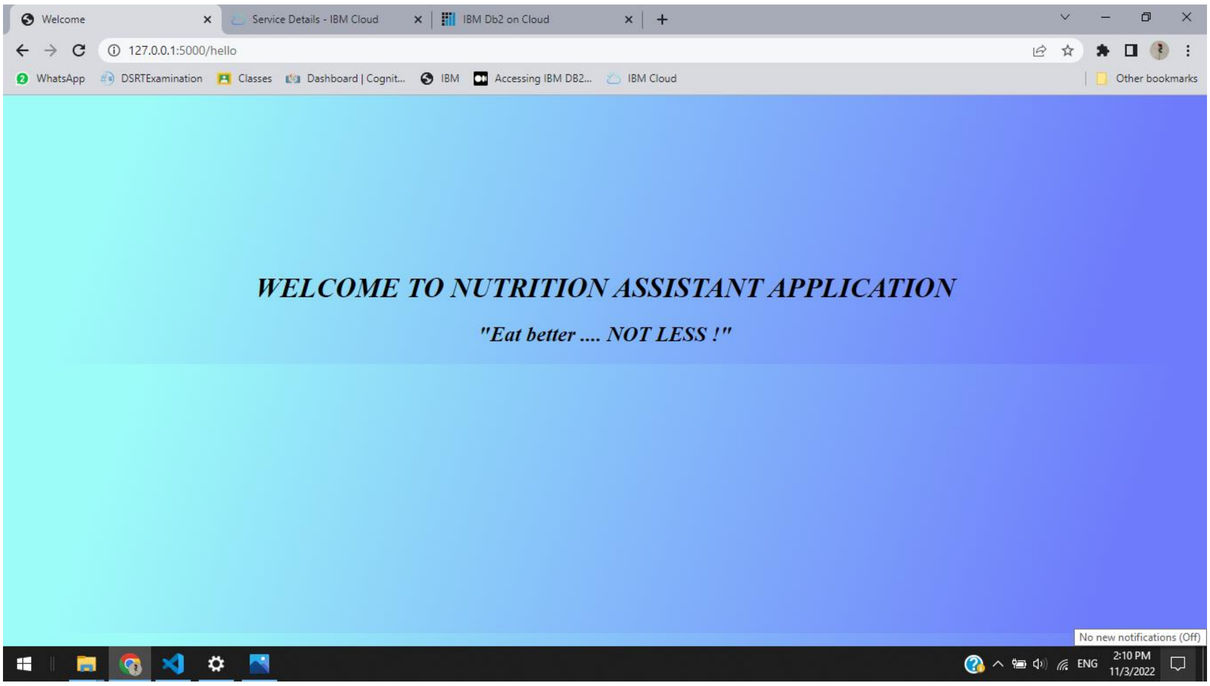
Username

Sargeetha

Password

••••••

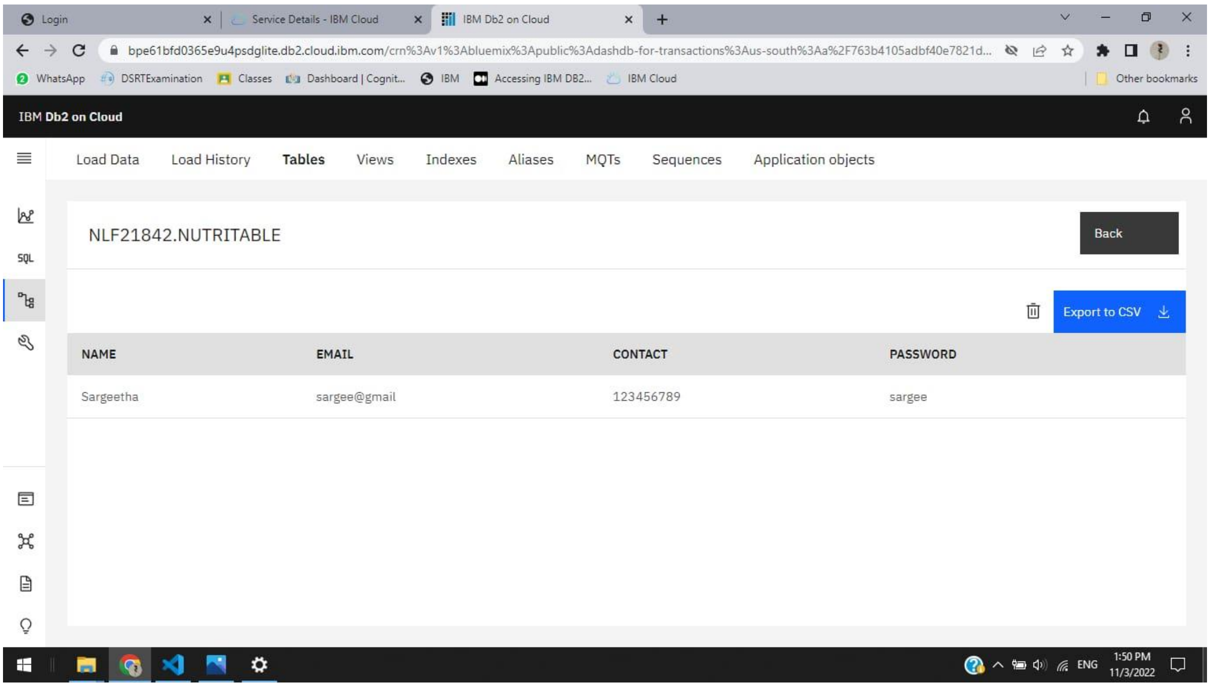
WelcomePage.html :



WELCOME TO NUTRITION ASSISTANT APPLICATION

"Eat better .... NOT LESS !"

DATABASE :



NLF21842.NUTRITABLE

NAME	EMAIL	CONTACT	PASSWORD
Sargeetha	sargee@gmail	123456789	sargee