

IMPLEMENTING WEB APPLICATION

CREATE IBM DB2 AND CONNECT WITH PYTHON

Date	24 September 2022
Team ID	PNT2022TMID54057
Project Name	Nutrition Assistant Application
Maximum Marks	2 Marks

Table created in ibm_db2 :

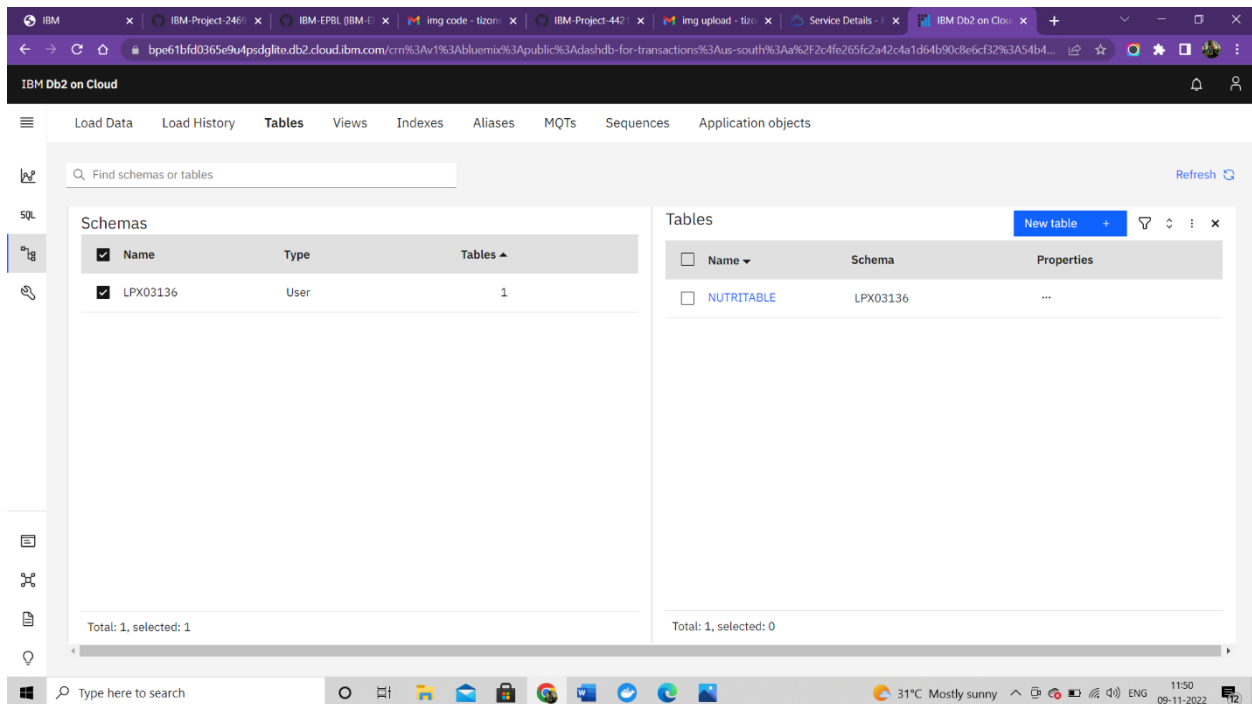


Table attributes :

The screenshot shows the IBM Db2 on Cloud console interface. The 'Tables' tab is selected, displaying a list of tables. The table 'NUTRITABLE' is highlighted, and its definition is shown in the 'Table definition' panel on the right.

Table definition: NUTRITABLE

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

The console also shows a search bar for schemas or tables, a 'New table' button, and a 'View data' button. The bottom status bar indicates the system temperature is 31°C and mostly sunny.

IBM DB2 created And Connected With Python

The screenshot shows a Python application in Visual Studio Code. The application is named 'app.py' and is running in a terminal. The code defines a Flask application with two routes: 'registration' and 'login'.

```
app.py
26 return render_template("registration.html")
27 @app.route('/registration', methods=['GET', 'POST'])
28 def registration():
29     if request.method == 'POST':
30         NAME=request.form['name']
31         EMAIL=request.form['mail']
32         CONTACT=request.form['contact']
33         PASSWORD=request.form['password']
34         insertQuery="INSERT INTO NUTRITABLE VALUES('"+NAME+"','"+EMAIL+"','"+CONTACT+"','"+PASSWORD+"')"
35         insertStmt=ibm_db.prepare(conn, insertQuery)
36         ibm_db.execute(insertStmt)
37         return redirect(url_for("login"))
38     return render_template("registration.html")
39 @app.route('/login', methods=['GET', 'POST'])
40 def login():
41     if request.method == 'POST':
42         name=request.form['name']
43         password=request.form['password']
44         query="select COUNT(*) from nutritable where name='"+name+"' and password='"+password+"'"
45         stmt5=ibm_db.exec_immediate(conn, query)
46         row=ibm_db.fetch_tuple(stmt5)
47         if (row[0]==1):
48             return redirect(url_for("hello"))
49     return render_template("login.html")
50 @app.route('/hello', methods=['GET', 'POST'])
51 def hello():
52     return render_template("hello.html")
53 if __name__ == '__main__':
54     app.run(debug=True)
```

The application is running in a terminal window, and the status bar at the bottom shows the file is at line 44, column 3, with 4 spaces, UTF-8 encoding, CRLF line endings, and Python syntax highlighting.

```
File Edit Selection View Go Run Terminal Help
app.py - register - Visual Studio Code
registration.html tempCodeRunnerFile.py app.py hello.html login.html

app.py
1 from flask import Flask, render_template, request, flash, redirect, url_for, session
2 import sqlite3
3 dsn_hostname="54a2f15b-5c8f-48df-8954-7e38e612c2b6.clogj3s0mgtsu8kqle0n.databases.appdomain.cloud"
4 dsn_uid="n1f21843"
5 dsn_pwd="youhu63IR6qgeen"
6 dsn_driver="{DSNVER11}"
7 dsn_database="bluab"
8 dsn_port="12711"
9 dsn_protocol="TCP/IP"
10 dsn_security="SSL"
11 dsn_cert="DigicertGlobalRootCA.crt"
12 dsn={
13     "DRIVER={0}:",
14     "DATABASE={1}:",
15     "HOSTNAME={2}:",
16     "PORT={3}:",
17     "PROTOCOL={4}:",
18     "UID={5}:",
19     "PWD={6}:",
20     "SECURITY={7}:"
21 }
22 "DSN=ServerCertificate={8}:".format(dsn_driver,dsn_database,dsn_hostname,dsn_port,dsn_protocol,dsn_uid,dsn_pwd,dsn_security,dsn_cert)
23 conn=dsn_db.connect(dsn,"")
24 app=Flask(__name__)
25 @app.route("/")
26 def index():
27     return render_template("registration.html")
28 @app.route("/registration", methods=['GET','POST'])
29 def registration():
30     if request.method == 'POST':
31         NAME=request.form['name']
32         EMAIL=request.form['mail']
33         CONTACT=request.form['contact']
34         PASSWORD=request.form['password']
```

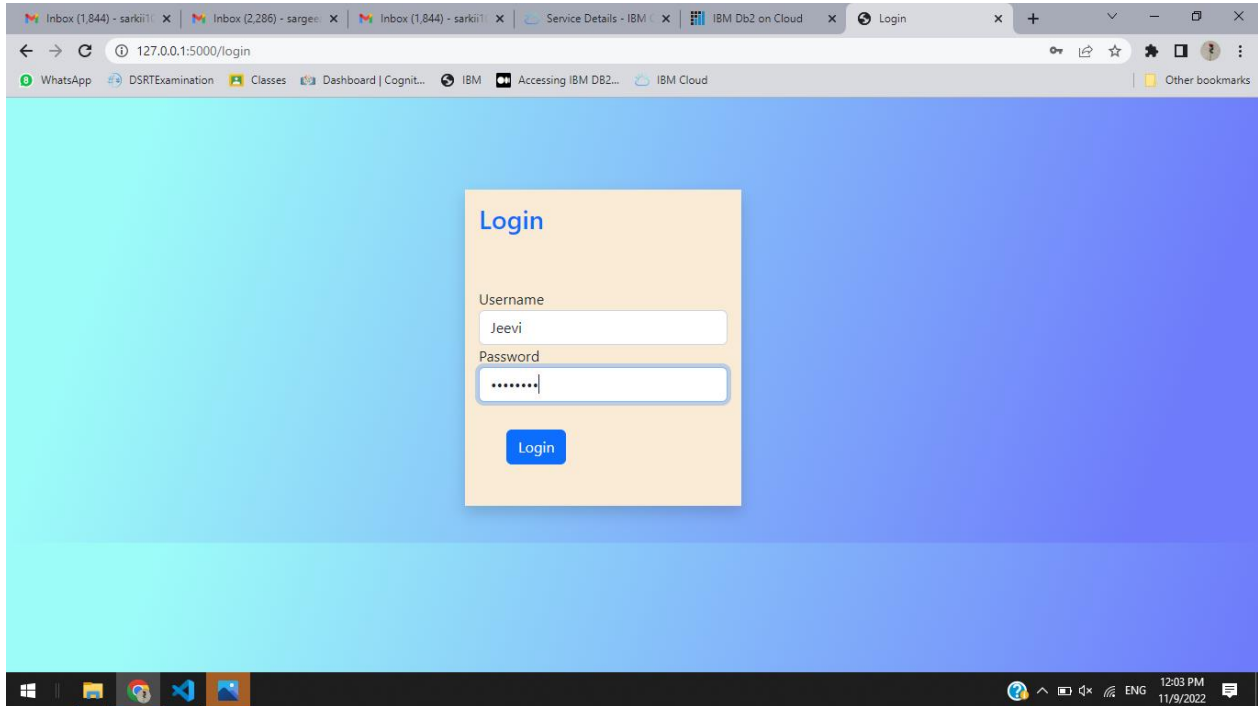
Registering :

The screenshot shows a web browser window with the address bar displaying '127.0.0.1:5000'. The page features a 'Registration' form with the following fields and values:

- Name: Jeevi
- Mail: Mrj@gmail.com
- Contact: 8472821098
- Password: (masked with dots)

At the bottom of the form, there are two buttons: 'Register' (green) and 'Login' (blue). The browser's taskbar at the bottom shows the time as 12:03 PM on 11/9/2022.

Log in :



Database :

IBM Db2 on Cloud

Load Data Load History **Tables** Views Indexes Aliases MQTs Sequences Application objects

LPX03136.NUTRITABLE

Export to CSV

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

31°C Mostly sunny 11:51 09-11-2022