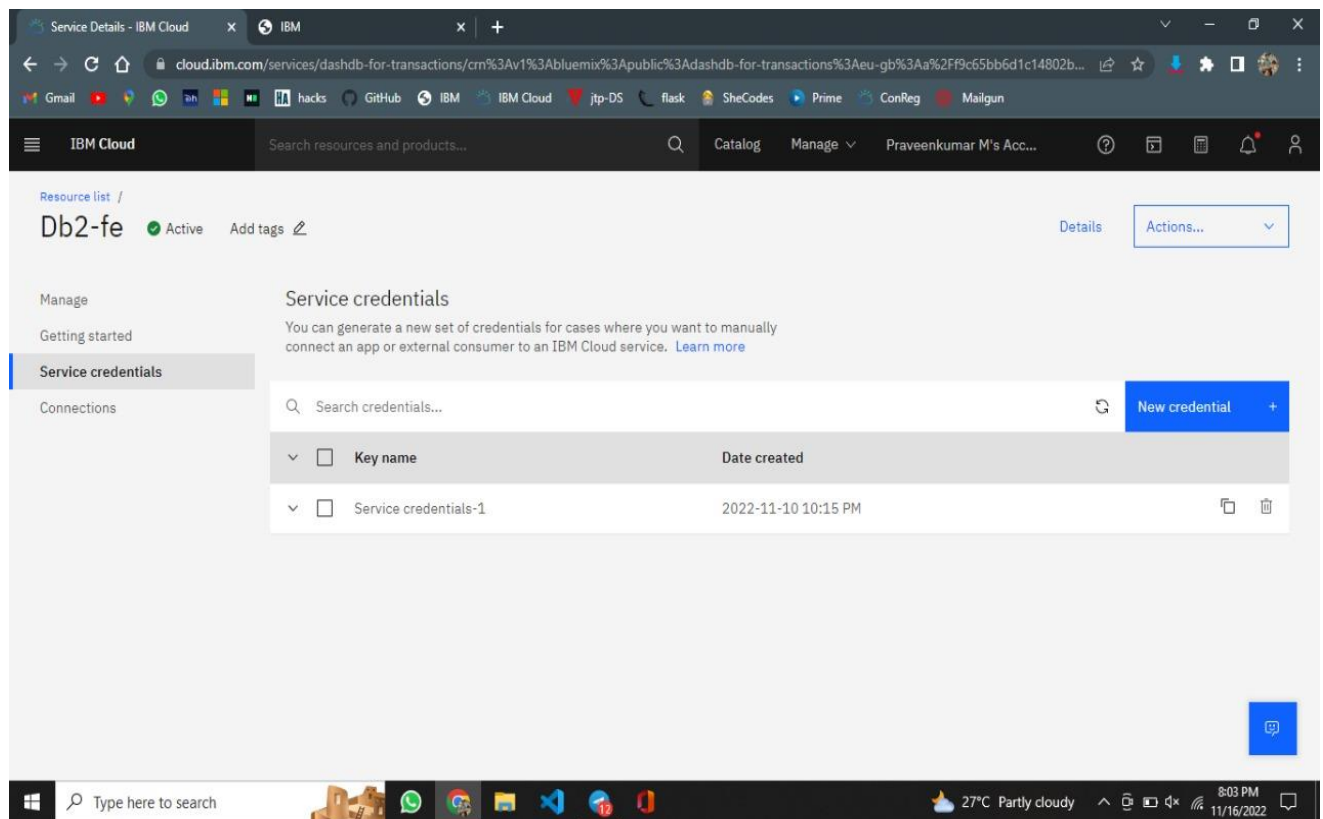


## Implementing Web Application

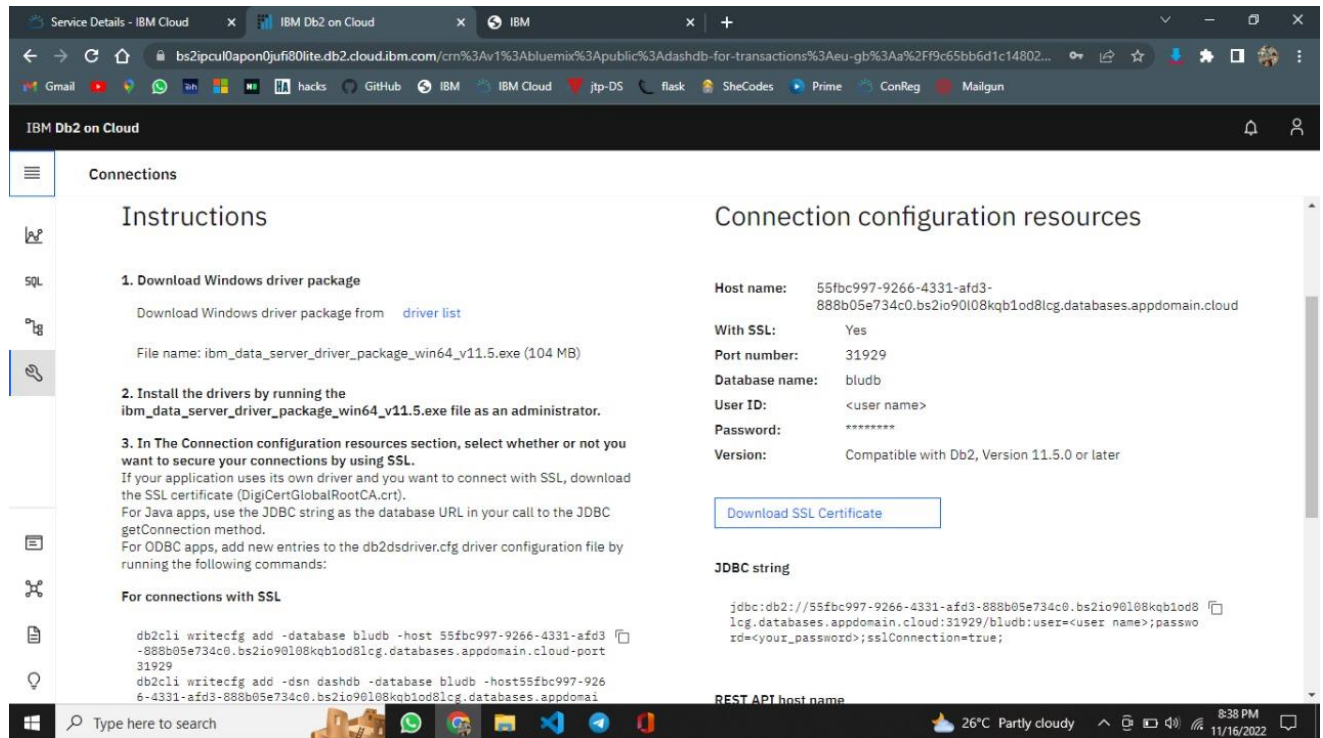
### Creating IBM-Db2 and Connect With Python

Date	27 October 2022
Team ID	PNT2022TMID29826
Project Name	CUSTOMER CARE REGISTRY

#### Step 1. Creating IBM data base



## Step 2. Downloading SSC certificate for connection



**Connections**

### Instructions

- Download Windows driver package**  
Download Windows driver package from [driver list](#)  
  
File name: `ibm_data_server_driver_package_win64_v11.5.exe` (104 MB)
- Install the drivers by running the `ibm_data_server_driver_package_win64_v11.5.exe` file as an administrator.**
- In The Connection configuration resources section, select whether or not you want to secure your connections by using SSL.**  
If your application uses its own driver and you want to connect with SSL, download the SSL certificate (DigicertGlobalRootCA.crt).  
For Java apps, use the JDBC string as the database URL in your call to the JDBC `getConnection` method.  
For ODBC apps, add new entries to the `db2dsdriver.cfg` driver configuration file by running the following commands:  
  
**For connections with SSL**  

```
db2cli writecfg add -database bludb -host 55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8lclg.databases.appdomain.cloud -port 31929  
db2cli writecfg add -dsn dashdb -database bludb -host 55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8lclg.databases.appdomain.cloud
```

### Connection configuration resources

**Host name:** 55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8lclg.databases.appdomain.cloud

**With SSL:** Yes

**Port number:** 31929

**Database name:** bludb

**User ID:** <user name>

**Password:** \*\*\*\*\*

**Version:** Compatible with Db2, Version 11.5.0 or later

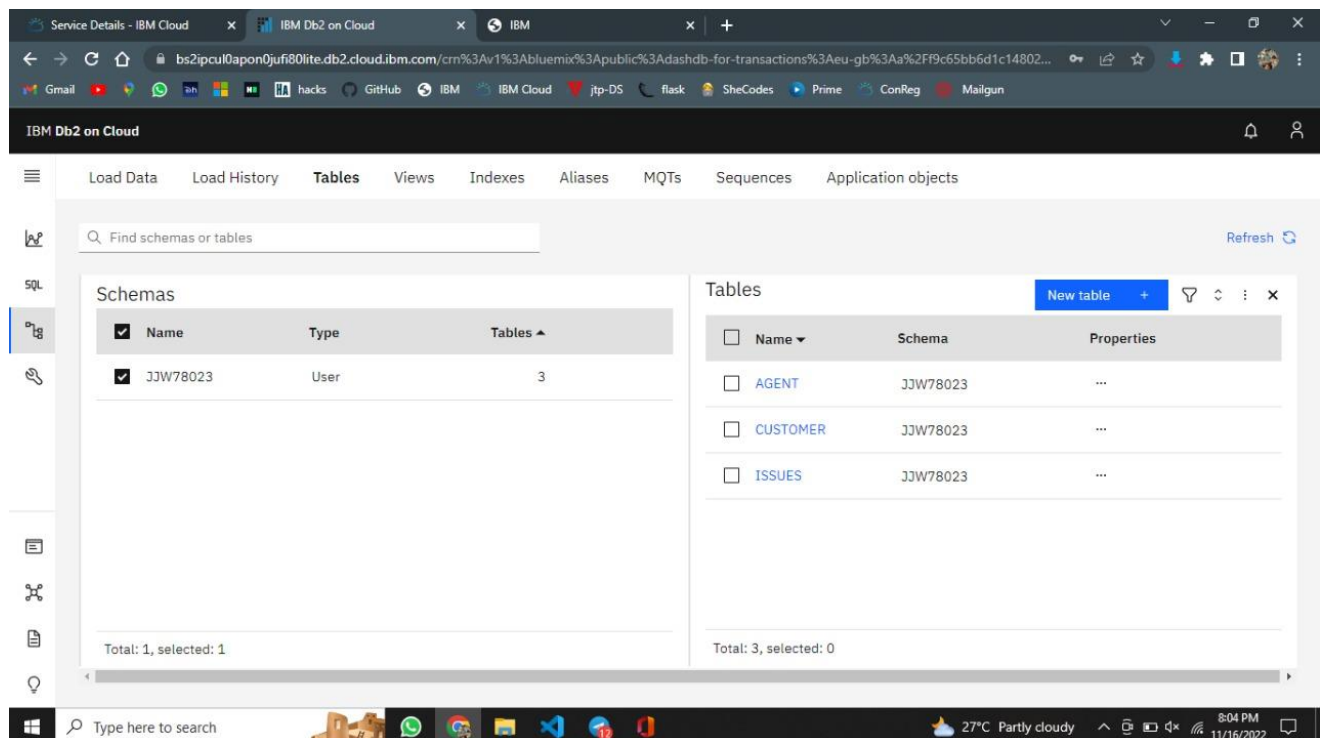
[Download SSL Certificate](#)

**JDBC string**

```
jdbc:db2://55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8lclg.databases.appdomain.cloud:31929/bludb:user=<user name>;password=<your password>;sslConnection=true;
```

**REST API host name**

## Step 3. Creating service credential which contains host name, DB name, UID, Password



**IBM Db2 on Cloud**

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

Find schemas or tables Refresh

### Schemas

Name	Type	Tables
JJW78023	User	3

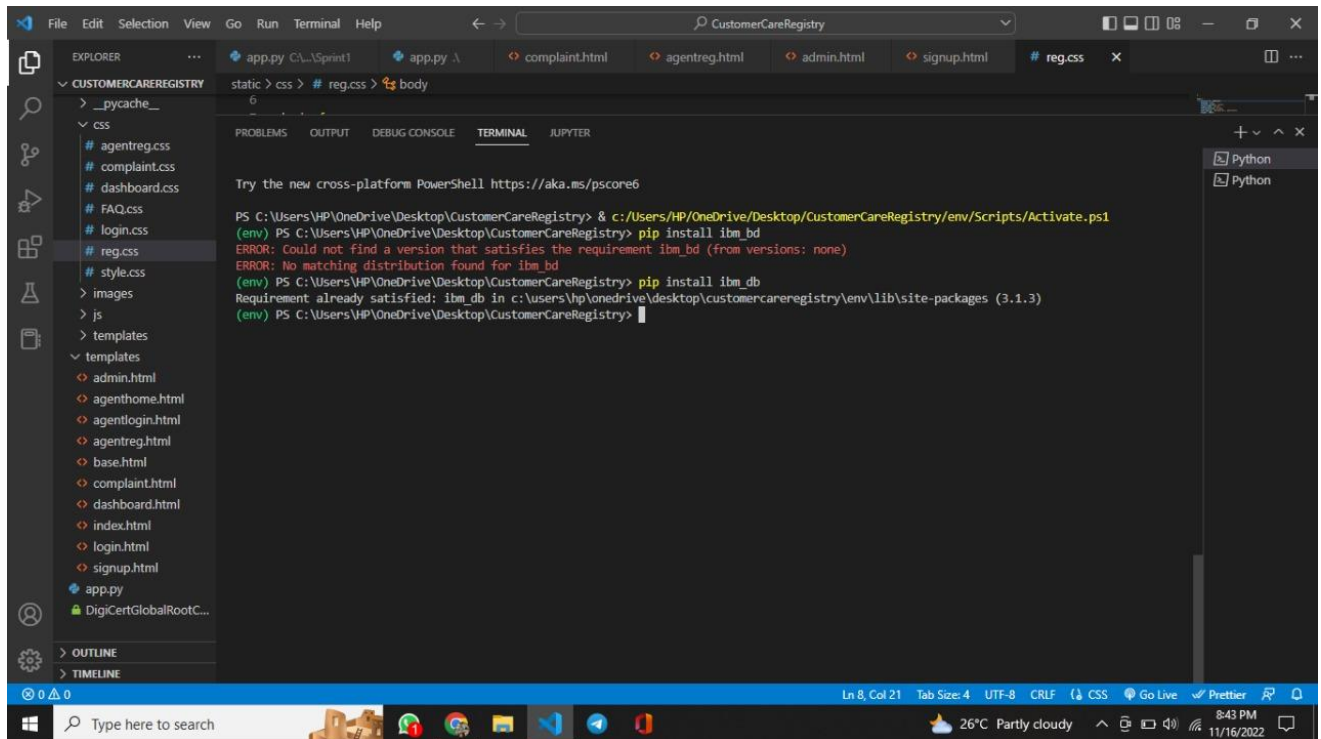
Total: 1, selected: 1

### Tables

Name	Schema	Properties
AGENT	JJW78023	...
CUSTOMER	JJW78023	...
ISSUES	JJW78023	...

Total: 3, selected: 0

## Step 4. Installing ibm-db package

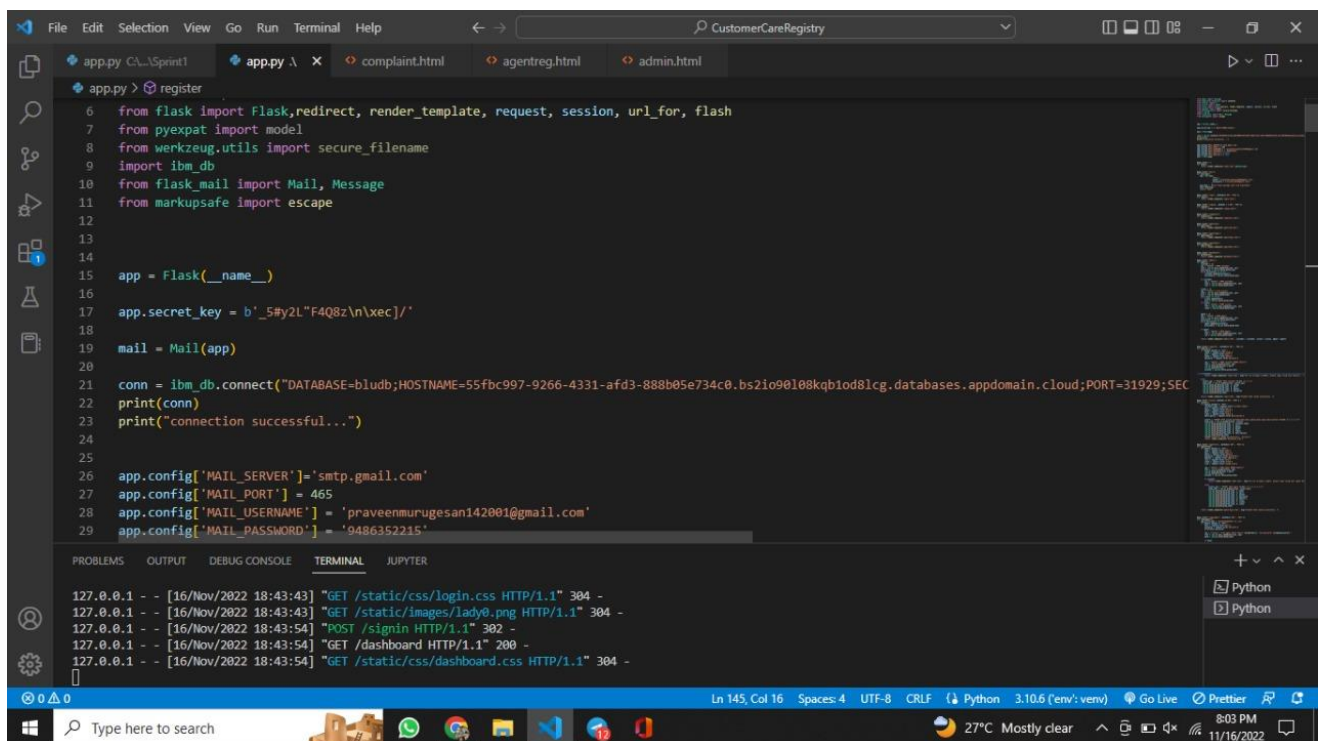


The screenshot shows the Visual Studio Code interface with a project named 'CustomerCareRegistry'. The Explorer pane on the left shows a file structure with folders like 'css', 'images', 'js', and 'templates', and files like 'agentreg.css', 'complaint.css', 'dashboard.css', 'FAQ.css', 'login.css', 'reg.css', 'style.css', 'admin.html', 'agenthome.html', 'agentlogin.html', 'agentreg.html', 'base.html', 'complaint.html', 'dashboard.html', 'index.html', 'login.html', 'signup.html', 'app.py', and 'DigiCertGlobalRootC...'. The Terminal pane at the bottom shows the following commands and output:

```
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\HP\OneDrive\Desktop\CustomerCareRegistry> & c:\Users\HP\OneDrive\Desktop\CustomerCareRegistry\env\Scripts\Activate.ps1
(env) PS C:\Users\HP\OneDrive\Desktop\CustomerCareRegistry> pip install ibm_db
ERROR: Could not find a version that satisfies the requirement ibm_db (from versions: none)
ERROR: No matching distribution found for ibm_db
(env) PS C:\Users\HP\OneDrive\Desktop\CustomerCareRegistry> pip install ibm_db
Requirement already satisfied: ibm_db in c:\users\hp\onedrive\desktop\customerregistery\env\lib\site-packages (3.1.3)
(env) PS C:\Users\HP\OneDrive\Desktop\CustomerCareRegistry>
```

## Step 4. Connecting IBM DB2 with python flask



The screenshot shows the Visual Studio Code interface with the same project. The 'app.py' file is open, showing the following code:

```
6 from flask import Flask, redirect, render_template, request, session, url_for, flash
7 from pyexpat import model
8 from werkzeug.utils import secure_filename
9 import ibm_db
10 from flask_mail import Mail, Message
11 from markupsafe import escape
12
13
14
15 app = Flask(__name__)
16
17 app.secret_key = b'_5#y2L"F4Q8z\n\xec]/'
18
19 mail = Mail(app)
20
21 conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8l1cg.databases.appdomain.cloud;PORT=3129;SEC
22 print(conn)
23 print("connection successful...")
24
25
26 app.config['MAIL_SERVER'] = 'smtp.gmail.com'
27 app.config['MAIL_PORT'] = 465
28 app.config['MAIL_USERNAME'] = 'praveenmurugesan142001@gmail.com'
29 app.config['MAIL_PASSWORD'] = '9486352215'
```

The Terminal pane at the bottom shows the output of the application running on a local server:

```
127.0.0.1 - - [16/Nov/2022 18:43:43] "GET /static/css/login.css HTTP/1.1" 304 -
127.0.0.1 - - [16/Nov/2022 18:43:43] "GET /static/images/lady0.png HTTP/1.1" 304 -
127.0.0.1 - - [16/Nov/2022 18:43:54] "POST /signin HTTP/1.1" 302 -
127.0.0.1 - - [16/Nov/2022 18:43:54] "GET /dashboard HTTP/1.1" 200 -
127.0.0.1 - - [16/Nov/2022 18:43:54] "GET /static/css/dashboard.css HTTP/1.1" 304 -
```

```
1 import email
2 from email import message
3 from importlib.resources import contents
4 from tkinter import S
5 from turtle import title
6 from flask import Flask, redirect, render_template, request, session, url_for, flash
7 from pyexpat import model
8 from werkzeug.utils import secure_filename
9 import ibm_db
10 from flask_mail import Mail, Message
11 from markupsafe import escape
12
13
14
15 app = Flask(__name__)
16
17 app.secret_key = b'_5#y2L"F4Q8z\n\xec]/'
18
19 mail = Mail(app)
20
21 conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=55fbc997-9266-4331-afd3-888b05e734c0.bs21o90108kqb1od8lcg.databases.appdomain1.com;UID=dbadmin;PWD=dbadmin")
22 print(conn)
23 print("connection successful...")
24
25
26 app.config['MAIL_SERVER'] = 'smtp.gmail.com'
27 app.config['MAIL_PORT'] = 465
28 app.config['MAIL_USERNAME'] = 'praveenmurugesan142001@gmail.com'
29 app.config['MAIL_PASSWORD'] = '9486352215'
30 app.config['MAIL_USE_TLS'] = False
31 app.config['MAIL_USE_SSL'] = True
32 mail = Mail(app)
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

## OUTPUT:

IBM Cloud CLI is installed successfully

