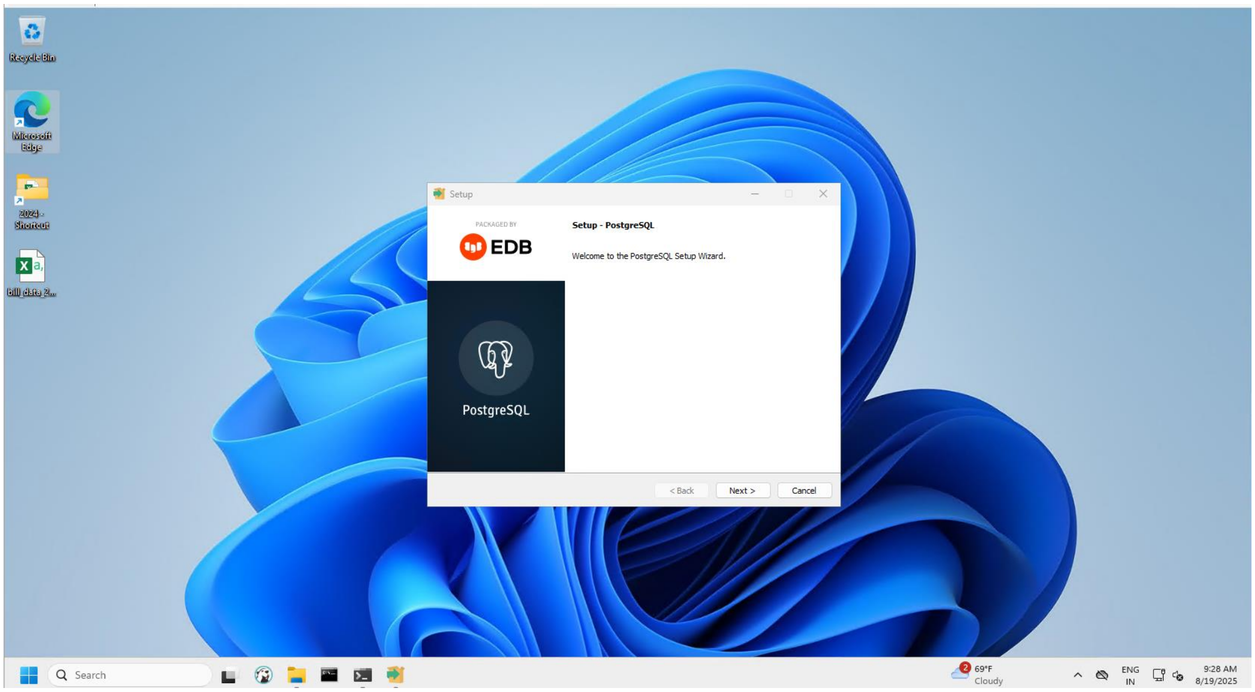
Module 12: Building Database Apps with PostgreSQL & Python

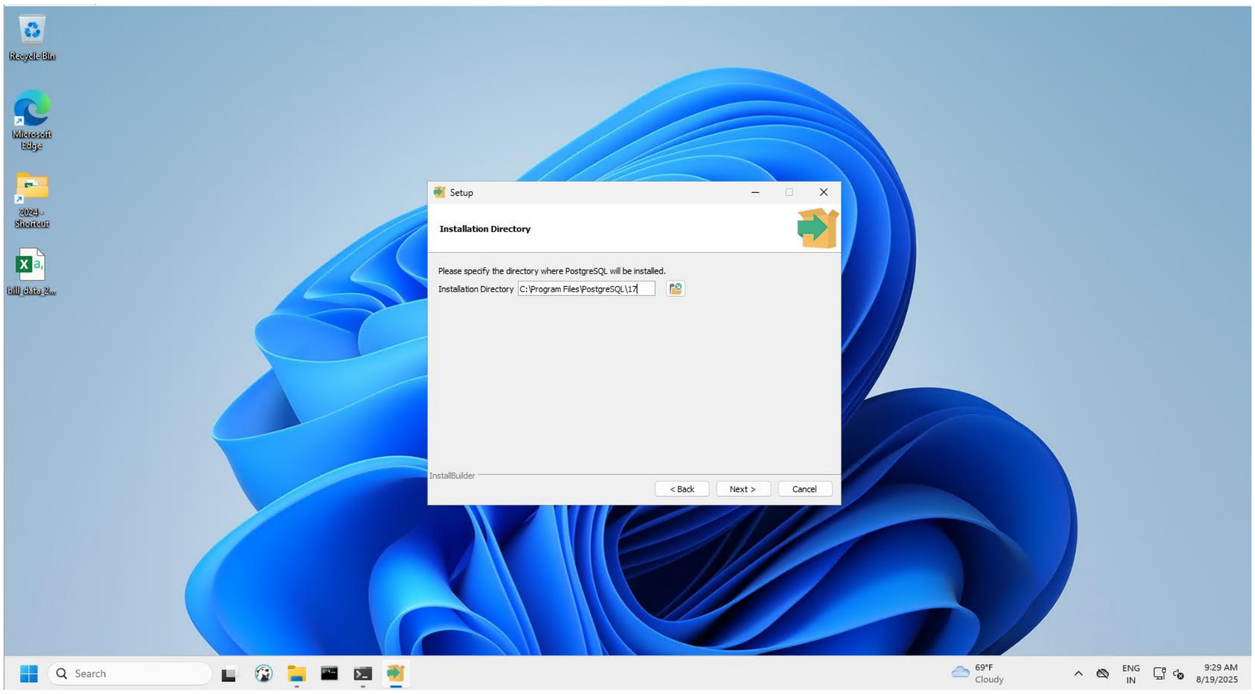
Contents

Installing PostgreSQL


Setup

Setup - PostgreSQL

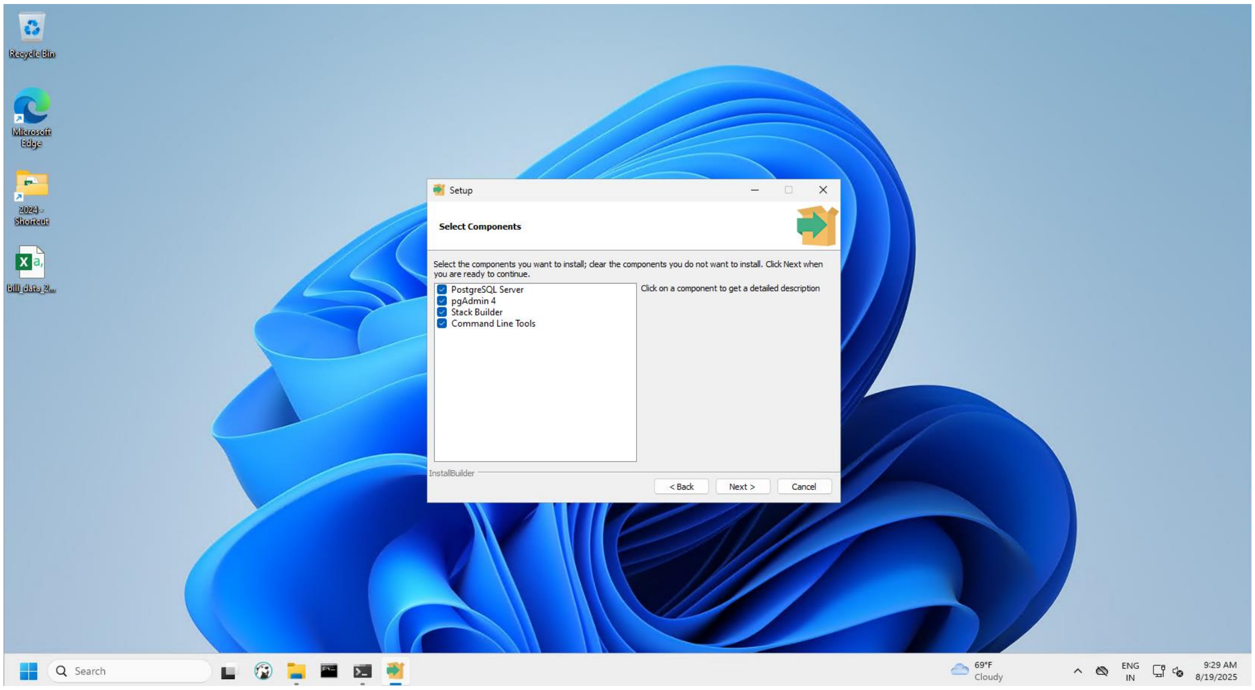
Welcome to the PostgreSQL Setup Wizard.


Setup

Installation Directory

Please specify the directory where PostgreSQL will be installed.

Installation Directory C:\Program Files\PostgreSQL\17

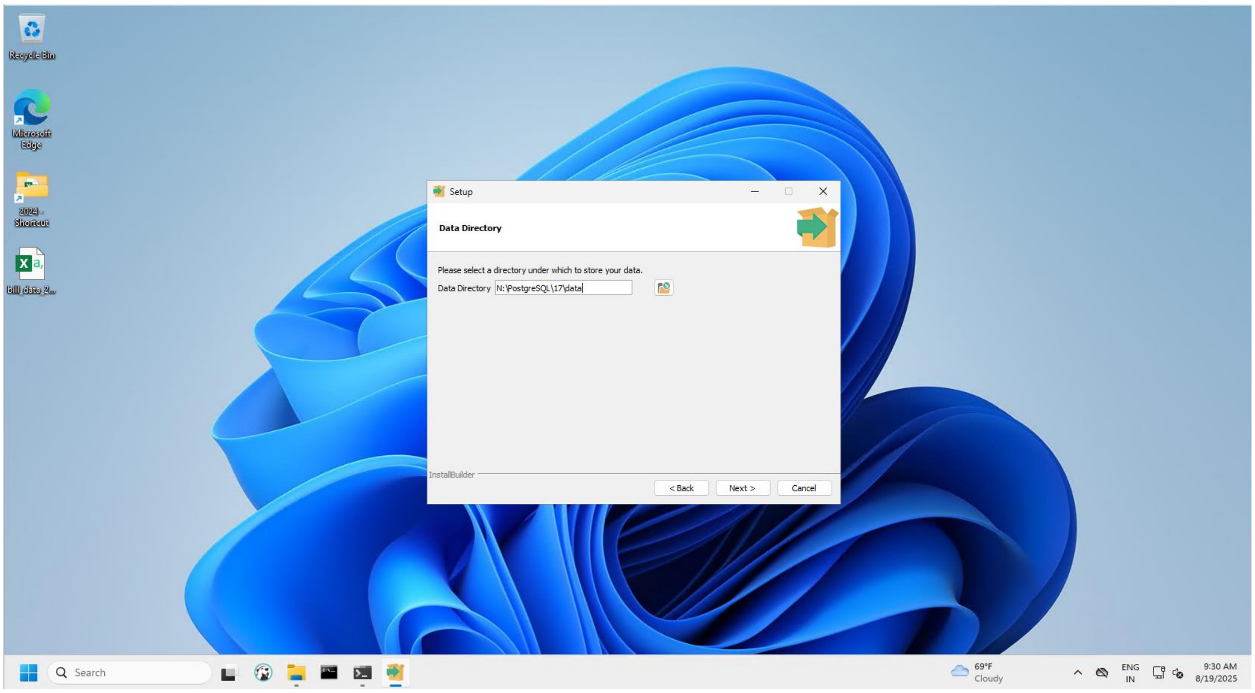


Select Components

Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

- ☑ PostgreSQL Server
- ☑ pgAdmin 4
- ☑ Stack Builder
- ☑ Command Line Tools

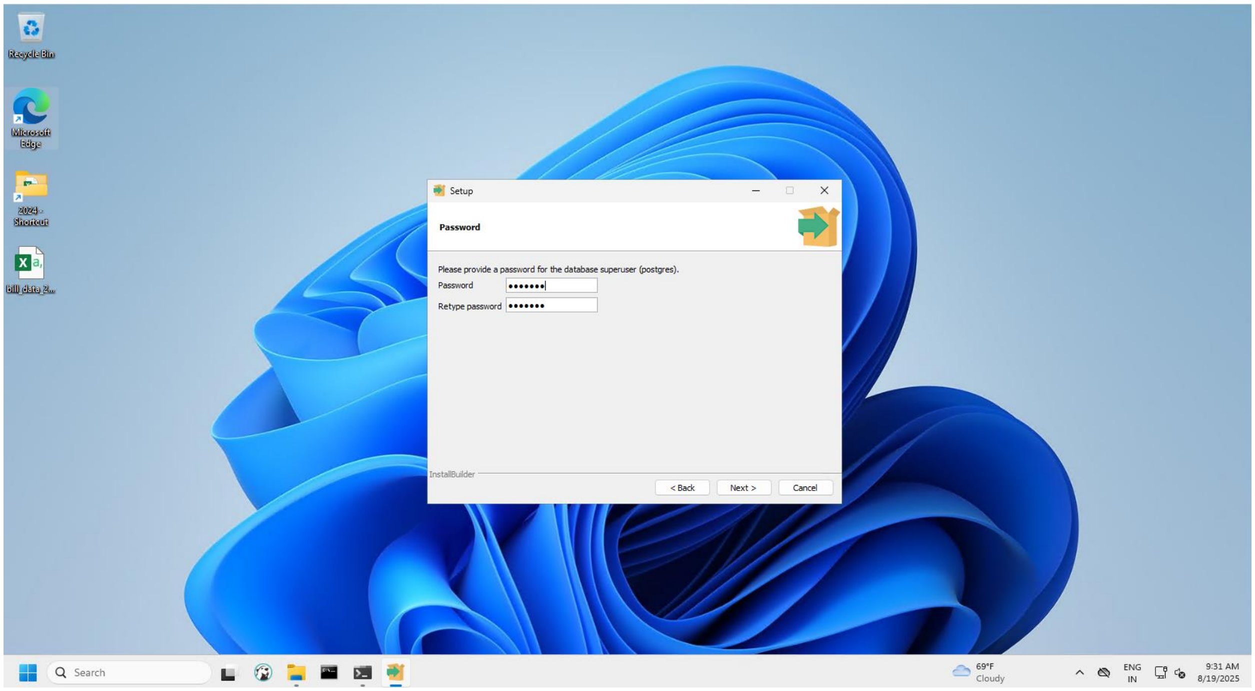
Click on a component to get a detailed description



Data Directory

Please select a directory under which to store your data.

Data Directory N:\PostgreSQL\17\data

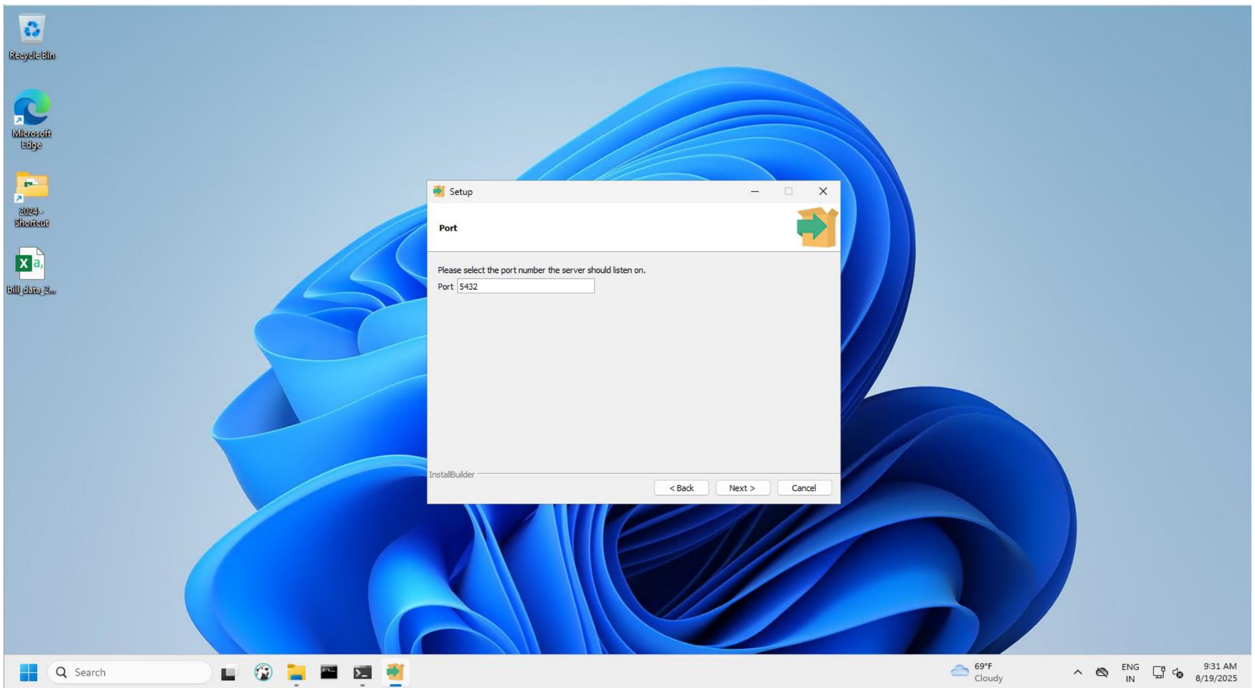


Password

Please provide a password for the database superuser (postgres).

Password •••••••

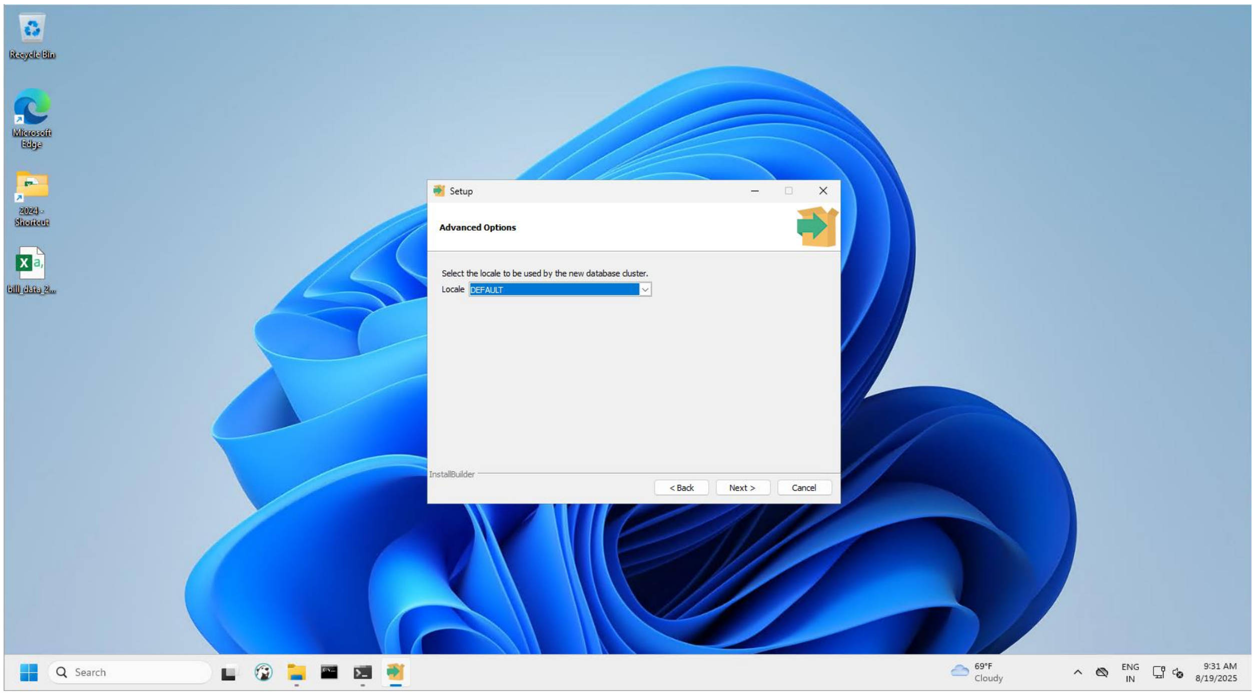
Retype password •••••••



Port

Please select the port number the server should listen on.

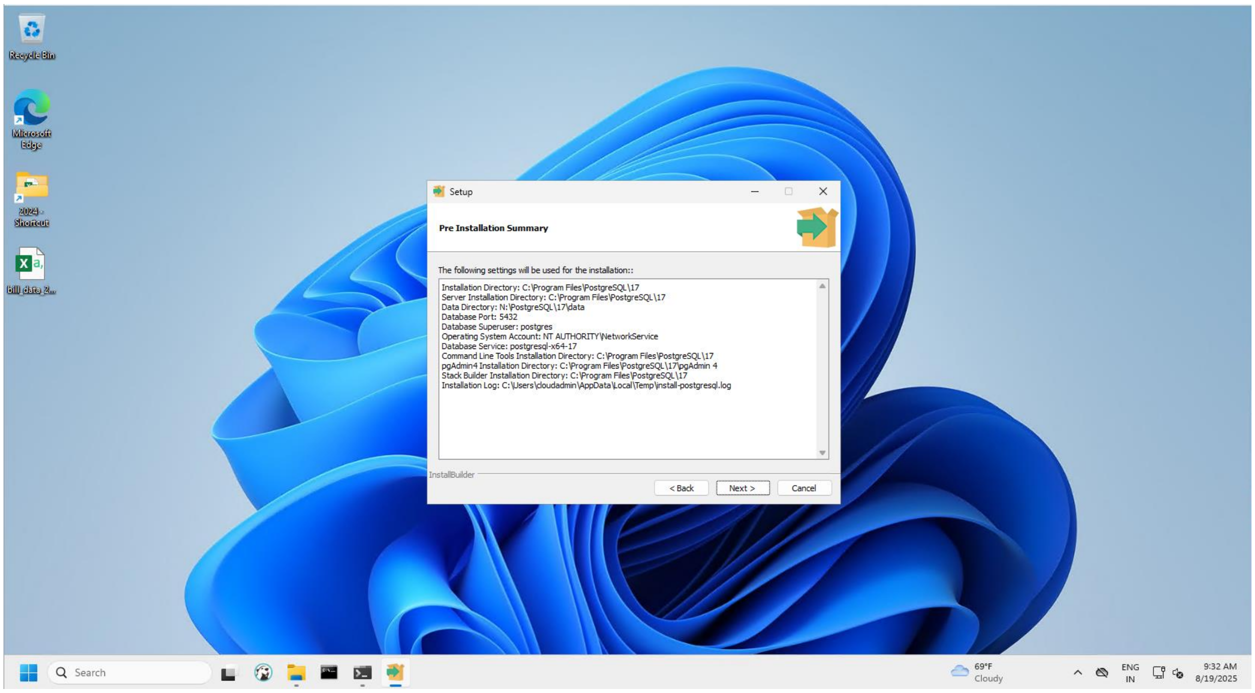
Port 5432



Advanced Options

Select the locale to be used by the new database cluster.

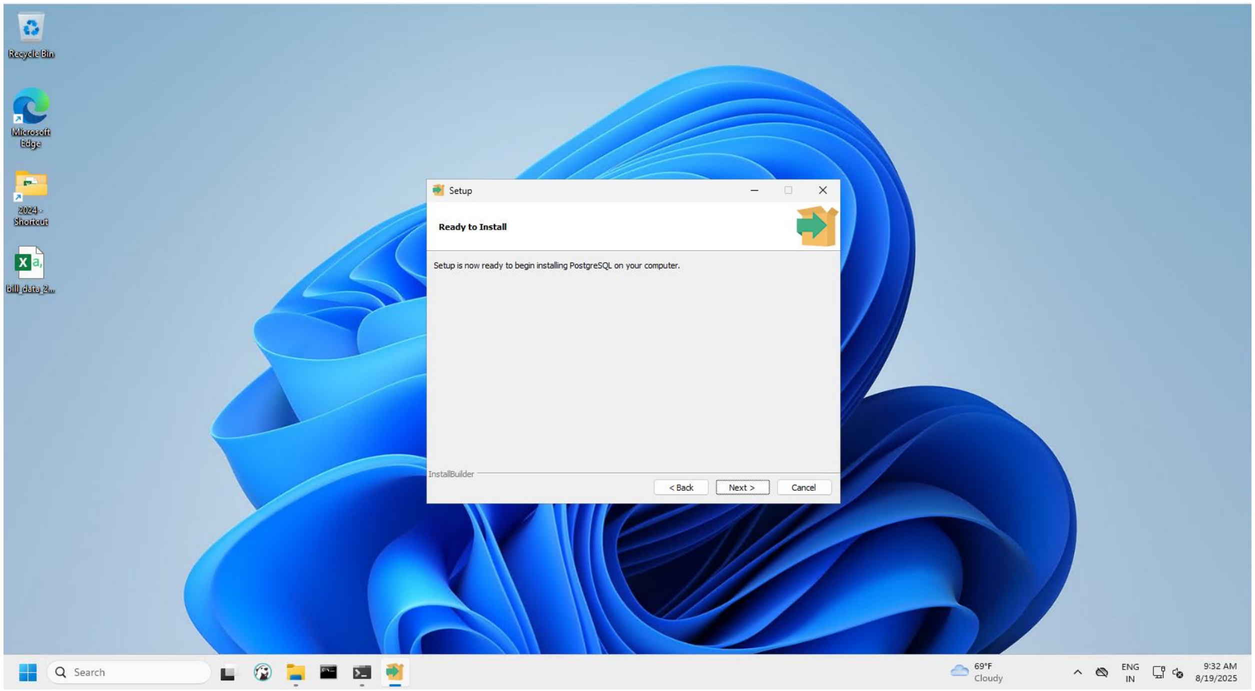
Locale: DEFAULT



Pre Installation Summary

The following settings will be used for the installation::

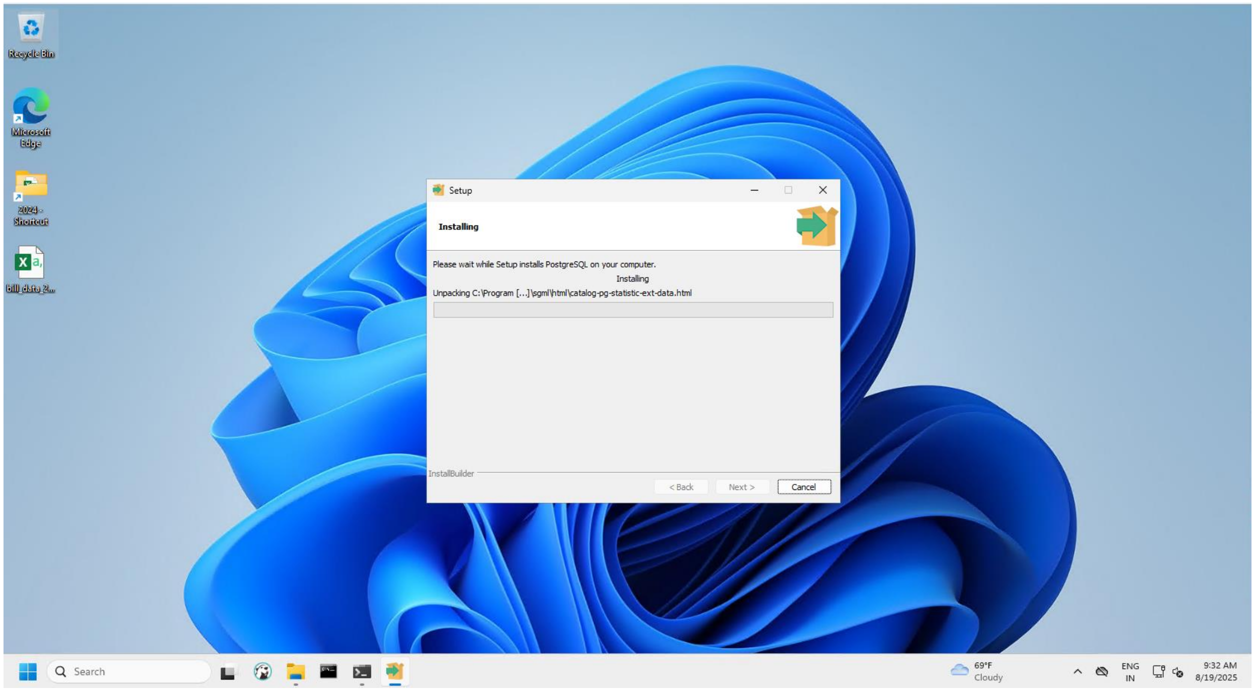
```
Installation Directory: C:\Program Files\PostgreSQL\17
Server Installation Directory: C:\Program Files\PostgreSQL\17
Data Directory: N:\PostgreSQL\17\data
Database Port: 5432
Database Superuser: postgres
Operating System Account: NT AUTHORITY\NetworkService
Database Service: postgresql-x64-17
Command Line Tools Installation Directory: C:\Program Files\PostgreSQL\17
pgAdmin4 Installation Directory: C:\Program Files\PostgreSQL\17\pgAdmin 4
Stack Builder Installation Directory: C:\Program Files\PostgreSQL\17
Installation Log: C:\Users\cloudadmin\AppData\Local\Temp\install-postgresql.log
```



Setup

Ready to Install

Setup is now ready to begin installing PostgreSQL on your computer.



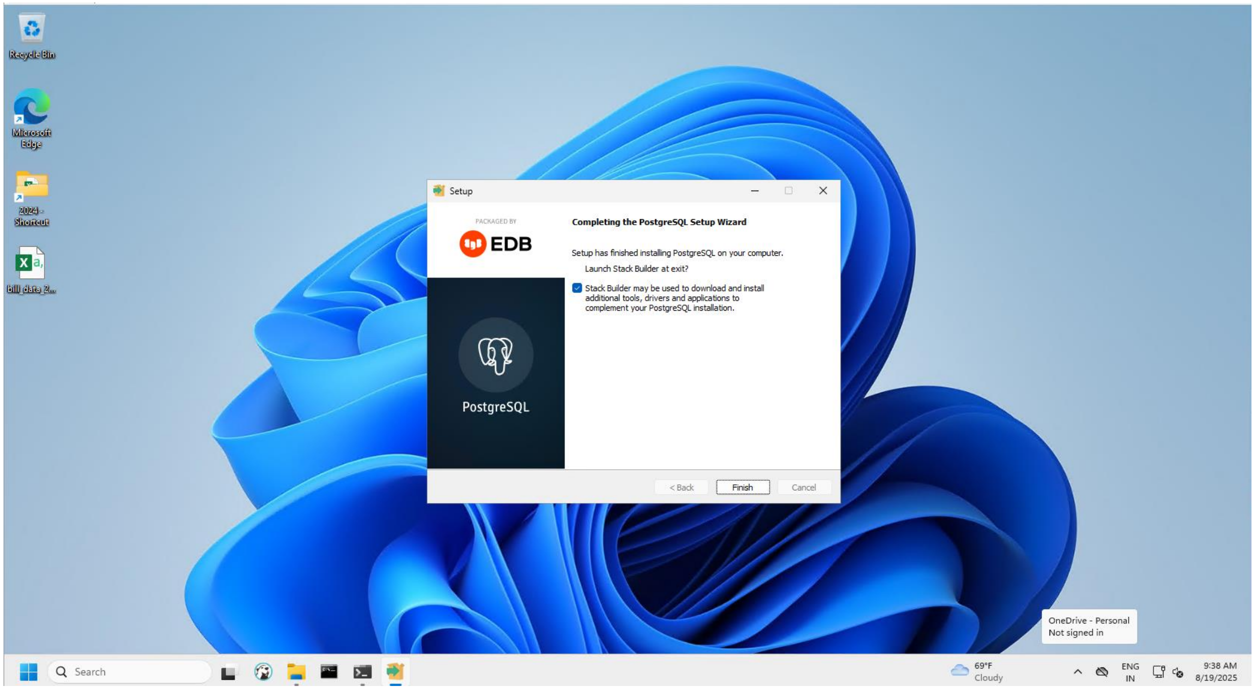
Setup

Installing

Please wait while Setup installs PostgreSQL on your computer.

Installing

Unpacking C:\Program [...]\sgml\html\catalog-pg-statistic-ext-data.html



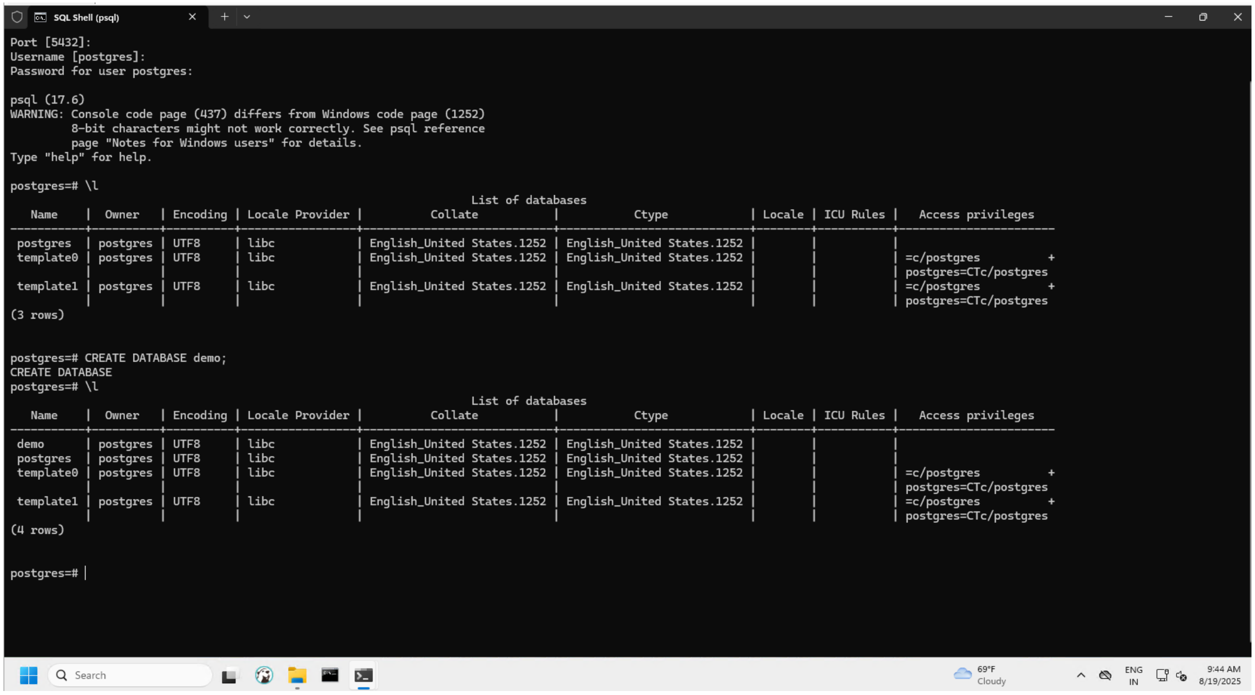
Completing the PostgreSQL Setup Wizard

Setup has finished installing PostgreSQL on your computer.

Launch Stack Builder at exit?

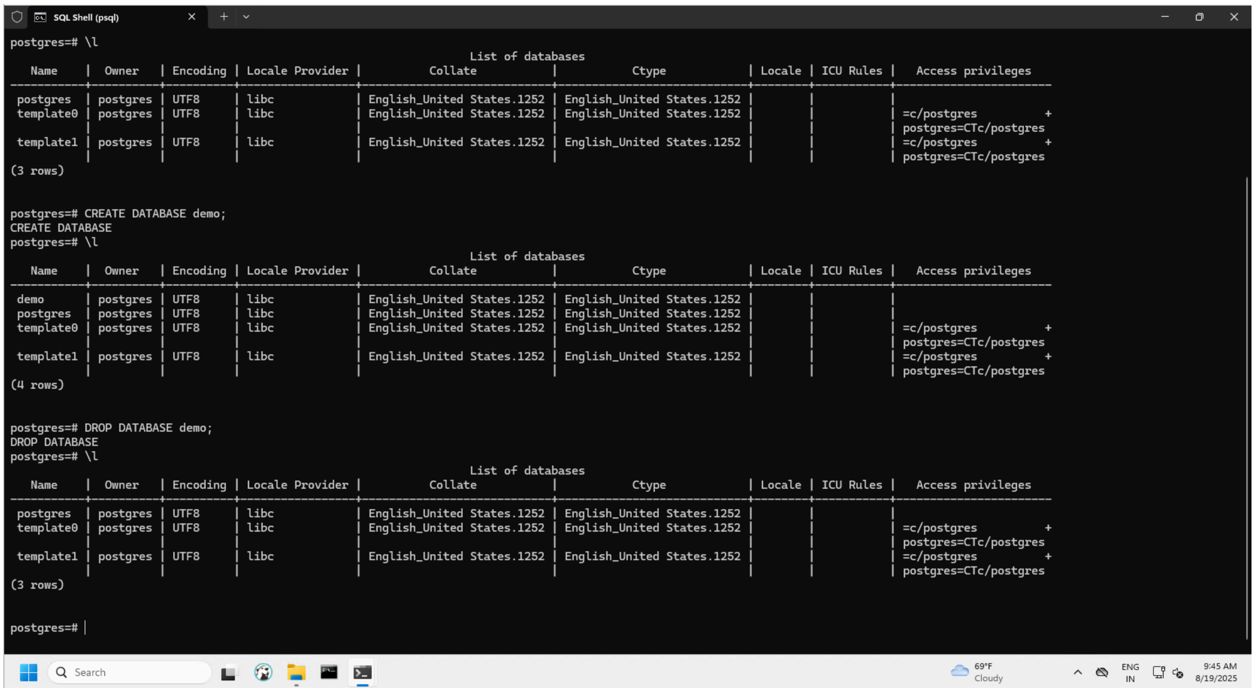
☑ Stack Builder may be used to download and install additional tools, drivers and applications to complement your PostgreSQL installation.

Creating a database



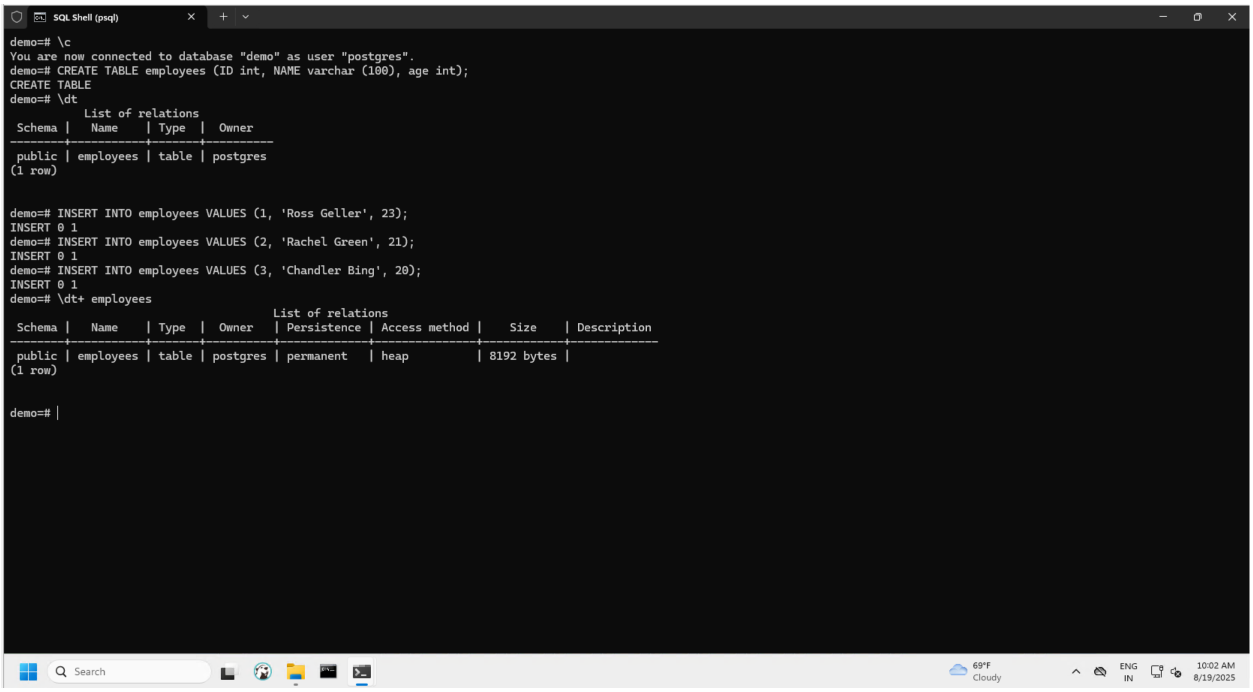
```
Port [5432]:
Username [postgres]:
Password for user postgres:

psql (17.6)
WARNING: Console code page (437) differs from Windows code page (1252)
         8-bit characters might not work correctly. See psql reference
         page "Notes for Windows users" for details.
Type "help" for help.

postgres=# \l
                                                           List of databases
   Name    |  Owner   | Encoding | Locale Provider |          Collate           |           Ctype            | Locale | ICU Rules |   Access privileges
-----------+----------+----------+-----------------+----------------------------+----------------------------+--------+-----------+-----------------------
 postgres  | postgres | UTF8     | libc            | English_United States.1252 | English_United States.1252 |        |           |
 template0 | postgres | UTF8     | libc            | English_United States.1252 | English_United States.1252 |        |           | =c/postgres          +
           |          |          |                 |                            |                            |        |           | postgres=CTc/postgres
 template1 | postgres | UTF8     | libc            | English_United States.1252 | English_United States.1252 |        |           | =c/postgres          +
           |          |          |                 |                            |                            |        |           | postgres=CTc/postgres
(3 rows)


postgres=# CREATE DATABASE demo;
CREATE DATABASE
postgres=# \l
                                                           List of databases
   Name    |  Owner   | Encoding | Locale Provider |          Collate           |           Ctype            | Locale | ICU Rules |   Access privileges
-----------+----------+----------+-----------------+----------------------------+----------------------------+--------+-----------+-----------------------
 demo      | postgres | UTF8     | libc            | English_United States.1252 | English_United States.1252 |        |           |
 postgres  | postgres | UTF8     | libc            | English_United States.1252 | English_United States.1252 |        |           |
 template0 | postgres | UTF8     | libc            | English_United States.1252 | English_United States.1252 |        |           | =c/postgres          +
           |          |          |                 |                            |                            |        |           | postgres=CTc/postgres
 template1 | postgres | UTF8     | libc            | English_United States.1252 | English_United States.1252 |        |           | =c/postgres          +
           |          |          |                 |                            |                            |        |           | postgres=CTc/postgres
(4 rows)


postgres=#
```

Deleting a database



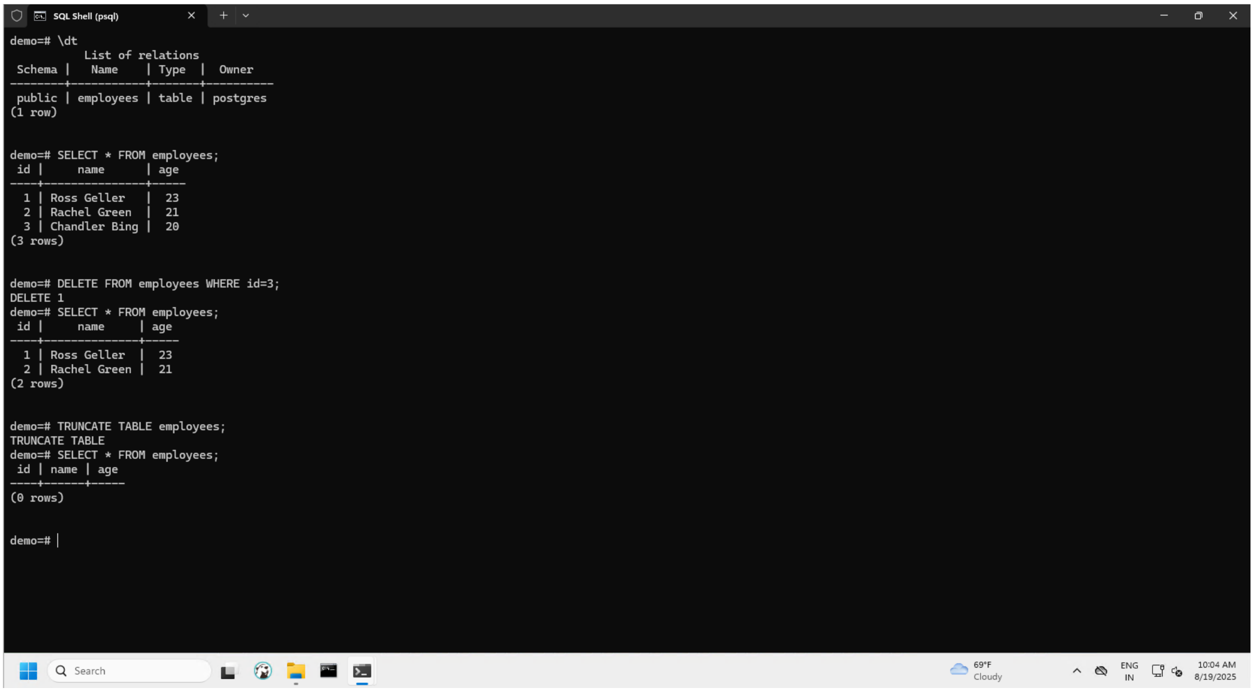
```
postgres=# \l
                                                   List of databases
   Name    |  Owner   | Encoding | Locale Provider |          Collate           |           Ctype            | Locale | ICU Rules |   Access privileges
-----------+----------+----------+-----------------+----------------------------+----------------------------+--------+-----------+-----------------------
 postgres  | postgres | UTF8     | libc            | English_United States.1252 | English_United States.1252 |        |           |
 template0 | postgres | UTF8     | libc            | English_United States.1252 | English_United States.1252 |        |           | =c/postgres          +
           |          |          |                 |                            |                            |        |           | postgres=CTc/postgres
 template1 | postgres | UTF8     | libc            | English_United States.1252 | English_United States.1252 |        |           | =c/postgres          +
           |          |          |                 |                            |                            |        |           | postgres=CTc/postgres
(3 rows)


postgres=# CREATE DATABASE demo;
CREATE DATABASE
postgres=# \l
                                                   List of databases
   Name    |  Owner   | Encoding | Locale Provider |          Collate           |           Ctype            | Locale | ICU Rules |   Access privileges
-----------+----------+----------+-----------------+----------------------------+----------------------------+--------+-----------+-----------------------
 demo      | postgres | UTF8     | libc            | English_United States.1252 | English_United States.1252 |        |           |
 postgres  | postgres | UTF8     | libc            | English_United States.1252 | English_United States.1252 |        |           |
 template0 | postgres | UTF8     | libc            | English_United States.1252 | English_United States.1252 |        |           | =c/postgres          +
           |          |          |                 |                            |                            |        |           | postgres=CTc/postgres
 template1 | postgres | UTF8     | libc            | English_United States.1252 | English_United States.1252 |        |           | =c/postgres          +
           |          |          |                 |                            |                            |        |           | postgres=CTc/postgres
(4 rows)


postgres=# DROP DATABASE demo;
DROP DATABASE
postgres=# \l
                                                   List of databases
   Name    |  Owner   | Encoding | Locale Provider |          Collate           |           Ctype            | Locale | ICU Rules |   Access privileges
-----------+----------+----------+-----------------+----------------------------+----------------------------+--------+-----------+-----------------------
 postgres  | postgres | UTF8     | libc            | English_United States.1252 | English_United States.1252 |        |           |
 template0 | postgres | UTF8     | libc            | English_United States.1252 | English_United States.1252 |        |           | =c/postgres          +
           |          |          |                 |                            |                            |        |           | postgres=CTc/postgres
 template1 | postgres | UTF8     | libc            | English_United States.1252 | English_United States.1252 |        |           | =c/postgres          +
           |          |          |                 |                            |                            |        |           | postgres=CTc/postgres
(3 rows)


postgres=#
```

Creating a table and adding data



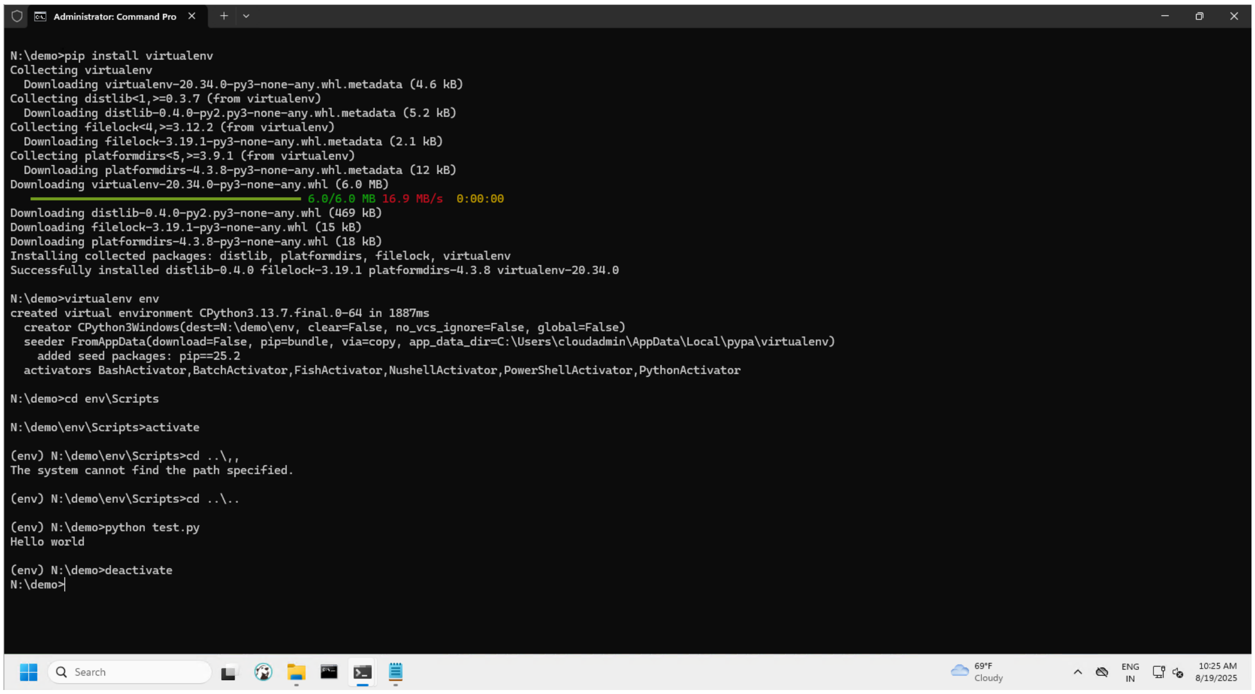
```
demo=# \c
You are now connected to database "demo" as user "postgres".
demo=# CREATE TABLE employees (ID int, NAME varchar (100), age int);
CREATE TABLE
demo=# \dt
           List of relations
 Schema |   Name    | Type  |  Owner
--------+-----------+-------+----------
 public | employees | table | postgres
(1 row)


demo=# INSERT INTO employees VALUES (1, 'Ross Geller', 23);
INSERT 0 1
demo=# INSERT INTO employees VALUES (2, 'Rachel Green', 21);
INSERT 0 1
demo=# INSERT INTO employees VALUES (3, 'Chandler Bing', 20);
INSERT 0 1
demo=# \dt+ employees
                                       List of relations
 Schema |   Name    | Type  |  Owner   | Persistence | Access method |    Size    | Description
--------+-----------+-------+----------+-------------+---------------+------------+-------------
 public | employees | table | postgres | permanent   | heap          | 8192 bytes |
(1 row)


demo=#
```

Retrieving data from the database and deleting the contents in the table



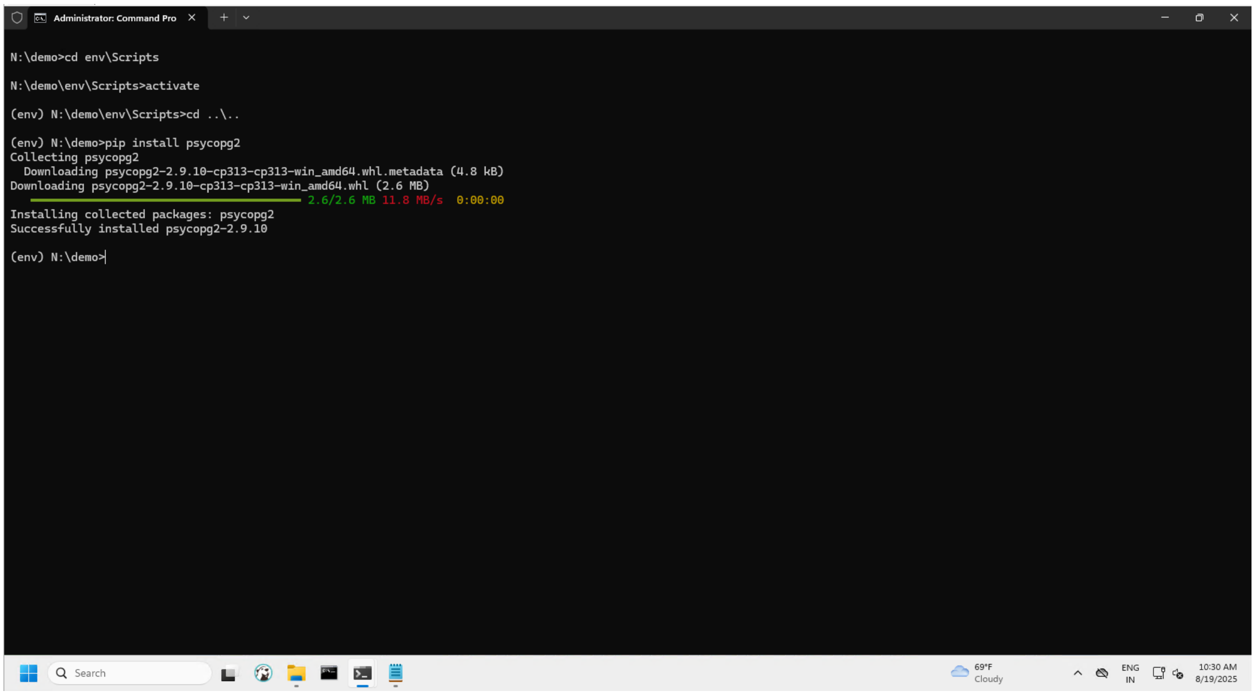
```
demo=# \dt
           List of relations
 Schema |   Name    | Type  |  Owner
--------+-----------+-------+----------
 public | employees | table | postgres
(1 row)


demo=# SELECT * FROM employees;
 id |     name      | age
----+---------------+-----
  1 | Ross Geller   |  23
  2 | Rachel Green  |  21
  3 | Chandler Bing |  20
(3 rows)


demo=# DELETE FROM employees WHERE id=3;
DELETE 1
demo=# SELECT * FROM employees;
 id |     name     | age
----+--------------+-----
  1 | Ross Geller  |  23
  2 | Rachel Green |  21
(2 rows)


demo=# TRUNCATE TABLE employees;
TRUNCATE TABLE
demo=# SELECT * FROM employees;
 id | name | age
----+------+-----
(0 rows)


demo=# |
```

Setting up *virtualenv*



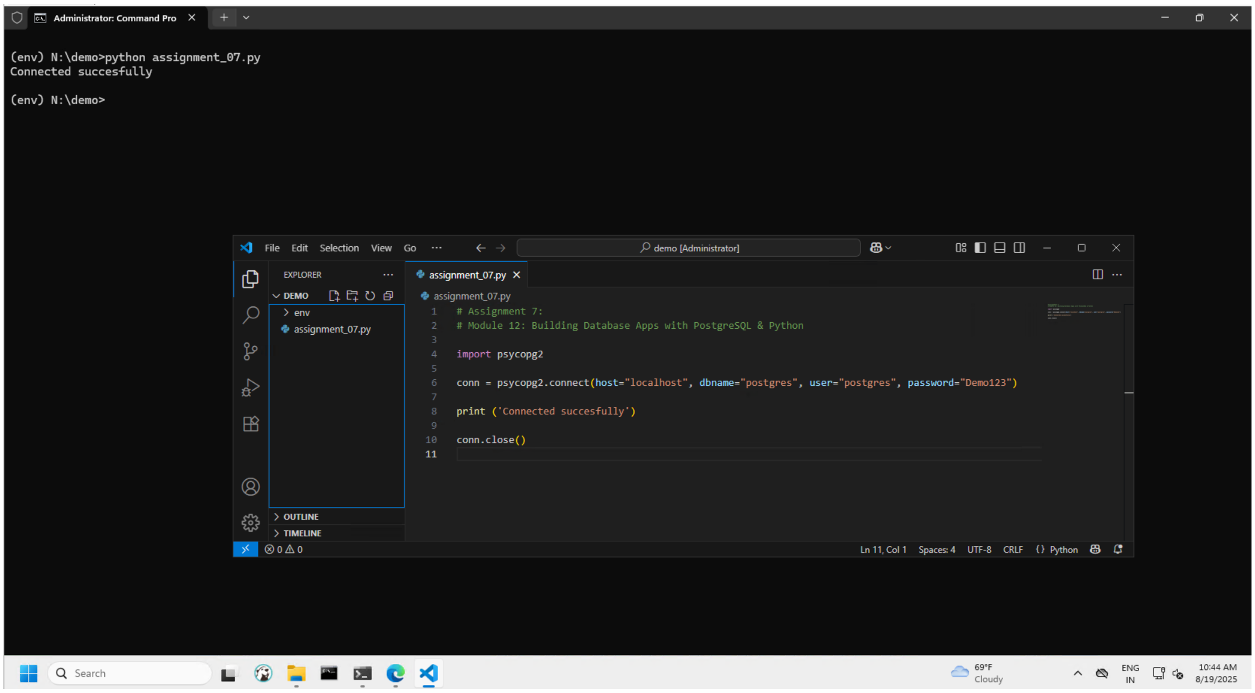
```
N:\demo>pip install virtualenv
Collecting virtualenv
  Downloading virtualenv-20.34.0-py3-none-any.whl.metadata (4.6 kB)
Collecting distlib<1,>=0.3.7 (from virtualenv)
  Downloading distlib-0.4.0-py2.py3-none-any.whl.metadata (5.2 kB)
Collecting filelock<4,>=3.12.2 (from virtualenv)
  Downloading filelock-3.19.1-py3-none-any.whl.metadata (2.1 kB)
Collecting platformdirs<5,>=3.9.1 (from virtualenv)
  Downloading platformdirs-4.3.8-py3-none-any.whl.metadata (12 kB)
Downloading virtualenv-20.34.0-py3-none-any.whl (6.0 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 6.0/6.0 MB 16.9 MB/s  0:00:00
Downloading distlib-0.4.0-py2.py3-none-any.whl (469 kB)
Downloading filelock-3.19.1-py3-none-any.whl (15 kB)
Downloading platformdirs-4.3.8-py3-none-any.whl (18 kB)
Installing collected packages: distlib, platformdirs, filelock, virtualenv
Successfully installed distlib-0.4.0 filelock-3.19.1 platformdirs-4.3.8 virtualenv-20.34.0

N:\demo>virtualenv env
created virtual environment CPython3.13.7.final.0-64 in 1887ms
  creator CPython3Windows(dest=N:\demo\env, clear=False, no_vcs_ignore=False, global=False)
  seeder FromAppData(download=False, pip=bundle, via=copy, app_data_dir=C:\Users\cloudadmin\AppData\Local\pypa\virtualenv)
    added seed packages: pip==25.2
  activators BashActivator,BatchActivator,FishActivator,NushellActivator,PowerShellActivator,PythonActivator

N:\demo>cd env\Scripts

N:\demo\env\Scripts>activate

(env) N:\demo\env\Scripts>cd ..\,,
The system cannot find the path specified.

(env) N:\demo\env\Scripts>cd ..\..

(env) N:\demo>python test.py
Hello world

(env) N:\demo>deactivate
N:\demo>
```

Installing *psycopg2*



```
N:\demo>cd env\Scripts

N:\demo\env\Scripts>activate

(env) N:\demo\env\Scripts>cd ..\..

(env) N:\demo>pip install psycopg2
Collecting psycopg2
  Downloading psycopg2-2.9.10-cp313-cp313-win_amd64.whl.metadata (4.8 kB)
Downloading psycopg2-2.9.10-cp313-cp313-win_amd64.whl (2.6 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 2.6/2.6 MB 11.8 MB/s  0:00:00
Installing collected packages: psycopg2
Successfully installed psycopg2-2.9.10

(env) N:\demo>
```

Connecting to the database



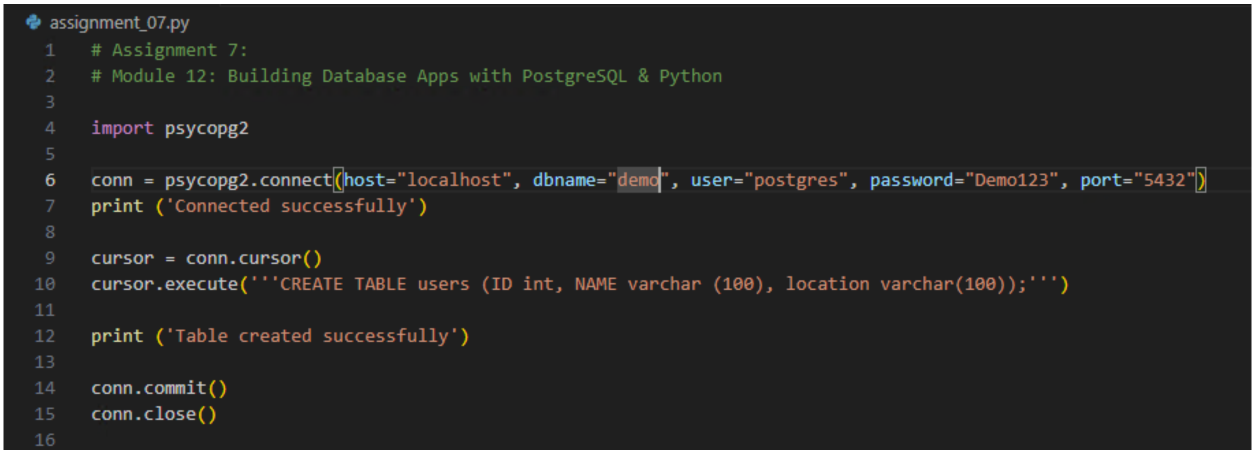
```
(env) N:\demo>python assignment_07.py
Connected succesfully

(env) N:\demo>
```

assignment_07.py

```python
# Assignment 7:
# Module 12: Building Database Apps with PostgreSQL & Python

import psycopg2

conn = psycopg2.connect(host="localhost", dbname="postgres", user="postgres", password="Demo123")

print ('Connected succesfully')

conn.close()

```

Creating table using Python

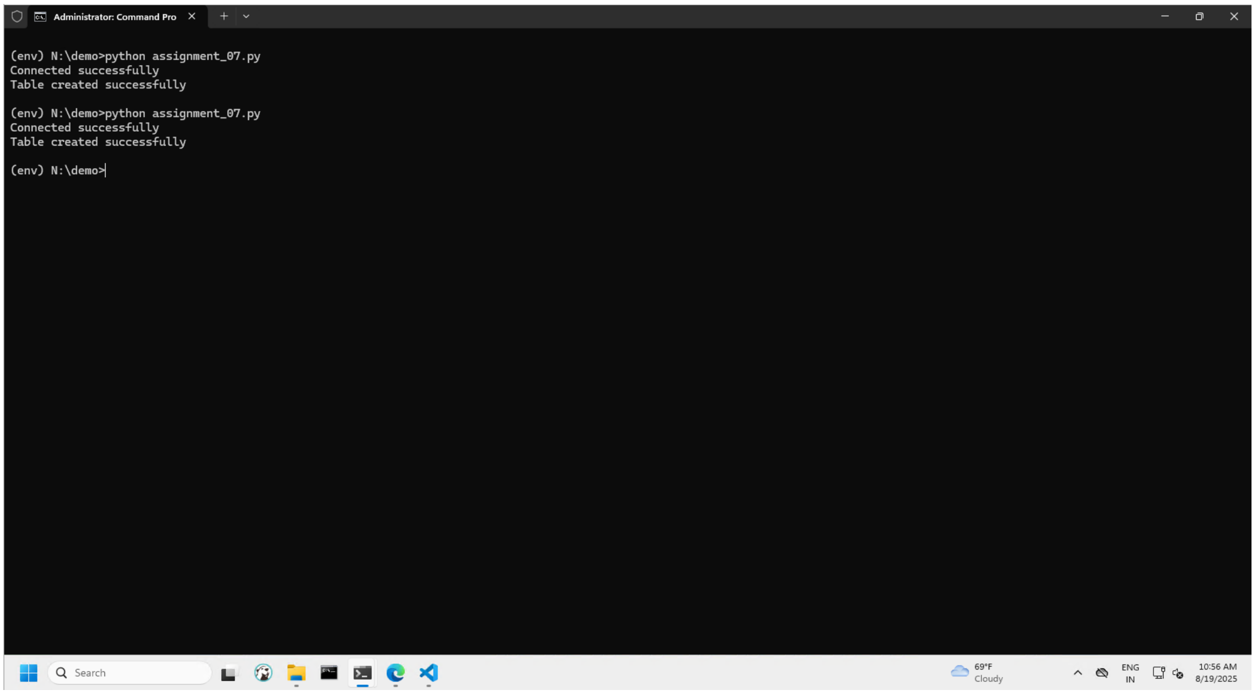
Python code



assignment_07.py

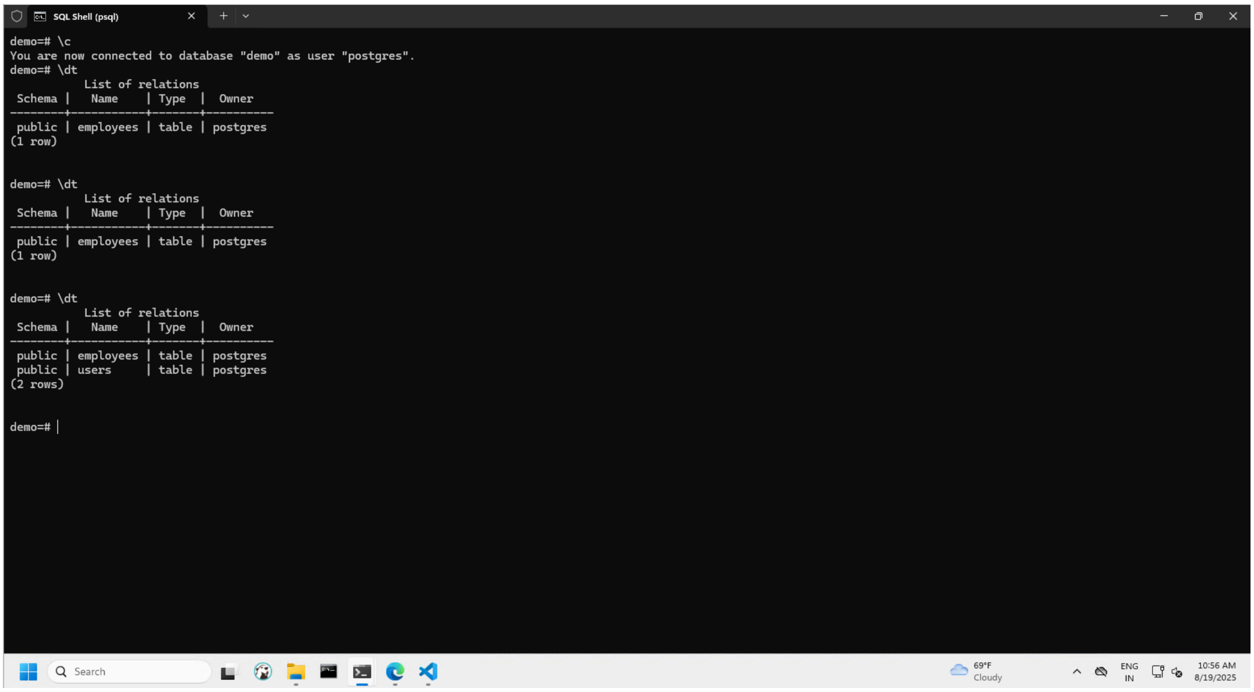
```python
# Assignment 7:
# Module 12: Building Database Apps with PostgreSQL & Python

import psycopg2

conn = psycopg2.connect(host="localhost", dbname="demo", user="postgres", password="Demo123", port="5432")
print ('Connected successfully')

cursor = conn.cursor()
cursor.execute('''CREATE TABLE users (ID int, NAME varchar (100), location varchar(100));''')

print ('Table created successfully')

conn.commit()
conn.close()

```

Python execution



```
(env) N:\demo>python assignment_07.py
Connected successfully
Table created successfully

(env) N:\demo>python assignment_07.py
Connected successfully
Table created successfully

(env) N:\demo>
```

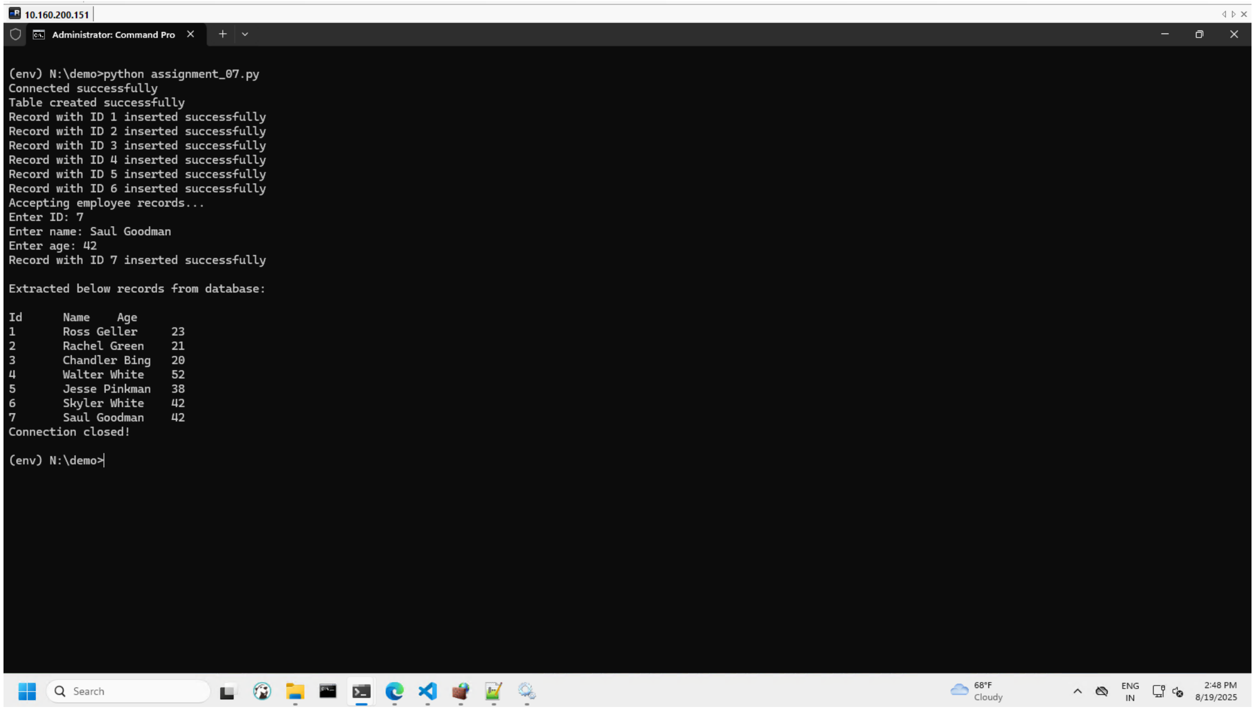
Psql console



```
demo=# \c
You are now connected to database "demo" as user "postgres".
demo=# \dt
          List of relations
 Schema |   Name    | Type  |  Owner
--------+-----------+-------+----------
 public | employees | table | postgres
(1 row)


demo=# \dt
          List of relations
 Schema |   Name    | Type  |  Owner
--------+-----------+-------+----------
 public | employees | table | postgres
(1 row)


demo=# \dt
          List of relations
 Schema |   Name    | Type  |  Owner
--------+-----------+-------+----------
 public | employees | table | postgres
 public | users     | table | postgres
(2 rows)


demo=#
```

Inserting the data using Python, extracting the data from the database and adding the input from the user

Source code: **assignment_07.py**



```
(env) N:\demo>python assignment_07.py
Connected successfully
Table created successfully
Record with ID 1 inserted successfully
Record with ID 2 inserted successfully
Record with ID 3 inserted successfully
Record with ID 4 inserted successfully
Record with ID 5 inserted successfully
Record with ID 6 inserted successfully
Accepting employee records...
Enter ID: 7
Enter name: Saul Goodman
Enter age: 42
Record with ID 7 inserted successfully

Extracted below records from database:

Id      Name    Age
1       Ross Geller     23
2       Rachel Green    21
3       Chandler Bing   20
4       Walter White    52
5       Jesse Pinkman   38
6       Skyler White    42
7       Saul Goodman    42
Connection closed!

(env) N:\demo>
```