

Lab 5 PostgreSQL Connection Setup

Connection Details for Database Access

Below are the required credentials for connecting to the database:

- **Host:** 134.173.236.12
- **Port:** 1433
- **Username:** GenAI_Spring25_Firstname_Lastname
- **Initial Password:** Genai2025!
- **Database_name:** GenAI_Spring25_Firstname_Lastname_db

Important:

- Ensure you are connected to the VPN before attempting to access the database. For guidance on setting up the VPN, please refer to [How to Install CGU VPN](#).
 - You must **change the initial password** to your own by following **Step 5** below.
-

1. Connect it from VS Code Extension

- **Step 1:** Open the **Extensions** panel in your **VS Code**.
- **Step 2:** Search for **Database Client**, and **Install** it
- **Step 3:** Click the database icon on the left panel and click **Connect Connection**.
- **Step 4:** Click **PostgreSQL**, then fill out all the required information (see top of the page): **Name**, **Host**, **Port**, **Username**, **Password**, and **Database**. (For the **Name** you can define any name you like**) **
- **Step 5:** After entering all the required information, click **Connect**. If everything is correct, you will see a 'Success!' message at the top. Then, you can click **Save** to store the connection.

If the connection is successful, the left side panel will appear as a database connection of the database name as shown below.

- **Step 5: Change Your Initial Password**
 1. **Open the Query Editor:**
 1. Navigate to **your database > Public**.
 2. Right-click on **Query > Create Query Book**.

3. Provide a name for your new query book.
2. Run the Following Code in the Query Book, and click run (Replace *"Your_Username"* and **"New_Password"** with your actual username and desired password.)

Please do not add any special characters in the password, just a combination of letters and numbers.

```
ALTER ROLE "Your_Username" WITH PASSWORD 'New_Password';
```

- 1. After updating your password, a **"Password Authentication Failed"** message may appear in the left-side panel.

To update it:

1. Right-click your database server in the Database Client panel.
 2. Select "Edit Connection".
 3. Enter your new password in the "Password" field.
 4. Click "Connect" to test, then "Save" to update.
- **Step 6: Testing.** Find your database in the list on the left, open it, and create a test table to ensure you have the necessary access.

2. Setting Up PostgreSQL Connection in a *.env* File

- **Step 1: Install Required Dependencies**

Method 1 (Terminal)

- Activate your **Virtual Environment (venv)** in the VS Code terminal. (*Refer to Lab 0 for activation instructions.*)
- Once your venv is activated, install the required packages **separately**:

For Windows:

```
pip install "psycopg[binary]"  
pip install SQLAlchemy
```

And then try this to see if you have the psycopg installed - psycopg 3.2.5

```
pip list | findstr psycopg
```

■

For macOS:

```
pip install "psycopg[binary]"
```

```
pip install SQLAlchemy
```

And then try this to see if you have the psycopg installed - psycopg 3.2.5

```
pip list | grep psycopg
```

■

If you encounter issues on **macOS/Linux**, ensure that **PostgreSQL client libraries (libpq)** are installed:

```
brew install libpq
```

```
pip install psycopg
```

■

- **Method 2 (Python Notebook)**

- If you are working in a **Jupyter Notebook** with an activated venv kernel, install the packages **separately** using:

For Windows:

```
%pip install "psycopg[binary]"
```

```
%pip install SQLAlchemy
```

And then try this to see if you have the psycopg installed - psycopg 3.2.5

```
pip list | findstr psycopg
```

■

For macOS:

```
%pip install "psycopg[binary]"
```

```
%pip install SQLAlchemy
```

And then try this to see if you have the psycopg installed - psycopg 3.2.5

```
pip list | grep psycopg
```

■

Step 2: Add this to your `.env` file and replace the parameters with your own (the red part).

```
DB_CONNECTION=
"postgresql+psycopg://username:password@134.173.236.12:1433/database_name"
```

●

Step 3: Generate a `database_connection_testing.ipynb` file, copy the following code into it and run it to test.

```
from langchain_postgres import PGVector
from dotenv import load_dotenv
import os
```

```
load_dotenv()
connection_string = os.getenv("DB_CONNECTION")
```

try:

```
# Test connection using PGVector (requires psycopg3)
```

```
test_db = PGVector(
    embeddings=None, # No embedding function needed for connection check
    collection_name="test_connection",
    connection=connection_string,
    use_jsonb=True,
)
```

```
print("Connection successful! You are using psycopg3, and langchain_postgres is working.")
```

except Exception as e:

```
print("Connection failed! Check your DB setup and psycopg3 installation.")
print(f"Error: {str(e)}")
```


●

By following this guide, you have successfully:

✓ **Set up and tested** the database connection in **VS Code**

✓ **Installed** the required dependencies using `pip` in your VM

✓ **Added** the database connection string to the `.env` file

 **Verified** database access by executing a test query