Deleting DAGs

**Solution: Use the Correct Container and Commands**

**Method 1: Access PostgreSQL through the Airflow Scheduler Container**

Since you have an Airflow setup, try accessing PostgreSQL through the scheduler container:

1. **Go back to your containers list** in Docker Desktop
2. **Click on the airflow-scheduler-1 container** (not the postgres one)
3. **Open terminal** for that container
4. **Run this command:**

bash

psql postgresql://airflow:airflow@postgres-1:5432/airflow

**Method 2: Fix the PostgreSQL Container Access**

If you want to use the postgres container directly:

1. **In the postgres container terminal**, try:

bash

su - postgres

psql -U airflow -d airflow

**Method 3: Use Docker Command Line (Recommended)**

Open **PowerShell or Command Prompt** on your Windows machine and run:

bash

*# Connect to PostgreSQL directly*

docker exec -it airflow3x2-postgres-1 psql -U airflow -d airflow

*# Or if that doesn't work, try:*

docker exec -it airflow3x2-postgres-1 su - postgres -c "psql -U airflow -d airflow"

**Method 4: Check Connection Details First**

Let's verify the connection details by checking your docker-compose file or environment variables:

bash

*# Check the database connection string from airflow scheduler*

docker exec -it airflow-scheduler-1 airflow config get-value database sql\_alchemy\_conn

**Method 5: Alternative PostgreSQL Access**

If the above methods don't work, try:

bash

*# Access postgres as superuser first*

docker exec -it airflow3x2-postgres-1 psql -U postgres

*# Then connect to airflow database*

\c airflow

**Once Connected to PostgreSQL:**

When you successfully connect, you can run these commands to clean up sample DAGs:

sql

*-- List all DAGs*

SELECT dag\_id FROM dag;

*-- Delete example DAGs*

DELETE FROM dag WHERE dag\_id LIKE 'example\_%';

DELETE FROM dag\_run WHERE dag\_id LIKE 'example\_%';

DELETE FROM task\_instance WHERE dag\_id LIKE 'example\_%';

*-- Exit*

\q

Try **Method 3** first (using PowerShell/Command Prompt) as it's usually the most reliable approach for accessing PostgreSQL in Docker containers.

A computer screen with white text

Description automatically generated