VM for Airflow (Windows 10)

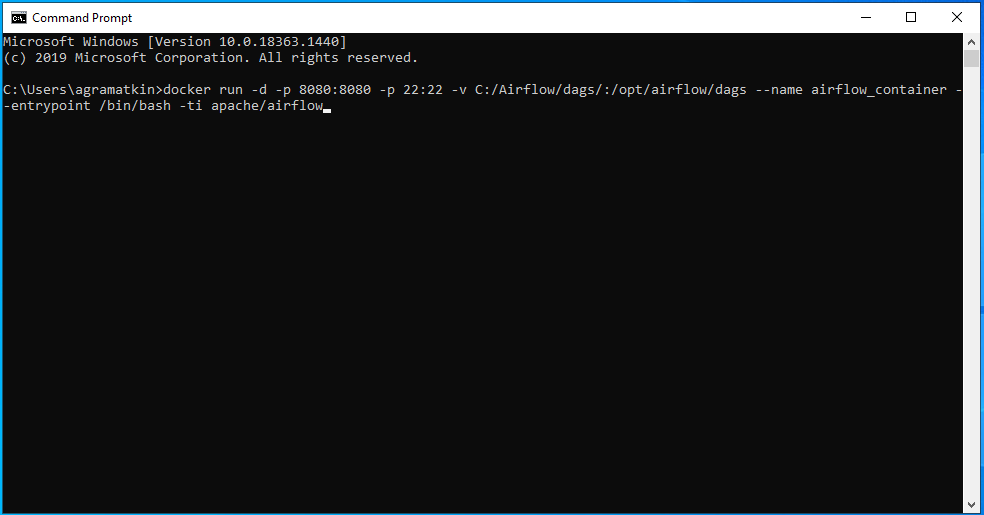
1. Install Git for Windows (<https://git-scm.com/download/win>)

2. Install Docker for Windows (<https://docs.docker.com/docker-for-windows/install/)>

3. Download Airflow Image for Docker (<https://hub.docker.com/r/apache/airflow>)

4. Run in CMD:

docker run -d -p 8080:8080 -p 22:22 -v C:/Airflow/dags/:/opt/airflow/dags --name airflow\_container --entrypoint /bin/bash -ti apache/airflow



-p 8080:8080 – this part expose port 8080 of the Docker Container to port 8080 of the VM, so link <https://localhost:8080> will show us page from Docker Image.

-p 22:22 – this part expose the port 22 of the Docker Container (default for SSH)

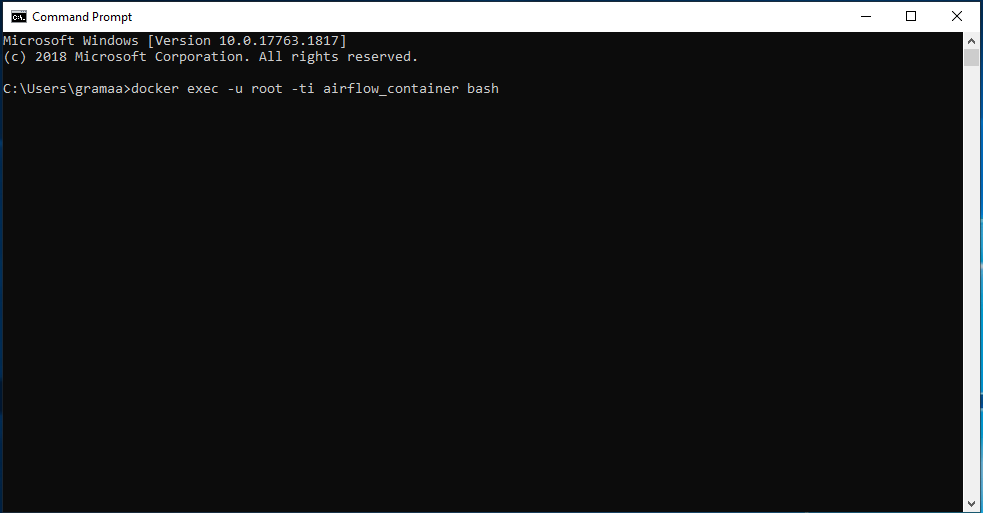
-v C:/Airflow/dags/:/opt/airflow/dags – link folder C:/Airflow/DAGS/ of the VM to the folder /opt/airflow/dags of the Docker Container. All files stored in the VM folder will be automatically copied to the Docker Container.

--name airflow\_container – name of the Docker Container.

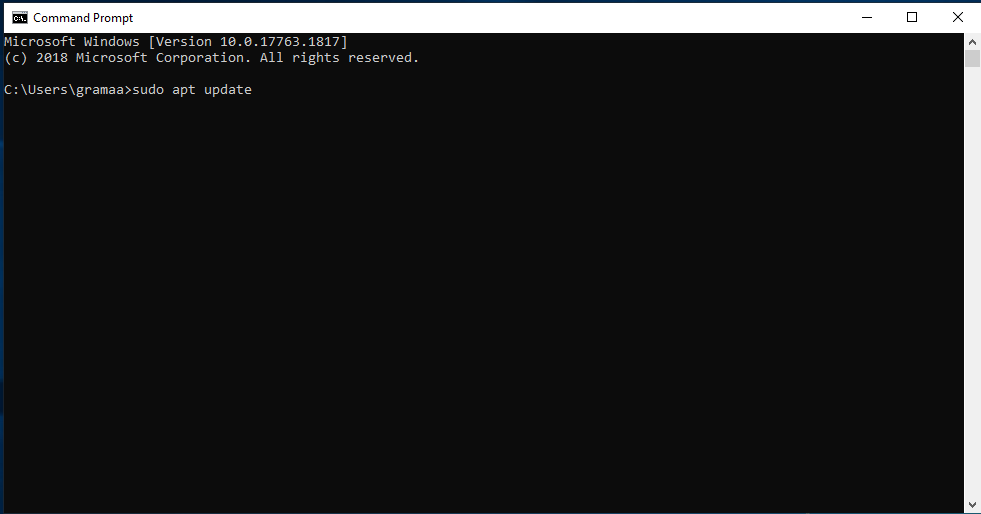
--ti apache/airflow – source Docker Image name

5. Install PostgreSQL on Docker Image

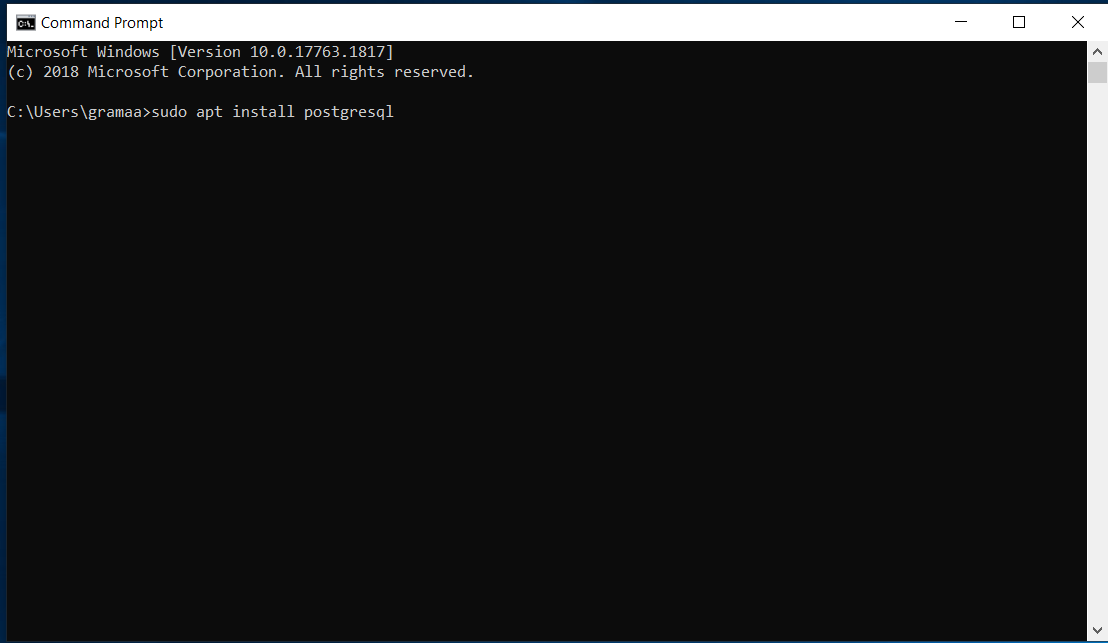
connect to Docker as root user : docker exec -u root -ti airflow\_container bash



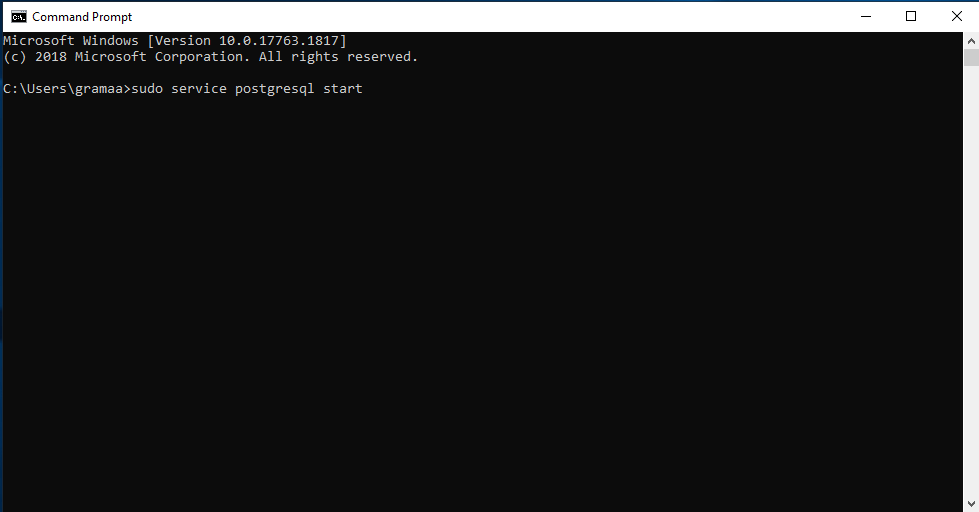
load last updates: sudo apt update



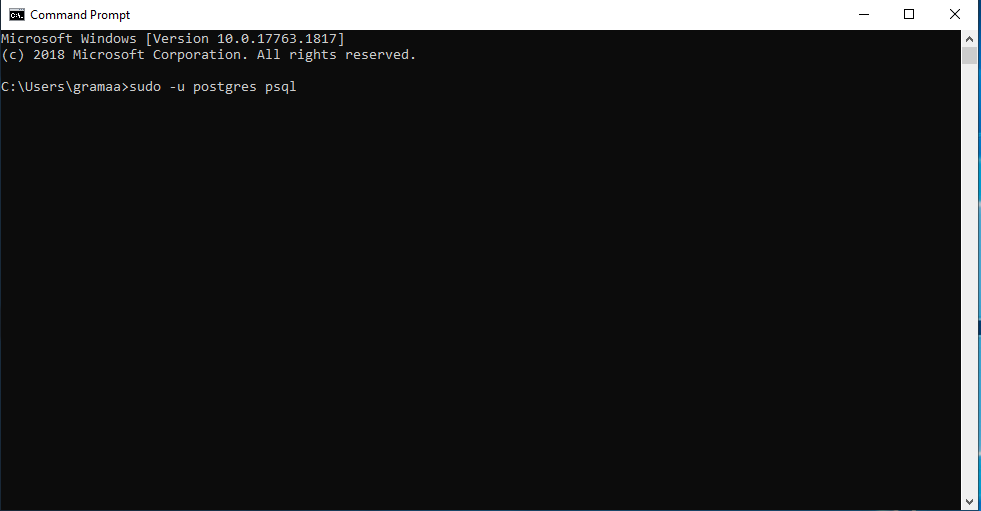
install postgresql : sudo apt install postgresql



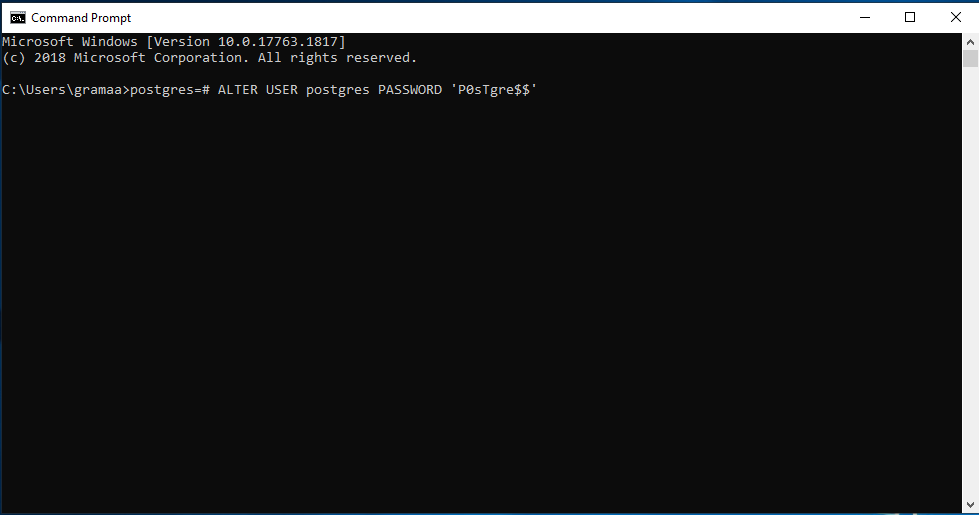
start postgresql instance: sudo service postgresql start



connect to postgresql instance to run SQL command: sudo -u postgres psql

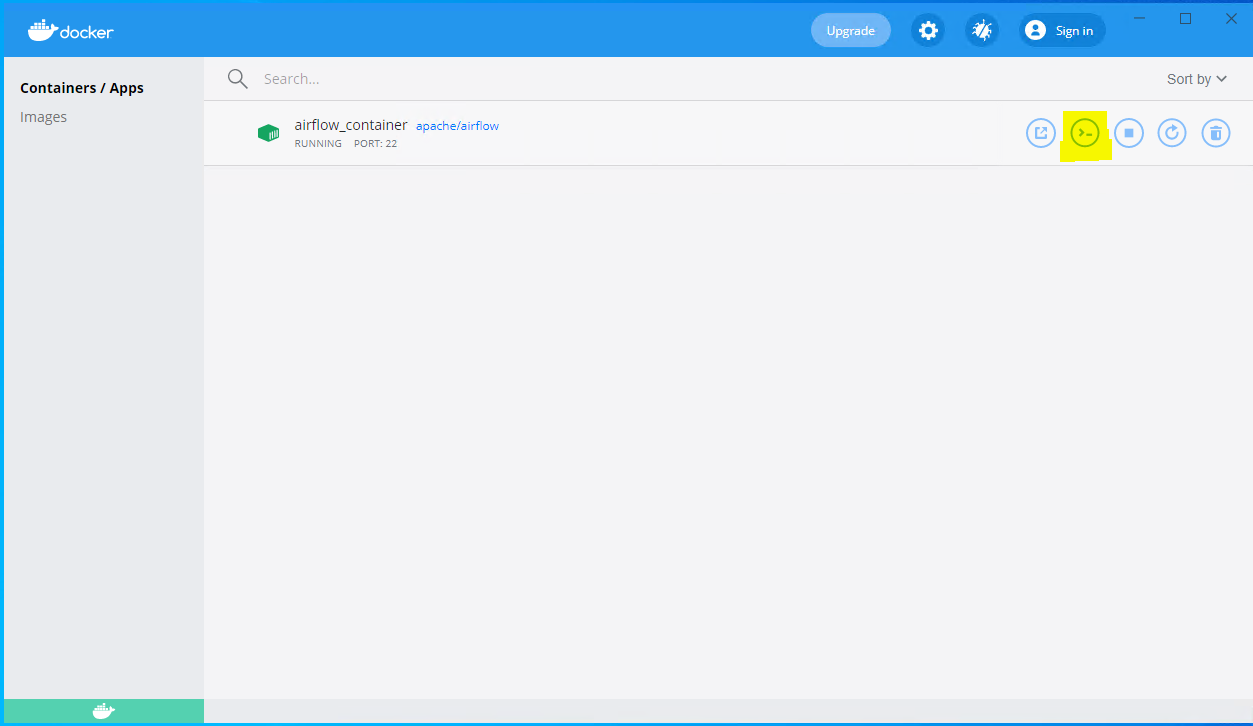


run sql command to change default password: postgres=# ALTER USER postgres PASSWORD 'P0sTgre$$'

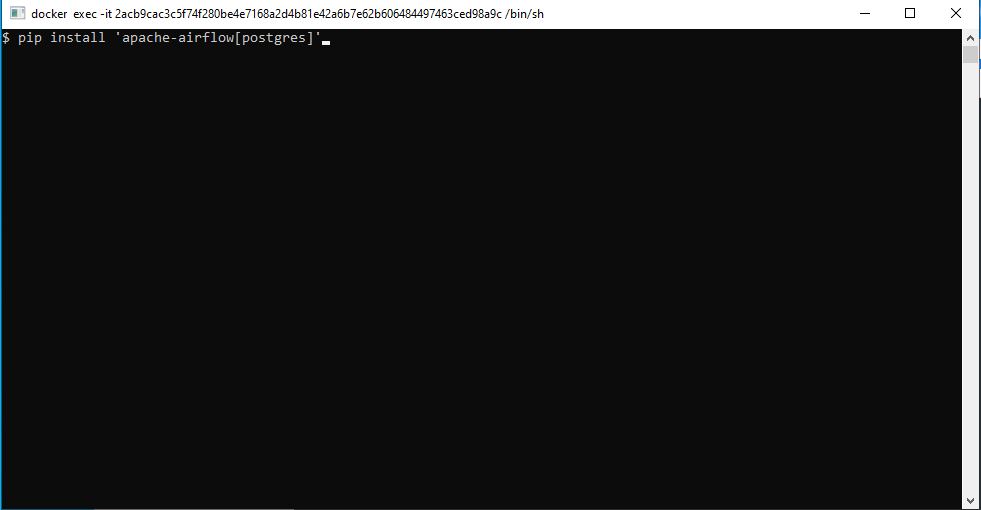


6. Install Airflow libraries for PostgreSQl backend

Run CLI for the Docker Container

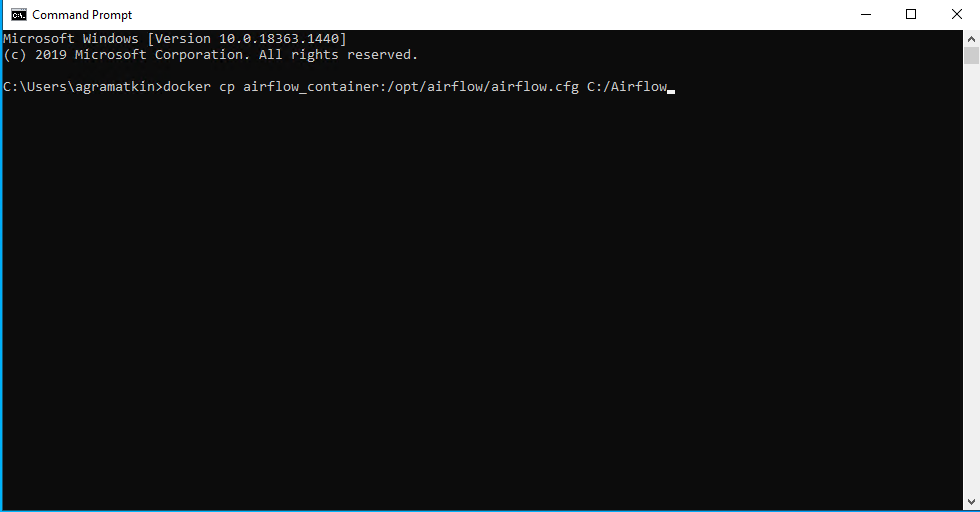


In the CLI window run: pip install 'apache-airflow[postgres]'

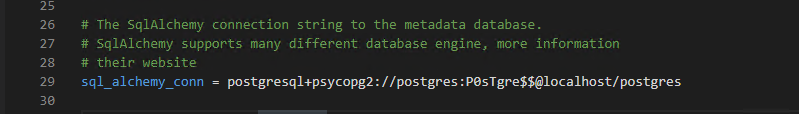


6. Change airflow.cfg to work with Local Executor

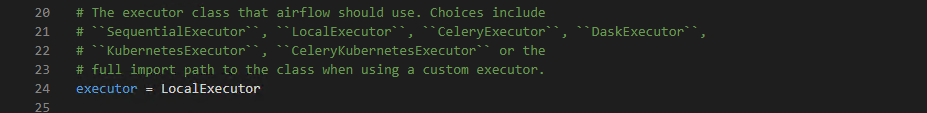
In CMD run (to copy airflow.cfg from the Docker Container): docker cp airflow\_container:/opt/airflow/airflow.cfg C:/Airflow



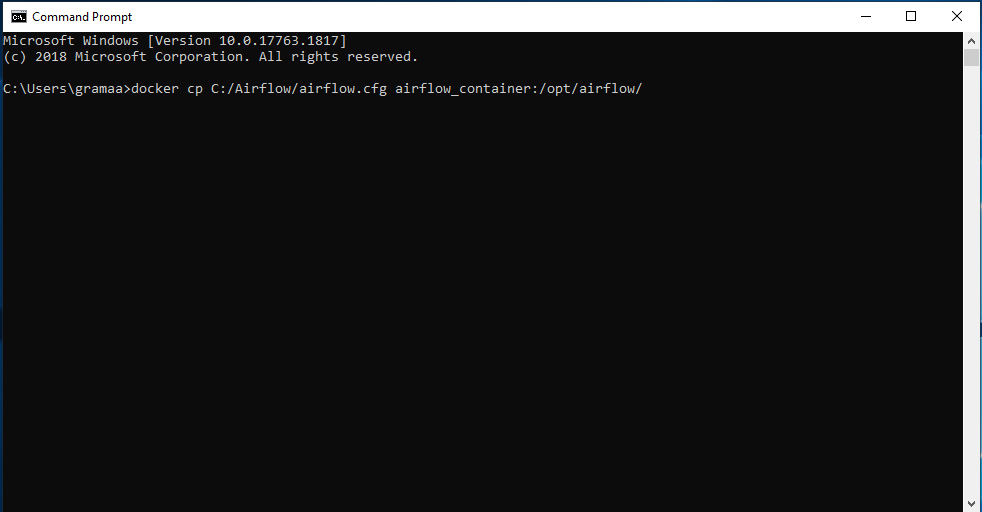
Open C:/Airflow/airflow.cfg and replace value for the key "sql\_alchemy\_conn" with "postgresql+psycopg2://postgres:P0sTgre$$@localhost/postgres"



and value for the key "executor" to "LocalExecutor"

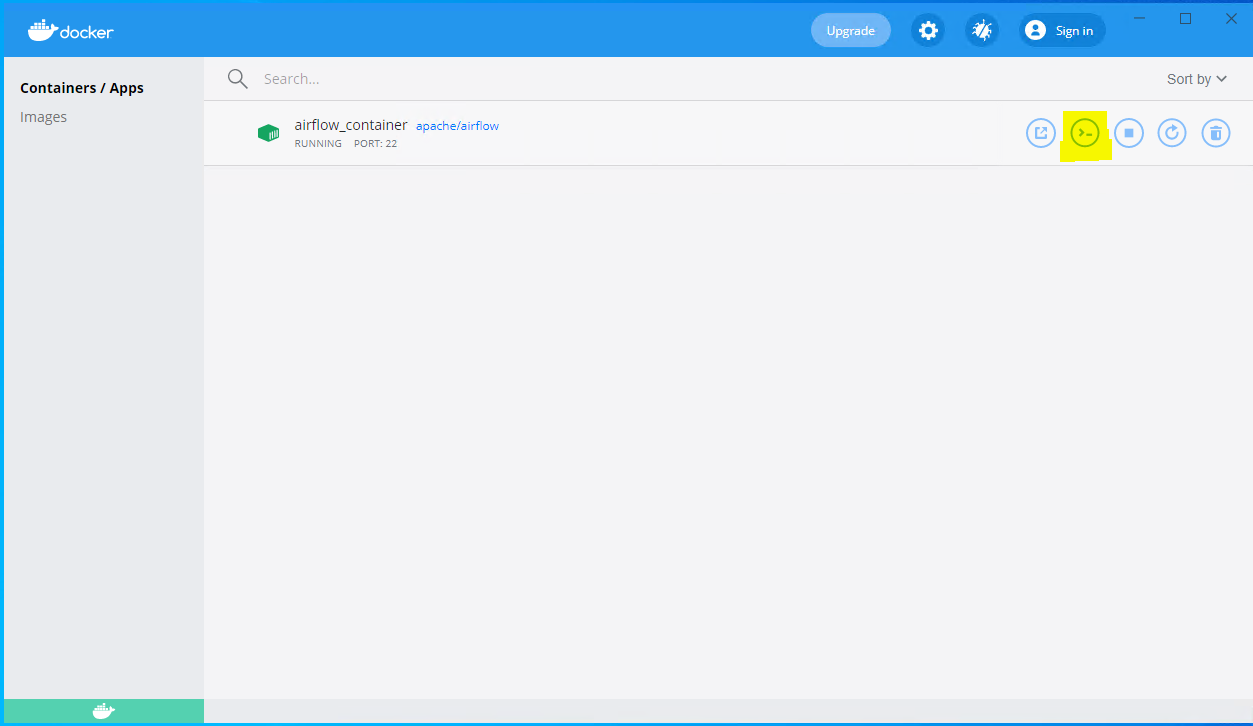


In CMD run (to copy airflow.cfg to the Docker Container): docker cp C:/Airflow/airflow.cfg airflow\_container:/opt/airflow/

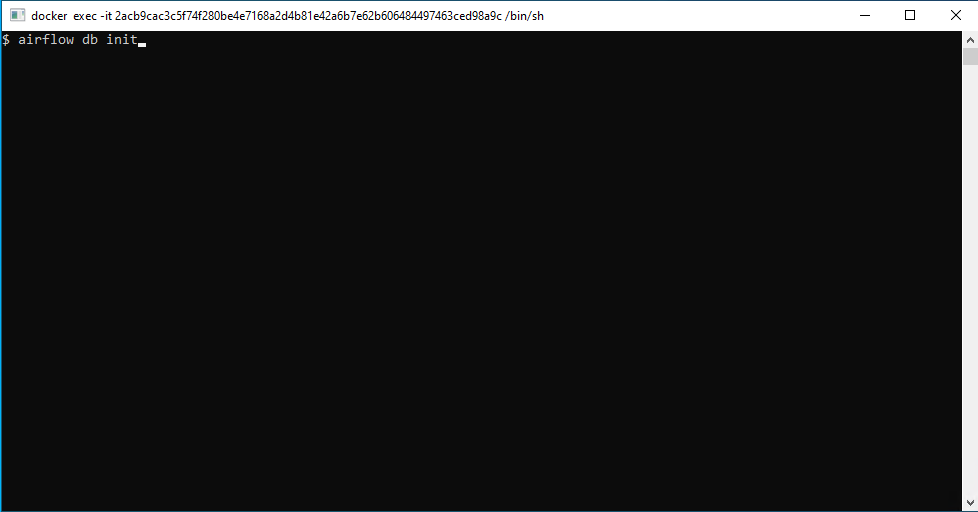


7. Initialize Airflow DB

Run CLI for the Docker Container



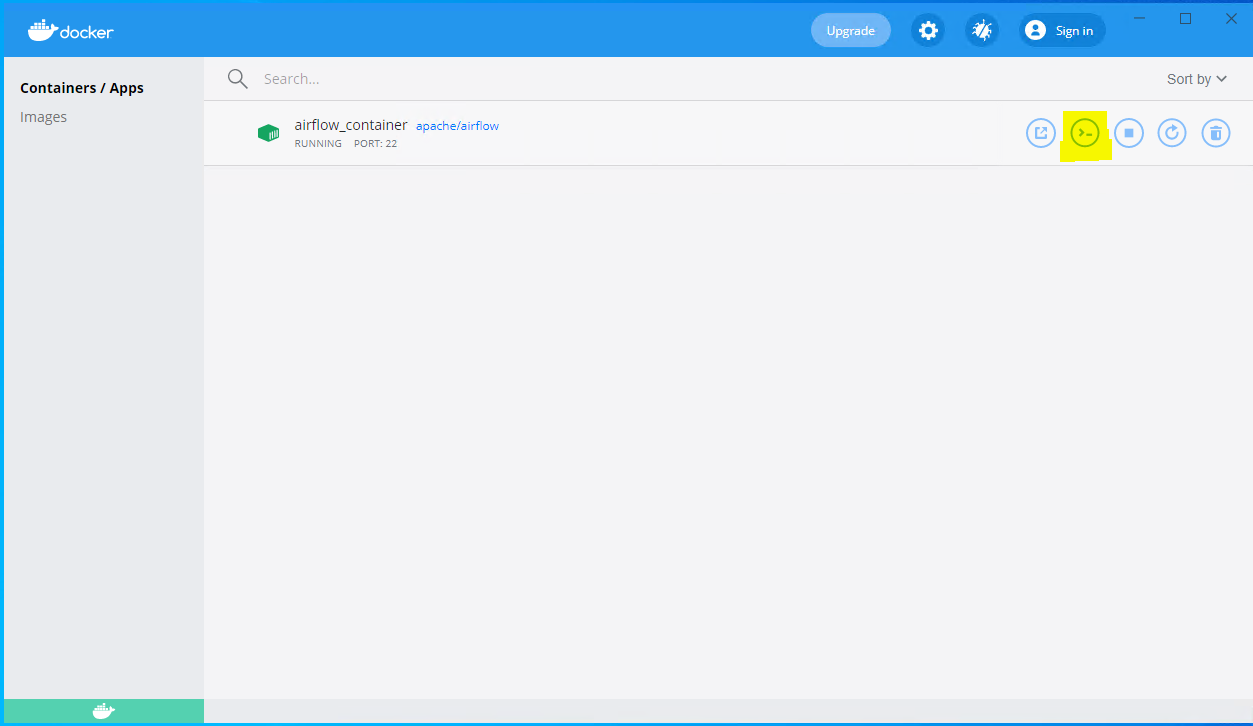
In the CLI window run: airflow db init



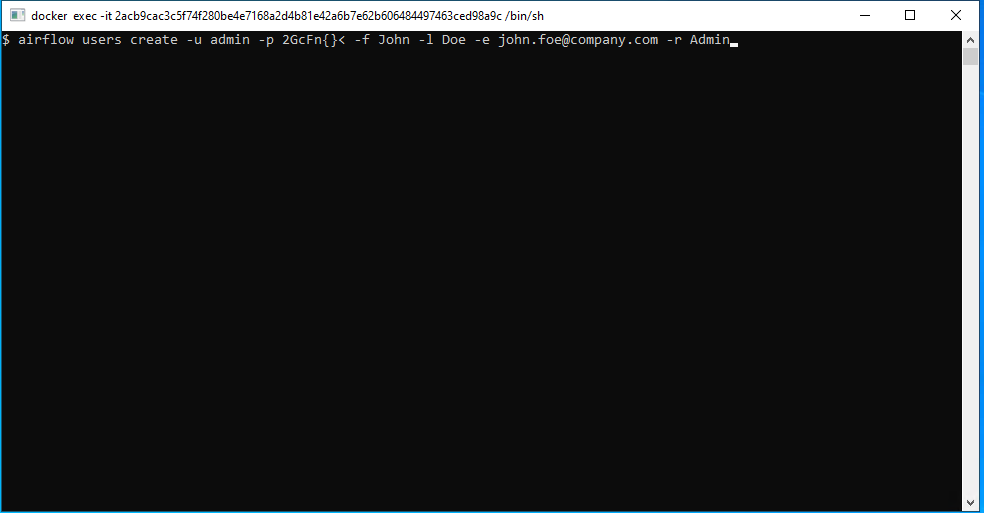
Close the CLI window

8. Create Airflow user:

Run CLI for the Docker Container



In the CLI window run: airflow users create -u admin -p Str0nGadMinPaS$w0rd\_743 -f <user\_first\_name> -l <user\_last\_name> -e <user\_email> -r Admin



-u – username, must be unique

-p – password

-f – user first name

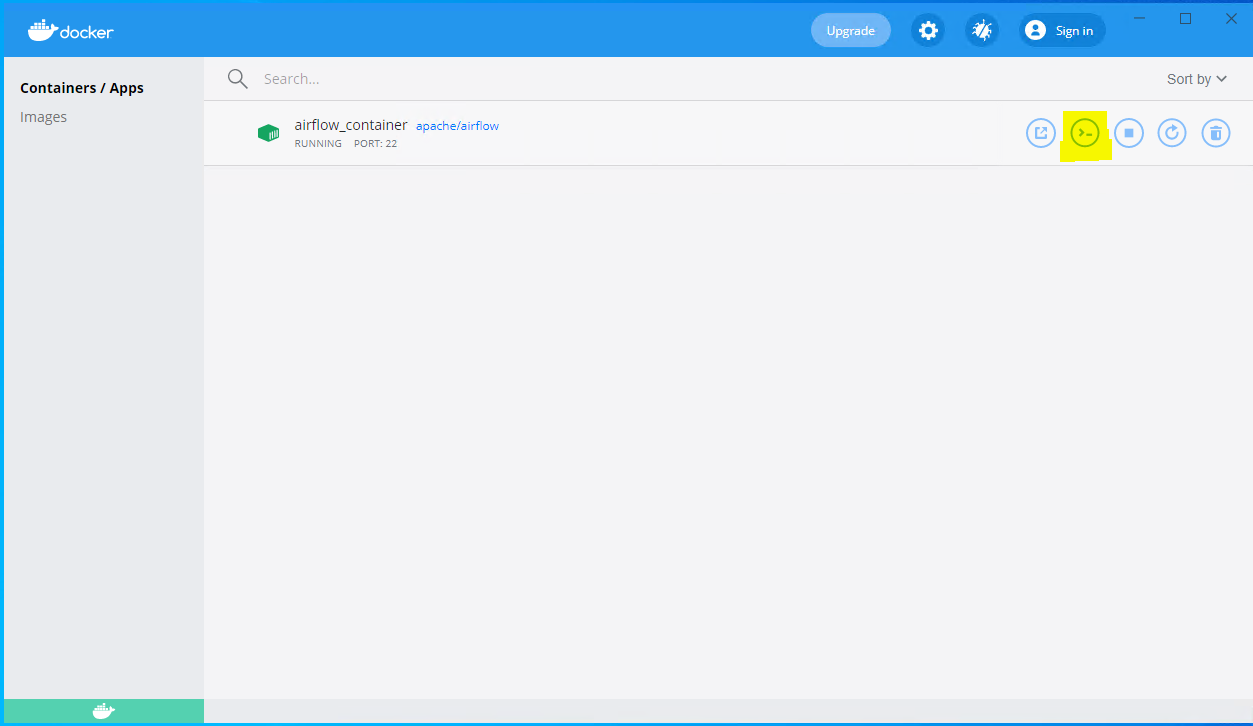
-l – user last name

-e – user email

-r – role, in our case role is Admin

9. Run Airflow webserver

Run CLI for the Docker Container



In the CLI window run: airflow webserver

10. Run Airflow Scheduler: airflow scheduler

11. Install Apache Airflow Snowflake connector: pip install apache-airflow-providers-snowflake