

Ex No: 6	Apache Airflow
Date: 15/10/25	

Objective:

Apache Airflow is an open-source platform created by Airbnb (now part of the Apache Software Foundation) for programmatically authoring, scheduling, and monitoring workflows. It is widely used in data engineering and ETL (Extract, Transform, Load) pipelines.

Outcomes:

1. Connecting to Airflow using Docker compose.
2. Installing Apache Airflow.
3. Running the DAGs(postgres) in Airflow.

Materials:

- ENV files (.env)
- Csv file(users.csv)
- Yaml source file (docker-compose) and dockerfile.
- Python source file like dagselt_pipeline_postgres

Lab Procedure:

Stage 1: Install Apache Airflow for windows PC

1. Open the command prompt and navigate to your lab folder (cd "c:\Users\< >\lab6")
2. Install the docker and docker compose by commands :

```
sudo apt-get update
```

```
sudo apt-get install docker.io docker-compose -y
```

3. This builds a fresh image using your Dockerfile.

```
docker build -t airflowsqlserver -f Dockerfile --no-cache .
```

```
docker-compose up
```

Name: Meghana G
USN: 1RVU23CSE264

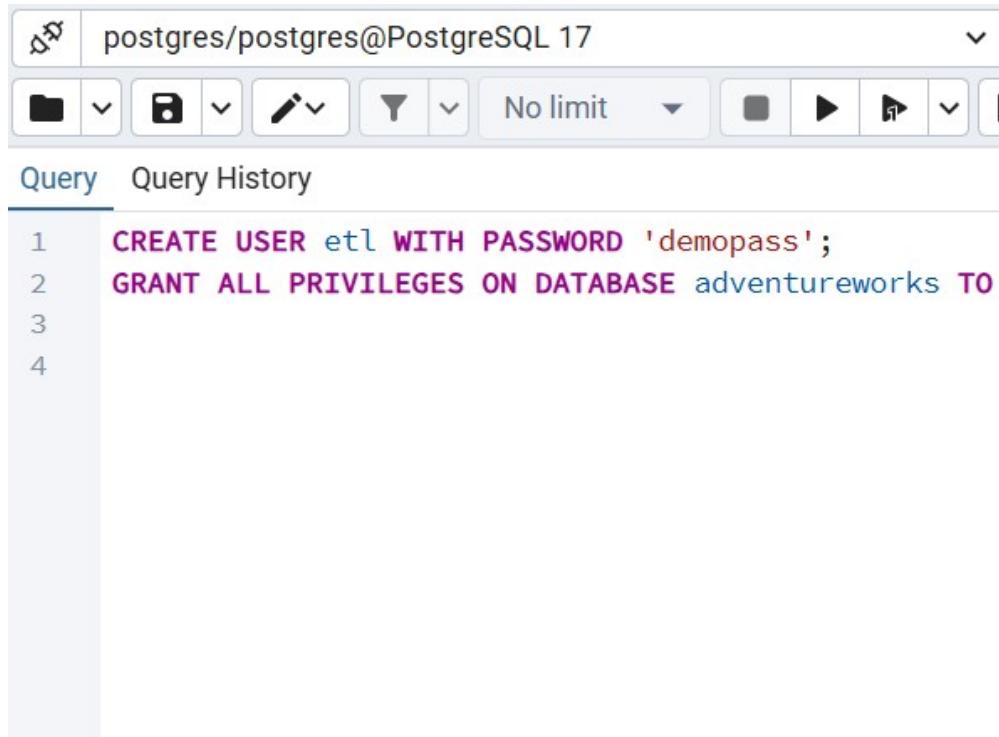
docker ps

- You should see containers for Airflow, Postgres, etc.
- Open a browser → Go to <http://localhost:8080> (usually Airflow UI).

4. Add this schema to pgadmin4 then save and run

```
CREATE SCHEMA IF NOT EXISTS etl_staging;
```

```
GRANT ALL PRIVILEGES ON SCHEMA etl_staging TO etl;
```



```
1 CREATE USER etl WITH PASSWORD 'demopass';
2 GRANT ALL PRIVILEGES ON DATABASE adventureworks TO
3
4
```

Stage 2: Access to the Airflow

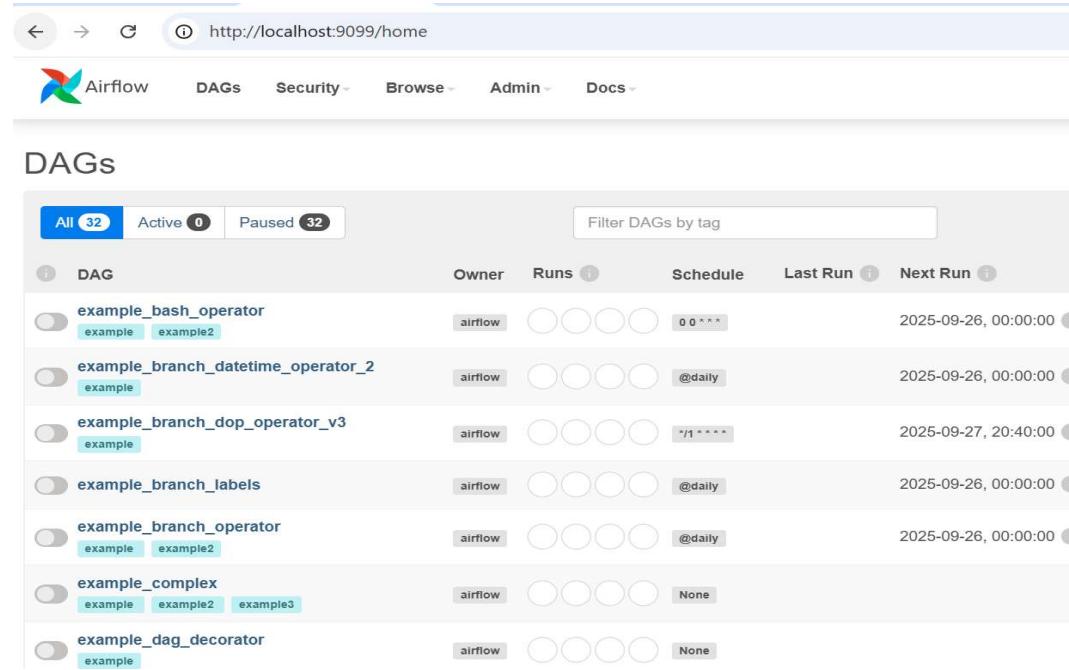
1. Open your browser:
<http://localhost:8080>
2. Default Airflow login (unless changed in .env):

username: airflow

password: airflow

3. Verify the DAGs

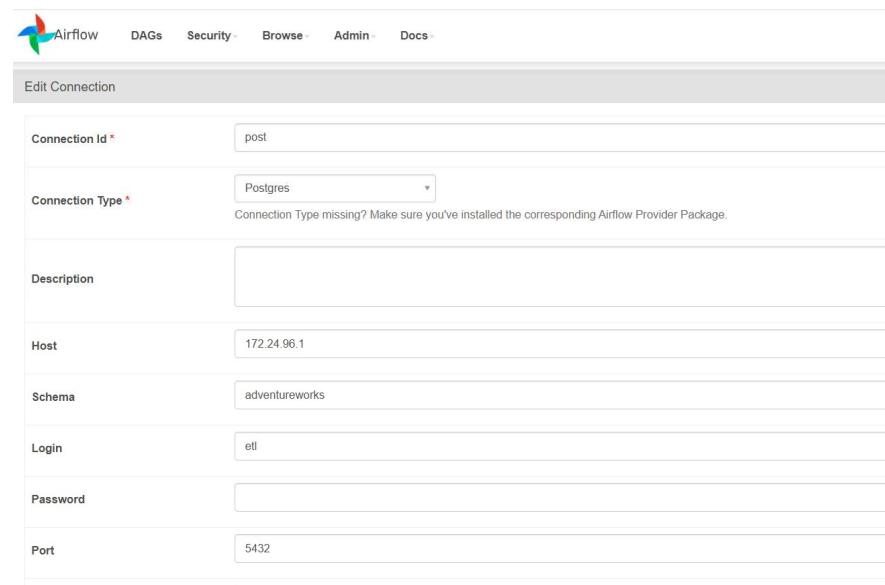
Name: Meghana G
USN: 1RVU23CSE264



The screenshot shows the Airflow UI at <http://localhost:9099/home>. The top navigation bar includes links for Airflow, DAGs, Security, Browse, Admin, and Docs. The main content area is titled "DAGs". At the top of the DAG list, there are three buttons: "All 32", "Active 0", and "Paused 32". A search bar labeled "Filter DAGs by tag" is also present. The table lists seven DAGs:

DAG	Owner	Runs	Schedule	Last Run	Next Run
example_bash_operator	airflow	0 0 * * *		2025-09-26, 00:00:00	
example_branch_datetime_operator_2	airflow	0 0 * * *	@daily	2025-09-26, 00:00:00	
example_branch_dop_operator_v3	airflow	0 0 * * *	*/1 * * * *	2025-09-27, 20:40:00	
example_branch_labels	airflow	0 0 * * *	@daily	2025-09-26, 00:00:00	
example_branch_operator	airflow	0 0 * * *	@daily	2025-09-26, 00:00:00	
example_complex	airflow	0 0 * * *	None		
example_dag_decorator	airflow	0 0 * * *	None		

- In the Airflow UI, go to the **DAGs** tab.
 - You should see your pipeline DAG from `dagselt_pipeline_postgres.py`.
 - Turn the toggle **ON** and trigger it.
4. Go to Edit connection and give host as your pc ip address , login and password save it

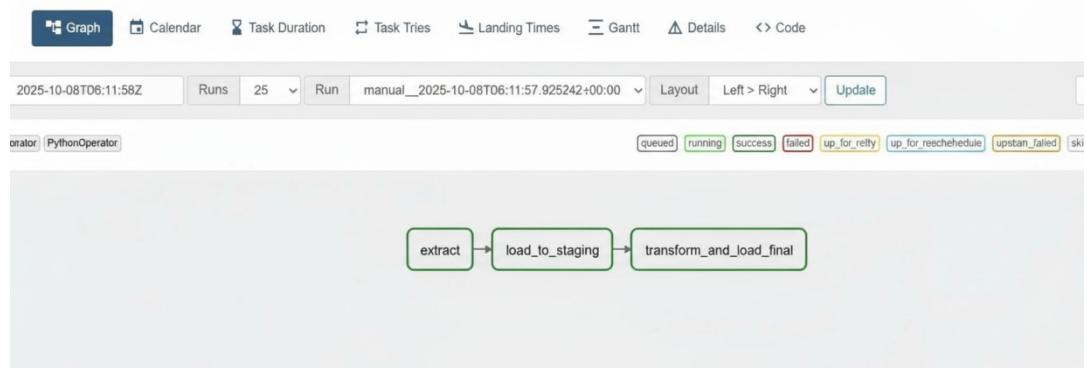


The screenshot shows the "Edit Connection" form in the Airflow UI. The connection type is set to "Postgres". The form fields include:

Connection Id *	post
Connection Type *	Postgres
Description	
Host	172.24.96.1
Schema	adventureworks
Login	etl
Password	[redacted]
Port	5432

5. Run the Dag

Name: Meghana G
USN: 1RVU23CSE264



It shows all green which means all stages run successfully.

Github link : <https://github.com/meghana1653/Data-Engineering/blob/main/Lab-6.zip>