19.8.25 - Task

Airflow - Building a Simple Data Pipeline

1. Project Overview

This project implements a simple data pipeline using Apache Airflow to automate the process of loading employee data into a PostgreSQL database. The pipeline follows these steps:

1. Create necessary tables (staging and final).
2. Download employee data (CSV).
3. Load data into staging table.
4. Clean and upsert data into the final table.

2. Airflow Setup

Steps followed:

1. Created project folder and subfolders:

   dags/ logs/ plugins/

2. Downloaded Airflow Docker Compose file:

   curl -LfO 'https://airflow.apache.org/docs/apache-airflow/stable/docker-compose.yaml'

3. Created .env file:

   echo -e "AIRFLOW_UID=$(id -u)" > .env

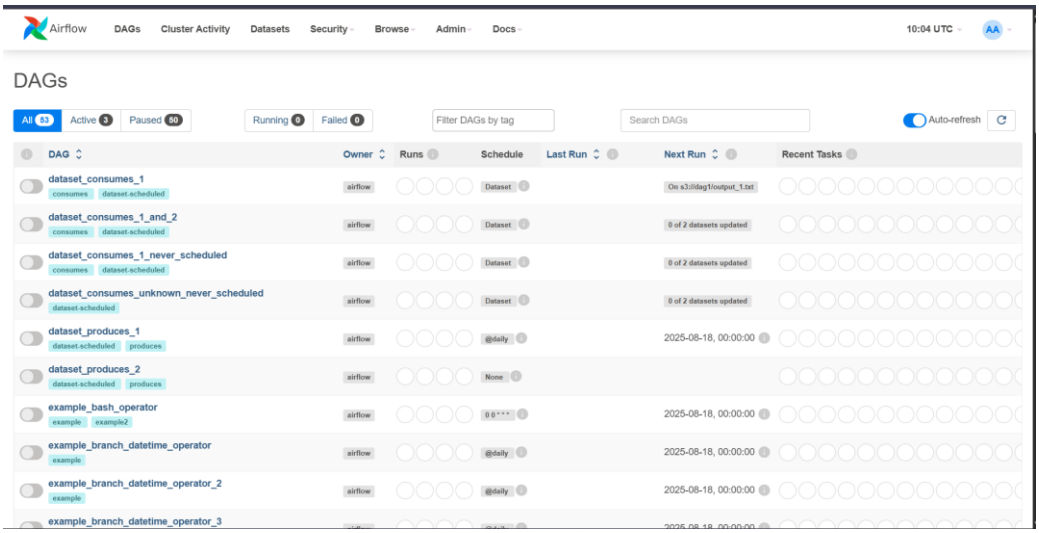
4. Initialized Airflow database:

   docker-compose up airflow-init

5. Started Airflow services:

   docker-compose up

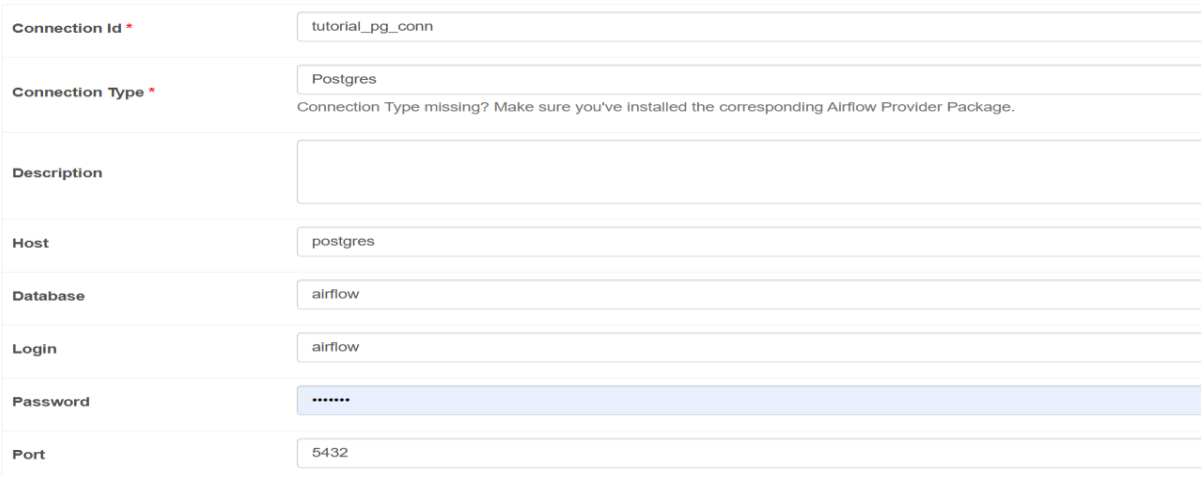
6. Accessed Airflow UI at http://localhost:8080



3.PostgreSQL Connection

Connection setup in Airflow UI:

Field	Value
Connection ID	tutorial_pg_conn
Connection Type	Postgres
Host	postgres
Schema	airflow
Login	airflow
Password	airflow
Port	5432



4.DAG Overview

DAG Name: process_employees
Schedule: Daily
Tasks:

1. create_employees_tables – Creates staging and final tables.
2. download_csv – Downloads employee data as CSV.
3. load_staging – Loads CSV data into staging table.
4. upsert_final – Cleans and upserts data into the final table.

DAG Graph:

create_employees_tables → download_csv → load_staging → upsert_final

5. Python & SQL Code

Create Tables

```
DROP TABLE IF EXISTS employees_temp;

CREATE TABLE employees_temp (

    "Serial Number" NUMERIC PRIMARY KEY,

    "Company Name" TEXT,

    "Employee Markme" TEXT,

    "Description" TEXT,

    "Leave" INTEGER

);


DROP TABLE IF EXISTS employees;

CREATE TABLE employees (
```

```
    "Serial Number" NUMERIC PRIMARY KEY,
    "Company Name" TEXT,
    "Employee Markme" TEXT,
    "Description" TEXT,
    "Leave" INTEGER
);
```

Python CSV Download

```
def save_csv_to_local(**kwargs):
    import csv
    data = [
        [1, 'ABC Corp', 'E101', 'Engineer', 2],
        [2, 'XYZ Ltd', 'E102', 'Manager', 0],
        [3, 'TechSoft', 'E103', 'Analyst', 1]
    ]
    file_path = '/tmp/employees.csv'
    with open(file_path, 'w', newline='') as f:
        writer = csv.writer(f)
        writer.writerow(["Serial Number","Company Name","Employee Markme","Description","Leave"])
        writer.writerows(data)
    return file_path
```

Load Data into Staging

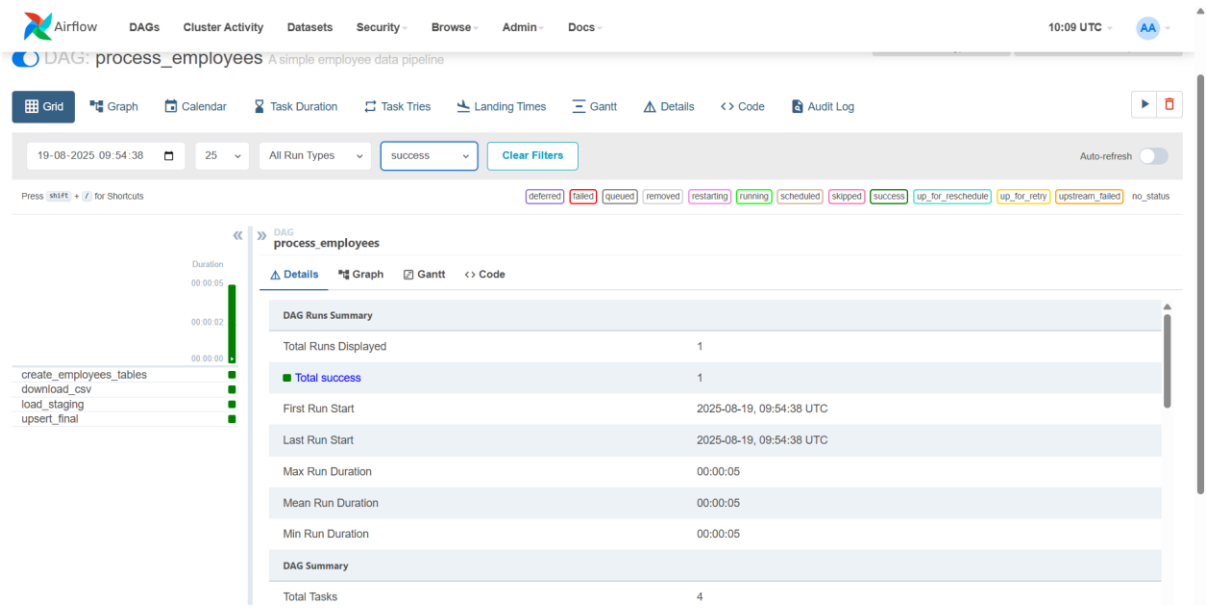
```
import psycopg2

file_path = '/tmp/employees.csv'

conn = psycopg2.connect(host="postgres", dbname="airflow",
user="airflow", password="airflow", port=5432)

cur = conn.cursor()

with open(file_path, 'r') as f:

    next(f)

    cur.copy_from(f, 'employees_temp', sep=',')

conn.commit()

cur.close()

conn.close()
```

Upsert to Final Table

```
DELETE FROM employees

WHERE "Serial Number" IN (SELECT "Serial Number" FROM
employees_temp);



INSERT INTO employees ("Serial Number", "Company Name",
"Employee Markme", "Description", "Leave")

SELECT "Serial Number", "Company Name", "Employee Markme",
"Description", "Leave"

FROM employees_temp;
```

7. Execution & Results

- Triggered the DAG in Airflow UI.
- All tasks completed successfully:
  - create_employees_tables
  - download_csv
  - load_staging
  - upsert_final



- Verified data in PostgreSQL:

SELECT * FROM employees;

```
PS C:\Users\harci\Materilas> docker-compose exec postgres psql -U airflow -d airflow
>>
time="2025-08-19T15:25:59+05:30" level=warning msg="C:\\Users\\harci\\Materilas\\docker
-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove i
t to avoid potential confusion"
psql (13.22 (Debian 13.22-1.pgdg13+1))
Type "help" for help.

airflow=# SELECT * FROM employees;
 Serial Number | Company Name | Employee Markme | Description | Leave
---------------+--------------+-----------------+-------------+-------
             1 | ABC Corp     | E101            | Engineer    |     2
             2 | XYZ Ltd      | E102            | Manager     |     0
             3 | TechSoft     | E103            | Analyst     |     1
(3 rows)
```