**Project statement:**

In this project you are required to write 2 resilient simple independent data integration (ETL and ELT) pipelines that aggregates data from multiple sales csv files.

**What is Data Integration?**

Data integration means combining information from different places to make it all work together. This can be done by a person or by a computer system.

**What is Data Integration Pipeline?**

A data integration pipeline is a structured way of transforming raw data from one or more sources into a format that's ready to be used or presented. This could mean creating a report, updating a database, or preparing data for some other type of display. Usually, the pipeline involves different stages for cleaning up and organizing the data before it's ready to be stored or analyzed in a database.

**Step 1:**

In this project, our task was to develop a database schema and link it with our script. For this purpose, I opted to utilize the RDBMS PostgreSQL and interact with the database using a user-friendly interface like pgAdmin. This interface facilitates the display of schemas and relationships within the tables, streamlining the management and visualization of our database structure.

Installation of Python

A computer screen with text on it

Description automatically generated

I used Python to read data from the CSV files. The pandas library is very efficient for this task.

The code snippet you provided shows the beginning of setting up an ETL (Extract, Transform, Load) pipeline using Python with pandas for data manipulation and psycopg2 for database interaction.

**Extract**: The pandas library is used to extract data from a CSV file. This is done with the pd.read\_csv() function, which reads the data from the CSV file into a pandas DataFrame. This is the "Extract" step of your ETL pipeline. Two transformations for the ETL process, focusing on handling date formats and ensuring numeric monetary values are correctly formatted and converted.

A screen shot of a computer program

Description automatically generated

**Loading :** Loading the transformed data into a PostgreSQL database using psycopg2. This function allows efficient bulk insertion of data into a PostgreSQL database. Using psycopg2.connect() with the provided database parameters, And then use execute\_values to insert the transformed DataFrame into the database table.

A computer screen shot of a black screen

Description automatically generated

**Step 2:**

For windows, I scheduled my Python script using Windows Task Scheduler, it enables automating various tasks and processes, making it easier to manage repetitive tasks and ensuring timely execution of your Python code.

A screenshot of a computer

Description automatically generated

In order to ensure the run of the script, I added this lien to create a text file, run , in order to know the date and time the script run

A screen shot of a computer

Description automatically generated

After the run of the script and the execution of data table in Postgres, the run text file was generated too after running the python code.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Step 3**

I used docker to run airflow: Apache Airflow is an open-source platform to programmatically author, schedule, and monitor workflows.

CREATING airflow containers

A screen shot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

List of containers:

A screenshot of a computer

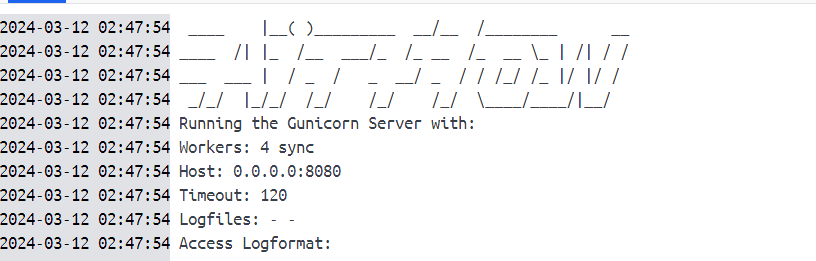
Description automatically generated

Open docker

A screenshot of a computer

Description automatically generated

All airflow containers are running



Airflow is running in docker

Then we open airflow webserver: Airflow web UI using the specified port (typically port 8080 by default).

A screenshot of a computer

Description automatically generated

Then I sign in by creating an account of user name admin and username admin

A computer screen with text

Description automatically generated

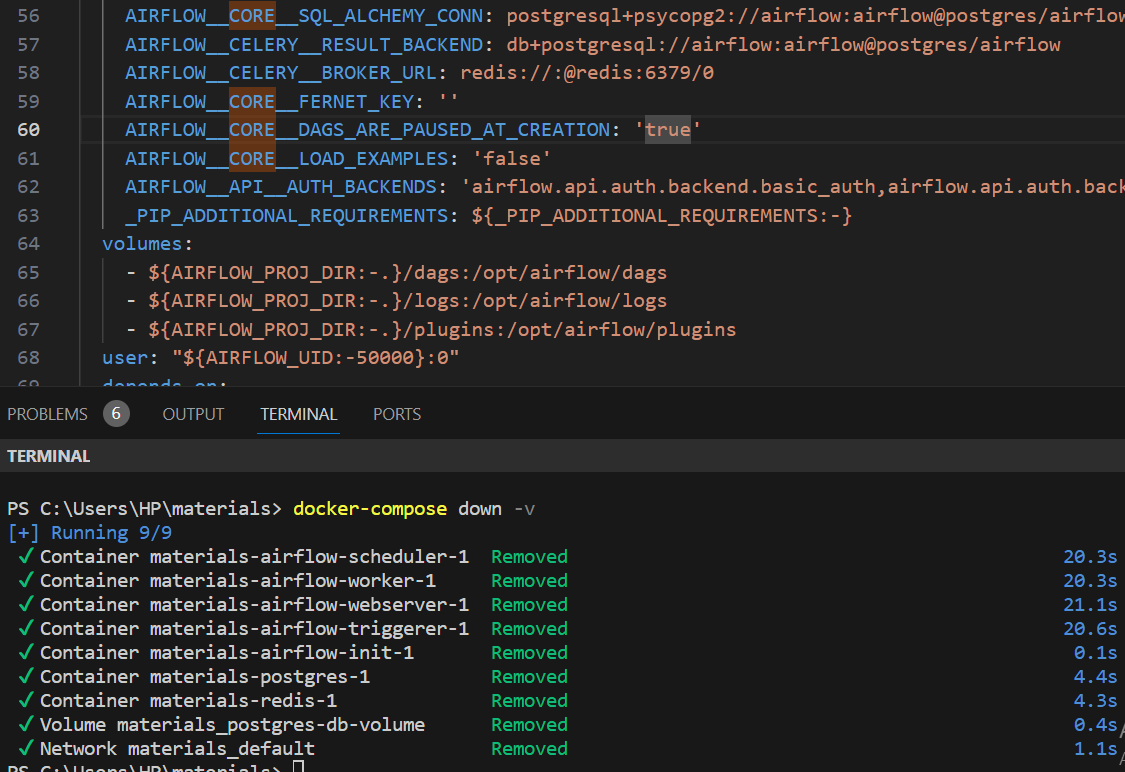
Then we login

A screenshot of a computer

Description automatically generated

DAGs (Directed Acyclic Graphs): Workflows in Airflow are defined as DAGs. A DAG is a collection of all the tasks you want to run, organized in a way that reflects their relationships and dependencies.

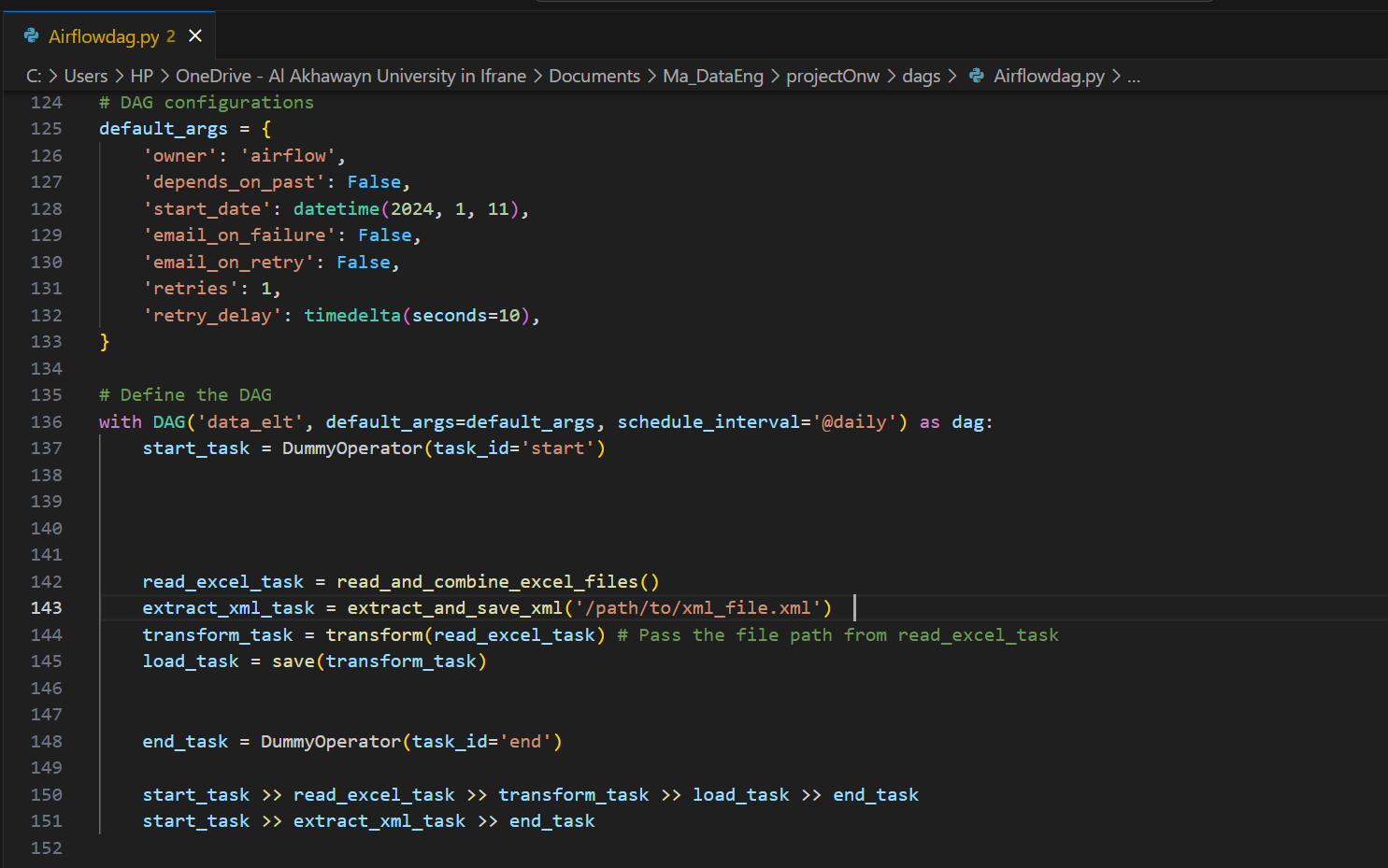
Then I remove all example dags



A screenshot of a computer

Description automatically generated

I run arirflow.py in Dags folder , to run The ETL tasks from step 1 and 2 and create the new Dag in airflow



Data\_elt Dag was created

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

For ELT part , I faced a problem in which I invested a lot of solve, but is still unsolved. You can check this link <https://youtu.be/etwZhnZZ3Mo> for more information

Links:

Step1 : <https://youtu.be/CzZuvK_pCeI>

Step 2 : <https://youtu.be/OIdZ7UnATzs>