

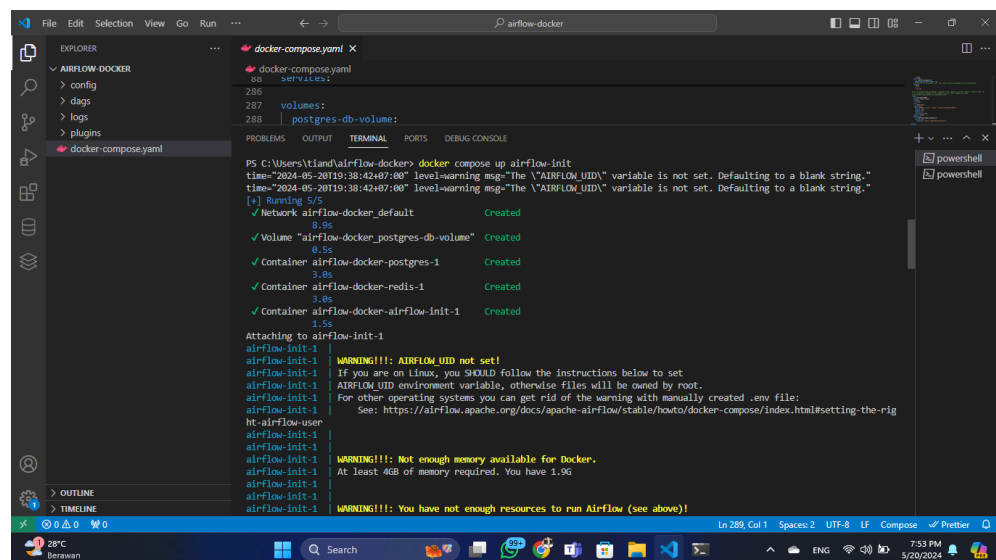
New Assignment - Airflow Connection & DAGs Running

Kelompok 12 :

1. Khidlira Arofat Nuciferadha
2. Ecclesiani
3. Riski Syahputra
4. Yuni Dwi Ariyanti
5. Tiandra Rizky Pramadhani

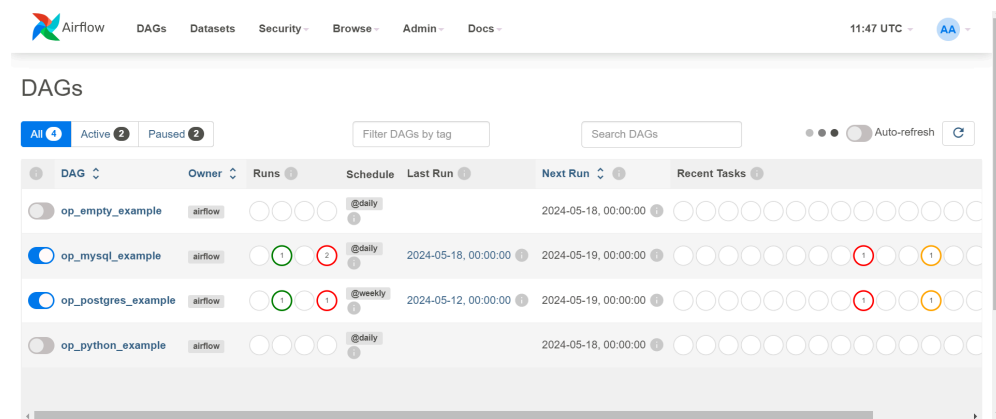
1. Jalankan docker apache airflow kemudian lampirkan bukti:

a. Screenshot terminal



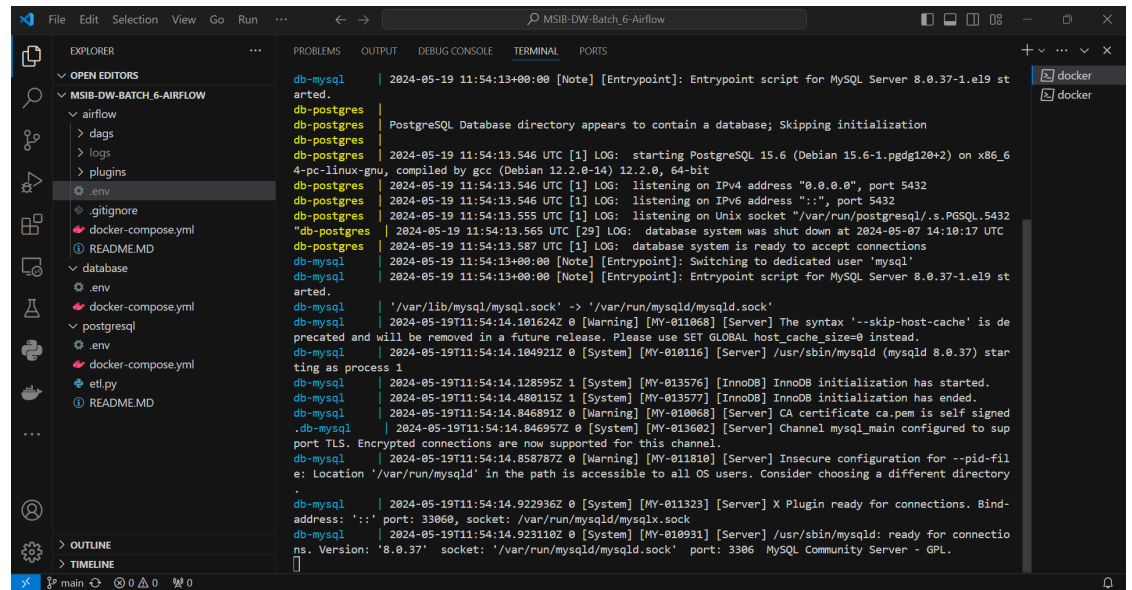
```
PS C:\Users\tiandra\airflow-docker> docker compose up airflow-init
time="2024-05-20T19:38:42+07:00" level=warning msg="The \"AIRFLOW_UID\" variable is not set. Defaulting to a blank string."
time="2024-05-20T19:38:42+07:00" level=warning msg="The \"AIRFLOW_UID\" variable is not set. Defaulting to a blank string."
[+] Running 5/5
✔ Network airflow-docker_default Created
✔ Volume "airflow-docker_postgres-db-volume" Created
✔ Container airflow-docker-postgres-1 Created
✔ Container airflow-docker-redis-1 Created
✔ Container airflow-docker-airflow-init-1 Created
Attaching to airflow-init-1
airflow-init-1 | WARNING!!!: AIRFLOW_UID not set!
airflow-init-1 | If you are on Linux, you SHOULD follow the instructions below to set
airflow-init-1 | AIRFLOW_UID environment variable, otherwise files will be owned by root.
airflow-init-1 | For other operating systems you can get rid of the warning by manually created .env file:
airflow-init-1 | See: https://airflow.apache.org/docs/apache-airflow/stable/howto/docker-compose/index.html#setting-the-rig
ht-airflow-user
airflow-init-1 |
airflow-init-1 | WARNING!!!: Not enough memory available for Docker.
airflow-init-1 | At least 4GB of memory required. You have 1.9GB
airflow-init-1 |
airflow-init-1 | WARNING!!!: You have not enough resources to run Airflow (see above)!
```

b. Screenshot localhost:8080

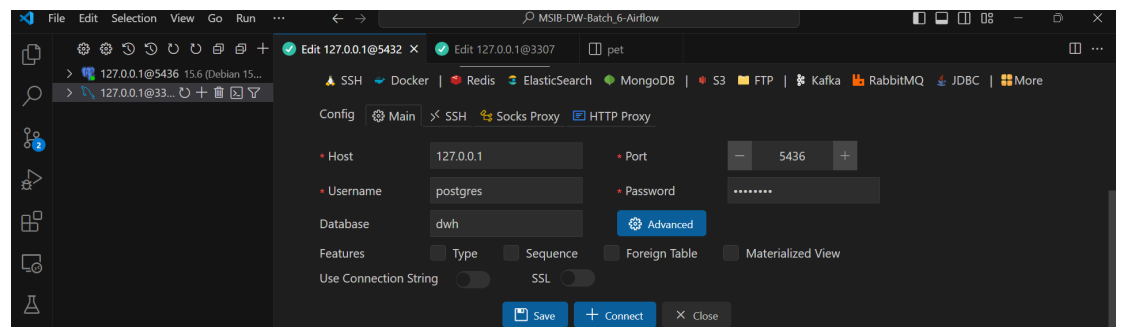


2. Jalankan docker database mysql dan postgresql kemudian lampirkan bukti:

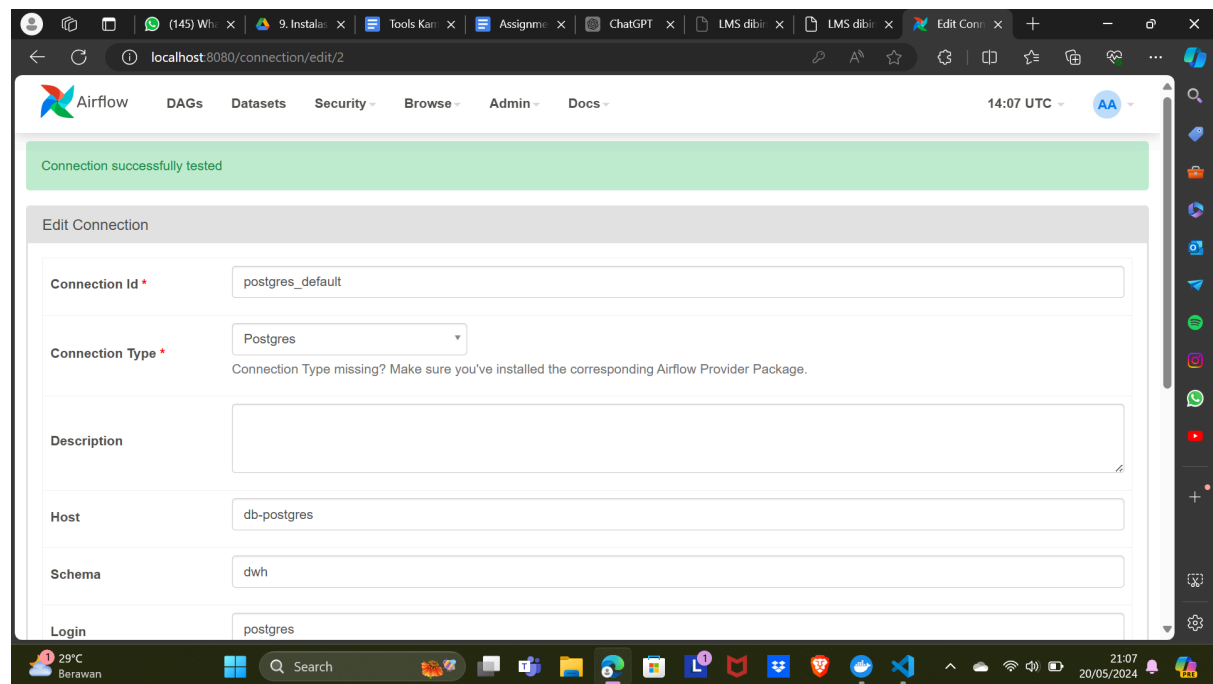
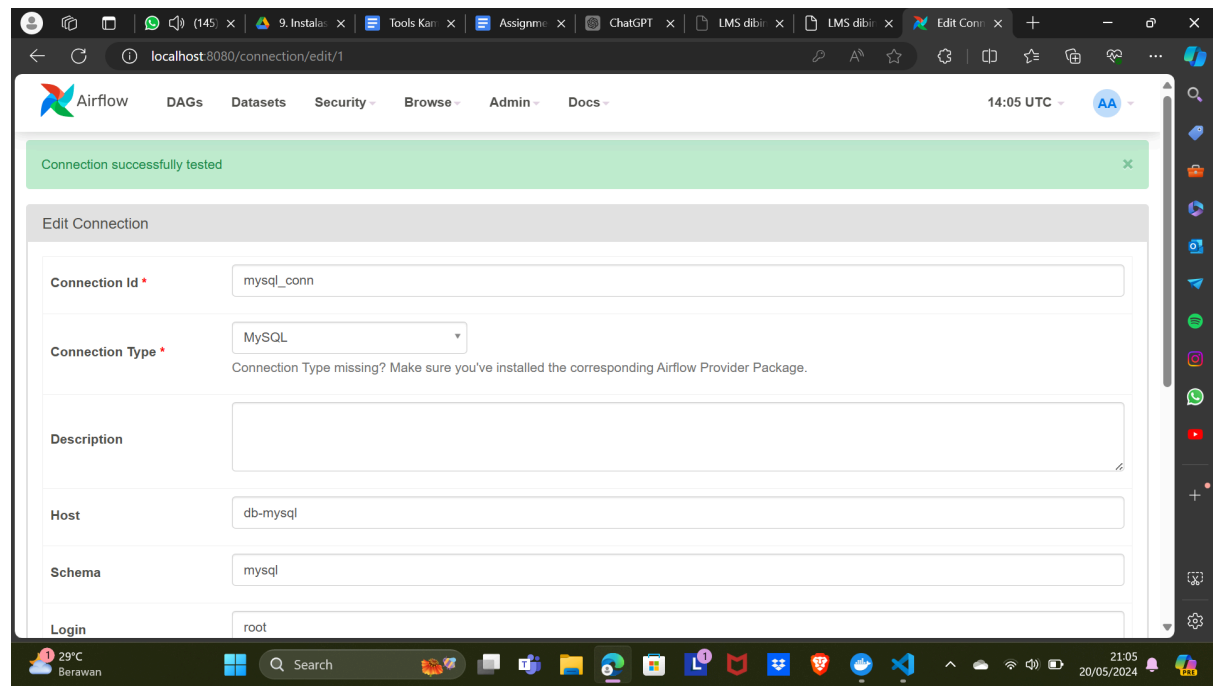
a. Screenshoot terminal



- b. Screenshoot Dbeaver / MySQL Connection Extension yang telah terhubung dengan postgresql dan mysql

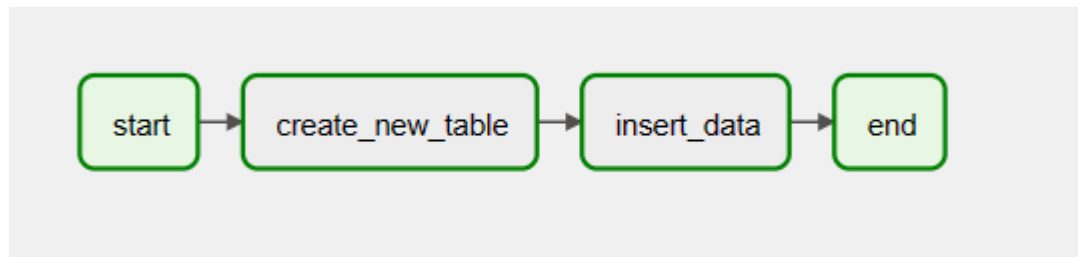


3. Buat connection MySQL dan PostgreSQL pada Apache Airflow kemudian
- a. Screenshoot UI hasil testing yang menunjukkan connection successful



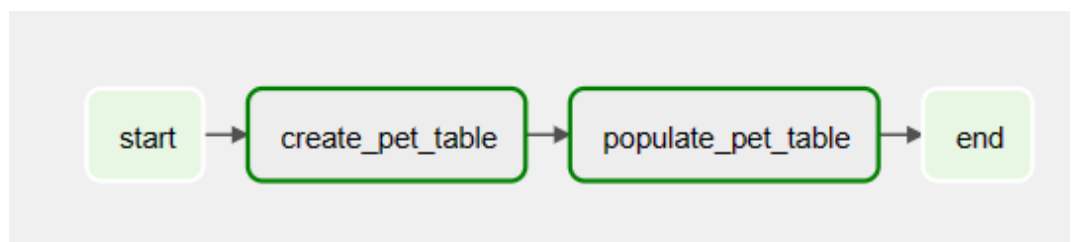
4. Modifikasi DAGs `op_mysql` dan `op_postgresql` yang sudah ada dengan menambahkan `EmptyOperator` sehingga tampilan kedua DAG tersebut menjadi sebagai berikut:

a. op_mysql :



```
op_postgresql.py 2, M  op_mysql.py 2, M X
airflow > dags > op_mysql.py > ...
1 from datetime import datetime
2 from airflow import DAG
3 from airflow.providers.mysql.operators.mysql import MysqlOperator
4 from airflow.operators.empty import EmptyOperator
5
6 with DAG(
7     dag_id='op_mysql_example',
8     start_date=datetime(2024, 5, 1),
9     catchup=False,
10    schedule_interval="@daily"
11 ) as dag:
12
13     start = EmptyOperator(
14         task_id='start'
15     )
16
17     create_new_table = MysqlOperator(
18         task_id='create_new_table',
19         mysql_conn_id='mysql_conn',
20         sql="CREATE TABLE IF NOT EXISTS students (student_id INT AUTO_INCREMENT PRIMARY KEY, name VARCHAR(256));",
21     )
22
23     insert_data = MysqlOperator(
24         task_id='insert_data',
25         mysql_conn_id='mysql_conn',
26         sql="INSERT INTO students (name) VALUES ('Alex Jones');",
27     )
28
29     end = EmptyOperator(
30         task_id='end'
31     )
32
33     start >> create_new_table >> insert_data >> end
34
35
```

b. op_postgresql



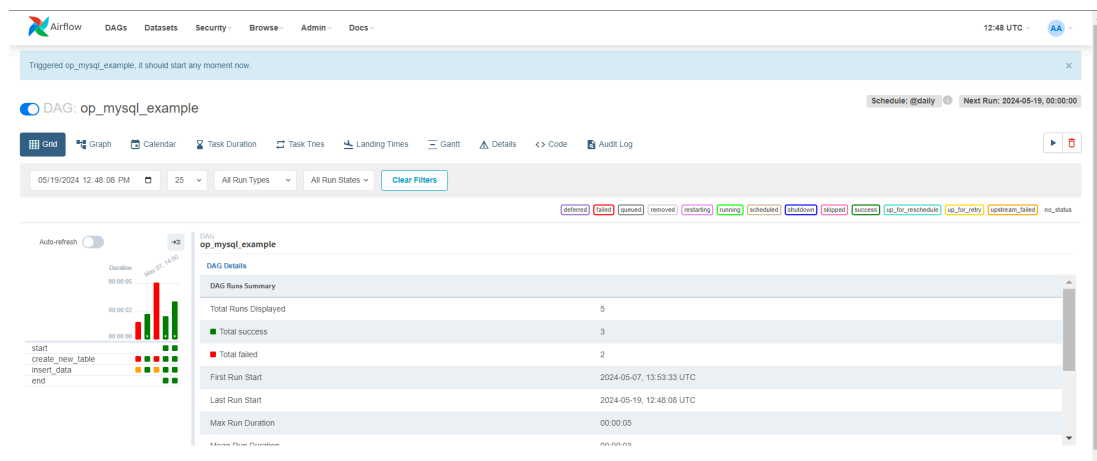
```

5
6
7 with DAG(
8     dag_id='op_postgres_example',
9     start_date=datetime.datetime(2024, 5, 1),
10    catchup=False,
11    schedule_interval="@weekly" # Set your desired schedule interval
12 ) as dag:
13
14    # EmptyOperator at the start
15    start = EmptyOperator(
16        task_id='start'
17    )
18
19    # Create a pet table in the PostgreSQL database
20    create_pet_table = PostgresOperator(
21        task_id='create_pet_table',
22        sql="""
23        CREATE TABLE IF NOT EXISTS pet (
24            pet_id SERIAL PRIMARY KEY,
25            name VARCHAR NOT NULL,
26            pet_type VARCHAR NOT NULL,
27            birth_date DATE NOT NULL,
28            owner VARCHAR NOT NULL
29        );
30        """,
31        postgres_conn_id='postgres_default', # Replace with your PostgreSQL connection ID
32    )
33
34    # Populate the pet table with sample data
35    populate_pet_table = PostgresOperator(
36        task_id='populate_pet_table',
37        sql="sql/insert_pet.sql", # Path to your SQL file containing INSERT statements
38        postgres_conn_id='postgres_default', # Replace with your PostgreSQL connection ID
39    )
40
41    # EmptyOperator at the end
42    end = EmptyOperator(
43        task_id='end'
44    )
45
46    # Set up task dependencies
47    start >> create_pet_table >> populate_pet_table >> end
48

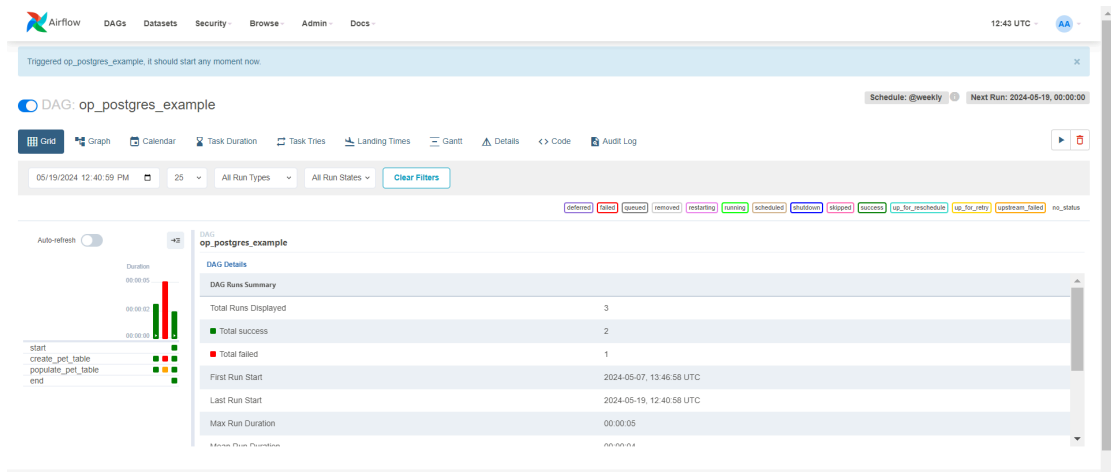
```

5. Jalankan DAGs `op_mysql` dan `op_postgresql` hingga success kemudian lampirkan hasil screenshotnya. Usahakan sampai pipeline berjalan success semua.

`op_mysql` :



`op_postgresql` :



6. Tampilkan hasil table yang dibuat oleh op_mysql dan op_postgresql

Properties DATA Monitor						
SELECT * FROM pet LIMIT 100						
Q Search results						
Cost: 55ms < 1 2 > Total 200						
		pet id integer	name varchar	pet type varchar	birth date date	owner varchar
> 1	1		6a0a428c86597f8481fcd0fcd	Cat	2024-01-28	811507e4b90cdf847beb795
> 2	2		04942882bad41f941b02027	Cat	2024-01-28	a72624ed57f4c20ad92b74b
> 3	3		2e9180703d5fe129ce4e4b6	Dog	2024-01-28	0a52691d05a5aa3062bb12e
> 4	4		117119822fb6f0eb819fbf84	Dog	2024-01-28	68ac153393d04f5d23b51d7
> 5	5		f940e16b466e7b837bb495e	Cat	2024-05-07	41c07cec0575530b71c98d5

7. Jelaskan apa yang dikerjakan oleh op_mysql dan op_postgresql? Apa perbedaan sebelum dan sesudah dijalanannya DAG? dan apa yang dapat di improve dari kedua DAG tersebut?

Yang dikerjakan MySQL & PostgreSQL

MySQL adalah sistem manajemen basis data relasional untuk menyimpan data sebagai tabel dengan baris dan kolom. Sistem ini populer sehingga digunakan di banyak aplikasi web, situs web dinamis, dan sistem tertanam. PostgreSQL adalah sistem manajemen basis data relasional-objek yang menawarkan lebih banyak fitur daripada MySQL.

Perbedaan sebelum dan sesudah dijalankan DAG

Perbedaan

MySQL terkenal dengan kecepatan dan keandalannya, dan sering digunakan untuk aplikasi web dan transaksi online karena metode pengelolaan datanya yang sederhana dan efektif. PostgreSQL menonjol karena fitur-fitur canggihnya, seperti dukungan untuk kueri kompleks, ekstensibilitas, dan kepatuhan standar SQL, menjadikannya database pilihan untuk aplikasi yang memerlukan pemrosesan data ekstensif dan kemampuan analitis.

Persamaan

MySQL dan PostgreSQL adalah dua sistem manajemen basis data relasional open source canggih yang menyediakan integritas data yang kuat, dukungan untuk fitur ACID (Atomicity, Consistency, Isolation, Durability), dan berbagai antarmuka bahasa pemrograman. Mereka mempunyai komitmen yang sama untuk menyediakan solusi terukur untuk berbagai aplikasi, mulai dari inisiatif skala kecil hingga penerapan di tingkat perusahaan, menjadikannya pilihan serbaguna bagi pengembang dan organisasi.

8. Buat Kesimpulan mengenai Apache Airflow

Apache Airflow adalah sebuah platform untuk menjadwalkan dan mengatur data pipeline atau workflow. Pengaturan data pipeline di sini mengacu pada pengaturan, koordinasi, penjadwalan, dan pengelolaan data pipeline yang kompleks dari sumber yang berbeda-beda. Data pipeline tersebut membawa data set yang siap dipergunakan, baik itu oleh aplikasi business intelligence dan data scientist.