Final Project
ETL Process of COVID-19 Jabar Data

—

Abu Bakar Adni

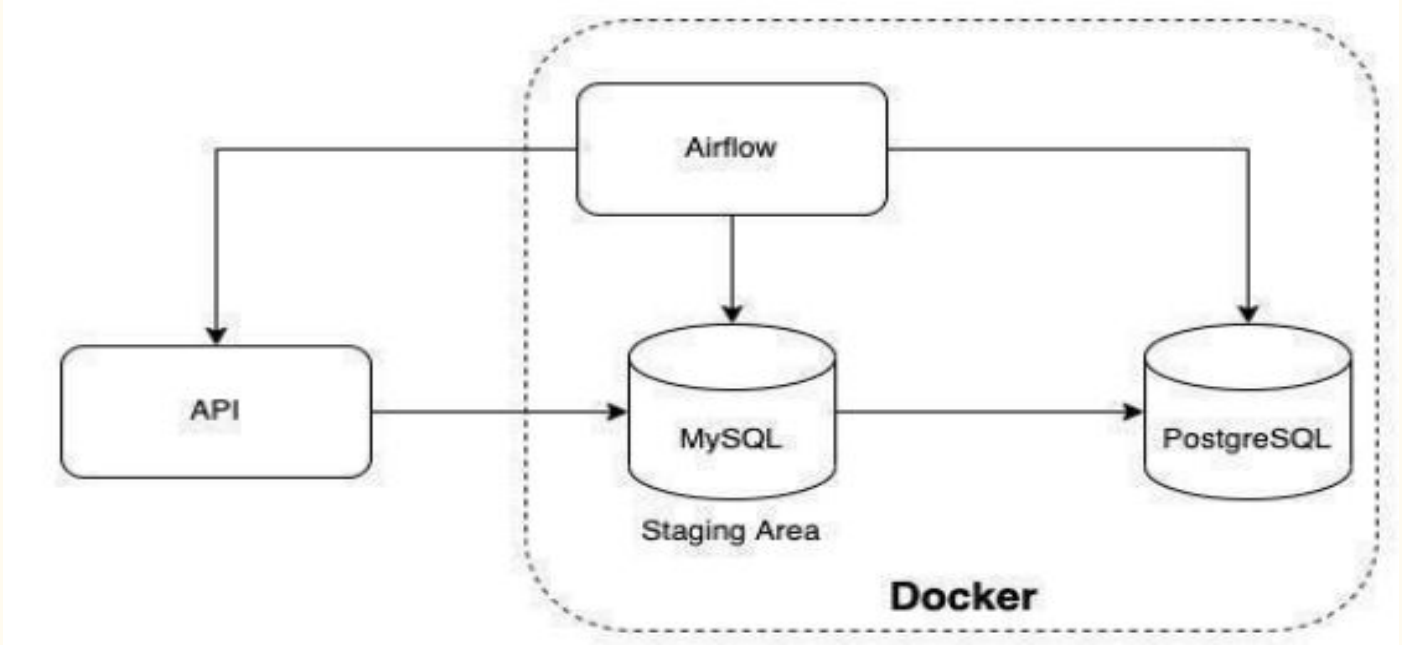
Project Scripts

- Please refer to this link in order to see all scripts and docker configurations.

Project Overview

- This project is about Dockerize ETL Pipeline using ETL tools and Airflow that extract Public API data from PIKOBAR, then load into MySQL (Staging Area) and finally aggregate the data and save into PostgreSQL.
- This will include many tools in order to run this ETL Process, there are:
  - Visual Studio Code : to run DAG file, extraction script, and dimension and fact tables creation scripts.
  - Docker Desktop : to create airflow and database containers with configuring docker-compose.yaml files.
  - Airflow : to schedule the whole process from extracting, load, and transform.
  - MySQL : Staging Area.
  - PostgreSQL : Data Mart will be loaded here.

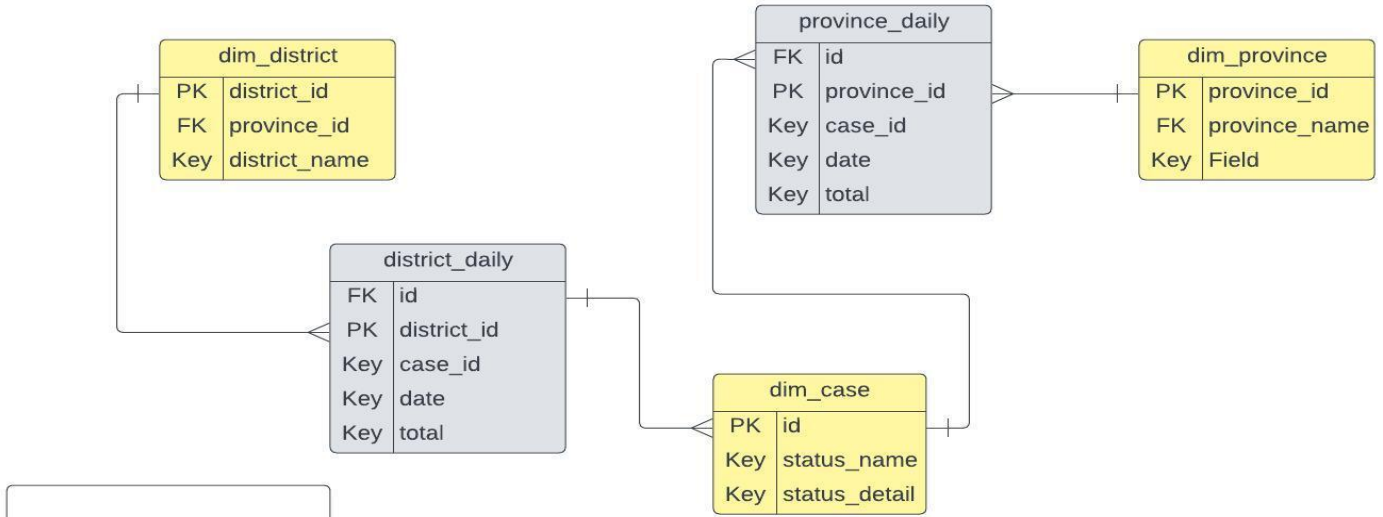
ETL Architecture Diagram



Project Steps

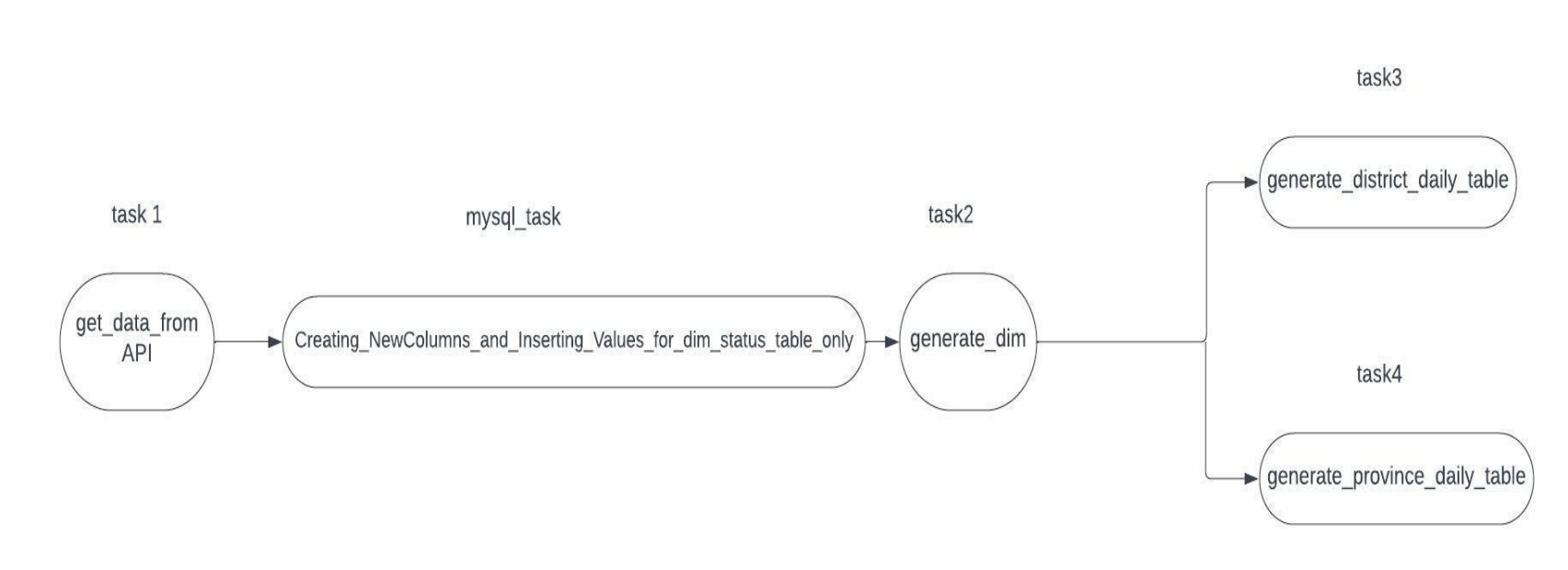
1. Create Docker (MySQL, Airflow and PostgreSQL) in your local computer.
2. Create Database in MySQL and PostgreSQL
3. First create connection on Airflow in order to extract data from API Endpoint.
4. Create DLL in MySQL.
5. Extract data from API Endpoint then the data will be loaded and staged in MySQL.
6. Create DDL in PostgreSQL for Fact table and Dimension table.
7. Create load data to Dimension table
8. Create script for aggregate Province Daily save to Province Daily Table
9. Create script for aggregate District Daily save to District Daily Table
10. Create DAG with schedule daily basis with task: a. get data from api, b. generate dimension c. district daily d. province daily

Relational Database Model Covid Jabar 2020

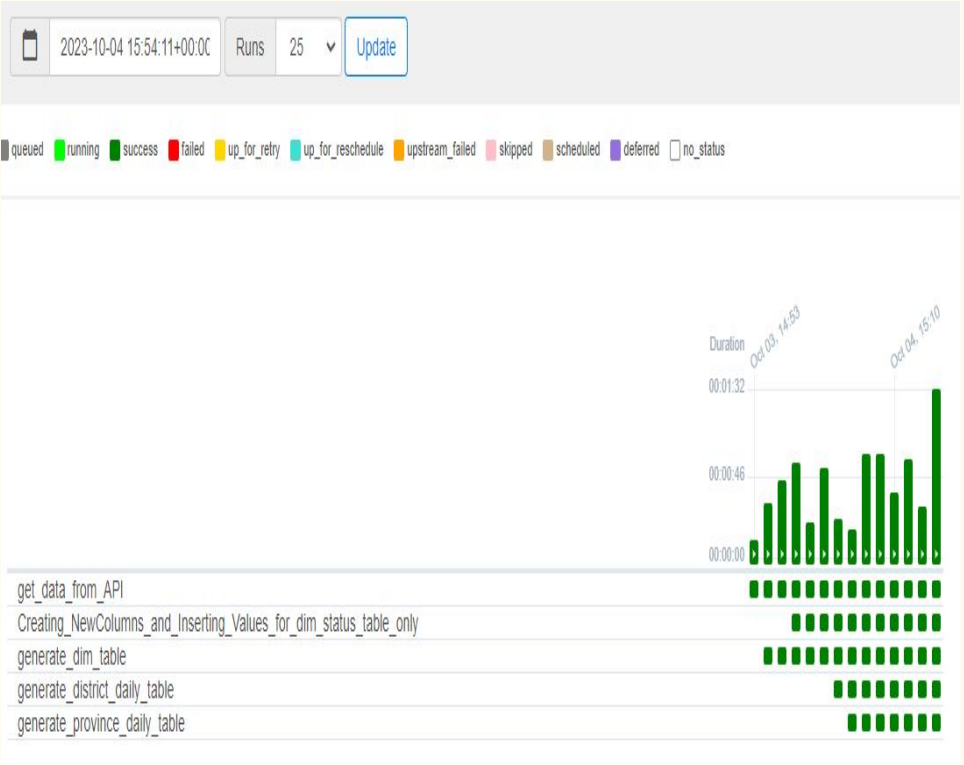


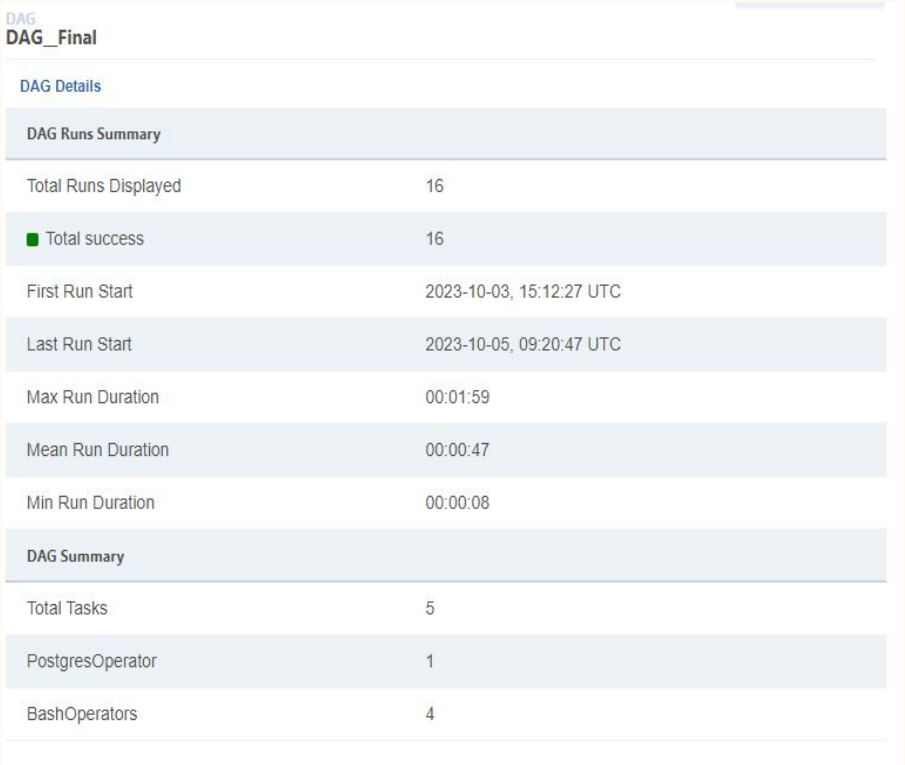
Yellow = Dimentional Table
Light Grey = Fact Table

DAG Flow


task 1
get_data_from API
mysql_task
Creating_NewColumns_and_Inserting_Values_for_dim_status_table_only
task2
generate_dim
task3
generate_district_daily_table
task4
generate_province_daily_table

Project's Outputs (Screenshot) : Airflow



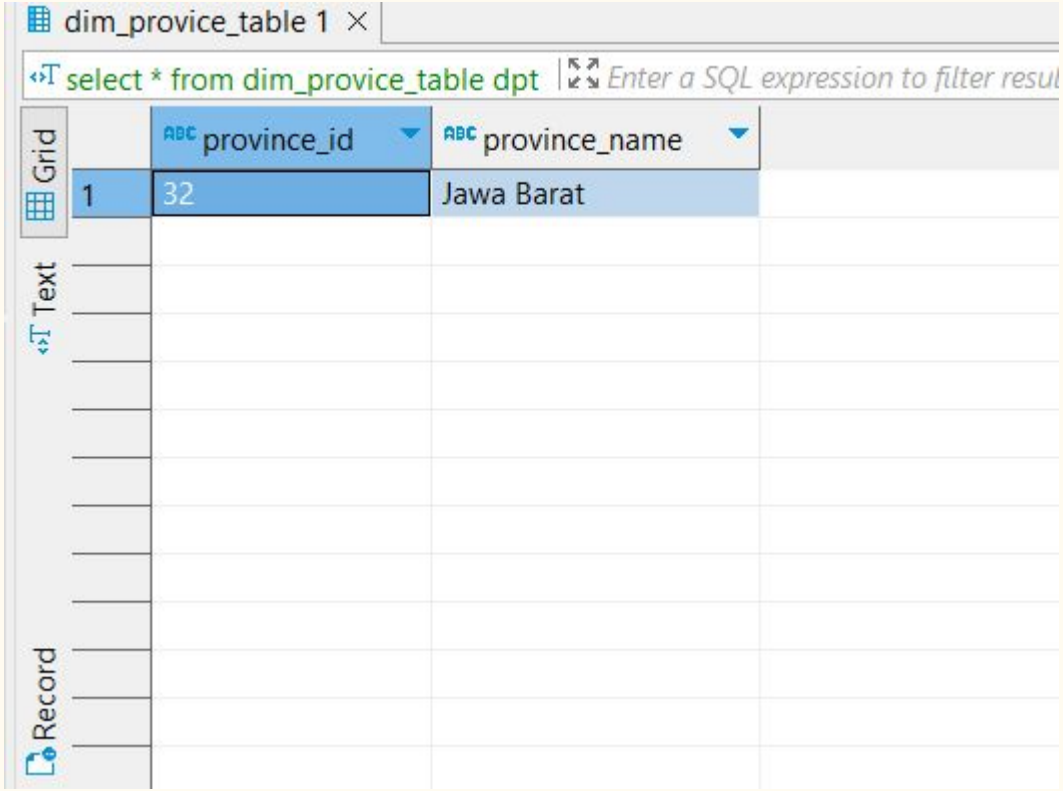


Project's Outputs (Screenshot) : Dim_District_Table

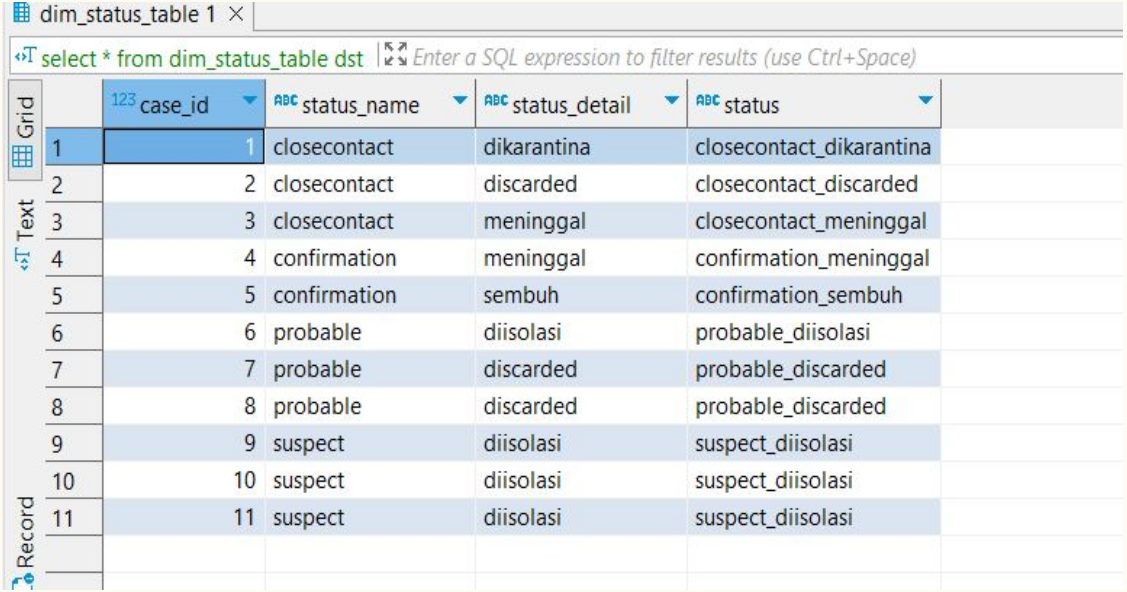
Grid
Text
Record

	district_id	district_name	nama_prov
1	3204	Kabupaten Bandung	Jawa Barat
2	3217	Kabupaten Bandung Barat	Jawa Barat
3	3216	Kabupaten Bekasi	Jawa Barat
4	3201	Kabupaten Bogor	Jawa Barat
5	3207	Kabupaten Ciamis	Jawa Barat
6	3203	Kabupaten Cianjur	Jawa Barat
7	3209	Kabupaten Cirebon	Jawa Barat
8	3205	Kabupaten Garut	Jawa Barat
9	3212	Kabupaten Indramayu	Jawa Barat
10	3215	Kabupaten Karawang	Jawa Barat
11	3208	Kabupaten Kuningan	Jawa Barat
12	3210	Kabupaten Majalengka	Jawa Barat
13	3218	Kabupaten Pangandaran	Jawa Barat

Project's Outputs (Screenshot) : Dim_Province_Table



Project's Outputs (Screenshot) : Dim_Status_Table



dim_status_table 1

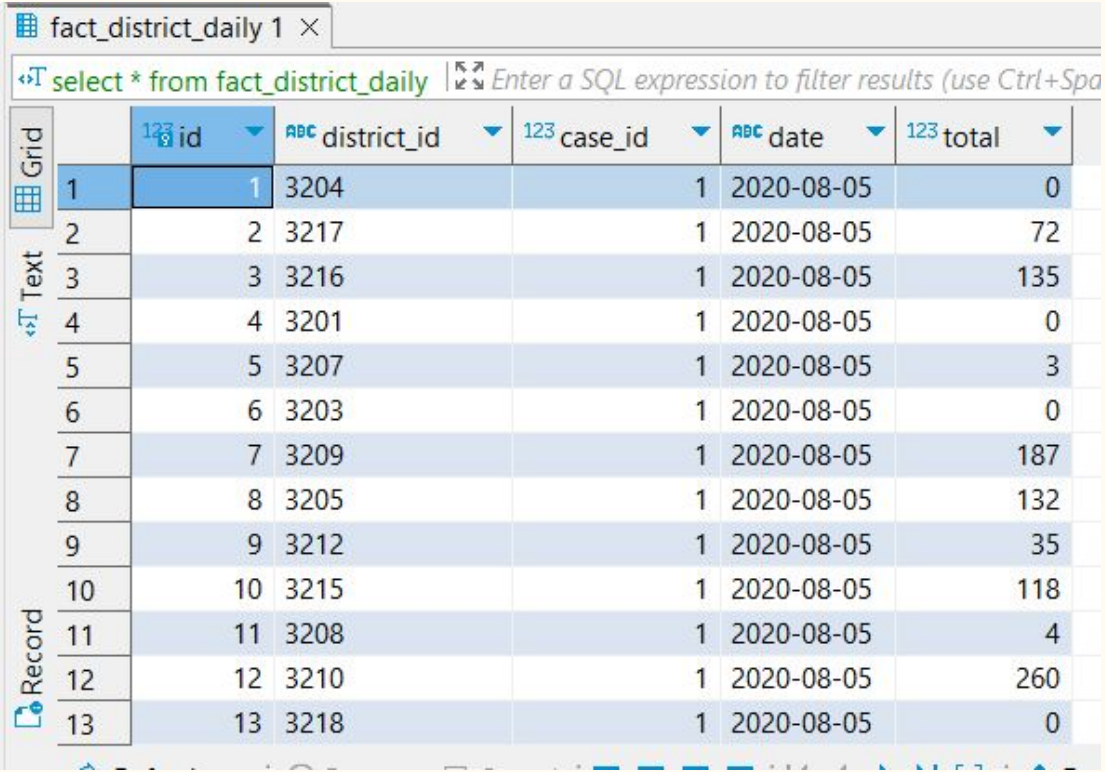
```
select * from dim_status_table dst
```

Enter a SQL expression to filter results (use Ctrl+Space)

Grid | Text | Record

	case_id	status_name	status_detail	status
1	1	closecontact	dikarantina	closecontact_dikarantina
2	2	closecontact	discarded	closecontact_discarded
3	3	closecontact	meninggal	closecontact_meninggal
4	4	confirmation	meninggal	confirmation_meninggal
5	5	confirmation	sembuh	confirmation_sembuh
6	6	probable	diisolasi	probable_diisolasi
7	7	probable	discarded	probable_discarded
8	8	probable	discarded	probable_discarded
9	9	suspect	diisolasi	suspect_diisolasi
10	10	suspect	diisolasi	suspect_diisolasi
11	11	suspect	diisolasi	suspect_diisolasi

Project's Outputs (Screenshot) : Fact_District_Daily



fact_district_daily 1

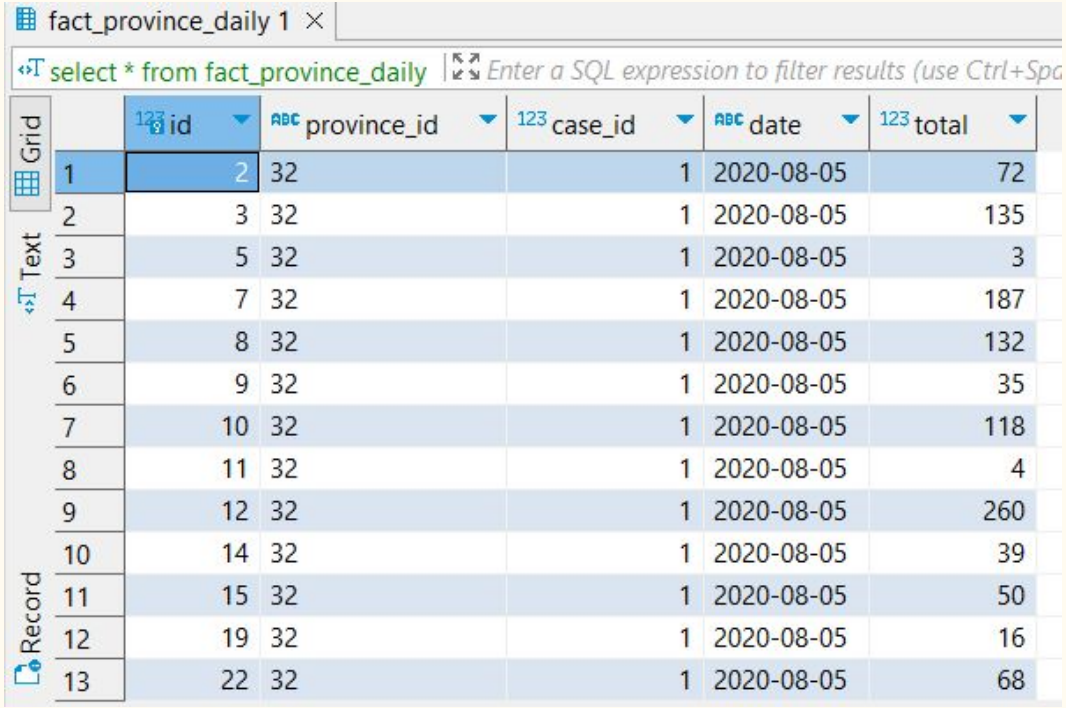
```
select * from fact_district_daily
```

Enter a SQL expression to filter results (use Ctrl+Spa

Grid | Text | Record

	id	district_id	case_id	date	total
1	1	3204	1	2020-08-05	0
2	2	3217	1	2020-08-05	72
3	3	3216	1	2020-08-05	135
4	4	3201	1	2020-08-05	0
5	5	3207	1	2020-08-05	3
6	6	3203	1	2020-08-05	0
7	7	3209	1	2020-08-05	187
8	8	3205	1	2020-08-05	132
9	9	3212	1	2020-08-05	35
10	10	3215	1	2020-08-05	118
11	11	3208	1	2020-08-05	4
12	12	3210	1	2020-08-05	260
13	13	3218	1	2020-08-05	0

Project's Outputs (Screenshot) : Fact_Province_Daily



fact_province_daily 1

```
select * from fact_province_daily
```

Enter a SQL expression to filter results (use Ctrl+Spa

Grid | Text | Record

	id	province_id	case_id	date	total
1	2	32	1	2020-08-05	72
2	3	32	1	2020-08-05	135
3	5	32	1	2020-08-05	3
4	7	32	1	2020-08-05	187
5	8	32	1	2020-08-05	132
6	9	32	1	2020-08-05	35
7	10	32	1	2020-08-05	118
8	11	32	1	2020-08-05	4
9	12	32	1	2020-08-05	260
10	14	32	1	2020-08-05	39
11	15	32	1	2020-08-05	50
12	19	32	1	2020-08-05	16
13	22	32	1	2020-08-05	68

Further Analysis Recommendation

- Deeper analysis like Single or Multiple Regression should be conducted in order to understand current trend of the data and predict future output if possible.
- Visualize the result by using visualisation tool (Metabase, Looker Studio, Power BI or Tableau).

Issues Found

- Both fact tables need to be melted (using .melt in pandas) to it will compatible with the status condition on table dim_status_table.
- Auto generate id column might cause error when Airflow is rerun for multiple times because the id might have been changed when it run multiple times.