



# End-to-End Data Pipeline With Airflow

Study Case : Data Covid-19 Provinsi Jabar

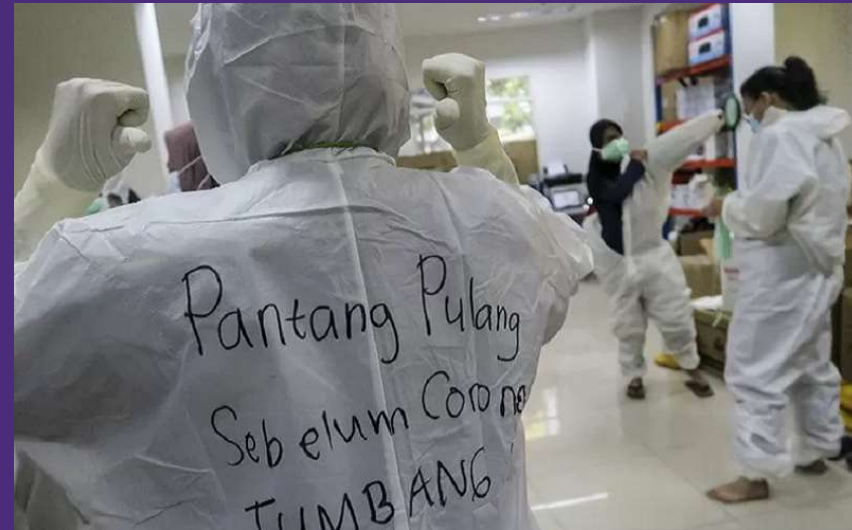
Present by : INDRA RAHMAWAN

# Latar Belakang

Dalam rangka menyambut liburan hari raya idul fitri 1443 H dan sehubungan akan dibukanya beberapa tempat wisata di provinsi Jawa Barat, pemerintah tentunya ingin mengetahui apakah di wilayah Jawa Barat sudah tergolong aman atau belum dari covid-19.

Maka dari itu, kami diminta oleh Pak Gubernur untuk menyajikan data perkembangan covid-19 khususnya wilayah Jawa Barat agar pemerintah dapat mengambil keputusan untuk dibukanya tempat wisata atau tidak.

#AYOPAKAIDATA



# Study Case

## Fetch From API

[https://covid19-public.digitalservice.id/api/v1/rekapitulasi\\_v2/jabar/harian?level=kab](https://covid19-public.digitalservice.id/api/v1/rekapitulasi_v2/jabar/harian?level=kab)

```
{,|
  "content": [
    {
      "tanggal": "2020-08-05",
      "kode_prov": "32",
      "nama_prov": "Jawa Barat",
      "kode_kab": "3204",
      "nama_kab": "Kabupaten Bandung",
      "SUSPECT": 2210,
      "CLOSECONTACT": 274,
      "PROBABLE": 26,
      "suspect_diisolasi": 31,
      "suspect_discarded": 2179,
      "closecontact_dikarantina": 0,
      "closecontact_discarded": 274,
      "probable_diisolasi": 0,
      "probable_discarded": 0,
      "CONFIRMATION": 0,
      "confirmation_sembuh": 0,
      "confirmation_meninggal": 0,
      "suspect_meninggal": 0,
      "closecontact_meninggal": 0,
      "probable_meninggal": 26
    }
  ],
}
```

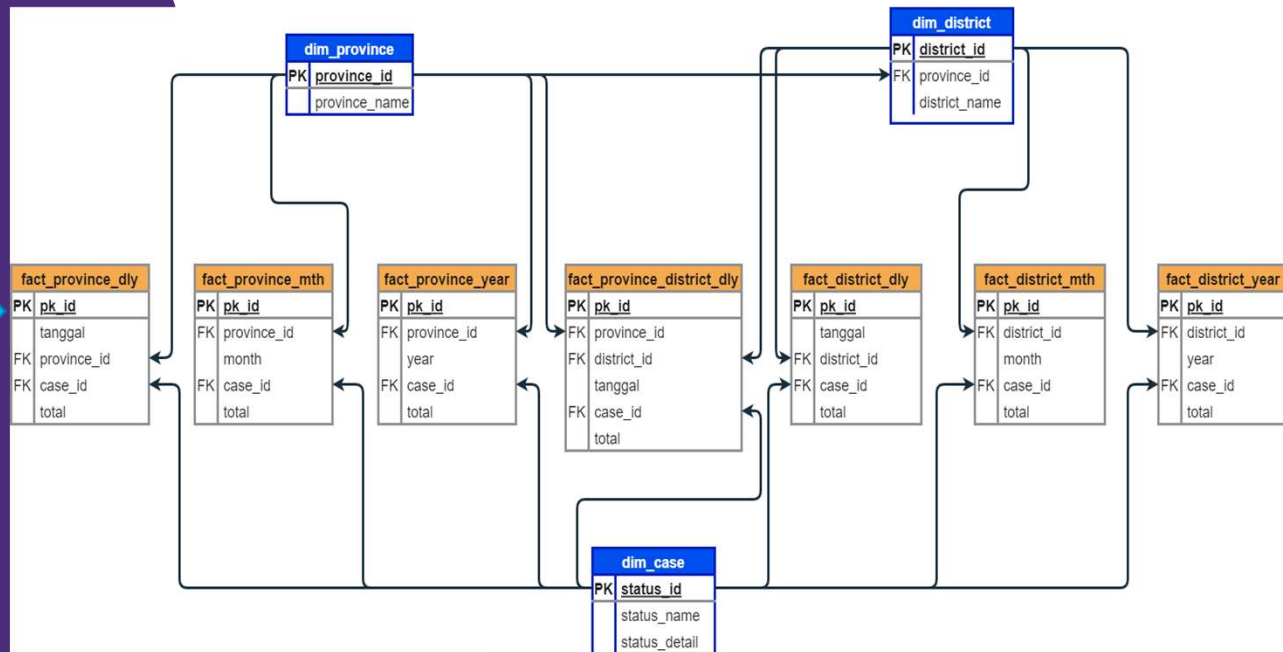
### Tujuan :

- Berapa kasus yang ada di masing-masing kota di provinsi jawa barat disetiap harinya ?
- Berapa kasus yang ada di di masing-masing kota di provinsi jawa barat disetiap bulannya ?
- Berapa kasus yang ada di masing-masing kota di provinsi jawa barat disetiap tahunnya ?
- Kota mana saja yang tergolong masih tinggi angka terpapar covid-19 ?

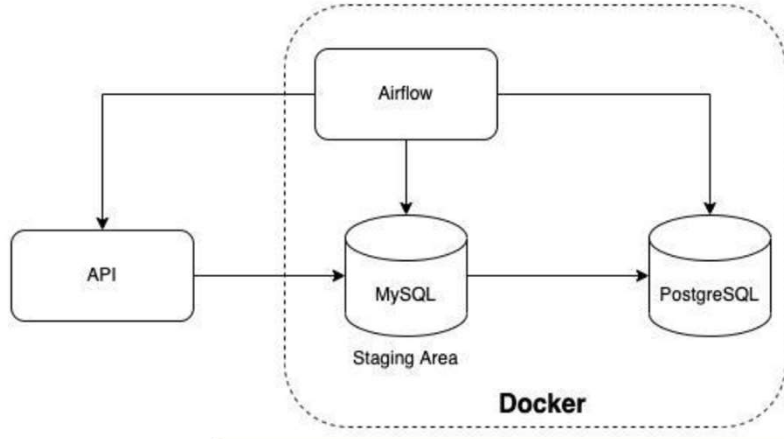
# Study Case

```
{,|
"content": [
  {
    "tanggal": "2020-08-05",
    "kode_prov": "32",
    "nama_prov": "Jawa Barat",
    "kode_kab": "3204",
    "nama_kab": "Kabupaten Bandung",
    "SUSPECT": 2210,
    "CLOSECONTACT": 274,
    "PROBABLE": 26,
    "suspect_diisolasi": 31,
    "suspect_discarded": 2179,
    "closecontact_dikarantina": 0,
    "closecontact_discarded": 274,
    "probable_diisolasi": 0,
    "probable_discarded": 0,
    "CONFIRMATION": 0,
    "confirmation_sembuh": 0,
    "confirmation_meninggal": 0,
    "suspect_meninggal": 0,
    "closecontact_meninggal": 0,
    "probable_meninggal": 26
  },

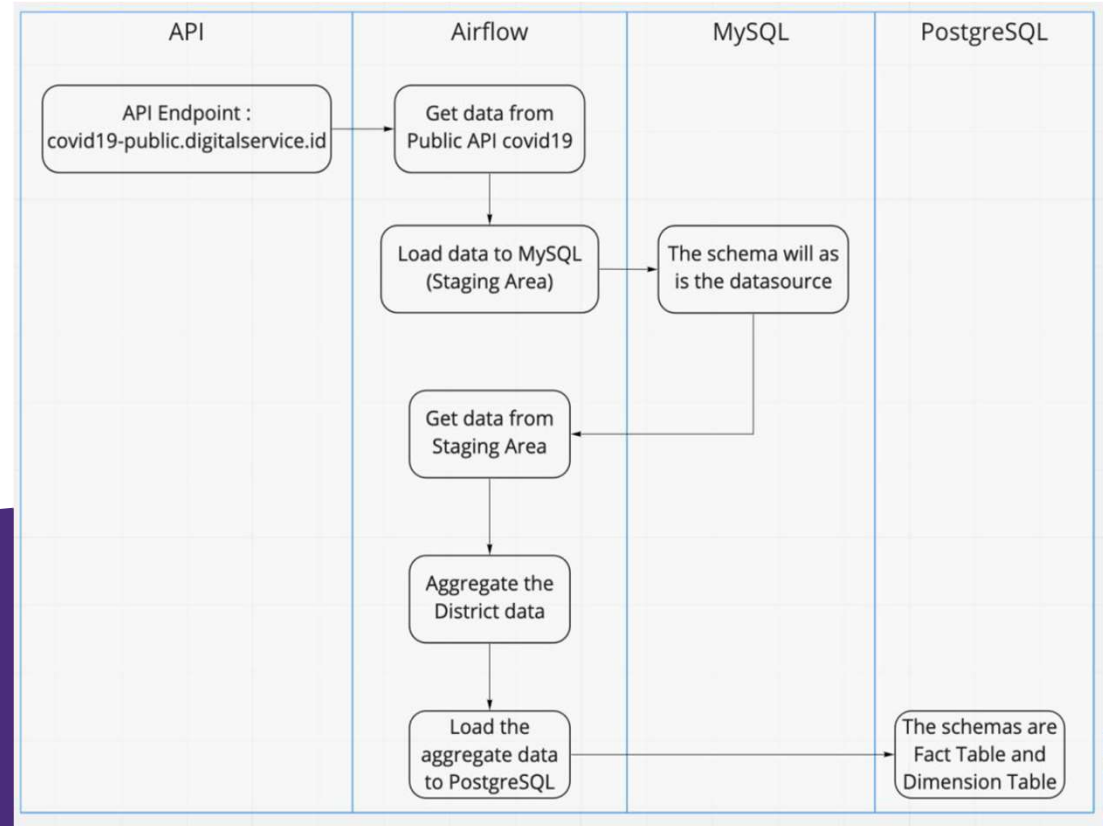
```



# Environment



ETL Architecture Diagram



Integration Design Diagram

# Create Docker (Airflow, MySQL, PostgreSQL)

```
curl -Lf0 'https://airflow.apache.org/docs/apache-airflow/2.3.0/docker-compose.yaml'
```

```
docker-compose.yml
1 version: '2'
2
3 services:
4   postgresql: ...
5   redis:
6     image: docker.io/bitnami/redis:6.0
7     volumes:
8       - 'redis_data:/bitnami'
9     environment:
10      # ALLOW_EMPTY_PASSWORD is recommended only for development.
11      - ALLOW_EMPTY_PASSWORD=yes
12   airflow-scheduler:
13     # TODO: to be reverted to use proper registry/distro on T39132
14     # image: docker.io/bitnami/airflow-scheduler:2
15     image: docker.io/bitnami/airflow-scheduler:2.2.2-debian-10-r9
16     environment:
17       - AIRFLOW_DATABASE_NAME=bitnami_airflow
18       - AIRFLOW_DATABASE_USERNAME=bn_airflow
19       - AIRFLOW_DATABASE_PASSWORD=bitnamil
20       - AIRFLOW_EXECUTOR=CeleryExecutor
21       - AIRFLOW_WEBSERVER_HOST=airflow
22     volumes:
23       - airflow_scheduler_data:/bitnami
24       - ./dags:/opt/bitnami/airflow/dags
25       - logs_volume:/opt/bitnami/airflow/logs
26       - ./requirements.txt:/bitnami/python/requirements.txt
27   airflow-worker:
28     # TODO: to be reverted to use proper registry/distro on T39132
29     # image: docker.io/bitnami/airflow-worker:2
30     image: docker.io/bitnami/airflow-worker:2.2.2-debian-10-r9
31     environment:
32       - AIRFLOW_DATABASE_NAME=bitnami_airflow
33       - AIRFLOW_DATABASE_USERNAME=bn_airflow
34       - AIRFLOW_DATABASE_PASSWORD=bitnamil
35       - AIRFLOW_EXECUTOR=CeleryExecutor
36       - AIRFLOW_WEBSERVER_HOST=airflow
37     volumes:
```

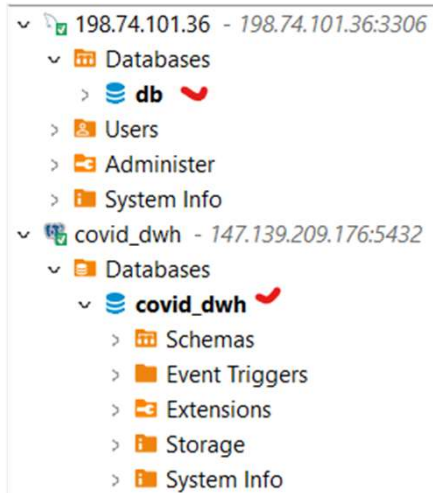
```
docker-compose.yml
72 pg:
73   image: docker.io/bitnami/postgresql:11
74   env_file:
75     - database.env
76   ports:
77     - '5433:5432'
78   expose:
79     - '5433'
80   volumes:
81     - 'pg_data:/bitnami/postgresql'
82   environment:
83     - 'ALLOW_EMPTY_PASSWORD=yes'
84   mysql:
85     image: mysql:5.7
86     restart: always
87     environment:
88       MYSQL_DATABASE: 'db'
89       # So you don't have to use root, but you can if you like
90       MYSQL_USER: 'user'
91       # You can use whatever password you like
92       MYSQL_PASSWORD: 'password'
93       # Password for root access
94       MYSQL_ROOT_PASSWORD: 'password'
95     ports:
96       # <Port exposed> : <MySQL Port running inside container>
97       - '3306:3306'
98     expose:
99       # Opens port 3306 on the container
100       - '3306'
101       # Where our data will be persisted
102     volumes:
103       - my-db:/var/lib/mysql
```

Install the latest PowerShell for new features and improvements! <https://aka.ms/PSWindows>

#AYOPAKAIDATA

PS D:\DE - DigitalSkola\Week 15\Final Project\etl\_covid\_airflow\_indra> docker-compose up|

## Create Database MySQL & PostgreSQL



## Create Connection from Airflow To PostgreSQL

Edit Connection

Connection Id *	pg_dwh
Connection Type *	Postgres
Connection Type missing? Make sure you've installed the corresponding Airflow Provider Package.	
Description	connection to postgres
Host	147.139.209.176
Schema	covid_dwh
Login	indra
Password	
Port	5432
Extra	



# Create DAG

Name	Date modified	Type	Size
dags	06/05/2022 14.47	File folder	
ERD	07/05/2022 11.24	File folder	
PPT	07/05/2022 15.51	File folder	
database.env	06/05/2022 19.24	ENV File	1 KB
docker-compose.yml	06/05/2022 14.42	Yaml Source File	4 KB
README.md	17/12/2021 20.47	Markdown Source ...	1 KB
requirements.txt	17/12/2021 20.47	Text Document	1 KB

```
dag_final_project.py > [e] dag

import time
from datetime import datetime, timedelta, date
from pprint import pprint

from airflow import DAG
from airflow.operators.python import PythonOperator
from airflow.operators.bash_operator import BashOperator
from airflow.providers.postgres.operators.postgres import PostgresOperator
from airflow.operators.dummy_operator import DummyOperator

from scripts.main import main

url = 'https://covid19-public.digitalservice.id/api/v1/rekapitulasi_v2/jabar/harian?level=kab'
default_args = {
    'owner': 'rahmawaminggu',
    'depends_on_past': False,
    'email_on_failuer': False,
    'email_on_retry': False,
    'retries': 0
}

with DAG(
    dag_id='dag_final_project',
    schedule_interval='0 3,15 * * *',
    start_date=datetime(2022, 5, 1),
    catchup=False,
    tags=['de'],
    default_args=default_args
) as dag:
```

```
start >> fetch_json >> run_ddl >> [run_dim_case, run_dim_province, run_dim_district] >> step_2 >> run_fact_province_district
run_fact_province_district >> run_fact_province_dly >> run_fact_province_mth >> run_fact_province_yearly >> step_3
run_fact_province_district >> run_fact_district_dly >> run_fact_district_mth >> run_fact_district_yearly >> step_3
```

```
# [END howto_operator_python_kwargs]
```

#AYOPAKAIDATA



# Get data from API and load to Mysql & PostgreSQL

```
from scripts.main import main
```

Memanggil func main() di main.py

```
) as dag:

    start = DummyOperator(
        task_id='start_job',
    )

    fetch_json = PythonOperator(
        task_id='fetch_json',
        python_callable=main,
        op_kwargs={'url': url}
    )
    # [END howto_operator_python]
```

```
scripts > main.py > ...
import requests
import pandas as pd
import mysql.connector
from sqlalchemy import create_engine
import psycopg2 as pg

def fetch_api(url):
    r = requests.get(url)
    res = pd.DataFrame(r.json()[ 'data' ][ 'content' ])
    res.columns = map(str.lower, res.columns)

    return res

def to_stg(res):
    engine = create_engine('mysql+mysqlconnector://user:password@198.74.101.36:3306/db', echo=False)
    res.to_sql(name='stg_covid_data', con=engine, if_exists = 'replace', index=False)

def to_dwh():
    engine1 = create_engine('mysql+mysqlconnector://user:password@198.74.101.36:3306/db', echo=False)
    pg_engine = create_engine('postgresql+psycopg2://indra:indra@147.139.209.176/covid_dwh')
    load_stg = pd.read_sql('select * from stg_covid_data', engine1)
    load_stg.to_sql(name='stg_covid_data', con=pg_engine, if_exists = 'replace', index=False, method='multi')

def main(url):
    res = fetch_api(url)
    to_stg(res)
    to_dwh()

if __name__ == '__main__':
    main()
```

Main.py → load from api to mysql -> mysql to postgres

## Create Fact & Dimension table

```
run_ddl = PostgresOperator(  
    task_id="run_ddl",  
    postgres_conn_id='pg_dwh',  
    sql='sql/ddl_final_prj.sql'  
)
```

Postgres operator untuk merunning ddl

```
> sql > ddl_final_prj.sql  
DROP TABLE IF EXISTS dim_case;  
create table dim_case  
(  
    status_id serial primary key,  
    status_name varchar(255),  
    status_detail varchar(255)  
);  
  
DROP TABLE IF EXISTS dim_province;  
create table dim_province  
(  
    province_id int,  
    province_name varchar(255)  
);  
  
DROP TABLE IF EXISTS dim_district;  
create table dim_district  
(  
    district_id int,  
    province_id int,  
    district_name varchar(255)  
);
```

Dimension table

```
DROP TABLE IF EXISTS fact_district_dly;  
create table fact_district_dly  
(  
    pk_id serial primary key,  
    tanggal date,  
    district_id int,  
    case_id int,  
    total bigint  
);
```

```
DROP TABLE IF EXISTS fact_district_mth;  
create table fact_district_mth  
(  
    pk_id serial primary key,  
    district_id int,  
    month int,  
    case_id int,  
    total bigint  
);
```

```
DROP TABLE IF EXISTS fact_district_year;  
create table fact_district_year  
(  
    pk_id serial primary key,  
    district_id int,  
    year int,  
    case_id int,  
    total bigint  
);
```

```
DROP TABLE IF EXISTS fact_province_dly;  
create table fact_province_dly  
(  
    pk_id serial primary key,  
    tanggal date,  
    province_id int,  
    case_id int,  
    total bigint  
);
```

```
DROP TABLE IF EXISTS fact_province_mth;  
create table fact_province_mth  
(  
    pk_id serial primary key,  
    province_id int,  
    month int,  
    case_id int,  
    total bigint  
);
```

```
DROP TABLE IF EXISTS fact_province_year;  
create table fact_province_year  
(  
    pk_id serial primary key,  
    province_id int,  
    year int,  
    case_id int,  
    total bigint  
);
```

Fact Table

## Insert into Dimension table

```
run_dim_case = PostgresOperator(  
    task_id="run_dim_case",  
    postgres_conn_id='pg_dwh',  
    sql='sql/load_case_dim.sql'  
)  
  
run_dim_province = PostgresOperator(  
    task_id="run_dim_province",  
    postgres_conn_id='pg_dwh',  
    sql='sql/load_province_dim.sql'  
)  
  
run_dim_district = PostgresOperator(  
    task_id="run_dim_district",  
    postgres_conn_id='pg_dwh',  
    sql='sql/load_district_dim.sql'  
)
```



```
sql > load_case_dim.sql  
truncate table dim_case;  
  
insert into dim_case (status_name, status_detail)  
with stg as (  
    select distinct unnest(array['suspect_meninggal','suspect_discarded','suspect_diisolasi','closecontact_dikarantina','closecontact_discarded',  
                                'probable_diisolasi','probable_discarded','probable_meninggal','confirmation_sembuh','confirmation_meninggal'])  
    from stg_covid_data  
) select case  
    when stg.status_detail like 'suspect%' then 'suspect'  
    when stg.status_detail like 'closecontact%' then 'closecontact'  
    when stg.status_detail like 'probable%' then 'probable'  
    when stg.status_detail like 'confirmation%' then 'confirmation'  
end as status_name,  
    stg.status_detail  
from stg  
order by stg.status_detail
```

```
sql > load_district_dim.sql  
TRUNCATE TABLE dim_district;  
  
INSERT INTO dim_district  
select  
    distinct  
    kode_kab::int as district_id,  
    kode_prov::int as province_id,  
    nama_kab as district_name  
from stg_covid_data;
```

```
sql > load_province_dim.sql  
TRUNCATE TABLE dim_province;  
  
INSERT INTO dim_province  
SELECT  
    DISTINCT  
    kode_prov::int as province_id,  
    nama_prov as province_name  
FROM stg_covid_data;
```

Postgres operator insert into dimension

#AYOPAKAIDATA



## Aggregate into *Fact* table

```
run_fact_province_district = PostgresOperator(  
    task_id="run_fact_province_district",  
    postgres_conn_id='pg_dwh',  
    sql='sql/load_province_district_dly.sql'  
)
```



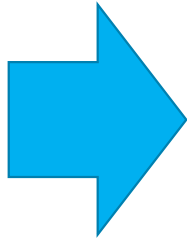
```
sql > load_province_district_dly.sql  
truncate table fact_province_district_dly;  
  
INSERT INTO fact_province_district_dly (province_id, district_id, tanggal, case_id, total)  
WITH ds as (  
    SELECT kode_prov::int      as province_id,  
           kode_kab::int      as district_id,  
           tanggal::date     as tanggal,  
           status_id          as case_id,  
           sum(case  
               when dc.status_detail in ('suspect_discarded') then coalesce(stg.suspect_discarded, 0)  
               when dc.status_detail in ('suspect_meninggal') then coalesce(stg.suspect_meninggal, 0)  
               when dc.status_detail in ('suspect_discarded') then coalesce(stg.suspect_discarded, 0)  
               when dc.status_detail in ('probable_diisolasi') then coalesce(stg.probable_diisolasi, 0)  
               when dc.status_detail in ('probable_discarded') then coalesce(stg.probable_discarded, 0)  
               when dc.status_detail in ('probable_meninggal') then coalesce(stg.probable_meninggal, 0)  
               when dc.status_detail in ('confirmation_sembuh') then coalesce(stg.confirmation_sembuh, 0)  
               when dc.status_detail in ('confirmation_meninggal') then coalesce(stg.confirmation_meninggal, 0)  
               when dc.status_detail in ('closecontact_dikarantina') then coalesce(stg.closecontact_dikarantina, 0)  
               when dc.status_detail in ('closecontact_discarded') then coalesce(stg.closecontact_discarded, 0)  
               when dc.status_detail in ('closecontact_meninggal') then coalesce(stg.closecontact_meninggal, 0)  
               else 0 end) as total  
           FROM stg_covid_data stg,  
                dim_case dc  
           GROUP BY province_id, district_id, tanggal, case_id  
       )  
SELECT province_id, district_id, tanggal, case_id, total  
FROM ds  
WHERE total > 0;
```



# Aggregate into *Fact table*

PostgresOperator untuk aggregate Province daily, monthlly, yearly

```
run_fact_province_dly = PostgresOperator(  
    task_id="run_fact_province_dly",  
    postgres_conn_id='pg_dwh',  
    sql='sql/load_province_dly.sql'  
)  
  
run_fact_province_mth = PostgresOperator(  
    task_id="run_fact_province_mth",  
    postgres_conn_id='pg_dwh',  
    sql='sql/load_province_mth.sql'  
)  
  
run_fact_province_yearly = PostgresOperator(  
    task_id="run_fact_province_yearly",  
    postgres_conn_id='pg_dwh',  
    sql='sql/load_province_yearly.sql'  
)
```



```
sql > load_province_dly.sql  
truncate table fact_province_dly;  
  
INSERT INTO fact_province_dly (tanggal, province_id, case_id, total)  
SELECT  
    tanggal,  
    province_id,  
    case_id,  
    sum(total) as total  
FROM fact_province_district_dly  
GROUP BY  
    province_id,  
    tanggal,  
    case_id;
```

```
sql > load_province_mth.sql  
truncate table fact_province_mth;  
  
INSERT INTO fact_province_mth (province_id, month, case_id, total)  
SELECT  
    province_id,  
    to_char(tanggal, 'YYYYMM')::int as month,  
    case_id,  
    sum(total) as total  
FROM fact_province_dly  
GROUP BY province_id, month, case_id;
```

```
sql > load_province_yearly.sql  
truncate table fact_province_year;  
INSERT INTO fact_province_year (province_id, year, case_id, total)  
SELECT  
    province_id,  
    left(month::varchar, 4)::int as year ,  
    case_id,  
    sum(total) as total  
FROM fact_province_mth  
GROUP BY  
    province_id,  
    year,  
    case_id;
```

# Aggregate into *Fact table*

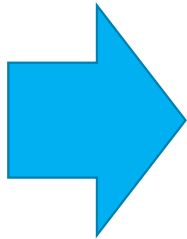
PostgresOperator untuk aggregate District daily, monthly, yearly

```
# [START DISTRICT AGG ]
run_fact_district_dly = PostgresOperator(
    task_id="run_fact_district_dly",
    postgres_conn_id='pg_dwh',
    sql='sql/load_district_dly.sql'
)

run_fact_district_mth = PostgresOperator(
    task_id="run_fact_district_mth",
    postgres_conn_id='pg_dwh',
    sql='sql/load_district_mth.sql'
)

run_fact_district_yearly = PostgresOperator(
    task_id="run_fact_district_yearly",
    postgres_conn_id='pg_dwh',
    sql='sql/load_district_yearly.sql'
)

# [END DISTRICT AGG ]
```



```
sql > load_district_dly.sql
truncate table fact_district_dly;
INSERT INTO fact_district_dly (tanggal, district_id, case_id, total)
SELECT
    tanggal,
    district_id,
    case_id,
    sum(total) as total
FROM fact_province_district_dly
GROUP BY tanggal, district_id, case_id;
```

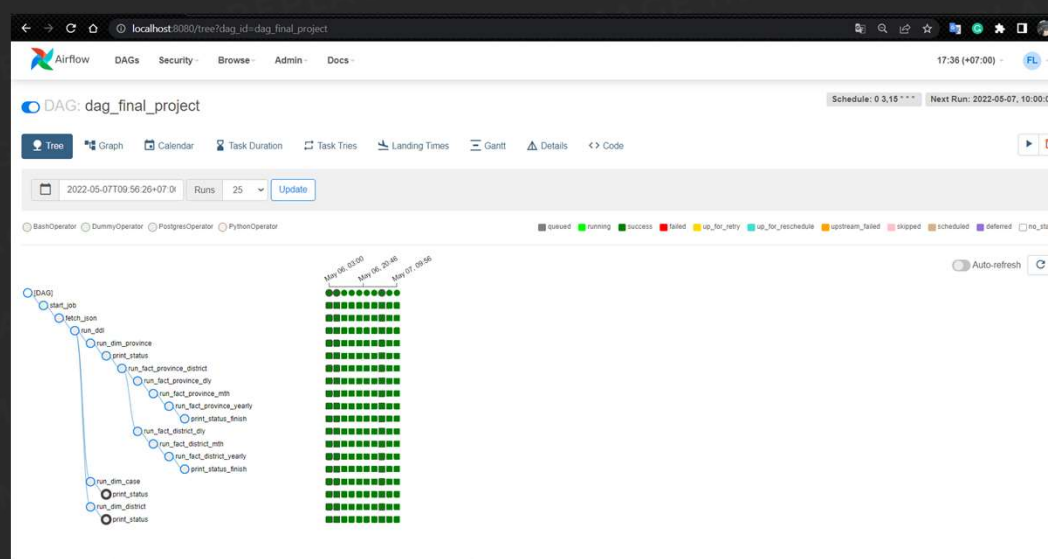
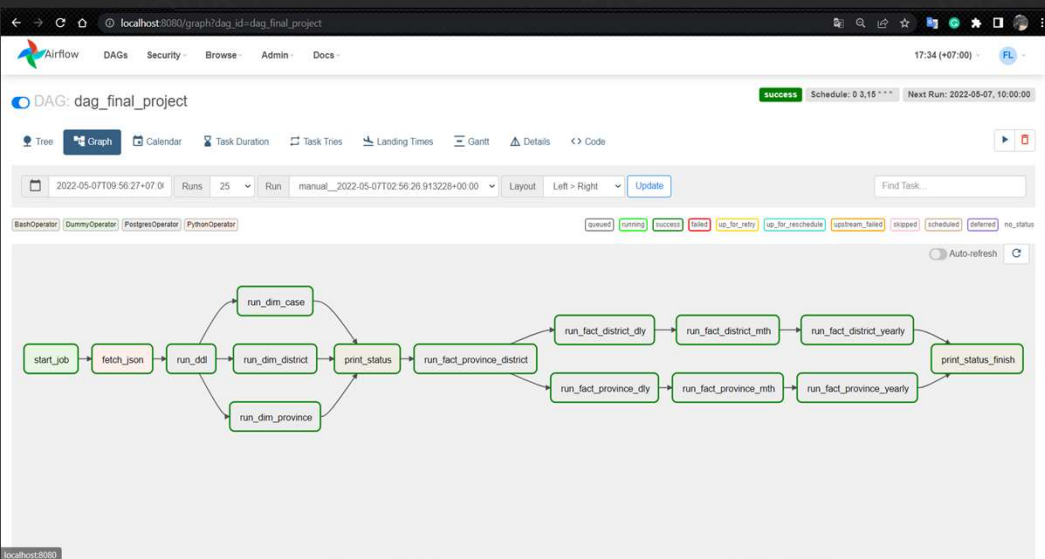
```
sql > load_district_mth.sql
truncate table fact_district_mth;

INSERT INTO fact_district_mth (district_id, month, case_id, total)
SELECT district_id, to_char(tanggal, 'YYYYMM')::int as month, case_id, sum(total) as total
FROM fact_district_dly
GROUP BY district_id, month, case_id;
```

```
sql > load_district_yearly.sql
truncate table fact_district_year;
INSERT INTO fact_district_year (district_id, year, case_id, total)
SELECT
    district_id,
    left(month::varchar, 4)::int as year,
    case_id,
    sum(total)
FROM fact_district_mth
GROUP BY district_id, year, case_id;
```



# Airflow Graph & Tree



#AYOPAKAIDATA

# Report table

www.website.com 16

stg\_covid\_data X

PropertiesDataER Diagram

198.74.101.36DatabasesdbTablesstg\_covid\_data

stg\_covid\_data Enter a SQL expression to filter results (use Ctrl+Space)

Grid	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Text	tanggal	kode_prov	nama_prov	kode_kab	nama_kab	suspect	closecontact	probable	suspect_diisolasi	suspect_discarded	c																			
1	2020-08-05	32	Jawa Barat	3204	Kabupaten Bandung	2,210	274	26	31	2,179																				
2	2020-08-05	32	Jawa Barat	3217	Kabupaten Bandung Barat	776	534	7	3	773																				
3	2020-08-05	32	Jawa Barat	3216	Kabupaten Bekasi	5,536	2,127	33	4,001	1,535																				
4	2020-08-05	32	Jawa Barat	3201	Kabupaten Bogor	0	0	163	0	0																				
5	2020-08-05	32	Jawa Barat	3207	Kabupaten Ciamis	2,075	1,295	3	2,075	0																				
6	2020-08-05	32	Jawa Barat	3203	Kabupaten Cianjur	0	0	0	0	0																				
7	2020-08-05	32	Jawa Barat	3209	Kabupaten Cirebon	300	442	1	5	295																				
8	2020-08-05	32	Jawa Barat	3205	Kabupaten Garut	2,888	2,382	0	2,644	244																				
9	2020-08-05	32	Jawa Barat	3212	Kabupaten Indramayu	1,640	235	73	347	1,293																				
10	2020-08-05	32	Jawa Barat	3215	Kabupaten Karawang	6,100	1,437	57	25	6,075																				
11	2020-08-05	32	Jawa Barat	3208	Kabupaten Kuningan	1,990	471	1	1,683	307																				
12	2020-08-05	32	Jawa Barat	3210	Kabupaten Majalengka	671	420	7	11	660																				
13	2020-08-05	32	Jawa Barat	3218	Kabupaten Pangandaran	573	537	0	0	573																				
14	2020-08-05	32	Jawa Barat	3214	Kabupaten Purwakarta	691	532	1	5	686																				
15	2020-08-05	32	Jawa Barat	3213	Kabupaten Subang	5,788	455	0	1	5,787																				
16	2020-08-05	32	Jawa Barat	3202	Kabupaten Sukabumi	4,718	332	25	4	4,714																				
17	2020-08-05	32	Jawa Barat	3211	Kabupaten Sumedang	1,043	125	4	991	52																				
18	2020-08-05	32	Jawa Barat	3206	Kabupaten Tasikmalaya	0	0	0	0	0																				
19	2020-08-05	32	Jawa Barat	3273	Kota Bandung	6,780	3,553	0	6,780	0																				
20	2020-08-05	32	Jawa Barat	3279	Kota Banjar	586	187	0	475	111																				
21	2020-08-05	32	Jawa Barat	3275	Kota Bekasi	0	0	0	0	0																				
22	2020-08-05	32	Jawa Barat	3271	Kota Bogor	2,292	892	1	61	2,231																				
23	2020-08-05	32	Jawa Barat	3277	Kota Cimahi	856	3,926	0	330	526																				
24	2020-08-05	32	Jawa Barat	3274	Kota Cirebon	371	198	0	4	367																				
25	2020-08-05	32	Jawa Barat	3276	Kota Depok	6,080	3,106	0	428	5,652																				
26	2020-08-05	32	Jawa Barat	3272	Kota Sukabumi	331	1,445	2	2	329																				
27	2020-08-05	32	Jawa Barat	3278	Kota Tasikmalaya	1,364	692	51	1,364	0																				
28	2020-08-06	32	Jawa Barat	3204	Kabupaten Bandung	2	0	26	0	7																				
29	2020-08-06	32	Jawa Barat	3217	Kabupaten Bandung Barat	0	0	7	0	0																				
30	2020-08-06	32	Jawa Barat	3216	Kabupaten Bekasi	54	38	33	0	3,947																				

Record

Table stg\_covid\_data

status_id	status_name	status_detail
1	closecontact	closecontact_dikarantina
2	closecontact	closecontact_discarded
3	closecontact	closecontact_meninggal
4	confirmation	confirmation_meninggal
5	confirmation	confirmation_sembuh
6	probable	probable_diisolasi
7	probable	probable_discarded
8	probable	probable_meninggal
9	suspect	suspect_diisolasi
10	suspect	suspect_discarded
11	suspect	suspect_meninggal

Table dim\_case

province_id	province_name
32	Jawa Barat

Table dim\_province

district_id	province_id	district_name
3,205	32	Kabupaten Garut
3,279	32	Kota Banjar
3,277	32	Kota Cimahi
3,271	32	Kota Bogor
3,203	32	Kabupaten Cianjur
3,215	32	Kabupaten Karawang
3,212	32	Kabupaten Indramayu
3,210	32	Kabupaten Majalengka
3,213	32	Kabupaten Subang
3,273	32	Kota Bandung
3,206	32	Kabupaten Tasikmalaya
3,214	32	Kabupaten Purwakarta
3,208	32	Kabupaten Kuningan
3,202	32	Kabupaten Sukabumi
3,201	32	Kabupaten Bogor
3,204	32	Kabupaten Bandung
3,216	32	Kabupaten Bekasi
3,278	32	Kota Tasikmalaya
3,275	32	Kota Bekasi
3,217	32	Kabupaten Bandung Barat
3,272	32	Kota Sukabumi
3,207	32	Kabupaten Ciamis
3,274	32	Kota Cirebon
3,276	32	Kota Depok
3,211	32	Kabupaten Sumedang
3,218	32	Kabupaten Pangandaran
3,209	32	Kabupaten Cirebon

Table dim\_district

# Report table

www.website.com 17

pk_id	tanggal	123 province_id	123 case_id	123 total
1	2021-04-22	32	10	505
2	2021-02-16	32	7	1,639
3	2021-01-21	32	4	73
4	2021-09-05	32	5	804
5	2021-04-02	32	4	5
6	2020-09-26	32	2	416
7	2021-06-21	32	8	1,574
8	2021-08-16	32	6	276
9	2020-12-11	32	6	110
10	2020-09-18	32	10	73
11	2022-04-20	32	6	276
12	2021-10-15	32	5	100
13	2022-04-15	32	5	189
14	2021-05-25	32	10	242
15	2021-10-16	32	6	276
16	2021-11-19	32	6	276
17	2020-09-24	32	5	249
18	2021-05-12	32	7	2,819
19	2020-08-13	32	6	44
20	2021-03-23	32	10	157
21	2021-04-13	32	2	1,338
22	2021-02-17	32	2	621
23	2020-10-15	32	1	300
24	2021-04-18	32	6	133
25	2021-03-07	32	2	876
26	2021-04-23	32	7	2,692
27	2021-07-22	32	6	308
28	2020-10-30	32	5	351
29	2021-02-21	32	4	4
30	2021-08-28	32	4	29
31	2020-09-16	32	10	174

Table fact\_province\_dly

pk_id	123 province_id	123 month	123 case_id	123 total
1	32	202,107	8	55,661
2	32	202,102	7	45,956
3	32	202,107	4	4,011
4	32	202,012	8	29,159
5	32	202,108	1	2,814
6	32	202,110	4	72
7	32	202,010	7	19,096
8	32	202,104	7	78,107
9	32	202,202	6	7,728
10	32	202,109	6	8,280
11	32	202,111	4	41
12	32	202,106	8	46,990
13	32	202,012	10	11,201
14	32	202,104	8	43,190
15	32	202,008	10	69,611
16	32	202,112	4	19
17	32	202,107	10	11,636
18	32	202,109	4	1,146
19	32	202,103	7	62,921
20	32	202,204	5	32,471
21	32	202,010	10	5,908
22	32	202,101	5	46,452
23	32	202,010	4	299
24	32	202,107	2	77,926
25	32	202,008	6	1,585
26	32	202,011	11	142
27	32	202,008	2	30,488
28	32	202,107	7	109,934
29	32	202,011	1	12,918
30	32	202,101	7	43,572
31	32	202,103	4	795

Table fact\_province\_mth

pk_id	123 province_id	123 year	123 case_id	123 total
1	32	2,021	2	423,166
2	32	2,020	2	162,890
3	32	2,022	7	503,874
4	32	2,021	10	107,756
5	32	2,021	1	72,689
6	32	2,020	4	1,163
7	32	2,020	6	15,039
8	32	2,021	8	634,299
9	32	2,022	6	34,776
10	32	2,021	4	13,584
11	32	2,020	11	3,125
12	32	2,022	4	1,013
13	32	2,020	7	97,644
14	32	2,022	5	394,582
15	32	2,022	8	269,010
16	32	2,020	1	55,133
17	32	2,020	5	68,645
18	32	2,021	7	1,132,586
19	32	2,020	10	100,562
20	32	2,021	6	80,132
21	32	2,021	5	622,761
22	32	2,020	8	106,694

Table fact\_province\_yearly

# Report table

pk_id	tanggal	district_id	case_id	total
1	2020-08-26	3,206	8	8
2	2021-07-01	3,215	2	206
3	2022-02-27	3,274	5	59
4	2021-02-13	3,272	7	2
5	2020-12-06	3,278	6	7
6	2022-04-01	3,203	8	41
7	2021-04-11	3,216	2	144
8	2021-06-20	3,274	8	1
9	2022-03-07	3,214	7	1
10	2020-12-06	3,211	10	4
11	2021-01-09	3,217	4	1
12	2021-02-22	3,275	5	426
13	2022-03-21	3,278	6	4
14	2022-05-03	3,212	8	158
15	2021-11-13	3,211	5	2
16	2022-04-23	3,206	5	1
17	2021-08-17	3,202	8	248
18	2020-11-17	3,210	4	1
19	2021-07-30	3,204	4	2
20	2021-07-29	3,216	7	1,064
21	2021-05-26	3,201	6	17
22	2021-01-13	3,278	6	7
23	2022-02-18	3,273	7	150
24	2020-11-09	3,217	2	14
25	2021-03-12	3,207	7	3
26	2021-02-22	3,277	5	2
27	2020-11-19	3,277	8	9
28	2021-05-22	3,212	4	1
29	2021-03-19	3,206	8	9
30	2020-08-06	3,216	6	1
31	2022-04-08	3,215	5	9

Table fact\_district\_dly

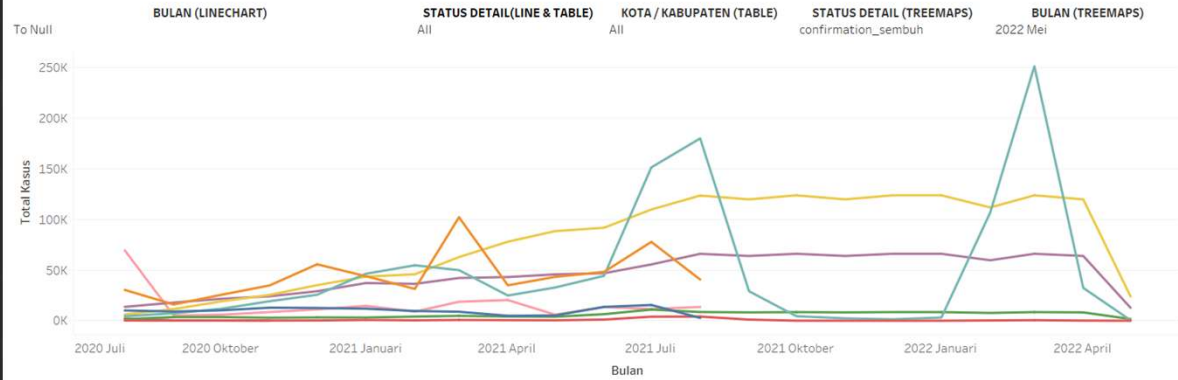
pk_id	district_id	month	case_id	total
1	3,273	202,103	2	2,182
2	3,274	202,111	8	30
3	3,203	202,109	7	2,460
4	3,211	202,107	2	202
5	3,215	202,110	5	90
6	3,206	202,107	7	186
7	3,209	202,204	5	269
8	3,213	202,010	5	44
9	3,277	202,102	2	393
10	3,277	202,105	5	828
11	3,203	202,203	4	14
12	3,271	202,202	7	308
13	3,208	202,011	1	15
14	3,213	202,112	4	1
15	3,271	202,201	4	3
16	3,216	202,109	8	4,230
17	3,278	202,011	4	19
18	3,271	202,009	8	1,528
19	3,210	202,105	4	17
20	3,271	202,009	1	244
21	3,209	202,104	4	4
22	3,273	202,103	10	1,825
23	3,212	202,109	5	771
24	3,278	202,009	7	1,982
25	3,271	202,108	1	31
26	3,272	202,008	5	54
27	3,276	202,104	4	213
28	3,216	202,103	2	3,174
29	3,201	202,203	8	12,338
30	3,209	202,008	2	310
31	3,214	202,009	5	58

Table fact\_district\_mth

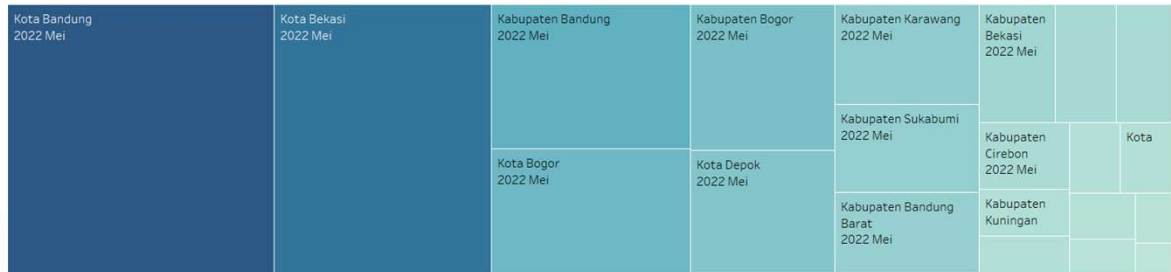
pk_id	district_id	year	case_id	total
1	3,205	2,020	11	9
2	3,208	2,021	2	20,939
3	3,272	2,021	8	9,541
4	3,208	2,020	8	683
5	3,208	2,021	5	12,181
6	3,274	2,021	4	513
7	3,205	2,022	8	504
8	3,201	2,020	4	16
9	3,278	2,020	4	28
10	3,275	2,020	1	1,672
11	3,279	2,021	6	305
12	3,205	2,022	5	7,000
13	3,275	2,020	10	11,301
14	3,278	2,020	8	1,250
15	3,271	2,020	1	1,510
16	3,204	2,022	5	19,112
17	3,204	2,020	8	4,478
18	3,278	2,021	5	12,385
19	3,273	2,022	5	48,198
20	3,203	2,021	4	199
21	3,205	2,020	5	2,385
22	3,213	2,020	7	2,670
23	3,202	2,020	6	274
24	3,276	2,020	1	5,574
25	3,217	2,020	2	1,502
26	3,218	2,022	7	756
27	3,274	2,020	8	125
28	3,279	2,021	1	296
29	3,213	2,020	11	62
30	3,210	2,022	8	13,860
31	3,207	2,021	10	59

Table fact\_district\_yearly





province_name	district_name	status_name	status_detail	2020	2021	2022
Jawa Barat	Kabupaten Bandung	closecontact	closecontact_dikarantina	1.392	568	
			closecontact_discarded	3.880	73.762	
		confirmation	confirmation_meninggal	30	580	23
			confirmation_sembuh	3.233	30.456	19.112
		probable	probable_diisolasi	45		
			probable_meninggal	4.478	11.315	3.906
	Kabupaten Bandung Barat	suspect	suspect_discarded	2.850	2.905	
			suspect_meninggal	158		
		closecontact	closecontact_dikarantina	773	1.130	
			closecontact_discarded	1.502	1.377	
		confirmation	confirmation_meninggal	15	251	31





# THANKYOU

<https://www.linkedin.com/in/rahmawaminggu/>

