Data Visualization for Food Price in DKI Jakarta

by Mikael Dewabrata

Based on Info Pangan website

Problem Definition

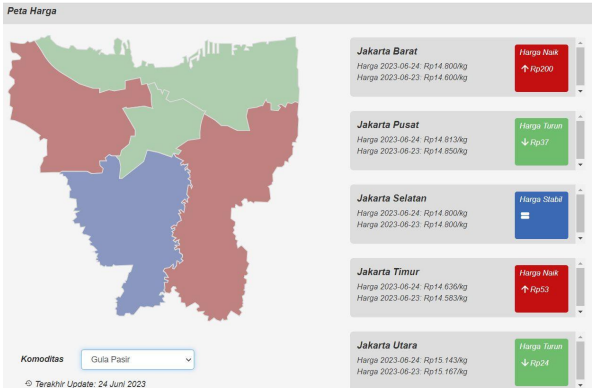
DKI Jakarta has a information portal about food price in Jakarta covering several traditional market. While the data is there, it is not presented well. This project is to ‘remake’ the local government data presentation to be more clear and reachable. Using this project, I am trying to recreate the data pipeline and present it with a more proper data visualization.

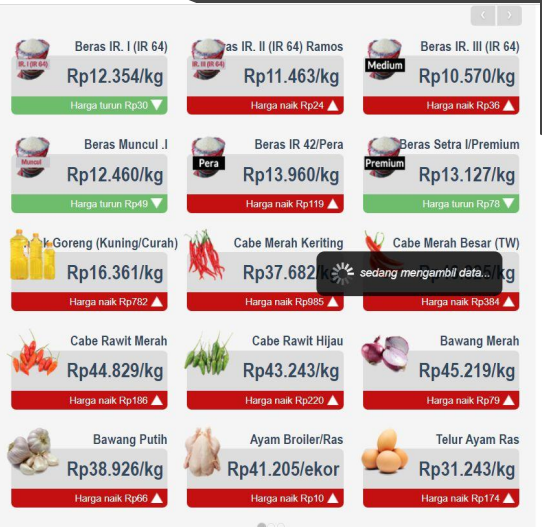
“Info Pangan DKI Jakarta is an excellent website, providing data for commodities in every traditional market in DKI Jakarta. However, the data is not well presented and quite slow.”

— Decision Making Questions

Data Selection

For this project, the data is taken from Informasi Pangan Jakarta (infopangan.jakarta.go.id). The information portal has complete pricing data from several traditional market in Jakarta.



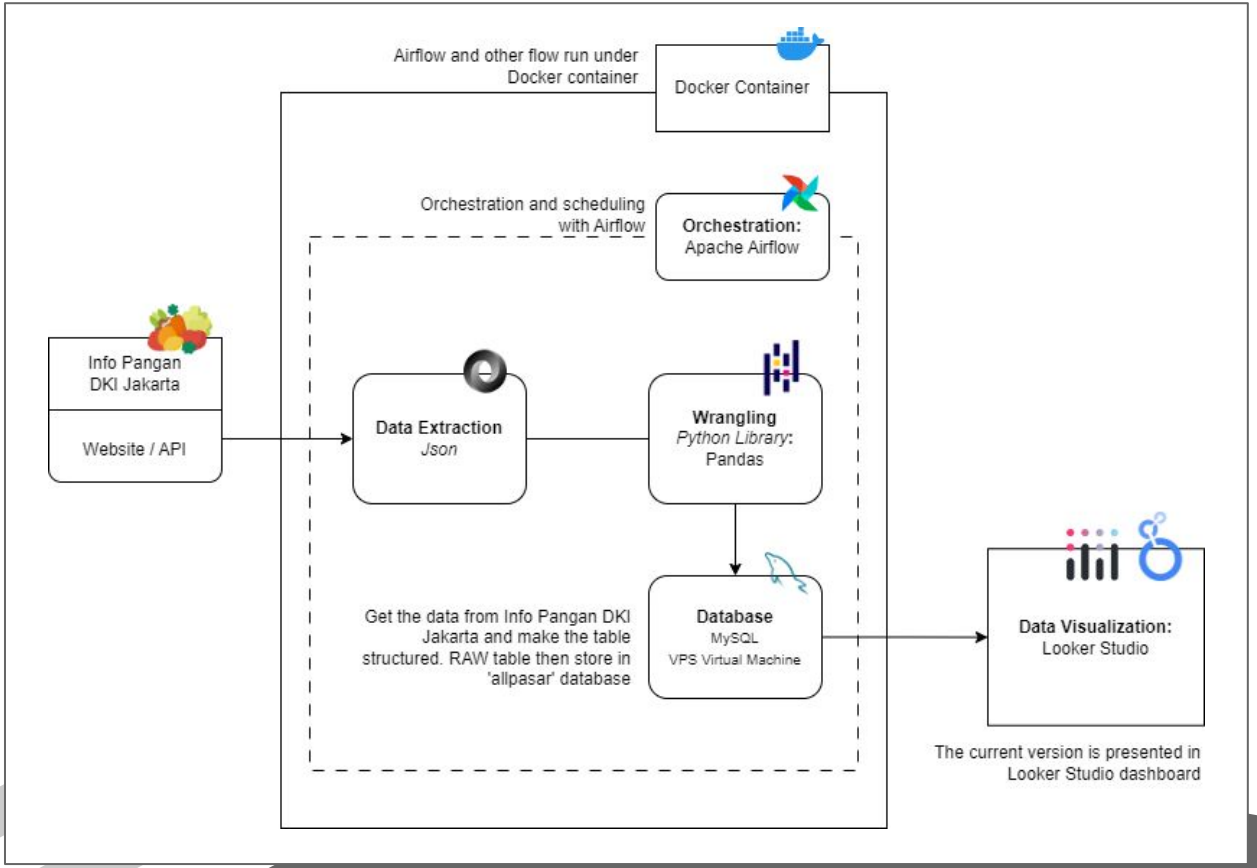


Data Problem

Though the data is updated regularly, the data is presented in very unstructured way. It's hard to read and not even using basic data visualization best practices.

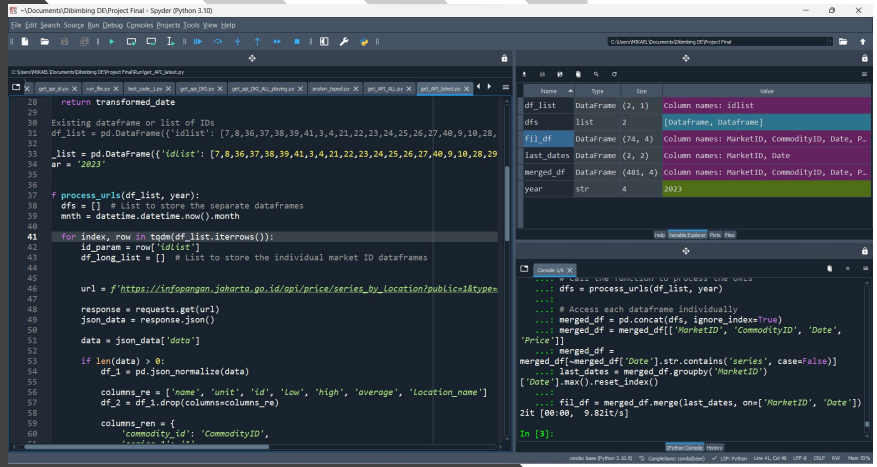
* taken from infopangan.jakarta.go.id

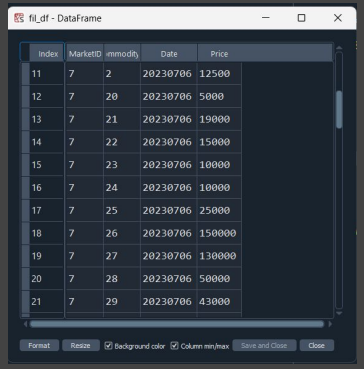
Data Pipeline



This system is run on Virtual Machine in a cloud service VPS.

The documentation can be seen in GitHub, including files and coding example. Data Visualization itself is going to be presented in Looker Studio, shared publicly.





Data Collection

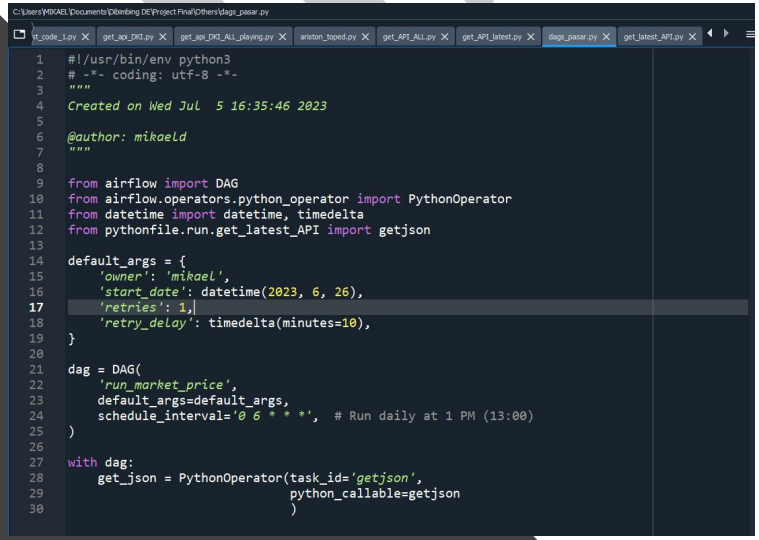
To get the data, no need for scraping since the website already provide the endpoint. Easily this can be done by getting the data with JSON schema.

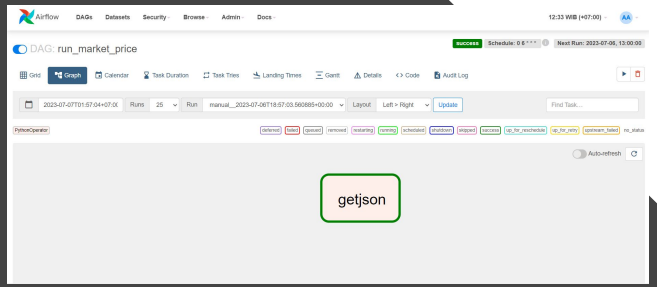
The more challenging process is to wrangle and clean the schema and make it a nicer table.

infopangan.jakarta.go.id/api/price/series_by_location?public=1&type=market&lid=41&m=6&y=2023

Index	Market ID	Market Name	Commodity	Date	Price
1	41	Pasar Jembatan Lima	Beras IR. II (IR 64) Ramos	20230607	13000
2	41	Pasar Jembatan Lima	Beras IR. II (IR 64) Ramos	20230621	35000

{"commodity_id":"2","name":"Beras IR. II (IR 64) Ramos","unit":"Rp","id":"2","low":"13000","high":"130000","average":"17720","location_name":"Pasar Jembatan Lima","series":{"1":"130000","2":"13000","3":"13000","4":"13000","5":"13000","6":"13000","7":"13000","9":"13000","10":"13000","11":"13000","12":"13000","13":"13000","16":"13000","17":"13000","18":"13000","19":"13000","20":"14000","21":"13000","22":"13000","24":"13000","25":"13000","26":"13000","27":"13000","28":"13000","29":"13000"}}



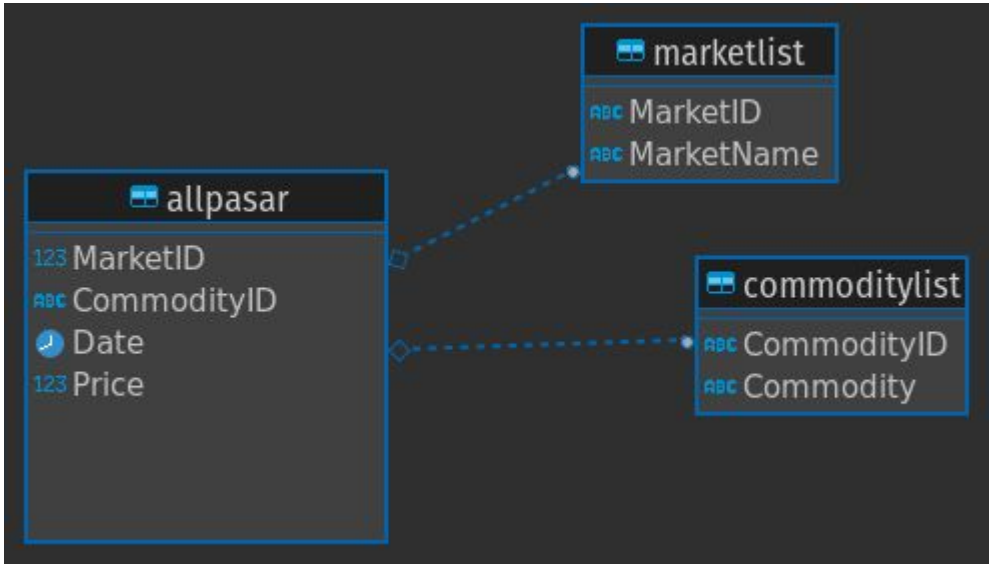


Data Orchestration

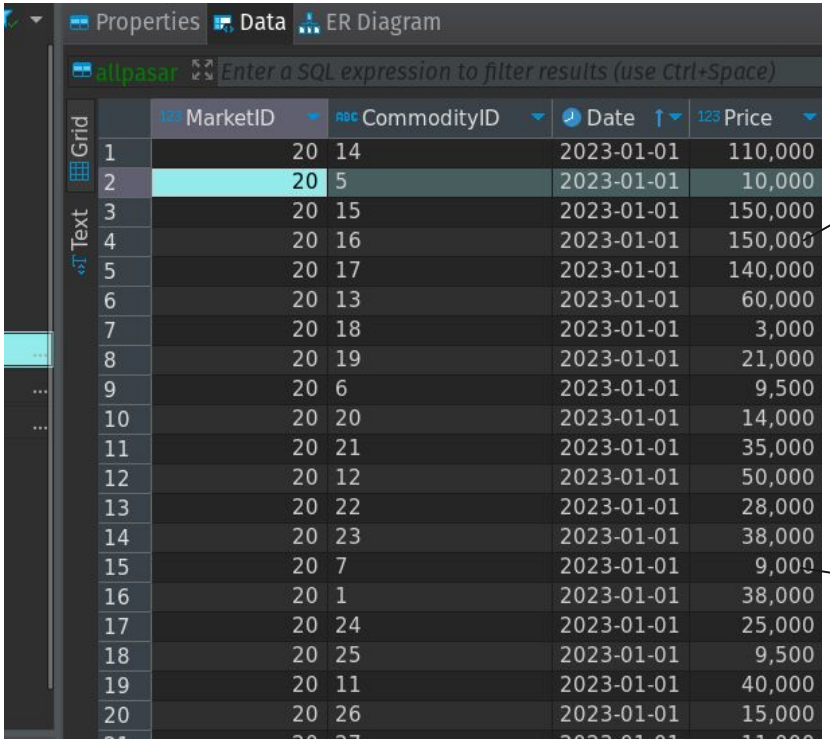
Using Airflow, the data then update daily so the dashboard can get the latest price for each commodity for each traditional market.

The Airflow is set under Docker container in VM environment using Linux Ubuntu.

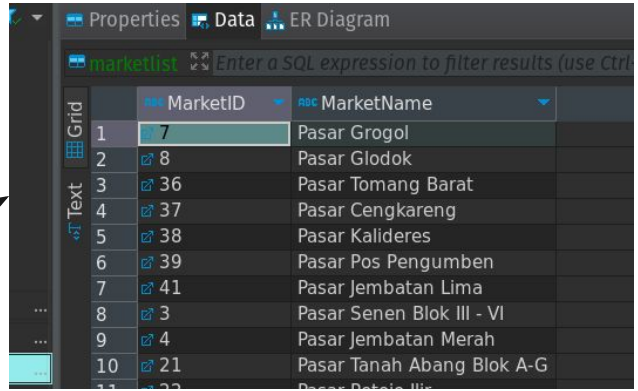
ERD - Data Pangan DKI



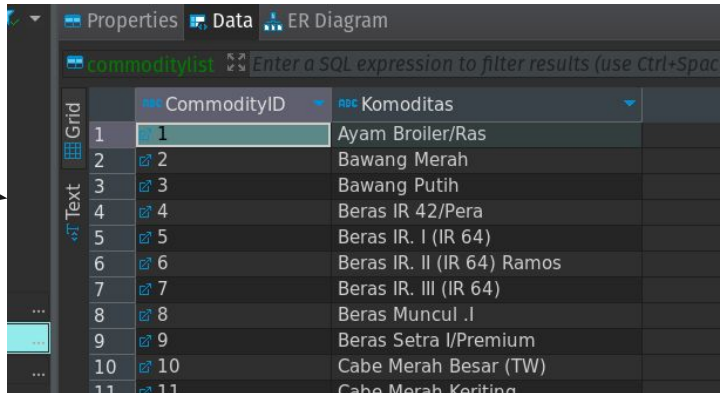
The ERD is pretty simple, only consist of few columns. The main table has Market ID, Commodity ID, Date, and Price. To see further details on this information, we can expand to other table consist of Market Name and Commodity.



	MarketID	CommodityID	Date	Price
1	20	14	2023-01-01	110,000
2	20	5	2023-01-01	10,000
3	20	15	2023-01-01	150,000
4	20	16	2023-01-01	150,000
5	20	17	2023-01-01	140,000
6	20	13	2023-01-01	60,000
7	20	18	2023-01-01	3,000
8	20	19	2023-01-01	21,000
9	20	6	2023-01-01	9,500
10	20	20	2023-01-01	14,000
11	20	21	2023-01-01	35,000
12	20	12	2023-01-01	50,000
13	20	22	2023-01-01	28,000
14	20	23	2023-01-01	38,000
15	20	7	2023-01-01	9,000
16	20	1	2023-01-01	38,000
17	20	24	2023-01-01	25,000
18	20	25	2023-01-01	9,500
19	20	11	2023-01-01	40,000
20	20	26	2023-01-01	15,000



	MarketID	MarketName
1	7	Pasar Grogol
2	8	Pasar Glodok
3	36	Pasar Tomang Barat
4	37	Pasar Cengkareng
5	38	Pasar Kalideres
6	39	Pasar Pos Pengumben
7	41	Pasar Jembatan Lima
8	3	Pasar Senen Blok III - VI
9	4	Pasar Jembatan Merah
10	21	Pasar Tanah Abang Blok A-G



	CommodityID	Komoditas
1	1	Ayam Broiler/Ras
2	2	Bawang Merah
3	3	Bawang Putih
4	4	Beras IR 42/Pera
5	5	Beras IR. I (IR 64)
6	6	Beras IR. II (IR 64) Ramos
7	7	Beras IR. III (IR 64)
8	8	Beras Muncul .I
9	9	Beras Setra I/Premium
10	10	Cabe Merah Besar (TW)

Data Pipeline Summary Understanding

Once the data extraction already defined, cleaning and wrangling and storing to MySQL need to be done. Using data orchestration, we can make the dashboard presenting the current data.

Summary:

- Creating an extraction flow from infopangan.jakarta.go.id
- Store raw the HTML or table to MySQL in raw database
- Connect to MySQL , retrieve raw for past pricing data and clean using Pandas to store in new database
- Create a scheduled scraping script to take recent data daily and store to new database (clean) database
- Data orchestration for worker and scheduler will be dockerized and using Airflow
- The clean data will be taken as data source for Looker Studio

Few notes on the pipeline

Python:

- Creating script to get the data from Info Pangan DKI Jakarta and then store it to MySQL database
- Clean and wrangling the data for visualization to make a better data visualization
- Using Dash Plotly app for visualization (soon)
- Library: requests, sqlalchemy, pandas

Airflow:

- Setup and dockerize Airflow in Virtual Machine in VPS
- Prepare dependencies for MySQL connector
- Create DAG to run scheduling to extract the data daily every 1 PM WIB
- Create error report or duplication in DAG (soon)

Docker:

- Install Docker in Virtual Machine
- Prepare the YAML for Airflow and Image for required library
- Compose Airflow container, run in background. The scheduler and worker are both running daily

THANK YOU