Data Lineage:

Of a dataset, describes the discrete steps involved in the creation, movement, and calculation of that dataset.

Why is data lineage important?

- **Instilling Confidence:** Being able to describe the data lineage of a particular dataset or analysis will build confidence in data consumers (engineer, analyst, scientist...) that our data pipeline is creating meaningful results using the correct datasets.

- **Defining Metrics:** allows everyone in the organization to agree on the definition of how a particular metric is calculated.

- **Debugging:** helps track down the root of errors when they occur. If each step is well defined and documented; it is easy to find.

Data Pipline Schedulers

Pipelines are often driven by schedules which determine what data should be analyzed and when.

Why Schedules?

- Pipline schedules can reduce the amount of data that needs to be processed on a given run. It helps scope the job to only run the data for the time period since the data pipline last ran. In a naive analysis, with no scope, we would analyze data all of the time.

- Using schedules to select data relevant to the time period of the given pipline execution can help improve the quality and accuracy of the analysis performed by our pipeline.

- Running Piplines on a schedule will reduce time it takes the pipeline to run.

Selecting the time period:

Determining the appropiate time period for a schedule is based on a number of factors which you need to consider as the pipline designer:

① What is the size of the data on average for a time period?

Depending on data Weight/time ; schedule will be more frequent or not.

② How frequently is data arriving and how often does the analysis need to be performed?

How up-to-date requires the information to be, by business, is a major requirement

③ Related datasets frequency?

Rule of Thumb → Frequency of a Pipline determined by the table with highest update frequency.

Data Partitioning:

Pipline data partitioning is the process of isolating data to be analyzed by one or more attributes, such as time, logical type, or data size.

→ Often leads to faster and more reliable pipelines.

Logical Partitioning: Breaks conceptually related data into discrete groups for processing.

Ex: Bikesharing → Split by: - trip
- stations

Size Partitioning: separates data for processing based on desired or required storage limits

Ex: each 1 GB (good to avoid Airflow Workers to break!

Data Quality:

Is the measure of how well a dataset satisfies its intended use.

Adherence to a set of requirements is a good starting point for measuring data quality.

→ Requirements should be defined by you and your data consumers before you start creating your data pipelines.

Requirements:

- Data must be a certain size
- Data must be accurate to some margin of error
- Data must arrive within a given timeframe from the start of execution.
- Pipelines must run on a particular schedule.
- Data must not contain any sensitive information.