**Data Ingestion and Processing**

Data ingestion is the initial step of a data pipeline where raw data is collected from various sources, like databases, APIs, or CSV files. The goal is to import and transfer this data to a centralized location for further processing and analysis. Once collected, the data undergoes processing, which involves cleaning, transforming, and enriching it. This includes tasks like dropping duplicates, handling missing values, and fixing typos.

**Data Pipelines (ETL / ELT)**

A data pipeline is a system for moving data from one point to another. The process is often described using the acronyms ETL or ELT.

**ETL** stands for **Extract, Transform, Load**.

* **Extract (E)**: This is the first step, where raw data is gathered from various sources, such as databases, APIs, or logs. A real-world example is collecting order data from an e-commerce website.
* **Transform (T)**: In this step, the data is cleaned, formatted, and validated. This includes tasks like removing duplicate orders or fixing missing information. Duplicate orders can happen due to system glitches or network retries.
* **Load (L)**: The final step where the cleaned data is stored in a target system, like a database or data warehouse, often for creating sales reports or other analyses.

In contrast,

**ELT** (Extract, Load, Transform) first loads the raw data into a warehouse, and the transformation happens later.

**Workflow Automation**

Workflow automation is the process of teaching a computer to perform a repetitive task automatically, saving time and reducing errors. A good analogy is an alarm clock: you set it once, and it rings automatically every day without you having to press a button. In data engineering, this is done by using Python scripts that run on a schedule, like every day at 9 AM, to ensure companies have fresh, clean data for their dashboards and reports.

**Data Processing Types**

There are different ways to process data, including batch processing and stream processing.

**Batch Processing**

This is a method where data is collected and stored over a period of time before being processed all at once as a single batch. It's efficient for handling large volumes of data. Examples include monthly payroll systems, where salaries are calculated and processed together at the end of the month, or generating bank statements at scheduled intervals. The main characteristic is that it has high latency, meaning there's a delay before the results are available.

**Stream Processing**

In this method, data is processed continuously as it's generated, one record at a time. This approach has very low latency and is essential for tasks that require immediate insights, such as real-time fraud detection, live dashboards, or stock market alerts. The system is always on and requires a highly available infrastructure. Errors can be detected as they happen, allowing for immediate action.