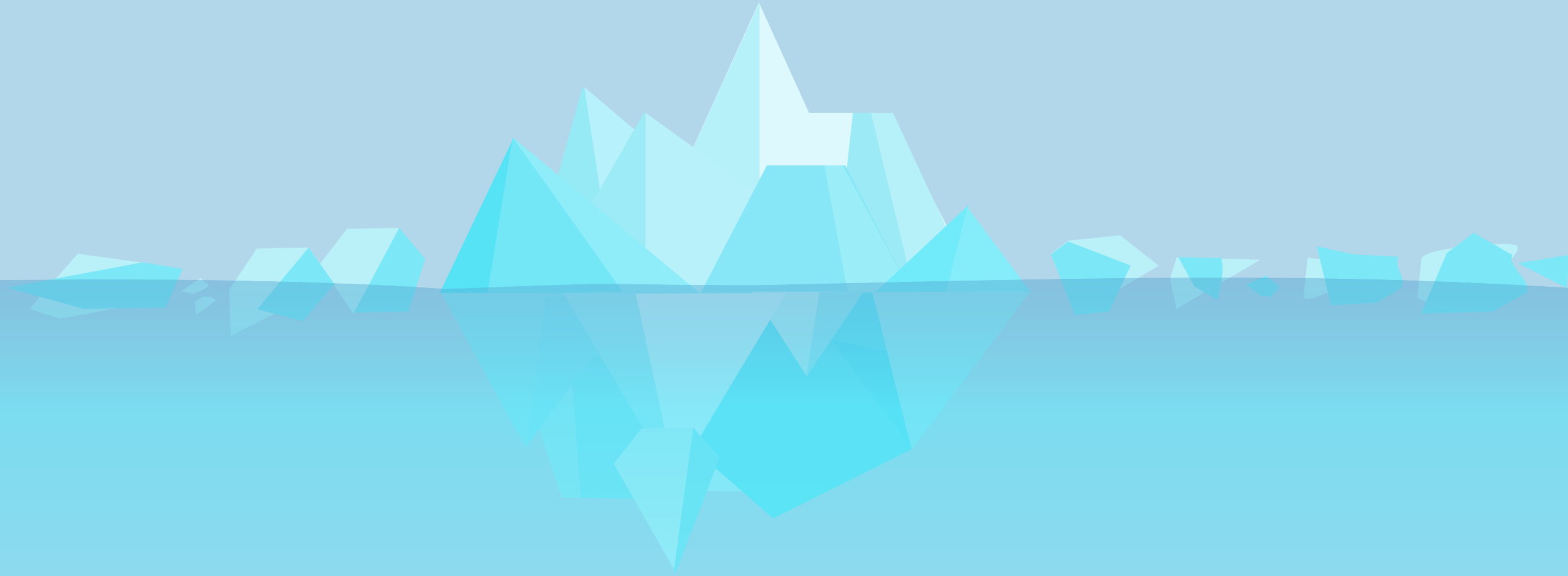
Iceberg of data science

A Titanic Journey into High level Data Science and Engineering



TOPICS OF INTEREST

Layered Approach

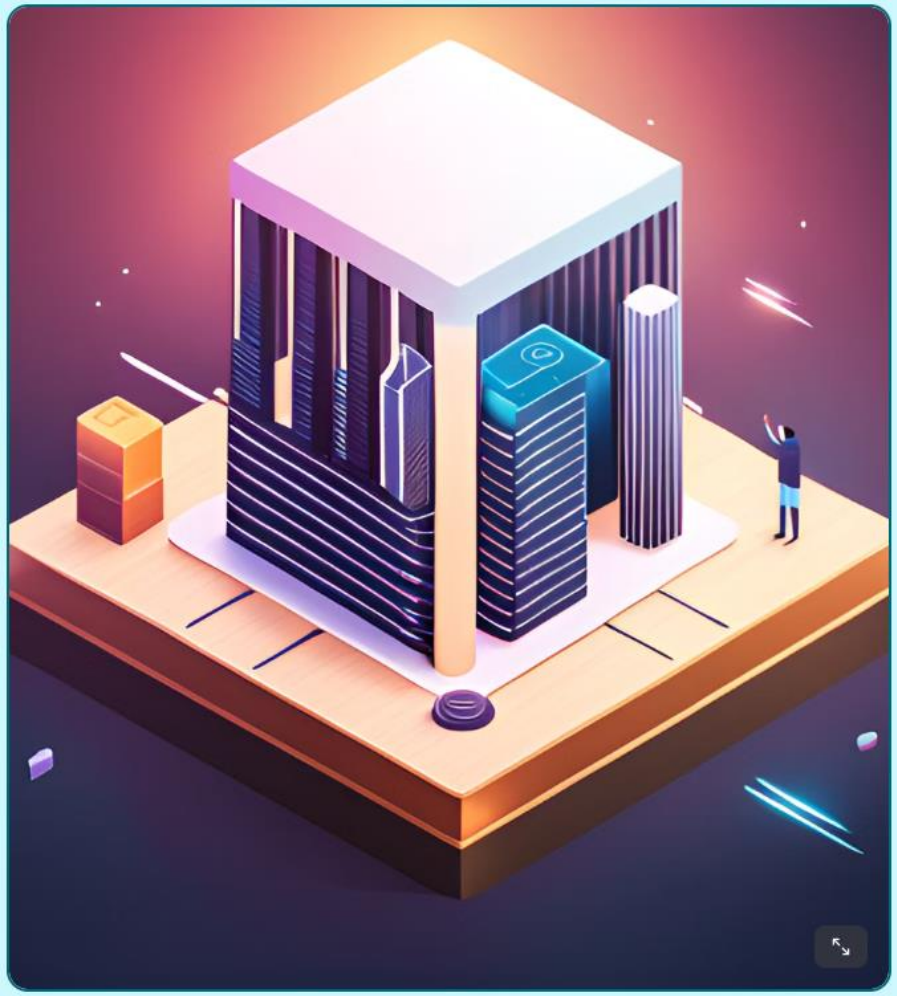
Data Collection

Data Processing

Data Storage

What is High-Level Data Engineering?

- High-Level Data Engineering focuses on designing and building end-to-end data systems that handle data at scale.
- It involves collecting, processing, storing, and analyzing data to generate actionable insights for decision-making.
- The goal is to build scalable, fault-tolerant, and maintainable data infrastructure.



What is a Layered Approach?

- A layered framework divides the data lifecycle into distinct layers, making the system more modular and scalable.
- Each layer focuses on a specific function (e.g., collection, processing, storage).
- Advantages:
  - Modularity: Easier maintenance
  - Scalability: Seamless growth
  - Reusability: Shared components
  - Fault Tolerance: Isolated failures

Types of Data Collected

STRUCTURED

SEMI STRUCTURED

UNSTrUCTURed

Methods of Data Collection

- **Batch Ingestion**: Data is collected at scheduled intervals, typically used when immediate access is not required, reducing resource usage.
- **Real-Time Streaming**: Data is ingested continuously as it is generated, enabling immediate processing and insights for time-sensitive applications.
- **Event-Driven Collection**: Data is gathered when specific events occur, allowing systems to respond dynamically to triggers or changes in conditions.

Challenges in Data Collection

- **Data Quality:** Ensuring completeness, accuracy, and reliability of data to avoid errors and improve decision-making.
- **Diverse Formats**: Handling various types of data, including structured, semi-structured, and unstructured formats, to ensure compatibility across systems.
- **Scalability:** Managing large volumes of incoming data efficiently to prevent bottlenecks and ensure smooth operation.

Tools for Data Collection

- **Logstash**: A powerful log collection tool that processes and transforms logs from various sources for further analysis.
- **Scrapy**: An open-source web scraping framework that efficiently extracts data from websites for various applications.
- **Flume**: A tool designed for collecting, aggregating, and transferring large volumes of log data to storage systems.

Overview of the Data Processing Layer

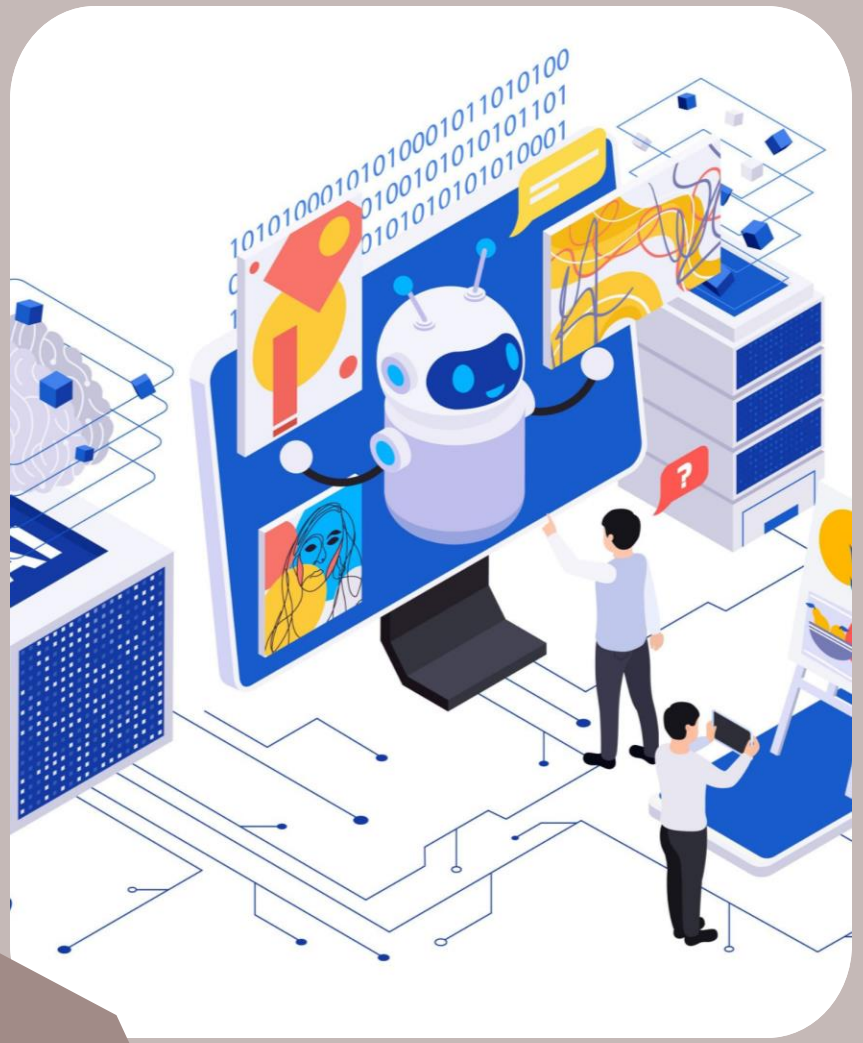
- The Data Processing Layer is where raw data is cleaned, transformed, and prepared for further use.
- This layer ensures that data quality and integrity are maintained.
- Tasks include:
  - Data cleaning: Handling missing values, duplicates
  - Data transformation: Changing formats, aggregating data
  - Data validation: Ensuring accuracy and consistency.

Steps in Data Processing

1. Data Ingestion: The initial step of collecting raw data from various sources for processing.

Data can be collected from various sources :

- Databases (SQL, NoSQL)
- Files (CSV, JSON, XML)
- Sensors

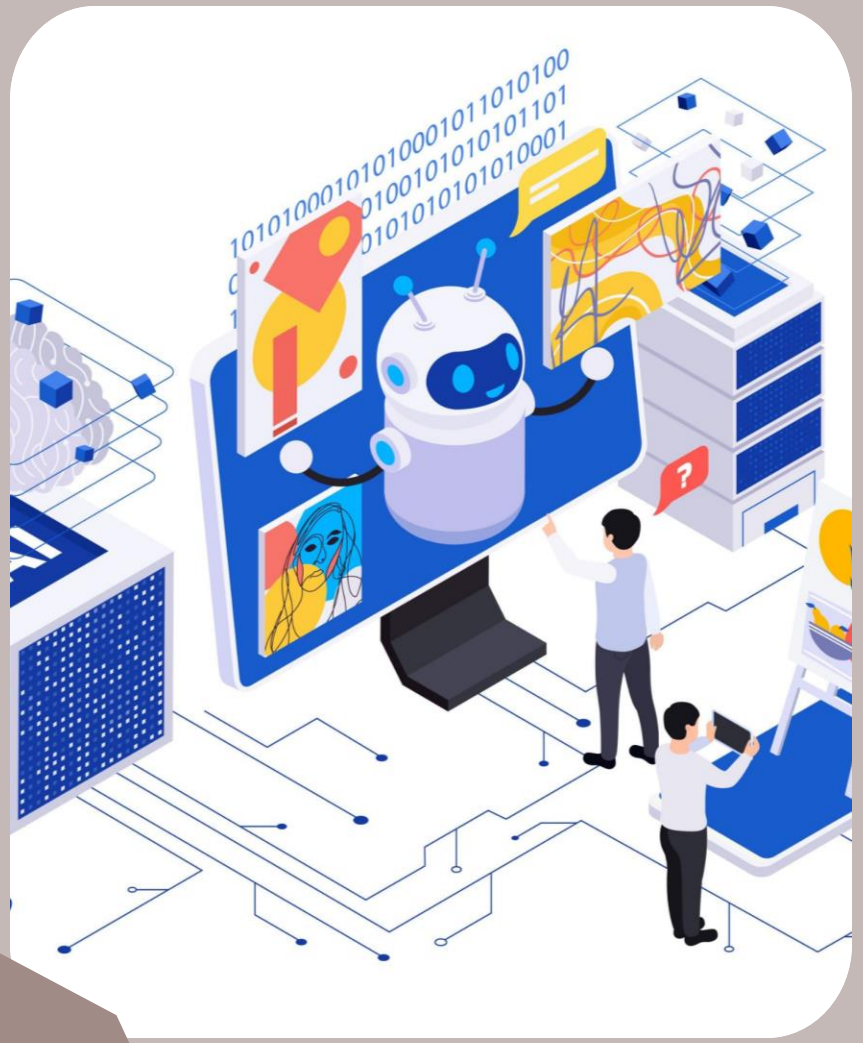


Steps in Data Processing

- Data cleaning: Identifying and correcting errors, inconsistencies, and missing values in the data to improve accuracy.

Common Issues In Data Cleaning:

- Missing values
- Duplicate Entries
- Inconsistent Formats
- Irrelevant Data
- Typographical Errors



Dataset Before Cleaning

Transaction_ID	Customer_Name	Amount	Date	Product_ID
1	John Smith	500	2023-01-01	101
2	Jane Doe		01-01-2023	102
3	John Smith	500	2023-01-01	101
5	NULL	700	2023-01-03	104

Final Dataset After Cleaning

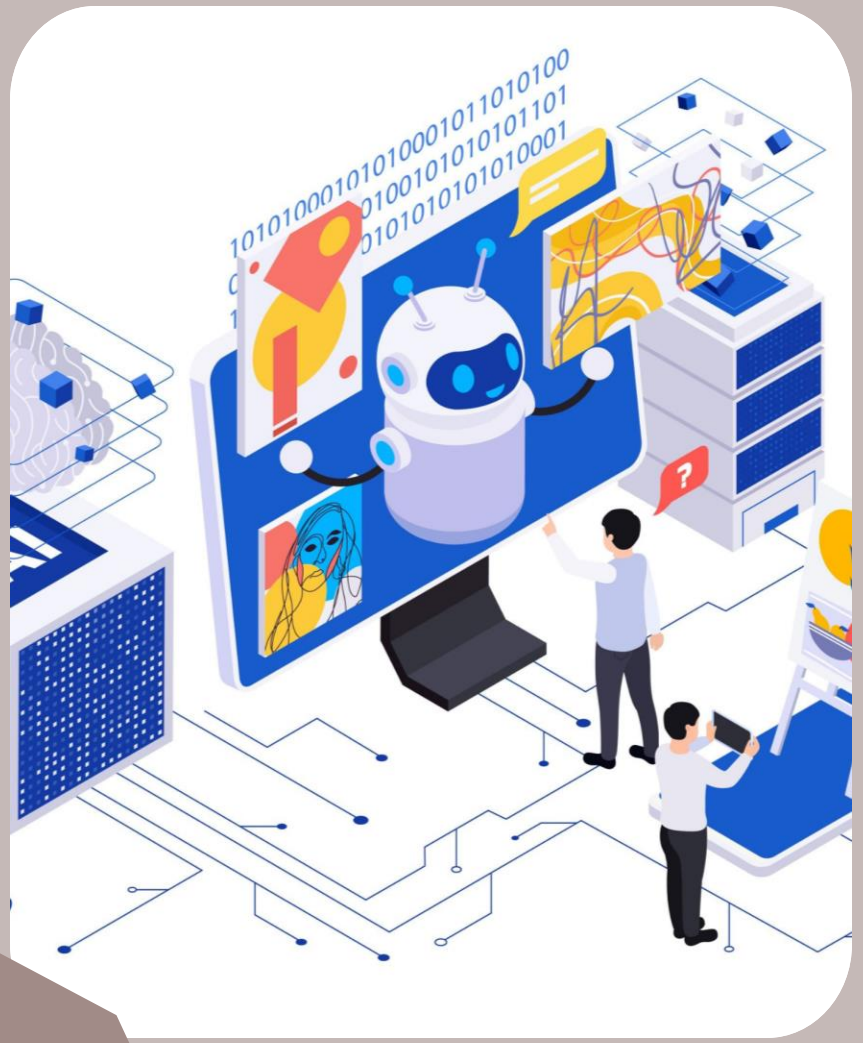
Transaction_ID	Customer_Name	Amount	Date	Product_ID
1	John Smith	500.0	2023-01-01	101
2	Jane Doe	600.0	2023-01-01	102
5	Unknown	700.0	2023-01-03	104

Steps in Data Processing

- Data Transformation: Converting data into usable formats, aggregating it, and applying business rules to make it analysis-ready.

Key Steps:

- Data Conversion
- Data Aggregation
- Feature Engineering
- Encoding
- Applying Business Rules



Original Dataset

Transaction_ID	Customer_Name	Amount	Date	Product_Category
1	John Smith	500	2023-01-01	Electronics
2	Jane Doe	300	2023-01-02	Clothing
3	Alice Brown	200	2023-01-02	Electronics
4	Bob Johnson	400	2023-01-03	Furniture

Dataset After Feature Engineering

Transaction_ID	Customer_Name	Amount	Date	Product_Category	Day_of_Week
1	John Smith	500	2023-01-01	Electronics	Sunday
2	Jane Doe	300	2023-01-02	Clothing	Monday
3	Alice Brown	200	2023-01-02	Electronics	Monday
4	Bob Johnson	400	2023-01-03	Furniture	Tuesday

Dataset After Encoding

Transaction_ID	Customer_Name	Amount	Date	Day_of_Week	Normalized_Amount	Product_Category_Clothing	Product_Category_Electronics	Product_Category_Furniture
1	John Smith	500	2023-01-01	Sunday	1.0	0	1	0
2	Jane Doe	300	2023-01-02	Monday	0.25	1	0	0
3	Alice Brown	200	2023-01-02	Monday	0.0	0	1	0
4	Bob Johnson	400	2023-01-03	Tuesday	0.5	0	0	1

Dataset Before Applying Business Rules:

Order ID	Customer Name	Total Order Value ($)	Order Date
101	Alice	45	2025-01-20
102	Bob	120	2025-01-21
103	Charlie	550	2025-01-22
104	Diana	30	2025-01-23
105	Eve	500	2025-01-24

Dataset After Applying Business Rules:

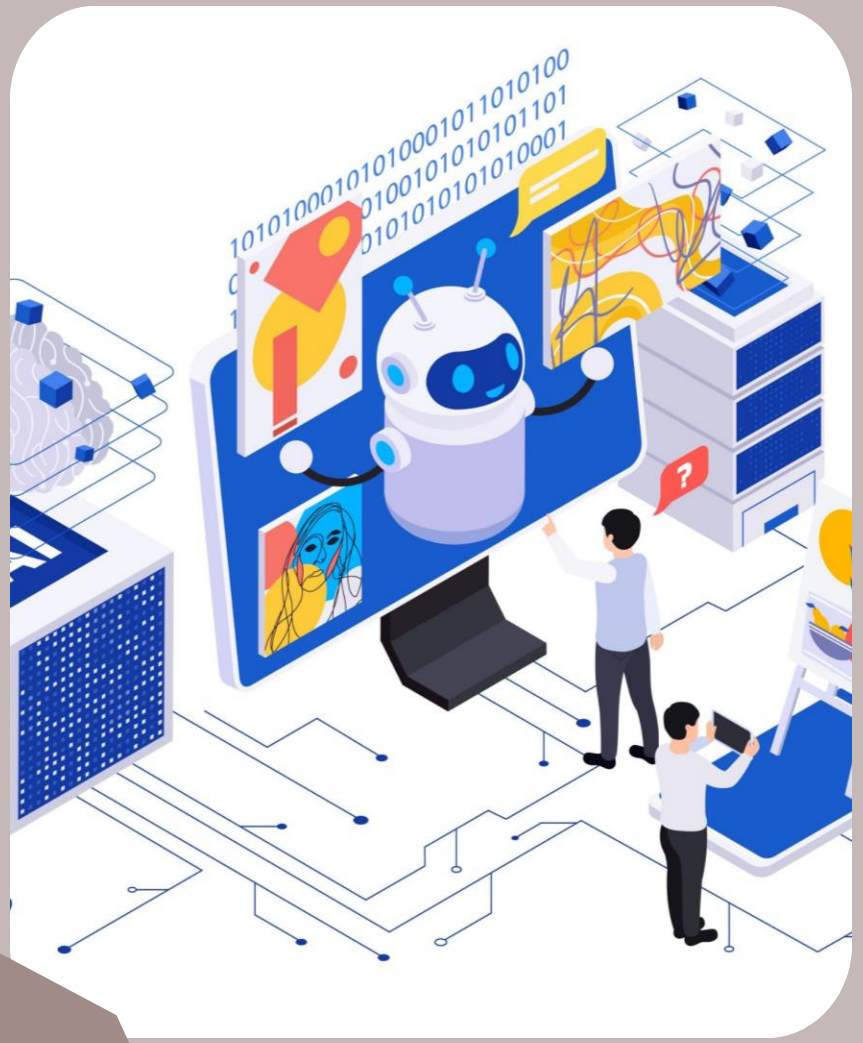
Order ID	Customer Name	Total Order Value ($)	Order Date	Customer Category
102	Bob	120	2025-01-21	Regular
103	Charlie	550	2025-01-22	Premium
105	Eve	500	2025-01-24	Regular

Steps in Data Processing

- Data Validation: Ensuring that processed data meets accuracy, consistency, and completeness standards.

Example:
A company is preparing a customer database for a marketing campaign. Data must be ensured of:

- Accuracy
- Consistency
- Completeness

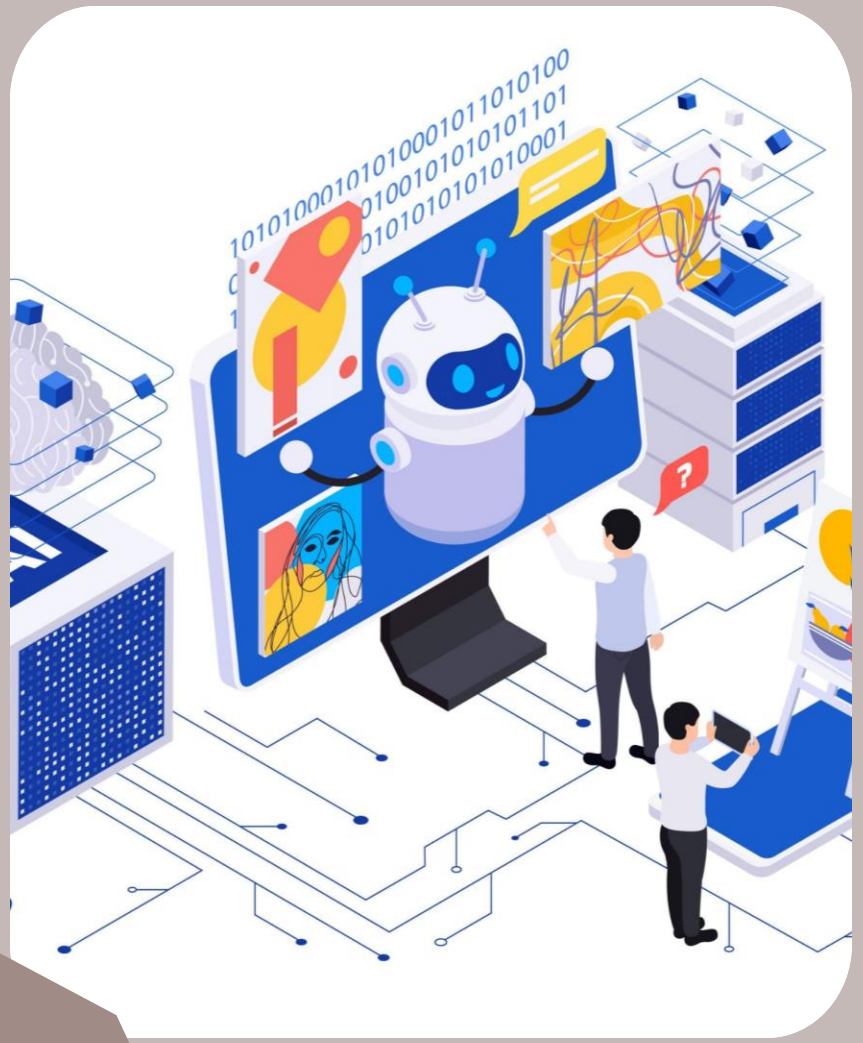


Steps in Data Processing

- Data Enrichment: Enhancing datasets by adding relevant external data to provide more context and value

Types of Data Enrichment:

- Demographic
- Geographic
- Behavioral
- External Data



Initial Dataset (Before Enrichment):

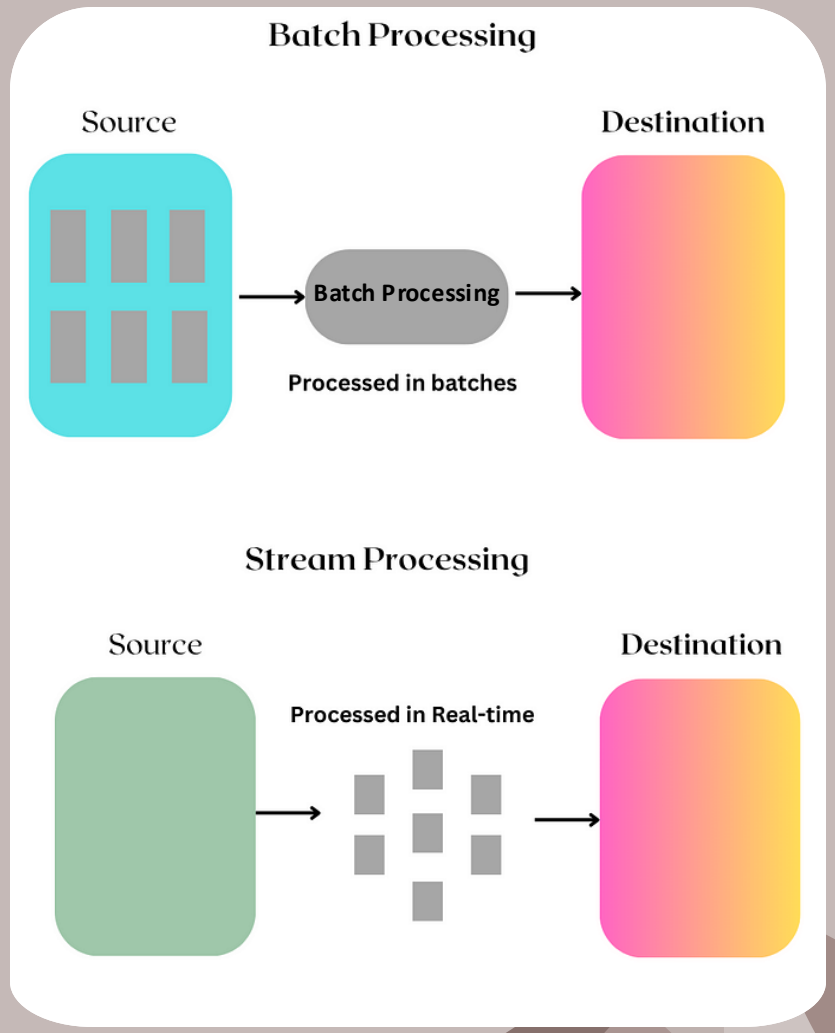
Customer ID	Name	Email	City
1	Alice	alice@example.com	New York
2	Bob	bob@example.com	Los Angeles
3	Charlie	charlie@example.com	Chicago
4	Diana	diana@example.com	Houston

Enrichment Data (External Source):

City	Population	Median Income ($)	Region
New York	8,336,817	68,486	Northeast
Los Angeles	3,979,576	62,474	West
Chicago	2,693,976	55,295	Midwest
Houston	2,320,268	52,338	South

Dataset After Enrichment:

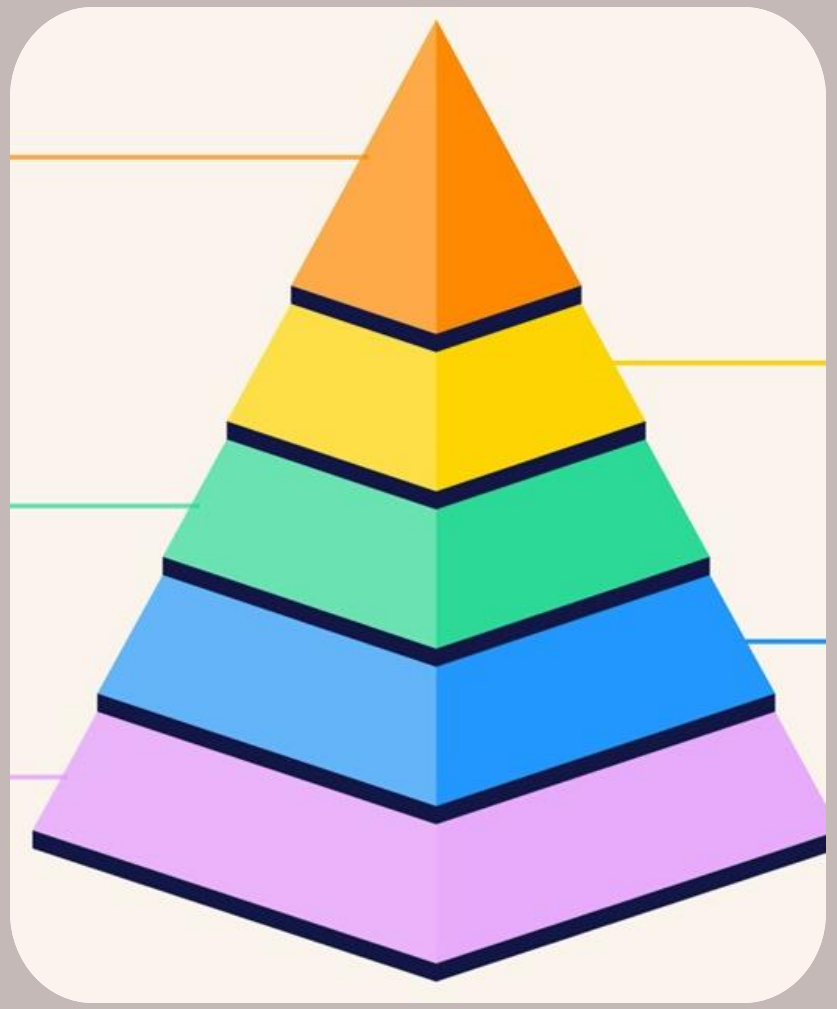
Customer ID	Name	Email	City	Population	Median Income ($)	Region
1	Alice	alice@example.com	New York	8,336,817	68,486	Northeast
2	Bob	bob@example.com	Los Angeles	3,979,576	62,474	West
3	Charlie	charlie@example.com	Chicago	2,693,976	55,295	Midwest
4	Diana	diana@example.com	Houston	2,320,268	52,338	South



Types of Data Processing

- Batch Processing: Processing large datasets in chunks at scheduled intervals, ideal for tasks that don’t require immediate results.

- Stream Processing: Continuously processing data as it is generated, suitable for real-time analytics and monitoring systems.

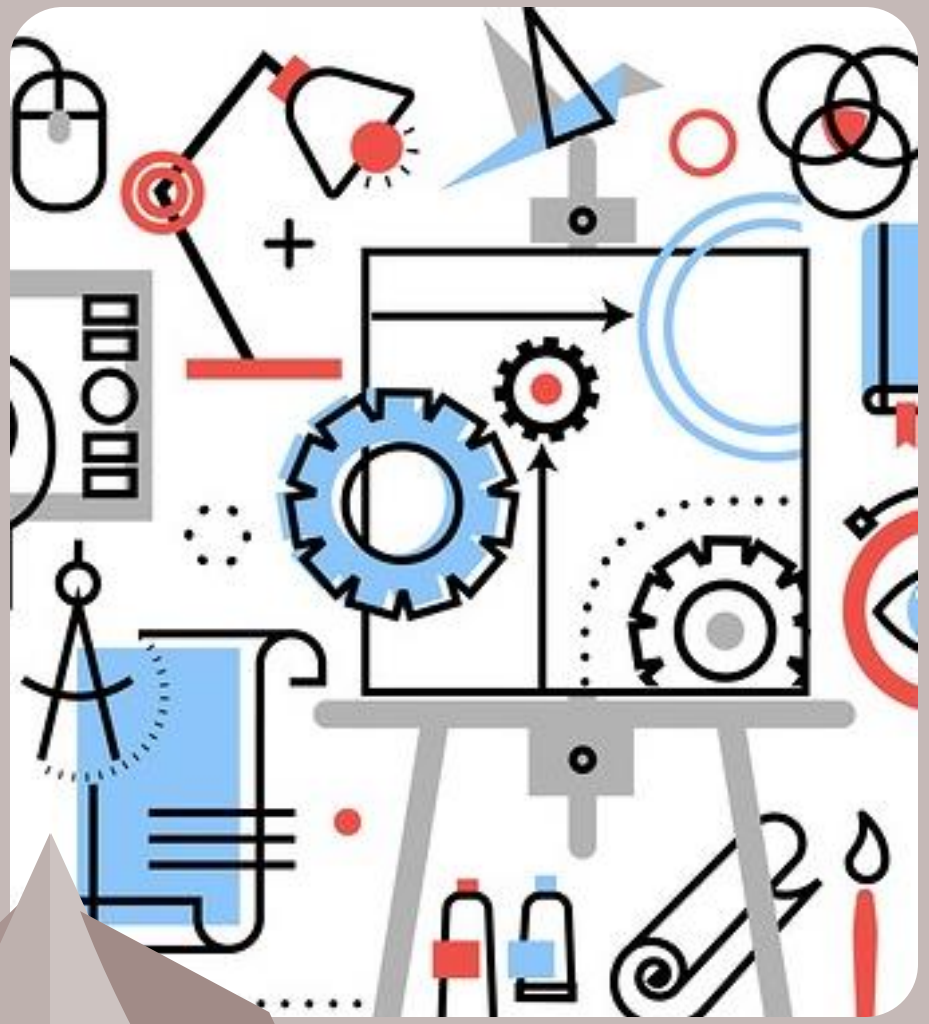


Challenges in Data Processing

- Scalability: Ensuring that the system can handle growing volumes of data without performance degradation.
- Data Quality: Maintaining the accuracy, completeness, and consistency of data throughout the processing stages.
- Fault Tolerance: Ensuring the system can recover from hardware failures, network issues, or software bugs without data loss.

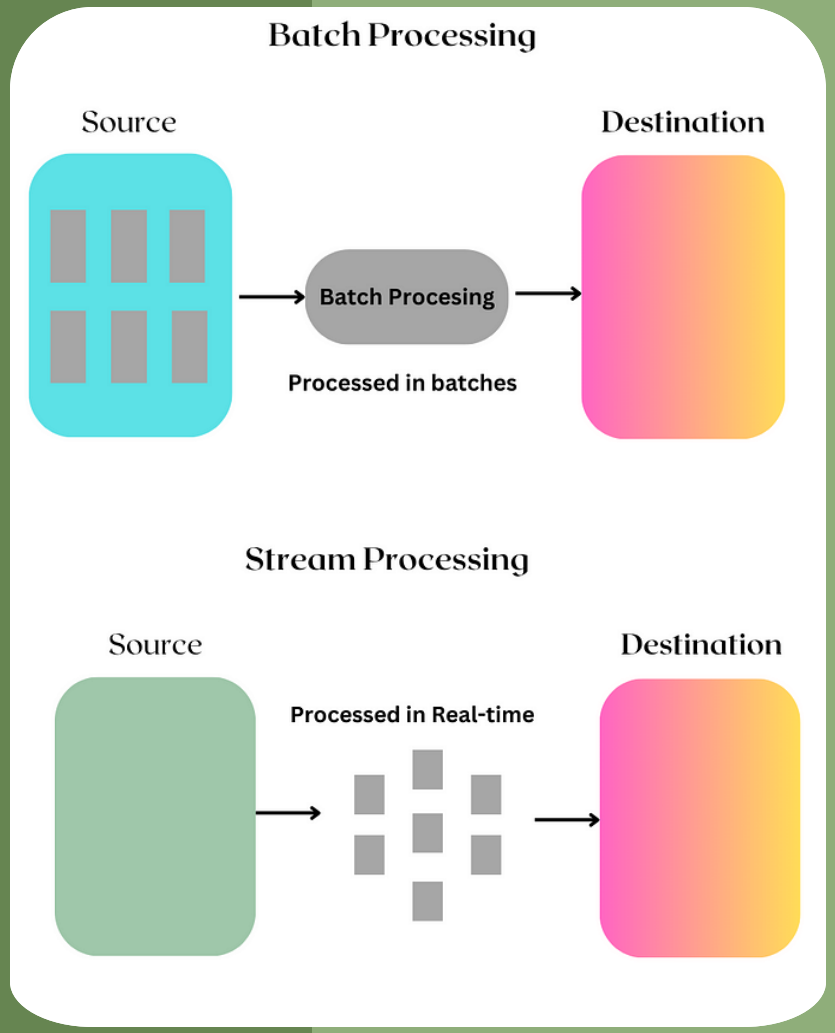
Tools for Data Processing

- A powerful distributed processing engine that handles large-scale data processing tasks across clusters.
- A workflow orchestration tool that automates the scheduling and execution of data pipelines.
- Python library for data manipulation and analysis, suitable for handling structured data.



Overview of the Data Storage Layer

- The Data Storage Layer is responsible for storing raw, processed, and analytical data in a secure and accessible manner.
- It ensures that data is available for future use, supports querying, and maintains data security and compliance.
- Types of storage:
  - Data warehouses: Optimized for storing structured data for analytics and reporting.
  - Data lakes: Designed to store raw, unstructured, and semi-structured data for flexible exploration.
  - Databases: Used for transactional data storage and ensuring fast, reliable access to structured data.

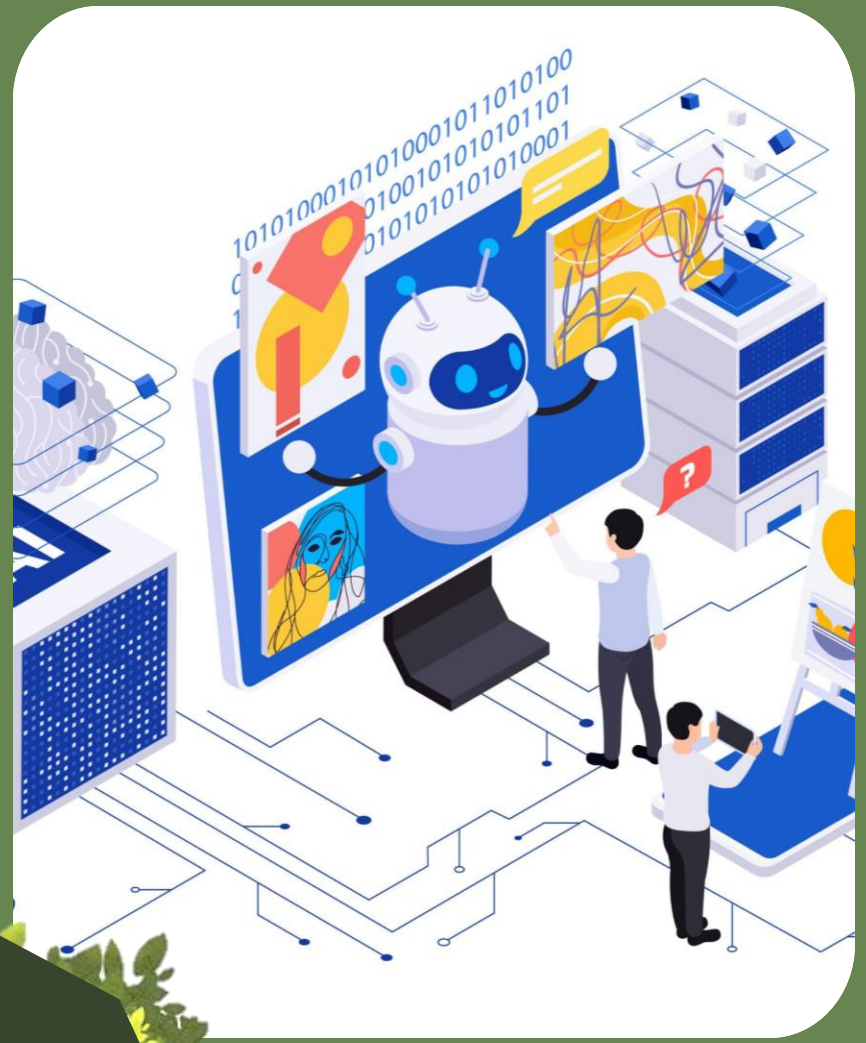


Types of Data Storage Systems

- Data Warehouse: Structured storage systems designed for analytics, examples include Snowflake and Amazon Redshift.
- Data Lake: Flexible storage solutions that keep data in its original format, examples include AWS S3 and Azure Data Lake.
- Databases: SQL-based databases like PostgreSQL and MySQL that store structured data and Non-relational databases like MongoDB and Cassandra that are optimized for handling unstructured and semi-structured data.

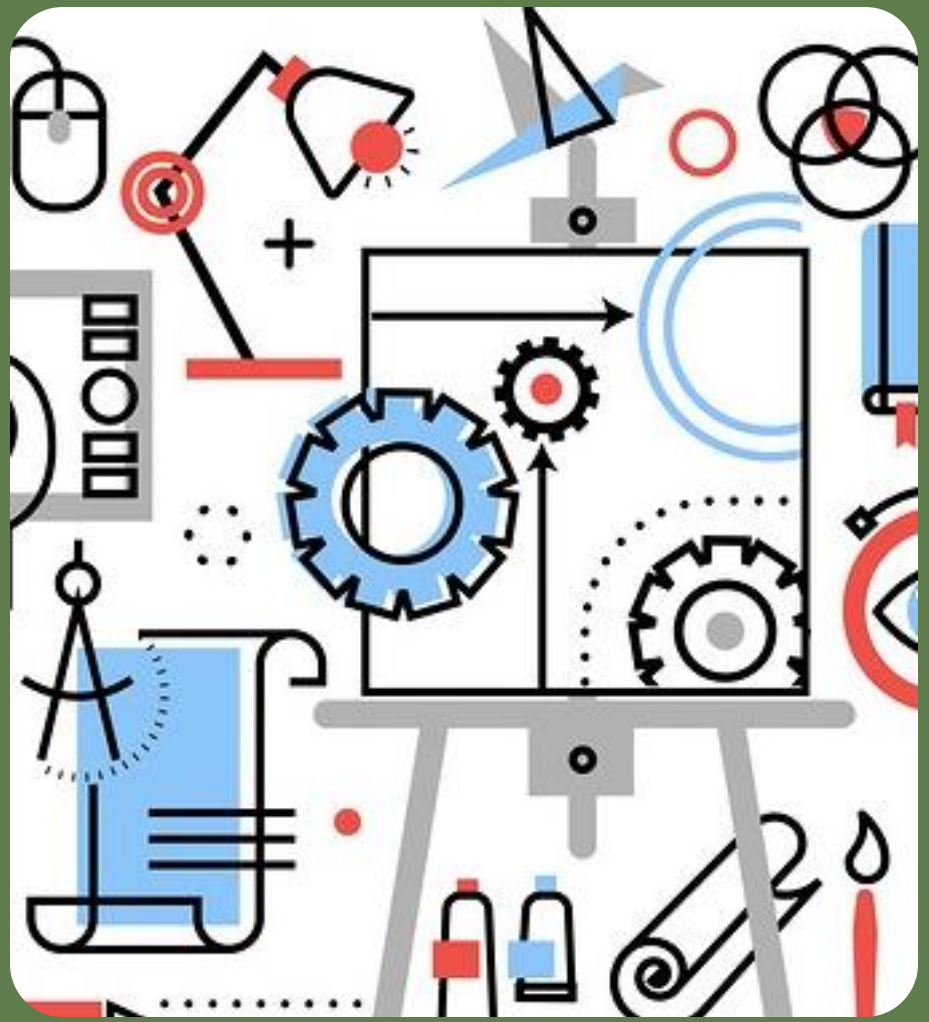
Challenges in Data Storage

- Data Security: Ensuring that sensitive data is protected from unauthorized access and breaches.
- Scalability: Managing the growing volume of data while maintaining performance and availability.
- Data Governance: Ensuring compliance with regulations and maintaining control over data access and usage.



Tools for Data Storage

- **Snowflake**: A cloud-based data warehouse that provides scalability and supports complex queries.
- S3: A scalable object storage service that provides secure and durable data storage.
- Databases: SQL based Dbs like postgresql, mysql, etc. and NoSQL Dbs like MongoDB, Cassandra or Graph Dbs like Neo4j and more.



MCS

Q1: Which of the following best describes the Event-Driven Collection method?

A. Collects data at regular, scheduled intervals.

B. Continuously ingests data as it is generated.

C. Collects data when specific events or triggers occur.

D. Processes data in distributed systems across multiple nodes.

MCS

Q2: Which tool is commonly used for real-time data streaming in the data collection layer?

A. Flume

B. Kafka

C. Logstash

D. Scrapy

MCS

Q3: What is a primary challenge in the data processing layer?

A. Ensuring data encryption

B. Handling diverse data formats

C. Reducing latency in real-time processing

D. Storing raw and analytical data

MCS

Q4: Which type of data storage system is best suited for storing unstructured or semi-structured data in its original format?

A. Data Warehouse

B. Relational Database

C. NoSQL Database

D. Data Lake

MCS

Q5: Which of the following is a best practice in the data storage layer?

A. Automate ingestion workflows to reduce manual intervention.

B. Use real-time monitoring to track incoming data.

C. Implement data validation at the source.

D. Ensure data encryption to protect sensitive information.