DATA ENGINEERING

1. Difference between ELT and ETL?

ELT(Extract, Load, Transform)	**ETL**(Extract, Transform, Load)
1) Data is extracted from various source systems.	1) Data is extracted from various source systems.
2) The raw data is loaded directly into the target system.	2) The extracted data is then transformed in a staging area. This includes data cleansing, filtering, aggregating, and converting into the desired format.
3) The data is transformed within the target system, often using its native processing power.	3) The transformed data is loaded into the target data warehouse or database.
4) ELT is better suited for modern, scalable data systems where leveraging the processing power of the target database can streamline and expedite the data integration process.	4) ETL is ideal for environments where data quality and complex transformations are critical before data is stored.

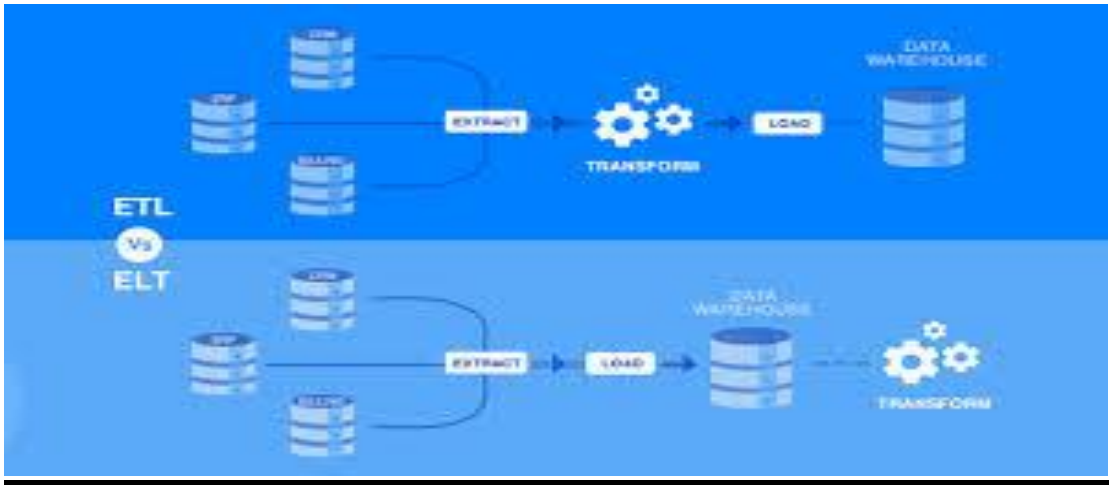
Use Case of ELT:

- Modern cloud-based data warehouses and big data platforms.
- Scenarios with high data volumes where loading speed is crucial.
- Environments where the target system has significant processing capabilities and can handle complex transformations efficiently.

Use Case of ETL:

- Traditional data warehousing.
- Scenarios where data quality and transformation complexity are critical.
- Systems with limited processing power in the target database.

Architecture:



Batch vs Streaming Pipeline

Batch Pipeline	Streaming Pipeline
Processes data in large chunks or batches at scheduled intervals.	Processes data continuously and incrementally as it arrives.
Higher latency as data is collected over a period and then processed.	Low latency with near real-time data processing capabilities.
Generally simpler to implement and manage compared to streaming pipelines.	More complex to implement and manage due to the continuous nature of data flow.
Use Cases: Suitable for tasks that do not require real-time data processing, such as end-of-day reports, data warehousing, ETL (Extract, Transform, Load) jobs, and offline analytics.	**Use Cases**: Suitable for applications that require real-time data processing, such as live analytics, monitoring, fraud detection, real-time recommendations, and IoT applications.

Advantages of Batch Pipeline:

- **Resource Efficiency**: Optimized for large-scale data processing, often with better resource utilization.
- **Error Handling**: Easier to manage and debug as the entire batch can be reprocessed if errors occur.
- **Cost**: Often more cost-effective for processing large volumes of data due to the lower frequency of execution.

Advantages of Streaming Pipeline:

- **Real-time Processing**: Provides immediate insights and actions based on data as it arrives.
- **Data Freshness**: Ensures data is always up-to-date.
- **Event-driven**: Can respond to individual events as they occur.

Architecture: