What is the ETL process?

ETL, which stands for extract, transform and load, is a data integration process that combines data from multiple data sources into a single, consistent data store that is loaded into a data warehouse or other target system.

Extract: During data extraction, raw data is copied or exported from source locations to a staging area. Data management teams can extract data from a variety of data sources, which can be structured or unstructured.

Transform: In the staging area, the raw data undergoes data processing. Here, the data is transformed according to its intended analytical use case. This phase can involve cleaning, filtering, aggregating, or joining data, among other things.

Load: In this step, the transformed data is loaded into the target system, such as a data warehouse, where it can be easily accessed and analyzed

What is the ELT process?

ELT stands for Extract, Load, and Transform. It is a process that is similar to ETL, but with the order of the steps reversed.

In the ELT process, data is first extracted from various sources, such as databases, flat files, web services, or other data sources, and then loaded into a target system, such as a data lake. Once the data is loaded, it is transformed into a format that can be easily processed and analyzed.

The main difference between ETL and ELT is the timing and location of the data transformation step. In ETL, data is transformed before it is loaded into the target system, while in ELT, data is loaded first and then transformed within the target system. ELT is becoming increasingly popular as organizations adopt big data technologies, such as Hadoop and Spark, which provide the flexibility and scalability needed to perform complex transformations on large volumes of data. ELT also allows organizations to store data in its raw form, without the need to define a schema upfront, and then transform it as needed for specific use cases.

What is the 3 tier architecture in data engineering?

The 3-tier architecture is a common architecture used in data engineering. It separates the components of an application into three different tiers or layers, with each layer serving a specific purpose.

The 3 tiers are as follows:

Presentation Layer: This is the top layer of the architecture and is responsible for presenting the data to the end-user. It may include web applications, mobile applications, or desktop applications that allow users to interact with the data.

Application Layer: This is the middle layer of the architecture and is responsible for processing and managing data. It may include data processing engines, data transformation tools, data integration tools, and other components that process and manage data.

Data Layer: This is the bottom layer of the architecture and is responsible for storing and managing data. It may include databases, data warehouses, data lakes, or other storage systems that store and manage data.

3 ETL tools:

1. Talend
2. Microsoft SQL Server Integration Services
3. Apache Spark

What is a Historical Load?

Historical loads are typically performed when implementing a new system or when updating an existing system. The process involves extracting data from the source system, transforming it into the format required by the target system, and loading it into the target system. The process may involve a large volume of data and may require significant resources such as computing power, storage, and network bandwidth.

What is a Full Load?

A full load involves loading all the data from the source system into the target system. This is typically done when implementing a new system or when updating an existing system. Full loads can be time-consuming and resource-intensive, but they ensure that the target system contains a complete and accurate representation of the source data.

What is an Incremental Load?

An incremental load involves loading only the changes or additions to the data since the last load. This is typically done to keep the target system up-to-date with the latest changes in the source system. Incremental loads are faster and less resource-intensive than full loads, but they require a way to track changes in the source data.