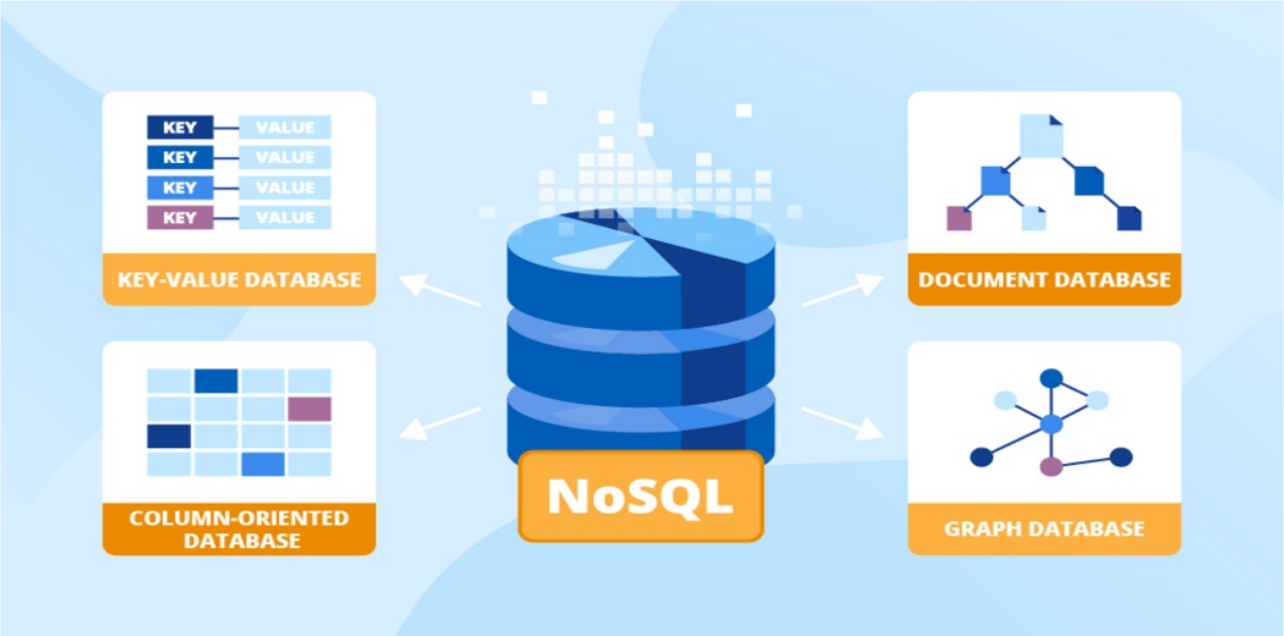
Overview of Relational Databases: Relational databases store structured data in the form of rows and columns called tables, facilitating efficient data retrieval, and establishing relationships between data.

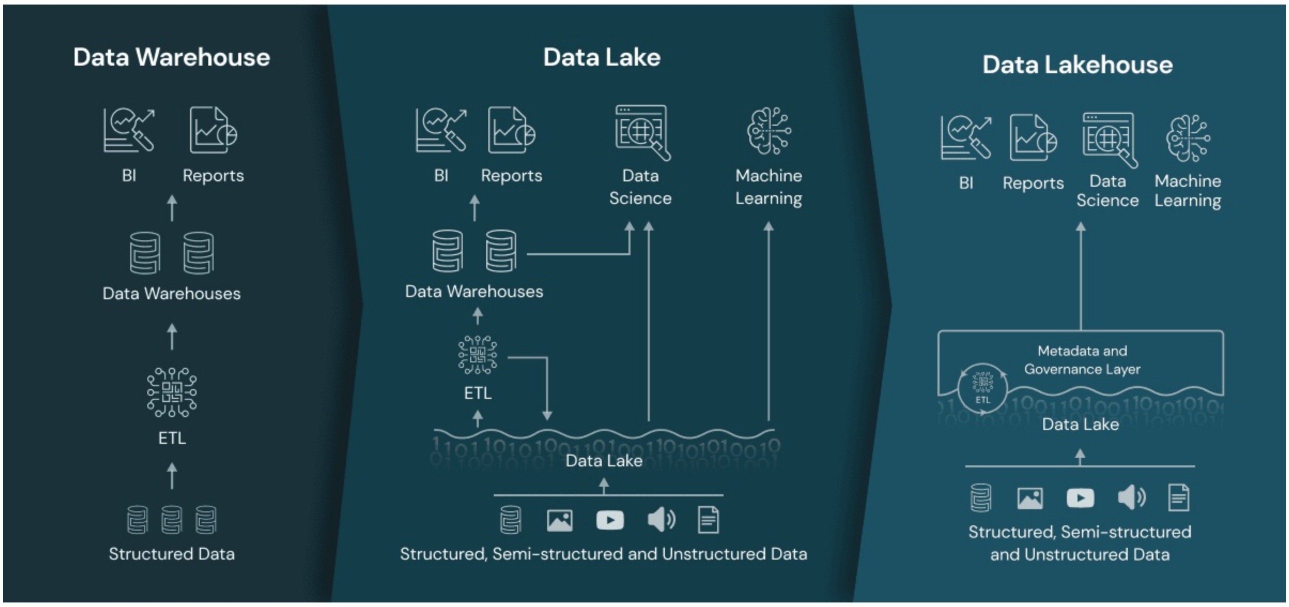
#A Comparison of NoSQL Database Management Systems and Models:
In this article, A comparison of NoSQL database management systems and models is covered, which also compares the models of these systems. It illustrates the distinctions between the four main subcategories of databases: document-based, key-value, column-family, and graph databases. Insights into their use cases, advantages, and disadvantages are provided.
Document-based databases: are highlighted for their flexibility and schema-less nature, which are suitable for unstructured data.

Key-value databases: exemplified by Redis, excel in high-speed data retrieval and caching.
Column-family databases: represented by Apache Cassandra, are ideal for handling large amounts of data with high write throughput.
Graph databases: such as Neo4j are praised for efficiently managing interconnected data.



#Understanding the Data Lakehouse:
In this article, Data Lakehouse explains how it combines the advantages of data lakes and data warehouses. By utilizing tools like Apache Spark and Delta Lake, it combines data processing and storage. Working with structured and semi-structured data, preserving data quality, and ensuring ACID transactions are all made possible by a Data Lakehouse for Data Engineers and Analysts. Additionally, it allows SQL querying, opening data analysis to more users. Other

important aspects highlighted in this article are data versioning and data lineage tracking, which provide greater data governance and auditability.



References:
https://www.digitalocean.com/community/tutorials/a-comparison-of-nosql-database-management-systems-and-models
https://www.databricks.com/glossary/data-lakehouse