Relational Database

A relational database is simply a database that uses a relational data model, which organises data in tables with rows of data entries and columns of predetermined data types. Relationships between tables are represented with foreign key columns that reference the primary key columns of other tables.

Most importantly, relational data models strictly enforce constraints to ensure that data values and relationships are always valid against the schema. ACID transactions are almost always implemented to ensure schema conformance.

Relational databases are almost always CP databases because guaranteeing consistency is important to upholding the relational model, and making sure that no matter what transactions occur, the database is always in a valid state.

SQL databases are based on a relational model, where data is organized into tables with rows and columns, and relationships between tables are defined using foreign keys. SQL databases are known for their consistency and reliability, and are used for applications that require complex transactions, joins, and data integrity. Examples of SQL databases include MySQL, PostgreSQL, and Microsoft SQL Server.

Non-Relational databases

Non-relational databases are optimized for specific use cases that need scalability, schema flexibility, or specialised query support. Non-relational databases are either AP databases or CP databases, because they're targeting specific use cases that have varying priorities between availability and consistency. In the case of AP non-relational databases, the model of eventual consistency is used to make sure that consistency still happens over time, it's just not guaranteed exactly after a transaction completes.

NoSQL databases, on the other hand, are based on a non-relational model and do not use a fixed schema. Data is stored in a variety of ways, such as key-value pairs, documents, or graphs. NoSQL databases are known for their scalability and performance, and are used for applications that require high availability, low latency, and large amounts of unstructured data. Examples of NoSQL databases include MongoDB, Cassandra, and Redis.

Basis Of Comparison	SQL Database	NoSQL Database
Structure	A table-based structure is a preferable choice for programs that demand multi-row operations, such as accounting systems, or for legacy applications designed with a relational structure in mind.	Document-based structure with Key-value pairs, graph databases, and wide-column stores.
Query Language	SQL is a Structured Query Language	There is no declarative query language that NoSQL uses. It differentiates as per the different databases.

Suitable	The database is best suitable for Complex queries but not good for hierarchical data storage.	The database is not suitable for complex queries. However, it suits well for the hierarchical data storage.
Scalability	These databases are vertically scalable. So, with boosting CPU, RAM, or SSD, you may increase the demand on a single server.	NoSQL databases are horizontally scalable. This means by sharding or adding multiple servers to this database, you can handle greater traffic.
Design for	Traditional RDBMS examine and retrieve data for further analysis using SQL syntax and queries. The OLAP systems take advantage of them.	The NoSQL database system is made up of many database technologies. These were created in line with the requirements placed on recent application development.

Schema	SQL databases consists of a predefined schema	NoSQL databases comprises of the dynamic schema for unstructured data.
Properties Followed	SQL accompanies ACID properties that stand as Atomicity, Consistency, Isolation and Durability.	NoSQL accompanies Brewers CAP theorem that stands as Consistency, Availability and Partition Tolerance.
Examples	MySQL, Sqlite, MS-SQL, PostgreSQL, etc.	Big Table, MongoDB, Hbase, Cassandra, Raven DB, etc.