Technical Research: Database and Document Generation

SQL vs NoSQL

SQL (Structured Query Language)

Language used for storing, manipulating, and retrieving (managing) data in relational databases and perform operations on the data in them. SQL is great for business intelligence tools that help companies make informed decisions, for cleaning, testing and transforming data in structured formats, and processing massive datasets. SQL is best used when handling structured. SQL ensures reliable, transaction-safe operations (would be important for financial systems).

Features of SQL:

- Has set commands to define and modify the structure of a database
- Has set commands to manipulate data within the database (add, modify and delete)
- Supports transaction processing (allowing for the ability to group a set of database operations into a single transaction that can be rolled back
- Features (such as specifying constraints on values) to enforce data integrity
- SQL is able to be used on multiple systems without much modification
- Used for modifying database tables and index structures

- The database consists of rows and columns of data. A single table holds records and each record is stored in a row of the table
    - Adding, updating and deleting rows of data
- Retrieving subsets of information from within relational database management systems (used for transaction processing and analytics applications

NoSQL (Not only Structured Query Language)

Language used for databases that handle large volumes of unstructured and semi-structured data. Flexible data models and supports horizontal scaling (great for high performance, scalability, and managing diverse data types). NoSQL isn't restricted to only relational databases, and the data model is stored as a document, key-value, column family or a graph. The data can change over time without a rigid definition. NoSQL would be ideal for large amounts of unstructured data where flexibility, availability and data tolerance is desired.

Features of NoSQL Databases:

- A dynamic schema that allows for flexible shaping of data (as well as flexible and dynamic data storage and recovery) to meet new requirements without the need change schemas
- Horizontal scalability for adding more nodes to existing ones and acquiring the necessary storage for the dataset
- Data is presented in flexible, semi-structured formats (e.g., JSON/BSON)
- Simple and fast access pattern (data stored in pairs of keys as values, and data is organised into columns instead of rows)
- Designed to handle node failures and data replication across multiple nodes in a database cluster (where more than 1 database server is connected to your system)

However, NoSQL databases may not provide consistency and strict data integrity, lack certain functionalities, and aren't as efficient as SQL at handling complex queries.

References:

Coursera (n.d) *NoSQL vs SQL: Key differences and when to use each*. Coursera. Available at: https://www.coursera.org/in/articles/nosql-vs-sql (Accessed: 1 February 2026).

GeeksforGeeks (n.d) *Difference between SQL and NoSQL*. GeeksforGeeks. Available at: https://www.geeksforgeeks.org/sql/difference-between-sql-and-nosql/ (Accessed: 1 February 2026).

GeeksforGeeks (2025) *Advantages and disadvantages of SQL*. GeeksforGeeks. Available at: https://www.geeksforgeeks.org/sql/advantages-and-disadvantages-of-sql/ (Accessed: 19 January 2026).

GeeksforGeeks (2025) *Features of Structured Query Language (SQL)*. GeeksforGeeks. Available at: https://www.geeksforgeeks.org/sql/features-of-structured-query-language-sql/ (Accessed: 19 January 2026).

GeeksforGeeks (2025) *Introduction to NoSQL*. GeeksforGeeks. Available at: https://www.geeksforgeeks.org/dbms/introduction-to-nosql/ (Accessed: 19 January 2026).

IBM (n.d) *SQL vs. NoSQL databases*. IBM. Available at: https://www.ibm.com/think/topics/sql-vs-nosql (Accessed: 1 February 2026).

Wikipedia (n.d) *NoSQL*. Wikipedia. Available at: https://en.wikipedia.org/wiki/NoSQL (Accessed: 1 February 2026).