Jose Franco

Assignment 1.3: **Basic Comparison of Relational vs. NoSQL Databases**

**What are relationships in relational databases?**

In relational databases, “relationships” define how tables in the database are connected to each other. The tables in a relational database store data in rows and columns, and relationships help link that data based on common attributes. The two most common types of relationships are:

**1. One-to-Many (1:N):** This is the most common type of relationship, where a record in one table can relate to multiple records in another table. For example, in a database for an online store, one customer (from a "Customers" table) can have many orders (in an "Orders" table), but each order is associated with only one customer.

**2. Many-to-Many (M:N):** This relationship occurs when multiple records in one table are associated with multiple records in another table. In a relational database, many-to-many relationships are typically broken down into two one-to-many relationships using a join table (or associative entity). For example: a university might have students enrolled in multiple courses, and each course can have multiple students. A "StudentCourse" join table is used to manage the many-to-many relationship between "Students" and "Courses."

**What are the advantages of relational databases?**

**1. Data Integrity and Consistency**: Relational databases follow ACID (Atomicity, Consistency, Isolation, Durability) principles, ensuring that transactions are reliably processed, and data remains accurate and consistent even in the event of system failures.

**2. Structured Query Language (SQL):** Relational databases use SQL, a powerful and standardized language for querying and managing data, making it easy to perform complex queries and updates.

**What are the advantages of NoSQL databases?**

**1. Scalability:** NoSQL databases are designed to scale horizontally, meaning they can handle large amounts of data and traffic by adding more servers. This makes them ideal for applications with rapidly growing data, such as social media platforms.

**2. Flexibility:** Unlike relational databases, NoSQL databases are schema-less or have flexible schemas, allowing developers to store unstructured or semi-structured data (e.g., JSON documents) without the need for predefined data models.

**What are the disadvantages of relational databases?**

**1. Scalability Limitations**: Relational databases are typically vertically scalable, meaning they require more powerful hardware as data grows, which can be costly. This limits their ability to handle extremely large datasets or high-traffic environments compared to NoSQL databases.

2. **Rigid Schema**: Relational databases require a predefined schema, making it harder to accommodate changes in data structure without significant modifications to the database schema.

**What are the disadvantages of NoSQL databases?**

**1. Lack of ACID Transactions**: Many NoSQL databases sacrifice ACID compliance to prioritize scalability and performance, which can lead to potential issues with data consistency and reliability in certain use cases.

**2. Limited Support for Complex Queries:** NoSQL databases may lack the powerful querying capabilities of SQL, making it harder to perform advanced operations such as multi-table joins and complex data relationships.

**Features of MySQL**

**1. Relational Database Management System (RDBMS):** MySQL is a relational database system, meaning it stores data in structured tables with predefined schemas. It supports complex querying through SQL, making it suitable for applications where relationships between data are important, for example, web applications with structured data like users, products, and orders.

**2. ACID Compliance**: MySQL ensures ACID-compliant transactions, which means that each transaction in the database is processed reliably and ensures data integrity. This is critical for applications where consistency and reliability are key, for example, financial systems.

**Features of MongoDB**

**1. Document-Oriented Storage:** MongoDB is a NoSQL database that stores data in flexible, JSON-like documents. This allows for a more dynamic schema where fields can vary from document to document, making MongoDB suitable for storing unstructured or semi-structured data, for example, user profiles in social media apps.

**2. Horizontal Scalability**: MongoDB is designed to scale horizontally using sharding, which distributes data across multiple servers. This makes it ideal for handling large volumes of data and high read/write operations, for example, real-time analytics systems.

**Sources:**

Elmasri, R. A., & Navathe, S. B. (2017). *Fundamentals of Database Systems*. Pearson.

Comeau, A. (2018). *MySQL explained: Your step-by-step guide to database design*. CA : OSTraining.

*MongoDB documentation*. MongoDB Documentation. (n.d.). https://docs.mongodb.com/