**Mark White**

**CSD-310**

**Assignment#1.3**

Relational structured databases are organized as tables called relationships, consisting of columns and rows. There are three types of relationships in relational database design; One-to-One, One-to-Many (or Many-to-One), or Many-to-Many.

One-to-one is the least common relationship type. In this type, data in Table A could have been stored along with other data in Table B, but for certain purposes, like the security of sensitive data, it is divided so that only certain users may access the information. For instance, Table A: employee table, and Table B: Pay. We could have easily put the hourly pay under the employee table but to ensure only certain user access, creating another table provides extra security.

Another is One-to-Many, which is the most common relationship type. In this type a row in table A can have matching rows in table B, but a row in table B can have only one matching row in table A. For instance, you can have table A city name and table B customers. Each customer will be assigned to one city, but one city can have many customers.

There are many benefits and advantages to working with Relational databases. They provide flexibility to structure data that is not capable of other database technologies. The key feature is the ability to make tables look like one, join multiple together on key fields, create complex indexes that perform well and are easy to manage, and maintain data integrity for maximum data accuracy.

**Advantages include:**

**Simplicity of Model**- the relational database model is much simpler. It does not require any complex queries because it has no query processing or structuring

**Ease of Use**- Users can easily access/retrieve their required information within seconds without indulging in the complexity of the database.

**Accuracy**- A key feature of relational databases is that they’re strictly defined and well-organized, so data doesn’t get duplicated.

**Data Integrity**- ensures accuracy and ease of use,

**Normalization**- breaking down information into manageable chunks reducing storage,

**Collaboration**-multiple users can access and retrieve information simultaneously, and

**Security**- allows authorized users to directly access data.

**Disadvantages include:**

**Maintenance problems**- become difficult overtime due to increase in data

**Costs**- The relational database system is costly to set up and maintain

**Physical storage**- A relational database is comprised of rows and columns, which requires a lot of physical memory because each operation performed depends on separate storage

**Lack of scalability**- using over mutilpe servers, structure changes and becomes difficuly to handle.

**Lack of complexity in structure**- can only store in tabular form making it difficult to represent complex relationships.

**Decrease in performance over time**- When there is a large number of tables and data in the system, it causes an increase in complexity, leading to slow response times.

---------------------------------------------------------------------------------------------------

NoSQL databases became popular because they provided a simple way to store data from multiple sources while using different formats. Fewer (or no) transformations are needed when the data is being stored or retrieved from storage. A large variety of data types, whether unstructured, structured, or semi-structured, can be stored in a NoSQL database.

**Strengths:**

-Can Support easy updates to schemas

-Can process large volumes of data at high speed

-Storage is scalable

-Can store structured, semi-structured and unstructured data.

-Developer friendly

**Disadvantages include:**

Lack of SQL

Lack of Standardization

Lack of ACID (Atomicity, Consistency, Isolation, and Durability)

**Features of MySQL:**

**Open Source**- The community edition of MySQL is enterprise ready, and supported by a GNU General Public License. For users who want access to equitable proprietary functionality of MySQL without the added price tag, there are other options within the ecosystem — like MariaDB — that can add similar levels of functionality and beyond.

**Security**- MySQL's maturity also lends itself to security. It's regularly updated, has a vibrant developer community, and, because of its wide adoption within the enterprise, many CVE patches are released before the CVE is announced. These factors combine to make MySQL a stable and secure choice among RDBMS.

**Features of MongoDB:**

**Database Triggers**- Database triggers in MongoDB Atlas are a powerful feature that allow you to execute code when certain events occur in your database.

**Sharding**- The process of splitting larger datasets across multiple distributed instances, or “shards.” When applied to particularly large datasets, sharding helps the database distribute and better execute what might otherwise be problematic and cumbersome queries. Without sharding, scaling a growing web application with millions of daily users is nearly impossible.

**References:**

DatabaseTown. (2023). *Relational Database Benefits and Limitations (Advantages & Disadvantages) - DatabaseTown*. Database Town. https://databasetown.com/relational-database-benefits-and-limitations/

Foote, K. D. (2022). *NoSQL Databases: Advantages and Disadvantages*. DATAVERSITY. https://www.dataversity.net/nosql-databases-advantages-and-disadvantages/

MongoDB. (2024). *MongoDB Features & Key Characteristics |* https://www.mongodb.com/resources/products/capabilities/features?msockid=16bf3781e9e667562aeb2585e8996616

‌