**What are Some of the Different SQL Databases and What are Their Pros and Cons?**

SQL databases are primarily relational databases that use structured query language to manage and manipulate data. Some of the most popular SQL databases include MySQL, PostgreSQL, SQLite, and SQL Server. Each has unique benefits tailored to different applications. For example, **MySQL** is known for its reliability and efficiency in web applications and open-source projects, while **PostgreSQL** is lauded for its robust compliance with SQL standards and advanced features. **SQLite** is lightweight and ideal for smaller-scale applications or mobile use. **SQL Server**, on the other hand, offers strong support for Microsoft environments and robust security features (Stanford, n.d.). The major advantage of relational databases is their ability to handle structured data with consistency and integrity. However, challenges include scalability issues for very large datasets and potential complexities in schema design​([Code the Change](https://codethechange.stanford.edu/guides/sql.html)).

**Why is it Useful to Know SQL?**

SQL is an essential skill for many data-driven roles because it is widely used across industries to store, query, and manipulate data efficiently. Its high demand is due to the sheer volume of data being generated and the need for specialists who can manage, analyze, and derive insights from this data (Joubert, 2020). SQL is not only industry-standard but also highly versatile, making it possible to work with various database management systems, like MySQL, PostgreSQL, and Oracle. Moreover, its relatively simple syntax and structure make it easy to learn, and the ability to quickly retrieve and process data allows businesses to make strategic decisions efficiently​([Graduate Programs](https://graduate.northeastern.edu/resources/sql-data-analyst/))

**What is Database Normalization and Why is it Important?**

Database normalization is the process of organizing a database into tables to reduce redundancy and improve data integrity. By following specific rules to separate data into related tables, normalization aims to ensure consistency, reduce duplicate data, and simplify database maintenance. For example, rather than storing repetitive customer details within every order record, normalized databases maintain customer details in a separate table, linking it to order records through foreign keys. While this enhances data integrity and optimizes storage, it can make queries more complex, as data is often distributed across multiple tables (Stanford, n.d.).

**What is the Difference Between Using Hive and Something Like PostgreSQL?**

**Hive** and **PostgreSQL** serve different purposes within the realm of data management. Hive is built on top of Hadoop and is designed for querying large datasets in a distributed storage environment. Its query language, HiveQL, is similar to SQL but optimized for batch processing and analytics, making it ideal for big data scenarios. However, Hive is not optimized for real-time querying due to its high latency. In contrast, **PostgreSQL** is a relational database management system suited for structured data storage, online transaction processing (OLTP), and real-time queries. It supports complex queries and provides strong ACID (Atomicity, Consistency, Isolation, Durability) compliance, making it optimal for real-time applications and smaller-to-medium-sized datasets (Stanford, n.d.).

**What is the Difference Between DML Versus DDL?**

In SQL, commands are categorized into DML (Data Manipulation Language) and DDL (Data Definition Language). **DML** deals with manipulating the data within the database and includes commands like SELECT, INSERT, UPDATE, and DELETE. These commands allow users to retrieve, add, modify, and remove data within tables. **DDL**, on the other hand, focuses on defining and altering the structure of the database objects. It includes commands such as CREATE, ALTER, DROP, and TRUNCATE, which help manage the schema and organization of the database. Simply put, DML is about handling the content of the database, whereas DDL is about managing the structure.

**References**

Joubert, S. (2020, January 22). *5 Reasons SQL Is The Need-to-Know Skill For Data Analysts*. Northeastern University Graduate Blog. <https://graduate.northeastern.edu>

Stanford. (n.d.). *Introduction to Database and Structured Query Language (SQL)*. Code the Change. <https://codethechange.stanford.edu>

Western Governors University. (n.d.). *Developing Your SQL Skills*. WGU. <https://www.wgu.edu>

Johns Hopkins University. (n.d.). *Complete Guide to SQL for Data Engineering: from Beginner to Advanced*. Imagine. <https://imagine.jhu.edu>