SQL vs. NoSQL

NoSQL (short for "Not Only SQL") is an alternative to traditional databases, focused on capturing and processing large amounts of data.

There are several types of NoSQL databases, each with a unique approach to data modeling and different uses cases.

**Key-value databases :**

organize data in pairs of keys and values, where each key is tied to a specific object, representing a data field. Providing a key allows you to view the data stored in the object it is paired with.

This is the simplest and most scalable type of NoSQL database, offering flexibility and improved performance.

-Redis

-Aerospike

-Riak

-Project Voldemort

**Document-based databases** :

also use key-value pairs they store into documents. These documents are further grouped into collections based on their content and use.

These databases most commonly store data as JSON, XML, BSON, or YAML forms, usually without implementing a schema. This approach makes them suitable for cases that require a flexible structure and the ability to quickly add and retrieve data.

-MongoDB

-Couchbase Server

-CouchDB

-CouchDB

-Elasticsearch

**Graph-based databases** :

represent data as a collection of nodes (data elements) connected by edges. In this data structure, nodes contain pieces of data, while edges define relationships between them.

This database type is commonly used to represent relationships between different data entries, such as friend connections on social networks. Users can perform complex queries and directly pull multiple pieces of data at the same time.

-Neo4J

-OrientDB

-RedisGraph

**Wide column-based databases** :

store data into separate columns, similar to how data is stored in tables with relational databases. Unlike relational databases, wide-column databases do not use predefined keys or column names.

This allows for variations in column names, even within the same table. It is also easy to add large amounts of data as new columns, or group existing ones into column families.

-Cassandra

-Cosmos DB

-HBase

**Object Databases :**

Object databases store data elements as objects to be used in object-oriented programming. They are designed to work with programming languages like Python, Ruby, Delphi, Java, etc.

-ObjectDB

-Ninja Database Pro

**Grid and Cloud Databases :**

Grid and cloud databases use a data grid - a network of systems working with data accessible through the cloud.

This type of database works with both SQL and NoSQL data models and is typically offered as a database-as-a-service.

-Infinispan

-Hazelcast

**Multi-Model Databases :**

Finally, multi-model databases combine the features of two or more different database types. This allows them to provide a solution for unique use cases where other database types are not suitable.

-ArangoDB

SQL stands for Structured Query Language. It's used with all kinds of relational databases.

MySQL :

MySQL has been at the top of the popularity ranking for several years. Why? It’s free, works for most applications, and runs on most popular platforms, including Linux, Windows, and macOS.

Here some users spot a controversy: Although Oracle representatives have sworn one version of MySQL will remain open source, many people fear it will change over to a proprietary (licensed) version.

PostgreSQL :

PostgreSQL is free,open-source, and will work in all possible situations and on all platforms. It has a very dynamic community of users who help develop the project and write their own plugins and extensions.

This is the database system used by Instagram, Twitch, and Skype. That should be a sufficient recommendation. Plus, StackOverflow research backs this up: PostgreSQL gains users each year.

Microsoft SQL Server :

Many companies trust Microsoft, use other solutions from this company, and do not want to include anything else in their IT ecosystem. So you will find MS Server in many businesses, especially larger corporations.

Despite the dominance of open source solutions on the market, MS Server is doing great. Don't be afraid that once you learn it, you will never use it. Every dollar you spend on learning MS SQL Server will pay off.

In addition to SQL Server, Microsoft also offers the great Cloud-based Azure platform. This means you can avoid installing MS Server on a physical computer, putting it on the Cloud instead. This facilitates database work for multiple users and also keeps things safer than a single onsite installation.

SQL Azure, like MS SQL Server, uses the relational query language T-SQL. Do you want to learn it? LearnSQL.com offers some great courses. If you’re starting your adventure with SQL, I recommend the SQL Fundamentals in MS SQL Server course. If you’re already an SQL user, try the Recursive Queries in MS SQL Server course. It will power up your SQL skills.