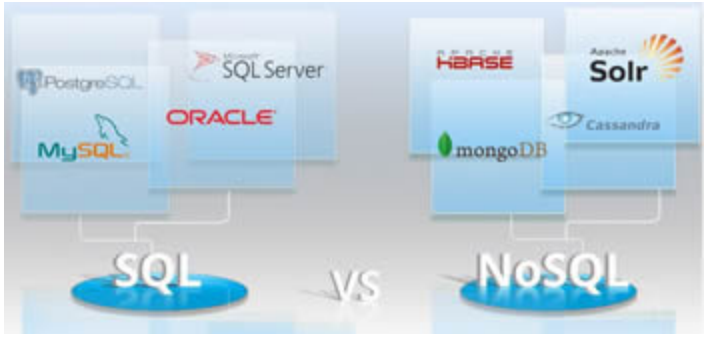
Difference Between | Descriptive Analysis and Comparisons

Search

Difference between NoSQL and SQL database

Like 4 Tweet G+

Key Difference: SQL are also known as relational databases and are more structured than NoSQL databases, which are known as non-relational databases. Non-relational databases tend to be more document-oriented and distributed rather than structured.



There are many different types of databases in use today. SQL Databases have been and still are some of the widely used databases out there. Some other popular one in use are MySQL, Oracle, as well as some other types of SQL databases. However, recently NoSQL databases, such as MongoDB, have been gaining popularity. Both SQL and NoSQL databases serve the same purpose for the post part, which is to act as a database, i.e. collect and organize data.

NoSQL databases actually predated SQL databases, as they existed as far back as the 1960s, whereas SQL was not developed until 1974. All databases that existed prior to the popularity of SQL fall under the moniker of NoSQL. SQL are also known as relational databases and are more structured than NoSQL databases, which are known as non-relational databases. Non-relational databases tend to be more document-oriented and distributed rather than structured. For example, it can be said that relational databases are structured like phone books that store phone numbers and addresses, whereas non-relational databases are like file folders that hold everything from a person's address and phone number to their Facebook likes and online shopping preferences.

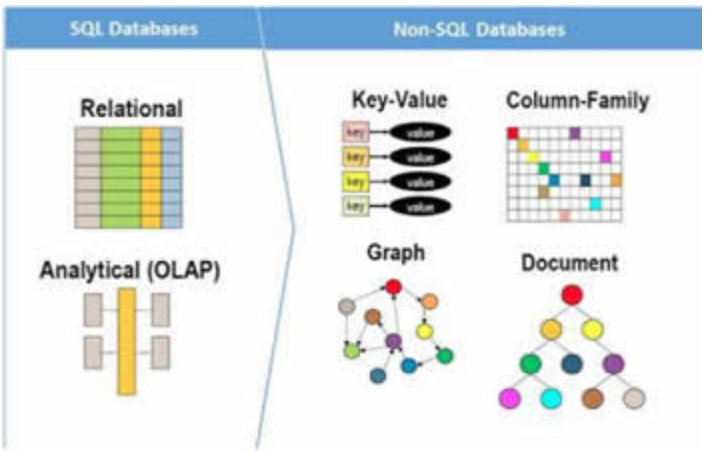
Popular Tags

Disneyland	Cryptocurrencies
Land Phones	Arboreal
Embassies & Consulates	Wordpad
Trade Bloc	Wire
Data Managements Systems	Other - Diseases

Interview Questions Answers

Newly Added Differences

The primary difference between the two is the fact that SQL databases utilize SQL, i.e. Structured Query Language as the standardized programming language for managing relational databases as well as for performing various operations on the data in them. Whereas, NoSQL databases typically do no use tabular relations for storage and retrieval of data that are common in relational databases. Instead they are document based that store key-value pairs, graph databases or wide-column stores. NoSQL are focused primarily on the collection of documents. It is also due to this that NoSQL are ideally used to store large set of data which are commonly stored in JSON documents.



Basically, this means that while the SQL databases represent data in the form of tables, NoSQL databases show key-value pair, documents, graph databases or wide-column stores, all of which do not have standard schema definitions that need to be followed. SQL databases, on the other hand, have to stick to a predefined schema.

In addition to these, there are various other differences between the two. The benefit of NoSQL over SQL is the fact that it is more variable and can be adapted to fit any need. Whereas, SQL has typically been more structured and less adaptable.

Comparison between NoSQL and SQL database:

Test vs Examination
Hybrid Cars vs Electric Cars
Ad vs Add
Research Paper vs Review Paper
Access vs Excess
Angular 2 vs Angular 4
Castor Oil vs Mineral Oil
Born vs Borne
Cheque vs Check
Ambience vs Ambiance

Most Popular Differences

Management vs Administration
Data vs Information
Android vs Windows Phone
Yoghurt vs Curd
SUV vs MUV vs XUV
Taekwondo vs Karate
Tablet vs Smartphone
Inotropic vs Chronotropic vs Dromotropic

Random Pick

Lenovo Thinkpad Twist vs Lenovo IdeaPad Yoga 11
Decriminalization vs Legalization
Gross vs Net
Appointment Letter vs Confirmation Letter
Borderline vs Sociopath
HTC Butterfly vs HTC First
Transcript vs Degree Certificate
ATA vs SATA

AdChoices

Database
Nosql SQL VS

Most Commented

	NoSQL database	SQL database
Also known as	Non-relational or distributed database	Relational Databases (RDBMS)
Language	Queries are focused on collection of documents. Sometimes it is also called as UnQL (Unstructured Query Language).	Use SQL (Structured Query Language) for defining and manipulating the data
Basis	Databases are document based, key-value pairs, graph databases or wide-column stores	Table based databases
Data storage	Hierarchical data storage	No hierarchical data storage
Schema	Have dynamic schema for unstructured data	Have predefined schema
Schema	Database is schema-agnostic and is dictated by the application. It allows for agility and highly iterative development	Schema must be maintained and kept in sync between application and database
Data Properties	Offers flexibility as not every record needs to store the same properties	Great for solutions where every record has the same properties

Android vs Windows Phone
Management vs Administration
Tablet vs Smartphone
Data vs Information
Indian Culture vs Western Culture
Joint Family vs Nuclear Family
MPhil vs Ph.D.
Yoghurt vs Curd

New data	New data can be added without any reservation	Adding new data might require altering schemas or backfilling data
Types of Data	Good for semi-structured, complex, or nested data	Good for structured data
Relationships	Relationships between the data are often captured by denormalizing data and representing all data for an object in a single record	Relationships are often captured in normalized models using joins to resolve references across tables.
Scalability	Are horizontally scalable. Are scaled by increasing the databases servers in the pool of resources to reduce the load.	Are vertically scalable. Are scaled by increasing the horse-power of the hardware.
Complex Queries	Are not good fit for complex queries	Are good fit for the complex query intensive environment
Standard interfaces	Don’t have standard interfaces to perform complex queries	Have standard interfaces to perform complex queries
Size of data set	Highly preferred for large data set as can store hierarchical data	Not ideal for large data set

Complex transactional applications	Not comparable and stable enough in high load and for complex transactional applications.	Better suited for heavy duty transactional type applications, as it is more stable and promises the atomicity as well as integrity of the data.
Adoption and Support	Not widely adopted and local community support only	Widely adopted and easier support available
Properties	Follows the Brewers CAP theorem (Consistency, Availability and Partition tolerance). Some databases may follow ACID properties.	Emphasizes on ACID properties (Atomicity, Consistency, Isolation and Durability)
Examples	MongoDB, BigTable, Redis, RavenDb, Cassandra, Hbase, Neo4j and CouchDb	MySql, Oracle, Sqlite, Postgres and MS-SQL

Reference: Wikipedia (SQL and NoSQL), The Geek Stuff, Microsoft, Upwork
Image Courtesy: blog.raremile.com, kvaes.wordpress.com

Like 4 Tweet G+

Most Searched in Beauty and Style

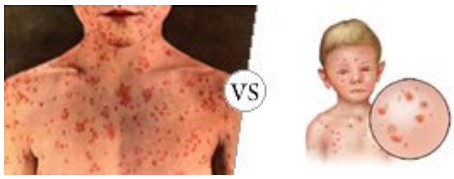
Most Searched in Home & Garden

Most Searched in Cars & Transportation

Most Searched Business & Finance

Could VS Would

Mall VS Plaza



Add new comment

Your name

Comment *

CAPTCHA
This question is for testing whether or not you are a human visitor and to prevent automated spam submissions.

I'm not a robot

reCAPTCHA
Privacy - Terms

Save

Build Sleek UI Fast

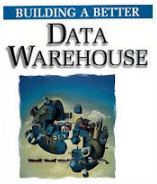
Ad Progress Telerik

Difference between SQL vs MySQL

differencebetween.info

Solid State Drives. - Largest Capacities.

Ad ngdsystems.com



Difference between Data Warehouse and Database

differencebetween.i

Go Beyond Spreadsheets

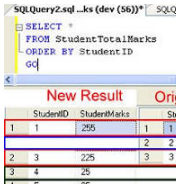
Ad Tableau Software

Difference between Cute and Gorgeous

differencebetween.info

Difference between Oracle and MySQL

differencebetween.info



Difference b Delete and SQL

differencebetween.i

Copyright © 2018, Difference Between | Descriptive Analysis and Comparisons