**SQL Databases (Relational Databases)**

**Definition:**

SQL (Structured Query Language) databases are relational databases that store data in tables (rows and columns) with a fixed schema. They use SQL to query, manipulate, and manage data.

**Characteristics:**

* Structured Schema: Data is organized in predefined tables with specific data types.
* ACID Compliance: Ensures data Atomicity, Consistency, Isolation, Durability – making transactions reliable
* Relationships: Use foreign keys to define relations between tables.
* Scalability: Vertical (scaling up by adding resources to a single server).
* Strong Data Integrity: Enforced by strict rules and constraints.
* Complex Query Support: SQL supports joins, subqueries, aggregations, etc.

**Common SQl Databases:**

* MySQL
* PostgreSQL
* Microsoft SQL Server
* Oracle DB
* SQLite

**NoSQL Databases**

**Definition:**

NoSQL databases are non-relational databases designed for scalable and flexible data storage. They do not use fixed schemas and are suited for handling large volumes of unstructured, semi-structured, or rapidly changing data.

**Characteristics:**

* Schema-less: Data can be stored without a predefined schema.
* Horizontal Scalability: Easily scaled across multiple servers (clusters)
* High Performance: Optimized for speed and flexibility.
* Eventual Consistency: Prioritizes performance over strict consistency in some cases (CAP theorem).
* Data Models: Can store structured, semi-structured, and unstructured data.

**Types of NoSQL Databases:**

* Document-oriented: Store data in JSON-like documents.
* Examples: MongoDB, CouchDB
* Key-Value Stores: Data stored as key-value pairs.
* Examples: Redis, DynamoDB
* Column-family Stores: Store data in columns instead of rows.
* Examples: Cassandra, HBase
* Graph Databases: Focus on relationships and graph structures.
* Examples: Neo4j, ArangoDB

**SQL vs NoSQL Databases**

|  |  |  |
| --- | --- | --- |
| **Feature** | **SQL Databases** | **NoSqlDatabases** |
| Data Model | Relational (tables with rows/columns) | Non-relational (documents, key-value, etc.) |
| Schema | Fixed Schema | Dynamic Schema |
| Query Language | SQL | Varies (MongoDB uses BSON queries, etc.) |
| ACID Compliance | Full ACID compliance | Often eventual consistency (BASE) |
| Scalability | Vertical | Horizontal |
| Relationships | Strong (foreign keys, joins) | Limited or handled differently |
| Examples | MySQL, PostgreSQL, SQL Server | MongoDB, Cassandra, Redis, Neo4j |
| Best For | Structured data with complex queries | |  | | --- | | queries |  |  | | --- | | Big data, real-time apps, unstructured data | |

**MongoDB – A Popular NoSQL Database**

**Definition:**

MongoDB is an open-source, document-oriented NoSQL database. It stores data in flexible, JSON-like documents (BSON – Binary JSON), allowing nested values and arrays.

**Key Features:**

Schema-less: Collections can hold documents of different structures.

High Performance: Fast read/write operations.

Horizontal Scaling: Supports sharding for distributed data.

Indexing: Supports various types of indexing (single field, compound, text, geospatial).

Aggregation Framework: Powerful tool for data analysis, transformations, and reporting.

Replication: Uses replica sets for high availability and data redundancy.

**Basic Concepts:**

Database: Like a schema in RDBMS.

Collection: Like a table.

Document: Like a row, but flexible in structure (uses JSON format).

Field: Like a column.