



LARANA, INC.

# DATABASES MODELS PROJECT

# SQL FUNCTIONALITY

## Schema and Structure

- SQL databases have a predefined schema with tables that define the structure of data.
- Tables have fixed columns and data types.

## Query Language

- SQL (Structured Query Language) is used to query and manipulate data.
- Standard SQL allows for complex queries involving joins, aggregations, and subqueries.

## Scalability:

Traditional SQL databases are vertically scalable (scale-up), typically by increasing hardware resources like CPU or RAM.

## Typical Use Cases

Applications requiring complex queries and transactions, such as financial systems, ERP (Enterprise Resource Planning), and CRM (Customer Relationship Management) systems.

## Data Integrity:

- SQL databases enforce ACID (Atomicity, Consistency, Isolation, Durability) properties, ensuring data integrity and reliability.
- Transactions are supported, allowing multiple operations to be grouped together.

# NOSQL FUNCTIONALITY

## Schema and Structure

- MongoDB is schema-less or schema-flexible, allowing documents (data records) to have different structures.
- Documents are stored in collections, which are analogous to tables in SQL.

## Query Language

- MongoDB uses a query language similar to JSON (JavaScript Object Notation) called BSON (Binary JSON).
- Supports CRUD (Create, Read, Update, Delete) operations and various query operators.

## Scalability:

- MongoDB is horizontally scalable (scale-out), distributing data across multiple servers (sharding).
- Designed to handle large volumes of unstructured or semi-structured data efficiently.

## Typical Use Cases

Content management systems, real-time analytics, IoT (Internet of Things) applications, and projects requiring flexible data models.

## Data Integrity:

- MongoDB offers eventual consistency by default, but can be configured for stronger consistency guarantees.
- Transactions are supported in recent versions, enabling atomic operations on multiple documents.

# DIFFERENCES

## SQL DATABASE

- SQL, supports structured data with schemas
- Tables with rows and columns
- Supports relationships with foreign keys and primary keys
- Relational
- Supports vertical scaling
- Architecture moved towards distributed databases for reliability
- Predefined schema required for data structure
- Uses SQL for querying and advanced analytics functions

## NOSQL DATABASE

- NoSQL, supports unstructured data
- Collections containing JSON documents
- No support for table relationships
- Non-relational
- Supports horizontal scaling (sharding)
- Built for resilience and availability
- No predefined schema, dynamic structure
- Limited document querying, no support for joins



LARANA, INC.

**DATABASES MODELS  
THE END**

