

## FECHA Y MARCA TEMPORAL

Estos tipos de datos se utilizan para almacenar fechas y horas en distintos formatos.

- **DATE**: Almacena solo la fecha en formato AAAA-MM-DD (por ejemplo, 2024-09-20).
- **TIME**: Almacena solo la hora en formato HH:MM:SS (por ejemplo, 14:35:00).
- **DATETIME**: Almacena tanto la fecha como la hora en formato AAAA-MM-DD HH:MM:SS (por ejemplo, 2024-09-20 14:35:00).
- **TIMESTAMP**: Similar a DATETIME, pero automáticamente ajusta la zona horaria y puede actualizarse automáticamente al momento de inserción o modificación de los datos.
- **YEAR**: Almacena solo el año en formato AAAA (por ejemplo, 2024).

## JSON

- El tipo de datos JSON permite almacenar y manipular datos en formato JavaScript Object Notation (JSON), útil para almacenar estructuras de datos complejas de forma estructurada y flexible.
- **JSON**: Este tipo permite almacenar datos en formato JSON directamente en una columna de MySQL. Es ideal para almacenar objetos o arrays sin tener que descomponer la estructura en múltiples columnas o tablas. MySQL proporciona funciones integradas para trabajar con estos datos, como JSON\_EXTRACT() para obtener valores específicos de un objeto JSON.

## DATOS DE CADENA

Los tipos de datos de cadena almacenan textos y secuencias de caracteres.

- **CHAR**: Almacena cadenas de longitud fija, hasta 255 caracteres. Las cadenas más cortas se rellenan con espacios.
- **VARCHAR**: Almacena cadenas de longitud variable, hasta 65,535 caracteres. Más eficiente para cadenas variables que CHAR.
- **TEXT**: Almacena grandes cantidades de texto, hasta 65,535 caracteres.
  - **TINYTEXT**: Almacena hasta 255 caracteres de texto.
  - **MEDIUMTEXT**: Almacena hasta 16,777,215 caracteres.
  - **LONGTEXT**: Almacena texto muy grande, hasta 4 GB.
- **BLOB**: Almacena datos binarios (como imágenes o archivos) en varios tamaños:
  - **TINYBLOB, BLOB, MEDIUMBLOB, LONGBLOB**, similares a los tipos TEXT, pero para datos binarios.

# TIPOS DE BASES DE DATOS EN MYSQL

CRISTIAN RAMÍREZ



## QUE ES UNA BASE DE DATOS?

Una base de datos (del inglés: database) se encarga no solo de almacenar datos, sino también de conectarlos entre sí en una unidad lógica.

## NUMERICOS

- **TINYINT**: Almacena enteros muy pequeños, de -128 a 127 o de 0 a 255 (sin signo).
- **SMALLINT**: Almacena enteros pequeños, de -32,768 a 32,767 o de 0 a 65,535 (sin signo).
- **MEDIUMINT**: Almacena enteros medianos, de -8,388,608 a 8,388,607.
- **INT o INTEGER**: Enteros estándar, de -2,147,483,648 a 2,147,483,647.
- **BIGINT**: Enteros grandes, de -9,223,372,036,854,775,808 a 9,223,372,036,854,775,807.
- **DECIMAL o NUMERIC**: Utilizados para almacenar números decimales exactos. Ejemplo: DECIMAL(10,2) almacena hasta 10 dígitos, con 2 decimales.
- **FLOAT**: Almacena números en coma flotante de precisión simple, útil para datos numéricos con decimales pero sin necesidad de precisión total.
- **DOUBLE**: Almacena números en coma flotante de doble precisión, permitiendo manejar valores numéricos con más exactitud que FLOAT.

## DATOS ESPACIALES

MySQL incluye soporte para tipos de datos espaciales que permiten almacenar datos geográficos y geométricos.

- **GEOMETRY**: Tipo base para representar cualquier tipo de dato geométrico (punto, línea, polígono).
- **POINT**: Almacena un punto en el espacio con coordenadas X e Y (generalmente usado para latitud y longitud).
- **LINESTRING**: Representa una línea o secuencia de puntos conectados.
- **POLYGON**: Representa una figura geométrica de tres o más puntos conectados, formando un área cerrada.
- **MULTIPOINT, MULTILINESTRING, MULTIPOLYGON**: Almacenan múltiples puntos, líneas o polígonos, respectivamente.