**MySQL**

**Introduction to MySQL Database**

MySQL is one of the most popular open-source relational database management systems (RDBMS) used for storing and managing data. It uses Structured Query Language (SQL) to interact with databases, allowing developers to create, read, update, and delete data efficiently. MySQL is widely used in web development, data analysis, and application development due to its speed, reliability, and flexibility. It is the database engine behind many well-known websites and applications such as WordPress, Facebook, and YouTube.

**Key Features of MySQL:**

* **Open Source**: MySQL is free to use and modify, making it accessible to both individuals and organizations.
* **Cross-platform**: It works on various operating systems such as Linux, Windows, and macOS.
* **High Performance**: MySQL is known for its fast data processing and query execution.
* **Scalability**: It is capable of handling large datasets and complex queries.
* **ACID Compliance**: MySQL ensures data integrity and reliability with support for transactions (Atomicity, Consistency, Isolation, Durability).
* **Security**: MySQL offers features like user authentication, data encryption, and secure connections.

MySQL uses a relational model to store data in tables, where each table consists of rows and columns. These tables are related to one another, which allows for efficient data storage and retrieval.

**Datatypes in MySQL**

Datatypes define the type of data that can be stored in a column of a table. MySQL offers a wide variety of datatypes, categorized into several types, including numeric, string, date and time, and spatial types.

**1. Numeric Data Types:**

These datatypes are used to store numeric values.

* **INT** (or **INTEGER**): Stores whole numbers.
  + Example: INT(11) can store integers between -2,147,483,648 and 2,147,483,647.
* **TINYINT**: Stores very small integers, typically between -128 and 127.
* **SMALLINT**: Used for small range integers. It stores values between -32,768 and 32,767.
* **MEDIUMINT**: Stores medium-sized integers with a range from -8,388,608 to 8,388,607.
* **BIGINT**: For large integers. Its range is -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807.
* **DECIMAL**: Used to store exact numeric values with a fixed number of digits after the decimal point. Example: DECIMAL(10,2) can store numbers up to 10 digits with 2 digits after the decimal point.
* **FLOAT**: Used to store floating-point numbers (approximate values).
  + Example: FLOAT(7,4) means the number has up to 7 digits in total, with 4 digits after the decimal point.
* **DOUBLE**: Similar to FLOAT, but with double precision for larger numbers.

**2. String Data Types:**

These datatypes are used to store text and strings.

* **CHAR**: A fixed-length string (up to 255 characters). Example: CHAR(10) stores exactly 10 characters.
* **VARCHAR**: A variable-length string that can store up to 65,535 characters, depending on the column’s total size.
* **TEXT**: A long text string, which can store up to 65,535 characters.
  + Variants include TINYTEXT, MEDIUMTEXT, and LONGTEXT, each with increasing capacity.
* **BLOB**: A Binary Large Object that stores binary data (e.g., images or files). Variants include TINYBLOB, MEDIUMBLOB, and LONGBLOB.
* **ENUM**: Stores a value from a predefined list of values (choices).
  + Example: ENUM('small', 'medium', 'large')
* **SET**: Similar to ENUM, but allows multiple values from the predefined set of values.

**3. Date and Time Data Types:**

These datatypes are used to store date and time values.

* **DATE**: Stores date values (YYYY-MM-DD).
  + Example: 2025-02-16
* **DATETIME**: Stores both date and time (YYYY-MM-DD HH:MM:SS).
  + Example: 2025-02-16 15:45:30
* **TIMESTAMP**: Stores date and time, but automatically adjusts to the current time zone.
* **TIME**: Stores time values (HH:MM:SS).
* **YEAR**: Stores a year in the format YYYY.
  + Example: 2025

**4. Other Data Types:**

* **BOOLEAN**: Represents boolean values (TRUE or FALSE).
* **JSON**: Stores JSON-formatted data as text.
* **UUID**: Stores Universally Unique Identifier (UUID) values, typically used to ensure uniqueness across systems.