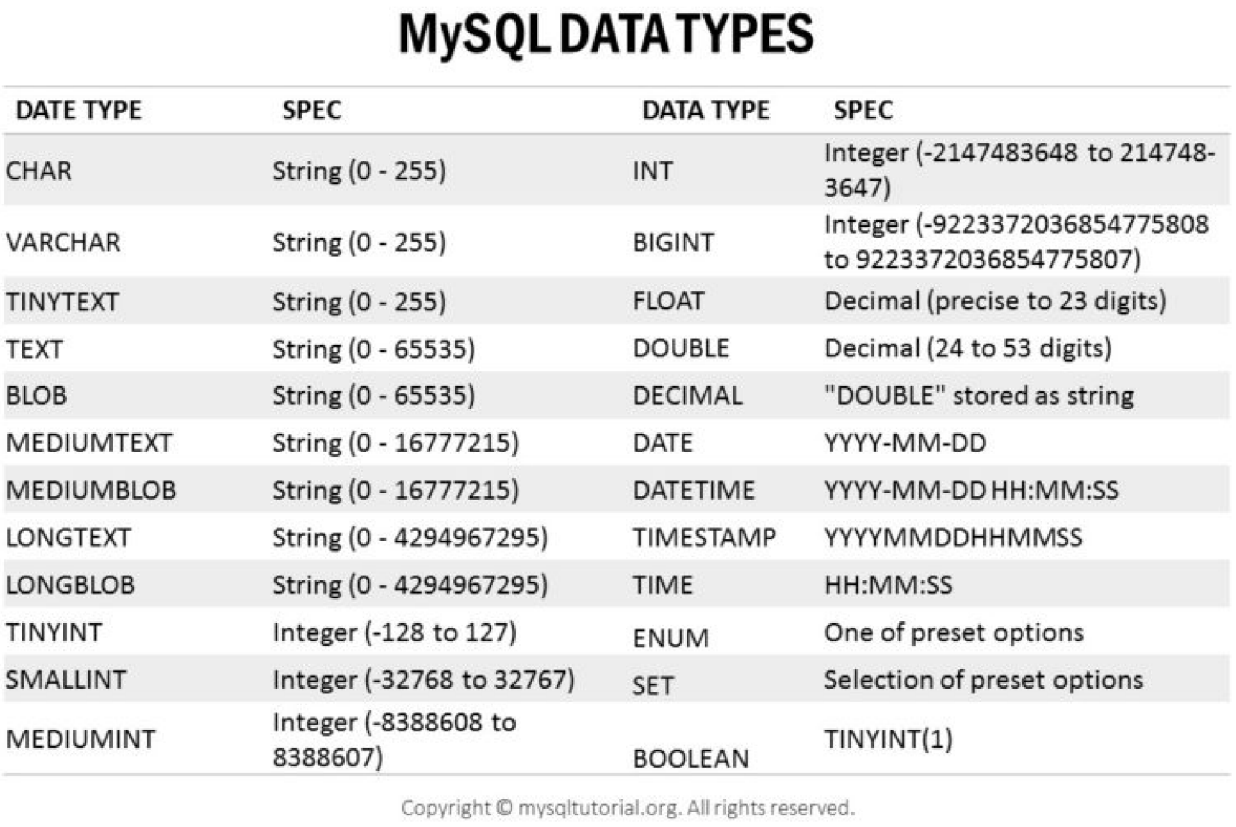
MySQLTUTORIAL

MySQL Data Types

Summary: in this tutorial, you will learn about **MySQL data types** and how to use them effectively in designing databases in MySQL.

A database table contains multiple columns with specific data types such as numeric or string. MySQL provides more data types other than just numeric and string. Each data type in MySQL can be determined by the following characteristics:

- The kind of values it represents.
- The space that takes up and whether the values are a fixed-length or variable length.
- The values of the data type can be indexed or not.
- How MySQL compares the values of a specific data type.



MySQL DATA TYPES

DATE TYPE	SPEC	DATA TYPE	SPEC
CHAR	String (0 - 255)	INT	Integer (-2147483648 to 214748-3647)
VARCHAR	String (0 - 255)	BIGINT	Integer (-9223372036854775808 to 9223372036854775807)
TINYTEXT	String (0 - 255)	FLOAT	Decimal (precise to 23 digits)
TEXT	String (0 - 65535)	DOUBLE	Decimal (24 to 53 digits)
BLOB	String (0 - 65535)	DECIMAL	"DOUBLE" stored as string
MEDIUMTEXT	String (0 - 16777215)	DATE	YYYY-MM-DD
MEDIUMBLOB	String (0 - 16777215)	DATETIME	YYYY-MM-DD HH:MM:SS
LONGTEXT	String (0 - 4294967295)	TIMESTAMP	YYYYMMDDHHMMSS
LONGBLOB	String (0 - 4294967295)	TIME	HH:MM:SS
TINYINT	Integer (-128 to 127)	ENUM	One of preset options
SMALLINT	Integer (-32768 to 32767)	SET	Selection of preset options
MEDIUMINT	Integer (-8388608 to 8388607)	BOOLEAN	TINYINT(1)

(https://www.mysqltutorial.org/wp-content/uploads/0211/03/MySQL-Data-Types.jpg)

Download MySQL Data Types Overview (https://www.mysqltutorial.org/wp-content/uploads/2018/03/MySQL-Data-Types.pdf)

MySQL numeric data types

In MySQL, you can find all SQL standard numeric types including exact number data type and approximate numeric data types including integer, fixed-point and floating-point. In addition, MySQL also has `BIT` (https://www.mysqltutorial.org/mysql-bit/) data type for storing bit values. Numeric types can be signed or unsigned except for the `BIT` type.

The following table shows the summary of numeric types in MySQL:

Numeric Types	Description
`TINYINT` (https://www.mysqltutorial.org/mysql-int/)	A very small integer
`SMALLINT` (https://www.mysqltutorial.org/mysql-int/)	A small integer
`MEDIUMINT` (https://www.mysqltutorial.org/mysql-int/)	A medium-sized integer
`INT` (https://www.mysqltutorial.org/mysql-int/)	A standard integer
`BIGINT` (https://www.mysqltutorial.org/mysql-int/)	A large integer
`DECIMAL` (https://www.mysqltutorial.org/mysql-decimal/)	A fixed-point number
`FLOAT`	A single-precision floating point number
`DOUBLE`	A double-precision floating point number
`BIT` (https://www.mysqltutorial.org/mysql-bit/)	A bit field

MySQL Boolean data type

MySQL does not have the built-in `BOOLEAN` (https://www.mysqltutorial.org/mysql-boolean/) or `BOOL` (https://www.mysqltutorial.org/mysql-boolean/) data type. To represent boolean values, MySQL uses the smallest integer type which is `TINYINT(1)`. In other words, `BOOLEAN` and `BOOL` are synonyms for `TINYINT(1).`

MySQL String data types

In MySQL, a string can hold anything from plain text to binary data such as images or files. Strings can be compared and searched based on pattern matching by using the `LIKE` (https://www.mysqltutorial.org/mysql-like/) operator, regular expression (https://www.mysqltutorial.org/mysql-regular-expression-regexp.aspx) , and full-text search (https://www.mysqltutorial.org/mysql-full-text-search.aspx) .

The following table shows the string data types in MySQL:

String Types	Description
CHAR (https://www.mysqltutorial.org/mysql-char-data-type/)	A fixed-length nonbinary (character) string
VARCHAR (https://www.mysqltutorial.org/mysql-varchar/)	A variable-length non-binary string
BINARY	A fixed-length binary string
VARBINARY	A variable-length binary string
TINYBLOB	A very small BLOB (binary large object)
BLOB	A small BLOB
MEDIUMBLOB	A medium-sized BLOB
LONGBLOB	A large BLOB
TINYTEXT (https://www.mysqltutorial.org/mysql-text/)	A very small non-binary string
TEXT (https://www.mysqltutorial.org/mysql-text/)	A small non-binary string
MEDIUMTEXT (https://www.mysqltutorial.org/mysql-text/)	A medium-sized non-binary string
LONGTEXT (https://www.mysqltutorial.org/mysql-	A large non-binary string

String Types	Description
text/)	
ENUM (https://www.mysqltutorial.org/mysql-enum/)	An enumeration; each column value may be assigned one enumeration member
SET	A set; each column value may be assigned zero or more SET members

MySQL date and time data types

MySQL provides types for date and time as well as the combination of date and time. In addition, MySQL supports the timestamp (https://www.mysqltutorial.org/mysql-timestamp.aspx) data type for tracking the changes in a row of a table. If you just want to store years without dates and months, you can use the `YEAR` data type.

The following table illustrates the MySQL date and time data types:

Date and Time Types	Description
DATE (https://www.mysqltutorial.org/mysql-date/)	A date value in CCYY-MM-DD format
TIME (https://www.mysqltutorial.org/mysql-time/)	A time value in hh:mm:ss format
DATETIME (https://www.mysqltutorial.org/mysql-datetime/)	A date and time value inCCYY-MM-DD hh:mm:ssformat
TIMESTAMP (https://www.mysqltutorial.org/mysql-timestamp.aspx)	A timestamp value in CCYY-MM-DD hh:mm:ss format
YEAR	A year value in CCYY or YY format

MySQL spatial data types

MySQL supports many spatial data types that contain various kinds of geometrical and geographical values as shown in the following table:

Spatial Data Types	Description
`GEOMETRY`	A spatial value of any type
`POINT`	A point (a pair of X-Y coordinates)
`LINESTRING`	A curve (one or more `POINT` values)
`POLYGON`	A polygon
`GEOMETRYCOLLECTION`	A collection of `GEOMETRY`values
`MULTILINESTRING`	A collection of `LINESTRING`values
`MULTIPOINT`	A collection of `POINT`values
`MULTIPOLYGON`	A collection of `POLYGON`values

JSON data type

MySQL supported a native `JSON (https://www.mysqltutorial.org/mysql-json/)` data type since version 5.7.8 that allows you to store and manage JSON documents more effectively. The native JSON data type provides automatic validation of JSON documents and optimal storage format.

In this tutorial, you have learned various MySQL data types that help you determine which data type you should use for columns when you create tables (https://www.mysqltutorial.org/mysql-create-table/) .