

# TABLAS DE TIPOS DE DATOS MYSQL Y SQLITE

## TIPOS DE DATOS MYSQL

Tipos de datos MySQL	
Tipos de datos de texto (String)	
Tipo	Descripción
CHAR(size)	Una cadena de longitud FIJA (puede contener letras, números y caracteres especiales). El parámetro de tamaño especifica la longitud de la columna en caracteres; puede ser de 0 a 255. El valor predeterminado es 1.
VARCHAR(size)	Una cadena de longitud VARIABLE (puede contener letras, números y caracteres especiales). El parámetro de tamaño especifica la longitud máxima de la columna en caracteres; puede ser de 0 a 65535.
BINARY(size)	Igual a CHAR (), pero almacena cadenas de bytes binarios. El parámetro de tamaño especifica la longitud de la columna en bytes. El valor predeterminado es 1.
VARBINARY(size)	Igual a VARCHAR (), pero almacena cadenas de bytes binarios. El parámetro de tamaño especifica la longitud máxima de la columna en bytes.
TINYBLOB	Para BLOB (objetos grandes binarios). Longitud máxima: 255 bytes.
TINYTEXT	Contiene una cadena con una longitud máxima de 255 caracteres.
TEXT(size)	Contiene una cadena con una longitud máxima de 65 535 bytes.
BLOB(size)	Para BLOB (objetos grandes binarios). Contiene hasta 65,535 bytes de datos.
MEDIUMTEXT	Contiene una cadena con una longitud máxima de 16.777.215 caracteres.
MEDIUMBLOB	Para BLOB (objetos grandes binarios). Contiene hasta 16.777.215 bytes de datos.
LONGTEXT	Contiene una cadena con una longitud máxima de 4.294.967.295 caracteres.
LOBLOB	Para BLOB (objetos grandes binarios). Contiene hasta 4.294.967.295 bytes de datos.
ENUM(val1, val2, val3, ...)	Un objeto de texto que solo puede tener un valor, elegido de una lista de valores posibles. Puede enumerar hasta 65535 valores en una lista ENUM. Si se inserta un valor que no está en la lista, se insertará un valor en blanco. Los valores se ordenan en el orden en que los ingresa.
SET(val1, val2, val3, ...)	Un objeto de texto que puede tener 0 o más valores, elegido de una lista de valores posibles. Puede enumerar hasta 64 valores en una lista SET.
Tipos de datos numéricos	

Tipo	Descripción
BIT (size)	Un tipo de valor de bit. El número de bits por valor se especifica en tamaño. El parámetro de tamaño puede contener un valor de 1 a 64. El valor predeterminado para el tamaño es 1.
TINYINT (size)	Un número entero muy pequeño. El rango con signo es de -128 a 127. El rango sin signo es de 0 a 255. El parámetro de tamaño especifica el ancho máximo de visualización (que es 255).
BOOL	Equivalente al tipo <b>boolean</b> en cualquier lenguaje de programación. El cero se considera falso, los valores distintos de cero se consideran verdaderos.
BOOLEAN	Exactamente igual que BOOL.
SMALLINT (size)	Un pequeño entero. El rango con signo es de -32768 a 32767. El rango sin signo es de 0 a 65535. El parámetro de tamaño especifica el ancho máximo de visualización (que es 255).
MEDIUMINT (size)	Un número entero medio. El rango con signo es de -8388608 a 8388607. El rango sin signo es de 0 a 16777215. El parámetro de tamaño especifica el ancho máximo de visualización (que es 255).
INT (size)	Un número entero medio. El rango con signo es de -2147483648 a 2147483647. El rango sin signo es de 0 a 4294967295. El parámetro de tamaño especifica el ancho máximo de visualización (que es 255).
INTEGER (size)	Exactamente igual que INT.
BIGINT (size)	Un entero grande. El rango con signo es de -9223372036854775808 a 9223372036854775807. El rango sin signo es de 0 a 18446744073709551615. El parámetro de tamaño especifica el ancho máximo de visualización (que es 255).
FLOAT (size, d)	Este tipo de datos ya no funciona, ha sido eliminado. Lo hemos dejado porque al buscar ejemplos en Internet podrías encontrar algo de este estilo si la información no está actualizada, lo cual no funcionará.
FLOAT (p)	Un número de coma flotante. MySQL usa el valor <b>p</b> para determinar si usar FLOAT o DOUBLE para el tipo de datos resultante. Si <b>p</b> es de 0 a 24, el tipo de datos se convierte en FLOAT (). Si <b>p</b> es de 25 a 53, el tipo de datos se convierte en DOUBLE ()
DOUBLE (size, d)	Un número de coma flotante de tamaño normal. El número total de dígitos se especifica en tamaño. El número de dígitos después del punto decimal se especifica en el parámetro d.
DOUBLE PRECISION (size, d)	
DECIMAL (size, d)	Un número de punto fijo exacto. El número total de dígitos se especifica en tamaño. El número de dígitos después del punto decimal se especifica en el parámetro d. El número máximo para el tamaño es 65. El número máximo para d es 30. El valor predeterminado para el tamaño es 10. El valor predeterminado para d es 0.
DEC (size, d)	Exactamente igual que DECIMAL.
Tipos de datos de fecha y hora	

Tipo	Descripción
DATE	Una fecha. Formato: <b>AAAA-MM-DD</b> . El rango admitido es de '1000-01-01' a '9999-12-31'.
DATETIME (fsp)	Una combinación de fecha y hora. Formato: <b>AAAA-MM-DD hh: mm: ss</b> . El rango admitido es de '1000-01-01 00:00:00' a '9999-12-31 23:59:59'. Agregar DEFAULT y ON UPDATE en la definición de la columna para obtener la inicialización automática y la actualización a la fecha y hora actuales.
TIMESTAMP (fsp)	Una marca de tiempo. Los valores de TIMESTAMP se almacenan como el número de segundos desde la época de Unix ('1970-01-01 00:00:00' UTC). Formato: <b>AAAA-MM-DD hh: mm: ss</b> . El rango admitido es de '1970-01-01 00:00:01' UTC a '2038-01-09 03:14:07' UTC. La inicialización automática y la actualización a la fecha y hora actuales se pueden especificar usando DEFAULT CURRENT_TIMESTAMP y ON UPDATE CURRENT_TIMESTAMP en la definición de columna.
TIME (fsp)	Una hora concreta. Formato: <b>hh: mm: ss</b> . El rango admitido es de '-838: 59: 59' a '838: 59: 59'.
YEAR	Un año en formato de cuatro dígitos. Valores permitidos en formato de cuatro dígitos: 1901 a 2155 y 0000.

## TIPOS DE DATOS SQLITE

Tipos de datos SQLite	
Tipo	Descripción
NULL	Los valores NULL significan información faltante o desconocida.
INTEGER	Los valores enteros son números enteros (positivos o negativos). Un número entero puede tener tamaños variables como 1, 2, 3, 4 u 8 bytes.
REAL	Los valores reales son números reales con valores decimales que utilizan flotantes de 8 bytes.
TEXT	TEXT se utiliza para almacenar datos de caracteres. La longitud máxima de TEXT es ilimitada. SQLite admite varias codificaciones de caracteres.
BLOB	BLOB significa un objeto binario grande que puede almacenar cualquier tipo de datos. El tamaño máximo de BLOB es, teóricamente, ilimitado.

Comparado con MySQL parecería que SQLite tiene muy pocos tipos de datos, sin embargo la brutal diferencia está en el hecho de que los tipos de datos de SQLite son dinámicos y, por tanto, se establecen al momento de guardarse en la base de datos. De este modo los tipos de datos SQLite tienen total afinidad con los tipos de datos de MySQL y otros gestores de bases de datos, lo cual quiere decir, por ejemplo, que el tipo VARCHAR será transformado automáticamente a TEXT. A continuación dejo una tabla con las afinidades de tipos:

Afinidad de tipos con SQLite	
Tipo no SQLite	Tipo SQLite afín
INT	INTEGER
INTEGER	
TINYINT	
SMALLINT	
MEDIUMINT	
BIGINT	

UNSIGNED BIG INT	
INT2	
INT8	
CHARACTER (20)	TEXT
VARCHAR (255)	
VARYING CHARACTER (255)	
NCHAR (55)	
NATIVE CHARACTER (70)	
NVARCHAR (100)	
TEXT	
CLOB	
BLOB	BLOB
REAL	REAL
DOUBLE	
DOUBLE PRECISION	
FLOAT	
BOOL	Se guardan como INTEGER pero se le coloca valores de 0 (falso) y 1 (verdadero).
BOOLEAN	