## Tipos de dados numéricos no MySQL

O MySQL tem todos os tipos numéricos possíveis, o que inclui exatos, aproximados, inteiros, de ponto fixo, ponto flutuante etc. Aqui estão alguns exemplos:

* TINYINT — número inteiro extremamnte pequeno (*tiny*);
* SMALLINT — número inteiro pequeno;
* MEDIUMINT — número médio;
* INT — número inteiro de tamanho normal;
* BIGINT — número inteiro de tamanho grande;
* DECIMAL — número decimal, de ponto fixo;
* FLOAT — número de ponto flutuante de precisão simples (32 bits);
* DOUBLE — número de ponto flutuante de precisão dupla (64 bits);
* BIT — um campo de um bit.

## Tipos de dados em strings

*Strings* são cadeias de caracteres. No MySQL, uma string pode ter qualquer conteúdo. Cadeias de caracteres podem ser comparadas e ser objeto de buscas.

* CHAR — uma cadeia de caracteres (*string*), de tamanho fixo e não-binária;
* VARCHAR — uma string variável e não-binária;
* BINARY — uma string binária fixa;
* VARBINARY — uma string binária variável;
* BLOB — um BLOB (*Binary Large OBject* – OBjeto Grande Binário) pequeno;
* TINYBLOB — um BLOB extremamente pequeno;
* MEDIUMBLOB — um BLOB de tamanho médio;
* LONGBLOB — um BLOB grande;
* TINYTEXT — uma string não-binária e de tamanho muito reduzido;
* TEXT — uma string não-binária e pequena;
* MEDIUMTEXT — uma string de tamanho normal e não-binária;
* LONGTEXT — uma string não-binária de tamanho grande;
* ENUM — string com um valor que precisa ser selecionado de uma lista predefinida na criação da tabela;
* SET — é um objeto que pode ter zero ou mais valores – cada um dos quais precisa ser escolhido de uma lista de valores predeterminados quando da criação da tabela.

## Armazenamento de data e hora

Há várias opções para armazenar dados relacionados a data e hora. Veja os tipos e suas descrições:

* DATE — o valor referente a uma data no formato 'CCYY-MM-DD';
* TIME — um valor horário no formato 'hh:mm:ss';
* TIMESTAMP — *timestamp* é uma sequência de caracteres ou informação codificada que identifica uma marca temporal ou um dado momento em que um evento ocorreu. No MySQL, ele tem o formato 'CCYY-MM-DD hh:mm:ss';
* YEAR — armazena um ano no formato 'CCYY' ou 'YY';

## Dados espaciais

O MySQL tem suporte a tipos de dados que correspondem às classes *OpenGIS*. Alguns destes carregam valores geométricos simples:

* GEOMETRY
* POINT
* LINESTRING
* POLYGON

O GEOMETRY pode armazenar qualquer tipo de valor geométrico. Os outros valores simples (POINT, LINESTRING e POLYGON) têm seus valores restritos aos tipos geométricos a que se referem.  
Os outros, que seguem listados, carregam valores relativos a coleções/coletivos:

* GEOMETRYCOLLECTION
* MULTILINESTRING
* MULTIPOINT
* MULTIPOLYGON

Assim, GEOMETRYCOLLECTION pode armazenar coletâneas de objetos de qualquer tipo. Os outros tipos coletivos (MULTILINESTRING, MULTIPOLYGON e GEOMETRYCOLLECTION) restringem-se a cada forma geométrica particular.

## Tabela de Tipos

|  |  |
| --- | --- |
| **Char** | Armazena uma cadeia de tamanho fixo. A cadeia poderá conter desde 0 até 255 caracteres |
| **VarChar** | Armazena uma cadeia de tamanho variável. A cadeia poderá conter desde 0 até 255 caracteres. Dentro dos tipos de cadeia pode-se distinguir dois subtipos, os tipo Test e os tipo Blob (Binary Large Object) A diferença entre um tipo e outro é o tratamento que recebem na hora de ordená-los e compará-los. No tipo test ordena-se sem ter importância as maiúsculas e as minúsculas e no tipo blob ordena-se tendo em conta as maiúsculas e minúsculas. |
| **TinyText/TinyBlob** | Coluna com uma longitude máxima de 255 caracteres |
| **Blob/Text** | Texto com um máximo de 65535 caracteres |
| **MediumBlob/MediumText** | Texto com um máximo de 16.777.215 caracteres |
| **LongBlob/LongText** | Texto com um máximo de caracteres 4.294.967.295. Há que se ter em conta que, devido aos protocolos de comunicação, os pacotes podem ter um máximo de 16 Mb |
| **Enum** | Campo que pode ter um único valor de uma lista que se especifica. O tipo Enum aceita até 65535 valores diferentes |
| **Set** | Campo que pode conter nenhum, um ou vários valores de uma lista. A lista pode ter um máximo de 64 valores |
| **Bit/Bool** | Número inteiro que pode ser 0 ou 1 |
| **Int** | De -2147483648 até 2147483647 ou de 0 até 429.496.295 |
| **SmallInt** | De -32768 até 32767 ou de 0 até 65535 |
| **TinyInt** | De -128 até 127 ou de 0 a 255 |
| **MediumInt** | De -8.388.608 até 8.388.607 ou de 0 até 16777215 |
| **BigInt** | De -9.223.372.036.854.775.808 até 9.223.372.036.854.775.807 ou de 0 até 18.446.744.073.709.551.615 |
| **Float** | De -3.402823466E+38 até -1.175494351E-38,0 ou de 175494351E-38 até 3.402823466E+38 |
| **xReal/Double** | De -1.7976931348623157E+308 até -2.2250738585072014E-308 ou de 2.2250738585072014E-308 até 1.7976931348623157E+308 |
| **Decimal/Dec/Numeric** | Número em vírgula flutuante desempacotado. O número armazena-se em uma cadeia |
| **Date** | A margem de valores vai desde o 1 de Janeiro de 1001 ao 31 de dezembro de 9999. O formato de armazenamento é de ano-mes-dia |
| **DateTime** | Combinação de data e hora. A margem de valores vai desde o 1º de Janeiro de 1001 às 0 horas, 0 minutos e 0 segundos ao 31 de Dezembro de 9999 às 23 horas, 59 minutos e 59 segundos. O formato de armazenamento é de ano-mes-dia horas:minutos:segundos |
| **TimeStamp** | Combinação de data e hora. A margem vai desde 1º de Janeiro de 1970 ao ano 2037. O formato de armazenamento depende do tamanho do campo |
| **Time** | Armazena uma hora. A margem de horas vai desde -838 horas, 59 minutos e 59 segundos. O formato de armazenamento é ‘HH:MM:SS’ |
| **Year** | Armazena um ano. A margem de valores permitidos vai desde o ano 1901 ao ano 2155. O campo pode ter tamanho dois ou tamanho 4 dependendo de se queremos armazenar o ano com dois ou quatro algarismos |