



MySQL - Tipos de Dados

Principais Tipos de Dados

Existem vários tipos de dados que podem ser armazenados em um banco de dados MySQL. A utilização ou não de qualquer tipo de dado, entretanto, é uma decisão que cabe ao **DBA**, ou **Database Administrator**, pois cada banco de dados possui um contexto em que será utilizado.

Obs.: Caso tenham interesse em utilizar um certo tipo de dado em seus projetos relacionados a banco de dados MySQL, vocês podem, e devem, sempre estar checando a [documentação](#).

Tipos Numéricos

▼ Valores Inteiros

Para representar números inteiros, são utilizados tipos como **TINYINT**, **SMALLINT**, **MEDIUMINT**, **INT** (ou **INTEGER**) e **BIGINT**.

Type	Storage (Bytes)	Minimum Value Signed	Minimum Value Unsigned	Maximum Value Signed	Maximum Value Unsigned
TINYINT	1	-128	0	127	255
SMALLINT	2	-32768	0	32767	65535
MEDIUMINT	3	-8388608	0	8388607	16777215
INT	4	-2147483648	0	2147483647	4294967295
BIGINT	8	-2 ⁶³	0	2 ⁶³ -1	2 ⁶⁴ -1

Valor que cada um destes tipos de dados podem armazenar.

▼ Valores Exatos

Os tipos **DECIMAL** ou **NUMERIC** armazenam valores com uma certa quantidade de dígitos e uma certa quantidade de decimais. Por exemplo, em um determinado registro, para armazenar o valor decimal "11.61", podemos declarar uma variável seguinte maneira:

```
nome_da_variavel DECIMAL(2, 2);
```

▼ Valores Aproximados (Ponto-Flutuante)

Os tipos `FLOAT` e `DOUBLE` representam valores de dados numéricos aproximados. É importante observar que o MySQL usa 4 bytes para valores `FLOAT` e 8 bytes para valores `DOUBLE`.

▼ Valores em Bits

O tipo de dados `BIT` é usado para armazenar valores de bits. O tipo `BIT(M)` permite o armazenamento de valores de até M bits, onde M é um valor que pode variar de 1 a 64.

Tipos de String

▼ Quantidade de Caracteres

Os tipos `CHAR` e `VARCHAR` são parecidos, mas diferem na maneira como são armazenados. Eles também diferem no comprimento máximo e se os espaços à direita são retidos ou não. Eles são declarados com um comprimento que indica o número máximo de caracteres que desejamos armazenar. Por exemplo, `CHAR(30)` pode conter até 30 caracteres.

Value	CHAR(4)	Storage Required	VARCHAR(4)	Storage Required
' '	' '	4 bytes	' '	1 byte
'ab '	'ab '	4 bytes	'ab '	3 bytes
'abcd'	'abcd'	4 bytes	'abcd'	5 bytes
'abcdefgh'	'abcd'	4 bytes	'abcd'	5 bytes

Diferenças entre `CHAR` e `VARCHAR`, mostrando o resultado do armazenamento de vários valores de string nas colunas `CHAR(4)` e `VARCHAR(4)`.

Tipos de Data & Hora

Os tipos de dados de data e hora para representar valores temporais são `DATE`, `TIME`, `DATETIME`, `TIMESTAMP` e `YEAR`. Cada tipo temporal tem um intervalo de valores válidos, bem como um valor “zero” que pode ser usado quando você especifica um valor inválido que o MySQL não pode representar.

Obs.: Para saberem mais a respeito, recomendo que chequem a [documentação](#), pois falar destes tipos em aula e de suas funcionalidades seria um pouco extenso. Mesmo assim, eventualmente iremos realizar algum exercício mais simples com alguns deles, onde demonstrarei a forma de inserir dados de alguns destes tipos.