

Типы данных и наборы СИМВОЛОВ

Наборы символов:

Список установленных наборов

```
SHOW CHARACTER SET;
```

По умолчанию utf 8mb4

varchar(20) character set latin1 : при создании таблицы другой набор

набор по-умолчанию для всей таблицы

```
create database european_sales character set latin1;
```

Общая информация

- M максимальное количество. Для целых - кол-во цифр, для дробных - общее кол-во цифр, для строк макс. кол-во символов
- D кол-во цифр после запятой, должно быть не больше M-2
- fsp применимо к дате/времени, определяет точность (кол-во дробной части секунды)
- при strict mode перебор значения вызывает исключение, при restrictive - максимально возможное значение

Тип	Доп. информация
char(255 символов) varchar(65535)	char(20) 20 символов char фиксировано, при хранении добавляет справа PAD символы
tinytext(255) text(65535) mediumtext(16 млн) longtext(4 млрд)	если больше, то обрежутся конечные пробелы считаются при сортировке первые 1024 байта, но можно увеличить tinytext, text - можно не использовать

Тип	Доп. информация
tinyint(1) smallint(2) mediumint int bigint(8)	<p>указано кол-во байтов при создании указывается визуальное кол-во цифр+один символ на знак</p> <pre>CREATE TABLE employees (salary INTEGER(5) UNSIGNED);</pre> <p>int максимум 10, поэтому это преобразуется в int(11)</p> <pre>CREATE TABLE employees (id INT(255));</pre> <p>ZEROFILL заполняет 0 слева.</p> <pre>CREATE TABLE documents (document_no INT(5) ZEROFILL);</pre>
DECIMAL = NUMERIC	<p>фиксированное кол-во цифр. Равно int с коэффициентом деления. Первая цифра - общее кол-во цифр (M), вторая - кол-во цифр после запятой. Знак неявно добавляется. Т.е. от - 9.99 до 9.99</p> <pre>CREATE TABLE taxes (tax_rate DECIMAL(3, 2));</pre> <p>M < 65, по-умолчанию 10 при переборе ошибка, при выходе за пределы точности - округление по математическим правилам способ вывода больших чисел с игнором младших цифр</p> <pre>select truncate(tax_rate * 1000, 0) from taxes;</pre> <p>в агрегатных функциях работает общее ограничение на 64 цифры, вычисление точное.</p>

Тип	Доп. информация
float, double	<p>приближенные числа отображаются как были внесены, но в агрегатных функциях могут появиться неточности</p> <div><pre>CREATE TABLE typed_numbers(id TINYINT, float_values FLOAT, decimal_values DECIMAL(3, 2)); INSERT INTO typed_numbers VALUES (1, 1.1, 1.1), (2, 1.1, 1.1), (3, 1.1, 1.1); SELECT * FROM typed_numbers; ┌ id float_values decimal_values ├+-----+-----+-----+ ┌ 1 1.1 1.10 ┌ 2 1.1 1.10 ┌ 3 1.1 1.10 SELECT SUM(float_values), SUM(decimal_values) FROM typed_numbers; ┌ SUM(float_values) SUM(decimal_values) ├+-----+-----+ ┌ 3.3000000715255737 3.30 </pre></div> <p>Нельзя напрямую сравнивать в агрегатах, нужно сравнивать разницу $ABS(v1 - v2) > 0.1$</p> <div><pre>SELECT id, SUM(col1) as v1, SUM(col2) as v2 FROM temp GROUP BY id HAVING v1 <> v2;</pre></div> <p>нельзя, внешне одинаковые числа будут разными</p>

Тип	Доп. информация
битовый тип	<div data-bbox="798 154 1503 763" data-label="Text"> <p>column_name BIT(M)</p> <div data-bbox="815 208 1485 618" data-label="Code-Block"> <pre>CREATE TABLE bit_values (val BIT(7)); INSERT INTO bit_values VALUES(b'1011'); - равно VALUES(b'0001011'), в этой нотации b и B равны. INSERT INTO bit_values VALUES(0b1011); INSERT INTO bit_values VALUES(2); - автоматическое преобразование числа в битовую маску</pre> </div> <p>select выводит в виде 0x0A select bin(...) - без 0 слева select lpad(bin(...), 7, '0') - привычный вид с 0 слева</p> </div>

Тип	Доп. информация
date datetime timestamp year time	<div>00-00-0000 вместо NULL, не отображается</div> <div><div>%a Краткое имя дня недели - Sun, Mon, ... %b Краткое имя месяца — Jan, Feb, ... %c Числовое значение месяца (0..11) %d Числовое значение дня месяца (00..31) %f Число микросекунд (000000..999999) %H Час дня в 24-часовом формате (00..23) %h Час дня в 12-часовом формате (01..12) %i Минуты в часе (00..59) %j День года (001..366) %M Полное имя месяца (January..December) %m Числовое значение месяца %p AM или PM %s Число секунд (00..59) %W Полное имя дня недели (Sunday..Saturday) %w Числовое значение дня недели (0=Sunday..6=Saturday) %Y Значение года (четыре цифры)</div></div> <div>timestamp зависит от time_zone</div> <div><div>CREATE TABLE datetime_temp(ts TIMESTAMP, dt DATETIME); INSERT INTO datetime_temp VALUES(NOW(), NOW()); SELECT ts, dt FROM datetime_temp; ☐ 2017-10-14 18:10:25 2017-10-14 18:10:25 SET time_zone = '+03:00'; SELECT ts, dt FROM datetime_temp; ☐ 2017-10-14 21:10:25 2017-10-14 18:10:25 </div></div> <div>Если сервер MAXDB mode, TIMESTAMP = DATETIME column_name TIME;</div> <div><div>838:59:59 to 838:59:59 + 6 цифр миллисекунд некорректное в 00:00:00, нельзя отличить от корректно добавленного 00:00:00</div></div>

