

## Числовые типы данных

Тип данных	Объем памяти	Диапазон	Описание
TINYINT	1 байт	от -128 до 127 или от 0 до 255	<p>Целое число. Может быть объявлено положительным с помощью ключевого слова UNSIGNED, тогда элементам столбца нельзя будет присвоить отрицательное значение. Необязательный параметр M - количество отводимых под число символов. Необязательный атрибут ZEROFILL позволяет свободные позиции по умолчанию заполнить нулями.</p> <p><i>Примеры:</i></p> <p>TINYINT - хранит любое число в диапазоне от -128 до 127.</p> <p>TINYINT UNSIGNED - хранит любое число в диапазоне от 0 до 255.</p> <p>TINYINT ZEROFILL - свободные позиции слева заполнит нулями. Например, величина 2 будет отображаться, как 002.</p>
SMALLINT	2 байта	от -32768 до 32767 или от 0 до 65535	<p>Аналогично предыдущему, но с большим диапазоном.</p> <p><i>Примеры:</i></p> <p>SMALLINT - хранит любое число в диапазоне от -32768 до 32767.</p> <p>SMALLINT UNSIGNED - хранит любое число в диапазоне от 0 до 65535.</p> <p>SMALLINT ZEROFILL - свободные позиции слева заполнит нулями. Например, величина 2 будет отображаться, как 0002.</p>
MEDIUMINT	3 байта	от -8388608 до 8388608 или от 0 до 16777215	<p>Аналогично предыдущему, но с большим диапазоном.</p> <p><i>Примеры:</i></p> <p>MEDIUMINT - хранит любое число в диапазоне от -8388608 до 8388608.</p> <p>MEDIUMINT UNSIGNED - хранит любое число в диапазоне от 0 до 16777215.</p> <p>MEDIUMINT ZEROFILL - свободные позиции слева заполнит нулями. Например, величина 2 будет отображаться, как 00002.</p>
INT или INTEGER	4 байта	от -2147683648 до 2147683648 или от 0 до 4294967295	<p>Аналогично предыдущему, но с большим диапазоном.</p> <p><i>Примеры:</i></p> <p>INT - хранит любое число в диапазоне от -2147683648 до 2147683648.</p> <p>INT UNSIGNED - хранит любое число в диапазоне от 0 до 4294967295.</p> <p>INT ZEROFILL - свободные позиции слева заполнит</p>

			нулями. Например, величина 2 будет отображаться, как 00002.
BIGINT	8 байта	от $-2^{63}$ до $2^{63}-1$ или от 0 до $2^{64}$	<p>Аналогично предыдущему, но с большим диапазоном.</p> <p><i>Примеры:</i></p> <p>BIGINT - хранит любое число в диапазоне от <math>-2^{63}</math> до <math>2^{63}-1</math>.</p> <p>BIGINT UNSIGNED - хранит любое число в диапазоне от 0 до <math>2^{64}</math>.</p> <p>BIGINT ZEROFILL - свободные позиции слева заполнит нулями. Например, величина 2 будет отображаться, как 0000002.</p>
BOOL или BOOLEAN	1 байт	либо 0, либо 1	Булево значение. 0 - ложь (false), 1 - истина (true).
DECIMAL (M,D) или DEC (M,D) или NUMERIC (M,D)	M + 2 байта	зависят от параметров M и D	<p>Используются для величин повышенной точности, например, для денежных данных. M - количество отводимых под число символов (максимальное значение - 64). D - количество знаков после запятой (максимальное значение - 30).</p> <p><i>Пример:</i></p> <p>DECIMAL (5,2) - будет хранить числа от -99,99 до 99,99.</p>
FLOAT (M,D)	4 байта	мин. значение $+(-)$ $1.175494351 * 10^{-39}$ макс. значение $+(-)$ $3.402823466 * 10^{38}$	<p>Вещественное число (с плавающей точкой). Может иметь параметр UNSIGNED, запрещающий отрицательные числа, но диапазон значений от этого не изменится. M - количество отводимых под число символов. D - количество символов дробной части.</p> <p><i>Пример:</i></p> <p>FLOAT (5,2) - будет хранить числа из 5 символов, 2 из которых будут идти после запятой (например: 46,58).</p>
DOUBLE (M,D)	8 байт	мин. значение $+(-)$ $2.2250738585072015$ $* 10^{-308}$ макс. значение $+(-)$ $1.797693134862315$ $* 10^{308}$	<p>Аналогично предыдущему, но с большим диапазоном.</p> <p><i>Пример:</i></p> <p>DOUBLE - будет хранить большие дробные числа.</p>

Необходимо понимать, чем больше диапазон значений у типа данных, тем больше памяти он занимает. Поэтому, если предполагается, что значения в столбце не будут превышать 100, то используйте тип TINYINT. Если при этом все значения будут положительными, то используйте атрибут UNSIGNED. Правильный выбор типа данных позволяет сэкономить место для хранения этих данных.

### Строковые типы данных

Тип данных	Объем памяти	Максимальный размер	Описание
CHAR (M)	M символов	M символов	<p>Позволяет хранить <b>строку фиксированной длины</b> M. Значение M - от 0 до 65535.</p> <p><i>Примеры:</i></p> <p>CHAR (8) - хранит строки из 8 символов и занимает 8 байтов. Например, любое из следующих значений: "", 'Иван', 'Ирина', 'Сергей' будет занимать по 8 байтов памяти. А при попытке ввести значение 'Александра', оно будет усечено до 'Александ', т.е. до 8 символов.</p>
VARCHAR (M)	L+1 символов	M символов	<p>Позволяет хранить <b>переменные строки</b> длиной L. Значение M - от 0 до 65535.</p> <p><i>Примеры:</i></p> <p>VARCHAR (3) - хранит строки максимум из 3 символов, но пустая строка "" занимает 1 байт памяти, строка 'а' - 2 байта, строк 'аа' - 3 байта, строка 'ааа' - 4 байта. Значение более 3 символов будет усечено до 3.</p>
BLOB, TEXT	L+2 символов	$2^{16}-1$ символов	Позволяют хранить большие объемы текста. Причем тип TEXT используется для хранения именно текста, а BLOB - для хранения изображений, звука, электронных документов и т.д.
MEDIUMBLOB, MEDIUMTEXT	L+3 символов	$2^{24}-1$ символов	Аналогично предыдущему, но с большим размером.
LOB, LONGTEXT	L+4 символов	$2^{32}-1$ символов	Аналогично предыдущему, но с большим размером.
ENUM ('value1', 'value2', ..., 'valueN')	1 или 2 байта	65535 элементов	<p>Строки этого типа могут принимать только одно из значений указанного множества.</p> <p><i>Пример:</i></p> <p>ENUM ('да', 'нет') - в столбце с таким типом может храниться только одно из имеющихся значений. Удобно использовать, если предусмотрено, что в столбце должен храниться ответ на вопрос.</p>
SET ('value1', 'value2', ..., 'valueN')	до 8 байт	64 элемента	<p>Строки этого типа могут принимать любой или все элементы из значений указанного множества.</p> <p><i>Пример:</i></p> <p>SET ('первый', 'второй') - в столбце с таким типом может храниться одно из перечисленных значений, оба сразу или значение может отсутствовать вовсе.</p>

### Календарные типы данных

Тип данных	Объем памяти	Диапазон	Описание
DATE	3 байта	от '1000-01-01' до '9999-12-31'	Предназначен для хранения даты. В качестве первого значения указывается год в формате "YYYY", через дефис - месяц в формате "MM", а затем день в формате "DD". В качестве разделителя может выступать не только дефис, а любой символ отличный от цифры.
TIME	3 байта	от '-838:59:59' до '838:59:59'	Предназначен для хранения времени суток. Значение вводится и хранится в привычном формате - hh:mm:ss, где hh - часы, mm - минуты, ss - секунды. В качестве разделителя может выступать любой символ отличный от цифры.
DATETIME	8 байт	от '1000-01-01 00:00:00' до '9999-12-31 23:59:59'	Предназначен для хранения и даты и времени суток. Значение вводится и хранится в формате - YYYY-MM-DD hh:mm:ss. В качестве разделителей могут выступать любые символы отличные от цифры.
TIMESTAMP	4 байта	от '1970-01-01 00:00:00' до '2037-12-31 23:59:59'	Предназначен для хранения даты и времени суток в виде количества секунд, прошедших с полуночи 1 января 1970 года (начало эпохи UNIX).
YEAR	1 байт	от 1901 до 2155 для M=4	Предназначен для хранения года. Например, YEAR - 1970.