





# Типы данных в PostgreSQL

Последнее обновление: 17.03.2018





При определении таблицы для всех ее столбцов необходимо указать тип данных. Тип данных определяет диапазон значений, которые могут храниться в столбце, сколько они будут занимать места в памяти. PostgreSQL поддерживает богатую палитру различных типов данных, среди которые условно можно разделить на подгруппы: числовые, символьные, логические, дата и время, бинарные и ряд других.

### Числовые типы данных

- **serial**: представляет автоинкрементирующееся числовое значение, которое занимает 4 байта и может хранить числа от 1 до 2147483647. Значение данного типа образуется путем автоинкремента значения предыдущей строки. Поэтому, как правило, данный тип используется для определения идентификаторов строки.
- **smallserial**: представляет автоинкрементирующееся числовое значение, которое занимает 2 байта и может хранить числа от 1 до 32767. Аналог типа serial для небольших чисел.
- **bigserial**: представляет автоинкрементирующееся числовое значение, которое занимает 8 байт и может хранить числа от 1 до 9223372036854775807. Аналог типа serial для больших чисел.
- smallint: хранит числа от -32768 до +32767. Занимает 2 байта. Имеет псевдоним int2.
- integer: хранит числа от -2147483648 до +2147483647. Занимает 4 байта. Имеет псевдонимы int и int4.
- **bigint**: хранит числа от -9223372036854775808 до +9223372036854775807. Занимает 8 байт. Имеет псевдоним **int8**.
- **numeric**: хранит числа с фиксированной точностью, которые могут иметь до 131072 знаков в целой части и до 16383 знаков после запятой.

Данный тип может принимать два параметра precision и scale: numeric(precision, scale).

Параметр precision указывает на максимальное количество цифр, которые может хранить число.

Параметр scale представляет максимальное количество цифр, которые может содержать число после запятой. Это значение должно находиться в диапазоне от 0 до значения параметра precision. По умолчанию оно равно 0.

Например, для числа 23.5141 precision равно 6, a scale - 4.

- **decimal**: хранит числа с фиксированной точностью, которые могут иметь до 131072 знаков в целой части и до 16383 знаков в дробной части. То же самое, что и numeric.
- **real**: хранит числа с плавающей точкой из диапазона от 1E-37 до 1E+37. Занимает 4 байта. Имеет псевдоним float4.
- **double precision**: хранит числа с плавающей точкой из диапазона от 1E-307 до 1E+308. Занимает 8 байт. Имеет псевдоним float8.

#### Примеры использования:

```
Id SERIAL,
TotalWeight NUMERIC(9,2),
Age INTEGER,
Surplus REAL
```

### Типы для работы с валютой (денежными единицами)

Для работы с денежными единицами определен тип **money**, который может принимать значения в диапазоне от -92233720368547758.08 до +92233720368547758.07 и занимает 8 байт.

#### Символьные типы

- **character(n)**: представляет строку из фиксированного количества символов. С помощью параметра задается задается количетво символов в строке. Имеет псевдоним **char(n)**.
- **character varying(n)**: представляет строку из фиксированного количества символов. С помощью параметра задается задается количетво символов в строке. Имеет псевдоним **varchar(n)**.
- text: представляет текст произвольной длины.

### Бинарные данные

Для хранения бинарных данных определен тип **bytea**. Он хранит данные в виде бинарных строк, которые представляют последовательность октетов или байт.

#### Типы для работы с датами и временем

- **timestamp**: хранит дату и время. Занимает 8 байт. Для дат самое нижнее значение 4713 г до н.э., самое верхнее значение 294276 г н.э.
- **timestamp with time zone**: то же самое, что и timestamp, только добавляет данные о часовом поясе.
- date: представляет дату от 4713 г. до н.э. до 5874897 г н.э. Занимает 4 байта.
- **time**: хранит время с точностью до 1 микросекунды без указания часового пояса. Принимает значения от 00:00:00 до 24:00:00. Занимает 8 байт.
- time with time zone: хранит время с точностью до 1 микросекунды с указанием часового пояса. Принимает значения от 00:00:00+1459 до 24:00:00-1459. Занимает 12 байт.

• interval: представляет временной интервал. Занимает 16 байт.

#### Распространенные форматы дат:

- yyyy-mm-dd 1999-01-08
- Month dd, yyyy January 8, 1999
- mm/dd/yyyy 1/8/1999

#### Распространенные форматы времени:

- hh:mi 13:21
- hh:mi am/pm 1:21 pm
- hh:mi:ss 1:21:34

#### Логический тип

Тип boolean может хранить одно из двух значений: true или false.

Вместо true можно указывать следующие значения: TRUE, 't', 'true', 'y', 'yes', 'on', '1'.

Bместо false можно указывать следующие значения: FALSE, 'f', 'false', 'n', 'no', 'off', '0'.

#### Типы для представления интернет-адресов

- cidr: интернет-адрес в формате IPv4 и IPv6. Например, 192.168.0.1. Занимает от 7 до 19 байт.
- inet: интернет-адрес в формате cidr/y, где cidr это адрес в формате IPv4 или IPv6, а /y количество бит в адресе (если этот параметр не указан, то используется 34 для IPv4, 128 для IPv6). Например, 192.168.0.1/24 или 2001:4f8:3:ba:2e0:81ff:fe22:d1f1/128. Занимает от 7 до 19 байт.
- macaddr: хранит MAC-адрес. Занимает 6 байт.
- macaddr8: хранит MAC-адрес в формате EUI-64. Занимает 8 байт.

#### Геометрические типы

- point: представляет точку на плоскости в формате (x,y). Занимает 16 байт.
- line: представляет линию неопределенной длины в формате {A,B,C}. Занимает 32 байта.
- **Iseg**: представляет отрезок в формате ((x1,y1),(x2,y2)). Занимает 32 байта.
- **box**: представляет прямоугольник в формате ((x1,y1),(x2,y2)). Занимает 32 байта.
- **path**: представляет набор содиненных точек. В формате ((x1,y1),...) путь является закрытым (первая и последняя точка соединяются линией) и фактически представляет многоугольник. В формате [(x1,y1),...] путь является открытым Занимает 16+16n байт.
- polygon: представляет многоугольник в формате ((x1,y1),...). Занимает 40+16n байт.
- circle: представляет окружность в формате <(x,y),r>. Занимает 24 байта.

#### Остальные типы данных

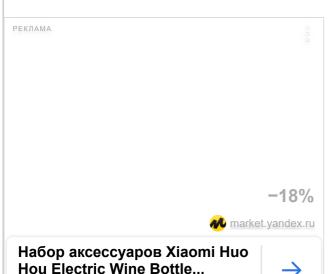
• **json**: хранит данные json в текстовом виде.

- **jsonb**: хранит данные json в бинарном формате.
- **uuid**: хранит универсальный уникальный идентификатор (UUID), например, а0eebc99-9c0b-4ef8-bb6d-6bb9bd380a11. Занимает 32 байта.
- **xml**: хранит даные в формате XML.



### Назад Содержание Вперед







### TAKKE HA METANIT.COM

2 326 ₽ 2 849 ₽

# Консольный клиент MySQL Shell

4 месяца назад • 1 коммент...

Консольный клиент MySQL Shell, подключение к серверу, создание базы ...

# Введение в корутины

3 месяца назад · 1 коммент...

Введение в корутины в языке программирования Kotlin, асинхронность, ...

## Оператор let. Привязка значений

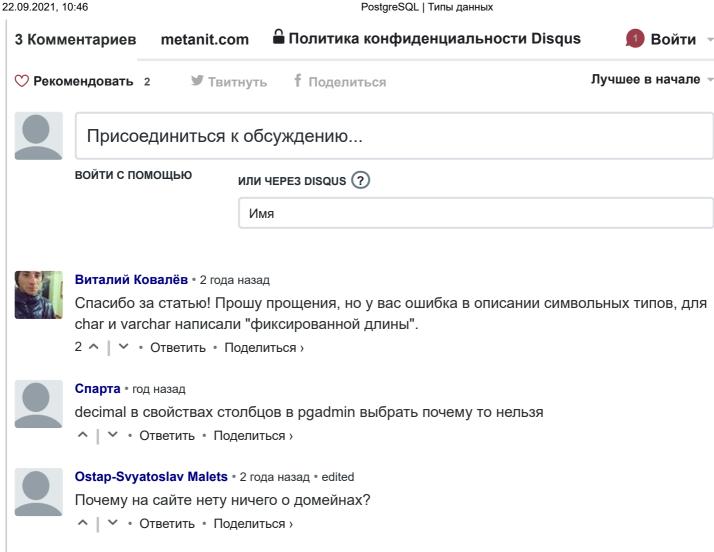
2 месяца назад · 1 коммент...

Оператор let. Привязка значений в языке программирования F#. ...

## Преобразо типов

2 месяца наза

Преобразовая програмить, оператор





### Помощь сайту WebMoney

- P378451176208
- · Z280152397659

### ЮМопеу/Яндекс-Деньги

• 410011174743222

## **PayPal**

• metanit22@mail.ru

## Перевод на карту

• Номер карты: 4048415020898850

# Вконтакте | Twitter | Канал сайта на youtube | Помощь сайту

Контакты для связи: metanit22@mail.ru

Copyright © metanit.com, 2012-2021. Все права защищены.