

Pesquisa sobre Tipos de Dados em Postgresql

Boolean Type

e

Bit String Types



PostgreSQL



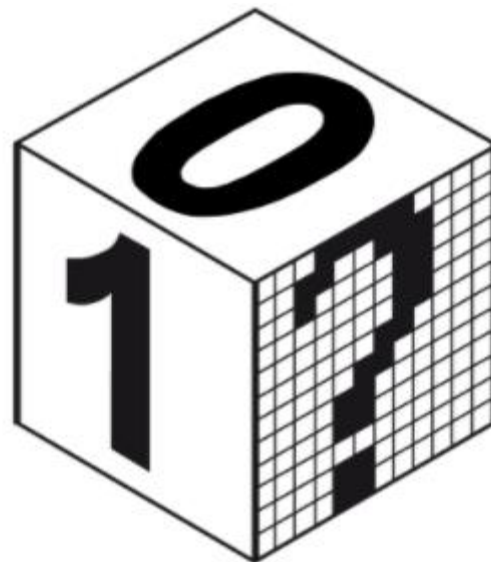
OS ZERESSEM S
PRO TEAM

Pesquisa sobre Tipos de Dados em Postgresql

Boolean Type:

A Lógica booleana é um tipo de dado primitivo que possui dois valores, podendo ser considerados como 0 ou 1, ou falso e verdadeiro. É assim chamada em homenagem ao grande matemático, George Boole.

Contudo, a forma como o tipo “**boolean**” é disponibilizado no programa Postgres pode chegar a possuir apenas um dos dois estados: “verdade” ou “falso”, podendo em alguns casos, exercer um terceiro estado, “desconhecido”, que por sua vez é representado pelo valor nulo do SQL.



Pesquisas sobre Tipos de Dados em Postgresql

Exemplos:

Valores literais válidos para o estado "**verdade**" são:

- **TRUE**
- 't'
- 'true'
- 'y'
- 'yes'
- '1'

Valores literais válidos para o estado "**falso**" são:

- **FALSE**
- 'f'
- 'false'
- 'n'
- 'no'
- '0'

Obs.: Geralmente, buscando estar em conformidade com o padrão SQL, há uma maior utilização das palavras chave **TRUE** e **FALSE** ao invés de 1 ou 0, por exemplo.



Pesquisas sobre Tipos de Dados em Postgresql

Exemplos práticos do tipo booleano no Postgresql utilizando as letras t e f:

```
CREATE TABLE teste1 (a boolean, b text);
INSERT INTO teste1 VALUES (TRUE, 'sic est');
INSERT INTO teste1 VALUES (FALSE, 'non est');
SELECT * FROM teste1;
```

a	b
t	sic est
f	non est

```
SELECT * FROM teste1 WHERE a;
```

a	b
t	sic est

```
\set null -
BEGIN;
CREATE TEMPORARY TABLE t (b BOOLEAN) ON COMMIT DROP;
INSERT INTO t VALUES(true);
INSERT INTO t VALUES(false);
INSERT INTO t VALUES(null);
SELECT * FROM t ORDER BY b;
COMMIT;
```

b
f
t
-

(3 linhas)

Obs.: Para o padrão SQL, o valor verdade é maior que o valor falso. Contudo, o PostgreSQL considera o valor nulo maior que estes dois.



Pesquisas sobre Tipos de Dados em Postgresql

Atualmente, o tipo booleano no PostgreSQL utiliza 1 byte para seu armazenamento:

Name	Storage Size	Description
<code>boolean</code>	1 byte	state of true or false

Outrossim, para o padrão SQL, o valor verdade é maior que o valor falso. Contudo, o PostgreSQL considera o valor nulo maior que estes dois, conforme mostrado neste exemplo.



Pesquisa sobre Tipos de Dados em Postgresql

Bit String Types:

As cadeias de bits são cadeias de zeros e uns, podendo ser usadas para armazenar ou visualizar máscaras de bits.

Atualmente, existem dois tipos de dado para bits no SQL, onde n é um número inteiro positivo:

- bit(n)
- bit varying(n)

O tipo de dado bit varying possui um comprimento variável até o máximo de n, cadeias mais longas são rejeitadas. Escrever bit sem o comprimento equivale a escrever bit(1), enquanto bit varying sem a especificação do comprimento significa comprimento ilimitado.



Pesquisa sobre Tipos de Dados em Postgresql

Se for feita uma conversão explícita do valor de uma cadeia de bits para `bit(n)`, os bits serão truncados ou completados à direita com zeros para ficar exatamente com `n` bits, sem ocasionar erro.

De forma semelhante, se for feita uma conversão explícita do valor de uma cadeia de bits para `bit varying(n)`, os bits serão truncados à direita se houver mais de `n` bits.

Obs.: Antes do PostgreSQL 7.2, os dados do tipo `bit` eram sempre truncados em silêncio ou completados à direita com zeros, com ou sem uma conversão explícita. Este comportamento foi modificado para ficar em conformidade com o padrão SQL.



Pesquisa sobre Tipos de Dados em Postgresql

Exemplo prático do tipo cadeia de bits no Postgresql:

```
CREATE TABLE teste (a BIT(3), b BIT VARYING(5));
INSERT INTO teste VALUES (B'101', B'00');
INSERT INTO teste VALUES (B'10', B'101');
ERRO: comprimento da cadeia de bits 2 não corresponde ao tipo bit(3)
INSERT INTO teste VALUES (B'10'::bit(3), B'101');
SELECT * FROM teste;
```

a	b
101	00
100	101

Um exemplo de Bit String Types é o uso do “DateTime”, que é usado para valores que contêm partes de data e hora, como por exemplo:

- 'YYYY-MM-DD hh:mm:ss'1000-01-01 00:00:00'9999-12-31 23:59:59'



Pesquisas sobre Tipos de Dados em Postgresql

Fontes:

[http://pgdocptbr.sourceforge.net/pg80/datatype-boolean.html#FTN.AEN6220;](http://pgdocptbr.sourceforge.net/pg80/datatype-boolean.html#FTN.AEN6220)
<https://www.postgresql.org/docs/9.6/datatype-boolean.html>



Grupo 01

- Caio Rodrigues
- Frederico Stilpen
- Gabriel Araújo
- João Paulo Bade
- Marcos Correa
- Volnei Neves



OS ZERESSEM  **S**
PRO TEAM