

TIPOS DE DATOS EN POSTGRESQL

Tipos de datos lógicos:

PostgreSQL incorpora el tipo lógico boolean, también llamado bool. Ocupa un byte de espacio de almacenamiento y puede almacenar los valores falso y verdadero.

Valor	Nombre
Falso	false, 'f', 'n', 'no', 0
Verdadero	true, 't', 'y', 'yes', 1

Tipos numéricos

PostgreSQL dispone de los tipos enteros smallint, int y bigint que se comportan como lo hacen los enteros en muchos lenguajes de programación.

Los numero con punto flotante real y double almacenan cantidades con decimales. Una característica de los números de punto flotante es que pierden exactitud conforme crecen o decrecen los valores.

Aunque esta pérdida de exactitud no suele tener importancia en la mayoría de las ocasiones, PostgreSQL incluye el tipo numeric, que permite almacenar cantidades muy grandes o muy pequeñas sin pérdida de información. Por supuesto esta pérdida tiene un coste, los valores de tipo numeric ocupan un espacio de almacenamiento considerablemente grande y las operaciones se ejecutan sobre ellos muy lentamente. Por lo tanto, no es aconsejable utilizar el tipo numeric si no se necesita una alta precisión o se prima la velocidad del procesamiento.

Nombre	Tamaño	Otros nombres	Comentario
smallint	2 bytes	int2	
int	4 bytes	int4, integer	

Nombre	Tamaño	Otros nombres	Comentario
bigint	8 bytes	int8	
numeric(p,e)	11 + (p/2)		'p' es la precisión, 'e' es la escala
real	4 bytes	float, float4	
double precision	8 bytes	float8	
serial			No es un tipo, es un entero auto-incrementable

La declaración serial es un caso especial, ya que no se trata de un nuevo tipo. Cuando se utiliza como nombre de tipo de una columna, está se tomará automáticamente valores consecutivos en cada nuevo registro.

Name	Aliases	Description
bigint	int8	signed eight-byte integer
bigserial	serial8	autoincrementing eight-byte integer
bit [(n)]		fixed-length bit string
bit varying [(n)]	varbit [(n)]	variable-length bit string
boolean	bool	logical Boolean (true/false)
box		rectangular box on a plane
bytea		binary data ("byte array")
character [(n)]	char [(n)]	fixed-length character string
character varying [(n)]	varchar [(n)]	variable-length character string
cidr		IPv4 or IPv6 network address
circle		circle on a plane
date		calendar date (year, month, day)
double precision	float8	double precision floating-point number (8 bytes)
inet		IPv4 or IPv6 host address
integer	int, int4	signed four-byte integer
interval [fields] [(p)]		time span
json		textual JSON data
jsonb		binary JSON data, decomposed
line		infinite line on a plane
lseg		line segment on a plane
macaddr		MAC (Media Access Control) address
macaddr8		MAC (Media Access Control) address (EUI-64 format)
money		currency amount
numeric [(p, s)]	decimal [(p, s)]	exact numeric of selectable precision
path		geometric path on a plane
pg_lsn		PostgreSQL Log Sequence Number
pg_snapshot		user-level transaction ID snapshot
point		geometric point on a plane
polygon		closed geometric path on a plane
real	float4	single precision floating-point number (4 bytes)
smallint	int2	signed two-byte integer
smallserial	serial2	autoincrementing two-byte integer
serial	serial4	autoincrementing four-byte integer
text		variable-length character string
time [(p)] [without time zone]		time of day (no time zone)
time [(p)] with time zone	timetz	time of day, including time zone
timestamp [(p)] [without time zone]		date and time (no time zone)
timestamp [(p)] with time zone	timestamptz	date and time, including time zone
tsquery		text search query
tsvector		text search document
txid_snapshot		user-level transaction ID snapshot (deprecated; see pg_snapshot)
uuid		universally unique identifier
xml		XML data

BIBLIOGRAFÍA

http://informatica.gonzalonazareno.org/plataforma/pluginfile.php/242/mod_resource/content/0/Apuntes/UOC_Postgres.pdf

<https://www.postgresql.org/docs/13/datatype.html>