



Universidad Nacional Autónoma de México.



Facultad de Ingeniería.

Bases de Datos.

Profesor: Ing. Fernando Arreola Franco.

Semestre 2025-2

Tarea 6.

Tipos de datos en Postgres.

Grupo: 01

Alumna:

Valencia Lerdo Dalia Jimena

Tipos de datos.

Numéricos. Los tipos numéricos consisten en enteros de dos, cuatro y ocho bytes, números de coma flotante de cuatro y ocho bytes, y decimales de precisión seleccionable. La Tabla 1 muestra los tipos disponibles.

Tabla 1. Tipo numérico.

Name	Storage Size	Description	Range
<code>smallint</code>	2 bytes	small-range integer	-32768 to +32767
<code>integer</code>	4 bytes	usual choice for integer	-2147483648 to +2147483647
<code>bigint</code>	8 bytes	large-range integer	-9223372036854775808 to 9223372036854775807
<code>decimal</code>	variable	user-specified precision, exact	no limit
<code>numeric</code>	variable	user-specified precision, exact	no limit
<code>real</code>	4 bytes	variable-precision, inexact	6 decimal digits precision
<code>double precision</code>	8 bytes	variable-precision, inexact	15 decimal digits precision
<code>serial</code>	4 bytes	autoincrementing integer	1 to 2147483647
<code>bigserial</code>	8 bytes	large autoincrementing integer	1 to 9223372036854775807

Enteros. Los tipos `smallint`, `integer` y `bigint` almacenan números enteros, es decir, sin componentes fraccionarios, de varios rangos.

Números de precisión arbitraria. El tipo numérico puede almacenar números con una precisión de hasta 1000 dígitos y realizar cálculos con exactitud. Se recomienda especialmente para almacenar cantidades monetarias y otras que requieren precisión.

Tipos de punto flotante. Los tipos de datos de precisión real y doble son tipos numéricos inexactos de precisión variable. En la práctica, estos tipos suelen ser implementaciones del estándar IEEE 754 para aritmética binaria de punto flotante (precisión simple y doble, respectivamente).

Tipos de serie. Los tipos de datos `serial` y `bigserial` no son tipos verdaderos, sino simplemente una conveniencia de notación para configurar columnas de identificadores únicos.

Carácter. SQL define dos tipos de caracteres principales: `character varying(n)` y `character(n)`, donde `n` es un entero positivo. Ambos tipos pueden almacenar cadenas de hasta `n` caracteres. Intentar almacenar una cadena más larga en una columna de estos tipos generará un error, a menos que los caracteres sobrantes sean espacios, en cuyo caso la cadena se truncará a la longitud máxima.

Tabla 2. Tipo carácter

Name	Description
<code>character varying(n)</code> , <code>varchar(n)</code>	variable-length with limit
<code>character(n)</code> , <code>char(n)</code>	fixed-length, blank padded
<code>text</code>	variable unlimited length

Fecha. PostgreSQL admite todos los tipos de fecha y hora de SQL , como se muestra en la Tabla 3. Estos tipos de datos almacenan datos de tiempo y duración.

Todos los tipos de datos de tiempo, excepto date los que permiten una precisión de nivel de microsegundos, y pueden aceptar un valor de provisión opcional p(por ejemplo, timetz [(p)]) para especificar la cantidad de dígitos fraccionarios a almacenar en el campo de segundos.

date, almacena valores con una precisión de un día.

Además de las fechas y horas tradicionales, date y timestamp puede aceptar algunos valores especiales (y autoexplicativos) como now, yesterday, tomorrow, etc. now se almacenarán como la hora actual; yesterday, today, y tomorrow se almacenan como 0:00 UTC en la fecha correspondiente.

Tabla 3. Tipo fecha

Name	Storage Size	Description	Low Value	High Value	Resolution
timestamp [(p)] [without time zone]	8 bytes	both date and time	4713 BC	5874897 AD	1 microsecond / 14 digits
timestamp [(p)] with time zone	8 bytes	both date and time, with time zone	4713 BC	5874897 AD	1 microsecond / 14 digits
interval [(p)]	12 bytes	time intervals	-178000000 years	178000000 years	1 microsecond / 14 digits
date	4 bytes	dates only	4713 BC	5874897 AD	1 day
time [(p)] [without time zone]	8 bytes	times of day only	00:00:00	24:00:00	1 microsecond / 14 digits
time [(p)] with time zone	12 bytes	times of day only, with time zone	00:00:00+1359	24:00:00-1359	1 microsecond / 14 digits

Geométricos. Los tipos de datos geométricos representan objetos espaciales bidimensionales. El tipo más fundamental, el punto, constituye la base de todos los demás tipos.

Tabla 4. Tipo geométrico.

Name	Storage Size	Representation	Description
point	16 bytes	Point on the plane	(x,y)
line	32 bytes	Infinite line (not fully implemented)	((x1,y1),(x2,y2))
lseg	32 bytes	Finite line segment	((x1,y1),(x2,y2))
box	32 bytes	Rectangular box	((x1,y1),(x2,y2))
path	16+16n bytes	Closed path (similar to polygon)	((x1,y1),...)
path	16+16n bytes	Open path	[(x1,y1),...]
polygon	40+16n bytes	Polygon (similar to closed path)	((x1,y1),...)
circle	24 bytes	Circle	<(x,y),r> (center and radius)

Puntos. Los puntos son el componente fundamental bidimensional de los tipos geométricos.

Segmentos de línea. Los segmentos de línea (lseg) se representan mediante pares de puntos.

Cajas. Las cajas se representan mediante pares de puntos que son vértices opuestos de la caja. Se pueden proporcionar dos esquinas opuestas cualesquiera en la entrada, pero los valores se reordenarán según sea necesario para almacenar las esquinas superior derecha e inferior izquierda, en ese orden.

Caminos. Las rutas se representan mediante listas de puntos conectados. Pueden ser abiertas, donde el primer y el último punto de la lista no se consideran conectados, o cerradas, donde el primer y el último punto sí se consideran conectados.

Polígonos. Los polígonos se representan mediante listas de puntos (los vértices del polígono). Probablemente deberían considerarse equivalentes a las trayectorias cerradas, pero se almacenan de forma diferente y cuentan con su propio conjunto de rutinas de soporte.

Círculos. Los círculos se representan mediante un punto central y un radio.

Booleanos.

Un tipo de dato booleano puede contener uno de tres valores posibles: verdadero, falso o nulo. Se utiliza la palabra clave boolean 'or' bool para declarar una columna con el tipo de dato booleano. Cuando inserta datos en una columna booleana, PostgreSQL los convierte en un valor booleano

1, yes, y, t, true los valores se convierten a true
0, no, false, f los valores se convierten a false.

Cuando selecciona datos de una columna booleana, PostgreSQL convierte los valores nuevamente, por ejemplo, t a verdadero, f a false y space a null.

Fuentes.

"Data Types". PostgreSQL Documentation. Accedido el 3 de abril de 2025. [En línea]. Disponible: <https://www.postgresql.org/docs/8.1/datatype.html> "PostgreSQL data types: what are they, and when to use each". CockroachDB | Distributed SQL for always-on customer experiences. Accedido el 6 de abril de 2025. [En línea]. Disponible: <https://www.cockroachlabs.com/blog/postgres-data-types/> "PostgreSQL Data Types - Numeric, Text, and More". Prisma's Data Guide. Accedido el 6 de abril de 2025. [En línea]. Disponible: <https://www.prisma.io/dataguide/postgresql/introduction-to-data-types> Neon. "Getting Started with PostgreSQL Data Types". Neon. Accedido el 6 de abril de 2025. [En línea]. Disponible: <https://neon.tech/postgresql/postgresql-tutorial/postgresql-data-types>