**Organización de datos en arreglos y conjuntos**

El arreglo juega un papel importante en PostgreSQL. Cada tipo de datos tiene su propia matriz compañera tipo, por ejemplo, integer tiene un integer[] tipo de arreglo, character tiene character[]el tipo de arreglo, etc. En el caso de definir su propio tipo de datos, PostgreSQL crea un tipo de arreglo correspondiente en el fondo para usted.

-PostgreSQL le permite definir una columna como un arreglo de cualquier tipo de datos válido, incluido el tipo incorporado, el tipo definido por el usuario o el tipo enumerado. PostgreSQL le permite actualizar cada elemento de una matriz o la matriz completa.

Las comparaciones de arreglos comparan el contenido del arreglo elemento por elemento, utilizando la función de comparación de árbol B predeterminada para el tipo de datos del elemento. En arreglos multidimensionales, los elementos se visitan en orden de fila principal (el último subíndice varía más rápidamente). Si el contenido de dos arreglos es igual pero la dimensionalidad es diferente, la primera diferencia en la información de dimensionalidad determina el orden de clasificación. (Este es un cambio de las versiones de PostgreSQL anteriores a 8.2: las versiones anteriores afirmarían que dos arreglos con el mismo contenido eran iguales, incluso si el número de dimensiones o rangos de subíndice fuera diferente).

En PostgreSQL puede crear matrices de cualquier tipo integrado, definido por el usuario o enumeración. De forma predeterminada, no hay límite para una matriz, pero puede especificarla.

Por defecto, PostgreSQL usa una convención de numeración basada en uno para los arreglos, es decir, un arreglo de n elementos comienza con el array[1] y termina con el array[n] .

Los arreglos se pueden usar para desnormalizar datos y evitar tablas de búsqueda. Una buena regla general para usarlos de esa manera es que usas principalmente la matriz como un todo, incluso si a veces puedes buscar elementos en el arreglo. El procesamiento más pesado será más complejo que una tabla de búsqueda.

-

CONJUNTOS

La lógica de estos tipos de estos tipos de operadores nace de la teoría de conjuntos. Permite unir 2 o más consultas en una sola. Cada consulta debe retornar la misma cantidad de columnas y del mismo tipo de dato. No importa si los nombres de las columnas son diferentes, siempre se muestran los nombres de la primera consulta. No importa la cantidad de registros retornados. Puede utilizar la cláusula ORDER BY o LIMIT, al final de la última consulta, esto afecta a todo el resultado.

UNION: Une distintas consultas, no muestra los registros duplicados.

UNION ALL: Une distintas consultas, si muestra los registros duplicados.

INTERSECT: Une distintas consultas, retorna únicamente los registros que son duplicados.

EXCEPT: Une distintas consultas, y solo muestra los registros de la primer tabla y no los de la segunda, cuando se habla de consultar 2 tablas.