PostgreSQL支持一维或多维数组， 数组的类型可以是数据库内建的类型(数值、字符)，也可以是用户自定义的类型、 枚举类型及组合类型。

**数组类型的定义**

通过在数组元素类型名后面附加方括号“[]”来定义数组：

CRETE TABLE t1(id int, col1 int[], col2 text[], col3 text[][]);

注意：如果在定义数组类型中填一个数组长度的数字， 这个数字是无效的， 不会限制数组的长度；定义时指定数组维度也是没有意义的， 数组的维度实际上也是根据实际插入的数据来确定的。也就是说， 下面两个声明的意义是相同的。

CRETE TABLE t1(id int, col1 int[], col2 text[], col3 text[][]);

CRETE TABLE t1(id int, col1 int[], col2 text[5], col3 text[]);

**数组值的输入**

insert into t1 values('{1,2,3}');--使用单引号+花括号，元素值之间用逗号分隔

insert into t1 values('{4,5,6}'),('{7,8,9}'); --一次输入多条值

insert into t1 values('{hello,hello,hello}');

insert into t1 values('{"world","world","world"}');

insert into t1 values('{"hello,world","world,hello"}');--字符串中有逗号使用双引号

insert into t1 values('{"I''am yewei","I''am weiye"}');--字符串中有单引号使用两个连接的单引号

insert into t1 values('{"{hello,world}","{world,hello}"}');--字符串中有括号{}把它们放到双引号中

insert into t1 values('{hello\"world,world\"hello}');--字符串中有双引号在双引号前加反斜扛

insert into t1 values('{null,hello}');--数组元素的值为NULL， 直接写上NULL（与大小写无关）

insert into t1 values('{"null",world}');--数组元素值为字符串"NULL"加上双引号

insert into t1 values('{{hello,world},{world,hello}}');--多维数组,各维度元素的个数必须相同， 否则会报错

insert into t1 values(array[1,2,3]);--使用array关键字

insert into t1 values(array[[1,2],[3,4]]);--多维数组，各维度元素的个数必须相同， 否则会报错

**访问数组**

select \* from t1;

select id from t1;

select id[1] from t1; --访问数组中的某个元素

select id[1:2] from t1; --数组切片， 如果id[]为{1,2,3}，切片[1:2]后为{1,2}

select id[1][1] from t1; --访问多维数组

**修改数组**

数组值可以整个被替换， 也可以只替换数组中的单个元素。

update t1 set col1='{7,8,9}' where id=1;--修改整个数组

update t1 set col1[1]=10 where id=1; --修改单个数组元素

update t1 set col1[1][1]=11 where id=1; --修改多维数组的元素

select \* from t1;

**数组的操作符**

|  |  |  |
| --- | --- | --- |
| 操作符 | 描述 | 例子 |
| @> | 包含 | ARRAY[1,4,3] @> ARRAY[3,1,3] → t |
| <@ | 被包含 | ARRAY[2,2,7] <@ ARRAY[1,7,4,2,6] → t |
| && | 重叠 | ARRAY[1,4,3] && ARRAY[2,1] → t |
| || | 连接两个数组 | ARRAY[1,2,3] || ARRAY[4,5,6,7] → {1,2,3,4,5,6,7}  ARRAY[1,2,3] || ARRAY[[4,5,6],[7,8,9]] → {{1,2,3},{4,5,6},{7,8,9}} |

**数组的操作函数**

|  |  |  |
| --- | --- | --- |
| 函数 | 描述 | 例子 |
| array\_append | 向数组的末端追加元素 | array\_append(ARRAY[1,2], 3) → {1,2,3} |
| array\_cat | 连接两个数组 | array\_cat(ARRAY[1,2,3], ARRAY[4,5]) → {1,2,3,4,5} |
| array\_length | 返回数组维度的长度 | array\_length(array[1,2,3], 1) → 3 |
| array\_position | 返回数组中指定值的下标 | array\_position(ARRAY['sun', 'mon', 'tue', 'wed', 'thu', 'fri', 'sat'], 'mon') → 2 |
| array\_prepend | 在数组的开头添加元素 | array\_prepend(1, ARRAY[2,3]) → {1,2,3} |
| array\_remove | 从数组中移除元素 | array\_remove(ARRAY[1,2,3,2], 2) → {1,3} |
| array\_replace | 替换数组中的元素 | array\_replace(ARRAY[1,2,5,4], 5, 3) → {1,2,3,4} |
| cardinality | 返回数组中元素的总数 | cardinality(ARRAY[[1,2],[3,4]]) → 4 |
| unnest | 将数组展开为多行 | unnest(ARRAY[1,2,3,4,5]) →  1  2  3  4  5 |

其他数组操作符和函数：http://www.postgres.cn/docs/14/functions-array.html

数组还有一个聚合函数array\_agg

insert into t1 select n from generate\_series(1,20);

select array\_agg(id) from t1;