PostgreSQL中可以如C语言中的结构体一样定义一个复合类型。

**复合类型的定义**

定义一个 person 的复合类型

CREATE TYPE person AS (

name text,

age int,

sex boolean

);

注意： AS关键字是必须的。

复合类型的使用：

CREATE TABLE t1(

id int,

info person

);

**复合类型的输入**

使用单引号加圆括号，在此格式中，可以在任何字段值周围加上双引号， 如果值本身包含逗号或者圆括弧， 则必须用双引号括起来。

CREATE TABLE t1(

id int,

info person

);

insert into t1 values( 1, '("zhangsan",24,TRUE)');

insert into t1 values( 2, '("lisi",,TRUE)'); --插入null值

insert into t1 values(3, '("",,TRUE)');--插入空字符串

insert into t1 values( 4, ROW('wangwu', 26, TRUE));--使用ROW表达式构造符合类型值

insert into t1 values(5, ('zhaoliu', 28, TRUE)); --省略ROW

**访问复合类型**

select info.name from author;--报错

select (info).name from author;--正常访问

select (t1.info).name from author;--正常访问

**修改复合类型**

UPDATE t1 SET info = ROW('zhangsansan', 26, TRUE) WHERE id =1;--修改整个字段

UPDATE t1 SET info = ('lisisi', 28, TRUE) WHERE id =2;--修改整个字段

UPDATE t1 SET info.name = 'lisi' WHERE id =2;--修改name子域

UPDATE t1 SET info.age = (info).age + 1 WHERE id =2;--修改age子域

UPDATE t1 SET (info).name = 'lisisi' WHERE id =2;--报错，不能加圆括号

UPDATE t1 SET info.age = info.age + 1 WHERE id =2;--报错，等号右边的表达式里引用同一个字段必须加上圆括弧

INSERT INTO t1 (id, info.name, info.age) VALUES(6, 'sunqi',29);--插入子域，其他用null填充