

## 子查询和连接

### MySQL 子 查 询

一 . 由比较运算符引发的子查询：

> , < , = , != ( < > ) , >= , <=

✓ 查询最新的商品

```
SELECT goods_id,goods_name FROM goods WHERE goods_id =  
(SELECT MAX(good_id) FROM goods);
```

当子查询返回多个值时，需要在比较运算符后加上关键字：ANY、SOME、ALL

当用到这些关键字后，子查询返回的值为：

	ANY	SOME	ALL
>、>=	最小值	最小值	最大值
<、<=	最大值	最大值	最小值
=	任意值	任意值	
<>、!=			任意值

✓ 取出商品种类为 3 的商品的价格

```
SELECT goods_price FROM goods WHERE good_type_id=3;
```

假设其返回值为 3 个：500、700、900

若使用如下子查询语句，则子查询返回的结果为 500，即查询大于商品种类为 3

中的任意一个商品的价格的商品。

```
SELECT goods_id,goods_name FROM goods WHERE goods_price > ANY  
(SELECT goods_price FROM goods WHERE good_type_id=3);
```

二 . 由 [NOT] IN 引发的子查询

= ANY 运算符与 IN 等效

!=ANY 或 <> 运算符与 NOT IN 等效

- ✓ 取出每个种类下最新的产品(goods\_id 唯一)

```
SELECT goods_type_id,goods_id,goods_name FROM goods WHERE goods_id IN  
<SELECT MAX(goods_id) FROM goods GROUP BY goods_type_id>;
```

### 三. EXISTS 型子查询 (把外层查询结果拿到内层,看内层的查询是否成立)

- ✓ 查询哪些种类下有商品,商品种类表 category,商品表 goods

```
SELECT goods_type_id,good_type_name FROM category WHERE EXISTS  
<SELECT * FROM goods WHERE goods.goods_type_id=category.goods_type_id>;
```

## 内连接, 左连接, 右连接

### ① 内连接: 显示左表及右表符合连接条件的记录

查询结果是左右连接的交集,

语法: SELECT n1,n2,n3 FROM tb1 INNER JOIN tb2 ON tb1.n1= tb2.n2;

### ② 左连接: 显示左表的全部记录及右表符合连接条件的记录

以左表为准,去右表找数据,如果没有匹配的数据,则以 NULL 补空位。

语法: SELECT n1,n2,n3 FROM tb1 LEFT JOIN tb2 ON tb1.n1= tb2.n2

### 这里 ON 后面的表达式,不一定为=,也可以>, <等算术、逻辑运算符,连接完

成后,可以当成一张新表来看待,运用 where 等查询 ###

- ✓ 取出价格最高的五个商品,并显示商品的分类名称

```
SELECT goods_id,goods_name,goods_type_name from goods LEFT JOIN  
category ON goods.goods_type_id = category. goods_type_id  
ORDER BY shop_price DESC LIMIT 5;
```

### ③ 右连接: 显示右表的全部记录及左表符合连接条件的记录

a LEFT JOIN b 等价于 b RIGHT JOIN a

语法: SELECT n1,n2,n3 FROM tb1 RIGHT JOIN tb2 ON tb1.n1= tb2.n2

## 自身连接

我们就教学视频中无限极分类表设计中的例子进行详解，帮助大家更好的理解自身连接。

下图为无限级分类表 tdb\_goods\_types 中的全部记录：

```
mysql> SELECT * FROM tdb_goods_types;
```

type_id	type_name	parent_id
1	家用电器	0
2	电脑、办公	0
3	大家电	1
4	生活电器	1
5	平板电视	3
6	空调	3
7	电风扇	4
8	饮水机	4
9	电脑整机	2
10	电脑配件	2
11	笔记本	9
12	超级本	9
13	游戏本	9
14	CPU	10
15	主机	10

- ✓ 查看所有的子类以及它父类（即上一级分类名字）

此时我们的想象中的子父表应该为：

父表：p		子表：s		
type_id	type_name	type_id	type_name	parent_id
1	家用电器	1	家用电器	0
2	电脑、办公	2	电脑、办公	0
3	大家电	3	大家电	1
4	生活电器	4	生活电器	1
5	平板电视	5	平板电视	3
6	空调	6	空调	3
7	电风扇	7	电风扇	4
8	饮水机	8	饮水机	4
9	电脑整机	9	电脑整机	2
10	电脑配件	10	电脑配件	2
11	笔记本	11	笔记本	9
12	超级本	12	超级本	9
13	游戏本	13	游戏本	9
14	CPU	14	CPU	10
15	主机	15	主机	10

查询语句为：

```
SELECT s.type_id,s.type_name,p.type_name FROM tdb_goods_types AS s
LEFT JOIN tdb_goods_types AS p ON s.parent_id=p.type_id;
```

我们需要查看的列为图中黄色标注处，连接的条件是子表中的 parent\_id = 父表中的 type\_id

查找结果为：

type_id	type_name	type_name
1	家用电器	NULL
2	电脑、办公	NULL
3	大家电	家用电器
4	生活电器	家用电器
5	平板电视	大家电
6	空调	大家电
7	电风扇	生活电器
8	饮水机	生活电器
9	电脑整机	电脑、办公
10	电脑配件	电脑、办公
11	笔记本	电脑整机
12	超级本	电脑整机
13	游戏本	电脑整机
14	CPU	电脑配件
15	主机	电脑配件

✓ 查看所有父类以及父类下的子类（即下一级分类的名字）

此时我们的想象中的子父表应该为：

父表：p		子表：s		
type_id	type_name	type_id	type_name	parent_id
1	家用电器	1	家用电器	0
2	电脑、办公	2	电脑、办公	0
3	大家电	3	大家电	1
4	生活电器	4	生活电器	1
5	平板电视	5	平板电视	3
6	空调	6	空调	3
7	电风扇	7	电风扇	4
8	饮水机	8	饮水机	4
9	电脑整机	9	电脑整机	2
10	电脑配件	10	电脑配件	2
11	笔记本	11	笔记本	9
12	超级本	12	超级本	9
13	游戏本	13	游戏本	9
14	CPU	14	CPU	10
15	主机	15	主机	10

查询语句为：

```
SELECT p.type_id,p.type_name,s.type_name FROM tdb_goods_types AS p
LEFT JOIN tdb_goods_types AS s ON s.parent_id=p.type_id;
```

我们需要查看的列为图中黄色标注处，连接的条件是子表中的 parent\_id = 父表中的 type\_name

查找结果为：

type_id	type_name	type_name
1	家用电器	大家电
1	家用电器	生活电器
2	电脑、办公	电脑整机
2	电脑、办公	电脑配件
3	大家电	平板电视
3	大家电	空调
4	生活电器	电风扇
4	生活电器	饮水机
5	平板电视	NULL
6	空调	NULL
7	电风扇	NULL
8	饮水机	NULL
9	电脑整机	笔记本
9	电脑整机	超级本
9	电脑整机	游戏本
10	电脑配件	CPU
10	电脑配件	主机
11	笔记本	NULL
12	超级本	NULL
13	游戏本	NULL
14	CPU	NULL
15	主机	NULL

慕课网  
imooc.com