连接查询

1. 连接查询的基本概念

将两个以上表,连接起来,当作一个数据源,并从中获取想要的数据 连接:将每一个表每一行数据两两连接起来,每次对接都是连接结果的一行数据

连接结果:

		t *from			
id	c1	c2	id2	e1	e2
	c11			e11	e21
1	c11	c21	2	e12	e22
2	c12	c22	1	e11	e21
2	c12	c22	2	e12	e22
3	c13	c23	1	e11	e21
3	c13	c23	2	e12	e22
+	+	+	+		+

连接前:

无条件连接:

- (1) select *from join1, join2;
- (2) select *from join1 join join2;
- (3) select *from join1 cross join join2; 以上三种均为交叉连接,但一般这没什么实际意义

2. 连接基本形式

代码层面,连接基本形式为:

表1 [连接形式] join 表2 [on连接条件]

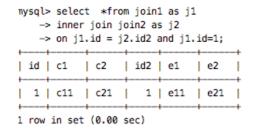
3. 连接的分类

(1) 交叉连接

没有条件的连接,将所有的数据行连接起来,又叫迪卡尔积表1(m1个字段,n1行) 表2(m2个字段,n2行) 连接结果为:虚拟表3(m1+m2个字段,n1*n2行);

(2) 内连接

有条件的连接,将所有的数据行连接起来,然后进行一定的条件过滤形式: select *from 表1 [inner] join 表2 on 连接条件 on条件看似外键才可以做条件对象,实际非外键同样可以



(3) 左外连接

实际上是将两表的内连接结果,加上左边表的不符合内连接条件的那些项

```
mysql> select *from join1 as j1
-> left outer join join2 as j2
-> on j1.id=j2.id2;

| id | c1 | c2 | id2 | e1 | e2

| 1 | c11 | c21 | 1 | e11 | e21
| 2 | c12 | c22 | 2 | e12 | e22
```

| 3 | c13 | c23 | NULL | NULL | NULL

3 rows in set (0.06 sec)

(4) 右外连接

实际上是将两表的内连接结果,加上右边表的不符合内连接条件的那些项mysql> select *from join2 as j2 right outer join join1 as j1 on j1.id=j2.id2;

id2	e1	e2	id	c1	c2
1 2	e11 e12 NULL	e21 e22 NULL	1 2 3	c11 c12 c13	c21 c22 c23

3 rows in set (0.00 sec)

(5) 全外连接

mysql好像不支持这个,不过意思和左右外连接类似

4. 示例

pro_id	pro_name	protype	e_id	price	pinpai	chandi
1			1	1999	康佳	深圳
2	索尼(SONY) 4G手机(黑色)		2	3238	索尼	深圳
3	海信(Hisense)55英寸智能电视		1	4199	海信	青岛
4	联想(Lenovo) 14英寸笔记本电脑		3	5499	联想	北京
5	索尼(SONY) 13.3英寸触控超极本		3	11499	索尼	天津
6	索尼(SONY) 60英寸全高清液晶电视		1	6999	索尼	北京
7	联想(Lenovo) 14英寸笔记本电脑	ĺ	3	2999	联想	北京
8	联想(Lenovo) 双卡双待3g手机		2	988	联想	北京
9	惠普(HP) 黑白激光打印机	ĺ	3	1169	惠普	天津

nysqt> select *Trom product_type;

	++
protype_id	protype_name
1	家用电器
2	手机数码
3	电脑办公
4	图书音像
5	家居家具
6	服装配饰
7	个护化妆
	运动户外
9	汽车用品
10	食品酒水
11	营养保健
· 	

(1) 找出索尼手机所属产品类别

nysql> select p.pro_name as '产品名称', t.protype_name as '产品类别'

- -> from product as p
- -> join product_type as t
- -> on p.protype_id = t.protype_id
- -> where p.pro_name like '%索尼%' and p.pro_name like '%手机%';

产品名称	一产品类别
索尼(SONY) 4G手机(黑色)	手机数码

(2) 找出所有手机数码的产品

nysql> select *from product as p
-> join product_type as t
-> on p.protype_id = t.protype_id
-> where t.protype_name='手机数码';

pro_id pro_name	protype_id	price	pinpai	chandi	protype_id protype_name	İ
2 索尼(SONY) 4G手机(黑色) 8 联想(Lenovo) 双卡双待3g手机	2			- 深圳 北京	- 2 手机数码 2 手机数码	
+	+	+	+	+	+	-+