

Mysql高级查询

第五章

课程内容

- 1、等值连接与非等值连接
- 2、自连接
- 3、外连接
- 4、子查询
- 5、派生表
- 6、联合查询

课程目标

- 1、熟练掌握多表在不同条件下的查询操作
- 2、熟练掌握内外连接
- 3、了解派生表
- 4、了解联合查询

主外键关系

- 当主表中没有对应的记录时，不能将记录添加到子表
——成绩表中不能出现在学员信息表中不存在的学号；
- 不能更改主表中的值而导致子表中的记录孤立
——把学员信息表中的学号改变了，学员成绩表中的学号也应当随之改变；
- 子表存在与主表对应的记录，不能从主表中删除该行
——不能把有成绩的学员删除了
- 删除主表前，先删子表
——先删学员成绩表、后删除学员信息表

连接查询

- 连接查询：同时涉及多个表的查询
- 连接条件或连接谓词：用来连接两个表的条件
- 一般格式：

[表名1.]<列名1> <比较运算符> [表名2.]<列名2>
[表名1.]<列名1> BETWEEN [表名2.]<列名2> AND [表名2.]<列名3>
- 连接条件中的各连接字段类型必须是可比的，但名字不必相同。

常见连接查询

- 等值与非等值连接查询
- 自身连接查询
- 外连接
- 复合条件连接

1.1等值连接

- 连接运算符为： =
- 例： 查询每个学生及其选修课程的情况

```
SELECT st.*,sc.* FROM st,sc  
WHERE st.sno = sc.sno;
```

1.2非等值连接查询

- 员工信息表enginfo

eid	ename	sal	age
1	jojo	1200	20
2	lily	2000	22
3	Mark	3000	24
4	Jack	8000	27
5	Tom	4500	30

- 工资等级表salgrade

id	local	maxline
1	700	2000
2	2001	4000
3	4001	6000
4	6001	8000

```
select e.ename as '姓名',s.id as '等级' from enginfo  
e,salgrade s where e.sal BETWEEN s.local and s.maxline;
```

姓名	等级
jojo	1
lily	1
Mark	2
Jack	4
Tom	3

1.3 自身连接

- 自身连接：一个表与其自己进行连接
- 需要给表起别名以示区别
- 由于所有属性名都是同名属性，因此必须使用别名前缀
- 例如：查询每门课程的先修课名称

```
SELECT fir.cno,fir.cname,fir.cpno,sec.cname  
FROM course fir, course sec  
WHERE fir.cpno = sec.cno
```

Course表设计

Cno (课程号)	Cname (课程名称)	Cpno (先修课程号)
1	数据库	5
2	高等数学	
3	信息系统	1
4	操作系统	6
5	数据结构	7
6	数据处理	
7	C语言	6

问题：查询每门课程的间接先修课名称（先修课的先修课）

2.外连接

- 外连接可以是左向外联接、右向外联接或完整外部联接
- 在 FROM 子句中指定外联接时，可以由下列几组关键字中的一组指定：
 - LEFT JOIN 或 LEFT OUTER JOIN
 - RIGHT JOIN 或 RIGHT OUTER JOIN
 - FULL JOIN 或 FULL OUTER JOIN

2.1左连接

- 左向外联接的结果集包括 LEFT OUTER子句中指定的左表的所有行，而不仅仅是联接列所匹配的行。
- 如果左表的某行在右表中没有匹配行，则在相关联的结果集行中右表的所有选择列表列均为空值。
- 例：

```
SELECT s.sno, sname, sex, age, dept, cno, grade  
FROM student s LEFT OUT JOIN sc ON s.sno=sc.sno
```

2.2右连接

- 右向外联接是左向外联接的反向联接。将返回右表的所有行。如果右表的某行在左表中没有匹配行，则将为左表返回空值。

2.3外连接

- 完整外部联接返回左表和右表中的所有行。当某行在另一个表中没有匹配行时，则另一个表的选择列表列包含空值。如果表之间有匹配行，则整个结果集行包含基表的数据值。

3.子查询

- 子查询允许把一个查询嵌套在另一个查询当中。
- 子查询可以包含普通select可以包括的任何子句，比如：distinct、group by、order by、limit、join和union等；但是对应的外部查询必须是以下语句之一：select、insert、update、delete、set或者do。

子查询的分类

- 1. 标量子查询：
 - 返回单一值的标量，最简单的形式。
- 2. 列子查询：
 - 返回的结果集是 N 行一列。
- 3. 行子查询：
 - 返回的结果集是一行 N 列。
- 4. 表子查询：
 - 返回的结果集是 N 行 N 列
- 可以使用的操作符： = > < >= <= <> ANY IN
SOME ALL EXISTS

3.1 标量子查询

- 是指子查询返回的是单一值的标量，如一个数字或一个字符串，也是子查询中最简单的返回形式。可以使用 `= > < >= <= <>` 这些操作符对子查询的标量结果进行比较，通常子查询的位置在比较式的右侧

```
SELECT * FROM article WHERE uid = (SELECT uid FROM user  
WHERE status=1 ORDER BY uid DESC LIMIT 1)
```

```
SELECT * FROM t1 WHERE column1 = (SELECT MAX(column2)  
FROM t2)
```

```
SELECT * FROM article AS t WHERE 2 = (SELECT COUNT(*) FROM  
article WHERE article.uid = t.uid)
```

3.2列子查询

- 指子查询返回的结果集是 N 行一列，该结果通常来自对表的某个字段查询返回。
- 可以使用 IN、ANY和 ALL 操作符

```
SELECT * FROM article WHERE uid IN(SELECT uid  
FROM user WHERE status=1)
```

```
SELECT s1 FROM table1 WHERE s1 > ANY (SELECT  
s2 FROM table2)
```

```
SELECT s1 FROM table1 WHERE s1 > ALL (SELECT s2  
FROM table2)
```

3.3行子查询

- 指子查询返回的结果集是一行 N 列，该子查询的结果通常是对表的某行数据进行查询而返回的结果集。

```
SELECT * FROM article WHERE (title,content,uid) =  
(SELECT title, content,uid FROM blog WHERE bid=2)
```

3.4 表子查询

- 指子查询返回的结果集是 N 行 N 列的一个表数据。

```
SELECT * FROM article WHERE (title, content, uid)  
IN (SELECT title, content, uid FROM blog)
```

4.EXIST谓词

- EXISTS是一个非常牛叉的谓词，它允许数据库高效地检查指定查询是否产生某些行。

```
select * from t1 where city='beijing' and exists  
(select * from t2 where t1.cid=t2.cid);
```

5. 派生表

- 在子查询返回的值中，也可能返回一个表，如果将子查询返回的虚拟表再次作为FROM子句的输入时，这就子查询的虚拟表就成为了一个派生表。

FROM (subquery expression) AS derived_table_alias

派生表使用

- 派生表一般与外连接，分组统计一起使用

```
SELECT t1.name,t2.sex,a.city,a.age  
FROM t1,(SELECT city, MAX(age) FROM t2  
GROUP BY city) a  
WHERE t1.age=t2.age;
```

6.联合查询

- 使用UNION或UNION ALL关键字

```
SELECT cname,sex FROM users  
UNION  
SELECT name,sex FROM teacher;
```

相同结果被
筛选掉了

```
SELECT cname,sex FROM users  
UNION ALL  
SELECT name,sex FROM teacher;
```

有ALL，保留
相同项