

## 03.多表查询

---

### 一、单元概述

通过本章的学习能够了解MySQL数据库中多表查询的含义，掌握多表查询的基本写法，掌握特殊的多表查询的写法，掌握内连接和外连接的区别

### 二、教学重点与难点

重点：

- 掌握多表查询的基本写法
- 掌握内连接和外连接的区别

难点：

- 外连接的用法和写法
- 特殊的多表查询的写法：自连接、非等值连接等

## 3.1 多表查询

### 3.1.1 什么是多表查询

- 从多个表中获取数据

- 思考如下问题？

- 写一条查询语句，查询员工姓名、部门名称、工作地点？

EMP				DEPT		
EMPNO	ENAME	...	DEPTNO	DEPTNO	DNAME	LOC
7839	KING	...	10	10	ACCOUNTING	NEW YORK
7698	BLAKE	...	30	20	RESEARCH	DALLAS
...				30	SALES	CHICAGO
7934	MILLER	...	10	40	OPERATIONS	BOSTON

EMPNO	DEPTNO	LOC
7839	10	NEW YORK
7698	30	CHICAGO
7782	10	NEW YORK
7566	20	DALLAS
7654	30	CHICAGO
7499	30	CHICAGO
...		
14 rows selected.		

### 3.1.2 什么是连接

- 连接是在多个表之间通过一定的连接条件，使表之间发生关联，进而能从多个表之间获取数据。
- 语法为：

```
SELECT    table1.column, table2.column
FROM      table1, table2
WHERE     table1.column1 = table2.column2;
```

- 在 WHERE子句中书写连接条件。

- 如果在多个表中出现相同的列名，则需要使用表名作为来自该表的列名的前缀。
- N个表相连时，至少需要N - 1个连接条件

### 3.1.3 多表连接类型

- 按连接条件分：
  - ○ 等值连接
  - ○ 非等值连接
- 按其他连接方法分
  - ○ 外连接
  - ○ 内连接

### 3.1.4 多表连接写法

- 多表连接包含多种写法，我们主要介绍：
  - ○ 基本写法：绝大多数符合SQL标准，其它关系型数据也适用。
  - ○ ANNSI 99写法：ANSI标准提供的写法，所有关系型数据库必须支持。

## 3.2 笛卡尔积

### 3.2.1 笛卡尔积

- 第一个表中的所有行和第二个表中的所有行都发生连接。

### 3.2.2 笛卡尔积产生情况

- 笛卡尔积在下列情况产生：
  - 连接条件被省略
  - 连接条件是无效的
- 为了避免笛卡尔积的产生，通常需要在WHERE子句中包含一个有效的连接条件。

### 3.2.3 笛卡尔积写法

- 笛卡尔积的写法

```
SELECT      emp.empno,      emp.ename,
emp.deptno,      dept.deptno, dept.loc
FROM        emp, dept;
```

EMP (14 行)

EMPNO	ENAME	...	DEPTNO
7839	KING	...	10
7698	BLAKE	...	30
...			
7934	MILLER	...	10

DEPT (4 行)

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

“笛卡尔积：  
14\*4=56行”

ENAME	DNAME
KING	ACCOUNTING
BLAKE	ACCOUNTING
...	
KING	RESEARCH
BLAKE	RESEARCH
...	
56 rows selected.	

# 3.3 等值连接

## 3.3.1 什么是等值连接

- 查询所有员工编号，姓名，部门编号，工作地点

EMP

EMPNO	ENAME	DEPTNO
7839	KING	10
7698	BLAKE	30
7782	CLARK	10
7566	JONES	20
7654	MARTIN	30
7499	ALLEN	30
7844	TURNER	30
7900	JAMES	30
7521	WARD	30
7902	FORD	20
7369	SMITH	20
...		
14 rows selected.		

DEPT

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
30	SALES	CHICAGO
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
30	SALES	CHICAGO
30	SALES	CHICAGO
30	SALES	CHICAGO
30	SALES	CHICAGO
20	RESEARCH	DALLAS
20	RESEARCH	DALLAS
...		
14 rows selected.		

```
SELECT      emp.empno,      emp.ename,
emp.deptno,      dept.deptno, dept.loc
FROM        emp, dept
WHERE       emp.deptno=dept.deptno;
```

	EMPNO	ENAME	DEPTNO	DEPTNO	LOC
7839	KING		10	10	NEW YORK
7698	BLAKE	30	30	CHICAGO	
7782	CLARK	10	10	NEW YORK	
7566	JONES	20	20	DALLAS	
...					

- 只想查询工作地点在NEW YORK的员工编号，姓名，部门编号，工作地点

EMPNO	ENAME	DEPTNO	DEPTNO	DNAME	LOC
7839	KING	10	10	ACCOUNTING	NEW YORK
7698	BLAKE	30	30	SALES	CHICAGO
7782	CLARK	10	10	ACCOUNTING	NEW YORK
7566	JONES	20	20	RESEARCH	DALLAS
7654	MARTIN	30	30	SALES	CHICAGO
7499	ALLEN	30	30	SALES	CHICAGO
7844	TURNER	30	30	SALES	CHICAGO
7900	JAMES	30	30	SALES	CHICAGO
7521	WARD	30	30	SALES	CHICAGO
7902	FORD	20	20	RESEARCH	DALLAS
7369	SMITH	20	20	RESEARCH	DALLAS
...			...		
14 rows selected.			14 rows selected.		

### 3.3.2 使用AND运算符增加其它查询条件

```

SELECT      emp.empno,    emp.ename,
emp.deptno,    dept.deptno, dept.loc
FROM        emp, dept
WHERE       emp.deptno=dept.deptno and loc=
'NEW YORK';

```

EMPNO	ENAME	DEPTNO	DEPTNO	LOC
7839	KING	10	10	NEW YORK
7782	CLARK	10	10	NEW YORK
...				

## 3.4 限制歧义列名

- 在用到多个表时可以使用表名作前缀来限定列；
- 通过使用表前缀可以提高性能；
- 通过使用列的别名可以区分来自不同表但是名字相同的列；

## 3.5 使用表的别名

- 通过使用表的别名来简化查询语句

```
SELECT emp.empno, emp.ename, emp.deptno,  
dept.deptno, dept.loc  
FROM emp, dept  
WHERE emp.deptno=dept.deptno;
```

```
SELECT e.empno, e.ename, e.deptno,  
d.deptno, d.loc  
FROM emp e, dept d  
WHERE e.deptno= d.deptno;
```

### 练习

1. 写一个查询，显示所有员工姓名，部门编号，部门名称。
2. 写一个查询，显示所有工作在CHICAGO并且奖金不为空的员工姓名，工作地点，奖金
3. 写一个查询，显示所有姓名中含有A字符的员工姓名，工作地点。

## 3.6 非等值连接

EMP

EMPNO	ENAME	SAL
7839	KING	5000
7698	BLAKE	2850
7782	CLARK	2450
7566	JONES	2975
7654	MARTIN	1250
7499	ALLEN	1600
7844	TURNER	1500
7900	JAMES	950
...		
14 rows selected.		

SALGRADE

GRADE	LOSAL	HISAL
1	700	1200
2	1201	1400
3	1401	2000
4	2001	3000
5	3001	9999

“EMP表中的薪水是在  
SALGRADE 表所规定的  
最低和最高范围内。”

- 查询每个员工的姓名，工资，工资等级

```
SELECT      e.ename, e.sal, s.grade
FROM        emp e,   salgrade s
WHERE       e.sal
BETWEEN     s.losal AND s.hisal;
```

ENAME	SAL	GRADE
JAMES	950	1
SMITH	800	1
ADAMS	1100	1
...		

## 3.7 多于两个表的连接

- 约定：1个客户可以有多个订单，1个订单可以包含多个商品。



CUSTOMER (客户表)		ORD (订单表)		ITEM (产品表)	
NAME	CUSTID	CUSTID	ORDID	ORDID	ITEMID
-----	-----	-----	-----	-----	-----
JOCKSPORTS	100	101	610	610	3
TKB SPORT SHOP	101	102	611	611	1
VOLLYRITE	102	104	612	612	1
JUST TENNIS	103	106	601	601	1
K+T SPORTS	105	102	602	602	1
SHAPE UP	106	106	...	...	...
WOMENS SPORTS	107	106	...	...	...
...	...	...	...	...	...
9 rows selected.		21 rows selected.		64 rows selected.	

- 查询每个顾客都订购了哪些商品？

```
SELECT c.name,o.itemid
FROM   customer c, order o,item i
WHERE  c.custid = o.custid and o.ordid =
i.ordid;
```

- 多个表连接和两个表连接一样，在构造SQL语句时，需要多考虑一个表之间的关联条件。

## 3.8 多表连接的写法分析

1. 分析要查询的列都来自于哪些表，构成FROM子句;
2. 分析这些表之间的关联关系，如果表之间没有直接关联关系，而是通过另一个中间表关联，则也要在FROM子句中补充中间关联表;

3. 接下来在WHERE子句中补充表之间的关联关系，通常N个表，至少要有N-1个关联关系；
4. 分析是否还有其它限制条件，补充到WHERE子句的表关联关系之后，作为限制条件；
5. 根据用户想要显示的信息，补充SELECT子句。
6. 分析是否有排序要求，如果排序要求中还涉及到其它表，则也要进行第2步补充排序字段所在的表，并且添加表之间的关联关系；


## 练习

1. 查询每个员工的编号，姓名，工资，工资等级，所在城市，按照工资等级进行升序排序。

## 3.9 自连接

- 思考：查询每个员工的姓名和直接上级姓名？
- 自身连接，也叫自连接，是一个表通过某种条件和本身进行连接的一种方式，就如同多个表连接一样。

EMP (WORKER)				EMP (MANAGER)	
EMPNO	ENAME	MGR		EMPNO	ENAME
-----	-----	-----		-----	-----
7839	KING			7839	KING
7698	BLAKE	7839		7839	KING
7782	CLARK	7839		7839	KING
7566	JONES	7839		7839	KING
7654	MARTIN	7698		7698	BLAKE
7499	ALLEN	7698		7698	BLAKE



“WORKER表中的 MGR 等于 MANAGER表中的EMPNO”

```
SELECT worker.ename 'WNAME',manager.ename
' LNAME'
FROM      emp worker, emp manager
WHERE      worker.mgr = manager.empno;
```

WNAME	LNAME
SMITH	FORD
ALLEN	BLACK
WARD	BLACK
...	

## 练习

1. 查询所有工作在NEW YORK和CHICAGO的员工姓名，员工编号，以及他们的经理姓名，经理编号。

## 3.10 ANSI SQL：标准的连接语法

### 3.10.1 ANSI SQL：1999标准的连接语法

- 除了上述自己的连接语法外，同时支持美国国家标准协会 (ANSI) 的SQL：1999标准的连接语法。

```
SELECT      table1.column, table2.column
FROM        table1
[JOIN table2
    ON(table1.column_name =
table2.column_name)] |
[LEFT | RIGHT OUTER JOIN table2
    ON (table1.column_name =
table2.column_name)];
```

### 3.10.2 外部连接

EMP			DEPT	
ENAME	DEPTNO		DEPTNO	DNAME
-----	-----		-----	-----
KING	10		10	ACCOUNTING
BLAKE	30		20	RESEARCH
CLARK	10		30	SALES
JONES	20		40	OPERATIONS
...				



查询没有雇员工作的部门

- 在多表连接时，可以使用外部连接来查看哪些行,按照连接条件没有被匹配上。
- ○ 左外连接以FROM子句中的左边表为基表，该表所有行数据按照连接条件无论是否与右边表能匹配上，都会被显示出来。
- ○ 右外连接以FROM子句中的右边表为基表，该表所有行数据按照连接条件无论是否与左边表能匹配上，都会被显示出来

### 3.10.3 左外连接写法

- 查询所有雇员姓名，部门编号，部门名称，包括没有部门的员工也要显示出来

```
SELECT      e.ename,e.deptno,d.loc
FROM        emp e
LEFT OUTER JOIN dept d
ON          (e.deptno = d.deptno);
```

ENAME	DEPTNO	LOC
MILLER	10	NEW YORK
KING	10	NEW YORK
CLARK	10	NEW YORK
FORD	20	DALLAS
...		

### 3.10.4 右外连接写法

- 查询所有雇员姓名，部门编号，部门名称，包括没有员工的部门也要显示出来

```
SELECT      e.ename,e.deptno,d.loc
FROM        emp e
RIGHT OUTER JOIN dept d
ON          (e.deptno = d.deptno);
```

ENAME	DEPTNO	LOC
SMITH	20	DALLAS
ALLEN	30	CHICAGO
JONES	20	DALLAS
MARTIN	30	CHICAGO
....		

**练习** 使用SQL-99写法，完成如下练习

1. 创建一个员工表和部门表的交叉连接。
2. 使用自然连接，显示入职日期在80年5月1日之后的员工姓名，部门名称，入职日期
3. 使用左连接，查询每个员工的姓名，经理姓名，没有经理的King也要显示出来。
4. 使用右连接，查询每个员工的姓名，经理姓名，没有经理的King也要显示出来。

## 3.11 本章小结

- 等值连接

- 不等值连接
- 自身连接
- 外连接

## 3.12 课后作业

1. 显示员工SMITH的姓名，部门名称，直接上级名称
2. 显示员工姓名，部门名称，工资，工资级别，要求工资级别大于4级。
3. 显示员工KING和FORD管理的员工姓名及其经理姓名。
4. 显示员工姓名，参加工作时间，经理名，参加工作时间，要求参加时间比经理早。