



## 3.3.2 连接查询

☞ 连接查询——将两个(以上)表连接进行查询

- \* 同时涉及多个表的查询
- \* 连接查询的意义等价于关系代数中的 $\theta$ 连接、等值连接和自然连接。

### 一、等值与非等值连接查询

[例33] 查询每个学生及其选修课程的情况。

```
SELECT Student.*, SC.*  
FROM Student, SC  
WHERE Student.Sno = SC.Sno; //等值连接
```

或

```
SELECT Student.Sno, Sname, Ssex, Sage, Sdept, Cno, Grade  
FROM Student, SC  
WHERE Student.Sno = SC.Sno; //自然连接
```

注：*WHERE*子句中参与比较的两个属性名若在另一表中存在，则要加所属表名作前缀以区别，无同名则可省前缀。



## 3.3.2 连接查询

### \* 连接查询的执行过程

- (1) 首先在表1中找到第一个元组，然后从头开始扫描表2，逐一查找满足连接条件的元组，找到后就将表1中的第一个元组与该元组拼接起来，形成结果表中一个元组。
- (2) 表2全部查找完后，再找表1中第二个元组，然后再从头开始扫描表2，逐一查找满足连接条件的元组，找到后就将表1中的第二个元组与该元组拼接起来，形成结果表中一个元组。
- (3) 重复上述操作，直到表1中的全部元组都处理完毕

### \* 提高查询效率的方法：

- (1) 对表2按连接字段建立索引
- (2) 对表1中的每个元组，依次根据其连接字段值查询表2的索引，从中找到满足条件的元组，找到后就将表1中的第一个元组与该元组拼接起来，形成结果表中一个元组



## 3.3.2 连接查询

### 二、自身连接——一个表与其自己进行连接

[例34] 查询每一门课的间接先修课（即先修课的先修课）。

Cno	Cname	Cpno	Ccredit	Cno	Cname	Cpno	Ccredit
1	数据库	5	4	5	数据结构	7	4
2	数学		2				
3	信息系统	1	4	1	数据库	5	4
4	操作系统	6	3	6	数据处理		2
5	数据结构	7	4	7	PASCAL语言	6	4
6	数据处理		2				
7	PASCAL语言	6	4	6	数据处理		2

关系代数表达式： $\pi_{1,7}(\sigma_{3=5}(C \times C))$

元组演算表达式：

$$\{ t \mid \exists u \exists v (C(u) \wedge C(v) \wedge u[3]=v[1] \wedge t[1]=u[1] \wedge t[2]=v[3]) \}$$



## 3.3.2 连接查询

### 二、自身连接——一个表与其自己进行连接

[例34] 查询每一门课的间接先修课（即先修课的先修课）。

```
SELECT FIRST.Cno, SECOND.Cpno
FROM Course FIRST, Course SECOND
WHERE FIRST.Cpno = SECOND.Cno ;
```

注：

- (1) 一个表与其自己进行连接，称为表的**自身连接**
- (2) 需要给表起别名以示区别
- (3) 由于所有属性名都是同名属性，因此必须使用别名前缀
- (4) 如何求一门课的**所有先修课程**(先修课，先修课的先修课，...)?



## 3.3.2 连接查询

### 三、外连接 (Outer Join)

Student表

Sno	Sname	Ssex	Sage	Sdept
95001	李勇	男	20	CS
95002	刘晨	女	19	IS
95003	王敏	女	18	MA
95004	张立	男	19	IS

SC表

Sno	Cno	Grade
95001	1	92
95001	2	85
95001	3	88
95002	2	90
95002	3	80

左外连接: Student  SC

Student.sno	Sname	Ssex	Sage	Sdept	Cno	Grade
95001	李勇	男	20	CS	1	92
95001	李勇	男	20	CS	2	85
95001	李勇	男	20	CS	3	88
95002	刘晨	女	19	IS	2	90
95002	刘晨	女	19	IS	3	80
95003	王敏	女	18	MA	NULL	NULL
95004	张立	男	19	IS	NULL	NULL



## 3.3.2 连接查询

[例35] 查询每个学生及其选修课程的情况(含未选课的学生信息)。

```
SELECT Student.Sno, Sname, Ssex, Sage, Sdept, Cno, Grade  
FROM Student LEFT OUTER JOIN SC  
ON Student.Sno = SC.Sno ;
```

或

```
SELECT Student.Sno, Sname, Ssex, Sage, Sdept, Cno, Grade  
FROM Student LEFT OUTER JOIN SC  
USING(Sno);
```

**注:**

- (1) USING表示在两个关系的某些同名列上进行连接
- (2) 右外连接: RIGHT OUTER JOIN
- (3) 全外连接: FULL OUTER JOIN
- (4) 内连接: INNER JOIN



## 3.3.2 连接查询

### 四、复合条件连接

-- WHERE子句中含多个连接条件

[例36] 查询选修2号课程且成绩在90分以上的所有学生的学号、姓名。

```
SELECT Student.Sno, student.Sname
FROM Student, SC
WHERE Student.Sno = SC.Sno AND // 连接谓词
      SC.Cno= '2' AND // 其他限定条件
      SC.Grade > 90; // 其他限定条件
```

[例37] 查询每个学生的学号、姓名、选修的课程名及成绩。

```
SELECT Student.Sno, Sname, Cname, Grade
FROM Student, SC, Course
WHERE Student.Sno = SC.Sno AND
      SC.Cno = Course.Cno;
```