

3.4 数据查询

3.4.1 单表查询

3.4.2 连接查询

3.4.3 嵌套查询

3.4.4 集合查询

3.4.5 基于派生表的查询

3.4.5 Select语句的一般形式

嵌套查询（续）

● 嵌套查询概述

- 一个SELECT-FROM-WHERE语句称为一个查询块
- 将一个查询块嵌套在另一个查询块的WHERE子句或HAVING短语的条件中的查询称为嵌套查询

```
SELECT Sname                                /*外层查询/父查询*/  
FROM Student  
WHERE Sno IN  
      ( SELECT Sno                          /*内层查询/子查询*/  
        FROM SC  
        WHERE Cno= ' 2 ' );
```

嵌套查询（续）

- 上层的查询块称为外层查询或父查询
- 下层查询块称为内层查询或子查询
- SQL语言允许多层嵌套查询
 - 即一个子查询中还可以嵌套其他子查询
- 子查询的限制
 - 不能使用ORDER BY子句, ORDER BY 子句只能对最终查询结果排序

嵌套查询求解方法

- 不相关子查询：

子查询的查询条件不依赖于父查询

- 由里向外 逐层处理。即每个子查询在上一级查询处理之前求解，子查询的结果用于建立其父查询的查找条件。

嵌套查询求解方法（续）

- **相关子查询：** 子查询的查询条件依赖于父查询
 - 首先取外层查询中表的第一个元组，根据它与内层查询相关的属性值处理内层查询，若WHERE子句返回值为真，则取此元组放入结果表
 - 然后再取外层表的下一个元组
 - 重复这一过程，直至外层表全部检查完为止



3. 4. 3 嵌套查询

- 1.带有IN谓词的子查询
- 2.带有比较运算符的子查询
- 3.带有ANY（SOME）或ALL谓词的子查询
- 4.带有EXISTS谓词的子查询

1. 带有IN谓词的子查询

[例 3.55] 查询与“刘晨”在同一个系学习的学生。

此查询要求可以分步来完成

① 确定“刘晨”所在系名

```
SELECT Sdept
```

```
FROM Student
```

```
WHERE Sname= '刘晨';
```

结果为： CS

带有IN谓词的子查询（续）

② 查找所有在CS系学习的学生。

```
SELECT  Sno, Sname, Sdept  
FROM    Student  
WHERE   Sdept= ' CS ';
```

结果为：

Sno	Sname	Sdept
201215121	李勇	CS
201215122	刘晨	CS

带有IN谓词的子查询（续）

将第一步查询嵌入到第二步查询的条件中

```
SELECT Sno, Sname, Sdept
```

```
FROM Student
```

```
WHERE Sdept IN
```

```
(SELECT Sdept
```

```
FROM Student
```

```
WHERE Sname= '刘晨');
```

此查询为不相关子查询

带有IN谓词的子查询（续）

用自身连接完成[例 3.55]查询要求

```
SELECT S1.Sno, S1.Sname, S1.Sdept
```

```
FROM   Student S1, Student S2
```

```
WHERE  S1.Sdept = S2.Sdept AND
```

```
       S2.Sname = '刘晨';
```

带有IN谓词的子查询（续）

[例 3.56]查询选修了课程名为“信息系统”的学生学号和姓名

```
SELECT Sno,Sname
```

③ 最后在Student关系中

```
FROM Student
```

取出Sno和Sname

```
WHERE Sno IN
```

```
(SELECT Sno
```

② 然后在SC关系找出选

```
FROM SC
```

修了3号课程的学生学号

```
WHERE Cno IN
```

```
(SELECT Cno
```

① 首先在Course关系找出

```
FROM Course
```

“信息系统”的课程号，为3号

```
WHERE Cname= '信息系统'
```

```
)
```

```
);
```

带有IN谓词的子查询（续）

用连接查询实现[例 3.56]：

```
SELECT Sno, Sname
```

```
FROM   Student,SC,Course
```

```
WHERE Student.Sno = SC.Sno AND
```

```
       SC.Cno = Course.Cno AND
```

```
       Course.Cname='信息系统';
```

3.4.3 嵌套查询

1. 带有IN谓词的子查询
2. 带有比较运算符的子查询
3. 带有ANY（SOME）或ALL谓词的子查询
4. 带有EXISTS谓词的子查询

2. 带有比较运算符的子查询

- 当能确切知道内层查询返回单值时，可用比较运算符（>，<，=，>=，<=，!=或<>）。

在[例 3.55]中，由于一个学生只可能在一个系学习，则可以用 = 代替 IN：

```
SELECT Sno,Sname,Sdept
FROM Student
WHERE Sdept =
    (SELECT Sdept
     FROM Student
     WHERE Sname= '刘晨');
```

带有比较运算符的子查询（续）

[例 3.57]找出每个学生超过他选修课程平均成绩的课程号。

```
SELECT Sno, Cno
FROM   SC x
WHERE  Grade >=(SELECT AVG (Grade)
                  FROM SC y
                  WHERE y.Sno=x.Sno);
```

相关子查询



带有比较运算符的子查询（续）

- 可能的执行过程

- ①从外层查询中取出SC的一个元组x，将元组x的Sno值（201215121）传送给内层查询。

```
SELECT AVG(Grade)
```

```
FROM SC y
```

```
WHERE y.Sno='201215121';
```


带有比较运算符的子查询（续）

- 可能的执行过程（续）

- ②执行内层查询，得到值88（近似值），将该值传到外层查询：

```
SELECT Sno,Cno
```

```
FROM SC x
```

```
WHERE Grade >=88;
```

带有比较运算符的子查询（续）

❖ 可能的执行过程（续）

➤ ③执行此时的外层查询，得到

(201215121,1)

(201215121,3)

然后外层查询取出下一个元组重复做上述①至③步骤，直到外层的SC元组全部处理完毕。结果为：

(201215121,1)

(201215121,3)

(201215122,2)

3.4.3 嵌套查询

1. 带有IN谓词的子查询
2. 带有比较运算符的子查询
3. 带有ANY（SOME）或ALL谓词的子查询
4. 带有EXISTS谓词的子查询

带有ANY（SOME）或ALL谓词的子查询（续）

谓词语义

ANY: 任意一个值

ALL: 所有值

使用ANY或ALL谓词时必须同时使用比较运算

语义为:

> ANY 大于子查询结果中的某个值

> ALL 大于子查询结果中的所有值

< ANY 小于子查询结果中的某个值

< ALL 小于子查询结果中的所有值

>= ANY 大于等于子查询结果中的某个值

>= ALL 大于等于子查询结果中的所有值

带有ANY（SOME）或ALL谓词的子查询（续）

使用ANY或ALL谓词时必须同时使用比较运算符
语义为（续）

\leq ANY 小于等于子查询结果中的某个值

\leq ALL 小于等于子查询结果中的所有值

$=$ ANY 等于子查询结果中的某个值

$=$ ALL 等于子查询结果中的所有值（通常没有实际意义）

\neq （或 $\lt>$ ）ANY 不等于子查询结果中的某个值

\neq （或 $\lt>$ ）ALL 不等于子查询结果中的任何一个值

带有ANY (SOME) 或ALL谓词的子查询 (续)

[例 3.58] 查询非计算机科学系中比计算机科学系任意一个学生年龄小的学生姓名和年龄

```
SELECT Sname,Sage
```

```
FROM Student
```

```
WHERE Sage < ANY (SELECT Sage
```

```
FROM Student
```

```
WHERE Sdept= ' CS ')
```

```
AND Sdept <> 'CS' ;
```

/*父查询块中的条件*/

带有ANY（SOME）或ALL谓词的子查询（续）

结果:

Sname	Sage
王敏	18
张立	19

执行过程:

- (1) 首先处理子查询，找出CS系中所有学生的年龄，构成一个集合（20,19）
- (2) 处理父查询，找所有不是CS系且年龄小于20 或 19的学生

带有ANY (SOME) 或ALL谓词的子查询 (续)

用聚集函数实现[例 3.58]

```
SELECT Sname,Sage
FROM Student
WHERE Sage <
      (SELECT MAX (Sage)
       FROM Student
       WHERE Sdept= 'CS ')
AND Sdept <> 'CS ';
```


带有ANY (SOME) 或ALL谓词的子查询 (续)

[例 3.59] 查询非计算机科学系中比计算机科学系所有学生年龄都小的学生姓名及年龄。

方法一：用ALL谓词

```
SELECT Sname,Sage
FROM Student
WHERE Sage < ALL
      (SELECT Sage
       FROM Student
       WHERE Sdept= ' CS ')
AND Sdept <> ' CS ';
```

带有ANY (SOME) 或ALL谓词的子查询 (续)

方法二：用聚集函数

```
SELECT Sname,Sage
```

```
FROM Student
```

```
WHERE Sage <
```

```
    (SELECT MIN(Sage)
```

```
        FROM Student
```

```
        WHERE Sdept= ' CS ')
```

```
    AND Sdept <>' CS ';
```

带有ANY（SOME）或ALL谓词的子查询（续）

表3.7 ANY（或SOME），ALL谓词与聚集函数、IN谓词的等价转换关系

	=	<>或!=	<	<=	>	>=
ANY	IN	--	<MAX	<=MAX	>MIN	>= MIN
ALL	--	NOT IN	<MIN	<= MIN	>MAX	>= MAX

3. 4. 3 嵌套查询

- 1.带有IN谓词的子查询
- 2.带有比较运算符的子查询
- 3.带有ANY（SOME）或ALL谓词的子查询
- 4.带有EXISTS谓词的子查询

嵌套查询总结

问题 1： WHERE的判断单元是什么？
元组

问题 2： 子查询的结果是什么？
表（集合）

问题 3： 子查询所得结果在哪些情况下是一个属性值？

- 集合中只有一个元素
- 聚集函数

嵌套查询总结

外层WHERE条件中，究竟可以写入什么内容？

1. 带有IN谓词的子查询 判断元素与集合的属于关系
2. 带有比较运算符的子查询 比较元素与元素
3. 带有ANY或ALL谓词的子查询 比较元素与集合
4. 带有EXISTS谓词的子查询 判断集合是否非空