



3.3.3 嵌套查询

四、带有EXISTS谓词的子查询

- **EXISTS谓词的意义：**
 - 是存在量词 \exists 在SQL中的应用；
 - 带有EXISTS谓词的子查询不返回任何数据，只产生逻辑真值“true”或逻辑假值“false”；
 - 若内层查询结果非空，则返回真值
 - 若内层查询结果为空，则返回假值
 - 由EXISTS引出的子查询，其目标列表表达式通常都用*，因为带EXISTS的子查询只返回真值或假值，给出列名无实际意义。



3.3.3 嵌套查询

[例43] 查询所有选修了1号课程的学生姓名。

元组演算表达式:

$$\{ t \mid \exists u \exists v (S(u) \wedge SC(v) \wedge u[1]=v[1] \wedge v[2]='1' \wedge t[1]=u[2]) \}$$

- (1) 元组演算公式是逻辑谓词，它由一组原子公式经过联结词进行关联，表达了查询结果集中的元组应当满足的条件，可以理解为由查询需要的表、条件限定和赋值三部分组成(这种解释并不正确，但有助于我们写出正确的元组演算表达式)。
- (2) 第一个 \exists 和最后的赋值部分可以转换为SELECT子句和FROM子句。
- (3) 第二个及以后的 \exists 和条件限定语句可以转换为子查询作为WHERE子句的条件， \exists 翻译为EXISTS。



3.3.3 嵌套查询

[例43] 查询所有选修了1号课程的学生姓名。

元组演算表达式:

$$\{ t \mid \exists u \exists v (S(u) \wedge SC(v) \wedge u[1]=v[1] \wedge v[2]='1' \wedge t[1]=u[2]) \}$$

```
SELECT Sname
FROM Student
WHERE EXISTS
  ( SELECT *
    FROM SC
    WHERE Sno = Student.Sno AND
          Cno= '1' );
```

* 此查询可以理解为：输出那些学生的姓名，当在选课表中存在他(她)选修1号课的记录。



3.3.3 嵌套查询

[例43] 查询所有选修了1号课程的学生姓名。

```
SELECT Sname
FROM Student
WHERE EXISTS
  ( SELECT *
    FROM SC
    WHERE Sno = Student.Sno AND
          Cno = '1' );
```

此查询的执行过程:

- (1)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；
- (2)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。

结论：是相关子查询，此例可用连接运算实现。



3.3.3 嵌套查询

[例44] 查询没有选修1号课程的学生姓名。

* 此查询可以理解为：输出那些学生的姓名，当在选课表中不存在他(她)选修1号课的记录。

* 元组演算表达式：

$$\{ t \mid \exists u \neg \exists v (S(u) \wedge SC(v) \wedge u[1]=v[1] \wedge v[2]='1' \wedge t[1]=u[2]) \}$$

* SQL语句：

```
SELECT Sname
```

```
FROM Student
```

```
WHERE NOT EXISTS
```

```
  ( SELECT *
```

```
    FROM SC
```

```
    WHERE Sno = Student.Sno AND Cno = '1' );
```

注：此例用连接运算无法实现！

此查询为相关子查询。



3.3.3 嵌套查询

注: (1) 所有带EXISTS或NOT EXISTS谓词的子查询能够被其他形式的子查询等价替换

(2) 所有带IN谓词、比较运算符、ANY和ALL谓词的子查询都能用带EXISTS谓词的子查询等价替换。

[例38] 查询与“刘晨”在同一个系学习的学生。

可以理解为：输出那些学生，当他(她)与“刘晨”所在系相同时。

用带EXISTS谓词的子查询替换：

```
SELECT Sno, Sname, Sdept
FROM Student S1
WHERE EXISTS
  (SELECT *
   FROM Student S2
   WHERE S2.Sdept = S1.Sdept
        AND S2.Sname = '刘晨');
```



3.3.3 嵌套查询

用 EXISTS/NOT EXISTS 实现全称量词 (难点)

- * SQL语言中没有全称量词 \forall (For All)
- * 可以把有全称量词的谓词转换为等价的有存在量词的谓词:

$$\forall x(P(x) \rightarrow Q) \equiv \neg \exists x(P(x) \wedge \neg Q)$$

[例45] 查询选修了全部课程的学生姓名。

元组演算表达式:

$$\{t \mid \exists u(S(u) \wedge \forall v(C(v) \rightarrow \exists w(SC(w) \wedge w[1]=u[1] \wedge w[2]=v[1])) \wedge t[1]=u[2])\}$$

⇕ 等价转换

$$\{t \mid \exists u(S(u) \wedge \neg \exists v(C(v) \wedge \neg \exists w(SC(w) \wedge w[1]=u[1] \wedge w[2]=v[1])) \wedge t[1]=u[2])\}$$

自然语义: 输出这样的学生, 不存在某门课程, 在他的选课记录里没有选这门课。



3.3.3 嵌套查询

[例45] 查询选修了全部课程的学生姓名。

* 元组演算表达式

$$\{t \mid \exists u(S(u) \wedge \neg \exists v(C(v) \wedge \neg \exists w(SC(w) \wedge w[1]=u[1] \wedge w[2]=v[1])) \wedge t[1]=u[2])\}$$

* SQL查询语句:

```
SELECT Sname
FROM Student
WHERE NOT EXISTS
  ( SELECT *
    FROM Course
    WHERE NOT EXISTS
      ( SELECT *
        FROM SC
        WHERE Sno = Student.Sno
          AND Cno = Course.Cno ) );
```




3.3.3 嵌套查询

[例45] 查询选修了全部课程的学生姓名。

不用EXISTS谓词实现的思路：

因为学生选修的课程均来源于课程表，换言之，查询即是要求学生的选课数量等于课程表的记录数量。

```
SELECT Sname
FROM Student,SC
WHERE Student.Sno=SC.Sno
      GROUP BY Student.Sno,Sname
      HAVING COUNT(*)=
          ( SELECT COUNT(*) FROM Course );
```

注：Sname要加入到分组中，否则结果中无法出现Sname。



3.3.3 嵌套查询

[例46] 查询至少选修学生95002选修的全部课程的学生号码。

元组演算表达式:

$$\{t \mid \exists u(\text{SC}(u) \wedge \forall v(\text{SC}(v) \wedge v[1]='95002' \rightarrow \exists w(\text{SC}(w) \wedge w[1]=u[1] \wedge w[2]=v[2])) \wedge t[1]=u[1]))\}$$

⇕ 等价转换

$$\{t \mid \exists u(\text{SC}(u) \wedge \neg \exists v(\text{SC}(v) \wedge v[1]='95002' \wedge \neg \exists w(\text{SC}(w) \wedge w[1]=u[1] \wedge w[2]=v[2])) \wedge t[1]=u[1]))\}$$

自然语义: 输出这样的学生, 不存'95002'选了的课程, 在他的选课记录中没有出现。



3.3.3 嵌套查询

[例46] 查询至少选修学生95002选修的全部课程的学生号码。

* 元组演算表达式:

$$\{t \mid \exists u(SC(u) \wedge \neg \exists v(SC(v) \wedge v[1]='95002' \wedge \neg \exists w(SC(w) \wedge w[1]=u[1] \wedge w[2]=v[2])) \wedge t[1]=u[1])\}$$

* SQL查询语句:

```
SELECT DISTINCT Sno
FROM SC SCX
WHERE NOT EXISTS
  ( SELECT *
    FROM SC SCY
    WHERE SCY.Sno = '95002' AND NOT EXISTS
      ( SELECT *
        FROM SC SCZ
        WHERE SCZ.Sno=SCX.Sno AND
              SCZ.Cno=SCY.Cno));
```