

# 嵌套查询（上）



# 讲授内容

- 1 嵌套查询的概念
- 2 嵌套查询的方式



## 嵌套查询的概念

### 一个完整的数据查询语句的格式

SELECT [ ALL|DISTINCT ] <目标列表达式1> [, <目标列表达式2> , ... ]

FROM <表名或视图名1> [, <表名或视图名2> , ... ]

[ WHERE <元组选择条件表达式> ]

[ GROUP BY <属性列名1> [, <属性列名2> , ... ] [ HAVING <组选择条件表达式> ]

[ ORDER BY <目标列名1> [ASC|DESC] [, <目标列名2> [ASC|DESC] , ... ]

父查询

嵌套查询

子查询

SELECT [ ALL|DISTINCT ] <目标列表达式1> [, <目标列表达式2> , ... ]

FROM <表名或视图名1> [, <表名或视图名2> , ... ]

[ ... ]



## 实现嵌套的方式

- 谓词IN
- 比较操作符
- 量词ANY或ALL
- 谓词EXISTS





## 谓词IN实现嵌套

- IN谓词用于判断一个值是否属于一个集合。

**E [NOT] IN (V1, V2, ..., Vn)**

子查询



## 谓词IN实现嵌套

### 查询选修“C02”课程的学生姓名

```
SELECT SN
FROM S
WHERE SNO IN
( SELECT SNO
  FROM SC
  WHERE CNO= 'C02' )
```

独立子查询：  
子查询的结果不依  
赖于父查询。

S (SNO, SN, SD, SB, SEX)

C (CNO, CN, PC)

SC (SNO, CNO, GRADE)

```
SELECT SN
FROM S
WHERE SNO IN
      (SELECT SNO FROM SC
       WHERE CNO='C02')
```

	SN
1	王玲
2	罗军
3	赵泽
4	王明
5	费汉萌



## 谓词IN实现嵌套

### ▶ 查询选修“C02”课程的学生姓名

```
SELECT SN  
FROM S, SC  
WHERE CNO = 'C02' AND  
S.SNO=SC.SNO
```

连接查询

S (SNO, SN, SD, SB, SEX)

C (CNO, CN, PC)

SC (SNO, CNO, GRADE)

```
SELECT SN  
FROM S, SC  
WHERE CNO = 'C02' AND S.SNO=SC.SNO
```

	SN
1	王玲
2	罗军
3	赵泽
4	王明
5	费汉萌



## 谓词IN实现嵌套

### ► 查询没有选修“C02”课程的学生姓名

```
SELECT SN  
FROM S  
WHERE SNO NOT IN  
      (SELECT SNO  
       FROM SC  
       WHERE CNO= 'C02' )
```

该查询能否用连接查询来实现?







## 比较操作符实现嵌套

- ▶ 查询选修“C01”课程的成绩高于王玲的学生的学号和成绩

SELECT SNO, GRADE

FROM SC

WHERE CNO = 'C01' AND

GRADE > ( )

王玲的C01课程成绩

S (SNO, SN, SD, SB, SEX)

C (CNO, CN, PC)

SC (SNO, CNO, GRADE)



## 比较操作符实现嵌套

- ▶ 查询选修“C01”课程的成绩高于王玲的学生的学号和成绩

```
SELECT SNO, GRADE  
FROM SC
```

```
WHERE CNO = 'C01' AND GRADE >
```

王玲的C01课程成绩

```
(SELECT GRADE  
FROM SC
```

```
WHERE CNO= 'C01' AND SNO IN  
(  
))
```

王玲的学号

S (SNO, SN, SD, SB, SEX)

C (CNO, CN, PC)

SC (SNO, CNO, GRADE)



## 比较操作符实现嵌套

- 查询选修“C01”课程的成绩高于王玲的学生的学号和成绩

```
SELECT SNO, GRADE
FROM SC
```

```
WHERE CNO = 'C01' AND GRADE >
```

王玲的C01课程成绩

```
(SELECT GRADE
FROM SC
```

```
WHERE CNO= 'C01' AND SNO IN
```

```
(SELECT SNO
FROM S
```

```
WHERE SN = '王玲' ) )
```

王玲的学号

S (SNO, SN, SD, SB, SEX)

C (CNO, CN, PC)

SC (SNO, CNO, GRADE)



## 比较操作符实现嵌套

- 查询每个学生所修课程成绩超过其所有选课平均成绩的学号和课程号

```
SELECT SNO, CNO
FROM SC
WHERE GRADE >
  SELECT AVG (GRADE)
  FROM SC
  WHERE SC .SNO=SC .SNO)
```

	SNO	CNO	GRADE
1	s01	c01	80.0
2	s01	c02	98.0
3	s01	c03	85.0
4	s01	c04	78.0
5	s01	c07	89.0
6	s02	c02	80.0
7	s02	c11	80.0
8	s03	c01	90.0
9	s03	c02	80.0

SC

	SNO	CNO	GRADE
1	s01	c01	80.0
2	s01	c02	98.0
3	s01	c03	85.0
4	s01	c04	78.0
5	s01	c07	89.0
6	s02	c02	80.0
7	s02	c11	80.0
8	s03	c01	90.0
9	s03	c02	80.0

SC



## 比较操作符实现嵌套

- 查询每个学生所修课程成绩超过其所有选课平均成绩的学号和课程号

```
SELECT SNO, CNO
FROM SC SC1
WHERE GRADE >
      (SELECT AVG (GRADE)
       FROM SC SC2
        WHERE SC2.SNO=SC1.SNO)
```

相关子查询：  
子查询的结果与父查询当前元组值有关

	SNO	CNO	GRADE
1	s01	c01	80.0
2	s01	c02	98.0
3	s01	c03	85.0
4	s01	c04	78.0
5	s01	c07	89.0
6	s02	c02	80.0
7	s02	c11	80.0
8	s03	c01	90.0
9	s03	c02	80.0

SC 1

	SNO	CNO	GRADE
1	s01	c01	80.0
2	s01	c02	98.0
3	s01	c03	85.0
4	s01	c04	78.0
5	s01	c07	89.0
6	s02	c02	80.0
7	s02	c11	80.0
8	s03	c01	90.0
9	s03	c02	80.0

SC 2





## 量词ANY或ALL实现嵌套

- ANY（子查询）
  - ANY 的语义为查询结果中的某个值，当子查询结果中有某一个值满足比较运算符，比较运算结果则为真。
- ALL（子查询）
  - ALL的语义为查询结果中的所有值，当子查询结果中每一个值都满足比较运算符，比较运算结果才为真。



## 量词ANY或ALL实现嵌套

- ▶ 查询其他系中比计算机系某一学生年龄小的学生

```
SELECT *  
FROM S  
WHERE SD < > '计算机' AND SB >  
    ANY (SELECT SB  
          FROM S  
          WHERE SD = '计算机' )
```

S (SNO, SN, SD, SB, SEX)

C (CNO, CN, PC)

SC (SNO, CNO, GRADE)



## 量词ANY或ALL实现嵌套

- ▶ 查询其他系中比计算机系学生年龄都小的学生

```
SELECT *  
FROM S  
WHERE SD < > '计算机' AND SB >  
ALL (SELECT SB  
      FROM S  
      WHERE SD = '计算机' )
```

S (SNO, SN, SD, SB, SEX)

C (CNO, CN, PC)

SC (SNO, CNO, GRADE)

独立子查询



## 量词ANY或ALL实现嵌套

- 量词ANY或ALL 与比较符配合的功能可用谓词IN或用聚集函数与比较符配合来实现。

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



## 量词ANY或ALL实现嵌套

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

- =ANY (子查询) → IN 谓词 (子查询)
- <ANY (子查询) → <MAX (子查询)
- >ANY (子查询) → >MIN (子查询)
- <>ALL (子查询) → NOT IN (子查询)
- <ALL (子查询) → <MIN (子查询)
- >ALL (子查询) → >MAX (子查询)





## 量词ANY或ALL实现嵌套

	=	< > 或 !=	<	<=	>	>=
ANY	IN	TRUE	<MAX	<=MAX	>MIN	>=MIN
ALL	FALSE	NOT IN	<MIN	<=MIN	>MAX	>=MAX

- <>ANY (子查询) ~~→~~ NOT IN (子查询)
- =ALL (子查询) ~~→~~ IN (子查询)



## 量词ANY或ALL实现嵌套

### ► 查询其他系中比计算机系某一学生年龄都小的学生

```
SELECT *  
FROM S  
WHERE SD <> '计算机' AND SB > ANY  
    (SELECT SB  
     FROM S  
     WHERE SD = '计算机' )
```

---

```
SELECT *  
FROM S  
WHERE SD <> '计算机' AND SB >  
    (SELECT MIN(SB)  
     FROM S  
     WHERE SD = '计算机' )
```



## 量词ANY或ALL实现嵌套

### ► 查询其他系中比计算机系学生年龄都小的学生

```
SELECT *  
FROM S  
WHERE SD <> '计算机' AND SB > ALL  
    (SELECT SB  
     FROM S  
     WHERE SD = '计算机' )
```

---

```
SELECT *  
FROM S  
WHERE SD <> '计算机' AND SB >  
    (SELECT MAX (SB)  
     FROM S  
     WHERE SD = '计算机' )
```