

习题解析-2

• 设有关系 R 和 S:

R:

A	B	C
3	6	7
2	5	7
7	2	3
4	4	3

S:

A	B	C
3	4	5
7	2	3

$R \cap S$ 的结果是:

A	B	C
7	2	3

$\pi_{3,2}(S)$

C	D
5	4
3	2

$R \bowtie S$ 的结果:

A	B	C	D	E	F
7	2	3	3	4	5

$\sigma_{B < 5}(R)$ 的结果:

A	B	C
7	2	3
4	4	3

$R \bowtie S$ 的结果:

A	B	C
7	2	3

计算 $R \cup S$, $R - S$, $R \cap S$, $R \times S$, $\pi_{3,2}(S)$, $\sigma_{B < 5}(R)$, $R \bowtie S$, $R \bowtie S$.

答: $R \cup S$ 的结果是:

A	B	C
3	6	7
2	5	7
7	2	3
4	4	3
3	4	5

$R - S$ 的结果是:

A	B	C
3	6	7
2	5	7
4	4	3

$R \times S$ 的结果是:

A	B	C	D	E	F
3	6	7	3	4	5
3	6	7	7	2	3
2	5	7	3	4	5
2	5	7	7	2	3
7	2	3	3	4	5
7	2	3	7	2	3
4	4	3	3	4	5
4	4	3	7	2	3

3.7 如果 R 是二元关系，那么下列元组表达式的结果是什么？

$$\{t \mid (\exists u) (R(t) \wedge R(u) \wedge (t[1] \neq u[1] \vee t[2] \neq u[2]))\}$$

答：当 R 的元组数 ≥ 2 时，R 中每个元组都存在与之不相同的元组，因此表达式的结果为关系 R；
当 R 的元组数为 0 或 1 时，表达式的结果为空关系。

3.8 假设 R 和 S 分别是三元和二元关系，试把表达式 $\pi_{1,5}(\sigma_{2=4 \vee 3=4}(R \times S))$ 转换成等价的：

①汉语查询句子；②元组表达式；③域表达式。

答：①汉语查询句子：在关系 R 和 S 的笛卡尔积中，选取第 2 个属性值与第 4 个属性值相等，或者第 3 个属性值与第 4 个属性值相等的那些元组，再取第 1 列和第 5 列组成新的关系。

②等价的元组表达式是：

$$\{t \mid (\exists u) (\exists v) (R(u) \wedge S(v) \wedge t[1]=u[1] \wedge t[2]=u[2] \wedge t[3]=u[3] \wedge t[4]=v[1] \wedge t[5]=v[2])\}$$

与 $\sigma_{2=4 \vee 3=4}(R \times S)$ 等价的元组表达式是：

$$\{t \mid (\exists u) (\exists v) (R(u) \wedge S(v) \wedge t[1]=u[1] \wedge t[2]=u[2] \wedge t[3]=u[3] \wedge t[4]=v[1] \wedge t[5]=v[2] \wedge (t[2]=t[4] \vee t[3]=t[4]))\}$$

与 $\pi_{1,5}(\sigma_{2=4 \vee 3=4}(R \times S))$ 等价的元组表达式是：

$$\{w \mid (\exists t) (\exists u) (\exists v) (R(u) \wedge S(v) \wedge t[1]=u[1] \wedge t[2]=u[2] \wedge t[3]=u[3] \wedge t[4]=v[1] \wedge t[5]=v[2] \wedge (t[2]=t[4] \vee t[3]=t[4]) \wedge w[1]=t[1] \wedge w[2]=t[5])\}$$

再对上述元组表达式化简（消去 t）可得：

$$\{w \mid (\exists u) (\exists v) (R(u) \wedge S(v) \wedge (u[2]=v[1] \vee u[3]=v[1]) \wedge w[1]=u[1] \wedge w[2]=v[2])\}$$

在熟练后，可以直接写出上式。

③ 再转换成域表达式:

$$\{w_1w_2 \mid (\exists u_1) (\exists u_2) (\exists u_3) (\exists v_1) (\exists v_2) (R(u_1u_2u_3) \wedge S(v_1v_2) \wedge (u_2=v_1 \vee u_3=v_1) \wedge w_1=u_1 \wedge w_2=v_2)\}$$

再化简 (消去 u_1, v_2) 可得:

$$\{w_1w_2 \mid (\exists u_2) (\exists u_3) (\exists v_1) (R(w_1u_2u_3) \wedge S(v_1w_2) \wedge (u_2=v_1 \vee u_3=v_1))\}$$

3.9 假设 R 和 S 都是二元关系, 试把元组表达式 $\{t \mid R(t) \wedge (\exists u) (S(u) \wedge u[1] \neq t[2])\}$ 转换成等价的:

①汉语查询句子; ②域表达式; ③关系代数表达式。

答: ①在关系 R 中选取第 2 列的值与关系 S 中某个元组的第 1 列值不相等的那些元组, 组成新的关系。

②域表达式为: $\{t_1t_2 \mid R(t_1t_2) \wedge (\exists u_1) (\exists u_2) (S(u_1u_2) \wedge u_1 \neq t_2)\}$

③关系代数表达式为:

$$\pi_{1,2}(\sigma_{2 \neq 3}(R \times S)) \text{ 或 } \pi_{1,2}(R \bowtie_{2 \neq 3} S)$$

3.10 试把域表达式 $\{ab \mid R(ab) \wedge R(ba)\}$ 转换成等价的: (1)汉语查询句子; (2)关系代数表达式; (3)元组表达式。

答: (1) 在关系 R 中选取属性值交换后仍是 R 中元组的那些元组, 组成新的关系。

(2) 关系代数表达式为: $\pi_{1,2}(\sigma_{1 \neq 2 \wedge 2 \neq 3}(R \times R))$

也可写成: $R \cap \pi_{2,1}(R)$

(3) 元组表达式为: $\{t \mid (\exists u) (\exists v) (R(u) \wedge R(v) \wedge u[1]=v[2] \wedge u[2]=v[1] \wedge t[1]=u[1] \wedge t[2]=u[2])\}$

或: $\{t \mid (\exists v) (R(t) \wedge R(v) \wedge t[1]=v[2] \wedge t[2]=v[1])\}$

3.11 有两个关系 R (A, B, C) 和 S (D, E, F), 试把下列关系代数表达式转换成等价的元组表达式:

$$\textcircled{1} \pi_A(R);$$

$$\textcircled{2} \sigma_{B='17'}(R);$$

$$\textcircled{3} R \times S;$$

$$\textcircled{4} \pi_{A,F}(\sigma_{C=D}(R \times S))$$

答: $\textcircled{1} \pi_A(R): \{ t | (\exists u)(R(u) \wedge t[1]=u[1]) \}$

$$\textcircled{2} \sigma_{B='17'}(R): \{ t | R(t) \wedge t[2]='17' \}$$

$$\textcircled{3} R \times S: \{ t | (\exists u)(\exists v)(R(u) \wedge S(v) \wedge t[1]=u[1] \wedge t[2]=u[2] \wedge t[3]=u[3] \wedge t[4]=v[1] \wedge t[5]=v[2] \wedge t[6]=v[3]) \}$$

$$\textcircled{4} \pi_{A,F}(\sigma_{C=D}(R \times S)): \{ t | (\exists u)(\exists v)(R(u) \wedge S(v) \wedge u[3]=v[1] \wedge t[1]=u[1] \wedge t[2]=v[3]) \}$$

3.12 设有三个关系

S (SNO,SNAME,AGE,SEX,SDEPT)

SC (SNO,CNO,GRADE)

C (CNO,CNAME,CDEPT,TNAME)

试用关系代数表达式表示下列查询语句：

(1) 检索 LIU 老师所授课程的课程号、课程名。

答: $\pi_{CNO,CNAME}(\sigma_{TNAME='LIU'}(C))$

(2) 检索年龄大于 23 岁的男学生的学号与姓名。

答: $\pi_{SNO,SNAME}(\sigma_{AGE>'23' \wedge SEX='男'}(S))$

(3) 检索学号为 S3 学生所学课程的课程名与任课教师名。

答: $\pi_{CNAME,TNAME}(\sigma_{SNO='S3'}(SC \bowtie C))$

(4) 检索至少选修 LIU 老师所授课程中一门课的女学生姓名。

答: $\pi_{SNAME}(\sigma_{SEX='女' \wedge TNAME='LIU'}(S \bowtie SC \bowtie C))$

(5) 检索 WANG 同学不学的课程的课程号。

答: $\pi_{CNO}(C) - \pi_{CNO}(\sigma_{SNAME='WANG'}(S \bowtie SC))$

(6) 检索至少选修两门课程的学生学号。

答: $\pi_1(\sigma_{2 \neq 5 \wedge 1=4}(SC \times SC))$

(7) 检索全部学生都选修的课程的课程号与课程名。

答: $\pi_{CNO,CNAME}(C \bowtie \pi_{SNO,CNO}(SC) \div \pi_{SNO}(S))$

(8) 检索选修课程包含 LIU 老师所授课程的学生学号。

答: $\pi_{SNO,CNO}(SC) \div \pi_{CNO}(\sigma_{TNAME='LIU'}(C))$

(6) 检索至少选修两门课程的学生学号。

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

(7)检索全部学生都选修的课程的课程号与课程名。

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

(8)检索选修课程包含 LIU 老师所授课程的学生学号。

$\{ t \mid (\exists u) (S(u) \wedge u[3] > '23' \wedge u[4] = '男' \wedge t[1]=u[1] \wedge t[2]=u[2]) \}$ (8) $\{ t \mid (\exists u) (SC(u) \wedge (\forall v) (C(v) \wedge (v[3]='LIU' \Rightarrow (\exists w) (SC(w) \wedge w[1]=u[1] \wedge w[2]=v[1]))) \wedge t[1]=u[1]) \}$

3.13 试用元组表达式表示 3.12 题中各个查询语句。

(1) 检索 LIU 老师所授课程的课程号、课程名。

(1) $\{ t \mid (\exists u) (C(u) \wedge u[3]='LIU' \wedge t[1]=u[1] \wedge t[2]=u[2]) \}$

(2) 检索年龄大于 23 岁的男学生的学号与姓名。

$\{ t \mid (\exists u) (S(u) \wedge u[3] > '23' \wedge u[4] = '男' \wedge t[1]=u[1] \wedge t[2]=u[2]) \}$

(3)检索学号为 S3 学生所学课程的课程名与任课教师名。

$\{ t \mid (\exists u) (\exists v) (SC(u) \wedge C(v) \wedge u[1]='S3' \wedge u[2]=v[1] \wedge t[1]=v[2] \wedge t[2]=v[3]) \}$

此处自然联接条件 $u[2]=v[1]$ 不要遗漏)

(4) 检索至少选修 LIU 老师所授课程中一门课的女学生姓名。

(4) $\{ t \mid (\exists u) (\exists v) (\exists w) (S(u) \wedge SC(v) \wedge C(w) \wedge w[3]='LIU' \wedge u[4]='F' \wedge u[1]=v[1] \wedge v[2]=w[1] \wedge t[1]=u[2]) \}$

(此处自然联接条件 $u[1]=v[1]$ 和 $v[2]=w[1]$ 不要遗漏)

(5) 检索 WANG 同学不学的课程的课程号。

(5) $\{ t \mid (\exists u) (\exists v) (\forall w) (C(u) \wedge S(v) \wedge SC(w) \wedge v[2]='WANG' \wedge (w[1]=v[1] \Rightarrow w[2] \neq u[1]) \wedge t[1]=u[1]) \}$

其意思是：在关系 C 中存在一门课程，在关系 S 中存在一个 WANG 同学，在关系 SC 中要求不存在 WANG 同学学这门课程的元组。也就是要求在关系 SC 中，WANG 同学的课程都不是这门课程（因此在元组表达式中要求全称量词 \forall ）。

S (SNO,SNAME,AGE,SEX,SDEPT)

SC (SNO,CNO,GRADE)

C (CNO,CNAME,CDEPT,TNAME)

3.14 试用域表达式表示第 3.12 题的各个查询语句。

(1) 检索 LIU 老师所授课程的课程号、课程名。

$$\{t_1 t_2 | (\exists u_1 u_2 u_3) (C(u_1 u_2 u_3) \wedge u_3 = \text{'LIU'} \wedge t_1 = u_1 \wedge t_2 = u_2)\} \\ \{t_1 t_2 | C(t_1 t_2 \text{'LIU'})\}$$

(2) 检索年龄大于 23 岁的男学生的学号与姓名。

$$\{t_1 t_2 | (\exists u_1 u_2 u_3 u_4) (S(u_1 u_2 u_3 u_4) \wedge u_3 > \text{'23'} \wedge u_4 = \text{'男'} \wedge t_1 = u_1 \wedge t_2 = u_2)\} \\ \{t_1 t_2 | (\exists u_3) (S(t_1 t_2 u_3 \text{'M'}) \wedge u_3 > \text{'23'})\}$$

(3) 检索学号为 S3 学生所学课程的课程名与任课教师名。

$$\textcircled{3} \{t_1 t_2 | (\exists u_1 u_2 u_3) (\exists v_1 v_2 v_3) (SC(u_1 u_2 u_3) \wedge C(v_1 v_2 v_3) \wedge u_1 = \text{'S3'} \wedge u_2 = v_1 \wedge t_1 = v_2 \wedge t_2 = v_3)\}$$

(4) 检索至少选修 LIU 老师所授课程中一门课的女学生姓名。

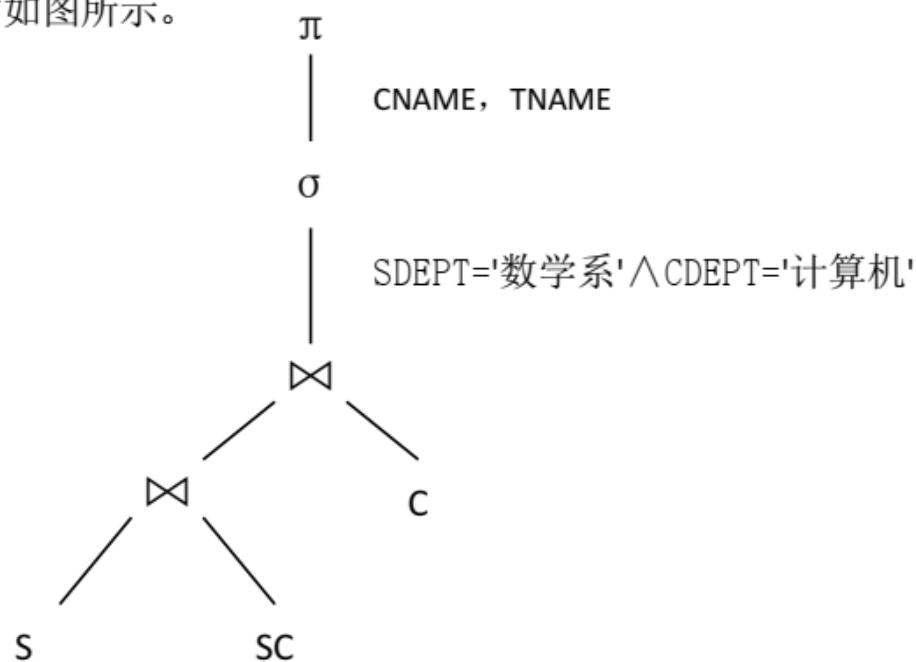
$$\textcircled{4} \{t_1 | (\exists u_1 u_2 u_3 u_4) (\exists v_1 v_2 v_3) (\exists w_1 w_2 w_3) (S(u_1 u_2 u_3 u_4) \wedge SC(v_1 v_2 v_3) \wedge C(w_1 w_2 w_3) \wedge w_3 = \text{'LIU'} \wedge u_4 = \text{'F'} \wedge u_1 = v_1 \wedge v_2 = w_1 \wedge t_2 = u_2)\}$$

3.15 在 3.12 题的三个关系中，用户有一查询语句：检索数学系的学生选修计算机课程的课程名和任课教师姓名。①试写出该查询的关系代数表达式；②画出该查询初始的关系代数表达式的语法树；③使用 3.4.4 节的优化算法，对语法树进行优化，试写出该查询优化的关系代数表达式；④画出优化后的语法树。

① 关系代数表达式为：检索数学系的学生选修计算机课程的课程名和任课教师姓名

$\pi_{CNAME, TNAME} (\sigma_{SDEPT='数学系' \wedge CDEPT='计算机'} (S \bowtie SC \bowtie C))$

② 上述关系代数表达式的语法树如图所示。



③ 上述的关系代数表达式为：

$\pi_{CNAME, TNAME} (\sigma_{SDEPT='数学系' \wedge CDEPT='计算机'} (\pi_L (\sigma_{S.SNO=SC.SNO \wedge SC.CNO=C.CNO} ((S \times SC) \times C))))$

此处 L 为 S、SC、C 中全部属性（公共属性只取一次）。

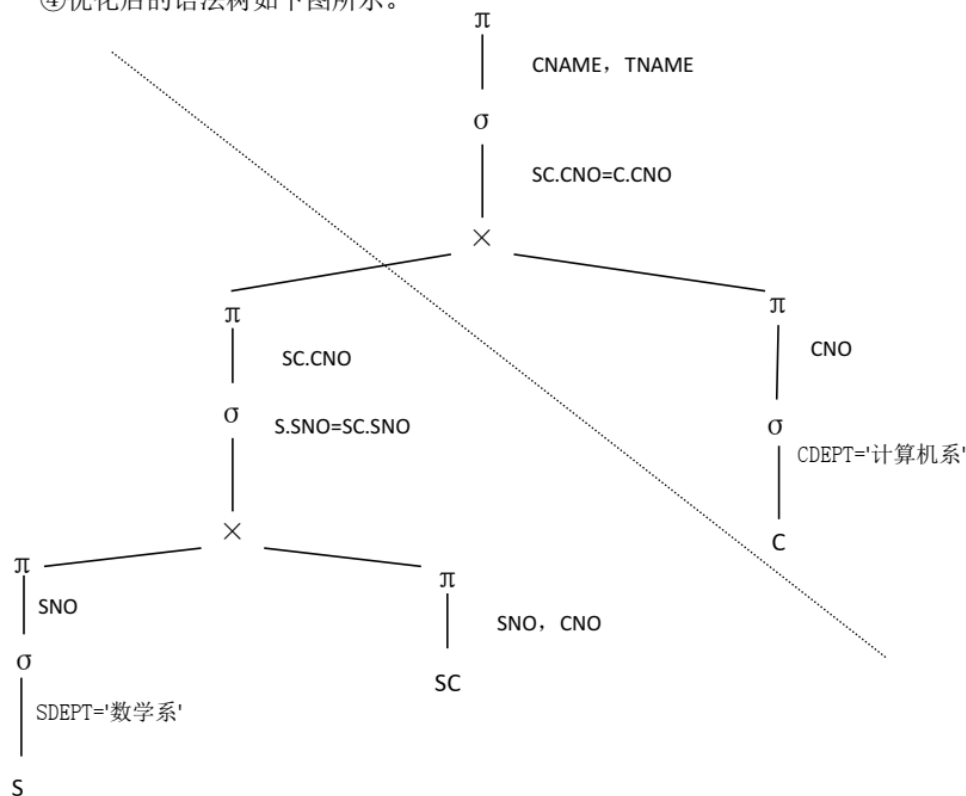
设 $L1 = \pi_{SNO} (\sigma_{SDEPT='数学系'} (S))$

$L2 = \pi_{SNO, CNO} (SC)$

则优化的关系代数表达式为：

$\pi_{CNAME, TNAME} (\sigma_{SC.CNO=C.CNO} (\pi_{SC.C\#} (\sigma_{S.SNO=SC.SNO} (L1 \times L2)) \times C))$

④优化后的语法树如下图所示。



S (SNO, SNAME, AGE, SEX, SDEPT)

SC (SNO, CNO, GRADE)

C (CNO, CNAME, CDEPT, TNAME)

3.16 为什么要对关系代数表达式进行优化？有哪三条启发式规则？对优化起什么作用？

答：关系代数表达式由关系代数操作组合而成。操作中，以笛卡尔积和联接操作最费时，并生成大量的中间结果。如果直接按表达式书写的顺序执行，必将花费很多时间，并生成大量的中间结果，效率较低。在执行前，由 **DBMS** 的查询子系统先对关系代数表达式进行优化，尽可能先执行选择和投影操作，以便减少中间结果，并节省时间。

优化工作是由 **DBMS** 做的，用户书写时不必关心优化一事，仍以简练的形式书写。三条启发式规则是：尽可能早执行选择操作；尽可能早执行投影操作；把笛卡尔积与附近的一连串选择和投影合并起来做。

使用这三条规则，可以使计算时尽可能减少中间关系的数据量。

4.2 S (SNO,SNAME, AGE, SEX,SDEPT)

SC (SNO, CNO,GRADE)

C (CNO, CNAME,CDEPT,TNAME) 试用 SQL 的查询语句表达下列查询:

1) 检索 LIU 老师所授课程的课程号和课程名。

答: SELECT CNO,CNAME FROM C WHERE TNAME='LIU' ;

(2) 检索年龄大于 23 岁的男学生的学号和姓名。

答: SELECT SNO,SNAME FROM S WHERE AGE>23 AND SEX='男' ;

(3) 检索学号为 S3 的学生所学课程的课程名和任课教师名。

答: SELECT CNAME,TNAME FROM SC,C WHERE SC.CNO=C.CNO AND CNO='S3' ;

4) 检索至少选修 LIU 老师所授课程中一门课程的女学生姓名。

答: 解法一: 使用联接查询方式

SELECT SNAME FROM S,SC,C WHERE S.SNO=SC.SNO **AND** SC.CNO=C.CNO **AND** SEX='女' AND TNAME='LIU';

解法二: 使用嵌套查询方式

SELECT SNAME FROM S WHERE SEX='女' AND SNO IN
(SELECT SNO FROM SC WHERE CNO IN (SELECT CNO FROM C WHERE TNAME='LIU'));

解法三: 使用存在量词查询方式

SELECT SNAME FROM S WHERE SEX='女' AND EXISTS
(SELECT * FROM SC WHERE S.SNO=SC.SNO AND EXISTS
(SELECT * FROM C WHERE SC.CNO=C.CNO AND TNAME='LIU'));

- 5) 检索 WANG 同学不学的课程的课程号。
 答:

```
SELECT CNO FROM C WHERE EXISTS
      (SELECT * FROM S WHERE SNAME='WANG'
      AND NOT EXISTS
      (SELECT * FROM SC WHERE SC.SNO=S.SNO AND SC.CNO=C.CNO));
```
- ```
Select cno from c where cno not in(select cno from s,sc where SNAME='WANG')
```
- (6) 检索至少选修两门课程的学生学号。  
 答: 

```
SELECT SNO FROM SC
 GROUP BY SNO
 HAVING COUNT(*)>=2;
```
- (7) 检索全部学生都选修的课程的课程号与课程名。  
 答: 

```
SELECT CNO,CNAME FROM C
 WHERE NOT EXISTS
 (SELECT * FROM S WHERE NOT EXISTS
 (SELECT * FROM SC WHERE SC.SNO=S.SNO AND SC.CNO=C.CNO));
```
- (8) 检索选修课程包含 LIU 老师所授课程的学生学号。  
 答: 

```
SELECT DISTINCT SNO FROM SC X WHERE NOT EXISTS
 (SELECT * FROM C WHERE TNAME='LIU'
 AND NOT EXISTS
 (SELECT * FROM SC Y WHERE Y.SNO=X.SNO
 AND Y.CNO=C.CNO));
```

- 4.6 试用 SQL 查询语句表达下列对教学数据库中三个基本表 S、 SC、 C 的查询：
  - (1) 统计有学生选修的课程门数。
  - 答： SELECT COUNT(DISTINCT CNO) AS 课程门数 FROM SC;
  - (2) 求选修 C4 课程的学生的平均年龄。
  - 答： SELECT AVG(AGE) AS 平均年龄 -----AGE 为 INT 型  
FROM S,SC WHERE S.SNO=SC.SNO AND CNO='C4';
  - 或: SELECT AVG(CAST(S.AGE AS INT)) AS 平均年龄 -----AGE 为字符型  
FROM S,SC WHERE S.SNO=SC.SNO AND CNO='C4';
  - (3) 求 LIU 老师所授课程的每门课程的学生平均成绩。
  - 答: SELECT SC.CNO,AVG(GRADE) AS 平均成绩 FROM SC,C  
WHERE SC.CNO=C.CNO AND TNAME='LIU' GROUP BY SC.CNO;
  - 或: SELECT SC.CNO,CAST(AVG(GRADE) AS numeric(4,2)) AS 平均成绩  
FROM SC,C WHERE SC.CNO=C.CNO AND TNAME='LIU' GROUP BY SC.CNO;

- 4) 统计每门课程的学生选修人数（超 10 人的课程才统计）。要求输出课程号和选修人数，查询结果按人数降序排列，若人数相同，按课程号升序排列。

答: SELECT SC.CNO,COUNT(SNO) AS 选修人数 FROM SC,C  
WHERE SC.CNO=C.CNO GROUP BY SC.CNO HAVING COUNT(\*)>2  
ORDER BY 2 DESC,SC.CNO ASC;

- (5) 检索学号比 WANG 同学大，而年龄比他小的学生姓名。
- 答: SELECT X.SNAME FROM S X, S Y WHERE Y.SNAME = 'WANG'  
AND X.SNO>Y.SNO AND X.AGE<Y.AGE
- 或: SELECT X.SNAME FROM S X INNER JOIN S Y ON X.SNO > Y.SNO AND X.AGE < Y.AGE  
WHERE (Y.SNAME = 'WANG');
- 或: SELECT SNAME FROM S WHERE SNO>(SELECT SNO FROM S WHERE SNAME = 'WANG')  
AND AGE<(SELECT AGE FROM S WHERE SNAME = 'WANG');
- 或: SELECT X.SNAME FROM S X WHERE X.SNO>SOME(SELECT SNO FROM S Y  
WHERE Y.SNAME = 'WANG' AND X.AGE<Y.AGE)
- 或: SELECT SNAME FROM S X WHERE (SNO>ANY(SELECT SNO FROM S Y  
WHERE Y.SNAME='WANG' AND X.AGE<Y.AGE));
- 或: SELECT SNAME FROM S WHERE SNO>(SELECT SNO FROM S WHERE SNAME = 'WANG')  
AND AGE<(SELECT AGE FROM S WHERE SNAME = 'WANG')

(6)在表 SC 中检索成绩为空值的学生学号和课程号。

答: SELECT SNO, CNO FROM SC WHERE GRADE IS NULL

(7)检索姓名以 WANG 打头的所有学生的姓名和年龄。

答: SELECT SNAME, AGE FROM S WHERE SNAME LIKE 'WANG%';

- (8)求年龄大于女同学平均年龄的男学生姓名和年龄。
- 答: SELECT SNAME, AGE FROM S  
WHERE SEX='男' AND AGE>(SELECT AVG(AGE) -----AGE 为 int 型  
FROM S WHERE SEX='女 ');
- 或: SELECT \* FROM S  
WHERE SEX='男' AND AGE>(SELECT AVG(CAST(S.AGE AS INT)) ----AGE 为字符型  
FROM S WHERE SEX='女');
- (9)求年龄大于所有女同学年龄的男学生姓名和年龄。
- 答: SELECT SNAME, AGE FROM S  
WHERE SEX='男' AND AGE>ALL(SELECT AGE FROM S WHERE SEX='女 ');
- 或: SELECT SNAME, AGE FROM S X WHERE SEX='男'  
AND NOT EXISTS (SELECT \* FROM S Y WHERE Y.SEX='女' AND Y.AGE>=X.AGE)



- 4.7 试用 SQL 更新语句表达对教学数据库中三个基本表 S、SC、C 的各个更新操作：
- (1) 往基本表 S 中插入一个学生元组 ('S9', 'WU', 18)。

答: INSERT INTO S(SNO,SNAME,AGE) VALUES('S9','WU',18);

(2) 在基本表 S 中检索每一门课程成绩都大于等于 80 分的学生学号、姓名和性别，并把检索到的值送往另一个已存在的基本表 STUDENT (SNO, SNAME, SEX)。

解法一：有两种情况：

### 1、假设每个学生都选课

```
INSERT INTO STUDENT
SELECT SNO,SNAME,SEX FROM S
WHERE 80<= ALL (SELECT GRADE FROM SC WHERE SC.SNO=S.SNO);
```

## 2、不是每个学生都选课

```
INSERT INTO STUDENT
SELECT DISTINCT(S.SNO),SNAME,SEX FROM S, SC X
WHERE S.SNO = X.SNO AND (80 <= ALL (SELECT GRADE FROM SC Y
WHERE Y.SNO = S.SNO));
```

- 解法二: SELECT DISTINCT S.SNO, S.SNAME, S.SEX  
FROM S INNER JOIN SC ON S.SNO = SC.SNO  
GROUP BY S.SNO, S.SNAME, S.SEX  
HAVING (MIN(SC.GRADE) >= 80);

解法三: SELECT DISTINCT S.SNO,SNAME,SEX  
FROM S,SC X WHERE S.SNO=X.SNO AND NOT EXISTS  
(SELECT \* FROM SC Y WHERE GRADE<80 AND S.SNO=Y.SNO);

解法四:有两种情况:

1、假设每个学生都选课

```
INSERT INTO STUDENT SELECT SNO,SNAME,SEX FROM S
WHERE NOT EXISTS (SELECT DISTINCT CNO FROM SC AS X
WHERE S.SNO = X.SNO AND NOT EXISTS
(SELECT * FROM SC AS Y WHERE S.SNO = Y.SNO
AND X.CNO = Y.CNO AND Y.GRADE >= 80));
```

- 2、不是每个学生都选课：

```
INSERT INTO STUDENT SELECT DISTINCT S.SNO,S.SNAME,S.SEX
FROM S,SCWHERE S.SNO = SC.SNO
AND (NOT EXISTS (SELECT DISTINCT CNO FROM SC AS X
WHERE S.SNO = X.SNO AND NOT EXISTS
(SELECT * FROM SC AS Y
WHERE S.SNO = Y.SNO
AND X.CNO = Y.CNO AND Y.GRADE >= 80))));
```

(3) 在基本表 SC 中删除尚无成绩的选课元组。

答： DELETE FROM SC WHERE GRADE IS NULL;

(4) 把 WANG 同学的选课成绩全部删去。

答： DELETE FROM SC WHERE SNO IN  
(SELECT SNO FROM S WHERE SNAME='WANG');

(5) 把选修 MATHS 课不及格的成绩 全改为空值。

答： UPDATE SC SET GRADE = NULL WHERE GRADE<60 AND  
CNO = (SELECT CNO FROM C WHERE CNAME='MATHS');

- (6)把低于总平均成绩的女同学成绩提高 5%。

答: UPDATE SC SET GRADE=GRADE\*1.05

WHERE SNO IN(SELECT SNO FROM S WHERE SEX='女')

AND GRADE<(SELECT AVG(GRADE) FROM SC);

- (7)在基本表 SC 中修改 C4 课程的成绩, 若成绩小于等于 75 分时提高 5%, 若成绩大于 75 分时提高 4% (用两个 UPDATE 语句实现,顺序不能颠倒)。

答: 用两个 UPDATE 语句实现:

- UPDATE SC SET GRADE=GRADE\*1.04

WHERE CNO='C4' AND GRADE>70;

- UPDATE SC SET GRADE=GRADE\*1.05

WHERE CNO='C4' AND GRADE<=70;

- 4.9 对于教学数据库中基本表 SC，建立一个视图：

```
CREATE VIEW S_GRADE(SNO,C_NUM,AVG_GRADE)
```

```
AS SELECT SNO,COUNT(CNO),AVG(GRADE) FROM SC GROUP BY SNO
```

试判断下列查询和更新操作是否允许执行。如允许，写出转换到基本表 SC 上的相应操作。

①**SELECT \* FROM S\_GRADE;**

答：允许查询。相应的操作如下：

```
SELECT SNO,COUNT(CNO) AS C_NUM,AVG(GRADE) AS AVG_GRADE
FROM SC GROUP BY SNO;
```

②**SELECT SNO,C\_NUM FROM S\_GRADE WHERE AVG\_GRADE>80;**

答：允许查询。相应的操作如下：

```
SELECT SNO,COUNT(CNO) AS C_NUM FROM SC GROUP BY SNO
HAVING AVG(GRADE)> 80;
```

- ③SELECT SNO,AVG\_GRADE FROM S\_GRADE  
WHERE C\_NUM >(SELECT C\_NUM FROM S\_GRADE WHERE SNO='S4');
- 答：允许查询。相应的操作如下：  
SELECT SNO,AVG(GRADE) AS AVG\_GRADE FROM SC  
GROUP BY SNO  
HAVING COUNT(CNO)>(SELECT COUNT(CNO) FROM SC  
GROUP BY SNO  
HAVING SNO='S4');
- 等价于：  
SELECT SNO,AVG(GRADE) AS AVG\_GRADE FROM SC  
GROUP BY SNO  
HAVING COUNT(CNO)>(SELECT COUNT(CNO) FROM SC  
WHERE SNO='S4'  
GROUP BY SNO);

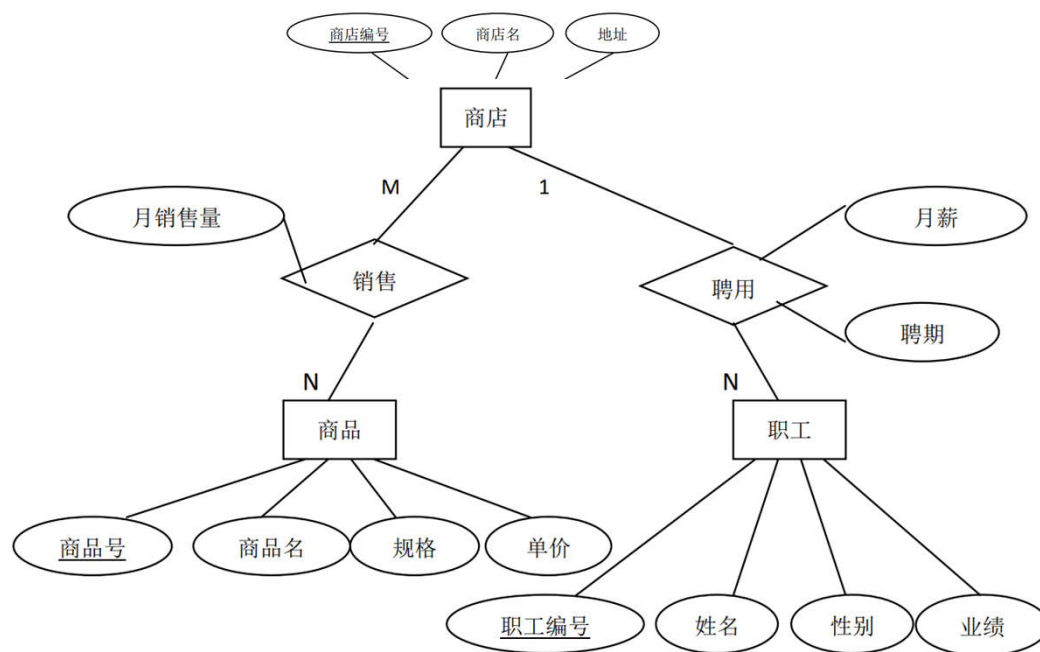
- ④ **UPDATE S\_GRADE**  
**SET SNO='S3'**  
**WHERE SNO='S4';**

答：不允许。C\_NUM 是对 SC 中的学生选修门数进行统计，在未更改 SC 表时，要在视图 S\_GRADE 中更改门数，是不可能的。

- ⑤ **DELETE FROM S\_GRADE WHERE C\_NUM>4;**

答：不允许。在视图 S\_GRADE 中删除选修门数在 4 门以上的学生元组，势必造成 SC 中这些学生学习元组的删除，这不一定是用户的原意，因此使用分组和聚合操作的视图，不允许用户执行更新操作。

## • 6.2



(2) 这个 ER 图可转换 4 个关系模式:

商店 (商店编号, 商店名, 地址)

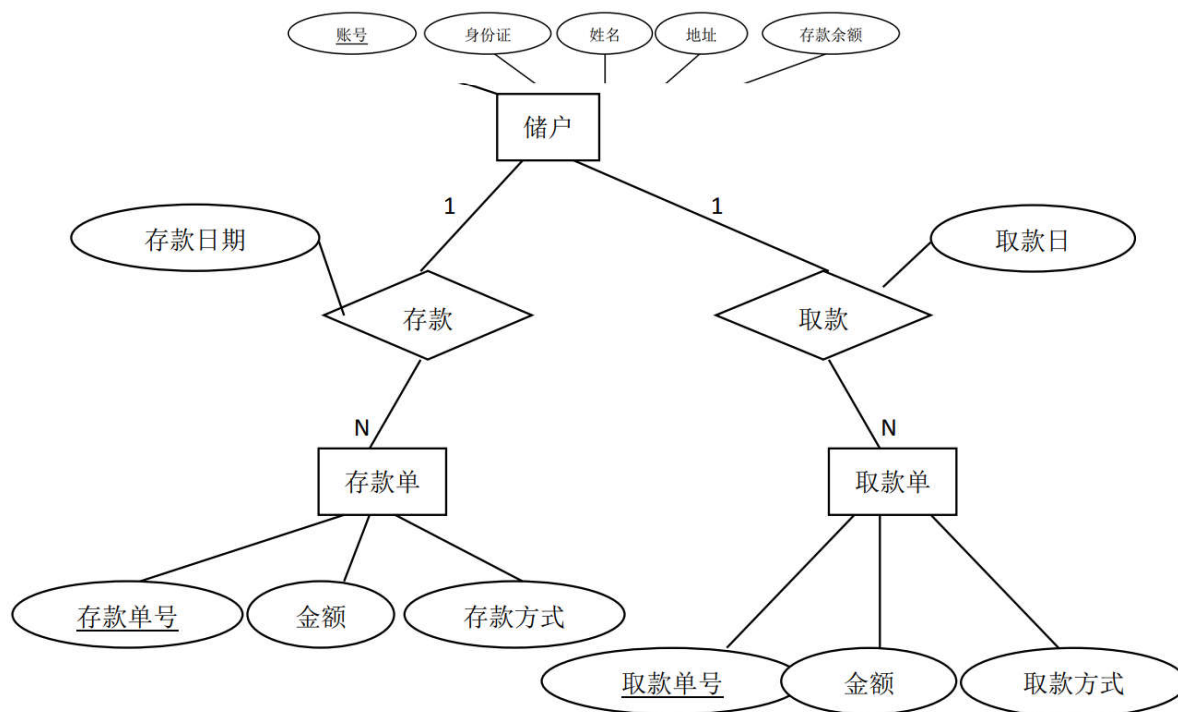
职工 (职工编号, 姓名, 性别, 业绩, 商店编号, 聘期, 月薪)

商品 (商品号, 商品名, 规格, 单价)

销售 (商店编号, 商品号, 月销售量)



## 6.4



• 6.6

