

数据库系统原理

第 2 章 关系数据库

刘济全

浙江大学生物医学工程与仪器科学学院

E-mail: liujq@zju.edu.cn

个人主页 : <http://person.zju.edu.cn/liujiquan>

第 2 章 关系数据库

- 2.0 关系数据库概述
- 2.1 关系数据的基本概念
- 2.2 关系模式
- 2.3 关系代数
- 2.5 关系演算

2.0 关系数据库概述

- 系统而严格地提出关系模型的是美国 IBM 公司的 E.F.Codd
 - 1970 年提出关系数据模型
 - E.F.Codd, “A Relational Model of Data for Large Shared Data Banks” , 《 Communication of the ACM 》 ,1970
 - 之后 , 提出了关系代数和关系演算的概念
 - 1972 年提出了关系的第一、第二、第三范式
 - 1974 年提出了关系的 BC 范式

2.0 关系数据库概述

- 关系数据库应用数学方法来处理数据库中的数据
- 80 年代后，关系数据库系统成为最重要、最流行的数据库系统

2.0 关系数据库概述

- 典型实验系统
 - System R
 - University INGRES
- 典型商用系统
 - ORACLE
 - MySQL
 - MariaDB
 - DB2
 - SQL SERVER

2.1 关系数据库的基本概念

1、关系及其性质

(1) 域

定义 2.1: 域是一组具有相同数据类型的值的集合。

- 在关系中用域来表示属性的取值范围
- 域中所包含的值的个数称**域的基数** (用 m 表示)

例: $D_1 = \{A, 2, 3, \dots, Q, K\}$ $M_1 = 13$

$D_2 = \{ \text{数据库原理}, \text{面向对象数据库技术} \}$ $M_2 =$

2.1 关系数据库的基本概念

(2) 笛卡尔积

定义 2.2: 给定一组域 D_1, \dots, D_n
(可有相同的域)。其笛卡尔积为：
 $D_1 \times D_2 \times \dots \times D_n = \{ \underline{(d_1, d_2, \dots, d_n)} \mid d_i \in D_i, i=1, 2, \dots, n \}$

d_i 为分量

n 元组

笛卡儿积也是一个集合

其中每一个元素 (d_1, d_2, \dots, d_n) 叫作一个 n 元组 (n -Tuple),
或简称为元组。元素中的每一个值 d_i 叫作一个分量 (Component)

若 D_i ($i = 1, 2, \dots, n$) 为有限集, 其基数 (Cardinal number) 为 m_i ($i = 1, 2, \dots, n$), 则 $D_1 \times D_2 \times \dots \times D_n$ 的基
 $m = \prod_{i=1}^n m_i$

2.1 关系数据库的基本概念

(3) 关系

定义 2.3 $D_1 \times D_2 \times \dots \times D_n$ 的有意义的子集称为在域 D_1, D_2, \dots, D_n 上的关系

其中： R 为关系的名， n 为关系的度（目）； $r \in R$ 表示 r 是 R 中的元组

- 子集元素是关系中的元组；
- 关系中的元组个数是关系的基数；
- 同样可以把关系看作是一个二维表：
 - 每一行对应一个元组；
 - 表的每一列对应一个域，每个域起一个名字——称为属性

2.1 关系数据库的基本概念

例：设 $D_1 = \text{男人集合 (MAN)} = \{ \text{王强、李东、张兵} \}$

$D_2 = \text{女人集合 (WOMAN)} = \{ \text{赵红、吴芳} \}$

$D_3 = \text{儿童集合 (CHILD)} = \{ \text{王一、李一、李二} \}$

(2) 构造面全家属关系函数表为：

FAMILY (MAN , WOMAN , CHILD

D Family

MAN	WOMAN	CHILD
王强	赵红	王一
李东	吴芳	李一
李东	吴芳	李二

M	W	C
王强	赵红	王一
王强	赵红	李一
王强	赵红	李二
王强	吴芳	王一
王强	吴芳	李一
王强	吴芳	李二
李东	赵红	王一
李东	赵红	李一
李东	赵红	李二
李东	吴芳	王一
李东	吴芳	李一
李东	吴芳	李二
张兵	赵红	王一
张兵	赵红	李一
张兵	赵红	李二
张兵	吴芳	王一
张兵	吴芳	李一
张兵	吴芳	李二

2.1 关系数据库的基本概念

候选码 (candidate key)

值能唯一标识一个元组的属性组，且不含多余属性，称该属性组为候选码。

主码 (primary key)

一个关系有多个候选码时，选定其中的一个作为主码。

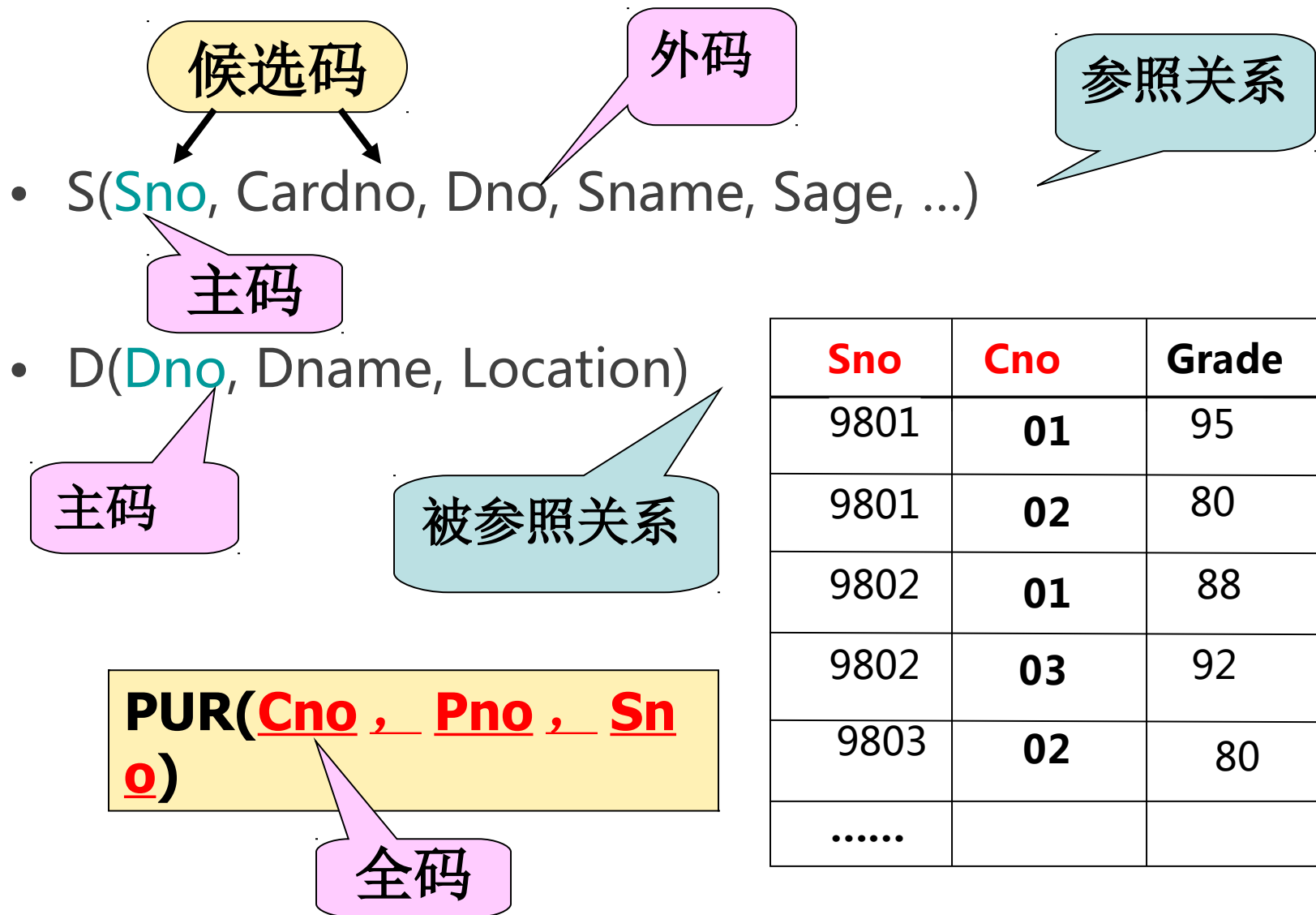
外码 (foreign key)

关系 R 的某一属性组 X 不是 R 的码，但是是其它某一关系的码，称 X 为 R 的外码。

全码 (all key)

若关系 R 整个属性的集合是 R 的候选码，则该候选码为全码。

2.1 关系数据库的基本概念



2.1 关系数据库的基本概念

(4) 三类关系

基本关系 (基本表或基表)

实际存在的表，是实际存储数据的逻辑表示

查询表

查询结果对应的表

视图表

由基本表或其他视图表导出的表，是虚表，不对应实际存储的数据

2.1 关系数据库的基本概念

(5) 关系的性质

- ① 每列的值为同一类型。
- ② 每列具有不同的属性名(可同域)
- ③ 任意两元组不能完全相同。
- ④ 行的次序可以互换。
- ⑤ 列的次序可以互换。
- ⑥ 分量值是原子的。

属性名			分量值
学号	姓名	年龄	
9801	黄林	19	元组
9802	李红	20	
9803	张英	21	
.....	
9830	王刚	20	
	、网虫	+5	不允许

2.2 关系模式

1、关系模式

关系模式 (Relation Schema) 是型

关系模式是对关系的描述

- 元组集合的结构

 - 属性构成

 - 属性来自的域

 - 属性与域之间的映象关系

- 元组语义以及完整性约束条件

- 属性间的数据依赖关系集合

2.2 关系模式

定义 2.4 :

关系的描述称关系模式，其表示为： $R(U, D, dom, F)$

域名集

属性间的依赖
关系集

属性名集

属性向域的映像集

- 关系模式可简记为关系的属性名表。

$R(U) = R(A_1, A_2, A_3, \dots, A_n)$

例：学生（学号，姓名，总成绩）

- 关系模式就是关系的框架（表框架）
- 它是对关系结构的描述

2.2 关系模式

关系模式可以形式化地表示为：

$R (U , D , \text{dom} , F)$

R 关系名

U 组成该关系的属性名集合

D 属性组 U 中属性所来自的域

dom 属性向域的映象集合

F 属性间的数据依赖关系集合

是关系结构的描述和定义，即二维表的表结构定义。

关系实质上是一张二维表。因此，关系模式必须指出表的结构，即它由哪些属性构成，这些属性来自哪些域，以及属性与域之间的对应关系。

2.2 关系模式

例：

导师和研究生出自同一个域——人，

取不同的属性名，并在模式中定义属性向域的映象
，即说明它们分别出自哪个域：

$$\begin{aligned} & \text{dom (SUPERVISOR-PERSON)} \\ &= \text{dom (POSTGRADUATE-PERSON)} \\ &= \text{PERSON} \end{aligned}$$

2.2 关系模式

关系模式通常可以简记为

$R(U)$ 或 $R(A_1, A_2, \dots, A_n)$

R 关系名

A_1, A_2, \dots, A_n 属性名

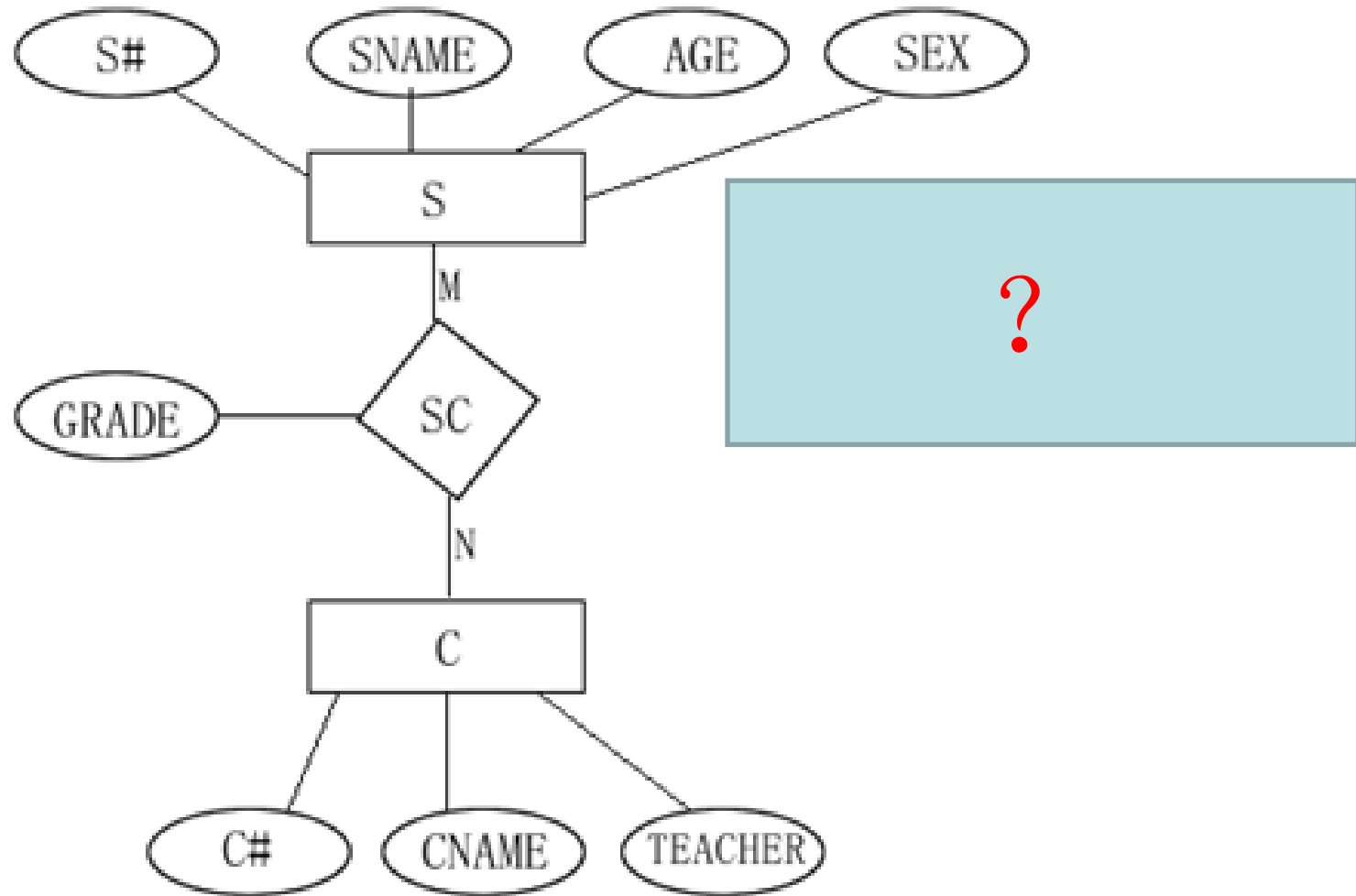
注：域名及属性向域的映象常常直接说明为属性的类型、长度

2.2 关系模式

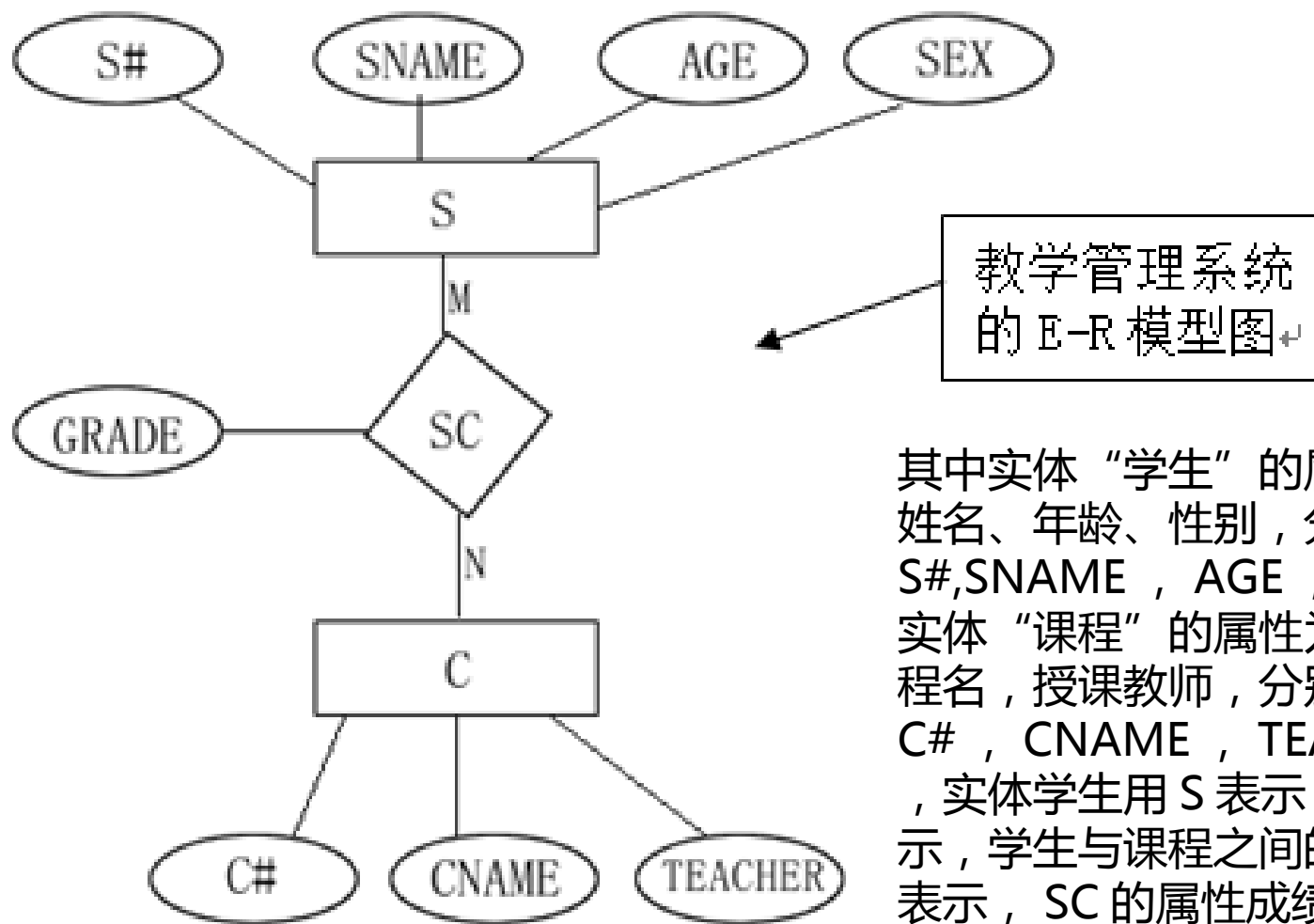
2、关系数据库

- 在关系模型中，实体以及实体间的联系都是用关系来表示。在一个给定的现实世界领域中，相应于所有实体及实体之间的联系的**关系的集合**构成一个关系数据库。
- 关系数据库也有型和值之分。关系数据库的型也称为**关系数据库模式**，是对关系数据库的描述，是关系模式的集合。关系数据库的值也称为**关系数据库**，是关系的集合。关系数据库模式与关系数据库通常统称为关系数据库。

参考习题

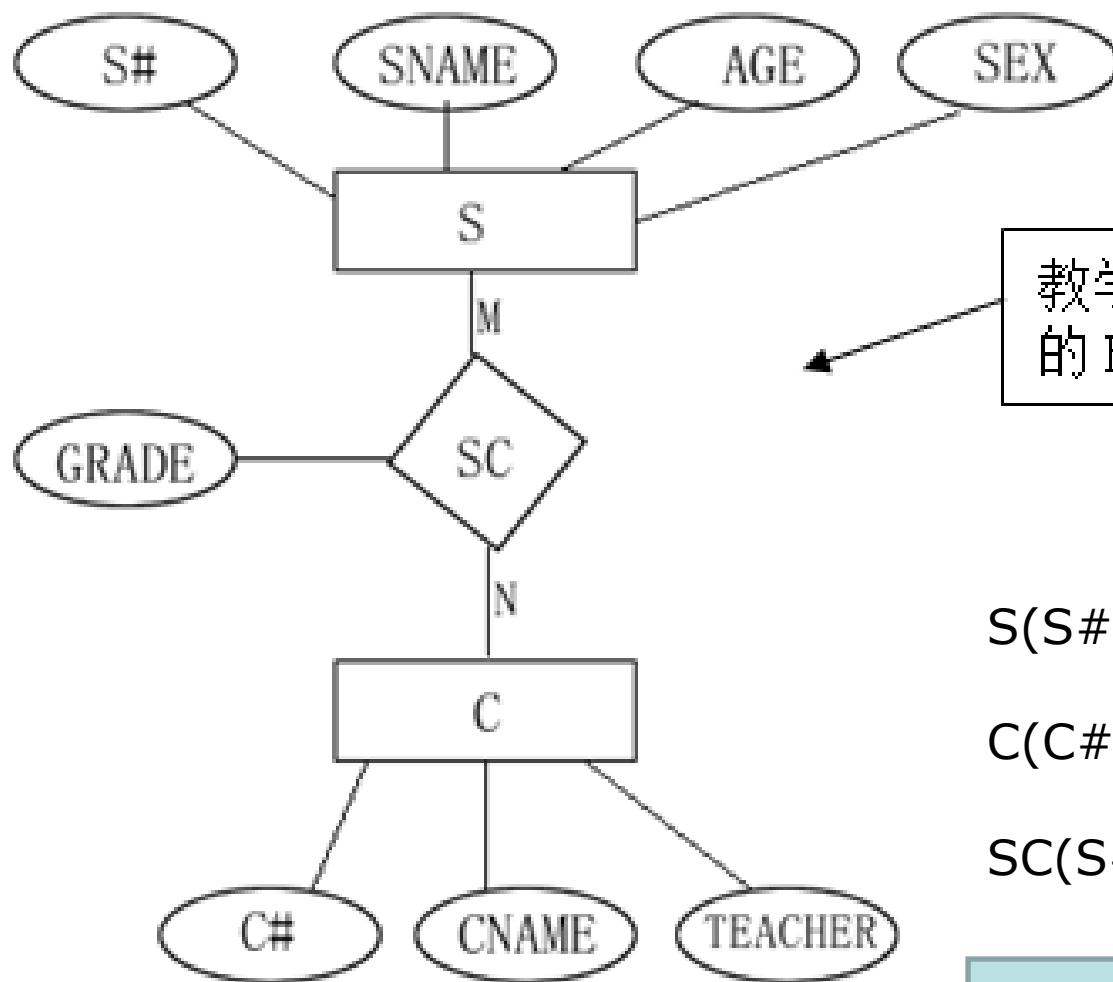


参考习题



其中实体“学生”的属性为学号，姓名、年龄、性别，分别用 S#、SNAME、AGE、SEX 表示，实体“课程”的属性为课程号，课程名，授课教师，分别用 C#、CNAME、TEACHER 表示，实体学生用 S 表示，课程用 C 表示，学生与课程之间的关系用 SC 表示，SC 的属性成绩用 GRADE 表示

参考习题



教学管理系统的
E-R模型图

S(S#,SNAME,AGE,SEX)

C(C#,CNAME,TEACHER)

SC(S#,C#,GRADE)

关键字?

2.2 关系模式

3、关系的完整性

- 三类完整性约束：

- 实体完整性
- 参照完整性
- 用户定义的完整性

不变性
由关系系统自动支持

是应用领域需要遵循的约束条件

- 说明：

(1) 实体完整性

- 规则 2.1 **实体完整性规则**：
- 若属性 A 是基本关系 R 的主码属性，则属性 A 不能取空值。

① 实体完整性规则是对基本关系的约束和限定。

② 实体具有唯一性标识——主码。

③ 主码属性不能取空值。

2.2 关系模式

(2) 参照完整性

- 引用关系：

- ❖ 关系中的某属性的值需要参照另一关系的属性来取值。

例 1：学生（学号，姓名，性别，专业号，年龄）

专业（专业号，专业名）



引用

例 2：学生（学号，姓名，性别，专业号，年龄，班长）



引用

2.2 关系模式

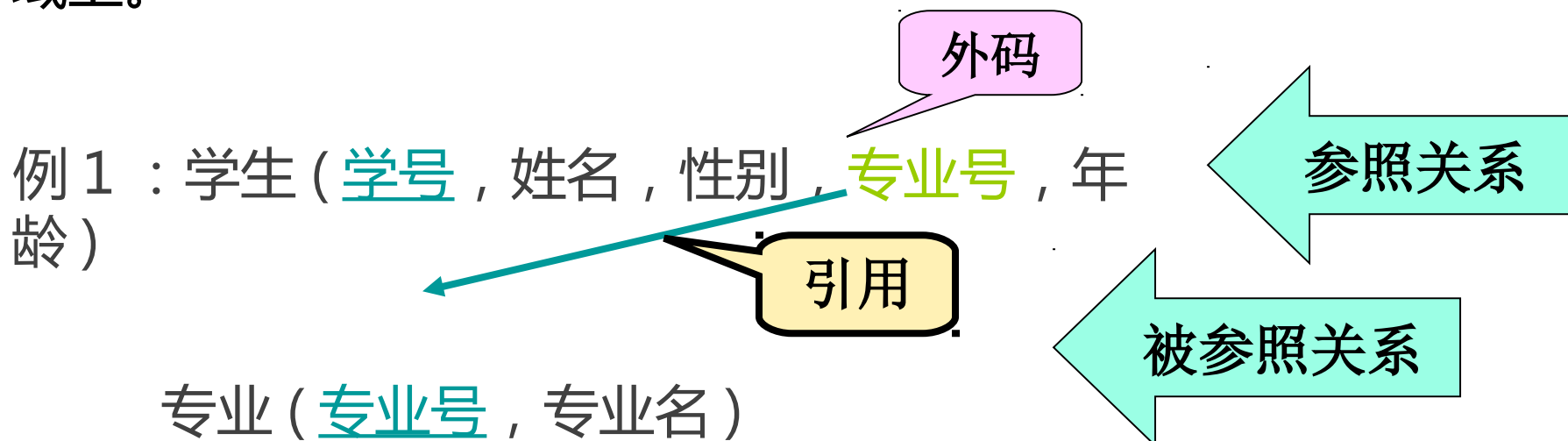
定义 2.5 : 设基本关系 R 、 S (可为同一关系)。

若 F 是 R 的一个 (组) 属性, 但不是 R 的码。

如果 F 与 S 的主码 K 相对应, 则称 F 是 R 的外码。

并称 R 为参照关系, S 为被参照关系 (目标关系)。

- 说明: S 的主码 K 和 R 的外码 F 必须定义在同一个 (或一组) 域上。



- 参照完整性规则定义了外码与主码之间的引用规则。

2.2 关系模式

规则 2.2 参照完整性规则

- 若属性 (组) F 是 R 的外码它与 S 的主码 K 相对应 , 则对于 R 中每个元组在 F 上的值必须为 :
 - 或者取空值 (F 的每个属性值均为空值) ;
 - 或者等于 S 中某个元组的主码值。

例 1 : 学生 (学号 , 姓名 , 性别 , 专业号 , 年龄)

关系中每个元组的专业号取值 :

- ① 空值 (未给该学生分配专业) ;
- ② 非空值 (是专业关系中某个元组的专业号值) 。

例 2 : 职工 EMP(EMP#,ENAME,JOB,DEPT#)

部门 DEPT(DEPT#,DNAME,LOC)

则 : EMP 中的 DEPT# 为空或为 DEPT 中的 DEPT# 的值

2.2 关系模式

(3) 用户定义的完整性

用户自定义完整性是针对某一具体数据的约束条件，反映某一具体应用所涉及的数据必须满足的语义要求，由应用环境决定。

例： 属性的取值范围 属性的非空限制

2.2 关系模式

(3) 用户定义的完整性

例：

课程 (课程号 , 课程名 , 学分)

- “课程名” 属性必须取唯一值
- 非主属性 “课程名” 也不能取空值
- “学分” 属性只能取值 {1 , 2 , 3 , 4}

参考习题

1. 关于关系模型，下面叙述不正确的是：

- (1) 一个关系至少要有有一个候选码 (2) 一个列的值可以来自不同的域
(3) 行的次序可以任意交换 (4) 列的次序可以任意交换

2. 下列说法正确的是：

- (1) 候选码都可以唯一标识一个元组 (2) 主属性可以取空值
(3) 候选码只能包含一个属性 (4) 关系的外码不可以取空值

3. 候选关键字的属性可以有 ()：

- (1) 多个 (2) 0 个
(3) 1 个 (4) 1 个或多个

参考习题

假设存在一张患者表，包含“性别”属性，要求这个属性的值只能取“男”或者“女”，这属于：

(1)实体完整性 (2) 参照完整性 (3) 用户定义的完整性 (4) 关系不变性

学生（学生号，姓名，年龄，学籍）和院系（院系代码，院系名称，院系地址）存在引用关系，其中_____是参照关系，_____是外码

2.2 关系模式

4、关系的数据操作

- 常用的关系操作

- 查询

- 选择、投影、连接、除、并、交、差

- 数据更新

- 插入、删除、修改

- 查询的表达能力是其中最主要的部分

- 关系操作的特点

- 集合操作方式，即操作的对象和结果都是集合。

- 非关系数据模型的数据操作方式：一次一记录
 - 文件系统的数据操作方式

2.2 关系模式

5、关系数据语言：分类

- 关系代数语言
 - 用对关系的运算来表达查询要求，典型代表：ISBL
- 关系演算语言：用谓词来表达查询要求
 - 元组关系演算语言
 - 谓词变元的基本对象是元组变量，典型代表：APLHA, QUEL
 - 域关系演算语言
 - 谓词变元的基本对象是域变量，典型代表：QBE
- 具有关系代数和关系演算双重特点的语言
 - 典型代表：SQL （ Structured Query Language ）

2.2 关系模式

5、关系数据语言：特点

- 关系语言是一种高度非过程化的语言
 - 存取路径的选择由 DBMS 的优化机制来完成
 - 用户不必用循环结构就可以完成数据操作
- 能够嵌入高级语言中使用
- 关系代数、元组关系演算和域关系演算三种语言在表达能力上完全等价

2.3 关系代数


2.3.1 传统的集合运算

2.3.2 专门的关系运算

2.3 关系代数

关系代数是一种抽象的查询语言。它以关系为运算对象，

通过对关系进行“组合”或“分割”，得到所需的数据集合——关系。

- 分类：集合运算（并、交、差；广义笛卡尔积）
关系运算（投影、选择、连接和除运算）

2.3 关系代数

关系代数运算的分类

传统的集合运算

并、差、交、广义笛卡尔积

专门的关系运算

选择、投影、连接、除

2.3 关系代数

– 集合运算符

- 将关系看成元组的集合
- 运算是从关系的“水平”方向即行的角度来进行

– 专门的关系运算符

- 不仅涉及行而且涉及列

– 算术比较符

- 辅助专门的关系运算符进行操作

– 逻辑运算符

- 辅助专门的关系运算符进行操作

2.3 关系代数

表 关系代数运算符

运算符		含义	运算符		含义
集合运算符	\cup	并	比较运算符	$>$	大于
	$-$	差		\geq	大于等于
	\cap	交		$<$	小于
	\times	笛卡儿积		\leq	小于等于
		广义笛卡儿积		$=$	等于
				\neq	不等于

2.3 关系代数

表 关系代数运算符（续）

运算符	含义		运算符	含义	
专门的关系运算符	σ π \bowtie \div	选择 投影 连接 除	逻辑运算符	\neg \wedge \vee	非 与 或

2.3.1 传统的集合运算

- 并
- 差
- 交
- 广义笛卡尔积

2.3.1 传统的集合运算

1. 并 (Union)

- R 和 S
 - 具有相同的目 n (即两个关系都有 n 个属性)
 - 相应的属性取自同一个域
- $R \cup S$
 - 仍为 n 目关系, 由属于 R 或属于 S 的元组组成
$$R \cup S = \{ t \mid t \in R \vee t \in S \}$$

2.3.1 传统的集合运算

1. 并 (Union)

R

A	B	C
a1	b1	c1
a1	b2	c2
a2	b2	c1

s

A	B	C
a1	b2	c2
a1	b3	c2
a2	b2	c1

R ∪ S

A	B	C
a1	b1	c1
a1	b2	c2
a1	b3	c2
a2	b2	c1

2.3.1 传统的集合运算

2. 差 (Difference)

- R 和 S
 - 具有相同的目 n
 - 相应的属性取自同一个域
- $R - S$
 - 仍为 n 目关系，由属于 R 而不属于 S 的所有元组组成

$$R - S = \{ t \mid t \in R \wedge t \notin S \}$$

2.3.1 传统的集合运算

2. 差 (Difference)

<i>R</i>	A	B	C
	a1	b1	c1
	a1	b2	c2
	a2	b2	c1

<i>s</i>	A	B	C
	a1	b2	c2
	a1	b3	c2
	a2	b2	c1

R-S

A	B	C
a1	b1	c1

2.3.1 传统的集合运算

3. 交（Intersection）

- R 和 S
 - 具有相同的目 n
 - 相应的属性取自同一个域
- $R \cap S$
 - 仍为 n 目关系，由既属于 R 又属于 S 的元组组成

$$R \cap S = \{ t \mid t \in R \wedge t \in S \}$$

$$R \cap S = R - (R - S)$$

2.3.1 传统的集合运算

3. 交 (Intersection)

R

A	B	C
a1	b1	c1
a1	b2	c2
a2	b2	c1

s

A	B	C
a1	b2	c2
a1	b3	c2
a2	b2	c1

$R \cap s$

A	B	C
a1	b2	c2
a2	b2	c1

2.3.1 传统的集合运算

4. 广义笛卡尔积 (Extended Cartesian Product)

- R
 - n 目关系, k_1 个元组
- S
 - m 目关系, k_2 个元组
- $R \times S$
 - 列: ($n+m$) 列的元组的集合
 - 元组的前 n 列是关系 R 的一个元组
 - 后 m 列是关系 S 的一个元组
 - 行: $k_1 \times k_2$ 个元组

$$R \times S = \{ \overbrace{tr \ ts} \mid tr \in R \wedge ts \in S \}$$

2.3.2 专门的关系运算

– 记号： $\widehat{t_r t_s}$

R 为 n 目关系， S 为 m 目关系。 t_r

$\in R$ ， $t_s \in S$ ， $\widehat{t_r t_s}$ 称为元组的连接。它是一个 $n + m$ 列的元组，前 n 个分量为 R 中的一个 n 元组，后 m 个分量为 S 中的一个 m 元组。

2.3.2 专门的关系运算

m 元关系

■ 设 :R、S 为不同类关系，则结果为不同类关系：

■ $R \times S = \{tr \ ts \mid (tr \in R) \wedge (ts \in S)\}$ $R \times S$

连接为 m+n 元关系

n 元关系

R

A_1	A_2	A_3
b	2	d
b	3	b
c	2	d
d	3	b

S

A_2	A_3
2	d
3	b

$R.A_1$	$R.A_2$	$R.A_3$	$S.A_2$	$S.A_3$
b	2	d	2	d
b	2	d	3	b
b	3	b	2	d
b	3	b	3	b
c	2	d	2	d
c	2	d	3	b
d	3	b	2	d
d	3	b	3	b

2.3.1 传统的集合运算

R

A	B	C
a1	b1	c1
a1	b2	c2
a2	b2	c1

S

A	B	C
a1	b2	c2
a1	b3	c2
a2	b2	c1

$R \times S$

A	B	C	A	B	C
$a1$	$b1$	$c1$	$a1$	$b2$	$c2$
$a1$	$b1$	$c1$	$a1$	$b3$	$c2$
$a1$	$b1$	$c1$	$a2$	$b2$	$c1$
$a1$	$b2$	$c2$	$a1$	$b2$	$c2$
$a1$	$b2$	$c2$	$a1$	$b3$	$c2$
$a1$	$b2$	$c2$	$a2$	$b2$	$c1$
$a2$	$b2$	$c1$	$a1$	$b2$	$c2$
$a2$	$b2$	$c1$	$a1$	$b3$	$c2$
$a2$	$b2$	$c1$	$a2$	$b2$	$c1$

2.3.2 专门的关系运算

表示记号

(1) R , $t \in R$, $t[A_i]$

设关系模式为 $R(A_1, A_2, \dots, A_n)$

它的一个关系设为 R 。 $t \in R$ 表示 t 是 R 的一个元组

$t[A_i]$ 则表示元组 t 中相应于属性 A_i 的一个分量

2.3.2 专门的关系运算

(2) A , $t[A]$

若 $A = \{A_{i_1}, A_{i_2}, \dots, A_{i_k}\}$, 其中 $A_{i_1}, A_{i_2}, \dots, A_{i_k}$ 是 A_1, A_2, \dots, A_n 中的一部分, 则 A 称为属性列或域列。

$t[A] = (t[A_{i_1}], t[A_{i_2}], \dots, t[A_{i_k}])$ 表示元组 t 在属性列 A 上诸分量的集合。

2.3.2 专门的关系运算

- 选择
- 投影
- 连接
- 除

2.3.2 专门的关系运算

1. 选择 (Selection)

- 1) 选择又称为限制 (Restriction)
- 2) 选择运算符的含义
 - 在关系 R 中选择满足给定条件的诸元组

$$\sigma_F(R) = \{t \mid t \in R \wedge F(t) = \text{'真'}\}$$

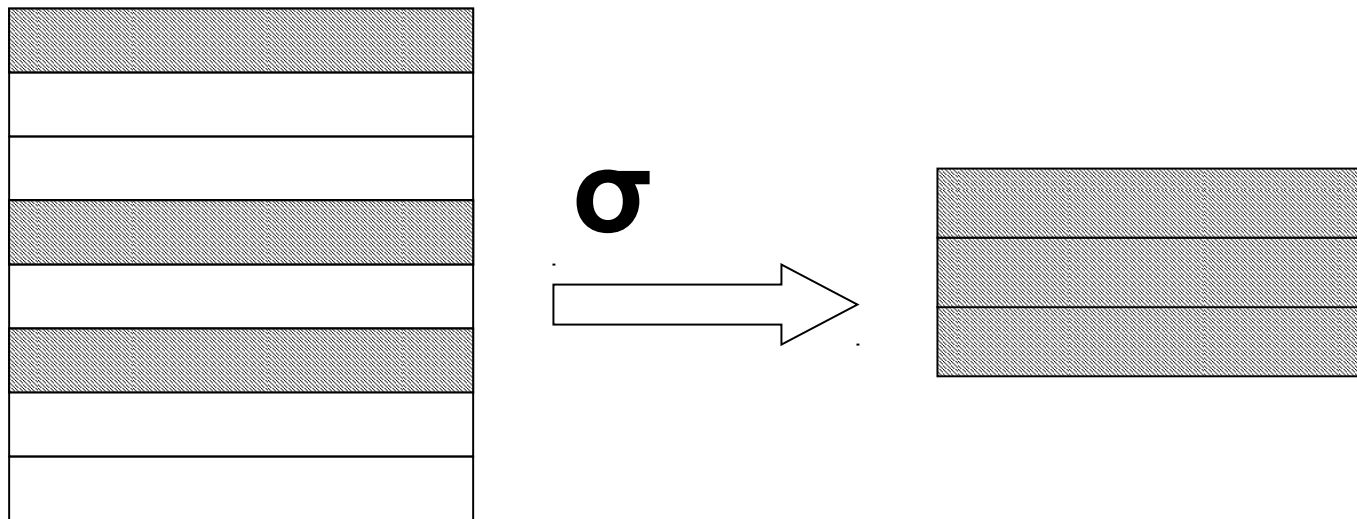
- F : 选择条件, 是一个逻辑表达式, 基本形式为 :

$$[\neg (\text{ } X_1 \theta Y_1 \text{ })][\varphi [\neg (\text{ } X_2 \theta Y_2 \text{ })]]...$$

2.3.2 专门的关系运算

1. 选择（ Selection ）

- 3) 选择运算是从行的角度进行的运算



2.3.2 专门的关系运算

1. 选择（ Selection ）

学 号 Sno	姓 名 Sname	性 别 Ssex	年 龄 Sage	系 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

Student

(a)

2.3.2 专门的关系运算

1. 选择（ Selection ）

课程号	课程名	先行课	学分
Cno	Cname	Cpno	Ccredit
1	数据库	5	4
2	数学		2
3	信息系统	1	4
4	操作系统	6	3
5	数据结构	7	4
6	数据处理		2
7	C 语言	6	4

Course

(b)

2.3.2 专门的关系运算

1. 选择（ Selection ）

学 号	课 程 号	成 绩
Sno	Cno	Grade
95001	1	92
95001	2	85
95001	3	88
95002	2	90
95002	3	80

SC

(c)

2.3.2 专门的关系运算

1. 选择（ Selection ）

例 1 查询仪器系（ IS 系 ）全体学生

$$\sigma_{Sdept = 'IS'}(Student)$$

结果：

Sno	Sname	Ssex	Sage	Sdept
95002	刘晨	女	19	IS
95004	张立	男	19	IS

2.3.2 专门的关系运算

1. 选择（ Selection ）

例 2 查询年龄小于 20 岁的学生

$\sigma_{\text{Sage} < 20}(\text{Student})$

结果：

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

2.3.2 专门的关系运算

2. 投影（ Projection ）

- 1) 投影运算符的含义
 - 从 R 中选择出若干属性列组成新的关系

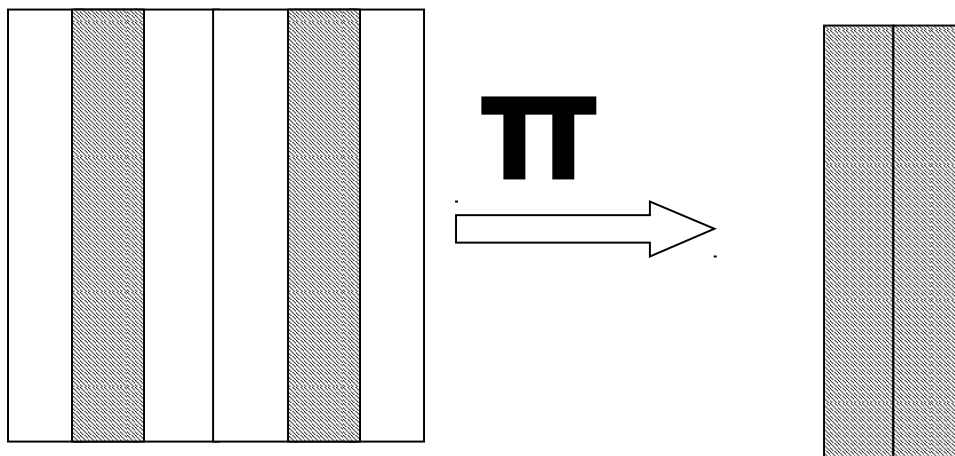
$$\pi_A(R) = \{ t[A] \mid t \in R \}$$

A : R 中的属性列

2.3.2 专门的关系运算

2. 投影（Projection）

- 2) 投影操作主要是从列的角度进行运算



- 但投影之后不仅取消了原关系中的某些列，而且还可能取消某些元组（避免重复行）

2.3.2 专门的关系运算

2. 投影 (Projection)

- 3) 举例

[例 3] 查询学生的姓名和所在系

即求 Student 关系上学生姓名和所在系两个属性上的投影

■ $\pi_{\text{Sname}, \text{Sdept}}(\text{Student})$

2.3.2 专门的关系运算

2. 投影（Projection）

Sname	Sdept
李勇	CS
刘晨	IS
王敏	MA
张立	IS

2.3.2 专门的关系运算

[例 4] 查询学生关系 Student 中都有哪些系

$\pi_{\text{Sdept}}(\text{Student})$

结果：

Sdept
CS
IS
MA

2.3.2 专门的关系运算

3. 连接（Join）

- 1) 连接也称为 θ 连接
- 2) 连接运算的含义
 - 从两个关系的笛卡尔积中选取属性间满足一定条件的元组

$$R \bowtie_{A\theta B} S = \{ \widehat{t_r t_s} \mid tr \in R \wedge ts \in S \wedge tr[A] \theta ts[B] \}$$

- A 和 B : 分别为 R 和 S 上度数相等且可比的属性组
- θ : 比较运算符
- 连接运算从 R 和 S 的广义笛卡尔积 $R \times S$ 中选取 (R 关系) 在 A 属性组上的值与 (S 关系) 在 B 属性组上值满足比较关系的元组。

2.3.2 专门的关系运算

3. 连接（Join）

- 3) 两类常用连接运算

- 等值连接（equijoin）

- 什么是等值连接

- θ 为 “=” 的连接运算称为等值连接

- 等值连接的含义

- 从关系 R 与 S 的广义笛卡尔积中选取 A 、 B 属性值相等的那些元组，即等值连接为：

$$R \bowtie_{A=B} S = \{ \overset{\cdot}{t_r} \overset{\cdot}{t_s} \mid tr \in R \wedge ts \in S \wedge tr[A] = ts[B] \}$$

2.3.2 专门的关系运算

3. 连接（Join）

– 自然连接（Natural join）

- 什么是自然连接

- 自然连接是一种特殊的等值连接

- » 两个关系中进行比较的分量必须是相同的属性组

- » 在结果中把重复的属性列去掉

- 自然连接的含义

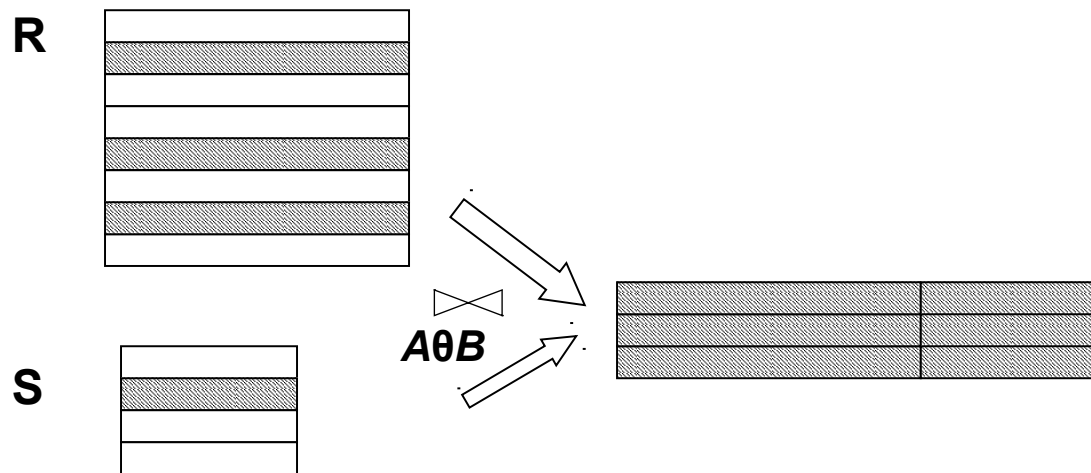
R 和 S 具有相同的属性组 B

$$R \bowtie S = \{ \widehat{t_r t_s} \mid t_r \in R \wedge t_s \in S \wedge t_r[B] = t_s[B] \}$$

2.3.2 专门的关系运算

3. 连接（Join）

- 4) 一般的连接操作是从行的角度进行运算。



自然连接还需要取消重复列，所以是同时从行和列的角度进行运算

2.3.2 专门的关系运算


连接示例

<i>A</i>	<i>B</i>	<i>C</i>
<i>a</i> ₁	<i>b</i> ₁	5
<i>a</i> ₁	<i>b</i> ₂	6
<i>a</i> ₂	<i>b</i> ₃	8
<i>a</i> ₂	<i>b</i> ₄	12

R

<i>B</i>	<i>E</i>
<i>b</i> ₁	3
<i>b</i> ₂	7
<i>b</i> ₃	10
<i>b</i> ₃	2
<i>b</i> ₅	2

S

R  *S*
C < *E*

2.3.2 专门的关系运算

连接示例

$$R \bowtie_{C < E} S$$

<i>A</i>	<i>R.B</i>	<i>C</i>	<i>S.B</i>	<i>E</i>
<i>a</i> ₁	<i>b</i> ₁	5	<i>b</i> ₂	7
<i>a</i> ₁	<i>b</i> ₁	5	<i>b</i> ₃	10
<i>a</i> ₁	<i>b</i> ₂	6	<i>b</i> ₂	7
<i>a</i> ₁	<i>b</i> ₂	6	<i>b</i> ₃	10
<i>a</i> ₂	<i>b</i> ₃	8	<i>b</i> ₃	10

2.3.2 专门的关系运算

等值连接示例

<i>A</i>	<i>B</i>	<i>C</i>
<i>a</i> ₁	<i>b</i> ₁	5
<i>a</i> ₁	<i>b</i> ₂	6
<i>a</i> ₂	<i>b</i> ₃	8
<i>a</i> ₂	<i>b</i> ₄	12

R

<i>B</i>	<i>E</i>
<i>b</i> ₁	3
<i>b</i> ₂	7
<i>b</i> ₃	10
<i>b</i> ₃	2
<i>b</i> ₅	2

S

R  *S*
R.B=S.B

2.3.2 专门的关系运算

等值连接示例

$$R \bowtie S$$

$R.B=S.B$

<i>A</i>	<i>R.B</i>	<i>C</i>	<i>S.B</i>	<i>E</i>
<i>a</i> ₁	<i>b</i> ₁	5	<i>b</i> ₁	3
<i>a</i> ₁	<i>b</i> ₂	6	<i>b</i> ₂	7
<i>a</i> ₂	<i>b</i> ₃	8	<i>b</i> ₃	10
<i>a</i> ₂	<i>b</i> ₃	8	<i>b</i> ₃	2

2.4.2 专门的关系运算

自然连接 

S

<i>A</i>	<i>B</i>	<i>C</i>	<i>E</i>
<i>a</i> ₁	<i>b</i> ₁	5	3
<i>a</i> ₁	<i>b</i> ₂	6	7
<i>a</i> ₂	<i>b</i> ₃	8	10
<i>a</i> ₂	<i>b</i> ₃	8	2

2.3.2 专门的关系运算

4) 象集 Z

给定一个关系 $R (X , Z)$, X 和 Z 为属性组。当 $t[X]=x$ 时 , x 在 R 中的象集 (Images Set) 为 :

$$Z_x = \{ t[Z] | t \in R , t[X]=x \}$$

它表示 R 中属性组 X 上值为 x 的诸元组在 Z 上分量的集合。

2.3.2 专门的关系运算

象集示例

R

<i>A</i>	<i>B</i>	<i>C</i>
<i>a</i>₁	<i>b</i>₁	<i>c</i>₂
<i>a</i>₂	<i>b</i>₃	<i>c</i>₇
<i>a</i>₃	<i>b</i>₄	<i>c</i>₆
<i>a</i>₁	<i>b</i>₂	<i>c</i>₃
<i>a</i>₄	<i>b</i>₆	<i>c</i>₆
<i>a</i>₂	<i>b</i>₂	<i>c</i>₃
<i>a</i>₁	<i>b</i>₂	<i>c</i>₁

(a_1, b_1) 象集

<i>C</i>
<i>c</i>₂

(a_1, b_2) 象集

<i>C</i>
<i>c</i>₃
<i>c</i>₁

(a_1) 象集

<i>B</i>	<i>C</i>
<i>b</i>₁	<i>c</i>₂
<i>b</i>₂	<i>c</i>₃
<i>b</i>₂	<i>c</i>₁

2.3.2 专门的关系运算

4. 除 (Division)

给定关系 $R(X, Y)$ 和 $S(Y, Z)$, 其中 X, Y, Z 为属性组。

R 中的 Y 与 S 中的 Y 可以有不同的属性名, 但必须出自相同的域集

R 与 S 的除运算得到一个新的关系 $P(X)$,

P 是 R 中满足下列条件的元组在 X 属性列上的投影 :

R 中元组在 X 上分量值 x 的象集 Y_x 包含 S 在 Y 上投影的集合。

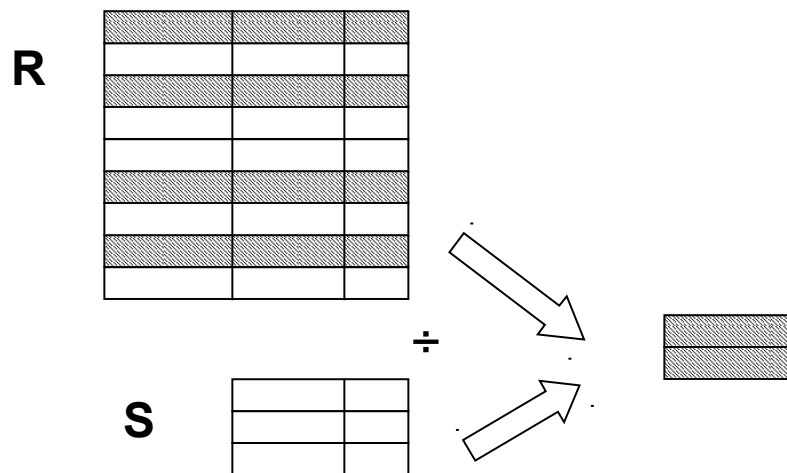
$$R \div S = \{t_r[X] \mid t_r \in R \wedge \pi_Y(S) \subseteq Y_x\}$$

Y_x : x 在 R 中的象集, $x = t[X]$

2.3.2 专门的关系运算

4. 除 (Division)

- 2) 除操作是同时从行和列角度进行运算



2.3.2 专门的关系运算

4. 除 (Division)

R

<i>A</i>	<i>B</i>	<i>C</i>
<i>a</i>₁	<i>b</i>₁	<i>c</i>₂
<i>a</i>₂	<i>b</i>₃	<i>c</i>₇
<i>a</i>₃	<i>b</i>₄	<i>c</i>₆
<i>a</i>₁	<i>b</i>₂	<i>c</i>₃
<i>a</i>₄	<i>b</i>₆	<i>c</i>₆
<i>a</i>₂	<i>b</i>₂	<i>c</i>₃
<i>a</i>₁	<i>b</i>₂	<i>c</i>₁

S

<i>B</i>	<i>C</i>	<i>D</i>
<i>b</i>₁	<i>c</i>₂	<i>d</i>₁
<i>b</i>₂	<i>c</i>₁	<i>d</i>₁
<i>b</i>₂	<i>c</i>₃	<i>d</i>₂

R* ÷ *S

2.3.2 专门的关系运算

分析：

在关系 R 中， A 可以取四个值 $\{a_1, a_2, a_3, a_4\}$

a_1 的象集为 $\{(b_1, c_2), (b_2, c_3), (b_2, c_1)\}$

a_2 的象集为 $\{(b_3, c_7), (b_2, c_3)\}$

a_3 的象集为 $\{(b_4, c_6)\}$

a_4 的象集为 $\{(b_6, c_6)\}$

S 在 (B, C) 上的投影为

$\{(b_1, c_2), (b_2, c_1), (b_2, c_3)\}$

只有 a_1 的象集包含了 S 在 (B, C) 属性组上的

2.3.2 专门的关系运算

5 . 综合举例

SC	学 号	课 程 号	成 绩
	Sno	Cno	Grade
	95001	1	92
	95001	2	85
	95001	3	88
	95002	2	90
	95002	3	80

查询至少选修 1 号课程和 3 号课程的学生号码

2.3.2 专门的关系运算

5 . 综合举例

首先建立一个临时关系 K :

Cno
1
3

然后求: $\Pi_{\text{Sno.Cno}}(\text{SC}) \div K$

2.3.2 专门的关系运算

5 . 综合举例

- $\pi_{\text{Sno.Cno}}(\text{SC})$

95001 象集 {1 , 2 , 3}

95002 象集 {2 , 3}

$\pi_{\text{Cno}}(K)=\{1 , 3\}$

于是： $\pi_{\text{Sno.Cno}}(\text{SC}) \div K = \{95001\}$

Sno	Cno
95001	1
95001	2
95001	3
95002	2
95002	3

2.3.2 专门的关系运算

5 . 综合举例

学 号 Sno	姓 名 Sname	性 别 Ssex	年 龄 Sage	系 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

Student

(a)

2.3.2 专门的关系运算

选择（续）

课程号	课程名	先行课	学分
Cno	Cname	Cpno	Ccredit
1	数据库	5	4
2	数学		2
3	信息系统	1	4
4	操作系统	6	3
5	数据结构	7	4
6	数据处理		2
7	C 语言	6	4

Course

(b)

2.3.2 专门的关系运算

选择（续）

学 号	课 程 号	成 绩
Sno	Cno	Grade
95001	1	92
95001	2	85
95001	3	88
95002	2	90
95002	3	80

SC

(c)

2.3.2 专门的关系运算

综合举例 (续)

查询选修了 **2** 号课程的学生们的学号

$$\Pi_{\text{Sno}} \left(\sigma_{\text{Cno}='2'} \left(\text{SC} \right) \right)$$
$$= \{ 95001, 95002 \}$$

2.3.2 专门的关系运算

综合举例 (续)

查询选修了一门其先行课为 **5** 号课程的课程的学生姓名。

2.3.2 专门的关系运算

综合举例

学 号 Sno	姓 名 Sname	性 别 Ssex	年 龄 Sage	系 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

Student

(a)

2.3.2 专门的关系运算

选择（续）

课程号	课程名	先行课	学分
Cno	Cname	Cpno	Ccredit
1	数据库	5	4
2	数学		2
3	信息系统	1	4
4	操作系统	6	3
5	数据结构	7	4
6	数据处理		2
7	C 语言	6	4

Course

(b)

2.3.2 专门的关系运算

选择（续）

学 号	课 程 号	成 绩
Sno	Cno	Grade
95001	1	92
95001	2	85
95001	3	88
95002	2	90
95002	3	80

SC

(c)

2.3.2 专门的关系运算

综合举例 (续)

$\pi_{\text{Sname}}(\sigma_{\text{Cpno}='5'}(\text{Course} \bowtie \text{SC} \bowtie \text{Student}))$

或

$\pi_{\text{Sname}}(\sigma_{\text{Cpno}='5'}(\text{Course}) \bowtie \text{SC} \bowtie \pi_{\text{Sno}, \text{Sname}}(\text{Student}))$

或

$\pi_{\text{Sname}}(\pi_{\text{Sno}}(\sigma_{\text{Cpno}='5'}(\text{Course}) \bowtie \text{SC}) \bowtie \pi_{\text{Sno}, \text{Sname}}(\text{Student}))$

2.3.2 专门的关系运算

综合举例 (续)

查询选修了全部课程的学生号码和姓名

2.3.2 专门的关系运算

综合举例 (续)

查询选修了全部课程的学生号码和姓名。

$\pi_{Sno, Cno} (SC) \div \pi_{Cno} (Course)$
 $\pi_{Sno, Sname} (Student)$

2.3 关系代数 (小结)

- 关系代数五种基本运算：
并、差、笛卡尔积、投影、选择
- 非基本运算用基本运算的表示

1. 交

$$R \cap S = R - (R - S)$$

2. 连接

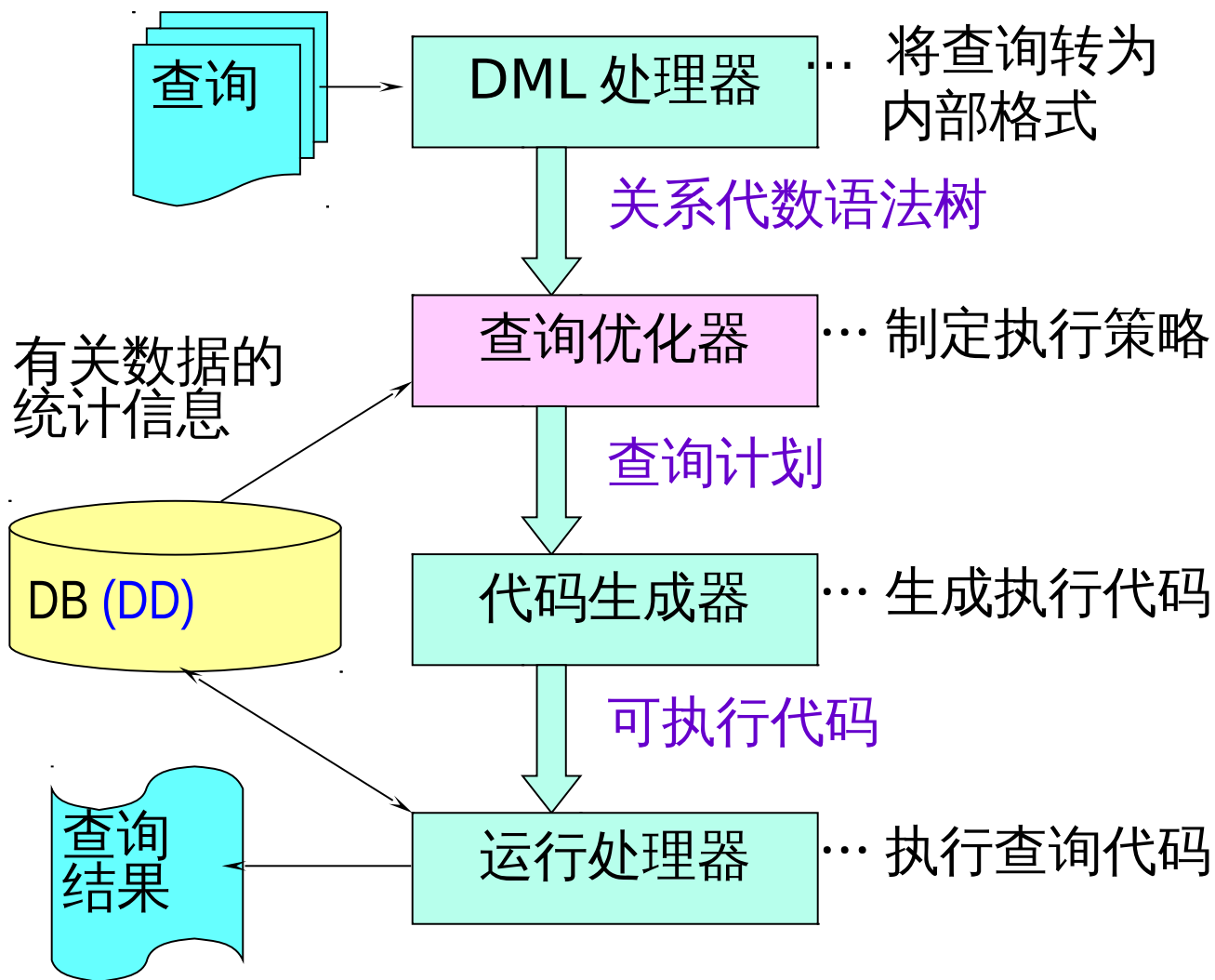
$$R \bowtie S = \sigma_{\substack{[i]\theta[m+j] \\ i\theta j}}(R \times S) \quad (\text{设 } R \text{ 为 } m \text{ 元、} S \text{ 为 } n \text{ 元关系})$$

$$\mathbf{R \bowtie S = \Pi_{i_1, i_2, \dots, i_m}(\sigma_{R.A_1=S.A_1 \dots R.A_k=S.A_k}(R \times S))}$$

其中：A1 , A2 , ... , Ak 为 R、S 的公共属性。

i1 , i2 , ... im 为从 R 与 S 的属性集中去掉
S.A1, S.A2 , ... S.Ak 后剩余的属性。

2.4.3 查询过程



2.4.3 查询过程

- 例：Student 表有 10^3 个学生记录，每人平均选 10 门课，SC 表共有 $1000 \times 10 = 10^4$ 个选课记录。要求：
查学生“王林”选课成绩在 85 分以上的课程号。
- SELECT cno
- FROM S , SC
- WHERE S.sno=SC.sno AND sname= ‘王林’ AND grade > 85 ;

条件

条件

条件

F1 等价的关系代数表示：

F2

F3

① $\prod_{\text{Cno}}(\sigma_{F1 \wedge F2 \wedge F3} (S \times SC))$

② $\prod_{\text{Cno}} (\sigma_{F2 \wedge F3} (S \bowtie SC))$

③ $\prod_{\text{Cno}}(\sigma_{F_2}(S) \times \sigma_{F_3}(SC))$

哪种效率高？

■连接的复杂度为:

① $10^3 \times 10^4 = O(10^7)$

② $10^3 \times 10 = O(10^4)$

③ $1 \times 10 = O(10^1)$

2.4.3 查询过程

- ① $\Pi_{Cno}(\sigma_{F1 \wedge F2 \wedge F3}(S \times SC))$
- ② $\Pi_{Cno}(\sigma_{F2 \wedge F3}(S \bowtie SC))$
- ③ $\Pi_{Cno}(\sigma_{F2}(S) \bowtie \sigma_{F3}(SC))$

■ 连接的复杂度为:

① $O(10^7)$

② $O(10^4)$

③ $O(10^1)$

- ❖ ① 先在两表上做 \times ，产生 $1000 \times 10000 = 10^7$ 个连接记录，再在其上进行先 σ 后 Π 操作。其基本运算的次数为: $10^7 + 3 \times 10^7$ 。
- ❖ ② 先在两个表上做 \bowtie ，产生 $1000 \times 10 = 10^4$ 个连接记录，再在其上进行先 σ 后 Π 操作。其基本运算的次数为: $10^4 + 2 \times 10^4$ 。
- ❖ ③ 先分别在两个表上做 σ ，再做 \bowtie ，产生 $1 \times 10 = 10$ 个连接记录，再在其上进行 Π 。其基本运算的次数为: $10^4 + 10^3 + 10^1$ 。

课堂作业

学 生 关 系 S(snum,sname,age,sex)

学 习 关 系 SC(snum,cnum,grade)

课程关系 C(cnum,cname,teacher)

- (1) 检索学习课程号为 C2 的学生学号与成绩
- (2) 检索学习课程号为 C2 的学生学号与姓名
- (3) 检索选修课程名为 MATHS 的学生学号与姓名
- (4) 检索选修课程号为 C2 或 C4 的学生学号
- (5) 检索不学课程号为 C2 的学生姓名与年龄
- (6) 检索学习全部课程的学生姓名
- (7) 检索所学课程包含学生 S3 所学课程的学生号

课堂作业答案

(1) 检索学习课程号为 C2 的学生学号与成绩

$$\pi_{\text{snum,grade}}(\sigma_{\text{cnum}='c2'}(\text{SC}))$$

(2) 检索学习课程号为 C2 的学生学号与姓名

$$\pi_{\text{snum,sname}}(\sigma_{\text{cnum}='C2'}(\text{S} \bowtie \text{SC}))$$

课堂作业答案

(3) 检索选修课程名为 MATHS 的学生学号与姓名

$$\pi_{\text{snum}, \text{sname}} (\sigma_{\text{cname} = \text{'Maths'}} (\text{S} \bowtie \text{SC} \bowtie \text{C}))$$

(4) 检索选修课程号为 C2 或 C4 的学生学号

$$\pi_{\text{snum}} (\sigma_{\text{cnum} = \text{'C4'} \vee \text{cnum} = \text{'C2'}} (\text{SC}))$$

课堂作业答案

(5) 检索不学课程号为 C2 的学生姓名与年龄

$$\pi_{\text{sname, age}}(S) - \pi_{\text{sname, age}}(\sigma_{\text{cnum}='C2'}(S \bowtie SC))$$

课堂作业答案

(6) 检索学习全部课程的学生姓名

$$\pi_{\text{sname}} (\text{S} \bowtie (\pi_{\text{snum}, \text{cnum}} (\text{SC}) \div \pi_{\text{cnum}} (\text{C})))$$

(7) 检索所学课程包含学生 S3 所学课程的学生
号

$$\pi_{\text{snum}, \text{cnum}} (\text{SC}) \div \pi_{\text{cnum}} (\sigma_{\text{snum}='S3'} (\text{SC}))$$

参考题

对于学生选课关系，其关系模式为：

学生（学号，姓名，年龄，所在系）

课程（课程号，课程名，先行课号）

选课（学号，课程号，成绩）

用关系代数和 SQL 分别完成以下查询：求学过数据库的先行课的学生学号

