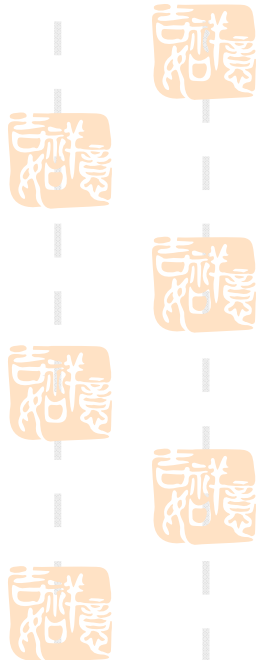




数据库系统概论

An Introduction to Database System

第二章 关系数据库



第二章 关系数据库



2.1 关系数据结构及形式化定义

2.2 关系操作

2.3 关系的完整性

2.4 关系代数

2.5 小结



关系模型



一 单一的数据结构

- 关系 --- 实体及实体之间的联系均用关系来表示

二 关系操作

- 关系代数语言
- 关系演算语言
- **SQL**

三 完整性约束

- 实体完整性
- 参照完整性
- 用户定义完整性



几个概念



1. 域 (Domain)

2. 笛卡尔积 (Cartesian Product)

3. 关系 (Relation)



1. 域 (Domain)

- 域是一组具有相同数据类型的值的集合。

例:

- 整数



- 实数



- 介于某个取值范围的整数



- 长度指定长度的字符串集合



- {'男', '女'}



-

2. 笛卡尔积 (Cartesian Product)

■ 笛卡尔积

给定一组域 D_1, D_2, \dots, D_n , 这些域中可以有相同的域。

D_1, D_2, \dots, D_n 的笛卡尔积为:



$$D_1 \times D_2 \times \dots \times D_n =$$



$$\{ (d_1, d_2, \dots, d_n) \mid d_i \in D_i, i=1,$$



$$2, \dots, n \}$$



➤ 所有域的所有取值的一个组合



➤ 不能重复

笛卡儿积 (Cartesian Product)



例: D1=导师集合SUPERVISOR=张清玫, 刘逸

D2=专业集合SPECIALITY=计算机专业, 信息专业

D3=研究生集合POSTGRADUATE=李勇, 刘晨, 王敏

则: $D1 \times D2 \times D3 =$

{ (张清玫, 计算机专业, 李勇),

(张清玫, 计算机专业, 刘晨),

(张清玫, 计算机专业, 王敏),

(张清玫, 信息专业, 李勇),

(张清玫, 信息专业, 刘晨),

(张清玫, 信息专业, 王敏),

(刘逸, 计算机专业, 李勇),

(刘逸, 计算机专业, 刘晨),

(刘逸, 计算机专业, 王敏),

(刘逸, 信息专业, 李勇),

(刘逸, 信息专业, 刘晨),

(刘逸, 信息专业, 王敏) }



笛卡尔积（续）



- 元组 (Tuple)

- 笛卡尔积中每一个元素 (d_1, d_2, \dots, d_n) 叫作一个 n 元组 (n-tuple) 或简称元组(Tuple)
- (张清玫, 计算机专业, 李勇)、(张清玫, 计算机专业, 刘晨)等都是元组



- 分量 (Component)

- 笛卡尔积元素 (d_1, d_2, \dots, d_n) 中的每一个值 d_i 叫作一个分量
- 张清玫、计算机专业、李勇、刘晨等都是分量



笛卡尔积（续）

■ 基数（Cardinal number）

- 一个域允许的不同取值个数称为这个域的基数
- 若 D_i ($i=1, 2, \dots, n$) 为有限集，其基数为 m_i ($i=1, 2, \dots, n$)，则 $D_1 \times D_2 \times \dots \times D_n$ 的基数 M 为：

$$M = \prod_{i=1}^n m_i$$

■ 笛卡尔积的表示方法

- 笛卡尔积可表示为一个二维表
- 表中的每行对应一个元组，表中的每列对应一个域

表 2.1 D_1, D_2, D_3 的笛卡尔积

SUPERVISOR	SPECIALITY	POSTGRADUATE
张清玫	计算机专业	李勇
张清玫	计算机专业	刘晨
张清玫	计算机专业	王敏
张清玫	信息专业	李勇
张清玫	信息专业	刘晨
张清玫	信息专业	王敏
刘逸	计算机专业	李勇
刘逸	计算机专业	刘晨
刘逸	计算机专业	王敏
刘逸	信息专业	李勇
刘逸	信息专业	刘晨
刘逸	信息专业	王敏

3. 关系 (Relation)

1) 关系

$D_1 \times D_2 \times \dots \times D_n$ 的子集叫作在域 D_1, D_2, \dots, D_n 上的关系, 表示为

$$R(D_1, D_2, \dots, D_n)$$

例: $R(\text{SUPERVISOR}, \text{SPECIALITY}, \text{POSTGRADUATE})$

{ (张清玫, 计算机专业, 李勇)
(张清玫, 计算机专业, 刘晨)
(刘逸, 信息专业, 王敏) }

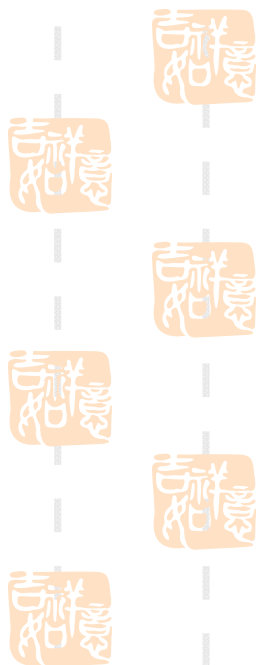
- R : 关系名
- n : 关系的目或度 (Degree)

关系（续）



2) 元组

关系中的每个元素是关系中的元组，通常用 t 表示。



关系（续）



4) 关系的表示--二维表

表2.2 SAP关系

SUPERVISOR	SPECIALITY	POSTGRADUATE
张清玫	信息专业	李勇
张清玫	信息专业	刘晨
刘逸	信息专业	王敏



关系（续）



5)属性

- 关系中不同列可以对应相同的域
- 为了加以区分，必须对每列起一个名字，称为属性（Attribute）

- n 目关系必有 n 个属性



关系（续）



6)码

超码/超键(superkey)

能唯一地标识关系中的一个元组的一属性组

- 如：关系**student**中的（学号，姓名）

候选码（Candidate key）

若关系中的某一属性组的值能唯一地标识一个元组，而其子集不能，则称该属性组为候选r码。

- 如：关系**student**中的（学号）
- 候选码是一个特殊的超键。是最小的超键。
- 简单的情况：候选码只包含一个属性

关系（续）

全码（All-key）

最极端的情况：关系模式的所有属性组是这个关系模式的候选码，称为全码（All-key）

主码

若一个关系有多个候选码，则选定其中一个为主码（Primary key）

主码应该选择从不或很少变化的属性。比如地址不应该作为主码的一部分。

主属性

候选码的诸属性称为主属性（Prime attribute）

不包含在任何侯选码中的属性称为非主属性（ Non-Prime attribute）或非码属性（Non-key attribute）

关系（续）



7) 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



关系（续）



8)基本关系的性质

- ① 列是同质的（Homogeneous）
- ② 不同的列可出自同一个域
 - 其中的每一列称为一个属性
 - 不同的属性要给予不同的属性名
- ③ 列的顺序无所谓,，列的次序可以任意交换
- ④ 任意两个元组的候选码不能相同
- ⑤ 行的顺序无所谓，行的次序可以任意交换



基本关系的性质(续)

⑥ 分量必须取原子值

这是规范条件中最基本的一条

表2.3 非规范化关系

SUPERVISOR	SPECIALITY	POSTGRADUATE	
		PG1	PG2
张清玫	信息专业	李勇	刘晨
刘逸	信息专业	王敏	

小表

2.1 关系数据结构



2.1.1 关系

2.1.2 关系模式

2.1.3 关系数据库



1. 什么是关系模式



- 关系模式（Relation Schema）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

- 关系是值



2. 定义关系模式

关系模式可以形式化地表示为：

$R(U, D, DOM, F)$

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

定义关系模式 (续)

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

3. 关系模式与关系



■ 关系模式

- 对关系的描述
- 静态的、稳定的

■ 关系

- 关系模式在某一时刻的状态或内容
- 动态的、随时间不断变化的



2.1 关系数据结构



2.1.1 关系

2.1.2 关系模式

2.1.3 关系数据库



2.1.3 关系数据库



- 关系数据库

- 在一个给定的应用领域中，所有关系的集合构成一个关系数据库



2.1.4 关系模型的存储结构

■ 关系数据库的物理组织

➤ 有的关系数据库管理系统中一个表对应一个操作系统文件，将物理数据组织交给操作系统完成

➤ 有的关系数据库管理系统从操作系统那里申请若干个大的文件，自己划分文件空间，组织表、索引等存储结构，并进行存储管理

第二章 关系数据库



2.1 关系模型概述

2.2 关系操作

2.3 关系的完整性

2.4 关系代数

2.5 小结



2.2.1 基本关系操作



■ 常用的关系操作

➤ 查询：选择、投影、连接、除、并、交、差、笛卡尔积

■ 选择、投影、并、差、笛卡尔积是5种基本操作

➤ 数据更新：插入、删除、修改

■ 关系操作的特点

➤ 集合操作方式：操作的对象和结果都是集合，一次一集

合的方式



2.2.2 关系数据库语言的分类

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

第二章 关系数据库



2.1 关系数据结构及形式化定义

2.2 关系操作

2.3 关系的完整性

2.4 关系代数

2.5 小结



2.3 关系的完整性



2.3.1 实体完整性

2.3.2 参照完整性



2.3.3 用户定义的完整性



关系的三类完整性约束

- 实体完整性和参照完整性：

关系模型必须满足的完整性约束条件，称为关系的两个不变性。由关系系统自动支持。

- 用户定义的完整性：

应用领域需要遵循的约束条件，体现了具体领域中的语义约束

2.3.1 实体完整性约束

- **实体完整性：** 一个基本关系的主码值应是唯一的。所有主码属性都不能取空值。
 - 由于主键可唯一地标识某实体，如果某主码属性为空值，表示“不知道”，或“无意义”的值，则说明存在不可标识的实体。
 - DBMS中允许用户不设定主键（即不进行实体完整性约束检查）

2.3.2 参照完整性



1. 关系间的引用

2. 外码

3. 参照完整性规则



1. 关系间的引用

- 在关系模型中实体及实体间的联系都是用关系来描述的，自然存在着关系与关系间的引用。

【例2.1】 学生实体、专业实体

主码

学生（学号，姓名，性别，专业号，年龄）

专业（专业号，专业名）

主码

- ❖ 学生关系引用了专业关系的主码“专业号”。
- ❖ 学生关系中的“专业号”值必须是确实存在的专业的专业号

关系间的引用（续）



例[2.2] 学生、课程、学生与课程之间的多对多联系

学生（学号，姓名，性别，专业号，年龄）

课程（课程号，课程名，学分）

选修（学号，课程号，成绩）



关系间的引用（续）

例[2.3] 学生实体及其内部的一对多联系

学生（学号，姓名，性别，专业号，年龄，班长）

学号	姓名	性别	专业号	年龄	班长
801	张三	女	01	19	802
802	李四	男	01	20	802
803	王五	男	01	20	802
804	赵六	女	02	20	805
805	钱七	男	02	19	805

“学号”是主码，“班长”是外码，它引用了本关系的“学号”

“班长”必须是确实存在的学生的学号

2. 外码 (Foreign Key)

- 设 F 是基本关系 R 的一个或一组属性，但不是关系 R 的码。
如果 F 与基本关系 S 的主码 K_s 相对应，则称 F 是 R 的**外码**
- 基本关系 R 称为**参照关系** (Referencing Relation)
- 基本关系 S 称为**被参照关系** (Referenced Relation)
或**目标关系** (Target Relation)

外码（续）

- [例2.1]中学生关系的“专业号”与专业关系的主码“专业号”相对应
 - “专业号”属性是学生关系的外码
 - 专业关系是被参照关系，学生关系为参照关

系

学生关系 $\xrightarrow{\text{专业号}}$ 专业关系
(a)

外码（续）

■ [例2.2]中

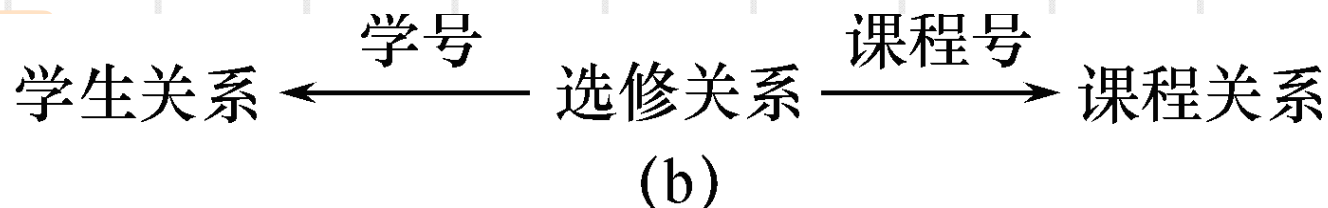
选修关系的“学号”与学生关系的主码“学号”相对应

选修关系的“课程号”与课程关系的主码“课程号”相对应

➤ “学号”和“课程号”是选修关系的外码

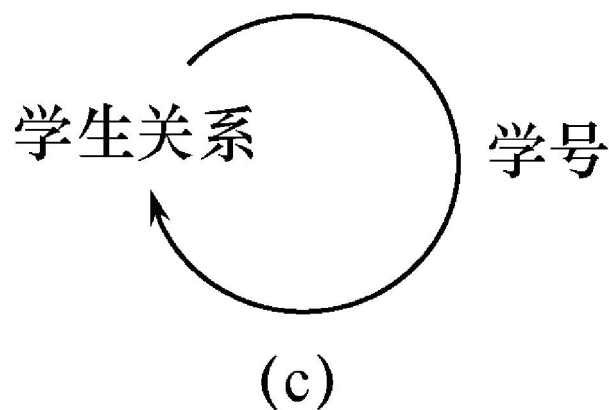
➤ 学生关系和课程关系均为被参照关系

➤ 选修关系为参照关系



外码（续）

- [例2.3]中“班长”与本身的主码“学号”相对应
 - “班长”是外码
 - 学生关系既是参照关系也是被参照关系



外码（续）



- 关系 R 和 S 不一定是不同的关系
- 目标关系 S 的主码 K_s 和参照关系的外码 F 必须定义在同一个（或一组）域上



- 外码并不一定要与相应的主码同名



当外码与相应的主码属于不同关系时，往往取相同的名字，以便于识别



3. 参照完整性规则

■ 规则2.2 参照完整性规则

若属性（或属性组） F 是基本关系 R 的外码它与基本关系 S 的主码 K_s 相对应（基本关系 R 和 S 不一定是不同的关系），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

思考：

选修(学号, 课程号, 成绩)

Student2(学号, 姓名, 性别, 专业号, 年龄, 班长)

外键分别为学号, 课程号和班长, 其取值可以为空值?

关系的完整性约束

- **用户定义的完整性：** 针对某一具体关系数据库的约束条件。它反映某一具体应用所涉及的数据必须满足的语义要求。包括某属性取唯一值，某一属性值应满足一定的函数关系，某一属性的取值范围等等。

➤ 可采用trigger, check clause

例如选修关系的成绩定义为整数，范围还太大，我们可以写如下规则把成绩限制在0~100之间：

```
CHECK (GRADE BETWEEN 0 AND 100)
```

第二章 关系数据库



2.1 关系模型概述

2.2 关系数据结构

2.3 关系的完整性

2.4 关系代数

2.5 小结



2.4 关系代数

- 关系代数是一种抽象的查询语言，它用对关系的运算来表达查询
- 关系代数
 - 运算对象是关系
 - 运算结果亦为关系
 - 关系代数的运算符有两类：集合运算符和专门的关系运算符
- 传统的集合运算是从关系的“水平”方向即行的角度进行
- 专门的关系运算不仅涉及行而且涉及列

2.4 关系代数

表2.4 关系代数运算符

运 算 符		含 义
集合 运算符	\cup	并
	$-$	差
	\cap	交
	\times	笛卡尔积
专门的 关系 运算符	σ	选择
	π	投影
	\bowtie	连接
	\div	除

关系代数运算符

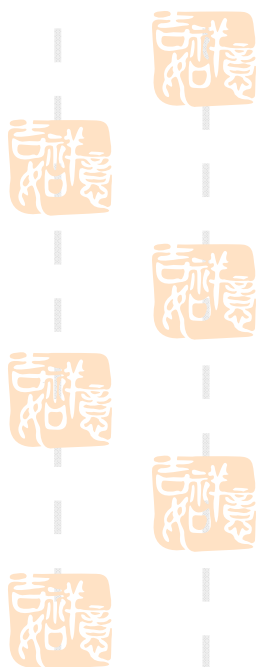
比较运算符和逻辑运算符

运算符		含义	运算符		含义
比较运算符	$>$	大于	逻辑运算符	\neg	非
	\geq	大于等于		\wedge	与
	$<$	小于		\vee	或
	\leq	小于等于			
	$=$	等于			
	$<>$	不等于			

2.4 关系代数



- 传统的集合运算
- 专门的关系运算



1. 并 (Union)

■ R 和 S

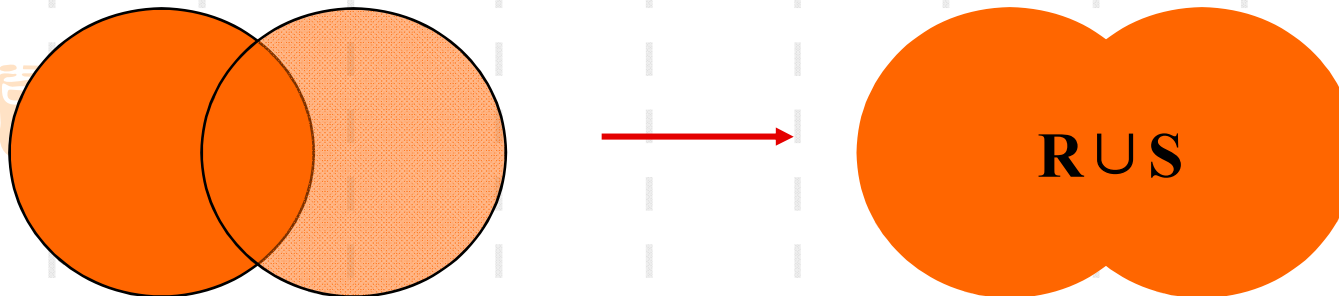
- 具有相同的目 n (即两个关系都有 n 个属性)
- 相应的属性取自同一个域

■ $R \cup S$

- 仍为 n 目关系, 由属于 R 或属于 S 的元组组成

$$R \cup S = \{ t | t \in R \vee t \in S \}$$

- 一个元素在并集中只出现一次



并(续)



R		
A	B	C
a_1	b_1	c_1
a_1	b_2	c_2
a_2	b_2	c_1



S		
A	B	C
a_1	b_2	c_2
a_1	b_3	c_2
a_2	b_2	c_1



RUS		
A	B	C
a_1	b_1	c_1
a_1	b_2	c_2
a_2	b_2	c_1
a_1	b_3	c_2

并(续)

顾客号	姓名	性别	年龄
S01	张宏	男	45
S02	李丽	女	34
S03	王敏	女	28

顾客号	姓名	性别	年龄
S02	李丽	女	34
S04	钱景	男	50
S06	王平	女	24

顾客号	姓名	性别	年龄
S01	张宏	男	45
S02	李丽	女	34
S03	王敏	女	28
S04	钱景	男	50
S06	王平	女	24

2. 差 (Difference)

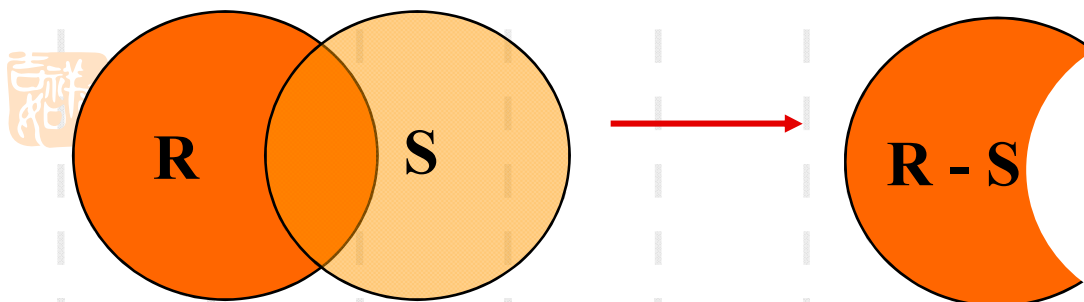
- R 和 S

- 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

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



差(续)



R		
A	B	C
a_1	b_1	c_1
a_1	b_2	c_2
a_2	b_2	c_1



S		
A	B	C
a_1	b_2	c_2
a_1	b_3	c_2
a_2	b_2	c_1



$R-S$		
A	B	C
a_1	b_1	c_1

差(续)



顾客号	姓名	性别	年龄
S01	张宏	男	45
S02	李丽	女	34
S03	王敏	女	28

顾客号	姓名	性别	年龄
S02	李丽	女	34
S04	钱景	男	50
S06	王平	女	24



顾客号	姓名	性别	年龄
S01	张宏	男	45
S03	王敏	女	28



3. 交 (Intersection)

- R 和 S

- 具有相同的目 n
- 相应的属性取自同一个域

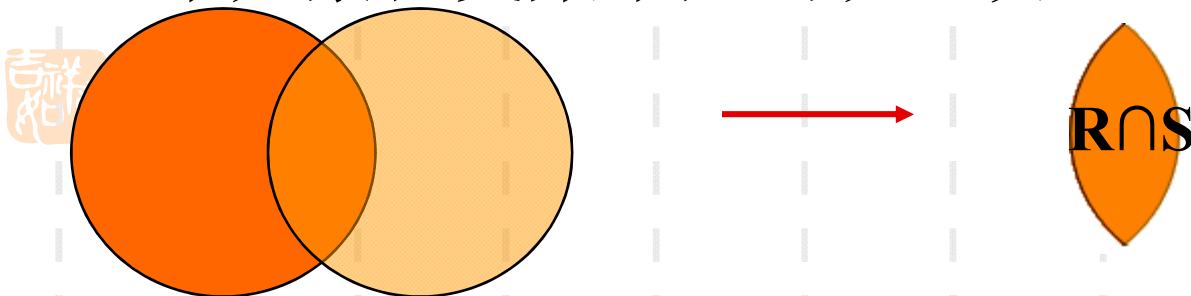
- $R \cap S$

- 仍为 n 目关系，由既属于 R 又属于 S 的元组组成

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

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

- 一个元素在交集中只出现一次



交 (续)



R		
A	B	C
a_1	b_1	c_1
a_1	b_2	c_2
a_2	b_2	c_1



S		
A	B	C
a_1	b_2	c_2
a_1	b_3	c_2
a_2	b_2	c_1

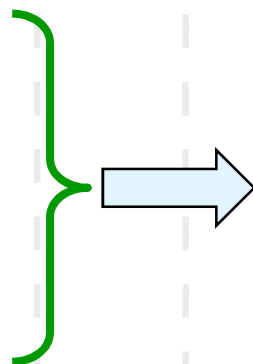


$R \cap S$		
A	B	C
a_1	b_2	c_2
a_2	b_2	c_1

交 (续)

顾客号	姓名	性别	年龄
S01	张宏	男	45
S02	李丽	女	34
S03	王敏	女	28

顾客号	姓名	性别	年龄
S02	李丽	女	34
S04	钱景	男	50
S06	王平	女	24



顾客号	姓名	性别	年龄
S02	李丽	女	34

4. 笛卡尔积 (Cartesian Product)

- 严格地讲应该是广义的笛卡尔积 (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
 - $R \times S = \{ \overbrace{t_r t_s} \mid t_r \in R \wedge t_s \in S \}$

笛卡尔积(续)



R		
A	B	C
a_1	b_1	c_1
a_1	b_2	c_2
a_2	b_2	c_1

S		
A	B	C
a_1	b_2	c_2
a_1	b_3	c_2
a_2	b_2	c_1

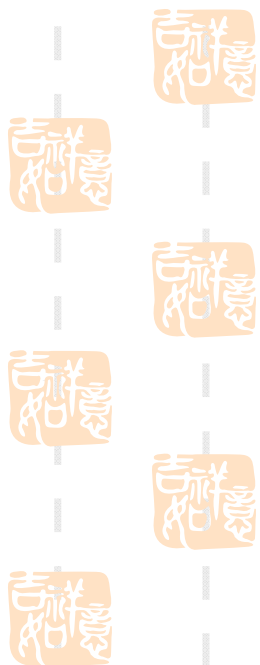
$R \times S$					
$R.A$	$R.B$	$R.C$	$S.A$	$S.B$	$S.C$
a_1	b_1	c_1	a_1	b_2	c_2
a_1	b_1	c_1	a_1	b_3	c_2
a_1	b_1	c_1	a_2	b_2	c_1
a_1	b_2	c_2	a_1	b_2	c_2
a_1	b_2	c_2	a_1	b_3	c_2
a_1	b_2	c_2	a_2	b_2	c_1
a_2	b_2	c_1	a_1	b_2	c_2
a_2	b_2	c_1	a_1	b_3	c_2
a_2	b_2	c_1	a_2	b_2	c_1



2.4 关系代数



- 传统的集合运算
- 专门的关系运算



2.4.2 专门的关系运算

先引入几个记号

(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

专门的关系运算(续)

(2) A , $t[A]$, \bar{A}

若 $A = \{A_{i1}, A_{i2}, \dots, A_{ik}\}$, 其中 $A_{i1}, A_{i2}, \dots, A_{ik}$ 是 A_1, A_2, \dots, A_n 中的一部分, 则 A 称为属性列或属性组。

$t[A] = (t[A_{i1}], t[A_{i2}], \dots, t[A_{ik}])$ 表示元组 t 在属性列 A 上诸分量的集合。

\bar{A} 则表示 $\{A_1, A_2, \dots, A_n\}$ 中去掉 $\{A_{i1}, A_{i2}, \dots, A_{ik}\}$ 后剩余的属性组。

专门的关系运算(续)



(3) $\widehat{t_r t_s}$

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。



$\widehat{t_r t_s}$ 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。



专门的关系运算(续)

(4) 象集 Z_x

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

专门的关系运算(续)



R

x_1	Z_1
x_1	Z_2
x_1	Z_3
x_2	Z_2
x_2	Z_3
x_3	Z_1
x_3	Z_3



象集举例



- x_1 在 R 中的象集

$$Z_{x1} = \{Z_1, Z_2, Z_3\},$$

- x_2 在 R 中的象集

$$Z_{x2} = \{Z_2, Z_3\},$$

- x_3 在 R 中的象集

$$Z_{x3} = \{Z_1, Z_3\}$$

专门的关系运算(续)



- 选择

- 投影

-  连接



-  除



专门的关系运算(续)

4) 学生-课程数据库:

学生关系Student、课程关系Course和选修关系SC

Student

学号 Sno	姓名 Sname	性别 Ssex	年龄 Sage	所在系 Sdept
200215121	李勇	男	20	CS
200215122	刘晨	女	19	IS
200215123	王敏	女	18	MA
200215125	张立	男	19	IS

(a)

专门的关系运算(续)

Course

课程号 Cno	课程名 Cname	先行课 Cpno	学分 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	PASCAL语言	6	4

(b)

专门的关系运算(续)

SC

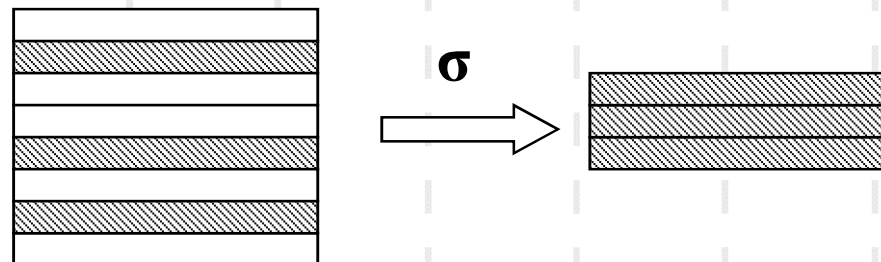
学号 Sno	课程号 Cno	成绩 Grade
200215121	1	92
200215121	2	85
200215121	3	88
200215122	2	90
200215122	3	80

(c)

1. 选择 (Selection)

按照给定条件从指定的关系中挑选出满足条件的元组构成新的关系。或者说，选择运算的结果是一个表的行的子集。记作

$\sigma_{\langle \text{选择条件} \rangle}(\langle \text{关系名} \rangle)$



例1

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

学号	姓名	性别	年龄	所在系
Sno	Sname	Ssex	Sage	Sdept
95001	李勇	男	20	CS
95002	刘晨	女	19	IS
95003	张敏	男	18	NSA
95004	张立	男	19	IS

例2 $\sigma_{Sage < 20}(Student)$

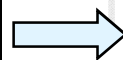
学号 Sno	姓名 Sname	性别 Ssex	年龄 Sage	所在系 Sdept
95002	荆晨	男	20	CS
95003	刘晨	女	18	NSA
95004	张敏	男	19	NSA
95004	张立	男	19	IS

选择（续）

- 例，选择系为数学系的男学生信息：

$\sigma_{\text{性别} = \text{'男'} \text{ AND 系} = \text{'数学系'}} (\text{STUDENT}) :$

学号	姓名	性别	年龄	系
000101	李晨	男	18	信息系
000102	王博	女	19	数学系
010101	刘思思	女	18	信息系
010102	王国美	女	20	物理系
020101	范伟	男	19	数学系



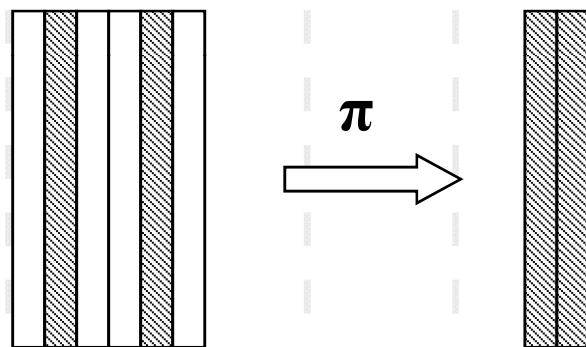
学号	姓名	性别	年龄	系
020101	范伟	男	19	数学系

2. 投影 (Projection)

从指定的关系中挑选出某些属性构成新的关系。

$\Pi_{\langle \text{属性表} \rangle} (\langle \text{关系名} \rangle)$

- 投影操作主要是从列的角度进行运算



- **注意：** 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行（避免重复行）

例3
 $\pi_{Sname, Sdept}(Student)$

学号 Sno	姓名 Sname	性别 Ssex	所在系 Sdept	年龄 Sage	所在系 Sdept
95001	李勇	男	CS	20	CS
95002	刘晨	女	IS	19	IS
95003	王敏	女	MA	18	MA
95004	张立	男	IS	19	IS

例4 $\pi_{Sdept}(Student)$

所在系 Sdept	姓名 Sname	性别 Ssex	年龄 Sage	所在系 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

删除多余的元组后

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

$$R \bowtie_{A\theta B} S = \{ t_r \hat{\smile} t_s \mid t_r \in R \wedge t_s \in S \wedge t_r[A] \theta t_s[B] \}$$

➤ 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 θ 的元组

例5 连接

$$R$$

A	B	C
a1	b1	5
a1	b2	6
a2	b3	8
a2	b4	12

$$S$$

B	E
b1	3
b2	7
b3	10
b3	2
b5	2

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

A	R.B	C	S.B	E
a1	b1	5	b2	7
a1	b1	5	b3	10
a1	b2	6	b2	7
a1	b2	6	b3	10
a2	b3	8	b3	10

连接(续)



■ 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：

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



等值连接



R

A	B	C
a1	b1	5
a1	b2	6
a2	b3	8
a2	b4	12

S

B	C
b1	3
b2	7
b3	10
b3	2
b5	2

$R \bowtie_{R.B=S.B} S$

A	R.B	C	S.B	E
a1	b1	5	b1	3
a1	b2	6	b2	7
a2	b3	8	b3	10
a2	b3	8	b3	2

连接操作的例子（等值连接）

例：

student

course

SC

学号	姓名	性别	年龄	系
000101	李晨	男	18	信息系
000102	王博	女	19	数学系
010101	刘思思	女	18	信息系
010102	王国美	女	20	物理系
020101	范伟	男	19	数学系

课程号	名称
1	数学
2	英语
3	计算机
4	制图

学号	课程号	成绩
000101	1	90
000101	2	85
000101	3	92
010102	1	77
010102	2	80

student  student.学号=SC.学号 SC

Student.学号	姓名	性别	年龄	系	SC.学号	课程号	成绩
000101	李晨	男	18	信息系	000101	1	90
000101	李晨	男	18	信息系	000101	2	85
000101	李晨	男	18	信息系	000101	3	92
010102	王国美	女	20	物理系	010102	1	77
010102	王国美	女	20	物理系	010102	2	80

连接(续)



➤ 自然连接 (Natural join)

- 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

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



连接操作的例子（自然连接）

例： student

学号	姓名	性别	年龄	系
000101	李晨	男	18	信息系
000102	王博	女	19	数学系
010101	刘思思	女	18	信息系
010102	王国美	女	20	物理系
020101	范伟	男	19	数学系

course

课程号	名称
1	数学
2	英语
3	计算机
4	制图

sc

学号	课程号	成绩
000101	1	90
000101	2	85
000101	3	92
010102	1	77
010102	2	80

student ⋈ sc

学号	姓名	性别	年龄	系	课程号	成绩
000101	李晨	男	18	信息系	1	90
000101	李晨	男	18	信息系	2	85
000101	李晨	男	18	信息系	3	92
010102	王国美	女	20	物理系	1	77
010102	王国美	女	20	物理系	2	80

连接（续）

■ 悬浮元组（Dangling tuple）

- 两个关系 R 和 S 在做自然连接时，关系 R 中某些元组有可能在 S 中不存在公共属性上值相等的元组，从而造成 R 中这些元组在操作时被舍弃了，这些被舍弃的元组称为悬浮元组。



连接(续)

■ 外连接

- 如果把舍弃的元组也保存在结果关系中，而在其他属性上填空值(Null)，这种连接就叫做外连接(OUTER JOIN)。

■ 左外连接

- 如果只把左边关系 R 中要舍弃的元组保留就叫做左外连接(LEFT OUTER JOIN或LEFT JOIN)

■ 右外连接

- 如果只把右边关系 S 中要舍弃的元组保留就叫做右外连接(RIGHT OUTER JOIN或RIGHT JOIN)。

外连接例子

R

A	B
a1	b1
a2	b2
a3	b5

S

B	C
b1	c1
b2	c2
b3	c3
b4	c4

自然连接

$R \bowtie S$

A	B	C
A1	b1	c1
a2	b2	c2

• 全外连接

$R^* \bowtie^* S$

A	B	C
a1	b1	C1
a2	b2	C2
a3	b5	Null
null	b3	C3
null	b4	C4

• 左外连接

$R^* \bowtie S$

A	B	C
a1	b1	c1
a2	b2	c2
a3	b5	null

• 右外连接

$R \bowtie^* S$

A	B	C
a1	b1	c1
a2	b2	c2
null	b3	c3
null	b4	c4

R 如何求 a_i 象集?

A	B	C
a_1	b_1	c_2
a_2	b_3	c_7
a_3	b_4	c_6
a_1	b_2	c_3
a_4	b_6	c_6
a_2	b_2	c_3
a_1	b_2	c_1

a_i 在 R 中的象集

A	B	C
a_1	b_1	c_2
a_1	b_2	c_3
a_1	b_2	c_1
a_2	b_3	c_7
a_2	b_2	c_3
a_3	b_4	c_6
a_4	b_6	c_6

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:

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, 记作:

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

$$Y_x: x \text{ 在 } R \text{ 中的象集}, Y_x = \{ t_r[Y] \mid t_r[X] = x \}$$

除法例子

R

1	2	3	4
a1	b1	c1	d1
a1	b1	c2	d2
a1	b1	c3	d3
a2	b2	c1	d1
a2	b2	c2	d2
a3	b3	c1	d1

S

3	4
c1	d1
c2	d2

$R \div S$

1	2
a1	b1
a2	b2

除法例子



R

SNO	CNO
99001	c1
99001	c4
99001	c7
99002	c8
99003	c3
99003	c6
99004	c2
99004	c5
99005	c1
99005	c5
99005	c7

S

CNO
c2
c5

$T := R \div S$

SNO
99004

除法例子

R

A	B	C
a1	b1	c1
a2	b1	c1
a1	b2	c1
a1	b2	c2
a2	b1	c2
a1	b2	c3
a1	b2	c4
a1	b1	c5

S1

C
c1

S2

C
c1
c2

S3

C
c1
c2
c3
c4

$R \div S1$

A	B
a1	b1
a2	b1
a1	b2

$R \div S2$

A	B
a1	b2
a2	b1

$R \div S3$

A	B
a1	b2

除法例子

例1：查询先修课程覆盖操作系统先修课程的课程名。

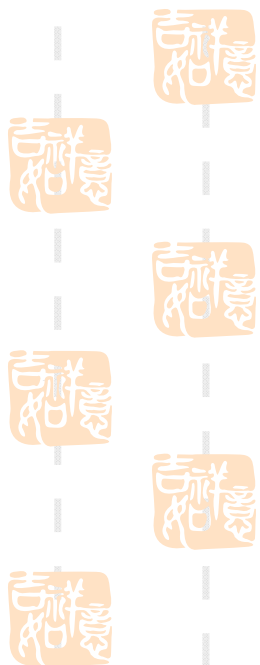
$\Pi_{\text{课程名, 先修课程号}}(\text{COURSE}) \div \Pi_{\text{先修课程号}}(\sigma_{\text{课程名} = \text{'操作系统'}}(\text{COURSE}))$

例2：查询选修了全部课程的学号和姓名

$\Pi_{\text{学号, 姓名, 课程号}}(\text{student} \bowtie \text{sc}) \div \Pi_{\text{课程号}}(\text{COURSE})$

复杂的关系代数表达式

- 可以运用多种关系代数运算构成复杂的查询表达式
- 可以使用关系的别名构成查询表达式



综合举例(续)



[例 8] 查询选修了2号课程的学生们的学号。

$\pi_{Sno} (\sigma_{Cno='2'} (SC))$



综合举例(续)



[例9] 查询至少选修了一门其直接先行课为5号课程的的学生姓名

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

或



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

或



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



综合举例(续)

- **STUDENT**(姓名, 学号, 性别, 出生年月, 系别)
- **Course**(课程名, 课程号, 学分, 开课时间, 先修课程号)
- **SC**(学号, 课程号, 成绩)

1、查询所有选修课程号为0001的男生的成绩。

- $\Pi_{\text{姓名, 成绩}} (\sigma_{\text{性别} = 'm' \text{ AND } \text{课程号} = '0001'} (\text{STUDENT} \bowtie \text{SC}))$

2、查询至少选修了二门课程的学生姓名

$\text{S_G1} := \text{STUDENT} \bowtie \text{SC}$

$\text{S_G2} := \text{S_G1}$

- $\Pi_{\text{姓名}} (\text{S_G1} \bowtie_{\text{S_G1.课程号} \neq \text{S_G2.课程号} \text{ and } \text{S_G1.学号} = \text{S_G2.学号}} (\text{S_G2}))$

或者

$\text{SC1} := \text{SC}$

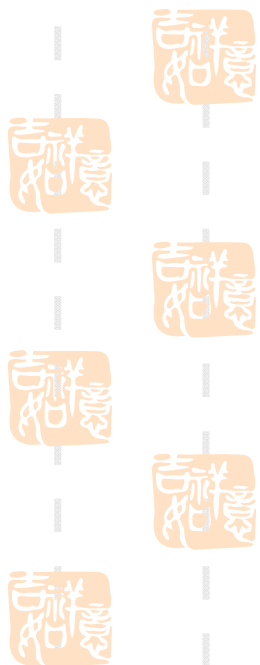
$\Pi_{\text{姓名}} (\Pi_{\text{学号}} (\text{SC} \bowtie_{\text{SC.课程号} \neq \text{SC1.课程号} \text{ and } \text{SC.学号} = \text{SC1.学号}} \text{SC1}) \bowtie \text{STUDENT})$

课堂练习

- **STUDENT**(姓名, 学号, 性别, 出生年月, 系别)
- **Course**(课程名, 课程号, 学分, 开课时间, 先修课程号)
- **SC**(学号, 课程号, 成绩)
- 1. 查询1994年1月1日后出生的女学生
- 2. 查询计算机系的男同学的姓名和学号
- 3. 查询计算机系学生的姓名、学号、选修的课程和成绩
- 4. 查询学生的姓名、学号、选修的课程和成绩, 没有选课的学生也要列出。

2.6 小结

- 关系数据库系统是目前使用最广泛的数据库系统



小结（续）



- 关系数据结构

- 关系

- 域

- 笛卡尔积

- 关系

- 关系，属性，元组

- 候选码，主码，主属性

- 基本关系的性质

- 关系模式

- 关系数据库



小结（续）



■ 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



小结（续）

- 关系的完整性约束

- 实体完整性

- 参照完整性

- 外码

- 用户定义的完整性



作业

- 1, 3, 6(用关系代数就可以了) (第5版)
- 补充:
 - 1、对于给定的表T

A	B	C	D
a1	b1	c1	d2
a2	b2	c2	d1
a3	b2	c1	d2
a4	b1	c2	d1

请找出所有的侯选键，说明原因。