第二章 关系数据库



基本要求:

- 1.掌握关系模型的三要素,在此基础上理解 关系数据库管理系统
- 2.关系代数运算是本章的重点和难点,要求熟练使用关系代数的有关操作完成查询功能



- 今天, 国际上主流数据库产品都采用了关系数据库技术
- 典型实验系统
 - □ System R
 - University Ingres
- 典型商用系统
 - □ Oracle, Sybase, DB2, SQL Server
 - □ Informix, Ingres
- 开源数据库
 - MySQL, PostGreSQL, SQLite

■ 关系数据库系统是支持关系模型的数 据库系统

- 关系模型的组成
 - 1.关系数据结构
 - 2.关系操作集合
 - 3.关系完整性约束



第二章 关系数据库

- 2.1 关系数据结构及其形式化定义
- 2.2 关系操作
- 2.3 关系的完整性(重点)
- 2.4 关系代数 (重点)



2.1 关系数据结构及其形式化定义

- 关系模型建立在集合代数的基础上
- 关系数据结构的基本概念
 - □关系
 - 域 (Domain)
 - 笛卡尔积 (Cartesian Product)
 - 关系 (Relation)
 - □关系模式
 - □关系数据库



一、关系

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
 - ■指定长度的字符串集合
 - ■介于某个取值范围的整数
 - ■{'男', '女'}
 - 介于某个取值范围的日期

M

2. 笛卡尔积 (Cartesian Product)

1) 笛卡尔积

给定一组域 D_1 , D_2 , ..., D_n , 这些域中可以有相同的。 D_1 , D_2 , ..., D_n 的笛卡尔积为:

$$D_1 \times D_2 \times ... \times D_n = \{(d_1, d_2, ..., d_n) \mid d_i \in D_i, i = 1, 2, ..., n\}$$

- □ 所有域的所有取值的一个组合
- □ 不能重复

2) 元组 (Tuple)

□ 笛卡尔积中每一个元素(d_1 , d_2 , ... , d_n)叫作一个n元组(n-tuple)或简称元组。

3) 分量 (Component)

□ 笛卡尔积元素 $(d_1, d_2, ..., d_n)$ 中的每一个值 d_i 叫作一个分量。

M

例 给出三个域:

 D_1 =SUPERVISOR ={ 张清玫,刘逸 } D_2 =SPECIALITY={<mark>计算机专业</mark>,信息专业} D_3 =POSTGRADUATE={李勇,刘晨,王敏} 则 D_1 , D_2 , D_3 的笛卡尔积为:

$D_1 \times D_2 \times D_3 =$

{(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, 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, 李勇), (刘逸, 信息专业, 王敏)}



4) 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, ..., n) 为有限集,其基数为 m_i (i = 1, 2, ..., n) ,则 $D_1 \times D_2 \times ... \times D_n$ 的基数M为:

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

在上例中,基数: $2 \times 2 \times 3 = 12$,即 $D_1 \times D_2 \times D_3$ 共有 $2 \times 2 \times 3 = 12$ 个元组



5) 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。

在上例中,12个元组可列成一张二维表

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

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

$$R (D_1, D_2, ..., D_n)$$

R: 关系名

n: 关系的目或度 (Degree)



2) 元组

□ 关系中的每个元素是关系中的<u>元组</u>,通常用 *t* 表示。

3) 单元关系与二元关系

- □ 当*n*=1时,称该关系为单元关系(Unary relation)。
- □ 当*n*=2时,称该关系为二元关系(Binary relation)。

4) 关系的表示

□ 关系也是一个二维表,表的每行对应一个元组,表的每 列对应一个域。

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



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)。
- $\square n$ 目关系必有n个属性。



6) 码

□ <u>候选码</u>

- 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<u>候选码(Candidate key)</u>。
- 候选码的诸属性称<u>为主属性(Prime attribute)</u>。
- ■不包含在任何侯选码中的属性称为<u>非主属性(Non-Prime attribute)或非码属性(Non-key attribute)</u>。
- 在最简单的情况下,候选码只包含一个属性。在最极端的情况下,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<u>为主码</u> <u>(Primary key)。</u>



例 在表2.1 的笛卡尔积中许多元组是没有意义的。

一般: 专业与导师: 1:n, 导师与研究生: 1:n

所以,表2.1中的一个子集才是有意义的。

于是:新关系SAP关系可以包含三个元组

{(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)}



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



■注意

- □关系是笛卡尔积的**有限子集**。无限关系在数据库 系统中是无意义的。
- □笛卡尔积不满足交换律,即

$$(d_1, d_2, ..., d_n) \neq (d_2, d_1, ..., d_n)$$

但关系满足交换律,即

$$(d_1, d_2, \dots, d_i, d_j, \dots, d_n) = (d_1, d_2, \dots, d_j, d_j, \dots, d_n)$$

<u>解决方法:为关系的每个列附加一个属性名以取</u> 消关系元组的有序性



8) 基本关系的性质

- ① 列是同质的 (Homogeneous)
 - 每一列中的分量是同一类型的数据,来自同一个域。
- 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 任意两个元组不能完全相同

■ 由笛卡尔积的性质决定

⑤ 行的顺序无所谓

■ 行的次序可以任意交换

⑥ 分量必须取原子值

■ 每一个分量都必须是不可分的数据项。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



二、关系模式(重点)

- 1. 什么是关系模式
- 2. 定义关系模式
- 3. 关系模式与关系



1. 什么是关系模式

- 关系模式 (Relation Schema) 是型, 关系是值
- 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
 - □完整性约束条件
 - □属性间的数据依赖关系集合



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。



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,如:

DOM (SUPERVISOR-PERSON)

- = DOM (POSTGRADUATE-PERSON)
- =PERSON



关系模式通常可以简记为

R(U) 或 $R(A_1, A_2, ..., A_n)$

- □R 关系名
- □ *A*₁, *A*₂, ..., *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
- 关系模式和关系往往统称为关系,通过上下文加以区别。



三、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
- 关系数据库的<u>型</u>与<u>值</u>



关系数据库的型与值

- <u>关系数据库的型</u>: 关系数据库模式 对关系数据库的描述。
- 关系数据库模式包括
 - 若干域的定义
 - 在这些域上定义的若干关系模式
- 关系数据库的值: 模式在某一时刻对应的关系



第二章 关系数据库

- 2.1 关系数据结构及其形式化定义(重点)
- 2.2 关系操作
- 2.3 关系的完整性
- 2.4 关系代数 (重点)



2.2 关系操作

1. 常用的关系操作

- □查询
 - 选择、投影、连接、除、并、交、差、笛卡尔乘积
- □数据更新
 - ■插入、删除、修改
- □查询的表达能力很强,是关系操作中最主要的部分

2. 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

M

3. 关系查询语言的分类

- □ 关系代数语言
 - 用对关系的运算来表达查询要求
 - 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



第二章 关系数据库

- 2.1 关系数据结构及其形式化定义(重点)
- 2.2 关系操作
- 2.3 关系的完整性
- 2.4 关系代数 (重点)



2.3 关系的完整性

- 关系模型的完整性规则是对关系的某种约束条件。
- 关系模型中三类完整性约束:
 - □实体完整性
 - □参照完整性
 - □用户定义的完整性
- 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
 - □ 称作是关系的两个<mark>不变性</mark>,应该由关系系统自动支持。



一、实体完整性

- ■实体完整性规则(Entity Integrity)
 - □ 若属性A是基本关系R的主属性,则属性A不能取空值。

所谓空值就是"不知道"或"无意义"的值。

例

SAP(SUPERVISOR, SPECIALITY, POSTGRADUATE) POSTGRADUATE属性为主码(假设研究生不会重名),则其不能取空值。



■ 对实体完整性规则的说明

- (1) 实体完整性规则是针对基本关系而言的。一个基本表通 常对应现实世界的一个实体集或多对多联系。
- (2) 现实世界中的实体和实体间的联系都是可区分的,即它们具有某种唯一性标识。
- (3) 相应地,关系模型中以主码作为唯一性标识。
- (4) 候选码中的属性即主属性不能取空值。
 - 如果主属性取空值,就说明存在某个不可标识的实体,即存在不可区分的实体,这与第(2)点相矛盾,因此这个规则称为实体完整性。



■注意

□实体完整性规则规定基本关系的所有主属性都 不能取空值,而不仅是主码整体不能取空值。

例:

选修(学号,课程号,成绩)

(学号,课程号)为主码,

则学号和课程号两个属性都不能取空值



二、参照完整性

- 1. 关系间的引用
- 2. 外码
- 3. 参照完整性规则

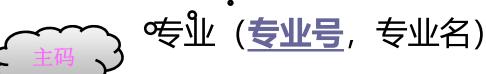


1. 关系间的引用

■ 在关系模型中实体及实体间的联系都是用关系来描述的,因此可能存在着关系与关系间的引用。

例1 学生实体、专业实体

学生(学号,姓名,性别,专业号,年龄)





- ❖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,即专业 关系中有该专业的记录。

学生(学号,姓名,性别,专业号,年龄)

学号	姓名	性别	专业号	年 龄
801	张 三	女	0 1	19
802	李 四	男	0 1	2 0
803	王 五	男	0 1	2 0
8 0 4	赵六	女	0 2	2 0
8 0 5	钱七	男	0 2	19

专业(专业号,专业名)

专业号	专业名
0 1	信息
0 2	数 学
0 3	计 算 机



例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)



学 号	姓 名	性别	专业号	年 龄
8 0 1	张 三	女	0 1	1 9
802	李 四	男	0 1	2 0
803	王 五	男	0 1	2 0
8 0 4	赵六	女	0 2	2 0
805	钱七	男	0 2	19

课程

课程号	课程名	学分
01	数据库	4
02	数据结构	4
03	编译	4
04	PASCAL	2

学生选课

学号	课程号	成绩	
801	04	92	
801	03	78	
801	02	85	
802	03	82	
802	04	90	
803	04	88	



例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	0 1	19	802
802	李四	男	01	20	
803	王五	男	01	20	802
804	赵六	女	02	20	805
805	钱七	男	0 2	19	

- ❖ "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是基本关系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)



[例1] : 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



[例2]:

选修关系的"学号"与学生关系的主码"学号"相对 应;

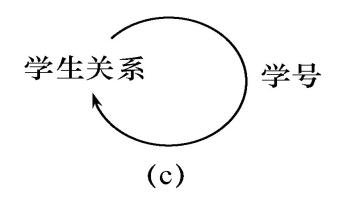
选修关系的"课程号"与课程关系的主码"课程号"相 对应;

- □ "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



[例3]: "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





- 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



3. 参照完整性规则

■ 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。



[例1]:

学生关系中每个元组的"专业号"属性只取两类值:

- (1) 空值,表示尚未给该学生分配专业
- (2) 非空值,这时该值必须是专业关系中某个元组
 - 的"专业号"值,表示该学生不可能分配一个不

存在的专业

M

〔例2〕:

选修(学号,课程号,成绩)

"学号"和"课程号"可能的取值:

- (1) 选修关系中的主属性,不能取空值
- (2) 只能取相应被参照关系中已经存在的主码值



[例3]:

学生(学号,姓名,性别,专业号,年龄,班长) "班长"属性值可以取两类值:

- (1) 空值,表示该学生所在班级尚未选出班长(按照前面给出的具体数据,空值还可表示本人即是班长)
- (2) 非空值,该值必须是本关系中某个元组的学号值



三、用户定义的完整性

- 用户定义的完整性是针对某一具体关系数据库的约束条件,反映某一具体应用所涉及的数据必须满足的语义要求。
- 关系模型应提供定义和检验这类完整性的机制,以便用统一的系统的方法处理它们,而不要由应用程序承担这一功能。

例:

课程(<u>课程号</u>,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}



第二章 关系数据库

- 2.1 关系数据结构及其形式化定义(重点)
- 2.2 关系操作
- 2.3 关系的完整性
- 2.4 关系代数 (重点)



关系代数

- 1. 关系代数
- 2. 运算的三要素
- 3. 关系代数运算的三个要素
- 4. 关系代数运算的分类
- 5. 表示记号



1. 关系代数

- □ 一种抽象的查询语言
- □ 用对关系的运算来表达查询

2. 运算的三要素

- □运算对象
- □运算符
- □运算结果



3. 关系代数运算的三个要素

□运算对象:关系

□运算结果: 关系

□运算符: 四类



□运算符

- 集合运算符
 - □将关系看成元组的集合
 - □从关系的"水平"方向即行的角度来进行运算
- ■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



表 关系代数运算符

运算	符	含义	运算	符	含义
集合	J	并	比	>	大于
	-	差	较	<u>></u>	大于等于
运算	Λ	交	运	<	小于
异 符	×	广义笛	算	<u> </u>	小于等于
1·J		卡尔积		=	等于
			符	< >	不等于



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 Χ	选择 投影 连接 除	逻辑运算符	\ \ \ \ \	非与或



4. 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



5、表示记号

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

设关系模式为 $R(A_1, A_2, ..., A_n)$ 。它的一个关系设为R。 $t \in R$ 表示 $t \not\in R$ 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) A, t[A], A

若 $A=\{A_{i1}, A_{i2}, ..., A_{ik}\}$, 其中 $A_{i1}, A_{i2}, ..., A_{ik}$ 是 $A_1, A_2, ..., A_n$ 中的一部分,则A称为属性列或域列。 $t[A]=(t[A_{i1}], t[A_{i2}], ..., t[A_{ik}])$ 表示元组t在属性列A上诸分量的集合。A则表示 $\{A_1, A_2, ..., A_n\}$ 中去掉 $\{A_{i1}, A_{i2}, ..., A_{ik}\}$ 后剩余的属性组。



(3) $t_r t_s$

R为n目关系,S为m目关系。 $t_r \in R$, $t_s \in S$, $t_r t_s$ 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。



(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:

 $Z_{\mathbf{x}} = \{t[Z] | t \in \mathbb{R}, t[X] = x\}$

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。



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

$$Z_{x1} = \{Z1, Z2, Z3\},$$

■ x₂在R中的象集

$$Z_{x2} = \{Z2, Z3\},$$

■ x₃在R中的象集

$$Z_{x3} = \{Z1, Z3\}$$



求象集举例:

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	
803	王五	男	01	20	802
804	赵六	女	02	20	805
805	钱七	男	02	19	



关系R(X, Z), X, Z是属性组, x是X上的取值, 定义x在R中的象集为

$$Z_x = \{ t[Z] \mid t \in R \land t[X] = x \}$$

从R中选出在X上取值为x的元组,去掉X上的分量,只留Z上的分量。

X	Z	x=刘军	Z_{x}	刘军同学所选修的全部课程
姓名	课程		课程	
刘军	物理		数学	
王红	数学		物理	
刘军	数学		177-2	



2.4 关系代数

2.4.1 传统的集合运算

- □并
- □差
- □交
- □广义笛卡尔积

2.4.2 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

$\blacksquare R \cup S$

□仍为n目关系,由属于R或属于S的元组组成 $R \cup S = \{ t | t \in R \lor t \in S \}$



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

 $R \cup S$

A	B	C
a1	b1	c1
a1	b2	c2
a1	<i>b</i> 3	c2
a2	b2	c1

 A
 B
 C

 s
 a1
 b2
 c2

 a1
 b3
 c2

b2

c1

a2



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

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



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

R-S

A	B	C
a1	b1	c1

S

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



3. 交 (Intersection)

■ R和S

- □具有相同的目n
- □相应的属性取自同一个域

$\blacksquare R \cap S$

□仍为n目关系,由既属于R又属于S的元组组成 $R \cap S = \{ t | t \in R \land t \in S \}$ $R \cap S = R - (R-S)$



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

 $R \cap S$

A	В	С
a1	b2	c2
a2	b2	c1

s a1 b2 c2 a1 b3 c2

B

A

a2 b2 c1

C

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

- R
 - □n目关系, k_1 个元组
- - □m目关系,k₂个元组
- R×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×*k*₂个元组

$$R \times S = \{ t_r t_s \mid t_r \in R \land t_s \in S \}$$



	A	В	C
R	a1	b1	c1
	a1	b2	c2
	a2	b2	c1

	A	В	С
S	a1	b2	c2
	a1	b3	c2
	a2	b2	c1

	A	В	C	A	B	C
	a1	b1	c1	a1	b2	c2
	a1	b1	c1	a1	b3	c2
	a1	b1	c1	a2	b2	c1
$R \times S$	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



2.4 关系代数

- 2.4.1 传统的集合运算
- 2.4.2 专门的关系运算
 - □选择
 - □投影
 - □连接
 - □除

þΑ

1. 选择 (Selection)

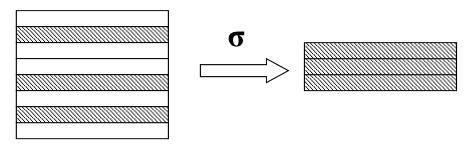
- 1) 又称为限制(Restriction)
- 2) 选择运算符的含义
 - □ 在关系R中选择满足给定条件的诸元组

$$\sigma_{\mathsf{F}}(R) = \{t | t \in R \land F(t) = '\underline{\mathbf{p}}'\}$$

- □ F: 选择条件, 是一个逻辑表达式, 基本形式为:
 - $[\neg(]X_1\theta Y_1[])[\varphi[\neg(]X_2\theta Y_2[])]...$
 - θ: 比较运算符 (> , ≥, < , ≤, =或<>)
 - X₁, Y₁等:属性名、常量、简单函数;属性名也可以用它的序号来代替;
 - φ: 逻辑运算符 (∧或∨)
 - []: 表示任选项
 - …: 表示上述格式可以重复下去



3) 选择运算是从行的角度进行的运算



4) 举例

设有一个学生-课程数据库,包括学生关系 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
2014001	李勇	男	20	CS
2014002	刘晨	女	19	IS
2014003	王敏	女	18	MA
2014004	张立	男	19	IS

(a)



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	PASCAL语言	6	4

Course



学 号	课程号	成 绩
Sno	Cno	Grade
2014001	1	92
2014001	2	85
2014001	3	88
2014002	2	90
2014002	3	80

SC

(c)



[例1] 查询信息系 (IS系) 全体学生

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

或 $\sigma_{5='|S'}$ (Student)

结果:

Sno	Sname	Ssex	Sage	Sdept
2014002	刘晨	女	19	IS
2014004	张立	男	19	IS



[例2] 查询年龄小于20岁的学生

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

或 $\sigma_{4<20}$ (Student)

结果:

Sno	Sname	Ssex	Sage	Sdept
2014002	刘晨	女	19	IS
2014003	王敏	女	18	MA
2014004	张立	男	19	IS



2. 投影 (Projection)

1) 投影运算符的含义

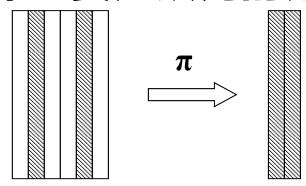
□从R中选择出若干属性列组成新的关系

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

A:R中的属性列



2) 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)



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

π_{Sname, Sdept}(Student) 或 π_{2, 5}(Student)

结果:

Sname	Sdept
李勇	CS
刘晨	IS
王敏	MA
张立	IS



[例4] 查询学生关系Student中都有哪些系 π_{Sdept}(Student)

结果:

Sdept
CS
IS
MA



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

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

- 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*×*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。



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:

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

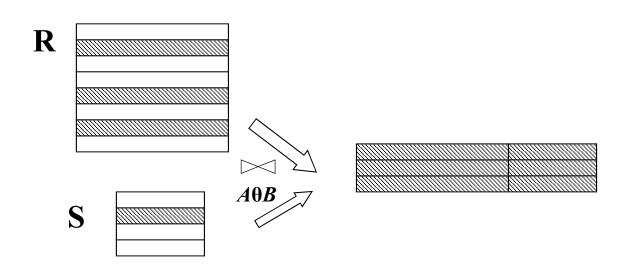


- □自然连接(Natural join)
 - 什么是自然连接
 - □ 自然连接是一种特殊的等值连接
 - 两个关系中进行比较的分量必须是相同的属性组
 - 在结果中把重复的属性列去掉
 - ■自然连接的含义

R和S具有相同的属性组B

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

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。



5) 举例

[例5]

A	В	C
a_1	b ₁	5
a_1	b ₂	6
a_2	b ₃	8
a_2	b ₄	12

В	E
\boldsymbol{b}_1	3
b_2	7
h	10
b_3	10
b_3	2

R

S



$R \underset{C \leq E}{\bowtie} S$

A	R.B	C	S.B	E
a_1	b ₁	5	\boldsymbol{b}_2	7
a_1	\boldsymbol{b}_1	5	b_3	10
a_1	b_2	6	$\boldsymbol{b_2}$	7
a_1	b_2	6	b_3	10
a_2	b_3	8	b_3	10



等值连接 R $\bowtie_{R.B=S.B}$ S

$oxed{A}$	R.B	<i>C</i>	S.B	E
<i>a</i> ₁	\boldsymbol{b}_1	5	\boldsymbol{b}_1	3
a_1	\boldsymbol{b}_2	6	\boldsymbol{b}_2	7
a_2	b_3	8	b ₃	10
a_2	b_3	8	b_3	2



自然连接 $R \bowtie S$

A	В	C	E
a_1	\boldsymbol{b}_1	5	3
a_1	\boldsymbol{b}_2	6	7
a_2	b_3	8	10
a_2	b ₃	8	2



4. 除 (Division)

关系R(X, Z), X, Z是属性组, x是X上的取值, 定义x在R中的象集为

$$Z_{x} = \{ t[Z] \mid t \in R \land t[X] = x \}$$

从R中选出在X上取值为x的元组,去掉X上的分量,只留Z上的分量。

X	Z	x=刘军	\mathbf{Z}_{x}	刘军同学所选修的全部课程
姓名	课程		课程	
刘军	物理		数学	思考:如何得到选修
王红	数学		物理	了全部课程的学生?
刘军	数学		177	



R

姓名	课程
刘军	物理
王红	数学
刘军	数学
王红	物理

S

课程 数学 物理 做法:逐个考虑*选课关系* 中的元组r,求r在*姓名*上的分量x,再求x在*选课 关系*中的象集*课程*、若 *课程*、包含了所有的课程,则x是满足条件的一个元组。

1) 除定义

□ 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:元组在 *X* 上分量值 *x*的象集 *Y*, 包含 *S*在 *Y*上投影的集合。

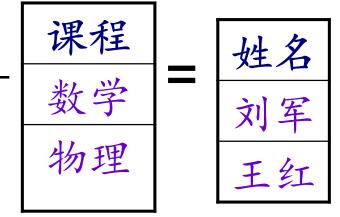
$$R \div S = \{t_r[X] \mid t_r \in R \land \pi_Y(S) \subseteq Y_X\}$$

 Y_x : x在R中的象集, $x = t_r[X]$

R

姓名	课程
刘军	物理
王红	数学
刘军	数学
王红	物理

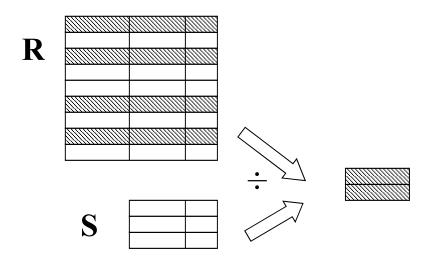
S



选修了全部 课程的学生



2) 除操作是同时从行和列角度进行运算





	\boldsymbol{A}	В	\boldsymbol{C}
	a_1	\boldsymbol{b}_1	c_2
	a_2	b_3	c_7
R	a_3	b_4	c_6
	a_1	b_2	c_3
	a_4	\boldsymbol{b}_6	c_6
	a_2	b_2	c_3
	a_1	\boldsymbol{b}_2	c_1

В	C	D
b ₁	c_2	d_1
b_2	c_1	d_1
b_2	c_3	d_2

$R \div S$
\boldsymbol{A}
a_1



-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(B, C) $_{a_1}$ 包含了S在(B, C)属性组上的投影,所以 $R \div S = \{a_1\}$



课堂练习

设关系订购和零件数据如下表所示, 求订购÷零件。

订购关系

工程号	零件号	数量
a1	b1	58
a2	b1	43
а3	b4	678
a1	b2	65
a4	b6	65
a2	b2	43
a1	b2	58

零件关系

零件号	零件名	颜色
b1	螺母	红色
b2	螺钉	蓝色



5. 综合举例

以学生-课程数据库为例

[例7]查询至少选修1号课程和3号课程的学生号码。

首先建立一个临时关系K:

然后求: π_{Sno, Cno}(SC)÷K

Cno	
1	
3	



例7 (续) π_{Sno.Cno}(SC)

2014001象集{1, 2, 3};

2014002象集{2, 3}

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

于是:

 $\pi_{Sno,Cno}(SC) \div K = \{2014001\}$

Sno	Cno
2014001	1
2014001	2
2014001	3
2014002	2
2014002	3

[例8] 查询选修了2号课程的学生的学号。

$$\pi_{\text{Sno}}$$
 ($\sigma_{\text{Cno}='2'}$ (SC)) = { 2014001, 2014002}

[例9] 查询至少选修了一门其直接先行课为5号课程的课程的学生姓名。

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

或

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

或

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

[例10] 查询选修了全部课程的学生号码和姓名。

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

[例11] 设有关系

S#	C#	SCORE
P01	001	96
P02	001	90
P03	002	88
P01	003	92

(1) 求未选修001号课程的学生号。

方案1: ∏ S# (σC#≠ '001' (SC))

方案2: $\prod_{S\#}$ (SC) - $\prod_{S\#}$ ($\sigma_{C\# = '001'}$ (SC))

确?

S#	C#	SCORE
P01	001	96
P02	001	90
P03	002	88
P01	003	92



(2) 求仅选修了001号课程的学生号。

分析: 选修001号课程的学生 - 不仅选001号课程的学生

(2) 求仅选修了001号课程的学生号。

分析: 选修001号课程的学生 - 不仅选001号课程的学生

$$= \prod_{S\#} (\sigma_{C\#} = \gamma_{001}, (SC)) - \prod_{S\#} (SC - \sigma_{C\#} = \gamma_{001}, (SC))$$

S#	C#	SCORE
P01	001	96
P02	001	90
P03	002	88
P01	003	92

S#	C#	SCORE
P01	001	96
P02	001	90

S#	C#	SCORE
P03	002	88
P01	003	92



说明

- (1) $\sigma_{C# = '001'}$ (SC)把SC表中选001号课程的同学记录都挑出来(包括也选其他课程的同学)
- (2) SC中包含了选所有课程学生记录,用它减去第一步,如果还有记录,则这个记录的同学就是不仅选了001



关系代数应用课堂练习

- 设商品销售数据库有三个关系:商品关系、售货员 关系和售货关系。三个关系的关系模式如下:
 - □商品(商品编号,商品名,产地,价格,等级)
 - □售货员(售货员编号,姓名,性别,年龄)
 - □售货(商品编号,售货员编号,数量)
- 1. 查询等级是'一等品'的所有商品信息。
- 2. 查询性别为男的所有售货员的编号和姓名。
- 3. 查询售出商品编号为'K002'的售货员的姓名。
- 4. 查询曾经销售过所有商品类别的售货员的编号和姓名。