

西北工业大学

Northwestern Polytechnical University

数据库系统原理

Database System

第二章 关系数据库

赵晓南

2024.09

- 数据库系统的**四个基本概念** (Data, DB, DBMS, DBS)
- 数据处理技术的发展 (人工管理、文件系统、数据库系统)
- 数据库系统的发展 (层次、网状、关系、新型)
- 数据库系统的**四个特点** (结构化、共享性、独立性、统一管控)
- 数据模型 (概念模型与逻辑模型、**数据模型要素**)
- **概念模型的核心 (实体与联系、ER图简介)**
- 关系模型的基本概念 (二维表、元组、属性、分量、域)
- 数据库系统体系结构: 三级模式和两层映像

关系模型

属性 (Attribute) 分量

主码 (Key)

元组 (Tuple)

关系 (Relation)

学 号	姓 名	性 别	专 业 号	年 龄
8 0 1	张 三	女	0 1	1 9
8 0 2	李 四	男	0 1	2 0
8 0 3	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	1 9

关系模式：对关系结构的描述，定义如下：

关系名 (属性1, 属性1, ..., 属性n)

例： 学生 (学号, 姓名, 年龄, 性别, 系, 年级)

1. 关于DBS中各类人员职责描述有误的是：

- ☐ A 应用程序员负责应用程序的编写与调试
- ☐ B 用户除了使用应用系统的界面或者命令式用户接口访问数据，也可以通过自己编程的方式访问数据库
- ☒ C 数据库设计人员不需要参加用户需求分析
- ☐ D 监控数据库的使用与运行是DBA的重要职责

提交

2.关于层次模型和网状模型，以下表述错误的是：

- ☐ A 层次模型中，根以外的其它结点有且只有一个双亲结点
- ☐ B 网状模型中，一个结点可以有多于一个的双亲节点
- ☐ C 层次模型和网状模型中，数据之间的联系可以通过链表的方式实现
- ☒ D 层次模型无法直接表达多对多的联系

提交

3.数据库系统与文件系统的根本区别是:

- ☒ A 数据结构化
- ☐ B 数据的共享性高,冗余度低,易扩充
- ☐ C 数据独立性高
- ☐ D 数据由DBMS统一管理和控制

提交

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.关系演算



➤重点

- ✓关系的三个完整性
- ✓关系代数
- ✓关系演算（选学）

➤难点

- ✓关系代数的综合运用
- ✓关系演算（选学）



- **关系** 
 - 域 (Domain)
 - 笛卡尔积 (Cartesian Product)
 - 关系 (Relation)
- 关系模式
- 关系数据库
- 关系模型的存储结构

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
指定字符串集合{ '男' , '女' }

• 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取值范围。 D_1, D_2, \dots, D_n 的笛卡尔积为:

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

• (d_1, d_2, \dots, d_n) ----- **(n) 元组** (n-tuple)

• 元组的每一值 d_i ----- **分量** (Component)

• **D_i 为有限集时**, 其基数为 m_i , 则笛卡尔积的**基数**为 $M = m_1 * m_2 * \dots * m_n$

• 笛卡尔积：例1

$A=\{a, b\}$, $B=\{1, 2, 3\}$ 则

$$A \times B = \{(a,1), (a,2), (a,3), (b,1), (b,2), (b,3)\}$$

$$B \times A = \{(1,a), (2,a), (3,a), (1,b), (2,b), (3,b)\}$$

注：1. 从每一集合中抽一个元素做组合。
2. 笛卡尔积没有交换律。



笛卡尔积 例2:

D1=导师集合 SUPERVISOR = 张清玫,刘逸

D2=专业集合 MAJOR = 计算机专业,信息专业

D3=研究生集合 POSTGRADUATE = 李勇, 刘晨, 王敏

$$D1 \times D2 \times D3 = ?$$

张清玫

刘逸

计算机专业

信息专业

李勇

刘晨

王敏

三者的笛卡尔积:

$2*2*3 = 12$ 个元组

SUPERVISOR

MAJOR

POSTGRADUATE

张清玫

张清玫

张清玫

张清玫

张清玫

张清玫

刘逸

刘逸

刘逸

刘逸

刘逸

刘逸

计算机专业

计算机专业

计算机专业

信息专业

信息专业

信息专业

计算机专业

计算机专业

计算机专业

信息专业

信息专业

信息专业

李勇

刘晨

王敏

李勇

刘晨

王敏

李勇

刘晨

王敏

李勇

刘晨

王敏

哪些元组是
没意义的呢?

关系：域 $D_1 \times D_2 \times \dots \times D_n$ 的子集称为这组域上的关系。

表示为： $R (D_1, D_2, \dots, D_n)$ 。

- R关系名；n为关系的度或目 (Degree)。
- $n=1$, 单元关系 (unary relation); $n=2$, 二元关系 (Binary relation)。
- 关系为笛卡尔积的有限子集，其也为二维表，行对应元组，列对应于域。
- 由于域可以相同，不同的列用不同的名字区分称为属性。

笛卡尔积的结果中有许多元组是无意义的，
其中有意义的元组构成关系，为实际的二维表。

关系分类

- ◆基本关系（基本表）-----数据库中的表
- ◆查询表-----查询结果集（临时表）
- ◆视图-----多个表导出（虚表）

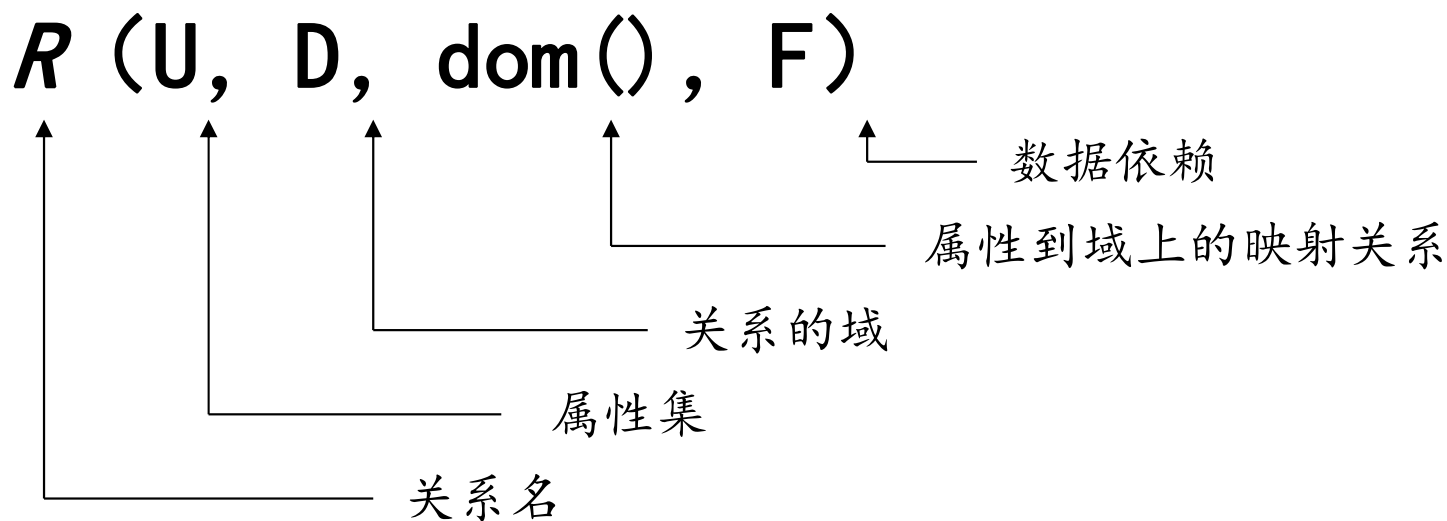
关系应具有六条性质

1. 列是同质的；
2. 不同的列可以出自同一个域；
3. 列序无关性；
4. 任意两个元组不能完全相同；
5. 行序无关性；
6. 分量必须取原子值。

➤ 关系的型与值

- 关系的**型**：关系的结构(字段名、字段个数、域等)
- 关系的**值**：关系中具体的元组，关系的实例。

➤ 关系模式(Relation Schema)即关系的**型的定义**



关系模式通常简记为：**R (U)**

2.1.2 关系模式 - 举例



关系模式举例： 关系名： SAP

SUPERVISOR	MAJOR	POSTGRADUATE
张清玫	计算机专业	李勇
张清玫	计算机专业	刘晨
刘逸	计算机专业	王敏

关系模式记为： $SAP(SUPERVISOR, MAJOR, POSTGRADUATE)$

$dom(SUPERVISOR) = dom(POSTGRADUATE) = PERSON$

dom: 说明各个属性分别出自哪个域

2.1.2 关系模式 - 几个重要概念



几个重要概念:

- 超码(Super Key): 任意一个候选码的超集。
- 候选码(Candidate Key): 能唯一标识元组的属性(组)。
- 主码(Primary Key): 多个候选码中选定一个作主码。
- 主属性(Prime Attribute): 候选码中的诸属性。
- 非主属性(Non-Key Attribute): 不出现在任何候选码中的属性。

选课

学号 Sno	课程号 Cno	成绩 Grade	开课学期 Semester	教学班 Teachingclass
95001	1	92	20201	1001-1
95001	2	85	20212	1102-2
95002	2	90	20221	0501-1
95002	3	80	20231	0803-2

学生

学号 Sno	姓名 Sname	性别 Sgender	出生日期 Sbirthdate	主修专业 Smajor
95001	李勇	男	2000-1-2	CS
95002	刘晨	女	2001-3-4	IS
95003	王敏	女	2003-9-9	MA
95004	张立	男	2002-10-31	IS

- ◆关系模式-----型。对关系的描述，**静态的、稳定的**。
关系的结构(字段名、字段个数、域等)
- ◆关系-----值(内容)。**动态的**，具体的一个实例。

关系模式与关系有时混用，但可从上下文进行区分。

关系数据库:

型: 实体、联系 -> 关系(表)，如：学生表、选课表

值: 某一时刻，这些对应关系表的集合 => 关系数据库

2.1.4 关系模型的存储结构



◆ 关系模型的存储结构（一个或者若干文件）

mysql5.7: 每个表对应两个操作系统文件
(.frm + .ibd)

The image shows two screenshots illustrating the storage structure of a MySQL table. The left screenshot shows the MySQL Navigator interface with the 'student' database selected. The right screenshot shows a Windows Explorer window displaying the files for the 'student' database, specifically the 'c' table. The files are listed in a table:

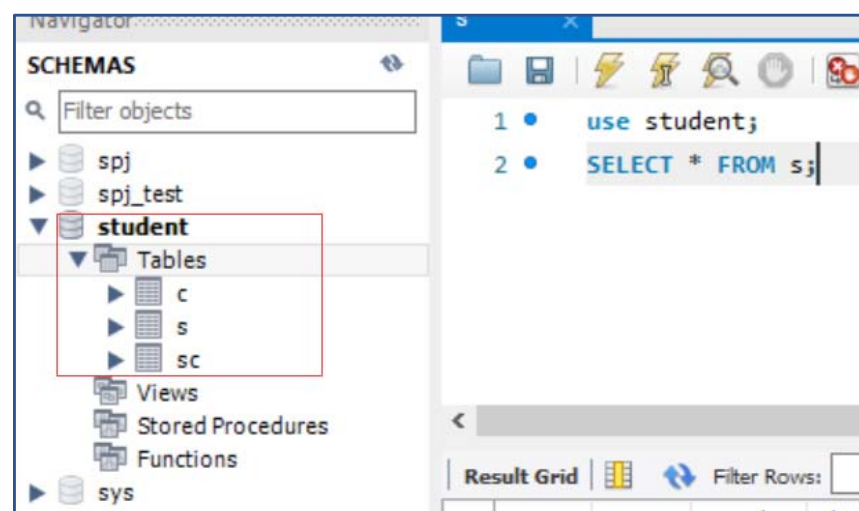
名称	修改日期	类型	大小
c.frm	2014/9/19 14:41	FRM 文件	9 KB
c.ibd	2014/9/19 14:42	IBD 文件	96 KB
db.opt	2014/9/19 14:13	OPT 文件	1 KB
s.frm	2014/9/19 14:16	FRM 文件	9 KB
s.ibd	2014/9/19 14:17	IBD 文件	96 KB
sc.frm	2014/9/19 14:40	FRM 文件	9 KB
sc.ibd	2014/9/19 14:42	IBD 文件	128 KB

2.1.4 关系模型的存储结构



关系模型的存储结构

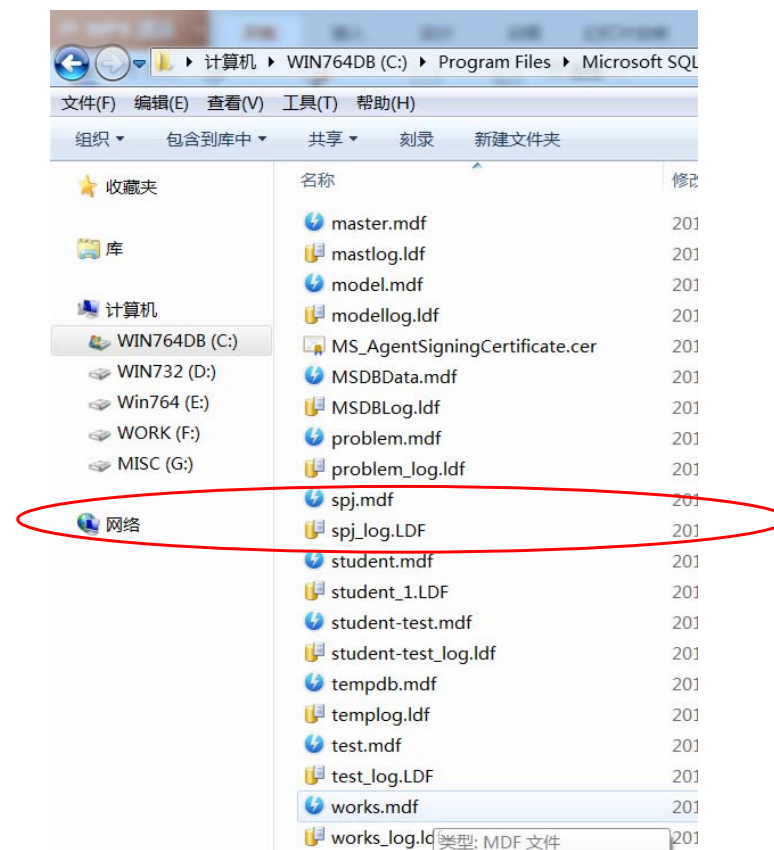
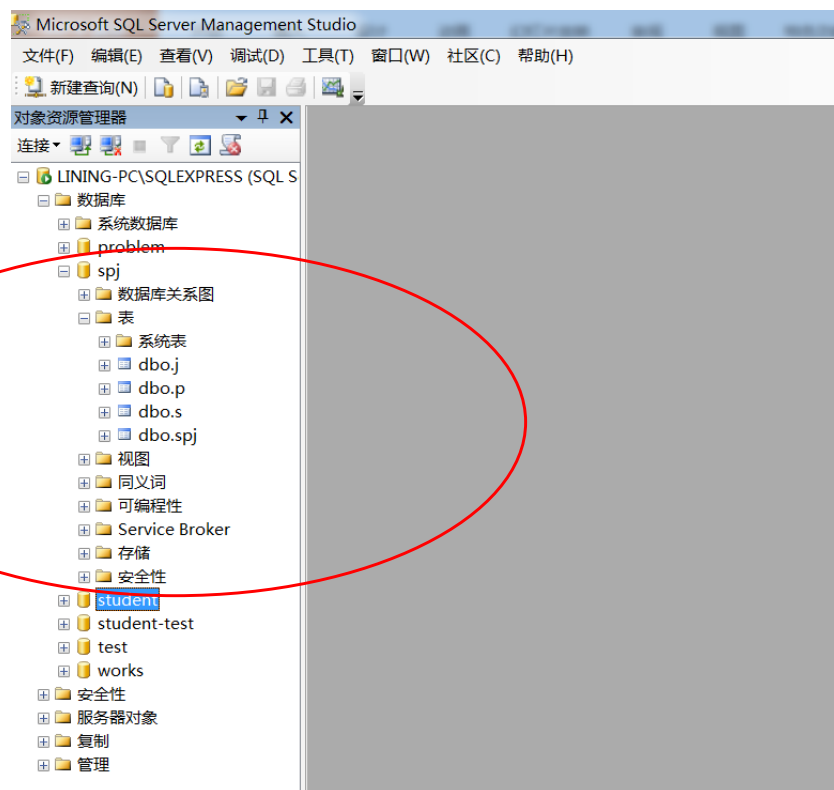
mysql8.0: 每个表对应一个ibd文件



« MySQL Server 8.0 » Data » student					搜索"student"
<input type="checkbox"/>	名称	修改日期	类型	大小	
<input type="checkbox"/>	c.ibd	2020/9/20 16:34	IBD 文件	112 KB	
<input type="checkbox"/>	s.ibd	2020/9/20 16:34	IBD 文件	112 KB	
<input type="checkbox"/>	sc.ibd	2020/9/20 16:34	IBD 文件	128 KB	

◆ 关系模型的存储结构

sqlserver: 一个数据库对应一个大的文件:



关于关系和关系模型，以下表述错误的是：

- ☐ A 在数据库中，笛卡尔积无序 即 $A \times B = B \times A$
- ☐ B 一个关系中，有可能存在多个候选码
- ☒ C 假设关系 $SC(sno, cno, grade)$ ，主属性是 (sno, cno) 的组合
- ☐ D 关系的型是静态的，值是动态的。

提交

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.关系演算

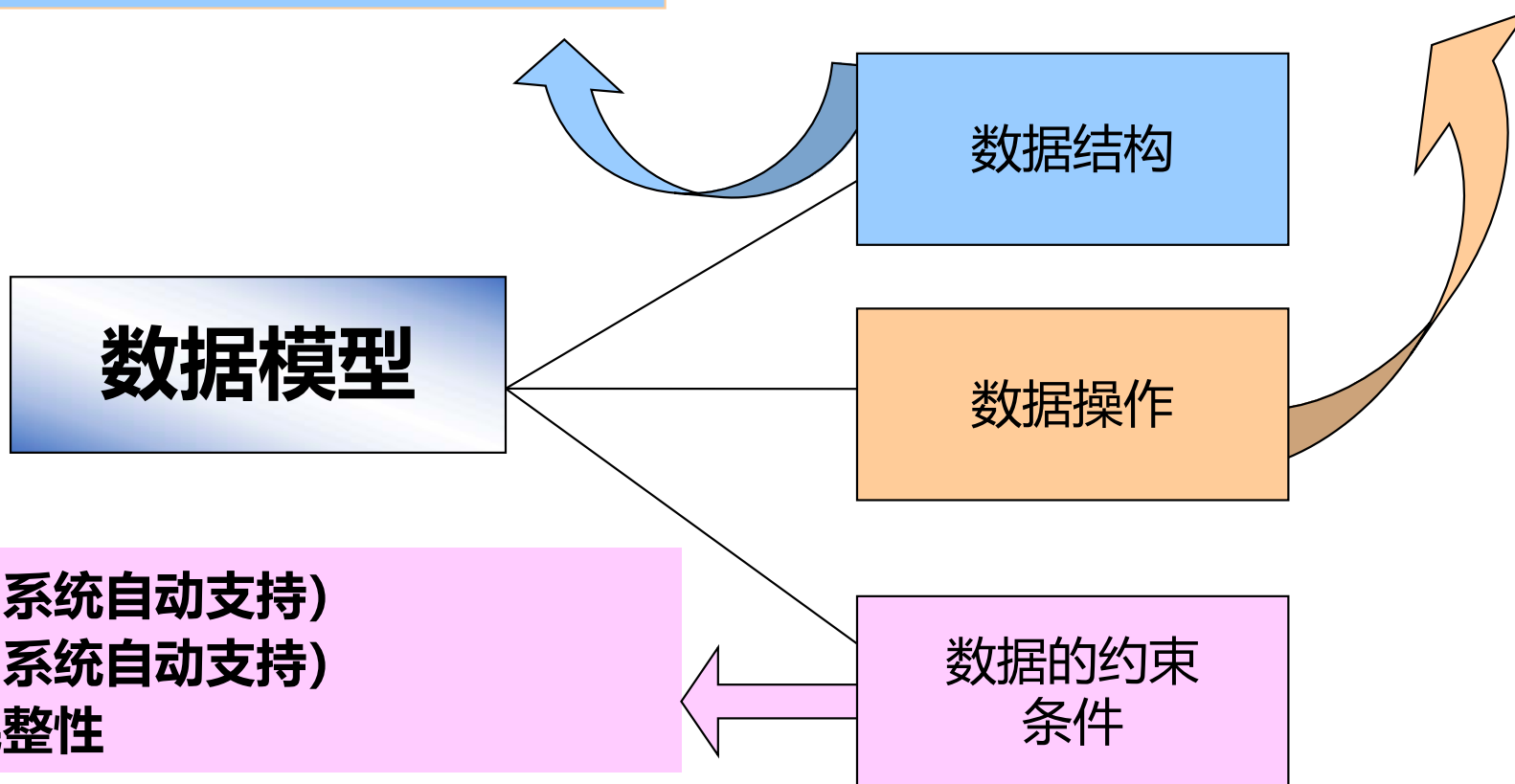


2.2.1 关系操作的核心要素



单一数据结构:
关系--- 实体以及联系都用关系表示

• **查询:** 选择, 投影, 连接, 除, 并, 交, 差, 笛卡尔积
• **更新:** 增加, 删除, 修改



• **实体完整性** (系统自动支持)
• **参照完整性** (系统自动支持)
• **用户定义的完整性**

关系操作

特点：集合操作方式、操作的对象和结果都是集合

关系数据语言的种类

关系代数语言——关系运算（代数方式）

关系演算语言——谓词变元（逻辑方式）

三种关系数据语言在表达能力上是等价的，都具有关系完备性

（Query By Example-通过例子进行查询）

具有关系代数和关系演算双重特点的语言——SQL

关系数据语言SQL的特点：

- 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

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.关系演算



完整性实质就是对关系的一些约束条件

■ **实体完整性**：若属性A是基本关系R的主属性，则属性A不能取空值

1. 实体完整性规则是针对**基本关系**而言的。
2. 现实世界中的实体是可区分的。
3. 关系模型中以主码作为唯一性标识。
4. 主码中的属性即主属性不能取空值。

学号	姓名	性别	班级编号
012134	李长江	男	10102
002321	方虹	男	10101
011111	毛中华	男	10201
...

■ 参照完整性

- 外码(Foreign key)
- 参照关系 (Referencing Relation)
- 被参照关系(Referenced Relation)

• F是基本关系R的一个或一组属性。
• F不是关系R的码。
• F与基本关系S的主码Ks相对应。
则称F是基本关系R的**外码**

F: 类型编号(R的外码)
R: 房间(房间号,电话,**类型编号**)
S: 房间类型(**类型编号**,类型名,价格)

基本关系R为**参照关系**

基本关系S为**被参照关系-目标关系**

- ✓ 关系R和S不一定是不同的关系。
 - ✓ 被参照关系S的主码Ks 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

■ 参照完整性

例1 学生实体和学院实体的联系

学生(学号,姓名,性别,**学院编号**,年龄)
学院(学院编号,学院名)

外码

参照关系

被参照关系

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)

被参照关系

参照关系

外码

外码

例3 同一关系内部的属性引用

学生(学号,姓名,性别,专业号,年龄,**班长学号**)

外码



■ 参照完整性

➤ 外码取值的可能情况:

1) 空(NULL);

2) 目标关系中存在的值;

*表内属性间的参照也要有存在的值;

为什么需要空值?

在进行关系操作时, 有时关系中的某属性值在当前是填不上, 比如档案中有“生日不详”、“下落不明”、“日程尚待公布”等, 这时就需要空值来代表这种情况。

学生(学号, 姓名, 性别, **学院编号**, 年龄)

学院(学院编号, 学院名)

学院编号:

- 空值: 表示尚未给该学生分配学院
- 非空值: 必须是确实存在的一个学院



■ 参照完整性（包含NULL值的兼容性）

□ 数据库中有空值，会影响许多方面，如影响聚集函数运算的正确性，不能参与算术、比较或逻辑运算等

□ $5 + ?$

□ $10 > ?$

□ 聚集函数：求平均数？某个人缺考没有成绩，算与不算该生的均值？

■ 用户定义的完整性

根据客观实际的一些约束条件。

例：性别(男,女)

学号范围：2021001-2021999

学号	姓名	性别
2021001	李长江	男
2021002	方虹	男
2021003	中华	男
...

关系模型应提供定义和检验这类完整性的机制，以使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。**通常由RDBMS的Check约束提供这类检查。**



希望在一个社交网络里找到最大深度为5的朋友的朋友。假设随机选择两个人，是否存在一条路径，使得关联他们的关系长度最多为5？对于一个包含100万人，每人约有50个朋友的社交网络，我们就以典型的开源图数据库**Neo4j**参与测试，结果下表所示。

深度2：希望查询朋友的朋友时，所需要的查询耗时？

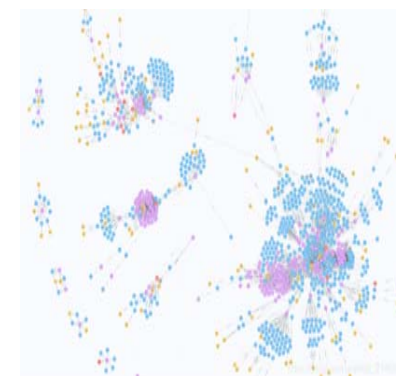
深度	关系数据库执行时间(s)	Neo4j执行时间 (s)	返回的最大记录数
2	0.016	0.01	2500
3	30.267	0.168	110 000
4	1543.505	1.359	600 000
5	?	2.132	800 000

问题选项：见下页

本例子来源于<http://www.iteye.com/news/32186>

深度	关系数据库执行时间(s)	Neo4j执行时间 (s)	返回的最大记录数
2	0.016	0.01	2500
3	30.267	0.168	110 000
4	1543.505	1.359	600 000
5	?	2.132	800 000

- ☐ A 189537.298 (约52小时)
- ☐ B 1895370.298(约520小时, 21天)
- ☒ C 未完成



提交

2.1.关系数据结构及形式化定义

2.2.关系操作

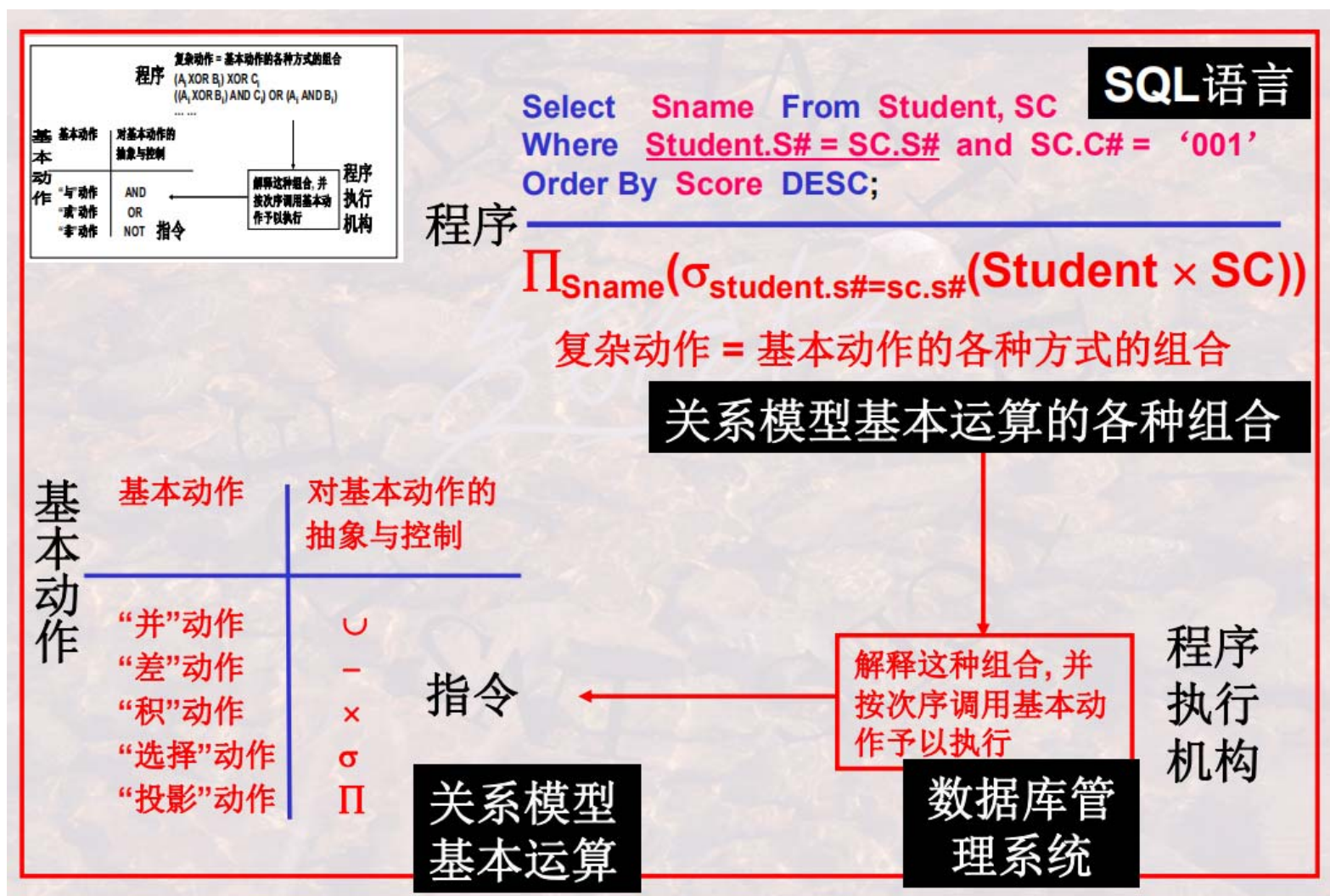
2.3.关系的完整性

2.4.关系代数

2.5.关系演算



■为何提出关系代数(Relational Algebra)?



2.4 关系代数 - 运算符



运 算 符		含 义
集运算 合符	\cup	并
	$-$	差
	\cap	交
专门的关系运算符	\times	广义笛卡尔积
	σ	选择
	π	投影
	\bowtie	连接
	\div	除
比较运算符	$>$	大于
	\geq	大于等于
	$<$	小于
	\leq	小于等于
	$=$	等于
	\neq	不等于
逻辑运算符	\neg	非
	\wedge	与
	\vee	或

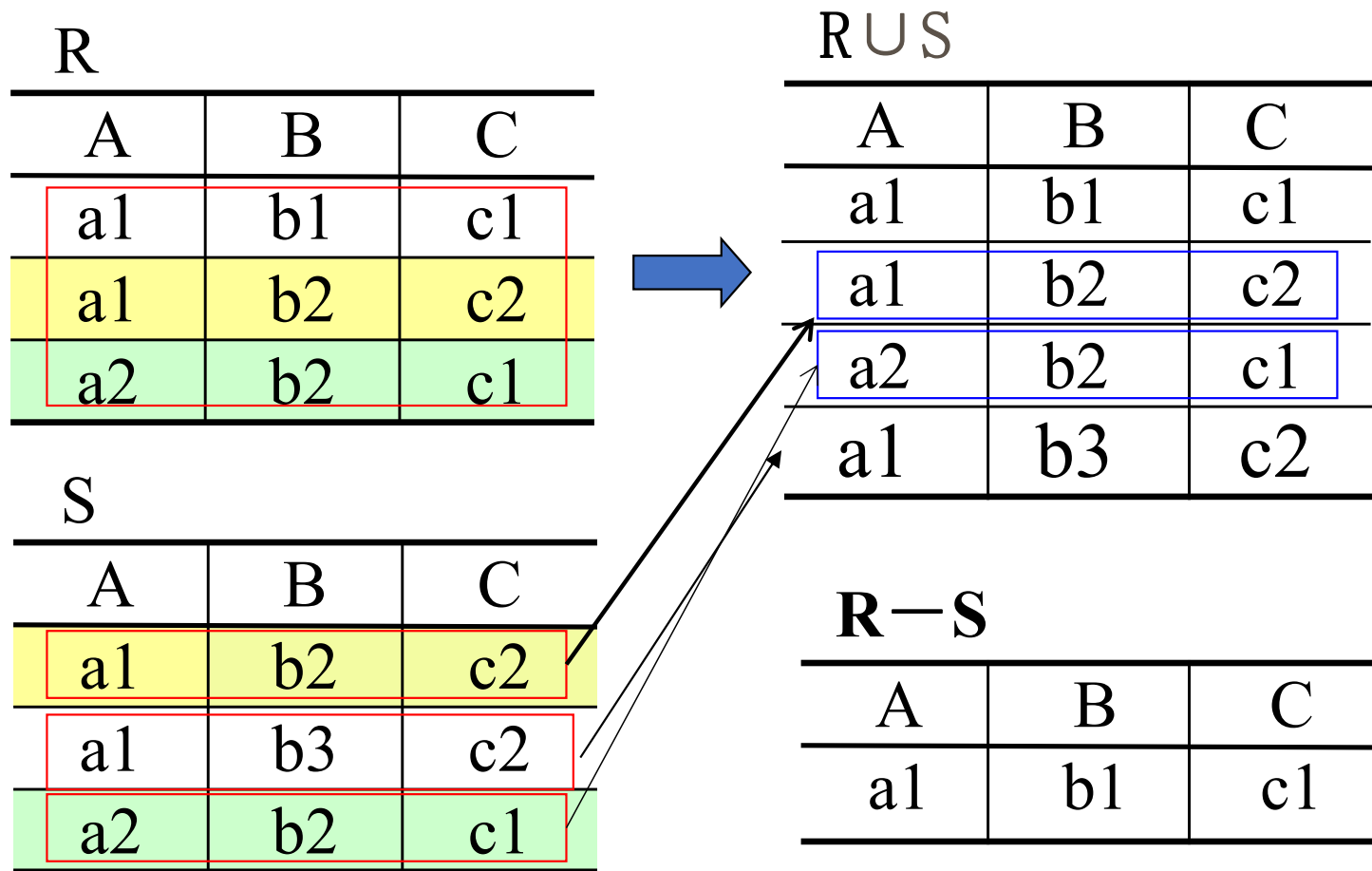
■ 关系代数(Relational Algebra): 以集合为基础, 定义一组运算, 由已知关系经过一系列运算得到需要的查询结果。

■ 并、交、差要求参与运算的两个关系要有相同的结构。

• 关系上的传统集合运算（行运算）

- 并: $R \cup S = \{ t \mid t \in R \vee t \in S \}$
- 差: $R - S = \{ t \mid t \in R \wedge t \notin S \}$
- 交: $R \cap S = \{ t \mid t \in R \wedge t \in S \}$
 $R \cap S = R - (R - S)$
- 广义笛卡尔积: $R \times S$ (其中R为 k_1 行 n 列, S为 k_2 行 m 列)
 - 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
 - $R \times S = \{ t_r \ t_s \mid t_r \in R \wedge t_s \in S \}$

■ 传统的集合运算举例



■ 传统的集合运算举例

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

$R \times S$

R.A	R.B	R.C	S.A	S.B	S.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

■ 并、交、差运算例子随意说



2.4.2 专门的关系运算 - 选择



➤ **选择**：给定条件，选择符合条件的元组；（行的集合）

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

• F为选择条件，为一**逻辑表达式**；

• $F = X_1 \theta Y_1 [\varphi X_2 \theta Y_2 \dots]$

• X1、Y1等是属性名或常量或简单函数

• θ 表示**比较运算符**：>、≥、<、≤、=或≠

• φ 表示**逻辑运算符**：¬、∧或∨。

select

A	B	C
a1	b1	C1
a1	b2	C2
a2	b2	c1

$\sigma_{B=b1 \wedge C=c1}(R)$



A	B	C
a1	b1	C1



σ



➤ 选择运算： 例题

Student

学号 Sno	姓名 Sname	性别 Sgender	出生日期 Sbirthdate	主修专业 Smajor
95001	李勇	男	2000-1-2	CS
95002	刘晨	女	2001-3-4	IS
95003	王敏	女	2003-9-9	MA
95004	张立	男	2002-10-31	IS

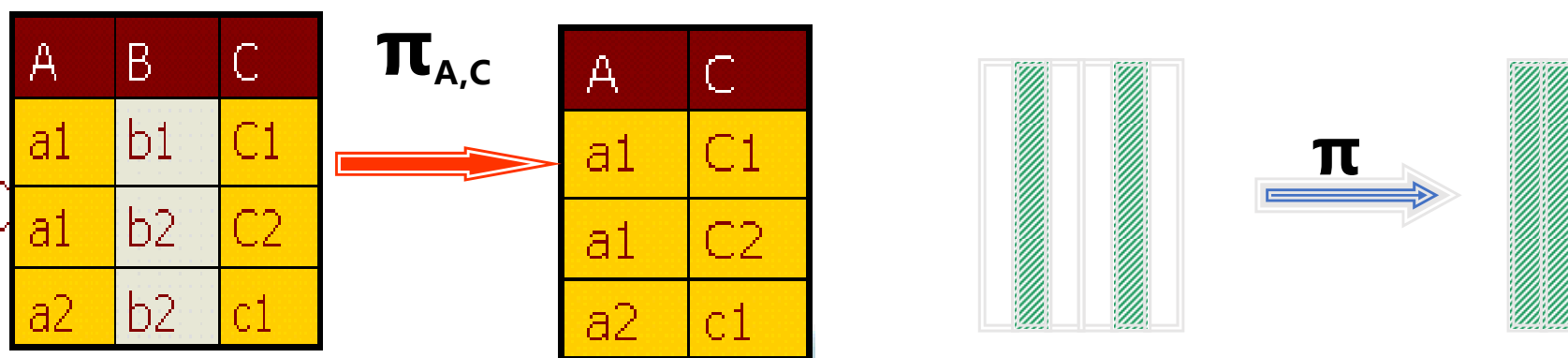
$\sigma_{Smajor='IS'}$ (Student)

Sno	Sname	Sgender	Sbirthdate	Smajor
95002	刘晨	女	2001-3-4	IS
95004	张立	男	2002-10-31	IS

➤ **投影**：从关系中选出若干列构成新的关系（列的运算）

- $\pi_A(R) = \{ t[A] \mid t \in R \}$
- 注：A为一属性组（若干列）

project



注：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）

2.4.2 专门的关系运算 - 投影运算例题



➤ 投影：例题，关系S如下

学号 Sno	姓名 Sname	性别 Sgender	出生日期 Sbirthdate	主修专业 Smajor
95001	李勇	男	2000-1-2	CS
95002	刘晨	女	2001-3-4	IS
95003	王敏	女	2003-9-9	MA
95004	张立	男	2002-10-31	IS

↓ $\pi_{Sno, Sname}(S)$

学 号 Sno	姓 名 Sname
95001	李勇
95002	刘晨
95003	王敏
95004	张立

↓ $\pi_{Smajor}(S)$

Smajor
CS
IS
MA

2.4.2 专门的关系运算 - 综合例题



- 例：学生表Student如下

学 号 Sno	姓 名 Sname	性 别 Sgender	出生日期 Sbirthdate	主修专业 Smajor
95001	李勇	男	2000-1-2	CS
95002	刘晨	女	2001-3-4	IS
95003	王敏	女	2003-9-9	MA
95004	张立	男	2002-10-31	IS

(1) 查询信息安全专业 (IS) 的全体学生

$\sigma_{\text{Smajor} = \text{'IS'}}(\text{Student})$ 或 $\sigma_5 = \text{'IS'}(\text{Student})$

(2) 查询2003年之后出生的学生的学号和姓名

$\pi_{\text{Sno}, \text{Sname}}(\sigma_{\text{Sbirthdate} \geq 2003-1-1}(\text{Student}))$

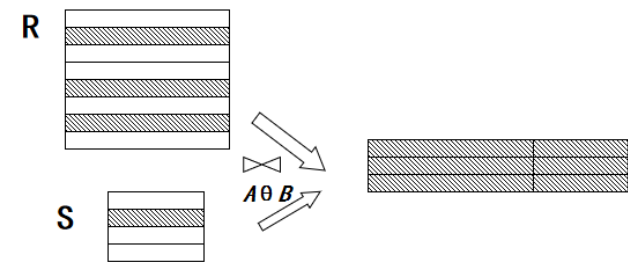
➤ 连接: \bowtie join

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。

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

□ A 和 B : 分别为 R 和 S 上度数相等且的属性组

□ θ : 比较运算符 例: $A > B$



- 等值连接: $A = B$
- 自然连接: AB 为相同属性组, 且去除重复的属性等值。(取消重复列)

2.4.2 专门的关系运算 - 一般连接



R :

A	B	C
a_1	b_1	5
a_1	b_2	6
a_2	b_3	8
a_2	b_4	12

S :

B	E
b_1	3
b_2	7
b_3	10
b_3	2
b_5	2

■ θ 连接解题思路:

1. 确定结果中的属性列
2. 确定参与比较的属性列
3. 逐一取 R 中的元组分别和 S 中与其符合比较关系的元组进行拼接。

$R \bowtie_{C < E} S$

A	$R.B$	C	$S.B$	E
a_1	b_1	5	b_2	7
a_1	b_1	5	b_3	10
a_1	b_2	6	b_2	7
a_1	b_2	6	b_3	10
a_2	b_3	8	b_3	10

2.4.2 等值连接与自然连接



R :

A	B	C
a_1	b_1	5
a_1	b_2	6
a_2	b_3	8
a_2	b_4	12

S :

B	E
b_1	3
b_2	7
b_3	10
b_3	2
b_5	2

$R \bowtie_{B=B} S$

A	$R.B$	C	$S.B$	E
a_1	b_1	5	b_1	3
a_1	b_2	6	b_2	7
a_2	b_3	8	b_3	10
a_2	b_3	8	b_3	2

$R \bowtie S$

A	B	C	E
a_1	b_1	5	3
a_1	b_2	6	7
a_2	b_3	8	10
a_2	b_3	8	2

2.4.2 自然连接的应用



S:

Sno	Sname	Sgender	Sbirthdate	Smajor
95001	李勇	男	2000-1-2	CS
95002	刘晨	女	2001-3-4	IS
95003	王敏	女	2003-9-9	MA
95004	张立	男	2002-10-31	IS

M:

Smajor	Mname
CS	计算机
IS	信息
MA	数学

S ⋈ M

Sno	Sname	Sgender	Sbirthdate	Smajor	Mname
95001	李勇	男	2000-1-2	CS	计算机
95002	刘晨	女	2001-3-4	IS	信息
95003	王敏	女	2003-9-9	MA	数学
95004	张立	男	2002-10-31	IS	信息

2.4.2 外连接



R:

<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

S:

<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



连接中舍弃的元组（悬浮元组），
也进行保留的连接称做外连接

<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
<i>a</i> ₂	<i>b</i> ₄	12	NULL
NULL	<i>b</i> ₅	NULL	2

2.4.2 外连接 - 左外连接

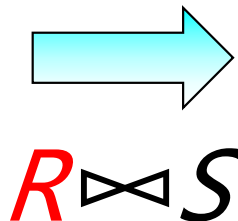


R :

A	B	C
a_1	b_1	5
a_1	b_2	6
a_2	b_3	8
a_2	b_4	12

S :

B	E
b_1	3
b_2	7
b_3	10
b_3	2
b_5	2



保留左边关系 R 中舍弃的元组的连接称做左外连接

A	B	C	E
a_1	b_1	5	3
a_1	b_2	6	7
a_2	b_3	8	10
a_2	b_3	8	2
a_2	b_4	12	NULL

LEFT OUTER JOIN

2.4.2 外连接 - 右外连接

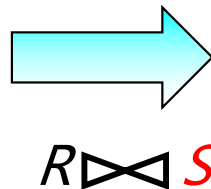


R :

A	B	C
a_1	b_1	5
a_1	b_2	6
a_2	b_3	8
a_2	b_4	12

S :

B	E
b_1	3
b_2	7
b_3	10
b_3	2
b_5	2



保留右边关系 S 中舍弃的元组的连接称做右外连接

A	B	C	E
a_1	b_1	5	3
a_1	b_2	6	7
a_2	b_3	8	10
a_2	b_3	8	2
NULL	b_5	NULL	2

RIGHT OUTER JOIN

- 关系模型的三个组成要素
数据结构，数据操作，完整性约束
- 关系代数
投影，选择，连接

