

# 数据库系统之一

## --基本知识与关系模型

战德臣

哈尔滨工业大学 教授.博士生导师  
黑龙江省教学名师  
教育部大学计算机课程教学指导委员会委员

Research Center on Intelligent  
Computing for Enterprises & Services,  
Harbin Institute of Technology



# 第3讲 关系模型之基本概念

战德臣

哈尔滨工业大学 教授.博士生导师

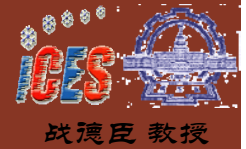
黑龙江省教学名师

教育部大学计算机课程教学指导委员会委员

Research Center on **I**ntelligent  
**C**omputing for **E**nterprises & **S**ervices,  
**H**arbin **I**nstitute of **T**echnology



# 本讲学习什么？



## 基本内容

1. 关系模型概述？
2. 什么是关系？
3. 关系模型中的完整性约束

## 重点与难点

- 一组概念的区分：围绕关系的相关概念，如域、笛卡尔积，关系，关系模式，关键字/键/码，外码/外键，主码/主键，主属性与非主属性。
- 三个完整性：实体完整性，参照完整性和用户自定义的完整性；



# 关系模型简述

战德臣

哈尔滨工业大学 教授.博士生导师

黑龙江省教学名师

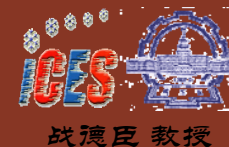
教育部大学计算机课程教学指导委员会委员

Research Center on **I**ntelligent  
**C**omputing for **E**nterprises & **S**ervices,  
**H**arbin **I**nstitute of **T**echnology



# 关系模型简述？

## (1)关系模型的提出？



- 最早由**E.F.Codd**在**1970**年提出
- 是从表(**Table**)及表的处理方式中抽象出来的, 是在对传统表及其操作进行数学化严格定义基础上, 引入**集合理论**与**逻辑学理论**提出的
- 是数据库的三大经典数据模型之一, 也是现在大多数商品化数据库系统所仍然使用的数据模型
- 标准的数据库语言(**SQL语言**)是建立在关系模型基础之上的, 数据库领域的众多理论也都是建立在关系模型基础之上的

班级	课程	教师	学期	学号	姓名	成绩
981101	数据库	李四	98秋	98110101	张三	100
981101	数据库	李四	98秋	98110102	张四	90
981101	数据库	李四	98秋	98110103	张五	80
981101	计算机	李五	98秋	98110101	张三	89
981101	计算机	李五	98秋	98110102	张四	98
981101	计算机	李五	98秋	98110103	张五	72
981102	数据库	李四	99秋	98110201	王三	30
981102	数据库	李四	99秋	98110202	王四	90
981102	数据库	李四	99秋	98110203	王武	78



# 关系模型简述?

## (2)关系模型研究什么



- 形象地说，一个关系(relation)就是一个Table
- 关系模型就是处理Table的，它由三个部分组成：
  - ❑ 描述DB各种数据的基本结构形式(Table/Relation)
  - ❑ 描述Table与Table之间所可能发生的各种操作(关系运算)
  - ❑ 描述这些操作所应遵循的约束条件(完整性约束)
- 就是要学习: Table如何描述，有哪些操作、结果是什么、有哪些约束等?

学生登记表

学号	姓名	性别	出生年月	入学日期	家庭住址
98110101	张三	男	1980.10	1998.09	黑龙江省哈尔滨市
98110102	张四	女	1980.04	1998.09	吉林省长春市
98110103	张五	男	1981.02	1998.09	黑龙江省齐齐哈尔市
98110201	王三	男	1980.06	1998.09	辽宁省沈阳市
98110202	王四	男	1979.01	1998.09	山东省青岛市
98110203	王武	女	1981.06	1998.09	河南省郑州市

操作

有哪些?

学生成绩单

班级	课程	教师	学期	学号	姓名	成绩
981101	数据库	李四	98秋	98110101	张三	100
981101	数据库	李四	98秋	98110102	张四	90
981101	数据库	李四	98秋	98110103	张五	80
981101	计算机	李五	98秋	98110101	张三	89
981101	计算机	李五	98秋	98110102	张四	98
981101	计算机	李五	98秋	98110103	张五	72
981102	数据库	李四	99秋	98110201	王三	30
981102	数据库	李四	99秋	98110202	王四	90
981102	数据库	李四	99秋	98110203	王武	78

结果

是什么?



### 关系模型的三个要素(详细内容在后面讲述)

□ **基本结构:** Relation/Table

□ **基本操作:** Relation Operator

基本的:  $\cup$  (并, UNION)、 $-$  (差, DIFFERENCE)、 $\times$  (广义积, PRODUCT)、 $\sigma$  (选择, SELECTION)、 $\pi$  (投影, PROJECTION)。

扩展的:  $\cap$  (交, INTERSECTION)、 $\bowtie$  (连接, JOIN)、

$\div$  (除, DIVISION) 运算

□ **完整性约束:** 实体完整性、参照完整性和用户自定义的完整性



- 关系运算：关系代数和关系演算；关系演算：元组演算和域演算。
- 关系代数示例：基于集合的运算
- 即：操作的对象及结果都是集合，是一次一集合(**Set-at-a-time**)的操作。而非关系型的数据操作通常是一次一记录(**Record-at-a-time**)的操作

$$\pi_{\text{姓名,课程名}}(\sigma_{\text{课程号}=\text{c2}}(R \bowtie S))$$

- 基于关系代数设计的数据库语言(**ISBL**): 用计算机可识别的符号表征关系代数的运算符号

$$((R * S) : \text{课程号} = \text{c2}) \% \text{姓名,课程名}$$

R:F表示选择运算， R%表示投影运算



- 元组演算示例: 基于逻辑的运算

$$\{ t \mid (\exists u)(R(t) \wedge W(u) \wedge t[3] < u[1]) \}$$

- 基于元组演算设计的数据库语言(Ingres系统的QUEL):用计算机可识别的符号表征元组演算的运算符号

range of t is R  
range of u is W  
retrieve t  
where t.sage < u.sage



- 域演算示例：基于示例的运算

$$\{ t_1, t_2, t_3 \mid S(t_1, t_2, t_3) \wedge R(t_1, t_2, t_3) \wedge t_1 < 20 \wedge t_2 > 30 \}$$

- 基于域演算设计的数据库语言示例：(QBE: Query By Example)

S	S#	Sname	Sage	Sclass	Sex
	<u>S1</u>	P.X			

C	C#	Cname	Cformat	Cteacher	Coffice
	<u>C1</u>	计算机原理			

SC	S#	C#	Semester	Scgrade
	<u>S1</u>	<u>C1</u>		P.Y



# 关系模型简述?

## (5)为什么要学习关系模型与关系数据库语言





- 理解关系(relation) , 理解如何用Relation对Table进行抽象或说严格定义
- 理解关系/表(relation/table)所具有的各种特性 , 理解关系模型
- 熟练掌握关系代数、元组演算和域演算(域演算以QBE为例来学习) , 用这些关系运算来表达各种复杂的检索需求 , 以便于后续SQL语言的学习
- 本讲将要学习 : 关系、关系模型的完整性
- 后续两讲将要学习 : 关系代数和元组演算



# 什么是关系？

--关系的数学定义

战德臣

哈尔滨工业大学 教授.博士生导师

黑龙江省教学名师

教育部大学计算机课程教学指导委员会委员

Research Center on **I**ntelligent  
**C**omputing for **E**nterprises & **S**ervices,  
**H**arbin **I**nstitute of **T**echnology



# 什么是关系？

## (1)什么是“表”？

为什么把“表”称为关系？  
怎样严格定义一个“表”？  
“表”和“关系”有什么异同？



# 什么是关系？

## (2)“表”的基本构成要素

### ➤如何严格地定义Table呢？

表/关系

家庭		
丈夫	妻子	子女
李基	王方	李键
张鹏	刘玉	张睿
张鹏	刘玉	张峰

列/字段/属性/数据项

列名

标题/模式

行/元组/记录

列值



# 什么是关系?

## (3)“表”的严格定义--关系?



➤首先定义“列”的取值范围“域(Domain)”

### 域(Domain)

- 一组值的集合, 这组值具有相同的数据类型
- 如整数的集合、字符串的集合、全体学生的集合
- 再如, 由8位数字组成的数字串的集合, 由0到100组成的整数集合
- 集合中元素的个数称为域的基数(Cardinality)

家庭

丈夫	妻子	子女
李基	王方	李健
张鹏	刘玉	张睿
张鹏	刘玉	张峰

$D_3 = \text{儿童集合(CHILD)} = \{\text{李健, 张睿, 张峰}\}$

$D_2 = \text{女人集合(WOMAN)} = \{\text{王芳, 刘玉}\}$

$D_1 = \text{男人集合(MAN)} = \{\text{李基, 张鹏}\}$



# 什么是关系？

## (2)“表”的严格定义--关系？

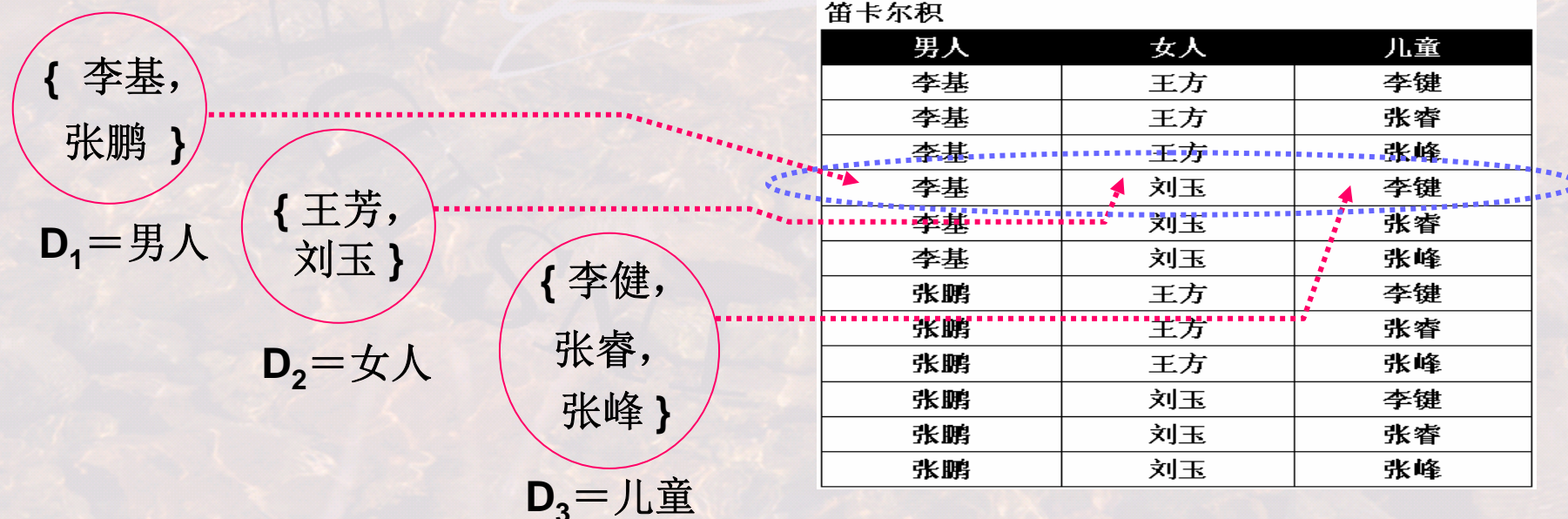
➤再定义“元组”及所有可能组合成的元组：笛卡尔积

## 笛卡尔积(Cartesian Product)

□一组域 $D_1, D_2, \dots, D_n$ 的笛卡尔积为:

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

□笛卡尔积的每个元素 $(d_1, d_2, \dots, d_n)$ 称作一个**n-元组 (n-tuple)**





# 什么是关系？

## (2)“表”的严格定义--关系？

- 元组( $d_1, d_2, \dots, d_n$ )的每一个值 $d_i$ 叫做一个分量(component)
- 元组( $d_1, d_2, \dots, d_n$ )是从每一个域任取一个值所形成的一种组合，笛卡尔积是所有这种可能组合的集合，即：笛卡尔积是由 $n$ 个域形成的所有可能的 $n$ -元组的集合

- 若 $D_i$ 的基数为 $m_i$ ，则笛卡尔积的基数，即元组个数为

$$m_1 \times m_2 \times \dots \times m_n$$

笛卡尔积			
男人	女人	儿童	域名
李基	王方	李键	
李基	王方	张睿	
李基	王方	张峰	
李基	刘玉	李键	
李基	刘玉	张睿	
李基	刘玉	张峰	
张鹏	王方	李键	
张鹏	王方	张睿	域值
张鹏	王方	张峰	
张鹏	刘玉	李键	
张鹏	刘玉	张睿	
张鹏	刘玉	张峰	



# 什么是关系？

## (2)“表”的严格定义--关系？

➤ 由于笛卡尔积中的所有元组并不都是有意义的，因此...

## 关系(Relation)

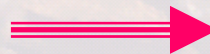
□ 一组域  $D_1, D_2, \dots, D_n$  的笛卡尔积的子集

□ 笛卡尔积中具有某一方面意义的那些元组被称作一个关系(Relation)

□ 由于关系的不同列可能来自同一个域，为区分，需要为每一列起一个名字，该名字即为**属性名**。

笛卡尔积

男人	女人	儿童
李基	王方	李键
李基	王方	张睿
李基	王方	张峰
李基	刘玉	李键
李基	刘玉	张睿
李基	刘玉	张峰
张鹏	王方	李键
张鹏	王方	张睿
张鹏	王方	张峰
张鹏	刘玉	李键
张鹏	刘玉	张睿
张鹏	刘玉	张峰



列名(属性名)

家庭		
丈夫	妻子	子女
李基	王方	李键
张鹏	刘玉	张睿
张鹏	刘玉	张峰

列值：来自域



# 什么是关系？

## (2)“表”的严格定义--关系？



□ 关系可用  $R(A_1:D_1, A_2:D_2, \dots, A_n:D_n)$  表示，可简记为  $R(A_1, A_2, \dots, A_n)$ ，这种描述又被称为**关系模式(Schema)**或表标题(head)

□  $R$  是关系的名字， $A_i$  是属性， $D_i$  是属性所对应的域， $n$  是关系的**度或目(degree)**，关系中元组的数目称为关系的**基数(Cardinality)**

□ 例如下图的关系为一**3目关系**，描述为

家庭(丈夫:男人, 妻子:女人, 子女:儿童)或家庭(丈夫, 妻子, 子女)

家庭		
丈夫	妻子	子女
李基	王方	李键
张鹏	刘玉	张睿
张鹏	刘玉	张峰



# 什么是关系？

## (2)“表”的严格定义--关系？



□关系模式  $R(A_1:D_1, A_2:D_2, \dots, A_n:D_n)$  中属性向域的映象在很多DBMS中一般直接说明为属性的类型、长度等

□ 例如：

**Student**( S# char(8), Sname char(10), Ssex char(2),  
Sage integer, D# char(2), Sclass char(6) )

再如：

**Course** ( C# char(3), Cname char(12), Chours integer,  
Credit float(1), T# char(3) )

**SC**( S# char(8), C# char(3), Grade float(1) )



# 什么是关系？

## (2)“表”的严格定义--关系？

### 关系模式与关系

- 同一关系模式下，可有很多的关系
- 关系模式是关系的结构，关系是关系模式在某一时刻的数据
- 关系模式是稳定的；而关系是某一时刻的值，是随时间可能变化的

**Student**(S# char(8), Sname char(10), Ssex char(2), Sage integer,  
D# char(2), Sclass char(6))

Student					
S#	Sname	Ssex	Sage	D#	Sclass
98030101	张三	男	20	03	980301
98030102	张四	女	20	03	980301
98030103	张五	男	19	03	980301
98040201	王三	男	20	04	980402
98040202	王四	男	21	04	980402
98040203	王五	女	19	04	980402

Student					
S#	Sname	Ssex	Sage	D#	Sclass
98030101	张三	男	20	03	980301
98030102	张四	女	20	03	980301
98030103	张五	男	19	03	980301
98040201	王三	男	20	04	980402
98040202	王四	男	21	04	980402



# 什么是关系?

## (2)“表”的严格定义--关系?

### 思维回顾

家庭		
丈夫	妻子	子女
李基	王方	李键
张鹏	刘玉	张睿
张鹏	刘玉	张峰

#### 2. 值域(Domain)

说清楚每一列数据可能的取值

#### 1. 指出有多少列

#### 4. 指出关系中的元组

关系中元组是有意义的组合  
----笛卡尔积的子集

#### 3. 指出一个元组 及所有可能的元组

元组是值的一个组合; 值域中值的所有可能的组合----笛卡尔积



# 什么是关系？

--关系的特性及相关的概念

战德臣

哈尔滨工业大学 教授.博士生导师

黑龙江省教学名师

教育部大学计算机课程教学指导委员会委员

Research Center on **I**ntelligent  
**C**omputing for **E**nterprises & **S**ervices,  
**H**arbin **I**nstitute of **T**echnology



# 什么是关系？

## (3)关系的特性？

➤列是同质：即每一列中的分量来自同一域，是同一类型的数据

Student					
S#	Sname	Ssex	Sage	D#	Sclass
98030101	张三	男	20	03	980301
98030102	张四	女	20	计算机	980301
98030103	张五	男	19	03	980301
98040201	王三	01	20	04	980402
98040202	王四	男	21	控制	980402
98040203	王五	02	19	04	980402

出现列不同质的现象



# 什么是关系？

## (3)关系的特性？

➤不同的列可来自同一个域，称其中的每一列为一个属性，不同的属性要给予不同的属性名。

□ 关系模式  $R(A_1:D_1, A_2:D_2, \dots, A_n:D_n)$  中， $A_i (i = 1, \dots, n)$  必须是不同的，而  $D_i (i = 1, \dots, n)$  可以是相同的

➤例，我们定义一个域为 **Person** = 所有男人、女人和儿童的集合 = {李基，张鹏，王芳，刘玉，李健，张睿，张峰}，则下述“家庭”关系的三个列将来自同一个域 **Person**，因此需要不同的属性名“丈夫”“妻子”“子女”以示区分。

家庭		
丈夫	妻子	子女
李基	王方	李健
张鹏	刘玉	张睿
张鹏	刘玉	张峰



# 什么是关系?

## (3)关系的特性?

- 列位置互换性: 区分哪一列是靠列名
- 行位置互换性: 区分哪一行是靠某一或某几列的值(关键字/键字/码字)
- 关系是以内容(名字或值)来区分的, 而不是属性在关系的位置来区分
- 如下面两个关系是完全相同的关系

家庭

丈夫	妻子	子女
李基	王方	李键
张鹏	刘玉	张睿
张鹏	刘玉	张峰

家庭

丈夫	子女	妻子
李基	李键	王方
张鹏	张峰	刘玉
张鹏	张睿	刘玉



# 什么是关系？

## (3)关系的特性？

- 理论上，**关系**的任意两个元组不能完全相同。（集合的要求：集合内不能有相同的两个元素）；现实应用中，**表(Table)**可能并不完全遵守此特性。
- 元组相同是指两个元组的每个分量都相同。

Student					
S#	Sname	Ssex	Sage	D#	Sclass
98030101	张三	男	20	03	980301
98030102	张四	女	20	03	980301
98030103	张五	男	19	03	980301
98040201	王三	男	20	04	980402
98040202	王四	男	21	04	980402
98040203	王五	女	19	04	980402
98030103	张五	男	19	03	980301

在同一个关系中存在相同的元组，去掉其中的重复元组



# 什么是关系?

## (3)关系的特性?

➤属性不可再分特性: 又被称为关系第一范式

Students				
sid	lname	fname	class	telephone
1	Jones	Allan	2	555-1234
2	Smith	John	3	555-4321
3	Brown	Harry	2	555-1122
5	White	Edward	3	555-3344

符合第一范式(Table)

Students							Head: structured type
sid	name		class	telephone	enrollment		
	lname	fname			cno	major	
1	Jones	Allan	2	555-1234	101	No	
					108	Yes	
2	Smith	John	3	555-4321	105	No	
3	Borwn	Harry	2	555-1122	101	Yes	
					108	No	
4	White	Edward	3	555-3344	102	No	
					105	No	

|  | | | | | | | Value: structured value collection of values |

不符合第一范式  
(Not Table)



# 什么是关系?

## (3)关系的特性?

### ➤属性不可再分特性(续)

家庭		
丈夫	妻子	子女
李基	王方	李键
张鹏	刘玉	张睿
张鹏	刘玉	张峰

符合第一范式(Table)

不符合第一范式  
(Not Table)

家庭			
丈夫	妻子	子女	
		第一个	第二个
李基	王芳	李健	
张鹏	刘玉	张睿	张峰



### 候选码(Candidate Key)/候选键

□关系中的一个属性组，其值能唯一标识一个元组，若从该属性组中去掉任何一个属性，它就不具有这一性质了，这样的属性组称作候选码。

□例如：“学生(S#, Sname, Sage, Sclass)”，**S#**就是一个候选码，在此关系中，任何两个元组的**S#**是一定不同的，而这两个元组的**Sname**, **Sage**, **Sclass**都可能相同(同名、同龄、同班)，所以**S#**是候选码。

□再如：“选课(S#, C#, Sname, Cname, Grade)”，(**S#**,**C#**)联合起来是一个候选码



# 什么是关系？

## (4)关系上的一些重要概念—候选码/候选键



□ 有时，关系中有很多组候选码，例如：

**学生(S#, Sname, Sage, Sclass, Saddress)**

其中属性**S#**是候选码，属性组**(Sname, Saddress)**也是候选码(同名同地址的两个同学是不存在的)

□ 再如

**Employee(EmpID, EmpName, Mobile)**

每一雇员有唯一的**EmpID**，没有两个雇员有相同的手机号**Mobile**，则**EmpID**是候选码，**Mobile**也是候选码



什么是关系？

(5)关系上的一些重要概念—主码/主键



## 主码(Primary Key)/主键

- 当有多个候选码时，可以选定一个作为主码。
- DBMS以主码为主要线索管理关系中的各个元组。
- 例如可选定属性S#作为“学生”表的主码，也可以选定属性组(Sname, Saddress)作为“学生”表的主码。选定EmpID为Employee的主码。



### 主属性与非主属性

- 包含在任何一个候选码中的属性被称作主属性，而其他属性被称作非主属性
  - 如“选课”中的S#, C#为主属性，而Sname, Cname, Grade则为非主属性；
- 最简单的，候选码只包含一个属性
- 最极端的，所有属性构成这个关系的候选码，称为全码(All-Key)。
  - 比如：关系“教师授课”(T#,C#)中的候选码(T#,C#)就是全码。



### 外码(Foreign Key)/外键

□关系R中的一个属性组，它不是R的候选码，但它与另一个关系S的候选码相对应，则称这个属性组为R的**外码**或**外键**。

□例如“合同”关系中的客户号不是候选码，但却是外码。因它与“客户”关系中的候选码“客户号”相对应。

□两个关系通常是靠**外码**连接起来的。

合同			
主码	合同号	合同名称	客户号
	HT0001	购煤合同	CUST01
	HT0002	销售机床合同	CUST01
	HT0003	购钢材合同	CUST02

客户			
主码	客户号	客户名称	客户地址
	CUST01	依兰煤矿	哈尔滨市
	CUST02	长春电机厂	长春市
	CUST03	鞍钢集团	鞍山市



# 什么是关系？

## (7)关系上的一些重要概念—外码/外键

主码

Student					
S#	Sname	Ssex	Sage	D#	Sclass
98030101	张三	男	20	03	980301
98030102	张四	女	20	03	980301
98030103	张五	男	19	03	980301
98040201	王三	男	20	04	980402
98040202	王四	男	21	04	980402
98040203	王五	女	19	04	980402

外码

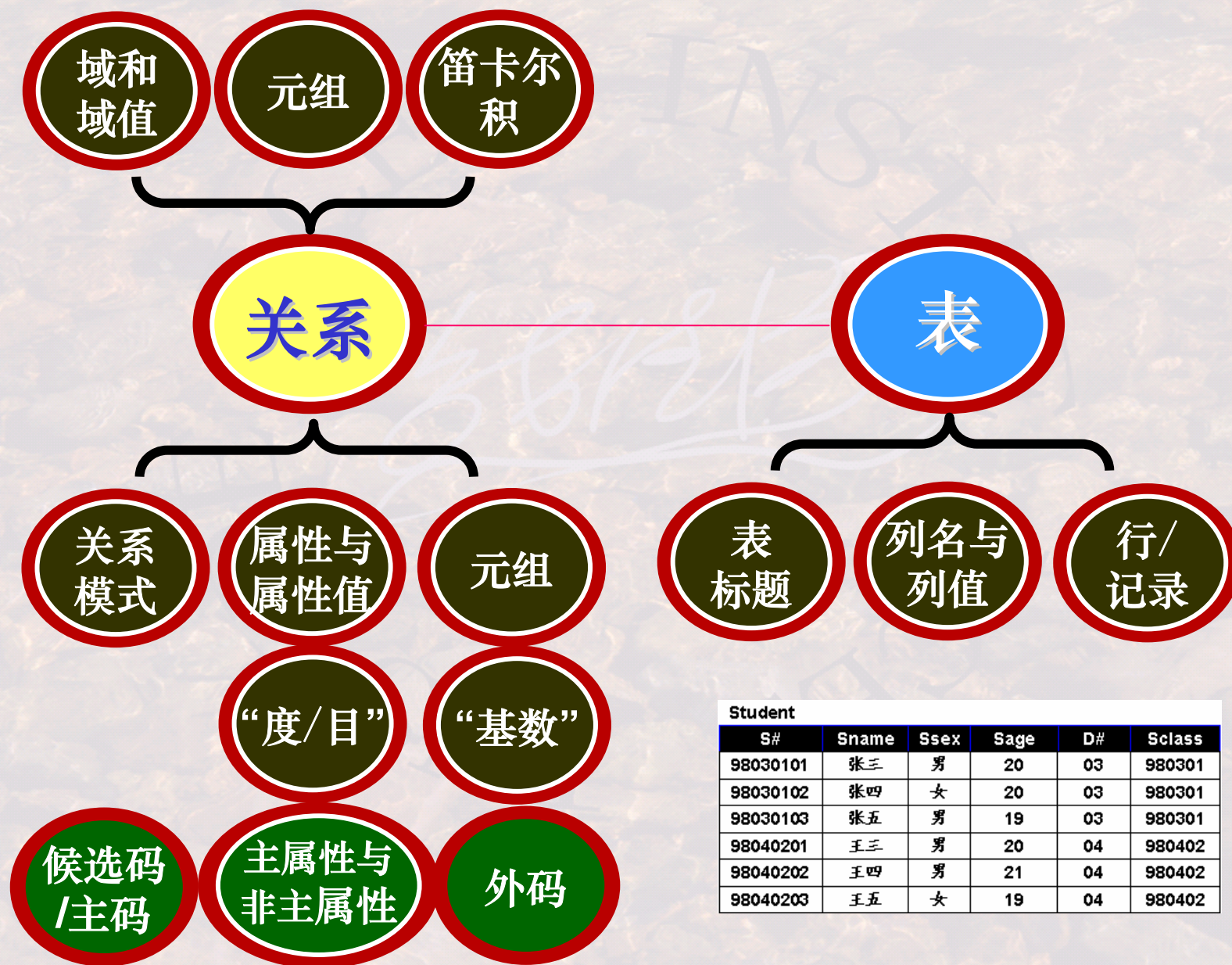
主码

Dept		
D#	Dname	Dean
01	机电	李三
02	能源	李四
03	计算机	李五
04	自动控制	李六



# 什么是关系？

## (8)小结





# 关系模型中的完整性？

战德臣

哈尔滨工业大学 教授.博士生导师  
黑龙江省教学名师  
教育部大学计算机课程教学指导委员会委员

Research Center on **I**ntelligent  
**C**omputing for **E**nterprises & **S**ervices,  
**H**arbin **I**nstitute of **T**echnology



### 实体完整性

- ❑ 关系的主码中的属性值不能为空值;
- ❑ 空值: 不知道或无意义的值;
- ❑ 意义: 关系中的元组对应到现实世界相互之间可区分的一个个体, 这些个体是通过主码来唯一标识的; 若主码为空, 则出现不可标识的个体, 这是不容许的。

主码

Student					
S#	Sname	Ssex	Sage	D#	Sclass
98030101	张三	男	20	03	980301
98030102	张四	女	20	03	980301
X	张五	男	19	03	980301
98040201	王三	男	20	04	980402
98040202	王四	男	21	04	980402
X	王五	女	19	04	980402



### ➤ 空值的含义

- 空值：不知道、不存在或无意义的值；
- 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的，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。关系模型中用‘?’表征
- 数据库中有了空值，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
- 例如：“ $3 + ?$ ”结果是多少呢？ “ $3 * ?$ ”结果是多少呢？ “ $? \text{ and } (A=A)$ ”结果又是多少呢？
- 再例如，一个班有30名同学，如所有同学都有成绩，则可求出平均成绩；如果有一个同学没有成绩，怎样参与平均成绩的计算呢，是当作0，还是当作100呢？还是不考虑他呢？
- 有空值的时候是需要特殊处理的，要特别注意。



### 参照完整性

□如果关系R1的外码Fk与关系R2的主码Pk相对应，则R1中的每一个元组的Fk值或者等于R2中某个元组的Pk值，或者为空值

□意义：如果关系R1的某个元组t1参照了关系R2的某个元组t2，则t2必须存在

□例如关系Student在D#上的取值有两种可能：

- ✓空值，表示该学生尚未分到任何系中
- ✓若非空值，则必须是Dept关系中某个元组的D#值，表示该学生不可能分到一个不存在的系中

R1----Student  
外码Fk----D#

Student					
S#	Sname	Ssex	Sage	D#	Sclass
98030101	张三	男	20	03	980301
98030102	张四	女	20	03	980301
98030103	张五	男	19	03	980301
98040201	王三	男	20	04	980402
98040202	王四	男	21	04	980402
98040203	王五	女	19	04	980402
98050206	孙六	男	21	05	980502
98060207	吴七	女	20		980602

R2----Dept  
主码Pk----D#

Dept		
D#	Dname	Dean
01	机电	李三
02	能源	李四
03	计算机	李五
04	自动控制	李六



## 用户自定义完整性

- ❑ 用户针对具体的应用环境定义的完整性约束条件
- ❑ 如**S#**要求是**10**位整数，其中前四位为年度，当前年度与他们的差必须在**4**以内
- ❑ 再如：

要求名字在5个  
汉字字符之内

Student					
S#	Sname	Ssex	Sage	D#	Sclass
98030101	张三	男	20	03	980301
98030102	张四	女	20	03	980301
98030103	张五	男	19	03	980301
98040201	王三	男	20	04	980402
98040202	王四	男	21	04	980402
98040203	王五	女	19	04	980402
98040204	詹姆斯. 卡梅隆	男	20	04	980402
98040206	王六	01	21	04	980402
98040207	王七	女	200	04	980402

年龄在[12, 35]  
之间

性别只能是  
“男”或“女”

颠覆完整性  
的示例



## DBMS对关系完整性的支持

- 实体完整性和参照完整性由**DBMS**系统自动支持
- **DBMS**系统通常提供了如下机制：
  - (1)它使用户可以自行定义有关的完整性约束条件
  - (2)当有更新操作发生时，**DBMS**将自动按照完整性约束条件检验更新操作的正确性，即是否符合用户自定义的完整性



# 回顾本讲学了什么？

战德臣

哈尔滨工业大学 教授.博士生导师

黑龙江省教学名师

教育部大学计算机课程教学指导委员会委员

Research Center on **I**ntelligent  
**C**omputing for **E**nterprises & **S**ervices,  
**H**arbin **I**nstitute of **T**echnology



# 回顾本讲学习了什么？

