



数据库原理

Theory of Database

李静

信息科学与技术学院

复 习

❖ 两大数据模型

概念层数据模型：E-R图。

组织层数据模型：关系模型。

关系数据库

❖ 关系数据库系统是**支持关系模型**的数据库系统

❖ 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**BCNF**范式

关系数据库

- 1 关系模型概述
- 2 关系数据模型的基本术语
- 3 关系模型的完整性约束
- 4 关系代数

关系数据模型概述

- 关系数据结构
- 关系操作
- 数据完整性约束

关系数据结构

- ❖ 关系数据模型源于**数学**。
- ❖ 用**二维表**来组织数据，这个二维表在关系数据库中就称为关系。
- ❖ 关系数据库就是**表或关系的集合**。
- ❖ 关系系统要求让用户所感觉的数据库就是一张张表。
- ❖ 在关系系统中，表是**逻辑结构**而不是物理结构。

关系操作

❖ 传统的关系运算：

- ✓ 并 (**Union**)
- ✓ 交 (**Intersection**)
- ✓ 差 (**Difference**)
- ✓ 笛卡尔乘积 (**Cartesian Product**)

❖ 专门的关系运算：

- 选择 (**Select**)
- 投影 (**Project**)
- 连接 (**Join**)
- 除 (**Divide**)

关系模型的数据操作

- ❖ 主要包括：查询、插入、删除和修改数据。
- ❖ 是基于集合的操作，操作对象和操作结果都是集合（或关系）。
- ❖ 是非过程化的。

关系数据库

- 1 关系模型概述
- 2 关系数据模型的基本术语
- 3 关系模型的完整性约束
- 4 关系代数

关系数据模型的基本术语

- ❖ **关系**：关系就是**二维表**。并满足如下性质：
 - 关系表中的每一列都是**不可再分**的基本属性；
 - 表中的行、列**次序并不重要**。
- ❖ **属性**：表中的**每一列**是一个属性值集，列可以命名，称为属性名。
- ❖ **值域**：属性的取值范围。如，性别只能是‘男’和‘女’两个值。

关系数据模型的基本术语

- ❖ **元组**：表中的**每一行**称作是一个元组，它相当于一个记录值。
- ❖ **分量**：元组中的**每一个属性值**称为元组的一个分量， n 元关系的每个元组有 n 个分量。
- ❖ **关系模式**：关系模式是关系的“型”或元组的结构共性的描述。关系模式实际上对应关系表的表头。设关系名为 R ，属性分别为 A_1, A_2, \dots, A_n ，则关系模式可以表示为：

$$R(A_1, A_2, \dots, A_n)$$

关系数据模型的基本术语

- ❖ **关系数据库**：对应于一个关系模型的所有关系的集合称为关系数据库。
- ❖ **候选码**：能够唯一标识关系中的一个元组的一个属性或最小属性组。
- ❖ **主码**：指定候选码中的一个作为主码。（编程时用）

关系数据模型的基本术语

- ❖ **主属性**：包含在任一候选码中的属性称为主属性。
- ❖ **非主属性**：不包含在任一候选码中的属性称为非主属性。
- ❖ **外码**

对关系的限定--关系的性质

- ❖ 关系中的每个分量都必须是不可再分的最小数据项。
- ❖ 表中列的数据类型是固定的，即每个列是同类的数据，来自相同的值域。
- ❖ 每个列称为一个属性，每个属性有不同的属性名。
- ❖ 关系表中行、列的顺序不重要。
- ❖ 同一个关系中元组不能重复。

关系模式和关系

- ❖ 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
 - 元组语义以及完整性约束条件
 - 属性间的数据依赖关系集合

关系数据库

- 1 关系模型概述
- 2 关系数据模型的基本术语
- 3 关系模型的完整性约束
- 4 关系代数

数据完整性约束

- ❖ **数据完整性**是指数据库中存储的数据是**有意义**的或**正确**的。
- ❖ 主要包括三大类：
 - 实体完整性
 - 参照完整性
 - 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**，应该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。

实体完整性

❖ 规则3.1 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

例：student(学号，姓名，性别，专业)

主码“学号”不能取空值

例：学生选课（学号，课程号，成绩）

属性组（学号，课程号）为主码，则这两个属性都不能取空值。

实体完整性

❖ 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。
- (2) 现实世界的实体是可区分的，即它们具有某种唯一性标识。
- (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。
 - ◆ 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。

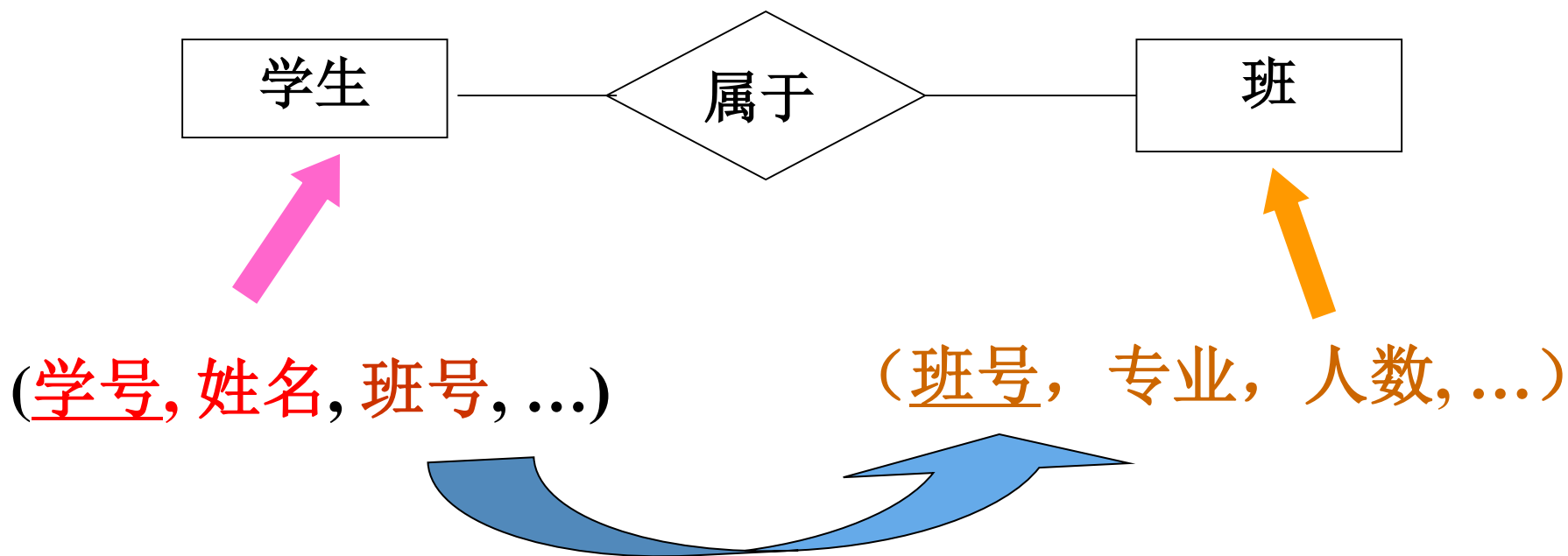
参照完整性

- ❖ 现实世界中的实体之间往往存在某种联系，在关系模型中，实体及实体间的联系都是用关系来描述的。
- ❖ 参照完整性用于描述实体之间的联系。
- ❖ 一般用外码实现。

外码：取作本表（子表）属性之一的外表（父表，主表）主码。
- ❖ 主码值先在主表中生成，后在子表中引用

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。



1. 关系间的引用

- 例如： 学生实体、专业实体
 - ◆ 学生（学号，姓名，性别，**专业号**，年龄）
 - ◆ 专业（**专业号**，专业名）
- 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。

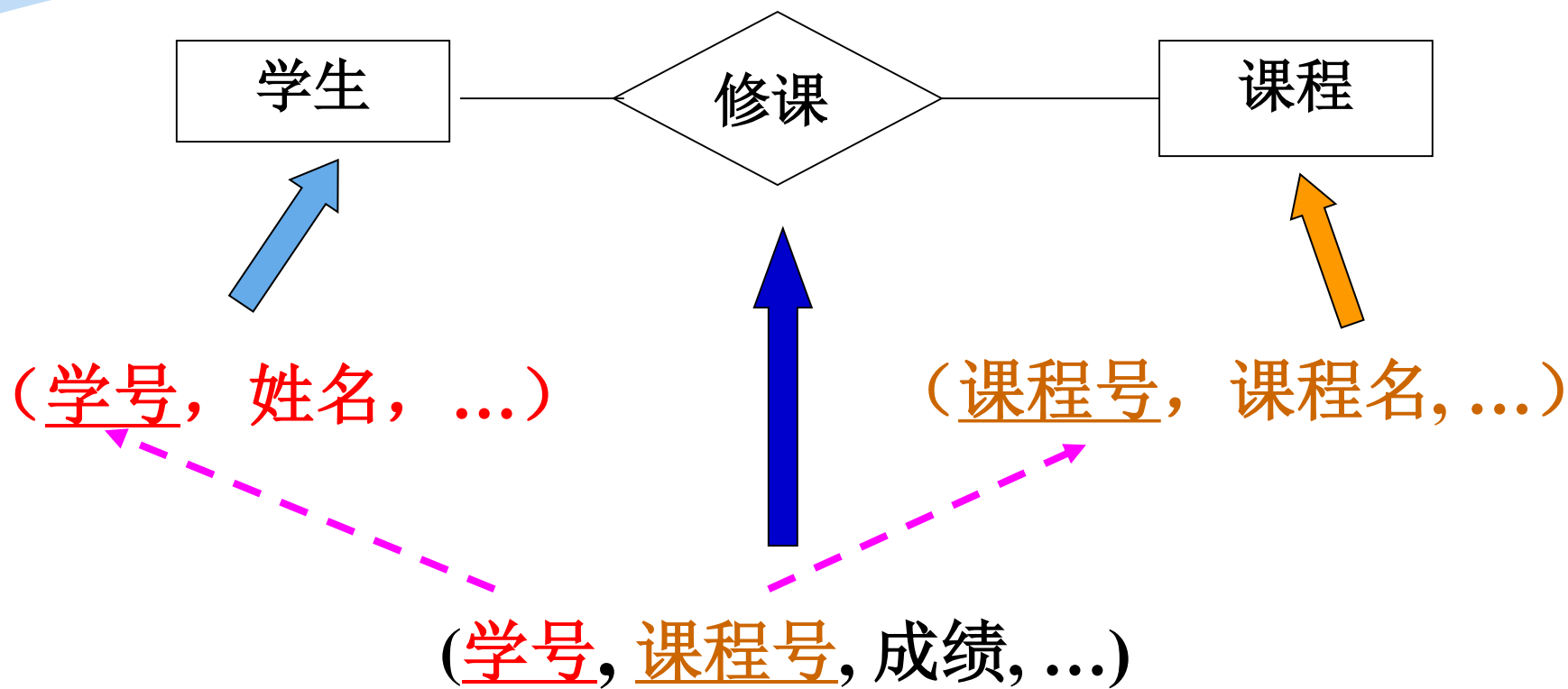
1. 关系间的引用

例：学生、课程、选修之间的多对多联系

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，分数）



2. 关系内的引用

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

■“学号”是主码，“班长”是作为班长的学生的学号，即“班长”引用了本关系的“学号”，即“班长”必须是确实存在的学生的学号。

3. 外码 (Foreign Key)

- ❖ 设 F 是基本关系 R 的一个或一组属性，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。
- ❖ 关系 R 称为参照关系 (Referencing Relation)。
- ❖ 关系 S 称为被参照关系 (Referenced Relation)，或目标关系 (Target Relation) (被引用表)。

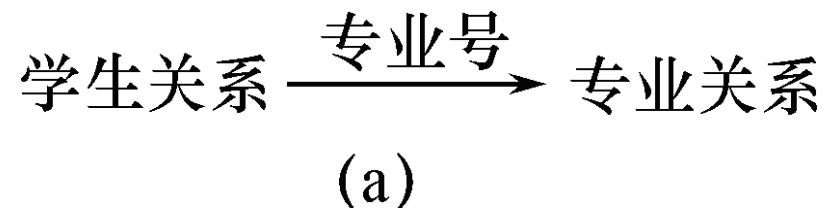
3. 外 码

❖ 例：如下关系模式

学生（学号， ..., 专业号）

专业（专业号， ...）

- “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



3. 外 码

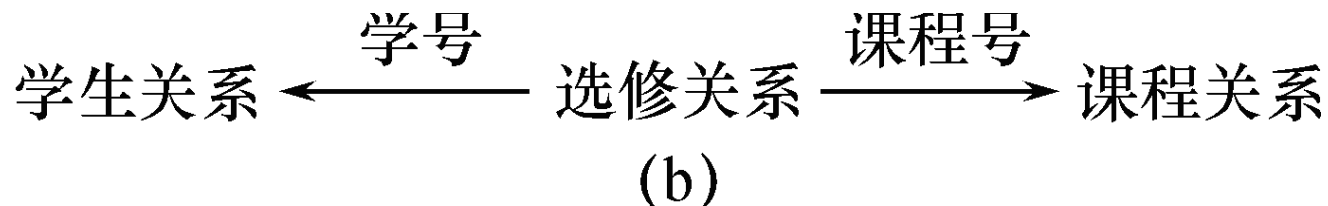
❖ 例：如下三个关系模式

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，分数）

- “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

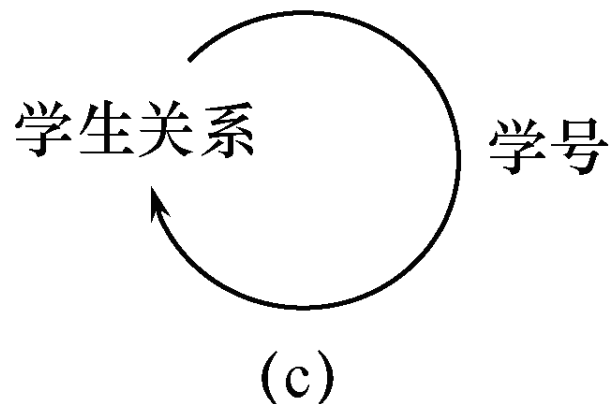


3. 外 码

❖ 例：“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

学生（学号，姓名，性别，专业号，年龄，班长）



参照完整性规则

- ❖ 规则3.2 参照完整性规则：若属性组 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(被引用表)某个元组的主码值

参照完整性规则

❖ 例：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

学生（学号，...，专业号）

专业（专业号，...）

参照完整性规则

❖ 例：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

（2）非空值，该值必须是本关系中某个元组的学号值

❖ 例：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

用户定义的完整性

- ❖ 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。

用户定义的完整性

❖ 例：课程(课程号，课程名，学分)

- “课程号” 属性非空，且必须取唯一值；
- 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}

实体完整性

用户定义完整性

课堂练习

供应商关系S（主码是“供应商号”）

供应商号	供应商名	所在城市
B01	红星	北京
S10	宇宙	上海
T20	黎明	天津
Z01	立新	重庆

今要向关系P中插入新行，新行的值分别列出如下。哪些行能够插入？

- A. ('037', '绿', null)
- B. (null, '黄', 'T20')
- C. ('201', '红', 'T20')
- D. ('105', '蓝', 'B01')
- E. ('101', '黄', 'T11')

零件关系P（主码是“零件号”，外码是“供应商号”）

零件号	颜色	供应商号
010	红	B01
312	白	S10
201	蓝	T20

复 习：关系数据库

■ 关系数据模型的基本术语

关系、属性、值域、元组、分量、关系模式、关系数据库、候选码、主码、外码、主属性、非主属性

复 习：关系数据库

■关系的性质

- 关系中的每个分量都必须是不可再分的**最小数据项**。
- 表中列的数据类型是固定的，即每个列是**同类的**数据，来自**相同的值域**。
- 每个列称为一个属性，每个属性有不同的**属性名**。
- 关系表中**行、列的顺序不重要**。
- 同一个关系中**元组不能重复**。

复 习：关系数据库

■ 关系模型的完整性约束

三类完整性约束：

- ✓ 实体完整性
 - ✓ 参照完整性
 - ✓ 用户自定义完整性
- } 每个关系都要满足！
- } 每个关系自由定义。

主码不能取空值！

或者取空值或者等于被参照表的某个元组的码值。