

关系数据库

关系模式

- 数据模型是对现实世界数据特征的抽象，是数据库系统的核心和基础。
- 基本概念

□ 信息世界中的基本概念

(1) 实体 (Entity)

可以是具体的人、事、物或抽象的概念。

(2) 属性 (Attribute)

实体所具有的某一特性称为属性。

(3) 码 (Key)

唯一标识实体的属性集称为码。

(4) 实体型 (Entity Type)

用实体名及其属性名集合来抽象和刻画同类实体称为实体型

(5) 实体集 (Entity Set)

同一类型实体的集合称为实体集

(6) 联系 (Relationship)

■ **实体内部的联系**通常是指组成实体的各属性之间的联系

■ **实体之间的联系**通常是指不同实体集之间的联系

■ 实体之间的联系有一对一、一对多和多对多等多种类型

- 关系模型的数据结构属于对比

关系模型的数据结构术语对比

关系术语	一般表格的术语
关系名	表名
关系模式	表头（表格的描述）
关系（Relation）	（一张）二维表
元组（Tuple）	记录或行
属性（Attribute）	列
属性名	列名
属性值	列值
分量	一条记录中的一个列值
非规范关系	表中有表（大表中嵌有小表）

关系模型的优缺点

优点

- 建立在严格的数学概念的基础上
- 概念单一
 - 实体和各类联系都用关系来表示
 - 对数据的检索结果也是关系
- 关系模型的存取路径对用户透明
 - 具有更高的数据独立性，更好的安全保密性
 - 简化了程序员的工作和数据库开发建立的工作

缺点

- 存取路径对用户透明，查询效率往往不如格式化数据模型
- 为提高性能，必须对用户的查询请求进行优化，增加了开发数据库管理系统的难度

关系数据库

关系模式可以表示为： $R(U, D, DOM, F)$ 。

其中R为关系名，U为组成该关系的属性名的集合，D为U中属性来自的域，DOM为属性向域的映像集合，F为属性间数据的依赖关系的集合。

关系模式可以简记为 $R(U)$ 或 $R(A_1, A_2, \dots, A_n)$

重要概念

1. 域：一组具有相同数据类型的值的集合
2. 候选码：若关系中的某一属性组能唯一地标识一个元组，则该属性组为候选码
3. 主码：一个关系中有多个候选码，选择其中一个作为主码。主码在定义关系时被选择和声明，一旦选中，无法改变，一般建议选择属性数最少的候选码作为主码。
4. 主属性：候选码的各个属性被称为主属性，不包含在任何候选码中的属性被称为非主属性（非码属性）

基本关系的性质

- 列是同质的
- 不同的列可出自同一个域（不同的属性需要不同的属性名）
- 任意两个元组的候选码不能相同
- 行列顺序无所谓，可以任意交换
- 插入数据时，有几种可能的情况：1) 未知；2) 未分配；3) 不适用。如果出现，则设置为空值，即：null。
PS：空值与0或空格不同。在SQL中，任何涉及null的算数表达式都被计算为null

关系完整性

三类关系关系完整性约束

1. 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。
2. 参照完整性：1) 关系间的引用：关系R引用的属性的值必须在关系S中存在；2) 外码：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Ks相对应，则称F是R的外码。关系R被称为参照关系，关系S被称为被参照关系（目标关系）。3) 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Ks相对应，则对于R中每个元组在F上的值必须为：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。
3. 用户定义的完整性：针对某一具体关系数据库的约束条件，关系模型应提供定义和检验这类完整性的机制。例如，关系：课程（课程号，课程名，学分）中，“课程号”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}。

对于1和2，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。对于3，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。