



1.2 数据模型(关系模型)

关系模型

* 数据结构

属性(Attribute) 分量

学号	姓名	性别	专业号	年龄
801	张三	女	01	19
802	李四	男	01	20
803	王五	男	01	20
804	赵六	女	02	20
805	钱七	男	02	19

码(Key) 元组(Tuple)

关系(Relation)

域：属性的取值范围。

关系模式：对关系结构的描述，定义如下：

关系名（属性1，属性2，...，属性n）

例：学生（学号，姓名，年龄，性别，系，年级）



1.2 数据模型(关系模型)

👉 数据操纵与完整性约束

- * 查询、插入、删除、修改

数据操作是集合操作，**操作对象和操作结果都是关系**，即若干元组的集合。

- * 实体完整性、参照完整性、用户定义完整性

👉 存储结构

实体和联系都作为数据文件来存储。

- * 实体型：直接用关系（表）表示。
- * 属性：用属性名表示。
- * 一对一联系：隐含在实体对应的关系中。
- * 一对多联系：隐含在实体对应的关系中。
- * 多对多联系：直接用关系表示。



1.2 数据模型(关系模型)

👉 优点

- * 建立在严格的数学概念的基础上
- * 概念单一。数据结构简单、清晰，用户易懂易用
 - 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导致查询效率往往不如非关系数据模型。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。



1.2 数据模型(关系模型)

- * **ORACLE**
- * **SYBASE**
- * **INFORMIX**
- * **DB/2**
- * **Ingres**
- * **OpenBase**
- * **SQL Server**
- * **PostgreSQL**
- * **MySQL**