

# 数据库系统原理

# 第2章 关系数据库1



分级通关平台,交流



#### □ 基础篇

第1章 绪论

第2章 关系数据库\*3

第3章 标准语言SQL\*3

第4章 数据库安全性

第5章 数据库完整性

实验1

#### □ 设计与应用开发篇

第6章 关系数据理论\*2

第7章 数据库设计

实验2

第8章 数据库编程

#### Ⅲ 系统篇

第9章 \*关系查询处理和优化 实验3

第10章 数据库恢复技术 第11章 并发控制 实验4







# 内容提要

- 1. 关系数据库的概况
- ✓ 什么是关系?
- ✓ 关系数据库发展
- ✓ 关系的数据结构
- ✓ 关系的操作
- 2. 关系的完整性
- ✓ 关系的完整性
- ✓ 传统的集合运算
- 3. 关系代数
- ✓ 专门的关系运算







## 0. 关系数据库发展

- 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<mark>范式</mark>
- 1974年提出了关系的BC范式





## 1. 要素——关系的数据结构

- ■关系
- ■关系模式
- ■关系数据库
- ■关系模型的存储结构





## 关系

## 逻辑结构----二维表

从用户角度,关系模型中数据的逻辑结构是一张二 维表

## 关系----单一的数据结构

现实世界的实体以及实体间的各种联系均用关系来表示

关系的操作建立在集合代数的基础上







# 关系例子

SUPERVISOR	SPECIALITY	POSTGRADUATE
张清玫	计算机专业	李勇
张清玫	计算机专业	刘晨
刘逸	信息专业	王敏





## 关系

#### (1) 关系

 $D_1 \times D_2 \times ... \times D_n$ 的<u>子集</u>叫作在域 $D_1$ , $D_2$ ,..., $D_n$ 上的 关系,表示为

$$R (D_1, D_2, ..., D_n)$$

■*R*: 关系名

■n: 关系的目或度(Degree)





#### 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
- ■指定长度的字符串集合
- { `男', '女' }
- **.....**





## 关系例子

SUPERVISOR	SPECIALITY	POSTGRADUATE
张清玫	计算机专业	李勇
张清玫	计算机专业	刘晨
刘逸	信息专业	王敏

#### 3个域:

- **❖ D1**=导师集合SUPERVISOR={张清玫,刘逸}
- **❖ D2=专业集合SPECIALITY=** {计算机专业,信息专业}
- ❖ D3=研究生集合POSTGRADUATE={李勇,刘晨,王敏}





## 关系

#### (2) 元组

关系中的每个元素是关系中的元组,通常用**t**表示。

▶ 单元关系与二元关系

当*n*=1时,称该关系为**单元**关系(Unary relation)

或一元关系

当*n*=2时,称该关系为二元关系(Binary relation)





## 关系

#### (3) 关系的表示

关系是一个二维表,表的每行对应一个<mark>元组</mark>,表的每 列对应一个<mark>域</mark>

#### (4) 属性

关系中不同列可以对应相同的域,必须对每列起一个名字, 称为属性(Attribute)

n目关系必有n个属性





#### (5) 码

候选码(Candidate key)

若关系中的某一属性组的值能<mark>唯一地标识</mark>一个元组,则称该属性组为候选码(特例:候选码只包含一个属性) 候选码的诸属性称为主属性(Prime attribute),其他 为非主属性(Non-Prime attribute)

- ▶ **主码(Primary key**):若一个关系有多个候选码,则选定其中一个为主码;
- > 全码 (All-key)

特例情况:关系模式的所有属性组是这个关系模式的 候选码,称为全码





## 关系例子

SUPERVISOR	SPECIALITY	POSTGRADUATE
张清玫	计算机专业	李勇
张清玫	计算机专业	刘晨
刘逸	信息专业	王敏

关系: SAP(SUPERVISOR, SPECIALITY, POSTGRADUATE)

假设:导师与专业: n:1, 导师与研究生: 1:n

主码: POSTGRADUATE (假设研究生不会重名)





## (6) 关系的性质

- ① 列是同质的。每一列中的分量必须来自同一个域,必须是同一类型数据。
- ② 不同的列可出自同一个域。不同的属性要给予不同的属性名
- ③ 列的顺序无所谓,列的次序可以任意交换
- ④ 任意两个元组的候选码不能相同
- ⑤ 行的顺序无所谓, 行的次序可以任意交换



SUPERVISO	SPECIALITY	POSTGRADUATE	
R		PG1	PG2
张清玫	计算机专业	李勇	刘晨
刘逸	信息专业	王敏	

包含多个



- ➤ 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
  - ●哪些属性构成
  - ●属性来自的域
  - ●属性与域之间的映象关系

完整性约束条件







> 关系模式可以形式化地表示为:

R (U, D, DOM, F)

R 关系名

U 组成该关系的属性名集合

D U中属性所来自的域

DOM 属性向域的映象集合

F 属性间数据的依赖关系的集合







关系模式通常可以简记为

R(U)  $\vec{x}$   $R(A_1, A_2, ..., A_n)$ 

R: 关系名

 $A_1, A_2, ..., A_n$ :属性名

注: 域名及属性向域的映象常常直接说明为属性的类型、长度

关系: SAP(SUPERVISOR, SPECIALITY, POSTGRADUATE)





- ◆ 关系模式 对关系的描述 静态的、稳定的
- ◆ 关系 关系模式在某一时刻的状态或内容 动态的、随时间不断变化的
- ▶ 关系模式和关系往往<mark>笼统称为关系</mark> 通过上下文加以区别





## 什么是关系数据库?

## > 关系数据库

在一个给定的应用领域中,所有关系的集合构成一 个关系数据库

✔ 关系数据库的型与值

**关系数据库的型:** 关系数据库模式,是对关系数据库的描述

**关系数据库的值:** 关系模式在某一时刻对应的关系的集合,通常称为关系数据库





## 数据库在物理层面如何实现?

- > 数据放在内存
  - ✓ 动态申请空间,使用
    结构数组形式
  - ✓ 链表,分块的链表,.....
- > 数据放在外存
  - ✔ 一个数据文件,需要文件头吗?
  - ✓ 多个数据文件, .....
  - 问题:如何支持 "查,增删改"?







## 数据库在物理层面如何实现?

- > 数据持久化,部分数据读入内存处理
  - ✔ 数据都放在外存
  - ✓ 采用B+树来实现高效的查找和其他处理
  - **√** .....
  - 问题:理解 "查,增删改"的效率 数据库也是一种操作系统







## 2. 要素——关系操作

- 常用的关系操作
  - ✓ <mark>查询</mark>操作:选择、投影、连接、除、并、差、交、 笛卡尔积
  - ▶ 选择、投影、并、差、笛卡尔积是5种基本操作
  - ✓ 数据更新:插入(增)、删除、修改







## 关系操作

■ 关系操作的特点 集合操作方式。操作的**对象和结果都是集合**,一次一 集合的方式

■ 关系代数语言 用对关系的运算来表达**查询要求** 代表: ISBL





## 关系操作

■ 关系操作的特点

集合操作方式。操作的**对象和结果都是集合**,

一次一集合的方式

- ✓ 关系<mark>查询</mark>的结果,也是关系
- ✓ 关系<mark>增删改</mark>的结果,还是关系
- ✓ 多个关系的组合,视图,也是关系





## 关系操作语言

- 关系代数语言 用对关系的运算来表达**查询要求** 代表: ISBL
- 关系演算语言:用谓词来表达**查询要求** 元组关系演算语言
  - ●谓词变元的基本对象是元组变量
  - ●代表: APLHA, QUEL

域关系演算语言

- ●谓词变元的基本对象是域变量
- ●代表: QBE





## 关系操作语言SQL

■ 关系代数语言: 用对关系的运算来表达查询要求

■ 关系演算语言:用谓词来表达查询要求

■ 具有关系代数和关系演算<mark>双重</mark>特点 的语言代表:

**SQL** (Structured Query Language)







## **SQL** (Structured Query Language)

- ▶ 丰富的查询功能
- ▶ 数据定义功能
  - ✓ 数据库对象的定义与修改
- ▶ 数据操纵功能
  - ✔ 操纵功能包括增加、修改、删除等操作
- ▶ 数据控制功能
  - ✓ 数据的安全性控制
  - ✓ 数据的完备性控制





## 小结

- ▶ 什么是关系
  逻辑结构是二维表
- > 关系与关系模式

域,元组,属性,码

 $R(A_1, A_2, ..., A_n)$ 

- ▶ 常用的关系操作
  查询,增删改
- ➤ SQL (Structured Query Language) 简介

数据库的定义、查询、操纵、控制语言





## MySQL环境

- ■下载
- 安装
- 使用
  - ▶ 超级用户,admin,口令
  - > 数据库对象
  - ▶ SQL语句窗口
  - > 运行结果窗口
  - ➤ SQL语句的保存

