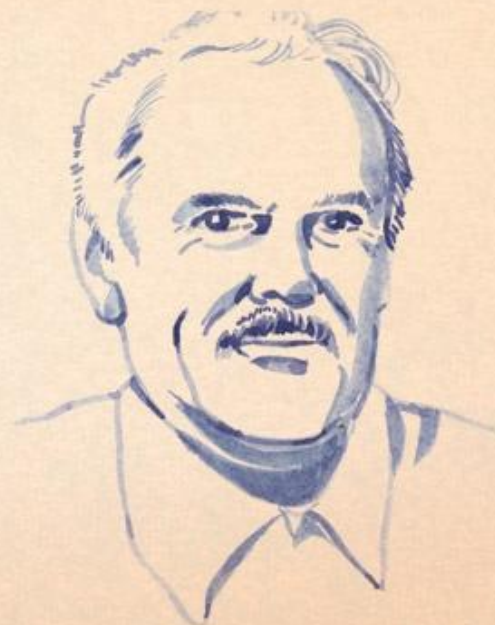


**Father of the  
Relational Database:  
Edgar F. Codd**

A British computer scientist, Codd made important contributions to the theory of relational databases. While working for IBM, he created the relational model for database management.



# 数据库设计 第1周

**【声明】** 本视频和幻灯片为炼数成金网络课程的教学资料，所有资料只能在课程内使用，不得在课程以外范围散播，违者将可能被追究法律和经济责任。

课程详情访问炼数成金培训网站

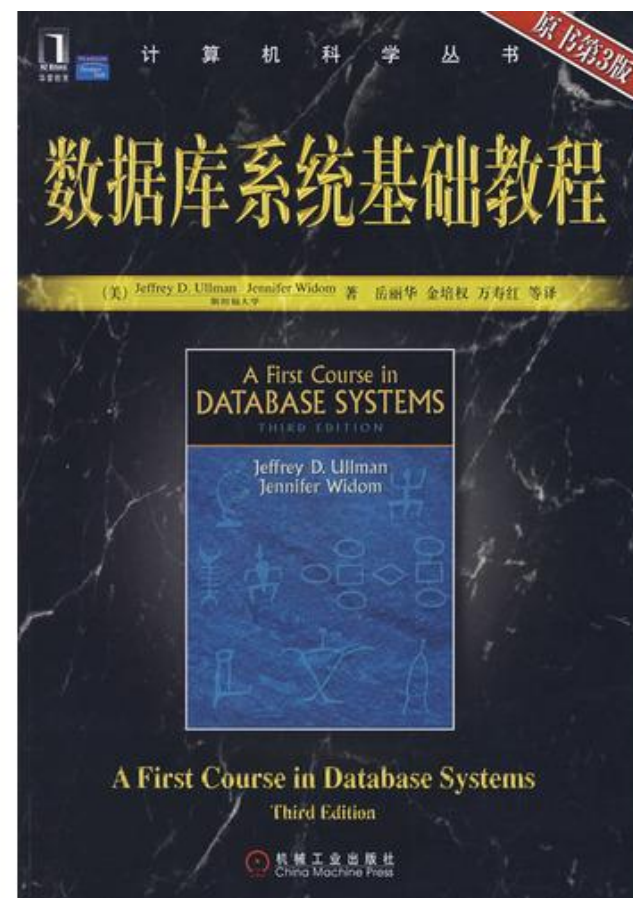
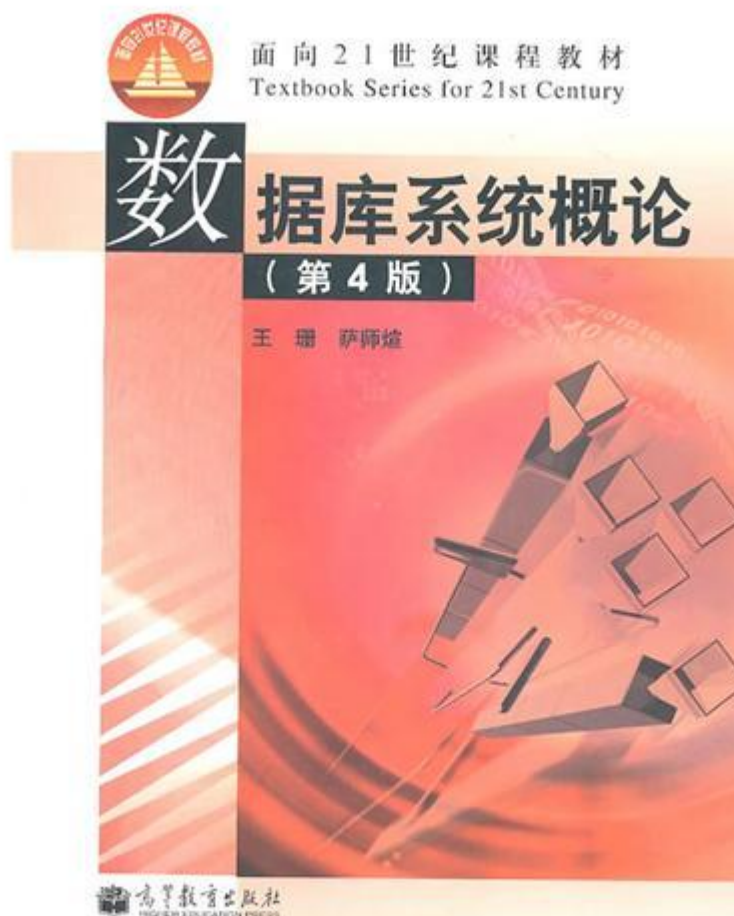
<http://edu.dataguru.cn>

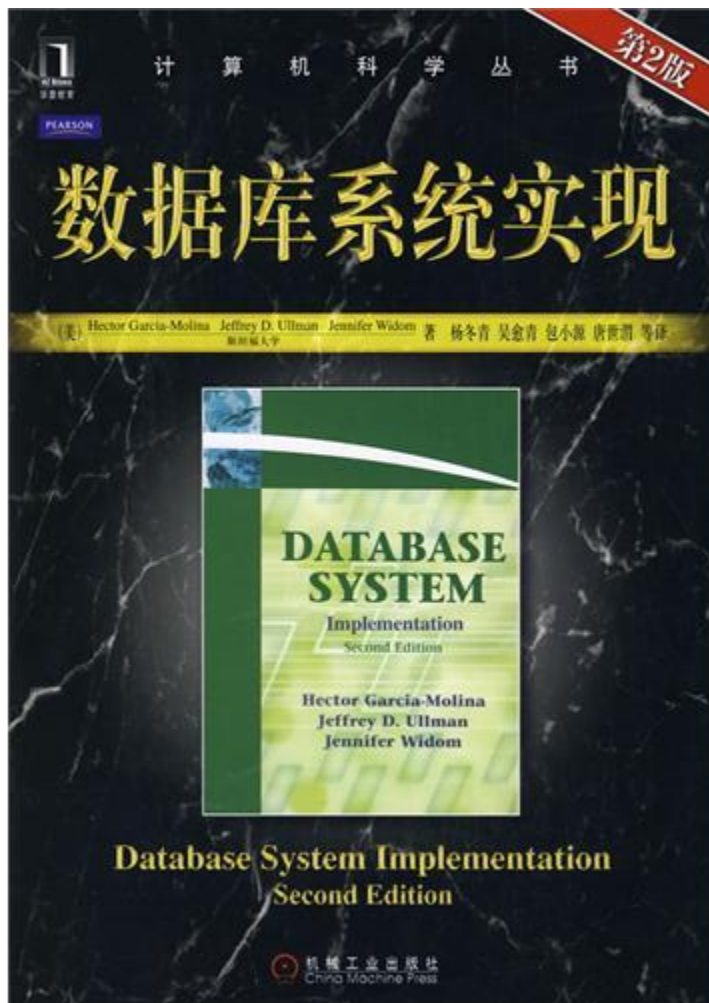
## 关注炼数成金企业微信

- 提供全面的数据价值资讯，涵盖商业智能与数据分析、大数据、企业信息化、数字化技术等，各种高性价比课程信息，赶紧掏出您的手机关注吧！

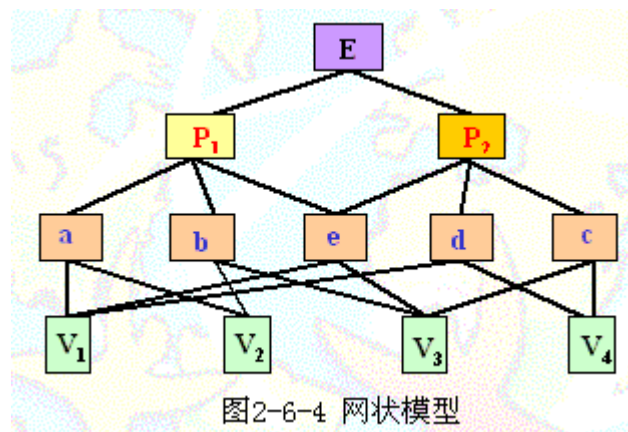
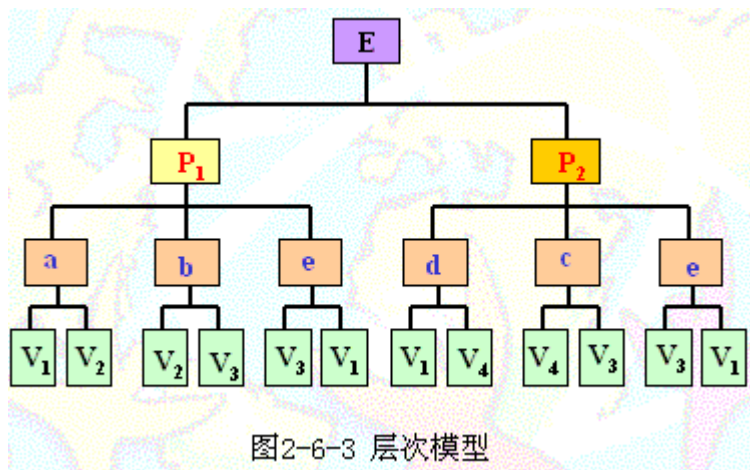


# 关于关系代数的参考书





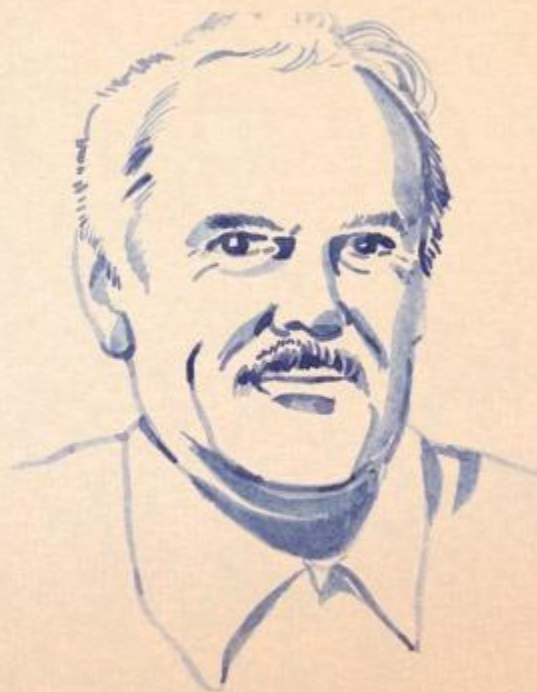
- 无库时代
- 层次状数据库
- 网状数据库
- 关系型数据库





## Father of the Relational Database: Edgar F. Codd

A British computer scientist, Codd made important contributions to the theory of relational databases. While working for IBM, he created the relational model for database management.



- 1970年，Codd划时代的论文《用于大型共享数据库的关系数据模型》
- 关系与关系代数

|   |    |    |    |
|---|----|----|----|
| R | A  | B  | C  |
|   | a1 | b1 | c1 |
|   | a1 | b2 | c2 |
|   | a2 | b2 | c1 |

|   |    |    |    |
|---|----|----|----|
| S | A  | B  | C  |
|   | a1 | b2 | c2 |
|   | a1 | b3 | c2 |
|   | a2 | b2 | c1 |

|     |    |    |
|-----|----|----|
| R-S |    |    |
| A   | B  | C  |
| a1  | b1 | c1 |

An Introduction to Database System

表2.4 关系代数运算符

| 运算符   | 含义                                  | 运算符   | 含义  |
|-------|-------------------------------------|-------|---|
| 集合运算符 | $\cup$<br>$-$<br>$\cap$<br>$\times$ | 比较运算符 | $>$<br>$\geq$<br>$<$<br>$\leq$<br>$=$<br>$\neq$ |
|       | 并<br>差<br>交<br>广义笛卡尔积               |       | 大于<br>大于等于<br>小于<br>小于等于<br>等于<br>不等于           |

| 运算符      | 含义                                       | 运算符   | 含义                           |
|----------|--|-------|------------------------------|
| 专门的关系运算符 | $\sigma$<br>$\pi$<br>$\bowtie$<br>$\div$ | 逻辑运算符 | $\neg$<br>$\wedge$<br>$\vee$ |
|          | 选择<br>投影<br>连接<br>除                      |       | 非<br>与<br>或                  |



## 准则0

一个关系形的关系数据库系统必须能完全通过它的关系能力来管理数据库。

## 准则1 信息准则

关系数据库系统的所有信息都应该在逻辑一级上用表中的值这一种方法显式的表示。

## 准则2 保证访问准则

依靠表名、主码和列名的组合，保证能以逻辑方式访问关系数据库中的每个数据项。

## 准则3 空值的系统化处理

全关系的关系数据库系统支持空值的概念，并用系统化的方法处理空值。

## 准则4 基于关系模型的动态的联机数据字典

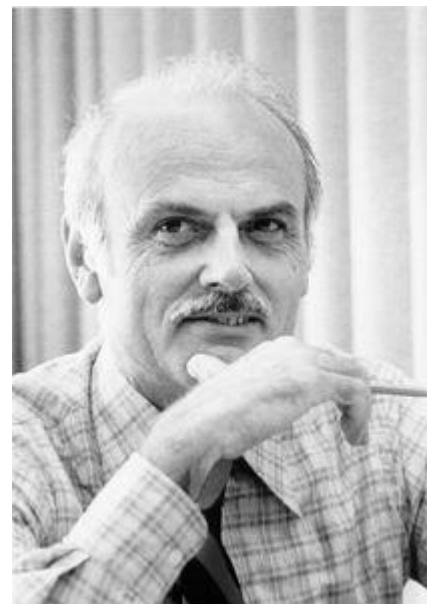
数据库的描述在逻辑级上和普通数据采用同样的表述方式。

## 准则5 统一的数据子语言

一个关系数据库系统可以具有几种语言和多种终端访问方式，但必须有一种语言，它的语句可以表示为严格语法规定的字符串，并能全面的支持各种规则。

## 准则6 视图更新准则

所有理论上可更新的视图也应该允许由系统更新。



准则7 高级的插入、修改和删除操作

系统应该对各种操作进行查询优化。

准则8 数据的物理独立性

无论数据库的数据在存储表示或存取方法上作任何变化，应用程序和终端活动都保持逻辑上的不变性。

准则9 数据逻辑独立性

当对基本关系进行理论上信息不受损害的任何改变时，应用程序和终端活动都保持逻辑上的不变性。

准则10 数据完整的独立性

关系数据库的完整性约束条件必须是用数据库语言定义并存储在数据字典中的。

准则11 分布独立性

关系数据库系统在引入分布数据或数据重新分布时保持逻辑不变。

准则12 无破坏准则

如果一个关系数据库系统具有一个低级语言，那么这个低级语言不能违背或绕过完整性准则。



- 唐·钱伯林 ( Don Chamberlin ) , 是 IBM Fellow , ACM及IEEE 特别会员。他是SQL关系数据库语言的发明人之一 , 也是XQuery语言的设计基础Quilt语言的发明人之一。Don拥有加利福尼亚大学博士学位。他目前在IBM Almaden 研究中心工作 , 在过去几年中 , 他代表IBM参与W3C XML Query 工作组的工作。

# Scott和他的猫的故事



■ 左起 Ed Oates、Bruce Scott、Bob Miner、Larry Ellison

# Oracle与Larry Ellison









- Michael Stonebraker , SQL Server/Sysbase奠基人。
- 著名的数据库科学家，他在1992 年提出对象关系数据库模型在加州伯克利分校计算机教授达25年。在此期间他创作了Ingres, Illustra, Cohera, StreamBase Systems和Vertica等系统。Stonebraker教授也曾担任过Informix的CEO，目前他是MIT麻省理工学院客座教授。
- Stonebraker 教授领导了称为Postgres的后Ingres项目。这个项目的成果非常巨大，在现代数据库的许多方面都做出的大量的贡献。Stonebraker 教授还做出了一件造福全人类的事情，那就是把Postgres 放在了BSD 版权的保护下。如今Postgres名字已经变成了PostgreSQL，功能也是日渐强大。
- 87年左右，Sybase联合了微软，共同开发SQL Server。原始代码的来源与Ingres有些渊源。后来1994年，两家公司合作终止。此时，两家公司都拥有一套完全相同的SQLServer代码。可以认为，Stonebraker教授是目前主流数据库的奠基人。

- 1973 年 IBM 启动了 System R 项目，项目组发表了一系列关于关系数据库的文章。两个伯克莱大学的科学家，[Michael Stonebraker](#) 和 [Eugene Wong](#) 读了这些文章后非常感兴趣，于是决定自己启动一个关系数据的研究项目。他们已经为一个地理数据库系统申请到了资金，他们称为 ingres，意思是交互式图形获取系统
- Ingres 的代码是可以免费获得的，到 1980 年止，共分发了 1000 份拷贝，不少公司使用这些代码形成了自己的产品线。Informix 是最早的用户之一，并且其员工完全是 Ingres 项目的人员。他们在 1984 发布了基于 Ingres 的第一版产品，到 1997 年已经成了第二大数据库供应商。然而，一系列管理和财会方面的失误在短短两年内毁了公司的信誉，并于 2000 年被 IBM 收购。

- Robert Epstein 在伯克立大学项目中的重要程序员，创建了 Britton-Lee，后来的 Sybase，Sybase 在 80 到 90 年期间是第二号数据库产品，Sybase 于 1992 年将产品卖给微软，微软称之为 MS SQL Server。
- Stonebraker 自己离开了伯克立大学在 1982 年创建了 Ingres 公司，Stonebraker 在这个公司工作到 1991 年，然后公司卖给了 ASK，1994 年 ASK/ingres 被 CA Computer Associates 收购。2004 年 CA 在开源许可下发布了 Ingres release 3，并继续开发销售 Ingres。
- 在返回 Berkeley 之后，Stonebraker 开始了一项后-Ingres 计划来致力于解决关系模型的数据库管理现有实现的局限性。其中主要的是它们不能让用户定义组合更简单域的新域（或者叫“类型”）（参见关系模型获得对术语“域”的解释。）结果的计划叫做 Postgres，以介入对增加完整的类型支持所需要的最小数目的特征为目标。其中包括定义类型的功能，还有完全描述联系的能力 - 联系至今已经广泛使用但仍由用户完全维护。在 Postgres 中数据库“理解”联系，并能以使用规则的自然方式在有关联的表中检索信息。详情请参见 PostgreSQL 的文章。在 1990 年 Stonebraker 再次离开 Berkeley 去商业化 Postgres，使用了名字 Illustra。Illustra 后来被 Informix 并购，而 Stonebraker 再次回去搞高等教育了。

- 什么是关系？
- 候选码与主码
- 主属性与非主属性
- 第一范式（什么叫范式？）
- 关系模式

- 集合运算：并、差、交、笛卡尔积
- 关系运算：选择、投影、连接、自然连接、外连接、除法

- 什么是关系模式？
- 关系模式和关系有什么区别？



- 选择运算 $\sigma$
- 投影运算 $\pi$
- 集合运算 $\cup - \cap$
- 多元运算

笛卡尔积

连接

等值连接和自然连接

关系除法

试用关系代数表达式写出以下查询

- 1 列出部门10中工资超过1500的员工姓名
- 2 列出姓名以J开头并且总收入在2000-5000之间的员工姓名
- 3 列出1981年进入公司工作的员工
- 4 列出在达拉斯工作的员工

- 消除重复 $\delta$ （关于包）
- 聚组操作 $\gamma$
- 排序操作 $\tau$
- 空值符号
- 外连接
- 重命名 $\rho$

- 试用关系代数表达式写出以下查询
- 1 列出全体员工，按进入公司的先后次序排列
- 2 列出全体员工，按总收入从高到低排列
- 3 列出每位员工的姓名和其所在的部门名称、城市
- 4 求工资最高的员工姓名
- 5 求所有姓名以J开头的员工人数

- 课本（《数据库系统概论》第三版p162，第四版p269）
- 练习

证明关系代数中的选择串接定律 $\sigma_{F1}(\sigma_{F2}(E)) \equiv \sigma_{F1 \wedge F2}(E)$ ，其中E是关系代数表达式，F1和F2是选择条件

- 见P272图（第三版p165）



- 物理路径
- 算法

- 排序算法
- 聚组算法
- 消除重复的算法
- 连接算法

- 循环嵌套法
- 哈希连接法
- 排序合并法

- Dataguru（炼数成金）是专业数据分析网站，提供教育，媒体，内容，社区，出版，数据分析业务等服务。我们的课程采用新兴的互联网教育形式，独创地发展了逆向收费式网络培训课程模式。既继承传统教育重学习氛围，重竞争压力的特点，同时又发挥互联网的威力打破时空限制，把天南地北志同道合的朋友组织在一起交流学习，使到原先孤立的学习个体组合成有组织的探索力量。并且把原先动辄成千上万的学习成本，直线下降至百元范围，造福大众。我们的目标是：低成本传播高价值知识，构架中国第一的网上知识流转阵地。
- 关于逆向收费式网络的详情，请看我们的培训网站 <http://edu.dataguru.cn>



# Thanks

## FAQ时间