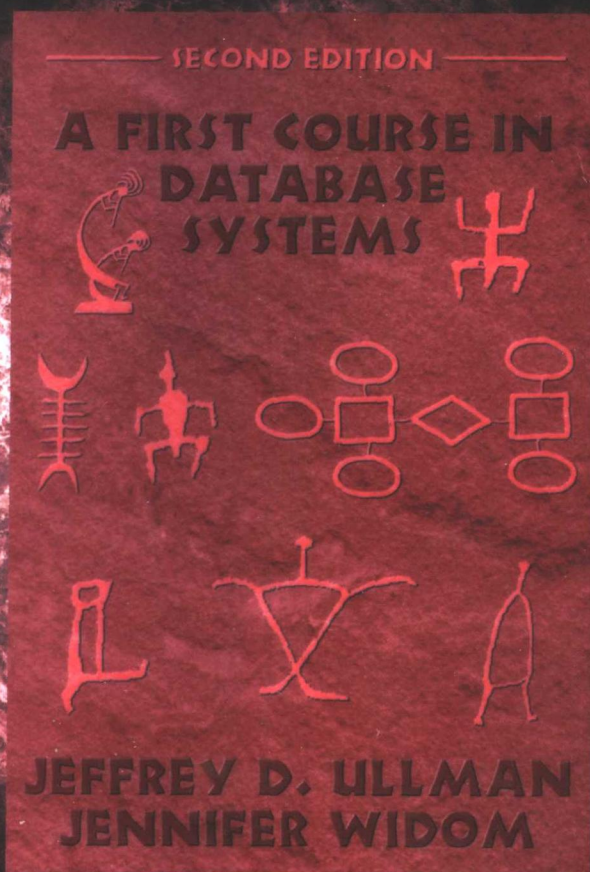


计 算 机 科 学 丛 书

数据库系统基础教程

(美) Jeffrey D. Ullman Jennifer Widom 著 岳丽华 龚育昌 等译



A First Course in Database Systems

机械工业出版社
China Machine Press

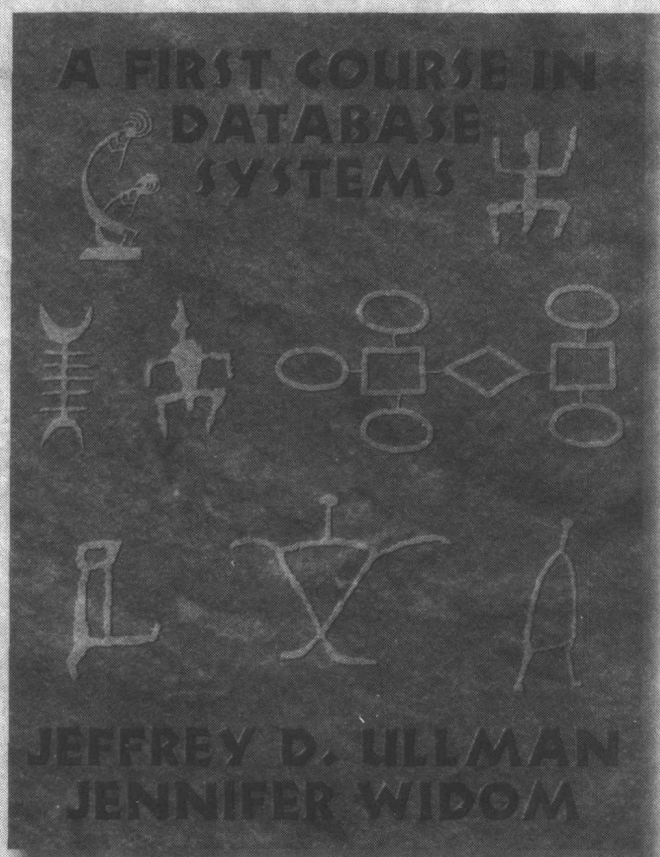


PEARSON
Prentice
Hall

计 算 机 科 学 丛 书

数据库系统基础教程

(美) Jeffrey D. Ullman Jennifer Widom 著 岳丽华 龚育昌 等译



A First Course in Database Systems



机械工业出版社
China Machine Press

本书由斯坦福大学知名计算机科学家Jeffrey D.Ullman和Jennifer Widom合作编写,涵盖了最新数据库标准SQL-99、SQL/PSM、SQL/CLI和ODL,而且比其他大多数书籍都更多地介绍了SQL内容,从数据库设计者、用户和应用程序员的角度深入浅出地介绍了数据库。

本书即可用作大学本科、研究生计算机及相关专业数据库课程教科书,也可用作数据库领域专业人员的参考书。

Simplified Chinese edition copyright © 2002 by PEARSON EDUCATION ASIA LIMITED and China Machine Press.

Original English language title: A First Course in Database System (ISBN: 0-13-035300-0) by Jeffrey D.Ullman and Jennifer Widom, Copyright 2002.

All rights reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Prentice Hall, Inc.

本书封面贴有Pearson Education(培生教育出版集团)激光防伪标签,无标签者不得销售。

本书中文简体字版由Prentice Hall, Inc.授权机械工业出版社独家出版。未经出版者书面许可,不得以任何方式复制或抄袭本书内容。

版权所有,侵权必究。

本书版权登记号:图字:01-2002-1881

图书在版编目(CIP)数据

数据库系统基础教程 / (美)沃尔曼(Ullman, J. D.)等著;岳丽华、龚育昌等译. -北京:机械工业出版社, 2003.8

(计算机科学丛书 数据库系列)

书名原文: A First Course in Database System

ISBN 7-111-10095-6

I. 数… II. ①沃… ②岳… III. 数据库系统-教材 IV. TP311.13

中国版本图书馆CIP数据核字(2003)第039003号

机械工业出版社(北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑:蒋 祎

北京中加印刷有限公司印刷·新华书店北京发行所发行

2003年8月第1版第1次印刷

787mm×1092mm1/16·21.25印张

印数:0 001-5 000册

定价:32.00元

凡购本书,如有倒页、脱页、缺页,由本社发行部调换

出版者的话

文艺复兴以降，源远流长的科学精神和逐步形成的学术规范，使西方国家在自然科学的各个领域中取得了垄断性的优势；也正是这样的传统，使美国在信息技术发展的六十多年间名家辈出、独领风骚。在商业化的进程中，美国的产业界与教育界越来越紧密地结合，计算机学科中的许多泰山北斗同时身处科研和教学的最前线，由此而产生的经典科学著作，不仅擘划了研究的范畴，还揭橥了学术的源变，既遵循学术规范，又自有学者个性，其价值并不会因年月的流逝而减退。

近年，在全球信息化大潮的推动下，我国的计算机产业发展迅猛，对专业人才的需求日益迫切。这对计算机教育界和出版界都既是机遇，也是挑战；而专业教材的建设在教育战略上显得举足轻重。在我国信息技术发展时间较短、从业人员较少的现状下，美国等发达国家在其计算机科学发展的几十年间积淀的经典教材仍有许多值得借鉴之处。因此，引进一批国外优秀计算机教材将对我国计算机教育事业的发展起积极的推动作用，也是与世界接轨、建设真正的世界一流大学的必由之路。

机械工业出版社华章图文信息有限公司较早意识到“出版要为教育服务”。自1998年开始，华章公司就将工作重点放在了遴选、移译国外优秀教材上。经过几年的不懈努力，我们与Prentice Hall, Addison-Wesley, McGraw-Hill, Morgan Kaufmann等世界著名出版公司建立了良好的合作关系，从它们现有的数百种教材中甄选出Tanenbaum, Stroustrup, Kernighan, Jim Gray等大师名家的一批经典作品，以“计算机科学丛书”为总称出版，供读者学习、研究及度藏。大理石纹理的封面，也正体现了这套丛书的品位和格调。

“计算机科学丛书”的出版工作得到了国内外学者的鼎力襄助，国内的专家不仅提供了中肯的选题指导，还不辞劳苦地担任了翻译和审校的工作；而原书的作者也相当关注其作品在中国的传播，有的还专诚为其书的中译本作序。迄今，“计算机科学丛书”已经出版了近百个品种，这些书籍在读者中树立了良好的口碑，并被许多高校采用为正式教材和参考书籍，为进一步推广与发展打下了坚实的基础。

随着学科建设的初步完善和教材改革的逐渐深化，教育界对国外计算机教材的需求和应用都步入一个新的阶段。为此，华章公司将加大引进教材的力度，在“华章教育”的总规划之下出版三个系列的计算机教材：除“计算机科学丛书”之外，对影印版的教材，则单独开辟出“经典原版书库”；同时，引进全美通行的教学辅导书“Schaum's Outlines”系列组成“全美经典学习指导系列”。为了保证这三套丛书的权威性，同时也为了更好地为学校和老师服务，华章公司聘请了中国科学院、北京大学、清华大学、国防科技大学、复旦大学、上海交通大学、南京大学、浙江大学、中国科技大学、哈尔滨工业大学、西安交通大学、中国人民大学、北京航空航天大学、北京邮电大学、中山大学、解放军理工大学、郑州大学、湖北工学院、中国国家信息安全测评认证中心等国内重点大学和科研机构在计算机的各个领域的著名学者组成“专家指导委员会”，为我们提供选题意见和出版监督。

这三套丛书是响应教育部提出的使用外版教材的号召，为国内高校的计算机及相关专业

的教学度身订造的。其中许多教材均已为M. I. T., Stanford, U.C. Berkeley, C. M. U. 等世界名牌大学所采用。不仅涵盖了程序设计、数据结构、操作系统、计算机体系结构、数据库、编译原理、软件工程、图形学、通信与网络、离散数学等国内大学计算机专业普遍开设的核心课程，而且各具特色——有的出自语言设计者之手、有的历经三十年而不衰、有的已被全世界的几百所高校采用。在这些圆熟通博的名师大作的指引之下，读者必将在计算机科学的宫殿中由登堂而入室。

权威的作者、经典的教材、一流的译者、严格的审校、精细的编辑，这些因素使我们的图书有了质量的保证，但我们的目标是尽善尽美，而反馈的意见正是我们达到这一终极目标的重要帮助。教材的出版只是我们的后续服务的起点。华章公司欢迎老师和读者对我们的工作提出建议或给予指正，我们的联系方式如下：

电子邮件：hzedu@hzbook.com

联系电话：(010) 68995264

联系地址：北京市西城区百万庄南街1号

邮政编码：100037

专家指导委员会

(按姓氏笔画顺序)

尤晋元
石教英
张立昂
邵维忠
周克定
郑国梁
高传善
裘宗燕

王 珊
吕 建
李伟琴
陆丽娜
周傲英
施伯乐
梅 宏
戴 葵

冯博琴
孙玉芳
李师贤
陆鑫达
孟小峰
钟玉琢
程 旭

史忠植
吴世忠
李建中
陈向群
岳丽华
唐世渭
程时端

史美林
吴时霖
杨冬青
周伯生
范 明
袁崇义
谢希仁

译者序

数据库已是当今信息社会须臾不可脱离的重要工具，数据库的教学也就成为计算机科学与技术专业的一门必修课程。为了使学生在大学学习中能够尽早地学习和使用计算机，许多院校将数据库课程教学分为两阶段教学，在低年级时，讲授数据库语言，以便使同学可以尽早地利用数据库语言来学习其他相关课程。另外，在我国大学计算机基础素质教育中，数据库也是其中重要的内容，也需要使学生建立较扎实的数据库语言基础，但是不需要更深入地了解数据库系统的内部结构，如数据库语言的查询处理，事务中的锁控制等等。对于这类课程的教学，Ullman教授的这本数据库基础教程则不失为一本恰当的好教材。

Jeffrey D. Ullman教授是斯坦福大学计算机系的资深教授，已从教近40年。自1980年编写了其第一本数据库教材《数据库系统原理》以来，已出版过多本数据库系统方面的教材。该书是他在斯坦福大学计算机系对大学生教授的第一门数据库课程（CS145）中使用的教材，其内容不仅介绍了当前流行的关系数据库标准语言SQL和传统的关系代数语言，而且还介绍了较新的面向对象语言OQL和逻辑语言DATALOG。该教材的主要特点是举例丰富详实，便于教师教学和自学者学习。书中在每一小节（而不是每一章）后都给出了大量的作业练习，并且标注了习题的难易程度，既便于教学安排，又便于学生循序渐进地掌握教学内容。另外，在Jeffrey D. Ullman教授的Web主页上还有关于该课程实验的内容，这对于本课程的实验教学有很大帮助。

本书由岳丽华负责翻译审校了1-8等章节，龚育昌负责翻译审校了9-10等章节。另外参加翻译工作的还有周英华、刘晓红、陆岚、杨洋、柳建平、齐冀等。

限于水平，译文中难免有错误与不足之处，欢迎读者批评指正。

译者

2003年5月

前 言

在斯坦福，因为执行的是一年四学期制，所以数据库引论课被分为两门课程。第一门课程是CS145，该课程中只要求学生学会使用数据库系统，而不必做关于DBMS实现方面的实验。CS145课是CS245的预修课。CS245是介绍DBMS实现。学生若想进一步学习数据库课程，可以学CS345（此课是理论课）、CS346（该课是DBMS实现实验课），以及CS347课程（该课介绍事务处理及分布式数据库）。

从1997年开始，我们已经出版了两本配套教材。《数据库系统基础课程》是为CS145课程编写的。《数据库系统实现》是为CS245课程以及部分CS346课程编写的。由于很多学校实行学期制，或者是将这两门数据库引论课组合成一门引论课程，因此，我们感到有必要将上述两本书合成一本《数据库系统全书》。

然而，与数据库的实现相比较，更多的学生需要知道如何使用数据库，我们仍然根据最初的《数据库系统基础课程》中的内容编成此书，另外加入了《数据库系统全书》一书前十章中关于建模与编程的新内容，这些新内容包括SQL/PSM（存储程序），SQL/CLI（C/SQL接口标准）和JDBC（与JAVA/SQL相同）。同时，我们将内容重新加以组织，从实体-联系模型的内容中删去了面向对象模型的处理；从关系代数的内容中删去对逻辑查询的讨论后，加入了新的关于操作符的内容。这些操作符不仅用于传统关系代数，而且用于表示SQL语义。

警告 此书是全书的前半部分，如果除本书内容外，您还希望学习关于数据库实现的部分，您应该购买《数据库系统全书》一书。

如何使用该书

本书中的内容讲半个学期绰绰有余，但如果只有四分之一学期的时间，就要省略一些内容了。我们认为2~7章是本书的核心。余下三章的内容可任意处置，虽然我们认为每个学生都应从第8章的一节中学会如何在宿主语言中嵌入SQL语句。

若如同我们在CS145课程中所做的那样，你想给学生一个真实的数据库应用设计和实现项目作业，则应该对书的讲解顺序做某些调整，较早开始对SQL的介绍。虽然学生在做数据库设计时需要规范化知识，但你可以推迟有关函数依赖的介绍。

预备知识

我们曾经将此书作为本科生和一年级研究生所修课程的教材。该课程正常的条件下是二年级课程，在此之前已学习过：（1）数据结构、算法、离散数学。（2）软件系统、软件工程和程序设计语言等。最重要的是学生至少要对如下内容有基本的理解：代数表达式和代数定律；逻辑、基本的数据结构如查询树和图；面向对象程序设计概念和程序设计环境。可是我们认为对于一般的计算机专业课程体系而言，最合适的学习时机是在修完三年级后。

习题

本书几乎在每一节都有补充练习，我们用感叹号对难题做了标记，对最难的习题用双感叹号做了标记。

有些习题或习题的小题部分标有星号，对这些习题我们将努力使读者可以通过该书的网页了解该题的相关解法，以方便读者。这些解是公开有效的，并且可用于自测。注意，在有些情况下习题B是对另一习题A的修改或调整，所以，如果A的某一部分有解，你也应该能解出B的对应部分。

WWW网上支持

本书的主页地址是：

<http://www-db.stanford.edu/~ullman/fcdb.html>

这里有对加星号习题的解的勘误表及书的备份。同时还有与课程CS145相关的作业、项目及考卷等内容。

致谢

有很多人曾帮助过我们，他们或是提供了本书及它的后续版内容的最初材料，或者是提供了本书或其他网上材料的勘误表。我们对所有这些帮助过我们的人表示感谢。他们是：

Marc Abromowitz, Joseph H. Adamski, Brad Adelberg, Gleb Ashimov, Donald Aingworth, Jonathan Becker, Margaret Bentiez, Larry Bonham, Phillip Bonnet, David Brokaw, Ed Burns, Karen Buter, Christopher Chan, Sudarshan Chawathe, Per Christensen, Ed Chang, Surajit Chaudhuri, Ken Chen, Rada Chirkova, Nitin Chopra, Bobbie Cochrane, Arturo Crespo, Linda DeMichiel, Tom Dienstbier, Pearl D' Souza, Oliver Duschka.

还有Xavier Faz, Greg Fichtenholtz, Bar Fisher, Jarl Friis, John Fry, Chiping Fu, Tracy Fujieda, Manish Godara, Meredith Goldsmith, Luis Gravano, Gerard Guillemette, Rafael Hernandez, Antti Hjelt, Ben Holtzman, Steve Huntsberry, Leonard Jacobson, Thulasiraman Jeyaraman, Dwight Joe, Seth Katz, Yeong-Ping Koh, Gyorgy Kovacs, Phillip Koza, Brian Kulman, Sang Ho Lee, Olivier Lobry, Lu Chao-Jun, Arun Marathe, Le-Wei Mo, Fabian Modoux, Peter Mork, Mark Mortensen.

还有Ramprakash Narayanaswami, Hankyung Na, Marie Nilsson, Torbjorn Norbye, Chang-Min Oh, Mehul Patel, Bert Porter, Limbek Reka, Prahash Ramanan, Ken Ross, Tim Roughgarden, Mema Roussopoulos, Richard Scherl, Catherine Tornabene, Anders Uhl, Jonathan Ullman, Mayank Upadhyay, Vassilis Vassalos, Qiang Wang, Kristian Widjaja, Janet Wu, Sundar Yamunachari, Takeshi Yokukawa, Min-Sig Yun, Torben Zahle, Sandy Zhang。当然书中的错误是我们的。

J. D. U.

J. W.

斯坦福，加州

目 录

出版者的话	2.1.11 E/R模型中的子类	19
专家指导委员会名单	2.1.12 习题	20
译者序	2.2 设计原则	22
前言	2.2.1 忠实性	22
第1章 数据库系统世界	2.2.2 避免冗余	22
1.1 数据库系统的发展	2.2.3 简单性考虑	23
1.1.1 早期的数据库管理系统	2.2.4 选择正确的联系	23
1.1.2 关系数据库系统	2.2.5 选择正确的元素种类	24
1.1.3 越来越小的系统	2.2.6 习题	25
1.1.4 越来越大的系统	2.3 约束的建模	26
1.1.5 客户-服务器和多层体系结构	2.3.1 约束的分类	27
1.1.6 多媒体数据	2.3.2 E/R模型中的键	27
1.1.7 信息集成	2.3.3 E/R模型中键的表示	28
1.2 数据库管理系统概述	2.3.4 单值约束	29
1.2.1 数据定义语言命令	2.3.5 引用完整性	29
1.2.2 查询处理概述	2.3.6 E/R图中的引用完整性	29
1.2.3 存储器和缓冲区管理器	2.3.7 其他类型的约束	30
1.2.4 事务处理	2.3.8 习题	30
1.2.5 查询处理器	2.4 弱实体集	31
1.3 数据库系统研究概述	2.4.1 弱实体集的来源	31
1.3.1 数据库设计	2.4.2 弱实体集的要求	32
1.3.2 数据库程序设计	2.4.3 弱实体集的符号	33
1.4 小结	2.4.4 习题	33
1.5 参考文献	2.5 小结	33
第2章 实体-联系数据模型	2.6 参考文献	34
2.1 E/R模型的要素	第3章 关系数据模型	35
2.1.1 实体集	3.1 关系模型的基础	35
2.1.2 属性	3.1.1 属性	35
2.1.3 联系	3.1.2 模式	35
2.1.4 实体-联系图	3.1.3 元组	36
2.1.5 E/R图实例	3.1.4 域	36
2.1.6 二元E/R联系的多样性	3.1.5 关系的等价描述	36
2.1.7 多路联系	3.1.6 关系实例	36
2.1.8 联系中的角色	3.1.7 习题	37
2.1.9 联系的属性	3.2 从E/R图到关系设计	37
2.1.10 多路联系到二元联系的转换	3.2.1 实体集到关系的转化	38

3.2.2 E/R联系到关系的转化	39	3.8 小结	75
3.2.3 组合关系	40	3.9 参考文献	76
3.2.4 处理弱实体集	41	第4章 其他数据模型	79
3.2.5 习题	43	4.1 面向对象概念的复习	79
3.3 子类结构到关系的转化	44	4.1.1 类型系统	80
3.3.1 E/R方式转化	44	4.1.2 类和对象	80
3.3.2 面向对象的方法	45	4.1.3 对象标识	80
3.3.3 使用空值组合关系	46	4.1.4 方法	80
3.3.4 各种方法的比较	46	4.1.5 类的层次	81
3.3.5 习题	46	4.2 ODL简介	81
3.4 函数依赖	47	4.2.1 面向对象设计	81
3.4.1 函数依赖的定义	48	4.2.2 类声明	82
3.4.2 关系的键	48	4.2.3 ODL中的属性	82
3.4.3 超键	50	4.2.4 ODL中的联系	83
3.4.4 找出关系中的键	50	4.2.5 反向联系	84
3.4.5 习题	51	4.2.6 联系的多重性	85
3.5 函数依赖的规则	52	4.2.7 ODL中的方法	86
3.5.1 分解/结合规则	52	4.2.8 ODL中的类型	87
3.5.2 平凡函数依赖	53	4.2.9 习题	88
3.5.3 计算属性的闭包	53	4.3 ODL中的其他概念	89
3.5.4 为什么能用闭包算法	55	4.3.1 ODL的多路联系	90
3.5.5 传递规则	56	4.3.2 ODL中的子类	90
3.5.6 函数依赖的闭包集合	57	4.3.3 ODL中的多继承	91
3.5.7 投影函数依赖	57	4.3.4 范围	92
3.5.8 习题	58	4.3.5 ODL中键声明	92
3.6 关系数据库模式设计	59	4.3.6 习题	94
3.6.1 异常	60	4.4 从ODL设计到关系设计	94
3.6.2 分解关系	60	4.4.1 从ODL属性到关系属性	95
3.6.3 Boyce-Codd范式	61	4.4.2 类中的非原子类型属性	95
3.6.4 分解为 BCNF	63	4.4.3 集合类型属性的表示	96
3.6.5 从分解中恢复信息	66	4.4.4 其他类型构建器的表示	97
3.6.6 第三范式	67	4.4.5 ODL中联系的表示	98
3.6.7 习题	69	4.4.6 如果没有键会怎样?	100
3.7 多值依赖	69	4.4.7 习题	100
3.7.1 属性独立及伴随其产生的冗余	69	4.5 对象关系模型	101
3.7.2 多值依赖的定义	70	4.5.1 从关系到对象关系	102
3.7.3 多值依赖的推论	71	4.5.2 嵌套关系	102
3.7.4 第四范式	72	4.5.3 引用	103
3.7.5 分解到第四范式	73	4.5.4 面向对象与对象关系的比较	104
3.7.6 范式间的联系	74	4.5.5 ODL设计到对象关系设计的转化	105
3.7.7 习题	74	4.5.6 习题	105

4.6 半结构化数据.....105	5.4.4 分组操作符139
4.6.1 为何需要半结构化数据模型106	5.4.5 扩展的投影操作符140
4.6.2 半结构化数据表示106	5.4.6 排序操作符141
4.6.3 信息集成与半结构化数据107	5.4.7 外连接141
4.6.4 习题108	5.4.8 习题142
4.7 XML及其数据模型108	5.5 关系的约束.....143
4.7.1 语义标签109	5.5.1 作为约束语言的关系代数143
4.7.2 格式规范的XML109	5.5.2 引用完整性约束144
4.7.3 文档类型定义 (DTD).....110	5.5.3 其他的约束举例145
4.7.4 使用DTD111	5.5.4 习题146
4.7.5 属性列表112	5.6 小结.....147
4.7.6 习题113	5.7 参考文献.....147
4.8 小结.....114	第6章 数据库语言SQL.....149
4.9 参考文献.....114	6.1 SQL中的简单查询.....149
第5章 关系代数117	6.1.1 SQL中的投影150
5.1 一个数据库模式的例子.....117	6.1.2 SQL中的选择152
5.2 关系代数操作.....118	6.1.3 字符串比较153
5.2.1 关系代数基础119	6.1.4 日期和时间154
5.2.2 关系中的集合操作119	6.1.5 空值和涉及空值的比较155
5.2.3 投影120	6.1.6 布尔值UNKNOWN156
5.2.4 选择121	6.1.7 输出排序157
5.2.5 笛卡儿积122	6.1.8 习题157
5.2.6 自然连接122	6.2 多个关系上的查询.....158
5.2.7 θ 连接123	6.2.1 SQL中的积和连接159
5.2.8 使用组合操作生成查询124	6.2.2 避免属性歧义159
5.2.9 重命名125	6.2.3 元组变量160
5.2.10 依赖的和非依赖的操作.....126	6.2.4 多关系查询的解释161
5.2.11 关系代数表达式中的线性符号.....127	6.2.5 查询的并、交、差163
5.2.12 习题.....127	6.2.6 习题164
5.3 包上的关系操作.....132	6.3 子查询.....165
5.3.1 为什么采用包132	6.3.1 产生标量值的子查询165
5.3.2 包的并、交、差133	6.3.2 含有关系的条件表达式166
5.3.3 包的投影操作134	6.3.3 含有元组的条件表达式167
5.3.4 包的选择135	6.3.4 关联子查询168
5.3.5 包的笛卡儿积135	6.3.5 FROM子句中的子查询169
5.3.6 包的连接135	6.3.6 SQL的连接表达式169
5.3.7 习题136	6.3.7 自然连接170
5.4 关系代数的扩展操作.....137	6.3.8 外连接171
5.4.1 消除重复137	6.3.9 习题172
5.4.2 聚集操作符138	6.4 全关系操作.....174
5.4.3 分组138	6.4.1 消除重复174

6.4.2 交、并、差中的重复	174	7.2.4 习题	209
6.4.3 SQL 中的分组和聚集	175	7.3 修改约束	210
6.4.4 聚集操作符	175	7.3.1 给约束命名	210
6.4.5 分组	176	7.3.2 修改表上约束	210
6.4.6 HAVING子句	177	7.3.3 习题	211
6.4.7 习题	178	7.4 模式层的约束和触发器	211
6.5 数据库更新	179	7.4.1 断言	212
6.5.1 插入	179	7.4.2 事件-条件-动作规则	214
6.5.2 删除	181	7.4.3 SQL中的触发器	214
6.5.3 更新	181	7.4.4 替换触发器 (Instead of Triggers)	217
6.5.4 习题	182	7.4.5 习题	217
6.6 SQL中的关系模式定义	183	7.5 小结	219
6.6.1 数据类型	183	7.6 参考文献	219
6.6.2 简单表定义	184	第8章 SQL 的系统特征	221
6.6.3 修改关系模式	184	8.1 编程环境下的SQL	221
6.6.4 默认值	185	8.1.1 阻抗不匹配问题	222
6.6.5 索引	185	8.1.2 SQL/宿主语言接口	222
6.6.6 索引选择简介	186	8.1.3 DECLARE节	223
6.6.7 习题	188	8.1.4 使用共享变量	223
6.7 视图定义	189	8.1.5 单元组选择语句	224
6.7.1 视图声明	189	8.1.6 游标	224
6.7.2 视图查询	190	8.1.7 游标修改	227
6.7.3 重命名属性	191	8.1.8 防止并发更新	227
6.7.4 视图更新	191	8.1.9 卷型游标	228
6.7.5 涉及视图的查询解释	193	8.1.10 动态SQL	229
6.7.6 习题	195	8.1.11 习题	230
6.8 小结	195	8.2 模式中的存储过程	231
6.9 参考文献	196	8.2.1 创建PSM函数和过程	231
第7章 约束和触发器	199	8.2.2 PSM中的简单语句格式	232
7.1 键和外键	199	8.2.3 分支语句	233
7.1.1 主键声明	199	8.2.4 PSM中的查询	234
7.1.2 用UNIQUE声明键	200	8.2.5 PSM中的循环	234
7.1.3 强制键约束	201	8.2.6 For 循环	236
7.1.4 外键约束声明	201	8.2.7 PSM的异常处理	236
7.1.5 维护引用完整性	202	8.2.8 使用PSM函数和过程	238
7.1.6 延迟约束检查	203	8.2.9 习题	238
7.1.7 习题	205	8.3 SQL 环境	240
7.2 属性和元组上的约束	206	8.3.1 环境	240
7.2.1 非空值约束	206	8.3.2 模式	240
7.2.2 基于属性的CHECK约束	207	8.3.3 目录	241
7.2.3 基于元组的CHECK约束	208	8.3.4 SQL 环境中的客户和服务	242

8.3.5 连接	242	9.1.6 子查询	273
8.3.6 会话	243	9.1.7 习题	274
8.3.7 模块	243	9.2 OQL 表达式的其他格式	276
8.4 使用调用层接口	243	9.2.1 量词表达式	276
8.4.1 SQL/CLI简介	244	9.2.2 聚集表达式	277
8.4.2 处理语句	245	9.2.3 分组表达式	277
8.4.3 从查询结果中取数据	246	9.2.4 HAVING 子句	279
8.4.4 向查询传递参数	248	9.2.5 并、交和差操作	279
8.4.5 习题	248	9.2.6 习题	280
8.5 Java数据库连接	248	9.3 OQL 中对象的赋值与创建	281
8.5.1 JDBC 简介	248	9.3.1 宿主语言变量的赋值	281
8.5.2 JDBC 中的创建语句	249	9.3.2 集合元素的提取	281
8.5.3 JDBC 中的游标操作	250	9.3.3 获取集的每一个成员	281
8.5.4 参数传递	250	9.3.4 OQL 中的常量	282
8.5.5 习题	251	9.3.5 创建新对象	283
8.6 SQL 中的事务	251	9.3.6 习题	284
8.6.1 可串行性	251	9.4 SQL 中的用户定义类型	284
8.6.2 原子性	253	9.4.1 在SQL 中定义类型	284
8.6.3 事务	254	9.4.2 用户定义类型中的方法	285
8.6.4 只读事务	255	9.4.3 用UDT声明关系	286
8.6.5 读脏数据	256	9.4.4 引用	286
8.6.6 其他隔离级别	257	9.4.5 习题	288
8.6.7 习题	258	9.5 对象关系数据上的操作	288
8.7 SQL 中的安全机制和用户认证	259	9.5.1 引用的跟随	288
8.7.1 权限	259	9.5.2 访问UDT类型元组的属性	289
8.7.2 创建权限	260	9.5.3 生成器和转换器函数	290
8.7.3 检查权限的处理	261	9.5.4 UDT类型联系的排序	291
8.7.4 授权	262	9.5.5 习题	292
8.7.5 授权图	263	9.6 小结	293
8.7.6 销权	264	9.7 参考文献	293
8.7.7 习题	266	第10章 逻辑查询语言	295
8.8 小结	267	10.1 一种关系逻辑	295
8.9 参考文献	268	10.1.1 谓词和原子	295
第9章 面向对象查询语言	269	10.1.2 算术原子	295
9.1 OQL简介	269	10.1.3 Datalog 规则和查询	296
9.1.1 一个面向对象的电影例子	269	10.1.4 Datalog 规则的意义	297
9.1.2 路径表达式	269	10.1.5 扩展谓词和内涵谓词	298
9.1.3 OQL 中 Select-From-Where 表 达式	271	10.1.6 Datalog规则应用于包	299
9.1.4 修改结果的类型	271	10.1.7 习题	300
9.1.5 复杂输出类型	272	10.2 从关系代数到Datalog	300
		10.2.1 交	300

10.2.2 并.....	300	10.3.2 计算递归Datalog 规则	307
10.2.3 差.....	301	10.3.3 递归规则中的非	311
10.2.4 投影.....	301	10.3.4 习题.....	313
10.2.5 选择.....	301	10.4 SQL 中的递归	314
10.2.6 积.....	303	10.4.1 在SQL 中定义IDB关系	314
10.2.7 连接.....	303	10.4.2 分层非.....	316
10.2.8 用 Datalog 模拟多重操作	304	10.4.3 有问题的递归SQL表达式	317
10.2.9 习题.....	305	10.4.4 习题.....	319
10.3 Datalog 的递归编程.....	306	10.5 小结	320
10.3.1 递归规则.....	307	10.6 参考文献	320

第1章 数据库系统世界

今天数据库已是每一项业务的基础。数据库被应用于维护商业内部记录，在万维网上为顾客和客户显示数据，以及支持很多其他商业处理。数据库同样出现在很多科学研究的核心中。天文学家、人类基因研究者和探索蛋白质医药性质的生物学家，以及其他很多科学家搜集的数据也是用数据库表示的。

数据库的能力来自于已发展了数十年的知识和技术，这些知识和技术蕴藏在被称做数据库管理系统（database management system）的专业化软件中。该软件也被称做DBMS，或更通俗地称为“数据库系统”。DBMS是一个能有效建立和管理大量数据的强大工具，并且能安全地长期保存这些数据。数据库系统是最复杂的软件系统之一。DBMS为用户提供的功能如下：

1. 持久存储 如同文件系统，DBMS支持对独立于应用的超大量数据的存储。然而，DBMS不仅只是在灵活性上优于文件系统，而且在数据结构上，还支持对超大量数据的有效访问。

2. 程序设计接口 DBMS允许用户或应用程序通过强有力的查询语言对数据进行访问和修改。而且，DBMS比文件系统更具优点，它不仅仅提供对文件的读写，而且还具备以更复杂的方式管理数据的灵活性。

3. 事务管理 DBMS支持数据的并发存取，即可以同时有很多不同的进程（称做“事务”）对数据访问。为了避免同时存取产生不期望的结果，DBMS要支持独立性（isolation）——一次执行一个事务，原子性（atomicity）——事务要么全部执行要么全部不执行。DBMS还支持持久性（durability），即具有能从很多类故障和错误中恢复的能力。

1.1 数据库系统的发展

数据库是什么？本质上讲，数据库就是信息的集合。这种集合可以存在很长时间，通常是很多年。一般讲，数据库是指由DBMS管理的数据的集合。DBMS需要有如下功能：

1. 允许用户使用专门的数据定义语言（data-definition language）建立新的数据库，并说明它们的模式（schema），即数据的逻辑结构。

2. 使用合适的查询语言（query language）或数据操作语言（data-manipulation language），为用户提供查询（query）和更新（modify）数据的能力。“查询”是数据库关于数据的提问的术语。

3. 支持超大数据量（吉字节或更多）数据的长时间存储，防止对数据意外的或非授权的访问，并且在数据库查询和更新时支持对数据的有效存取。

4. 控制多个用户对数据的立即存取，不允许一个用户的操作影响另一个用户，也不允许同时存取对数据的意外破坏。

1.1.1 早期的数据库管理系统

第一个商用数据库管理系统出现在20世纪60年代末。这些系统都是来自于文件系统，它们提供某些上述第3项功能；文件系统可以长期地存储数据，并且允许大数据量存储，可是，如果数据不做备份，文件系统通常并不保证数据不会丢失。当不知道数据项在某个文件中的存储位置时，文件系统也不提供数据的有效访问。

更进一步说，文件系统不直接支持第2项功能，即没有对文件的查询语言。对第1项功能的

支持（即数据模式的支持）也只限于文件目录结构的建立。最后，文件系统不满足第4项功能。当有多个用户或进程对文件并发访问时，文件系统不能防止两个用户对同一个数据的修改，于是将出现一个用户的修改被丢失的情形。

在DBMS第一批重要的应用中，数据由很多小数据项组成，并且完成很多查询或修改。下面是一些例子。

飞机订票系统

这类系统通常包括如下数据项：

1. 单个旅客预定某个航班的机票，包括的信息有座位分配、饮食习惯等。
2. 关于航班的信息，有飞机起飞和到达的机场、飞机起飞和到达时间或运营的飞机等等。
3. 有关机票价格、机票需求和剩余机票等数据。

典型的查询有：在某个时间范围内从一个城市到另一个城市的航班，有什么样的舱位，价格如何。典型的数据修改包括：旅客航班预定、坐席分配或注明饮食习惯等。在任一给定时间，很多代理机构将访问上述部分数据。DBMS必须允许这样的并发存取，并防止两个代理将同一个坐席同时分配给两个旅客这样的问题发生。当系统突然发生故障时还要保护数据不丢失。

银行系统

银行系统包括的数据有：顾客姓名、地址、账号、贷款、余额，以及顾客和他们的账号与贷款间的关联，例如，谁在哪个账号上有签字权。关于账目余额的查询是最常见的应用，但更常见的应用是修改，表现为对某个账号的付款或存款。

如同在飞机订票系统中一样，人们期望很多出纳员和顾客（通过ATM机或Web）能即时查询和修改银行数据。同时对一个账号的访问不会引起事务丢失的作用也极其重要。故障是不能容忍的。例如，一旦钱已从ATM机中弹出，即使是立即发生电源断电，银行也必须记录下这笔借方账目。另一方面，如果银行已在借方记账，但由于断电而未能付款，这种情况也是不允许的。处理这类操作的恰当方法并不那么简单，它被看作是DBMS方式的优点之一。

公司资料/数据

很多早期的应用涉及公司资料/数据，例如每次的销售记录、收支账目或职工信息（如他们的名字、地址、工资、福利选择、缴税状况，等等）。查询应用包括：打印应收款项、职工每周的薪金。每次销售、采购、账单、收据、职工雇用、辞退及晋升等，都导致数据库的修改。

早期的DBMS是从文件系统发展而来，它助长用户用接近数据的方式显现数据。这些数据库系统使用多个不同的数据模型描述数据库中的信息结构，主要的有基于树结构的模型、“层次”模型和基于图结构的“网状”模型。网状模型通过CODASYL（数据系统及语言委员会）的报告在20世纪60年代末被标准化^①。

早期模型和系统的一个问题是它们不支持高级查询语言。例如，CODASYL查询语言的语句只允许用户通过指向数据元素的指针，从一个数据元素跳到另一个数据元素。因此，即使是写一个非常简单的查询程序，用户也要花费很大的力气。

1.1.2 关系数据库系统

随着1970年Ted Codd著名论文的发表^②，数据库系统有了重大的改变。Codd提出数据库系统应该将数据组织成表的形式呈现给用户。这种形式称做关系（relation）。在关系的后面，可

① CODASYL “Data Base Task Group” April, 1971报告, ACM, New York.

② Codd, E. F., “大量共享数据库的关系模型” Comm. ACM 13:6 pp. 377-387.