

最畅销的 **SQL 指导手册**

代码通过 Oracle,
Microsoft,
Sybase 和
Informix 测试

The Practical SQL Handbook (Fourth Edition)

实用参考手册

第四版

SQL

使用 SQL 各种版本

Judith S. Bowman

Sandra L. Emerson 著

Marcy Darnovsky

康 博 译



清华大学出版社
<http://www.tup.tsinghua.edu.cn>



Pearson Education 出版集团



CD 包括 Sybase Adaptive
Server Anywhere 软件和
在所有例子中使用的范例
数据库。

The Practical SQL Handbook (Fourth Edition)

SQL 实用参考手册

(第四版)

Judith S. Bowman

Sandra L. Emerson 著

Marcy Darnovsky

康 博 译

清 华 大 学 出 版 社

Pearson Education 出版集团

(京) 新登字 158 号

北京市版权局著作权合同登记号: 01-2001-4056

内 容 简 介

SQL 是一种关系型数据库语言, 用于定义、查询、修改和控制关系型数据库中的数据, 它受到了关系型数据库管理系统开发者的广泛支持。本书介绍 SQL 的基础知识, 提供了具有商业应用价值的实际解决方案, 并研究关系数据库领域各个知名公司的不同 SQL 版本。本书包括了扩展的平台 SQL 知识, 以及根据 SQL 用户反馈信息编写的各种实际示例。

本书主要内容包括: 详细介绍了创建数据库、表和索引以及添加、修改和删除数据的各个命令; 使用 SELECT 命令检索具体的数据; 在关系数据库中处理 NULL 值; 连接表, 包括自连接和外连接; 使用嵌套查询(子查询)从多个表中获取数据; 创建视图(虚拟表)以实现数据的定制访问功能; 使用 SQL 函数。

本书适用于所有使用关系型数据库系统的用户。

Simplified Chinese edition copyright © 2002 by Pearson Education NORTH ASIA LIMITED and Tsinghua University Press.

The Practical SQL Handbook: Fourth Edition by Judith S.Bowman, Sandra L.Emerson, Marcy Darnovsky, Copyright © 2002.

All Rights Reserved.

Published by arrangement with Pearson Education, Inc., publishing as PH PTR.

This edition is authorized for sale only in the People's Republic of China (excluding the Special Administrative Region of Hong Kong and Macau).

本书中文简体字版由清华大学出版社和美国培生教育出版集团合作出版。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, 翻印必究。

本书封面贴有 Pearson Education 出版集团激光防伪标签, 无标签者不得销售。

书 名: SQL 实用参考手册(第四版)

作 者: Judith S. Bowman, Sandra L. Emerson, Marcy Darnovsky 著 康博 译

出 版 者: 清华大学出版社(北京清华大学学研大厦, 邮编 100084)

<http://www.tup.tsinghua.edu.cn>

责任编辑: 夏兆彦

印 刷 者: 中国科学院印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 787×1092 1/16 印张: 22.75 字数: 582 千字

版 次: 2002 年 4 月第 1 版 2002 年 4 月第 1 次印刷

书 号: ISBN 7-900641-64-5

印 数: 0001~5000

定 价: 46.00 元(含一张光盘)

第四版前言

如果您正在使用与 SQL 兼容的数据库或现在准备学习使用这种数据库，那么您一定会喜欢本书。

当第一次看到这本书时，我已经作了 7 年的数据库顾问了，然而在处理比较复杂的数据库查询(自我连接)时还是遇到了问题。我浏览了一下本书的目录就发现了所需要的章节，在第三版中，这就是第 7 章“连接表进行综合数据分析”。在阅读第 7 章时，我感到本书语言非常流畅，示例和图形使用得当，该章内容讲解得很透彻。该章内容首先介绍了如何通过使用简单而又便于使用的语言从几个表中提取信息。该章不仅仅介绍了基础知识，它还清晰而又透彻地介绍了一些高级的查询知识。

在解决了遇到的问题后，我花费了一天的时间来释读本书中的其他章节，如“解决业务问题”和“错误和解决错误的方式”。通过学习这些章节，使我真正地掌握了如何使用 SQL。

本书讲述了我以前所阅读的书中所没有的内容，我认为本书的优点在于它站在专家的角度向初学者讲述知识。学习本书就像在厨房向一个专业厨师学习烹调一样，而不是像看一些冰箱、烤箱和搅拌器的说明书。

作为 Institute for Global Communications(全球通信学院，这是位于 San Francisco 的全球培训网络)的技术负责人，我主要指导编程初学人员如何在他们的程序中集成 SQL 数据库，并要求他们阅读本书的开始几章。一旦学习了“设计数据库”、“创建和填充数据库”和“从数据库中选择数据”这些章节之后，就为他们以后的学习打开了坚实的基础，这样我们就可以编写用于 Cold Fushion, Perl, PHP 和 Microsoft Access 的网络应用程序。

本书可以面向使用 Microsoft SQL Sever, Oracle 和其他与 SQL 兼容的数据库的编程人员和非编程人员，并且适用于任何使用 SQL 的人员。无论您是使用 Microsoft SQL Sever, Oracle，还是本书中不支持的那些数据库(比如开放源代码 mysql Server)，您也可以应用书中的绝大多数内容。编程人员可以通过使用本书来学习如何创建、查询和修改 SQL 数据库。对于非编程人员，他们也可以尽快掌握数据库的工作方式及其原因，可以明白如何改善数据库，如何从公司数据库中提取所需信息等知识。

第四版比起第三版来说，其优越性在于新版本介绍了更多的针对数据库服务器的信息，提供了更多示例代码，并根据多年来热情读者反馈的信息做了进一步的解释。

如果您正在使用 SQL 或将要学习如何使用 SQL，我建议您购买本书。本书无论是结构还是内容都非常优秀，能够最大限度地帮助那些希望学习如何使用 SQL 的人。它从入门知识开始，提供了许多精彩的示例和专家建议。祝您好运！



Michale de Beer

原全球通信学院技术负责人

<http://www.igc/madebeer>

February 2001

Michale de Beer 是 Association for Progressive Communications(APC)的顾问, 该组织是一个推进计
算机网络普及的国际联合机构, 由 130 多个国家的非官方机构和积极分子组成。

前言

推出新版本的原因

自从 1989 年本书第一次问世以来,技术领域发生了巨大的变化,SQL 也不例外。SQL 语言在用户数量和命令数量方面都大大地扩展了,关系数据库的市场份额不断稳步上升。

我们在编写本书第一版时, American National Standards Institute(ANSI, 美国国家标准协会)已通过 1986 年的 SQL 标准, 且 International Standards Organization(ISO, 国际标准化组织)在 1987 年采用了该标准。1989 年的版本就是在 ANSI 和 ISO 的帮助下制定的。1986 年的 SQL 标准简单、缺少大部分商业开发商提供的功能, 1989 年该标准得到了完善, 但还有许多重要内容没有定义。

在本书第一版中, 我们主要关注了业界的实际作法: 通常, 每一位开发商都要关注其他的开发商所做的工作, 他们所提供的产品的核心内容都非常类似, 以便在竞争中吸引其他开发商的顾客群和正在寻找数据库系统以便开发数据库程序的新用户。因此, 我们没有介绍还不太定型的 ANSI 标准和各个开发商具体的实现(这些问题由在这些领域的专家来讨论), 我们只介绍了一些常见内容: 通用的或业界使用的 SQL。我们的目标是向聪明的读者提供关于如何实际有效地使用 SQL 的信息。

1992 年的 ANSI 标准(常称为 SQL-2 或 SQL-92)是 SQL 技术发展的一个新的里程碑, 该标准比起 1989 年的标准更全面: 所包含的内容比以前的版本多了 4 倍。数据库开发商接受了大部分 1992 年的标准, 随着 SQL-92 标准的普遍推广, 业界的处理方法和 ANSI / ISO 标准开始接轨。

尽管开发商各不相同, 但是却存在有普遍适用于业界的核心 SQL 命令, 所有的用户都必须理解这些命令。贯彻标准并不是一两天的事情, 而是一个长期的过程, 因此, 开发商也要不断地适应并继续研制出相应的产品。本书的目的是要 SQL 用户掌握该语言的基础知识, 并了解一下各个具体版本的特征。

新版本的修改之处

新版本的变更有以下三部分:

- 包含更多实际的示例。
- 强调许多开发商实现的 SQL-92 特征。
- 提供可方便使用的软件。

1. 包含更多实际的示例

在和一些新的开发 SQL 的用户交流时, 我们常常听到他们说, 需要有更多的示例来



学习、修改、摘录有价值的作法并以这些示例为基础进行扩展。相应地，在第二版中的许多新添加的素材包括许多代码“清单”。第 11 章“解决业务问题”的内容就是根据提出的问题和来自计算机新闻组的方法而选择的代码样本。我们在这一章使用本书的样例 bookbiz 数据库来讲解那些有趣的问题和解决方案。第 11 章还包括示例，即使用 CASE 函数进行条件判断、格式化结果和查找日期数据。各种样本对应于不同的类型，它们并不能解决所有的问题，而仅仅包括使用 DISTINCT 遇到的问题和对 SQL 功能的错误理解范例。

2. 强调 SQL-92

在第三版中，我们对本书作了修正以加入大部分开发商接受的 SQL-92 标准，包含新的数据类型，增加了允许内置完整性约束的 CREATE TABLE 语句，修改了 ORDER BY 子句和 GROUP BY 子句，用于 LIKE 关键字的新转义字符，改变 GRANT 和 REVOKE 等。

3. 提供可以方便使用的软件

使用本书配套光盘上的测试版软件 Sybase Adaptive Server Anywhere(ASA)，您可以在计算机上运行该示例(或您自己的修改后的示例)。通常我们都认为学习 SQL 的关键在于实践，现在，您可以随意进行练习，使用您所知道的数据对代码进行测试并检查是否能得到所预期的结果。当您觉得使用复杂的代码比较困难时，您可以将其拆成一个个小的完整的程序段，再单独运行这些小程序段，您要能理解每个代码段，然后再将它们组合成一个复杂的大程序。

配套光盘上的 Sybase Adaptive Server Anywhere 测试版的期限是 60 天，这个测试版本允许您创建对象和查询现有的对象。使用更新版的 bookbiz 数据库可以构建 ASA，对于第四版，我们增加了图书的价格并更新了最新的日期，配套光盘上还包括脚本，这些脚本可以在 ASA 上和本书介绍的其他系统上创建该数据库(防止破坏源数据库)。

关于第 4 版的说明

本书第 4 版继续重点讨论了业界开发的 SQL，但其所包含的信息比以前的版本更通用，也更具体。在第 4 版中，我们有两个目标：

- 扩展测试基础，分别在 5 个不同的数据库系统上运行。
- 显示针对各个开发商的系统区别。

1. 在更多的系统上测试

第 4 版中所有的示例都可以在 5 个系统中运行(Sybase Adaptive Server Anywhere, Adaptive Server Enterprise, Microsoft SQL Server, Oracle, Informix)，而不是像第 3 版那样只能在 Sybase 系统中运行，这种可扩展的测试基础使得信息更具有普遍性。这样我们可以不再局限于在某个特殊的系统中所研究和校验的内容，如果在 Adaptive Server Anywhere(ASA, 即 CD 上包含的 RDBMS)中不存在某个重要的功能，我们就可以从 Oracle 或 SQL Server 系统中显示这个功能。

2. 显示针对开发商不同产品的示例

另一方面，本书第 4 版更具有针对性，它包含了有关 5 种 SQL 产品的特性信息，告诉读者这些产品的区别。因此这就符合了实际情况——许多数据库用户都依次或同时使用多个系统。感兴趣的 SQL 的区别被标记为 SQL Variants，这是显示不同开发商在 SQL 的用法中的具体区别的文本内容。例如，如何使用单引号和双引号，如何添加数据库用户，如何指定外连接。您不必每次都去查看 5 个系统中的每一个系统的代码，但是要在某一个特定的问题上知道它们的区别。

致谢

在这里我们要感谢为本书做出贡献的下列人员：Donna Jeker 和 Stu Schuster 提供了及时的支持和鼓励；Jeff Lichtman 和 Howard Torf 提供了建议、示例、一些小故事和仔细的校对；Tom Bondur, Susie Bowman, John Cooper 和 Wayne Duquesne 提供了源代码和其他信息；Paul Winsberg 对第 1 版中数据库章节作了评论；Karen Ali 对第 3 版的 SQL Anywhere CD 作了大量工作；Theo Posselt 对第 3 版作了技术评论。

对于第 4 版，我们要感谢 iAnywhere Solutions(一家 Sybase 公司)的 Mike Radencich 提供了 Adaptive Server Anywhere 软件和对该软件光盘的使用许可；Tilden Park Software 的 Lance Batten 和 Paragon Software 的 Sanford Jacobs 提供了注释和更正；我们还要感谢下列评论家提供的及时而又详细的反馈信息：Oracle Corporation 的教学负责人 Vijayanandan Venkatachalam；Alternative Technologies 的 David McGoveran；Roger Snowden；Sticksel Data Systems 的 Amy Sticksel；Emerald Solutions 的 Paul Irvine 和 Karl Batten-Bowman。

序 言

SQL 的起源

SQL 是由 IBM 发明的。

SQL 是 Structured Query Language 的首字母缩写,它属于一种一体化的语言,用于定义、查询、修改和控制关系型数据库中的数据。它的正式发音是 ess-cue-ell(按照 American National Standards Institute),但许多人发音为 sequel。在本书中,SQL 的发音为 sequel。

数据库管理系统的关系模型是由 IBM 研究实验室位于 San Joes, California 的 E.F.Codd 博士于 1970 年提出,在随后的十年中,许多大学和实验室对其继续进行了研究。SQL,作为由早期工作发展出的几种语言之一,一直是主要的关系型数据库语言。尽管有不同类型的开发商,但关系型数据库管理系统的开发商都支持 SQL。美国国家标准化组织和国际标准化组织共同提出了 SQL 标准的一些版本,并还在制定新的标准。

在 1970 到 1980 年前后,低性能的关系型数据库管理系统制约了它们的商业生存能力。在早期关系系统中,关系模型的重要支柱——数学上的有效描述和直观的显示——不能解决大型数据库管理系统既慢又难以使用的问题,可以说这些问题有时根本就不可能解决。直到出现了两种变化才改变了这种情况:即大容量、高速度的计算机和开发了优秀的数据检索、数据存储和数据访问方法,这些方法支持关系系统的后台处理功能。

1981 年,IBM 发布了它的第一个基于 SQL 的商业产品 SQL / DS。在 80 年代早期,Oracle, Relational Technology 等其他几个开发商也纷纷发布了各自的基于 SQL 的关系型数据库管理系统。到 1989 年,至少出现了 75 种以上的 SQL 或类似于 SQL 的数据库管理系统,这些系统运行在各种计算机上,从单用户微机到处理多用户计算机。今天,在公司、政府和公益组织的全球网络上,人们使用 SQL 处理“关键任务型”信息管理和数据处理的任务。

关系型数据库管理系统市场的强烈竞争导致出现了各种各样的 SQL 实现版本,每一种版本都是开发商在开发完整和高效关系模型表达语言方面的最大努力。随着 SQL-92 标准的广泛实施,开发商的各种 SQL 实现版本逐渐趋于统一,许多实现版本都有一组语法和语意相似的核心命令。另外,许多开发商也不断地扩展他们的 SQL 版本,使这种语言不断地发展。

SQL 在不断地发展,因为 SQL 的原始设计方案在许多方面都比较含糊(因为数据库开发业界难以等到这些问题得到澄清)。即使业界实现了更多 SQL 标准,这些标准也只是对 SQL 实现版本进行封装。最后,更重要的是,基于 Internet 应用程序中数据库的作用决定了 SQL 的作用就是在面向世界的公共范围内安全地管理多媒体对象。

幸运的是,SQL 很灵活,也容易扩展。作为关系型数据库的主要语言,SQL 正在不



断地变化以满足市场的需要。

SQL 的商业化进程

除了贯彻标准外, SQL 实现版本由于处于顶层 RDBMS 开发商间的竞争已经趋于统一(本来存在大量数据库开发商,但由于合并、兼并和其他市场活动,此行业逐渐只剩下了少数开发商)。当然,各个公司都试图向未来的用户提供同样内容的 SQL 函数和扩展,也试图吸引其他关系系统的用户并试图尽量减少从一种基于 SQL 系统向另一种系统转换时的代价。

这些外因和标准使得产品的商业化实现版本更为趋进,总之,商业利益和标准委员会会导致 SQL 具有以下特点:

- 实现日益完善的关系模型。
- 尽量减少不兼容性和实现方式的独立性。
- 为新的函数详细添加语言以维持与早期实现版本的兼容性。

本书(配套光盘中含有 Adaptive Server Anywhere 的限时版本)有助于让您掌握 SQL 的基础知识,然后您可以将所学到的知识应用到其他的 SQL 版本。一旦理解了这些基础知识,您就会发现在不同版本间的调整非常容易。

本书面向的读者对象

如果要使用一个关系型数据库系统,则您就是本书的使用读者——无论您是和其他用户共享该系统还是在一个单用户个人计算机上单独使用。本书假设您是一位聪明的业余编程人员,无论您是在大公司、政府部门、非盈利部门的最终用户,或做小生意的老板、小公司的经理、处理个人事物的家庭计算机用户和学习数据库技术的学生。您可以从一个 PC 文件管理器转而使用关系型数据库系统,从不基于 SQL 的数据库管理系统转换到使用基于 SQL 的数据库管理系统,或一开始就使用数据库管理系统。

我们假定您对计算机和计算机工具有一定程度的了解,当然,最好能熟悉数据库系统。如果要开发一个复杂的数据库应用程序,您可能需要在编码语言中嵌入 SQL(如 C++ 或 Java)或将 SQL 和脚本语言一起使用(如 XML 或 JavaScript)。但是您不必编写一行编程代码或脚本代码就可以成功地使用本书,并学习 SQL 基础知识和掌握许多更高级的主题。

本书不面向理论高深的读者。我们认为读者对高深的关系理论和复杂的 ISO-ANSI 争论不感兴趣。另一方面,我们认为您应该了解有关 SQL 的争论,这样您可以知道使用 SQL 的一些技巧。总之,我们认为您需要了解真正实用的知识,并且能最快地掌握这些知识。

相应地,本书的重点是学习交互式使用 SQL——输入命令并直接在屏幕上获取结果,这一点不同于与将它们嵌入到编程语言中。每一种商业 SQL 实现版本都有一个交互式的使用界面,以利于学习该语言和编写特殊查询。许多实现版本也提供了一组工具,如报表编写器、管理控制台,还提供了一个与 SQL 相关使用的高级脚本语言,以开发具有一定复杂程度的应用程序。

本书的重点

本书重点讨论商用 SQL 的实际应用——“工业 SQL”——而不是重点讨论 ANSI 标准给出的 SQL 抽象主题，这是因为工业 SQL 在注释范围、面向的对象和总体功能上提供一个更好的教学工具。

- 用于记录大部分工业 SQL 的语法注释相当清楚，并且可以直观地理解。另一方面，ISO-ANSI 出版物使用 BNF(Backus Naur Form)注释，该注释非常明确但很难阅读和理解。

- 所有的工业 SQL 对于初学或偶尔使用 SQL 的用户都支持交互式界面。对于程序员和应用程序开发人员来说，ISO-ANSI 标准主要是处理嵌入式 SQL 界面。

- 关系型数据库系统的开发商实现客户所需要的功能，其中也包括由竞争者提出的一些功能。采用 ANSI SQL 特征的方法在各个 SQL 版本之间都不同。本书以业界的处理方法(不是 ANSI 标准)作为基础，重点讨论了现今可以使用的功能。

ANSI 标准是一个大得惊人的文档，其中包含了子句、防止误解的说明和脚注。它的 BNF 注释非常适合于获知每一个语言元素的功能，而不是它的语法。如果您要进行研究，我们建议您以 C.J.Date 的《SQL 标准指南》为学习指南。Date 是关系模型的理论家，是关于关系模型的多产作家之一。他的作品介绍了如何阅读 BNF，并对 SQL 和标准的优缺点提供了自己的简明观点。

在反复思考了 ISO-ANSI 文档后，即使使用了 Date 的指南帮助，您可能也要倾向于使用最新的关系型数据库管理方面的用户手册，但是用户手册也可能存在缺陷和局限性。

虽然一些用户手册适合于学习 SQL 基础知识，但是大多数手册非常简单和模糊。也许更大的问题是，用户手册(包括本书的作者编写的内容)自然要强调某一种版本语法的细节和特点——这就使读者难以清楚概念之间的总体关系。

在使用最好的用户手册都难以解决问题的情况下，就需要使用本书。本书逐步介绍 SQL 基础知识，然后介绍如何设计基于 SQL 的数据库应用程序。本书还讨论了在用户手册中常常忽略或过于简化的主题：数据库设计、索引、无效值、连接、子查询、视图、性能和数据完整性。

使用本书学习 SQL 的方法

首先我们提出一些要求。第一，希望您能阅读本书的大部分内容。第二，希望您研究和复制一些示例。第三，希望您能练习、测试和研究一些问题。没有任何方法可以替代交互式练习，即使您的最终目标是编写一个非常复杂的应用程序也要有这个过程。

在这里，有许多工作需要您来完成。学习方式各个不同：一些人通过深入思考泛泛的解释而吸收新的知识，一些人仅仅通过观察图片就可以理解概念。无论是详细阅读本书的全部内容，还是简短浏览本书内容，并希望通过研究示例来进行学习，我们希望本书都能有所帮助。

在本书中一些示例比较简单，其目的是演示基本概念。其他的例子则比较复杂。在复杂示例的末尾您可以看到一些 SQL 语句，可以作为您的应用程序的模型。无论示例有多么复杂，我们都将对它们进行剖析和详细解释。



SQL 的每一个版本都互不相同，至少在细节上有所不同，因此，没有通用的 SQL 书籍能保证所有的示例都像所希望的那样正确地使用。现在，由于本书以广泛的商业 SQL 知识为基础，使得在各个系统之间的转换过程变得更加直接。通过参考 SQL 产品提供的材料，以及在附录 B 中跨系统的关键字和运算符类比列表，再经过思考之后，您就可以在所使用的 SQL 或类似于 SQL 的系统上测试大部分示例。用于测试代码的系统见表 0-1 所示。本书中的所有示例都可以在测试的系统中使用，我们没有深入讨论 SQL 的任何单个实现版本，而是讨论在业界 SQL 不同版本间的共性部分。

表 0-1 本书使用的关系型数据库管理系统

开 发 商	软 件	版 本	工 具	备 注
Sybase	Adaptive Server Anywhere(ASA) 也称为 SQL Anywhere	6.0.3	交互式 SQL	包 含 在 附 带 光 盘 上
Sybase	Adaptive Server Enterprise(ASE)	11.5	SQL Advantage	T-SQL
Microsoft	SQL Server	7.0	Query Analyzer	T-SQL
Oracle	Personal Oracle	8.1.7	SQL Plus	
Informix	Informix	7.30	dbaccess	

所有的示例都源自样本数据库 bookbiz，几乎没有什么例外。第 2 章和第 3 章解释了该样本数据库。您可以不使用与本书相同的样本数据库，但如果使用相同的数据库，则将是最好的方法，它将有助于您快速看到结果是否正确。

bookbiz 数据库非常小，在您解决一个企业级的工程以前使用这种小而简单的数据库比较适合。您可以复制 Adaptive Server Anywhere 提供的 bookbiz 来完成本书中的示例，然后再重建自己系统中的数据库。另一方面，bookbiz 又具有一定的复杂性，可以说明关系问题中的重要方面。

本书的结构安排

在本书的每一章都介绍了一个 SQL 功能或许多相关功能，它们以下列的形式进行讨论：

- 定义：做什么。
- 微型语法：命令的一种简单通用版本，此版本中去除了不同 SQL 版本之间中不同的子句和扩展。关于语法约定的解释见第 3 章。
- 简单的示例。

在最初对语法和用法描述之后，我们详细论述了在关系模型中这种功能的作用和在数据库应用程序中它的可能用法。

必要的话，我们还可以提供附加的语法——提供附加功能或精心调制的性能的可选子句——以及更复杂的示例。这样，对每一个新的功能就都有了完整的描述和示例。

要尽可能以前面的示例为基础构建后面的示例，但是，由于每一章的示例都是单独的，因此您可以在该章构建独立的实例。学习 SQL 类似于学习其他任何外语一样：开始是模仿，随后才可以理解，最后达到熟练应用的程度。在每一个阶段，成功的关键都是大量练习。

使用本书的方法

如果您遵照下面过程，将可以节省非常多的时间，和 SQL 交互练习也将更舒适和更有效：

- 在操作系统文件中存储您练习的 SQL 语句

(您的系统应该提供能实现这种方法的方法。)如果您不能确定 SQL 查询是否(数据检索操作)正产生预期的结果，就要存储该结果，以便脱机检查。您也要记录可以使用的内容和不可以使用的的内容，如果可能，还要存储错误信息。

- 存储成功的方案

将设计优秀的 SQL 方案存储到文件：可能在后面要针对不同的目的对它们进行模拟。

- 将应用程序查询构造为一个单独的模块或子程序

像结构模块化编程一样，优秀的 SQL 应用程序应该由多个子例程组成并可以不断地再应用和再循环。

- 准备记录便条

即使系统已经提供了一个快速参考卡片，也一定要准备自己的清单和快速查找常用命令的卡片，这样将巩固您的学习效果。不久您将发现有些 SQL 命令比其他的命令用得更多频繁。

- 改进您的解决方案

根据我们的经验，使用 SQL 越多，所编写的语句就会变得越简单和越精致。

学习 SQL 和交互式使用 SQL 可以正确地编写任何期望的操作，从而得到所需的结果。要达到这种熟悉程度，您需要探索和试验关系系统的 SQL。您肯定不希望发现，当运行大容量多用户系统时，您所期望的结果被一个 SQL 错误破坏了。

因为 SQL 是外来语言，而且掌握它的实体——系统语法分析器——并不代表您自己已掌握了这种语言，所以您还需要不断练习才可以顺利使用 SQL。尽管 SQL 的关键字和操作符有限且相对容易掌握，但有些方面还是需要技巧。像许多计算机高级语言一样，SQL 具有明确的语法和结构，还有一些具体的句法规则。SQL 可能更像 English，但它并不是一种自然性语言。在使用过程中，您总会遇到一些 SQL 不能执行的操作符。

本书将帮助您学习 SQL 的强大功能和缺点，有助于防止由于糟糕的数据库设计或不实用的、不可维护的基于 SQL 的应用程序而带来的潜在危险，并且它还将使得学习 SQL 更轻松快捷。

本书概况

- 第 1 章：SQL 和关系型数据库管理系统



本章简单地定义和非正式说明关系模型，并将以 SQL 语言的主要特征来说明关系模型。

- 第 2 章：设计数据库

设计数据库通常是一种令人生畏的工作。本章介绍最有帮助的技术，使用示例数据库演示数据的分析和与数据库设计相关的决策。它讨论主键码和外键码、实体-关系模型和标准化规则，这是用于设计优秀数据库的指导方法。

- 第 3 章：创建和填充数据库

在本章实现上一章提出的数据库设计方案，详细介绍 SQL 命令，如创建数据库、表、索引，添加、修改和删除数据等。关于 SQL 句法约定的解释和初始化信息一起放在 SQL 语言的使用手册中。

- 第 4 章：从数据库中选择数据

在本章，您可以使用光盘上的 Adaptive Server Anywhere 来运行示例。本章给出了 SELECT 命令的基本元素，解释如何从单个的表中检索行和列并讨论计算值，比较运算符和逻辑运算符。

- 第 5 章：数据排序和其他选择技术

在 SELECT 语句中的其他子句允许对数据排序，从结果中消除复制数据，使用聚集函数计算平均值、列总值和行总值。

- 第 6 章：对数据分组并形成报表

SELECT 语句也可以使用上一章介绍的聚集函数对数据分组并汇总为报表。本章还要解决关系型数据库管理系统如何处理空值的问题。

- 第 7 章：连接表进行综合数据分析

连接操作是关系模型的特点之一。本章介绍如何使用联合操作从一个表或多个表中检索数据。连接是对简单选择功能的复杂应用，解决了用户在分析和验证数据检索结果方面的重要问题。

- 第 8 章：使用子查询构造查询

本章讨论嵌套查询或子查询的正确用法和应用方面，并使用大量示例介绍相关的子查询。

- 第 9 章：创建和使用视图

本章讨论视图和视图在定制访问数据方面的应用。视图也可以提供数据安全性，因为您可以规定其他用户使用指定的操作访问表的指定部分。在本章也将详细介绍更新视图的棘手问题。

- 第 10 章：安全性、事务、性能和完整性

本章介绍在现实数据库管理中存在的问题，它介绍了 SQL 命令用于指定权限、返回索引的主题，讨论它在提高性能方面的作用，并介绍了事务管理机制，描述了 SQL 语言的扩展以提供数据库的一致性和推理完整性。

- 第 11 章：解决业务问题

在本章您可以练习在前面章节学到的技巧，根据在 Internet 上发现的问题和答案来练

习 SQL 代码实例，并根据 bookbiz 数据库复制这些问题和解决办法。本章还将学习 CASE 和其他条件判断函数，编写分离列数据和组合列数据的代码，转换数据类型和学习处理日期问题。本章结尾介绍如何避免产生错误。

- 附录 A：本书中使用的关于 SQL 的语法总结

在本书中使用的有关 SQL 约定的注释包含在 SQL 语句的列表中。

- 附录 B：业界的 SQL 版本

比较图表可以让您看到 RDBMSs 在命名约定、数据类型和 SQL 函数之间的区别。

- 附录 C：术语表

在这里按约定的字母顺序列出一些重要的术语。

- 附录 D：bookbiz 样例数据库

本附录包含了一些数据图表、数据结构图表和 CREATE 语句以及 INSERT 语句。

- 附录 E：资源

本附录列出了一些可以参考的书籍、网站和新闻组。

目 录

第 1 章 SQL和关系型数据库管理系统	1
1.1 关系型数据库管理系统	1
1.2 关系模型：关系就是表	2
1.3 独立性	3
1.4 高级语言	4
1.4.1 数据操纵	4
1.4.2 数据定义	5
1.4.3 数据管理	5
1.5 关系操作	6
1.5.1 投影	6
1.5.2 选择	7
1.5.3 连接	7
1.6 查看数据的另一种方法	9
1.7 NULL	10
1.8 安全性	11
1.9 完整性	11
1.10 SQL 函数	12
1.11 小结	13
第 2 章 设计数据库	14
2.1 设计考虑因素	14
2.1.1 是否需要专家指导	15
2.1.2 数据定义和定制	15
2.1.3 如何进行数据库设计	15
2.1.4 介绍样例数据库	18
2.2 数据实体和关系	19
2.2.1 实体：独立存在的事物	19
2.2.2 一对多关系	22
2.2.3 多对多关系	24
2.2.4 一对一关系	25
2.2.5 实体-关系方法总结	25



2.3	规范化准则	26
2.3.1	第一范式	27
2.3.2	第二范式	28
2.3.3	第三范式	29
2.3.4	第四范式	30
2.4	回顾数据库设计	31
2.4.1	总结 bookbiz 数据库	33
2.4.2	测试您的数据库设计方案	33
2.4.3	定义数据库的其他考虑问题	34
2.5	小结	34
第 3 章	创建和填充数据库	35
3.1	从理论到实践	35
3.1.1	区分 SQL 示例和语法	36
3.1.2	应对失败情况	37
3.2	使用数据库	38
3.2.1	开始	38
3.2.2	识别角色	39
3.2.3	为数据库对象预留存储空间	39
3.2.4	选择数据库	40
3.3	创建简单的表	40
3.3.1	选择数据类型	42
3.3.2	指定 NULL 和 NOT NULL	44
3.3.3	计划表	45
3.3.4	定义 bookbiz 数据库中的表	46
3.4	创建索引	46
3.4.1	CREATE INDEX 语句	47
3.4.2	同索引相关的事务	48
3.5	使用 SQL-92 约束创建表	49
3.5.1	概述约束	49
3.5.2	在单列上实现约束	51
3.5.3	实现多列约束	52
3.6	修改、删除数据库和对象	54
3.6.1	修改数据库	54
3.6.2	修改表定义	54
3.6.3	删除数据库	55
3.6.4	删除表	55