

The Practical SQL Handbook

# 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：

### 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，邮政编码：100084)

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

责任编辑：夏兆彦

印 刷 者：

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：182 千字

版 次：2001 年 4 月第 1 版 2001 年 4 月第 1 次印刷

书 号：ISBN 7-900641-64-5

印 数：0001 ~ 5000

定 价：46.00 元

# 本书光盘使用说明

删除的内容: The Practical SQL Handbook

带格式的: 项目符号和编号

(1) 本光盘包括：

？ Sybase Adaptive Server Anywhere 60 天测试版本；

？ 本书中用到的 bookbiz 样例数据库；

？ 用于说明的 readPSH4.txt 文件；

？ 在其他 DBMS 系统(MS SQL Server、Oracle、Informix 和 Sybase Adaptive Server Enterprise)——但本光盘未提供用于这些 DBMS 的软件)上用于创建 bookbiz 表的文件。

删除的内容:

带格式的

(2) 要装入 Adaptive Server Anywhere，将本光盘插入光驱。如果没有立即启动安装程序，从 Anywhere 目录中选择正确的平台目录(CE、Netware、Win16 或 Win32)并启动 setup.exe 文件 执行命令指示。如果空间不够，可以安装容量最小的 Personal Server。Sybase Central 和 Help 文件是有用的附加项。

(3) 一旦已安装了 Adaptive Server Anywhere，使用 Windows Explorer(或类似的实用程序)将 bookbiz.db 文件从光盘复制到硬盘(如需复制指令，可以打印本光盘的 read PSH4.txt 文件即可)。

删除的内容: 你

(4) 启动 Adaptive Server Anywhere/Interactive SQL(从 Windows 中的 Programs 菜单中或按系统约定)。

(5) 打开 Connect to Adaptive Server Anywhere 窗口(若该窗口未打开，在 Interactive SQL 窗口底部第三个的 command 栏输入“connect”，并单击 Execute，这样将弹出 Connect to Adaptive Server Anywhere 窗口)。

(6) 在注册行输入 DBA(必须大写)，在口令行输入 SQL。

(7) 不必单击 OK 按钮，在窗体顶部单击 Database 标签即可。

(8) 在 Database name 栏输入 bookbiz，在 Database file 栏输入 bookbiz.zb 的全路径名称(见第 3 步，例如，如果您要复制 bookbiz.db 至 C:\SQL，则 Database file 的地址为 C:\SQL\bookbiz.db)。

删除的内容: 你

(9) 在窗体的底部单击 OK，则 Interactive SQL 窗口的顶部标题栏提示您在 bookbiz 中的名称为 DBA。现在已经建立并运行 bookbiz 数据库。

删除的内容: 你

(10) 在 Interactive SQL 窗口的 command 栏输入样例查询，如“select \* from publishers”。当单击 Execute 按钮时，则在 Interactive SQL 窗口的顶部的 Data 栏显示出结果。

删除的内容: 你

(11) 使用 Sybase Central(它可能在 Programs 菜单中显示为“Manage Adaptive Server Anywhere”)可以添加用户、选择表列定义等等。在 Sybase Central 下的 Tools 菜单选择 connect 选项可以进入 Sybase Central，弹出 Connect to Adaptive Server Anywhere 窗口，执行上面的步骤 6~9 即可。

(12) 在光盘的 CREATE、INSERT 和 View 目录中的脚本用于创建表、添加数据和创建视图。脚本和附录 A 中含有使用它们的指令。如果要恢复删除的表或数据，可以使用这些文件。

删除的内容: 你

(13) 其他 RDBMS(Oracle、SQL Server、Informix 和 Sybase ASE)的用于创建 bookbiz 表的脚本也包含在本光盘的 CREATE、INSERT 和 View 文件中。

# 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)时不是遇到了问题。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 SQL1-92

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，配套光盘上还包含脚本，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。

# 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

带格式的

带格式的

带格式的

3.6.5 删除索引	56
3.7 添加、修改和删除数据	56
3.7.1 添加新行：INSERT	57
3.7.2 改变现有数据：UPDATE	60
3.7.3 删除数据：DELETE	63
3.8 小结	63
第 4 章 从数据库中选择数据	65
4.1 SELECT 概述和语法	65
4.1.1 基本的 SELECT 语法	65
4.1.2 完整的 SELECT 语法	68
4.2 选取列：SELECT 子句	69
4.2.1 选择所有的列：SELECT*	69
4.2.2 选择特定的列	70
4.2.3 多个列名	71
4.3 指定表：FROM 子句	79
4.3.1 使用表别名	80
4.3.2 忽略 FROM	80
4.4 选择行：WHERE 子句	81
4.4.1 比较运算符	82
4.4.2 带有逻辑运算符的连接条件	84
4.5 限制范围(BETWEEN 和 NOT BETWEEN)	90
4.5.1 列表(IN 和 NOT IN)	92
4.5.2 选择空值	93
4.5.3 匹配字符串：LIKE	95
4.5.4 IN 与 LIKE 相同吗	99
4.5.5 三者之间的比较	101
4.6 小结	101
第 5 章 数据排序及其他选择技术	102
5.1 一批新的 SELECT 语句中的子句	102
5.2 排序查询结果 ORDER BY	102
5.2.1 ORDER BY 语法格式	102
5.2.2 字符集和顺序排序	104
5.2.3 排序中的排序	105
5.2.4 升序排序，降序排序	107
5.2.5 编写更复杂些的表达式	107

5.2.6	如何对空值排序 .....	110
5.3	删除复制行：DISTINCT 和 ALL .....	111
5.3.1	DISTINCT 语法格式 .....	113
5.3.2	带有多个 SELECT 列表项的 DISTINCT .....	113
5.3.3	DISTINCT 和非选择列表 ORDER BY .....	115
5.4	聚集函数 .....	116
5.5	小结 .....	125
第 6 章	对数据分组并形成报表 .....	126
6.1	数据分组及聚集 .....	126
6.2	GROUP BY 子句 .....	126
6.2.1	GROUP BY 语法 .....	126
6.2.2	注意事项及限制条件 .....	128
6.2.3	Null 及组 .....	132
6.2.4	GROUP BY：聚集交互作用 .....	134
6.2.5	带有 WHERE 子句的 GROUP BY .....	137
6.2.6	GROUP BY 和 ORDER BY .....	138
6.3	HAVING 子句 .....	139
6.3.1	普通的 HAVING .....	140
6.3.2	WHERE, GROUP BY, HAVING 和 ORDER BY 语句 .....	143
6.4	关于 NULL 的介绍 .....	143
6.4.1	NULL 和数据库设计 .....	144
6.4.2	涉及 NULL 的比较 .....	144
6.4.3	NULL 及运算 .....	145
6.4.4	默认选择 NULL 值 .....	146
6.4.5	与 NULL 共同使用的函数 .....	146
6.4.6	NULL 函数和 “what-if” 运算 .....	147
6.5	小结 .....	147
第 7 章	连接表用于综合数据分析 .....	149
7.1	定义连接 .....	149
7.2	需要连接的原因 .....	149
7.2.1	将各个单独表中的数据联系在一起 .....	150
7.2.2	提供灵活性 .....	150
7.3	获得良好的连接 .....	151
7.3.1	FROM/WHERE 连接语法 .....	152
7.3.2	SQL-92 连接语法 .....	152

7.3.3	分析连接	154
7.4	提高连接结果的可读性	155
7.4.1	避免重复	155
7.4.2	限制显示的宽度	156
7.4.3	在 FROM 子句的表/视图列表中使用别名	157
7.5	指定连接条件	158
7.5.1	基于相等的连接	158
7.5.2	不基于相等的连接	160
7.5.3	多于两个表的连接	161
7.6	研究特殊连接	161
7.6.1	与自身连接的表：自连接	162
7.6.2	显示背景：外部连接	164
7.7	避免常见错误	168
7.7.1	理解笛卡尔积	168
7.7.2	使用笛卡尔积	169
7.7.3	约束笛卡尔积	170
7.8	其他运算：UNION、INTERSECT、MINUS	170
7.8.1	UNION	171
7.8.2	INTERSECT 和 MINUS	175
7.9	小结	176
第 8 章	使用子查询构造查询	177
8.1	子查询的含义	177
8.2	子查询的工作过程	178
8.2.1	简单的子查询过程	179
8.2.2	关联子查询过程	180
8.2.3	简单 - 关联子查询性能问题	181
8.3	使用联合还是使用子查询	181
8.3.1	子查询	181
8.3.2	连接	182
8.3.3	子查询与自连接	183
8.3.4	哪一种更好	183
8.4	子查询规则	184
8.5	返回 0 或更多值的子查询	184
8.5.1	由 IN 引入的子查询	184
8.5.2	由 NOT IN 引入的子查询	186
8.5.3	由 IN 引入的关联子查询	187

8.6	返回单独值的子查询 .....	194
8.6.1	保证单值的聚集函数 .....	195
8.6.2	GROUP BY 和 HAVING 返回单个值 .....	195
8.6.3	使用比较运算符进行关联子查询 .....	196
8.7	子查询测试数据的存在性 .....	197
8.7.1	NOT EXISTS 查询空集 .....	199
8.7.2	使用 EXISTS 查找交集和差集 .....	200
8.7.3	EXISTS 的替代查询方法 .....	201
8.8	多个嵌套层中的子查询 .....	201
8.9	在 UPDATE, DELETE 和 INSERT 语句中的子查询 .....	202
8.10	在 FROM 和 SELECT 子句中的子查询 .....	203
8.10.1	在 FROM 子句中的子查询 .....	203
8.10.2	在 SELECT 子句中的子查询 .....	205
8.11	小结 .....	206
<b>第 9 章</b>	<b>创建和使用视图 .....</b>	<b>207</b>
9.1	使用 View Toward Flexibility .....	207
9.2	视图命令 .....	207
9.2.1	创建视图 .....	208
9.2.2	通过视图显示数据 .....	209
9.2.3	删除视图 .....	210
9.3	视图的优点 .....	210
9.3.1	显示、简化和定制 .....	214
9.3.2	安全性 .....	214
9.3.3	独立性 .....	214
9.4	视图的工作方式 .....	215
9.4.1	命名视图列 .....	216
9.4.2	使用多个优先对象创建视图 .....	218
9.4.3	分解视图 .....	220
9.5	使用视图修改数据 .....	224
9.6	创建数据副本 .....	228
9.7	小结 .....	229
<b>第 10 章</b>	<b>安全性、事务、性能和完整性 .....</b>	<b>231</b>
10.1	实际的数据库管理 .....	231
10.2	数据安全性 .....	232
10.2.1	用户身份和特定的用户 .....	232

---

---

10.2.2	GRANT 和 REVOKE 命令	234
10.2.3	作为安全性机制的视图	237
10.3	事务	238
10.3.1	事务和并行处理	238
10.3.2	事务和恢复	239
10.3.3	用户定义事务	239
10.4	性能	241
10.4.1	基准测试	241
10.4.2	设计和进行索引	242
10.4.3	优化查询	242
10.4.4	监控和提高性能的其他工具	245
10.5	数据完整性	247
10.5.1	域约束	247
10.5.2	实体完整性	248
10.5.3	参照完整性	248
10.5.4	存储过程和触发器	249
10.6	小结	249
第 11 章	解决业务问题	251
11.1	在工作中使用 SQL	251
11.2	考虑有条件的情况	252
11.2.1	CASE/DECODE	252
11.2.2	改变 NULL 显示	255
11.3	格式化数据和显示数据	257
11.3.1	将一列显示为两列	257
11.3.2	将两列显示为一列	259
11.3.3	将数据从一种类型转换为另一种类型	261
11.4	使用模式	263
11.4.1	匹配大写和小写字母	264
11.4.2	当您不知道值时在一定范围内查找数据	265
11.4.3	查找日期数据	267
11.4.4	按时间单位显示数据	270
11.5	避免错误	271
11.5.1	区别 DISTINCT	271
11.5.2	删除副本	274
11.5.3	查找第一项	276
11.6	小结	276



# 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