

国外优质职业教育资源教学用书

SQL应用教程

A Guide to SQL

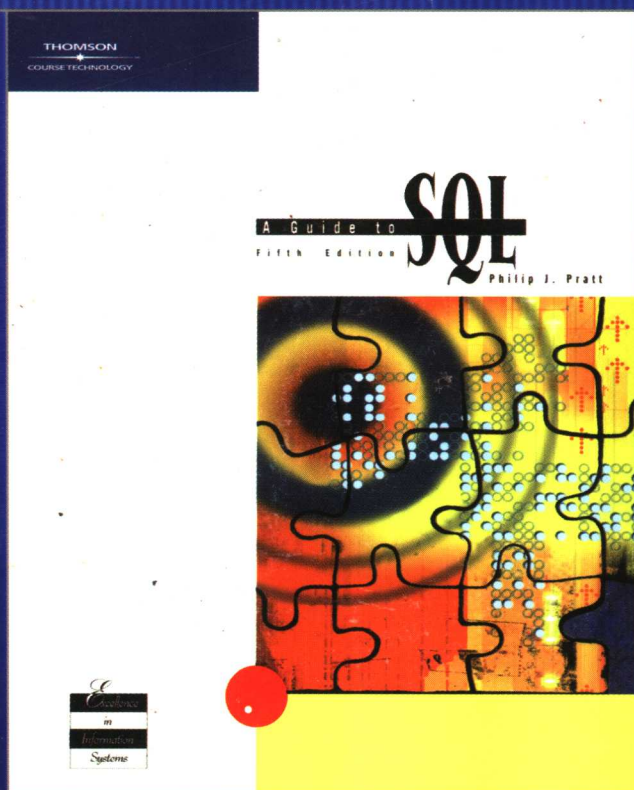
第5版 双语版权：辅助教材

5th Edition

[美] Philip J. Pratt 著
铁 军 等编译



高等教育出版社
Higher Education Press



国外优质职业教育资源教学用书

SQL 应用教程

(第五版 双语版权:辅助教材)

[美] Philip J. Pratt 著

铁 军 等编译

高等教育出版社

内 容 提 要

本书为“国外优质职业教育资源教学用书”的信息系列之一,采用双语版权引进方式出版。

本册是双语版权:辅助教材,本书以案例学习贯穿全书,重点讲解数据库技术中 SQL 的应用。内容包括:数据库管理初步、SQL 初步、单/多表查询、更新数据、数据库管理、报表、嵌入式 SQL 编程等。

本书适用于高等职业教育、高等专科学校教育、成人高等教育、本科院校及举办的二级职业技术学院、继续教育学院和民办高校使用,也可供有关人员学习参考。

图书在版编目(CIP)数据

SQL 应用教程:第 5 版/(美)普拉特(Pratt,P.J.)

著;铁军等编译. — 北京:高等教育出版社,2003.9

书名原文: A Guide to SQL, Fifth Edition

ISBN 7-04-012742-3

I. S... II. ①普... ②铁... III. 关系数据库—数据库管理系统,SQL—高等学校:技术学校—教材
IV. TP311.138

中国版本图书馆 CIP 数据核字(2003)第 061746 号

前 言

SQL(结构化查询语言)是一种流行的计算机语言,被家庭计算机用户、小型商业业主、大型机构的终端用户和程序员等各类群体广泛使用。虽然本书使用 Oracle8 作为教学工具,但书中的各章内容、例子和练习可应用于任何 SQL 系统上。

第五版的《A Guide to SQL》适用的读者包括从选修计算机科学入门课程到攻读高级信息系统课程的各层次学生。本教科书可以用作学习 SQL 课程的单独教材,也可以与数据库基本原理的教科书一起供了解学习 SQL 使用。

本书中的章节是按顺序编排的。为了获得最好的学习效果,学生应该完成每章后面的练习和各章中的例子。由于第 8 章中的内容是假设读者已经学习过至少一种程序设计语言,导师可决定学生是否需要了解这些概念。对于没有编程背景的学生,理解嵌入式 SQL 会存在困难,当学生不适应编程时,导师可以从课本中删除第 8 章。

一、编写特色

1. 使用示例

每章中都包括了若干使用 SQL 解决问题的例子。在每个例子的后面,学生将阅读到用于解决指定问题的命令,然后会看到用来求解的 SQL 命令。对于大多数学生来说,学习这些例子学习是掌握相关知识的最有效途径。正是由于这个原因,教师应该鼓励学生阅读这些章节并在计算机上输入图中给出的命令。

2. 案例研究

每章的所有示例以及每章后的第一套练习都是使用 Premiere Products 数据库作为案例。虽然为了便于管理,该数据库很小,但是关于它的示例和练习都很好地模拟 SQL 命令在实际商务中的应用。在各章和其后的练习中使用同一案例作为示例确保巩固所学内容。

在每章后面的第二套练习中还使用了另一个不同的案例—Henry Books 数据库。该案例为学生提供了没有教材中示例进行直接指导、自己进行探索的机会。

3. 问题与解答部分

在各章中采用一种称为 Q&A(问题和解答)的特殊类型练习。这些练习的目的在于促使学生思考一些特殊的问题并在继续往下学习前理解这些重要的问题。每个 Q&A 的解答都在问题的后面。鼓励学生在学教科书中提供的答案前尝试自己回答这些问题,以保证他们在继续学习之前能理解这些新资料。

4. 练习

在每章中都提供了两套练习,要求学生使用 SQL 命令来解决 Premiere Products 和 Herry Books 数据库里的实际问题。除非特别指出,否则学生都应该在计算机上使用 SQL 命令完成这些练习。

5. 参考附录

教科书后给出了两个参考附录。附录 A 是关于 SQL 的,它描述了主要 SQL 命令的用途和句法。学生可以使用附录 A 来迅速确定什么时候和如何使用这些主要的 SQL 命令。附录 B 给出了一个包括“*How do I*”出处的列表,它让学生通过搜索问题答案的方式交叉引证附录 A 中的资源。

6. 单号练习答案

附录 C 中给出本书中单号练习的答案,让学生有办法检查他们所完成的章末练习。

7. 教师支持

在第 5 版《A Guide to SQL》的教师手册包括详细的教学信息、每章末双号练习的答案以及测试题(和答案)。还提供包括教科书中大多数插图的幻灯片的光盘。

二、教材内容的组织

第 1 章:数据库管理初步

以 Premiere Products 和 Henry Books 数据库为例介绍数据库和数据库管理系统的概念。这一章提供的许多 Q&A 练习确保学生在开始上机前明白如何在纸上操作数据库。

第 2 章:SQL 简介

在本章中学生将学习与关系数据库相关的重要概念和术语。他们将建立并运行 SQL 命令来创建表,使用数据类型,并添加行到表中。第 2 章还讨论空值的作用及其用法。

第 3 章:单表查询

介绍使用 SQL 命令查询数据库。本章中的所有查询都是单表查询。本章还讨论下列内容:简单条件,复合条件,计算列, BETWEEN、LIKE 和 IN 等 SQL 运算符,SQL 函数的应用,嵌套查询,分组数据和检索有空值的列。

第 4 章:多表查询

通过演示连接多个表的查询来完成查询数据库的讨论。本章中讨论 IN 和 EXISTS 等 SQL 运算符、嵌套子查询、使用别名、表的自连接、SQL 集合运算、ALL 与 ANY 运算符的使用。

第 5 章:更新数据

在本章中,学生将学习如何使用 COMMIT、ROLLBACK、UPDATE、INSERT 和 DELETE 等 SQL 命令来更新表的数据。学生也将学习如何用现有的表来创建一个新表,以及如何改变一个表的结构。

第 6 章:数据库管理

讲述 SQL 的数据库管理特性,包括:使用视图,授权和撤销用户的数据库权限,创建、删除和使用索引,使用和从系统目录中获取信息,使用完整性约束来控制数据的输入。

第 7 章:报表

讲授基于表或视图中的数据来创建一个基本或复杂的报表。学生将学习如何连接数据、为报表建立视图、改变列标题与格式、添加报表标题等。学生还将学习计算报表的总计和小计以及分组数据,并讨论脚本和假脱机。

第 8 章：嵌入式 SQL

本章供那些已经完成至少一门编程课程的学生选学。主要讲述嵌入到诸如 COBOL 等过程化语言的 SQL 命令。虽然该章中以 COBOL 作为载体来说明概念，但这些资料同样适用于支持嵌入编程的任何语言。本章中还讨论使用嵌入式 SQL 来插入新行、更改和删除现有行，并讨论如何使用嵌入式 SQL 命令来进行单行检索和如何使用游标进行多行检索。

附录 A：SQL 参考

包括主要的 SQL 子句命令和运算符。当学生在构建 SQL 命令时可以使用附录 A 作为速查资源。每一个命令都包括简短的描述、一个说明所需要和可选择的子句及运算符的表，以及一个例子及其结果。

附录 B：“我怎样做”参考

给学生提供一个询问诸如“我如何删除行”等问题的机会，以使他们能够从附录 A 中找到正确内容来获得答案。当学生知道他们希望完成的目标但却忘记了所需的正确的 SQL 命令时，附录 B 是特别有用的。

附录 C：单号练习答案

在该附录给出所有章节中单号练习的答案，使学生可以确信他们正确地完成了这些练习。

三、学生注意事项

1. 课内问题

每章中都包含有一些特殊问题，以帮助学生理解重要的概念。有时这些问题的目的是确保学生在继续学习之前掌握一些特别重要的内容，在其他情况下，这些问题是为了给学生提供思考一些超出当前介绍内容的特殊概念。在所有情况下，在问题后面都立即给出其答案。学生可以简单地阅读这些问题和它们的答案，但是学生在继续阅读之前先尝试回答这些问题，然后再用课本中给出的解答来检查自己的答案时，就会获得更大的益处。

2. 章末材料

章末材料包括总结和练习。在总结中简要地描述了全章的内容。浏览总结可以确信所有这些概念都是已经熟悉掌握的。在总结后是两套练习：第一套练习使用的是该章中例子使用的同一个 Premiere Products 数据库，该套练习让学生掌握使用该章中给出的命令；完成第二套使用 Henry Books 数据库的练习，使学生能够将该章所学的知识应用到不熟悉的数据库中。（在附录 C 中给出了这两套练习中单号习题的答案，学生可以进行自我检查。）

致谢

我要感谢各位为准备该书做出贡献的人。我感谢以下每位审阅本书原稿并提出很多帮助意见的人的努力：University College of Cape Breton 的 Danny Yakimchuk, Grand Valley State University 的 Paul Leidig, Purdue University Calumet 的 Misty Vermaat, Old Dominion University 的 Lorna Bowen St. George, 以及 Santa Barbara Community College 的 George Federman。

Course Technology 出版公司的管理编辑(Managing Editor)Jennifer Normandin 和制作编辑(Production Editor)Anne Valsangiacomo 所做的巨大努力使本书的出版成为可能。

与本书的开发编辑(Development Editor)Jessica Evans 工作总是非常愉快。谢谢您的努力。您永远是最棒的！

编译者附话

这是一本指导学生学习 SQL 语言的教材。全书用一个数据库案例——Premiere Products 为线索贯穿始终，从创建该数据库的需求背景逐层展开，介绍了关系数据库、表、查询、数据更新、视图、用户权限控制、数据完整性等基本概念和实现它们所使用的 SQL 命令，并介绍了使用 SQL 实现报表设计和嵌入式 SQL。

本书不是按传统的叙述方式来介绍 SQL 这一关系数据库的标准语言，而是通过大量示例先提出任务目标，然后用简洁的语言说明完成任务需要使用的 SQL 命令，最后以插图给出正确的 SQL 语句及运行结果，并在插图中辅以直观的解释，对一些难点、重要概念和容易混淆的地方用注释和问题/解答的方法来进行说明。作者以十分细腻的方式来划分知识点，使学生能够循序渐进地掌握 SQL 的主要命令、基本句法和使用方法。在每章后面，作者提供了两套练习，第一套练习与书中示例使用同一个数据库，用于巩固学生所学内容；作者提供的第二套练习是面向没有实例指导的 Henry Books 数据库，以培养学生应用所学 SQL 命令来解决实际问题的能力，可谓独具匠心。通过这种方式，作者没有将 SQL 强大功能的展示变得冗长、乏味，而是生动直观，浅显易懂，使读者在阅读和按照示例进行练习、并完成各章的习题后，能够轻松地掌握 SQL 的应用，从而达到教学目标。

我们在接受高等教育出版社高职高专分社要求将本书进行适当编译以作为学习 SQL 的双语教材的任务后，系统地浏览了全书并为该书的写作方式所吸引。在考虑如何进行编译时，我们感到该书特点是：理论概念介绍简洁实用，以上机操作为主要学习手段。为了不淹没该书的特点，决定采取部分翻译的方式来完成任務。

在进行翻译时，我们按如下的原则来进行。

1. 避免翻译重复内容。例如，书中各章的教学目标、前言和总结部分都是对该章内容的概述，我们只翻译其中的教学目标。

2. 书中的插图和表格内容不翻译。这些内容比较直观，语句严谨，容易阅读和理解。读者在阅读译文时可以直接在原文中查找对应的图表。

3. 每章末的练习部分不翻译。作为双语教材，要求学生在掌握专业知识的同时还应该掌握必要的专业外语，将练习部分保留原文是促使学生理解原文的有效手段。

4. 考虑到第 8 章中介绍的 COBOL 语言在国内很少使用和接触到，并且嵌入式 SQL 属于较深的内容，作者也将其作为可选内容来推荐，因此没有翻译该章。

根据我们的教学经验，该书可以作为学生在学习完数据库原理后选修 SQL 及相关课程的教材，也可以作为数据库课程的上机指导书来使用，同时还可以供相关技术人员作为学习

SQL 的参考书。

鉴于国内学生较少接触和使用作者应用的 Oracle 数据库,译者推荐使用 SQL Server 7/2000 作为上机环境,使用其提供的查询分析器能够完成除第 7 章报表之外的所有上机内容。但是,读者在使用 SQL Server 时,应注意 SQL Server 所用 T-SQL 本身的一些特点,例如多表之间连接可以在 FROM 子句中使用关键字 JOIN 来完成,意义更为直观并使 WHERE 子句简练。

全书各部分的翻译如下:古梅负责第 1 章;肖彬和付瑞萍共同翻译第 2 章;铁军负责第 3 章和前言部分;付瑞萍负责第 4 章;周严平负责第 5 章;杜春涛负责第 6 章和第 7 章。最后由铁军统稿和组织整理。

在翻译过程中,我们虽然力求准确和风格一致,尽量保持原文的特色,但限于学识和水平,肯定存在疏漏、错误之处,请读者在使用过程中给予指正。我们的联系地址是 tiejun@ncut.edu.cn。

译者

2003 年 4 月

目 录

第 1 章 数据库管理初步	1	空值	21
学习目标	1	总结	22
Premiere Products 数据库	1	练习	22
Henry Book 数据库	3		
总结	5	第 4 章 多表查询	23
练习	5	学习目标	23
第 2 章 SQL 简介	6	引言(略)	
学习目标	6	查询多表	23
引言(略)		使用别名	27
关系数据库	6	更复杂的连接	27
数据库的建立	8	集合运算	29
SQL 命令	9	ALL 和 ANY	30
删除表	10	总结	30
数据类型	10	练习	30
空值	10		
向表中添加数据	11	第 5 章 更新数据	31
查看表中数据	12	学习目标	31
修改数据库中的错误	12	引言(略)	
其他表	12	提交和回退	31
描述一个数据表	13	改变表中现有数据	31
总结	13	向表中添加新行	32
练习	13	从表中删除行	32
		用现有表创建新表	33
第 3 章 单表查询	14	改变列的值为空	33
学习目标	14	改变表的结构	34
引言(略)		总结	35
简单查询	14	练习	35
排序	17		
使用函数	18	第 6 章 数据库管理	36
嵌套查询	19	学习目标	36
分组	20	引言(略)	
		视图	36

安全	41	练习	54
索引	43		
系统目录	45	第 8 章 嵌入式 SQL 编程	55
SQL 中的完整性	46	学习目标(略)	
总结	47	引言(略)	
练习	47	数据分区(略)	
第 7 章 报表	48	程序分区(略)	
学习目标	48	产生多行的 SELECT 语句(略)	
引言(略)		更新游标(略)	
连接列	48	错误句柄(略)	
创建和使用脚本	49	总结(略)	
为报表创建视图	49	练习(略)	
执行 SELECT 命令	50		
改变列标题	50	附录 A SQL 参考(略)	
改变报表中列的格式	51	附录 B “我怎样做” 参考(略)	
添加报表标题	51	附录 C 单号练习答案(略)	
分组报表数据	52	索引(略)	
在报表中包括总计和小计	52		
发送报表为文件	53		
产生报表的完整脚本	53		
总结	54		

第 1 章

数据库管理初步

学习目标

- 理解 Premiere Products 数据库，该数据库针对用具、家用器皿和体育用品的批发商。
- 理解 Henry Books 数据库，该数据库针对连锁书店。

■ Premiere Products 数据库

Premiere Products 数据库的管理者是一家用具、家用器皿和体育用品的批发商，他认为公司近期业务的增长使得很难再灵活地使用人工系统来维护客户、订单和存货清单数据。只有将这些数据存放在安装了功能全面的数据库管理系统的计算机里，才能确保数据是实时的，并且比当前使用的人工系统更为精确。管理者还将能产生一系列有用的报表。另外，管理者希望能够对数据库中的数据进行查询，方便快速地获得结果。

在决定哪些数据必须存储在数据库里的过程中，管理者确定 Premiere Products 必须维护下列关于销售代表、客户以及分类存货清单的信息。

1. 编号、姓氏、名字、地址、全部佣金、每个销售代表的代理费比率；
 2. 客户编号、姓氏、名字、地址、当前余额、每个客户的信用限额以及负责该客户的销售代表的编号；
 3. 物品编号、物品描述、现存数量、种类、存储此类物品的仓库编号、每种物品的单价；
- Premiere Products 数据库也必须保存订单信息。图 1.1 显示了一个示例订单。

示例订单有三个组成部分：

1. 订单头(订单顶部)。包括订单编号；订货日期；客户编号、姓名、地址；以及销售代表的编号和姓名。
2. 订单主体包括许多订单明细，有时也称为订购项目。每个订单明细都包含物品编号，物品描述，此物品的订购数量，物品的报价。每个订单明细还包含一个总计，通常叫做“扩展项”，它是订货数量和物品报价的乘积。

3. 最后, 订单合计(订单底部)是所有订单明细的总和。

Premiere Products 数据库也必须为每个客户订单保存下列项目:

1. 对于每张订单有: 订单编号, 订货日期, 下订单的客户的编号。客户的姓名、地址以及负责此客户的销售代表的编号被存储在客户信息里。销售代表的姓名存储在销售代表信息里。
2. 对于每个订购项目有: 订单编号、物品编号、订货数量、报价。物品描述被存储在物品信息里。订货数量和报价的乘积没有被保存, 因为它可以在需要的时候由计算得出。
3. 总的订单合计并未被作为数据库的一部分保存下来。而是在订单被打印或显示在屏幕上时计算得出。

图 1.2 显示了 Premiere Products 数据库的一些示例数据。

这里有 3 个销售代表(SALES_REP), 他们用编号 03、06 和 12 来识别。编号为 03 的销售代表的姓名是 Mary Jones, 住在 Grant, MI, 街道地址是 123 Main, 邮政编码为 49219。她的总代理费是 2 150 美元, 代理费比率是 5%(0.05)。

Premiere Products 数据库里有 10 个客户(CUSTOMER), 这些客户用编号 124、256、311、315、405、412、522、567、587 及 622 来标识。编号为 124 的客户姓名是 Sally Adams, 住在 Lansing, MI, 街道地址是 481 Oak, 邮政编码为 49224。她的当前余额是\$818.75, 信用限额是\$1 000。SLSREP_NUMBER 栏里的 03 表示编号为 03 的销售代表负责客户 Sally 的业务。

接下来是物品表(PART), 表里有 10 种物品, 这些物品用编号 AX12、AZ52、BA74、BH22、BT04、BZ66、CA14、CB03、CX11 及 CZ81 来标识。AX12 代表熨斗, 这种熨斗现存 104 个。熨斗属于 HW(家用器皿)类, 存放在 3 号仓库, 单价是 24.95 美元。其他的类别还有 AP(用具)和 SG(体育用品)。

再回到订单表(ORDERS), 这个表里有 7 个订单, 这些订单用编号 12489、12491、12494、12495、12498、12500 及 12504 来标识。编号为 12489 的订单是由 124 号客户(Sally Adams)在 2002 年 9 月 2 号产生的。

ORDER_LINE 订单明细表(ORDER_LINE)乍一看可能有点奇怪。为什么需要一个单独的表来存放订单明细呢? 难道不可以把它们包含在 ORDERS 表里吗? 答案是肯定的。ORDERS 表可以被构造如图 1.3 所示的样式。注意这张表里包含有同图 1.2 中所示客户编号、订货日期都相同的重复的订单。另外, 图 1.3 中表里的每一行涵盖了订单的全部订单明细。举个例子, 验证一下第 5 行, 可以看到订单 12498 有两条订单明细。一条是以单价 12.95 美元购买 2 个编号为 AZ52 的物品, 另一条是以单价 24.95 美元购买 4 个编号为 BA74 的物品。

【问题:】图 1.3 里相同的信息在图 1.2 里是如何表示的?

【解答:】查看图 1.2 显示的 ORDER_LINE 表, 注意第 6 和 7 行。第 6 行显示的是 12498 号定单的一条订单明细, 内容是以单价 12.95 美元购买 2 个编号为 AZ52 的物品。第 7 行显示的也是 12498 号订单的一条订单明细, 内容是以单价 24.95 美元购买 4 个编号为 BA74 的物品。这样, 图 1.3 里相同的信息在图 1.2 里就被用独立的两行来显示。

相同的信息能用一行显示就不用两行, 这看起来似乎更有效率, 但是却带来一个问题, 如图 1.3 所示, 表格的结构更复杂。图 1.2 所示的表, 每一个单元只有一个项目。图 1.3 所示的表, 有几条单元却包含了多个项目, 这就导致难以在不同列之间检索信息。举个例子, 在订单

编号为 12498 的那一行，至关重要是要清楚 AZ52 与 NUMBER_ORDERED 列里的 2 相匹配(而不是 4)，同样的 AZ52 与 QUOTED_PRICE 列里的 12.95 美元相匹配(而不是 24.95 美元)。另外，一个复杂的表意味着许多令人担忧的实际问题，例如：

1. 给这些多个项目多大的空间？
2. 如果一个订单有多个订单明细，给它多少空间？
3. 对指定的物品，怎样确定哪个订单包含此物品的订单明细？

当然，这些问题都是可以解决的，只是增加了图 1.2 里的表结构所没有的复杂度。在图 1.2 所示的表结构里，没有多重明细需要担心，因此就是每个订单都有大量的订单明细也没什么关系，并且在大量的订单明细中查找某种指定物品也是很容易的(只需用给定的物品编号在所有订单明细的 PART_NUMBER 列搜索即可)。一般来说，这种简单的结构是更可取的，这也是为什么把订单明细放在一张独立的表里的原因。

为了测试对 Premiere Products 数据的理解，请回答下列基于图 1.2 里数据的问题。

【问题：】 Mary Jones 负责代理的客户的编号是什么？

【解答：】 124, 412 和 622(先在 SALES_REP 表里查找到 Mary Jones 的编号 03,然后再在 CUSTOMER 表里找到 SLSREP_NUMBER 列为 03 的所有客户)。

【问题：】 产生 12491 号订单的客户的姓名是什么？负责此客户的销售代表的姓名又是什么？

【解答：】 客户叫 Don Charles, 销售代表叫 Miguel Diaz。(先在 ORDERS 表里查找到客户的编号 311,然后再在 CUSTOMER 表里找到编号为 311 的客户名,再用负责此客户的销售代表的编号 12,就可以在 SALES_REP 表里找到销售代表的姓名。)

【问题：】 12491 号订单里都列出了哪些物品？对每一个物品，给出具体描述、订货数量以及报价。

【解答：】 物品编号：BT04；物品描述：煤气烤架；订购数量：1；报价：\$149.99。物品编号：BZ66；物品描述：洗衣机；订购数量：1；报价：\$399.99。(在 ORDER_LINE 表里查找到订单编号为 12491 的所有行，每一行都包含物品编号、订购数量和报价。用物品编号就可以在 PART 表里找到与其匹配的物品描述。)

【问题：】 为什么 QUOTED_PRICE 列会成为 ORDER_LINE 表的一部分？能否只保留物品编号而通过 PART 表来查找价格？

【解答：】 如果 ORDER_LINE 表里没有 QUOTED_PRICE 列，那就需要通过查找 PART 表来获得订单明细里某个物品的价格。尽管这可能并不坏，但却阻碍了 Premiere Products 针对不同客户改变同一物品价格的能力。正是因为 Premiere Products 想要一定的报价弹性并能根据不同客户变更不同价格，所以才把 QUOTED_PRICE 列包含在 ORDER_LINE 表里。如果检查 ORDER_LINE 表，就会发现报价与 PART 表里的实际价格相一致的例子，同时也会发现不一致的情况。

■ Henry Books 数据库

Henry 是一家名叫 Henry Books 的连锁书店的所有者，他与 Premiere Products 数据库的

管理者一样，认为现在是用计算机管理工作的时候了。同 Premiere Products 一样，他也计划将数据存储在数据库里，并希望得到同样的益处：即确保数据的及时和准确，还希望能生成几种重要的报表。另外，他还希望能够对数据库中的数据进行查询，并能简便快速地获得答案。

为了运行这个连锁书店，Henry 需要收集并组织有关出版社、作者和书刊的信息。每本书都有一个惟一的识别码。另外，书店还要记录书名、出版社、图书类型、价格以及是否是平装本。同时，书店也记录着作者以及连锁书店各分店的存书数量。

Henry 将把这些数据存储在一系列表里。图 1.4、1.5 和 1.6 显示了 Henry Books 数据库的一些示例数据。

【问题：】为了检查你对出版社与图书之间关系的理解，请回答下列问题：谁出版了 Knockdown？哪本书是由 Signet 出版的？

【解答：】Pocket Books 出版了 Knockdown；Cujo、Organ、Carrie 和 Magritte 是由 Signet 出版的。在图书表(BOOK)里(图 1.5)，Knockdown 的出版社的代码是 PB。再检查出版社表(PUBLISHER)(图 1.4)会发现 PB 是分配给 Pocket Books 的代码。

要找到 Signet 出版的图书，先要在 PUBLISHER 表里找到它的代码 SI。然后在 BOOK 表里找到出版社代码为 SI 的所有记录，就会发现 Signet 出版了 Cujo、Organ、Carrie 和 Magritte。

WROTE 表(图 1.6)将图书和著作者联系起来。次序号表示的是某本书应署名作者的先后顺序。INVENT 表(图 1.6)表示的是某本书当前在某分店的存货数量。举例来说，第 1 行表示代码为 0180 的图书当前在代码为 1 的分店里存有 2 本。

【问题：】为了检查对作者与图书之间关系的理解，请回答下列问题：谁写了 Organ(以正确的顺序列出作者)？哪本书是由 Jeffrey Archer 写的？

【解答：】Peter Williams 和 Barbara Owen 写了 Organ；Kane and Abel、Prodigal Daughter 和 First Among Equals 是由 Jeffrey Archer 写的。要确定谁写了 Organ，先要在 BOOK 表里找到此书的图书代码 5163。然后，再在 WROTE 表里查找图书代码(BOOK_CODE 列)为 5163 的所有行，会发现有两条这样的行。其中一行的作者编号(AUTHOR_NUMBER 列)是 12，另一条是 13。接下来要做的就是到 AUTHOR 表里找到编号为 12 和 13 的作者，就可以得到 Barbara Owen(12)和 Peter Williams(13)。编号为 12 的作者的次序号是 2，而编号为 13 的作者的次序号却是 1。因此，以正确顺序列出作者的结果就是 Peter Williams 和 Barbara Owen。要找到 Jeffrey Archer 写的书，先要在 AUTHOR 表里的 AUTHOR_NUMBER 列中找到他的作者编号 01。然后在 WROTE 表里查找作者编号为 01 的所有行，会发现有 3 行，其图书代码分别是 0189、2766 和 3743。在 BOOK 表里查找这些代码，就会找到 Jeffrey Archer 所写的 Kane and Abel、Prodigal Daughter 和 First Among Equals 这 3 本书。

【问题：】有一位顾客在 1 号分店希望买一本 Vortex，请问 1 号分店现在有这本书吗？

【解答：】没有。在 BOOK 表里可以找到 Vortex 的图书代码是 3906。要查明 1 号分店有多少存书，就是要在 INVENT 表里查找 BOOK_CODE 列为 3906 且 BRANCH_NUMBER 列为 1 的行。因为没有找到这样的行，所以 1 号分店没有 Vortex 的存书。

【问题：】你可能想要给此客户提供 Vortex 这本。那么哪个分店有此书的存货？每个分店又有几本存书呢？

【解答:】 2 号分店有一本, 3 号分店有两本。已经知道 Vortex 的代码是 3906(如果不知道, 可以在 BOOK 表里找到)。要找出哪几个分店现有此存书, 就是要在 INVENT 表里找到 BOOK_CODE 列为 3906 的行, 可以找到 2 个这样的行。第一行表明 2 号分店现有一本存书, 第二行表明 3 号分店现有两本存书。

■ 总结

(略)

■ 练习

(略)

第 2 章

SQL 简介

学习目标

- 理解关系数据库的有关概念和术语
- 创建和运行 SQL 命令
- 使用 SQL 创建表
- 识别和使用数据类型定义 SQL 表中的列
- 理解和使用空值
- 为表添加行

■ 关系数据库

正如在第 1 章 Premiere Products 数据库中看到的那样，关系数据库实质上是一个表的集合(见图 2.1)。使用关系数据库就感觉像是在使用一个集合(这仅仅意味着用户怎样建立这些表，而不需要用户知道数据库管理系统在其中起什么作用)。如果数据库是表的集合，为什么不把关系型数据库模型叫做“表”模型或者是其他类似的名称呢？从形式上讲，这些表可以称为关系，关系型数据库因此而得名。

实体、属性和关系

下面的一些名词和概念可以对理解数据库环境非常重要。实体、属性和关系，这 3 个概念对探讨数据库非常重要。实体像一个名词，如一个人，一个地方，一个东西或是一件事情。例如，对于 Premiere Products，实体有：客户、订单和销售代表。对于一个学校，实体有学生、教职员工和班级；房地产公司关心的是客户、房屋和代理商；汽车经销商关心的是汽车、顾客和生产商。

属性是一个实体的特征，这个名词在这的含义与日常英语完全一样。例如对于个人实体来说，有眼睛的颜色、身高等一些属性。对于 Premiere Products 来说，客户实体的属性包括客户的姓氏、名字、地址、所在城市等等。

最后一个关键名词是关系，关系是实体之间的联系。例如，在 Premiere Products 中，客

户和销售代表存在着一些联系。一个销售代表联系着他的所有客户，而一个客户联系着他的销售代表。用术语说：一个销售代表关系着他的所有客户，而一个客户关系着他的销售代表。

这种特殊的关系叫做一对多关系，因为一个销售代表联系着很多客户，但是一个客户仅和一个销售代表相联系(在这种关系中，多这个字用处与日常英语有不同的含义，它并不总是很多的意思。例如在这里，多这个字意味着销售代表可以与任何数量的客户联系，也就是说，一个销售代表可能与一个客户或者是更多客户联系，但也有可能是零个客户)。

关系数据模型中的 DBMS(数据库管理系统)是怎样处理实体、实体属性以及实体之间的关系呢？实体和属性比较简单，每个实体都有自己的表。在 Premiere Products 数据库中，销售代表有张表，客户有张表，如此等等。实体的属性是表中的列，例如在销售代表表中，一列是销售代表的编号，一列是销售代表的名字等等。

关系起什么作用？在 Premiere Products 中，销售代表和客户存在一对多的关系(即每个销售代表都与多个他所联系的客户有关系，而每一个客户总是只和一个联系他的销售代表有关系)。在关系型数据库中，如何实现这种关系呢？答案就是：在两个或者两个以上的表中使用相同的列。再看看图 2.1，在 SLSREP_NUMBER 表中和 CUSTOMER 表中，都有 SLSREP_NUMBER 这一列，用来实现销售代表与客户之间的关系。对于一个给定的销售代表，可以通过这些相同的列去确定他所代表的所有客户；对于一个给定的客户，也可以通过这些相同的列去找到联系他的销售代表。

有了上述这些知识后，就不难发现关系实质上就是一个二维表。然而，如果再看看图 2.1 中的表，就会发现在这些关系中有特定的限制。每一列必须有惟一的名称，每一列的数据要和列的名称相“匹配”。例如，如果一列的名字是 CREDIT_LIMIT，在这列的所有数据必须都是信用限额。还有，每一行都应当是惟一的，如果两行是相同的，第二行就不能提供任何新的信息。表中灵活性最大的是，列和行的次序不重要。尽管有这么多要求，但是表要尽可能简单。为了做到这一点，在一个表中可以限制每一个地方只能有一个数据项，不允许出现多个数据项(或重复组)(见图 2.2)。

定义：关系就是一个二维表，在表中的项目都是单值的(即每个地方只能包含一个数据项)，每列有一个严格的名称(或属性名)，每列的值都具有相同的属性，列和行的次序没有要求，每行都包含惟一的值。

从这个定义中，可以看到关系型数据库是个关系的集合。

注意：在表中行通常称为记录，列通常称为字段。行也称为元组，列也称为属性。

下面有一个被广泛采用来简略表示关系型数据库结构的方法，即可以先写出表的名字，在圆括号中列出表的所有列(字段)。例如，下面就是对 Premiere Products 数据库的描述：

```
SALES_REP(SLSREP_NUMBER, LAST, FIRST, STREET, CITY, STATE, ZIP_CODE,
TOTAL_COMMISSION, COMMISSION_RATE)
CUSTOMER(CUSTOMER_NUMBER, LAST, FIRST, STREET, CITY, STATE, ZIP_CODE,
BALANCE, CREDIT_LIMIT, SLSREP_NUMBER)
ORDERS(ORDER_NUMBER, ORDER_DATE, CUSTOMER_NUMBER)
ORDER_LINE(ORDER_NUMBER, PART_NUMBER, NUMBER_ORDERED, QUOTED_PRICE)
```