



Beginning SQL

SQL 入门经典

(美) Paul Wilton 著
John Colby 译
敖富江

著
译



清华大学出版社

SQL 入门经典

(美) Paul Wilton 著
John Colby
敖富江 译

清华大学出版社

北 京

Paul Wilton John Colby

Beginning SQL

EISBN: 0-7645-7732-8

Copyright © 2005 by John Wiley & Sons, Inc.

All rights reserved. Authorized translation from the English language edition published by John Wiley & Sons, Inc.

本书中文简体字版由 John Wiley & Sons, Inc. 授权清华大学出版社出版。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。

北京市版权局著作权合同登记号 图字: 01-2005-4449

版权所有, 翻印必究。举报电话: 010-62782989 13501256678 13801310933

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

本书防伪标签采用特殊防伪技术, 用户可通过在图案表面涂抹清水, 图案消失, 水干后图案复现; 或将表面膜揭下, 放在白纸上用彩笔涂抹, 图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

SQL 入门经典/(美)威尔顿(Wilton, P.), (美)科尔比(Colby, J.)著; 敖富江译. —北京: 清华大学出版社, 2006.6

书名原文: Beginning SQL

ISBN 7-302-12833-2

I. S… II. ①威…②科…③敖… III. 关系数据库—数据库管理系统 IV. TP311.138

中国版本图书馆 CIP 数据核字(2006)第 033222 号

出版者: 清华大学出版社 地 址: 北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编: 100084

社总机: 010-62770175 客户服务: 010-62776969

组稿编辑: 曹 康

文稿编辑: 李 阳

封面设计: 孔祥丰

版式设计: 孔祥丰

印刷者: 清华大学印刷厂

装订者: 三河市金元印装有限公司

发行者: 新华书店总店北京发行所

开 本: 185 × 260 印张: 26.25 字数: 672 千字

版 次: 2006 年 6 月第 1 版 2006 年 6 月第 1 次印刷

书 号: ISBN 7-302-12833-2/TP · 8166

印 数: 1 ~ 4000

定 价: 48.00 元

前言

数据，数据，数据！只要有计算机的地方，就会有数据的存在，无论是处理百万次的计算，还是为您的阿姨 Maude 的生日保存一个记录。当需要存储数据时，就要用到数据库。在我 8 年的职业编程生涯中，我所接触的每一个项目都与数据库有关——大多数商业应用程序和项目都需要使用数据库。当然也得承认，在某些领域，例如计算机游戏，对数据库使用得并不是很多。我能够猜测，“Mega Doom 99: The Final Bloody Massacre”在后台中没有使用 Oracle 数据库！

但是，我得承认，大约 10 年前，当我开始接触数据库时，我发现它们非常令人迷惑。那时我已经在几年的空闲时间中练习了编程，并习惯于使用文本文件存储信息。我决定改变一下这种方式，开始创建数据库并编写 SQL 语句，并且得到了非常混乱和奇怪的结果。数据库及其设计和基本概念与在单个文件中存储数据的差别很大，并且用于在数据库中访问和操作数据的结构化查询语言(SQL)与过程化语言的差别也很大。本书的目标之一是介绍一些新概念和一些完成操作的方式。这之后，解释了开始学习数据库和 SQL 编程之前所需要的所有基本概念和理论。本书将详细地介绍如何从数据库中获得想要的数据和所有可能得到的结果，因为如果不事先介绍这些，SQL 可能会产生令您非常吃惊的结果。

本书的另外一个目标是让您能够快速高效地在实际环境中设计数据库并编写 SQL 代码。就个人而言，我并不喜欢覆盖所有细节的书籍，否则几个月之后您可能才可以开始进行自己的工作。我坚持所介绍的内容能够用于大多数数据库应用程序中，并将更详细和更高级的内容放在后面的章节中。前面几章的目标是让您能够快速地了解并运行 SQL，同时也讨论了一些核心的概念、代码和技巧，并利用大量的实际示例加深对它们的了解。

最后，我是一个实用且实际的人，所以，我所喜欢阅读的计算机书籍中并不会包含很多理论。本书反映了我的“将其付诸于行动”的宗旨，包含多个示例，能够使您获得大量的实用经验。书中也介绍并解释了理解基础内容所需要的理论，但最终目标是将理论付诸于实践。我在编程的大多数日子中，都用到了数据库和 SQL 语言，我希望能够将这些实际经验带到本书中。

本书读者对象

本书首先介绍数据库和 SQL 语言的基础知识。学习本书不需要预备的数据库或者 SQL 知识，因为本书覆盖了从数据库设计到创建您的第一个数据库和理解 SQL 语言是如何用于数据库等方面的所有内容。

如果您已经具有关于数据库和 SQL 方面的经验，则将有一个很好的起点，并且您可能想跳过第 1 章。但第 1 章中介绍了如何创建本书的示例数据库，这个数据库将用于本书中所有的示例。

本书主要内容

本书将介绍结构化查询语言，或者简称为 SQL。在数据库中可以使用 SQL 创建数据库、插入并提取数据。因此，有必要了解数据库系统中的理论和概念。因此，本书也涵盖了数据库理论和数据库设计，这样使得您能够创建一个高效的数据库。

本书中的 SQL 代码反映了诸如美国国家标准化组织(ANSI)和国际标准化组织(ISO)等标准化组织所制定的现代 SQL 标准集。但是，虽然标准是伟大的，但在实际使用中又是另外一种情形。因此，本书集中于介绍大多数现代数据库系统所支持的 SQL 类型。您应当能够发现，大多数代码能够经过较少修改或不用修改就可以运行于最近 6 或 7 年之内发布的大多数数据库系统中。

本书结构

本书分为两个主要的部分。第 I 部分由第 1~3 章组成，提供了用于理解数据库和 SQL 语言的基础知识。第 I 部分的目标是加速您了解各种必要的细节，涵盖了以下内容：

- 必要的数据库理论
- 编写 SQL 代码
- 优秀的数据库设计
- 创建数据库
- 使用 SQL 语句输入、更新和删除数据
- 使用 SQL 语句提取数据——更特别的是，介绍了如何利用数据库回答真实情形中的问题

阅读完第 3 章后，您将可以创建自己的数据库并编写自己的 SQL 代码。在进入本书的第 II 部分之前，您可能希望创建一些自己的数据库。

本书的第 II 部分从第 4 章开始，一直到结束，介绍了更多的细节并阐述了一些更高级的主题。它的目标是为 SQL 编程的许多方面提供非常宽广和完整的基础。主要包括以下一些主题：

- 高级数据库设计，介绍标准化方面的理论和实际应用，以及如何提高数据库的效率和可靠性
- 利用 SQL 的内嵌数据操作和计算函数来使用和操作数据
- 从多个不同表中选择数据
- 数据库的安全
- 数据库的优化

本书还包括 3 个附录。附录 A 包含了每一章中练习的答案，因此在您回答完问题之前，不要参阅它们。附录 B 介绍了如何下载、安装和使用本书所用到的 5 种数据库系统。附录 C 包含了向示例数据库中添加初始数据的 SQL 语句，如果不想亲自输入它们，可以从 Web 站点 www.wrox.com 和 www.tupwk.com.cn/downpage 下载它们。

使用本书的要求

为了充分利用本书并运行其中的示例，您需要通过一个数据库系统来进行实践。本书的代码已经在下面的 5 种常用数据库系统中测试过：

- MySQL
- Microsoft SQL Server
- IBM DB2
- Microsoft Access
- Oracle 10g

您几乎能够从网上免费下载所有这些数据库，可以下载它们的完全版本或测试版本。在附录 B 中，您将了解从什么地方下载它们，如何安装它们，以及如何使用它们。

如果您使用的是一种不同的数据库系统，也不会存在什么问题，因为我尽可能地避免使用与数据库系统相关的代码，尽量使用被大多数数据库系统支持的 SQL 语句。在所有的数据库系统中使用相同的代码是不可能的，我在书中列举出了一些针对不同数据库的不同语法。您将很可能发现，其中一种语法的变种能够运行于您的系统中，但可能需要进行一些小小的修改。

源代码

在读者学习本书中的示例时，可以手工输入所有的代码，也可以使用本书附带的源代码文件。本书使用的所有源代码都可以从本书 Web 站点 <http://www.wrox.com/> 或 www.tupwk.com.cn/downpage 上下载。登录到站点 <http://www.wrox.com/>，使用 Search 工具或使用书名列表就可以找到本书。接着单击本书细目页面上的 Download Code 链接，就可以获得所有的源代码。

注释：

由于许多图书的标题都很类似，所以按 ISDN 搜索是最简单的，本书英文版的 ISBN 是 0-7645-7732-8。

在下载了代码后，只需用自己喜欢的解压缩软件对它进行解压缩即可。另外，也可以进入 <http://www.wrox.com/dynamic/books/download.aspx> 上的 Wrox 代码下载主页，查看本书和其他 Wrox 图书的所有代码。

勘误表

尽管我们已经尽了各种努力来保证文章或代码中不出现错误，但是错误总是难免的，如果您在本书中找到了错误，例如拼写错误或代码错误，请告诉我们，我们将非常感激。通过勘误表，可以让其他读者避免受挫，当然，这还有助于提供更高质量的信息。

请给 fwkbook@tup.tsinghua.edu.cn 发电子邮件，我们就会检查您的信息，如果是正确的，就把它传送到该书的勘误表页面上，或在本书的后续版本中采用。

要在网站上找到本书的勘误表，可以登录 <http://www.wrox.com>，通过 Search 工具或书名列

表查找本书,然后在本书的细目页面上,单击 **Book Errata** 链接。在这个页面上可以查看到 Wrox 编辑已提交和粘贴的所有勘误项。完整的图书列表还包括每本书的勘误表,网址是 www.wrox.com/misc-pages/booklist.shtml。

P2P.WROX.COM

要与作者和同行讨论,请加入 p2p.wrox.com 上的 P2P 论坛。这个论坛是一个基于 Web 的系统,便于您张贴与 Wrox 图书相关的消息和相关技术,与其他读者和技术用户交流心得。该论坛提供了订阅功能,当论坛上有新的消息时,它可以给您传送感兴趣的论题。Wrox 作者、编辑和其他业界专家和读者都会到这个论坛上来探讨问题。

在 <http://p2p.wrox.com> 上,有许多不同的论坛,它们不仅有助于阅读本书,还有助于开发自己的应用程序。要加入论坛,可以遵循下面的步骤:

- (1) 进入 p2p.wrox.com, 单击 **Register** 链接。
- (2) 阅读使用协议,并单击 **Agree**。
- (3) 填写加入该论坛所需要的信息和自己希望提供的其他信息,单击 **Submit**。
- (4) 您会收到一封电子邮件,其中的信息描述了如何验证账户,完成加入过程。

注释:

不加入 P2P 也可以阅读论坛上的消息,但要张贴自己的消息,就必须加入该论坛。

加入论坛后,就可以张贴新消息,响应其他用户张贴的消息。可以随时在 Web 上阅读消息。如果要让该网站给自己发送特定论坛中的消息,可以单击论坛列表中该论坛名旁边的 **Subscribe to this Forum** 图标。

关于使用 Wrox P2P 的更多信息,可阅读 P2P FAQ,了解论坛软件的工作情况以及 P2P 和 Wrox 图书的许多常见问题。要阅读 FAQ,可以在任意 P2P 页面上单击 **FAQ** 链接。

目 录

第 1 章 SQL 简介	1
1.1 数据库简史	1
1.2 结构化查询语言	5
1.2.1 SQL 查询简介	5
1.2.2 SQL 与其他编程语言的比较	5
1.2.3 理解 SQL 标准	6
1.3 创建数据库	7
1.4 创建、更改和删除表	16
1.4.1 创建表	17
1.4.2 更改现有表	18
1.4.3 删除现有表	18
1.5 优秀的数据库设计	19
1.5.1 获得并分析数据需求	19
1.5.2 逻辑地划分数据	20
1.5.3 选择正确的数据类型	22
1.5.4 使用主键	24
1.6 创建示例数据库	25
1.7 本章小结	29
1.8 练习	30
第 2 章 输入信息	31
2.1 插入新数据	31
2.2 将数据插入到案例分析数据库	34
2.3 更新数据	35
2.4 删除数据	39
2.5 本章小结	39
2.6 练习	40
第 3 章 提取信息	41
3.1 SELECT 语句	41
3.1.1 仅返回不同的行	42
3.1.2 使用别名	43

3.1.3	利用 WHERE 子句过滤结果	44
3.2	逻辑运算符和运算符优先级	49
3.2.1	运算符优先级概述	49
3.2.2	使用逻辑运算符	52
3.3	利用 ORDER BY 排序结果	62
3.4	从多个表中选择数据	75
3.4.1	在 MS Access 中使用圆括号括起内部联合	84
3.4.2	SQL 是基于集合的	86
3.5	NULL 数据概述	95
3.6	本章小结	97
3.7	练习	97
第 4 章	数据库的高级设计	98
4.1	标准化	98
4.1.1	第一范式	98
4.1.2	第二范式	100
4.1.3	第三范式	101
4.2	利用约束确保数据的有效性	103
4.2.1	NOT NULL 约束	104
4.2.2	UNIQUE 约束	105
4.2.3	CHECK 约束	109
4.2.4	主键和 PRIMARY KEY 约束	111
4.2.5	外键	113
4.3	利用索引加速结果查询	118
4.3.1	改进 Film Club 数据库的设计	121
4.3.2	提高数据的有效性和效率	123
4.3.3	设计更好的数据库的一些技巧	131
4.4	本章小结	132
4.5	练习	132
第 5 章	操作数据	133
5.1	理解 SQL 算术	133
5.1.1	基本的数学运算	133
5.1.2	常用数学函数	135
5.1.3	舍入数值	138
5.2	字符串函数	143
5.2.1	SUBSTRING()函数	143
5.2.2	大小写转换函数	145
5.2.3	REVERSE()函数	145
5.2.4	TRIM()函数	146
5.2.5	LENGTH()函数	147

5.2.6	SOUNDEX()函数和 DIFFERENCE()函数	149
5.2.7	日期函数	152
5.3	不同数据类型的转换	153
5.4	再论 NULL	154
5.4.1	NULL 和数学	154
5.4.2	NULL 和字符串	155
5.4.3	COALESCE()函数	156
5.5	INSERT INTO 和 SELECT 语句的综合使用	158
5.6	本章小结	160
5.7	练习	161
第 6 章	分组和聚合数据	162
6.1	分组结果	162
6.2	汇总和聚合数据	164
6.2.1	统计结果的数目	164
6.2.2	累加结果	168
6.2.3	求结果的平均值	170
6.2.4	求结果的最大值和最小值	172
6.3	综合使用 HAVING 子句和 GROUP BY 语句	174
6.4	本章小结	176
6.5	练习	177
第 7 章	从不同的表中选择数据	178
7.1	再论联合	178
7.1.1	深入介绍内部联合	179
7.1.2	外部联合	188
7.2	利用 UNION 运算符组合结果集	196
7.3	本章小结	201
7.4	练习	202
第 8 章	子查询	203
8.1	子查询术语	203
8.2	SELECT 列表中的子查询	204
8.3	WHERE 子句中的子查询	207
8.4	子查询中的运算符	210
8.4.1	再论 IN 运算符	210
8.4.2	使用 ANY、SOME 和 ALL 运算符	213
8.4.3	使用 EXISTS 运算符	216
8.4.4	在 HAVING 子句中使用子查询	219
8.5	相关子查询	220
8.6	用于其他语句的子查询	222

8.6.1	在 INSERT 语句中使用子查询	222
8.6.2	在 UPDATE 语句中使用子查询	224
8.6.3	在 DELETE FROM 语句中使用子查询	225
8.7	本章小结	227
8.8	练习	228
第 9 章	高级查询	229
9.1	更新数据库	229
9.2	处理复杂的查询	235
9.3	高效查询的一些技巧	248
9.4	本章小结	250
9.5	练习	250
第 10 章	视图	252
10.1	视图概述	252
10.1.1	创建视图	253
10.1.2	视图的类型	254
10.1.3	更新视图	258
10.1.4	删除视图	262
10.2	本章小结	262
10.3	练习	263
第 11 章	事务	264
11.1	事务简介	264
11.1.1	示例数据	265
11.1.2	ANSI 事务模型	268
11.1.3	Transact-SQL	270
11.2	事务日志	272
11.3	锁	273
11.3.1	锁的粒度	274
11.3.2	锁的级别	275
11.3.3	设置锁的参数	276
11.4	隔离级别	278
11.4.1	SET TRANSACTION 语句	278
11.4.2	SERIALIZABLE 隔离级别	278
11.4.3	REPEATABLE READ 隔离级别	279
11.4.4	READ COMMITTED 隔离级别	279
11.4.5	READ UNCOMMITTED 隔离级别	279
11.4.6	版本控制	280
11.5	问题示例	280
11.5.1	更新丢失	280

11.5.2	未确认的数据	281
11.5.3	不一致的数据	281
11.5.4	幻影插入	281
11.6	重新编写示例代码	282
11.7	本章小结	284
11.8	练习	285
第 12 章	SQL 安全	286
12.1	安全的概念	286
12.1.1	用户 ID	287
12.1.2	组 ID(角色)	290
12.1.3	对象	292
12.1.4	许可权	293
12.1.5	视图和安全	294
12.1.6	授予许可权	296
12.1.7	废除许可权	299
12.2	本章小结	303
12.3	练习	303
第 13 章	数据库的调整	304
13.1	调整硬件	304
13.1.1	工作站	305
13.1.2	数据库文件	305
13.1.3	处理器	306
13.1.4	GHz 网络或划分网络	306
13.1.5	Cache	306
13.2	调整 SQL	309
13.2.1	SQL 调整意味着什么	309
13.2.2	SQL 调整的原因	309
13.2.3	SQL 调整的方法	310
13.3	一些调整技巧	314
13.4	本章小结	316
13.5	练习	316
附录 A	练习题答案	317
附录 B	建立并使用五个数据库系统	338
B.1	安装并使用 MySQL 4.1	338
B.2	安装 MySQL 的工具	346
B.3	安装并使用 IBM DB2	353
B.4	使用 Command Center 工具	360
B.5	安装并使用 Microsoft Access XP	365

B.6	安装并使用 Microsoft SQL Server 2000	369
B.7	使用 SQL Server 工具	375
B.8	安装并使用 Oracle 10g.....	381
B.9	使用 Oracle 的 SQL*Plus 工具	383
附录 C	初始数据设置	386
C.1	Category 表.....	386
C.2	MemberDetails 表	387
C.2.1	非 Oracle 数据库	387
C.2.2	Oracle 数据库	390
C.3	Location 表	393
C.4	FavCategory 表.....	394
C.5	Films 表	397
C.6	Attendance 表	401
C.6.1	非 Oracle 数据库	402
C.6.2	Oracle 数据库	405

第 1 章 SQL 简介

本章是一个入门章节，首先介绍了什么是数据库，以及为什么和什么时候需要使用数据库。然后介绍 SQL，并揭示它是如何与数据库链接在一起的，以及它是如何起作用的。介绍完 SQL 的基础知识和它的工作原理之后，您将了解到如何使用它创建数据库。本章也将介绍如何创建示例数据库的结构，本书从始到末将用到该示例数据库。

阅读完本章之后，您应当了解到：数据库是如何高效地组织和检索所需要的数据，以及如何创建一个具有完全功能的数据库，该数据库能够接受数据的添加。但是在深入了解 SQL 代码的编写之前，了解一些数据库的背景知识是非常有必要的。

1.1 数据库简史

现代数据库出现在 20 世纪 60 年代，这要感谢 IBM 公司和其他一些公司的研究。这些研究主要集中在办公自动化，特别是数据存储的自动化和以前需要大量手工劳动的检索任务。如今，计算能力和存储已经变得非常便宜，使得计算机应用于数据检索和存储一个可行的解决方案成为可能。数据库领域的先驱之一是 Charles W. Bachman，他因数据库技术方面的开创性工作获得了 1973 年的图灵奖。在 1970 年，IBM 的研究员 Ted Codd 出版了关于关系型数据库的第一篇论文。

尽管 IBM 是数据库研究的领导者，但 Honeywell 信息系统公司在 1976 年首先发布了一个信息系统商业产品，该产品的原理与 IBM 信息系统相同，但它的设计与实现独立于 IBM 的工作。

20 世纪 80 年代早期，一些公司开发了构建于 SQL 标准之上的数据库系统，例如 Oracle 公司的 Oracle Version 2，IBM 公司的 SQL/DS，以及其他一些公司的大量数据库系统。

现在您已经简单了解了数据库的起源，接着您将了解到什么是数据库和什么时间以及为什么需要使用它们。

认识数据库

什么是数据库？

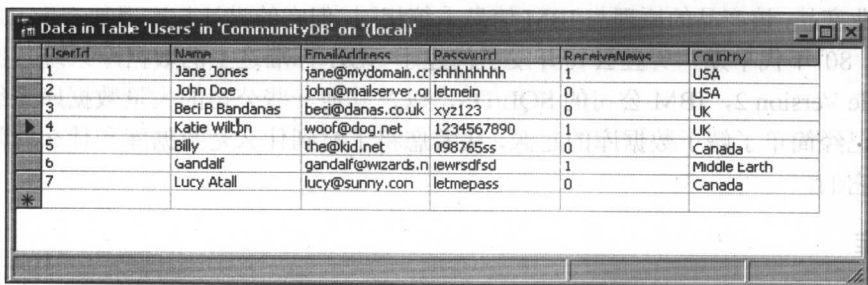
免费在线计算字典(<http://foldoc.doc.ic.ac.uk>)将数据库定义为“永久数据的一个或多个大型结构化集合，它通常与更新和查询数据的软件相关。简单的数据库可以是一个包含许多记录(record)的文件，其中每个记录包含相同的字段(field)集合，并且每个字段具有特定的固定宽度。”

将这个定义划分为一些更易理解的部分，首先它说明一个数据库包含多个关于数据的结构化集合，这意味着数据库包含数据的集合。例如，数据库可能包含关于 Bob 叔叔的高尔夫成绩的细节，或者包含关于一个图书馆中所有书籍的数据。您很可能不愿意混合这两个数据集合，否则当想查找关于一本书的数据时，将不得不浏览一些关于高尔夫成绩的不相关数据。简而言之，数据库帮助您组织数据。数据库以表(table)的方式存储它的数据集合，关于表的概念将在第 2 章中进一步介绍。

这个定义接着说明，数据库通常与更新数据和查询数据的软件相关。数据库软件的实际示例包括微软的 Access、Oracle 的 10g、MySQL AB 的 MySQL 和微软的 SQL Server 2000。通常这些程序被称作数据库，但严格地说，它们是数据库管理系统(DBMS)。一个数据库是一些聚合为一个实体的集合(相关数据的集合)。例如，可以创建一个 Access 数据库，并称它为 MyDatabase，在该数据库中包含各种数据集合，并利用微软的 Access 软件管理所有的数据。

最后，这个定义表明：像 Access 数据库示例一样，一个简单的数据库可以只包含一个文件，该文件中具有多个记录，每个记录可以被划分为多个字段。但记录和字段是什么呢？字段是关于特定事物的单个数据项。事物可以是一个人，关于一个人的单个数据项可以是他的出生日期。或者事物可以是一个家庭的地址，关于地址的特定数据项可以是它的街道号。在书籍示例中，书籍的发行年份是一个特定的数据，它可以被存储在一个字段中。另一个字段可以是书籍的标题；而另外一个字段可以是作者的名字。对于本书，发行年份字段包含 2005，标题字段包含 *Beginning SQL*，作者字段包含 Paul Wilton 和 John Colby。所有这些字段被认为是一个特定事物，即一本称作 *Beginning SQL* 的书。这些字段的集合称作一个记录。每一本书具有它自己的集合，所有的记录集中存储在数据库中一个称为表的地方。一个数据库可以包含一个或多个表。如果您无法立即理解所有这些信息，请不要担心：我将在本章的后面再次介绍字段和记录的概念。

到目前为止，希望您具有这样的思想：数据库能够帮助您存储、组织和检索数据。需要提到的最后一件事情是术语“关系型数据库(relational database)”，这种数据库所包含的数据被组织和相互链接到一起。数据库中的所有记录被组织到表中。相关数据，例如销售人员的细节，被组织到一个表中。可以将他们销售出的汽车的细节存放在另一个表中，然后声明一种关系，表明哪个销售员出售了哪些汽车——例如，销售员 X 在日期 Z 时出售了汽车 Y。图 1-1 演示了示例数据库中的一个表。第一眼看上去，您可能会注意到它类似于一个电子表格(sheet)，其中行是记录，列包含了记录的字段。在第 3 章中您将发现，实际需要考虑的是数据的集合。



Id	Name	EmailAddress	Password	RetriveNews	Country
1	Jane Jones	jane@mydomain.cc	shhhhhhh	1	USA
2	John Doe	john@mailserver.oi	letmein	0	USA
3	Beci B Bandanas	beci@danas.co.uk	xyz123	0	UK
4	Katie Wilton	woof@dog.net	1234567890	1	UK
5	Billy	the@kid.net	098765ss	0	Canada
6	Gandalf	gandalf@wizards.n	iewrsdfsd	1	Middle Earth
7	Lucy Atall	lucy@sunny.com	letmepass	0	Canada

图 1-1

注意：

当今的大多数数据库管理系统是关系型的，被称作关系型数据库管理系统(RDBMS)。这些系统使得存储数据和返回结果变得更加容易、更加高效。它们允许对数据库提出不同的问题——甚至是数据库的初始设计者所不期望被问及的问题。

1. 为什么和什么时候使用数据库

既然存在大量的存储数据的方式，为什么还要不辞辛劳地创建数据库？数据库具有什么优点？

数据库主要的优点是数据检索的快速和高效性。数据库帮助您以一种逻辑方式组织数据。数据库管理系统是一种精细(fine-tuned)的系统,它能够以您所希望的方式快速地检索您所需要的数据。数据库也能够帮助您将数据划分为特定的部分。从数据库中检索数据称为“查询(querying)”。您将经常看到术语“SQL 查询(SQL query)”,它是指从数据库中提取数据的任何 SQL 代码。在本章后面的部分中将更深入地介绍这个主题。

关系型数据库更进一步的优点是:允许声明不同的数据是如何相互关联的,像在汽车销售数据库示例中看到的一样。如果在关系型数据库中存储销售细节和销售员的数据,问题“销售员 X 在一月份售出了多少辆汽车?”将变得非常容易。如果只是将所有的信息堆积在一个大的文本文件中,则您将发现查询数据并找出特定的答案是一项巨大的任务。

数据库也允许您建立规则,确保在添加、更新和删除数据时,数据保持一致性。假设您的假想汽车销售公司有两个销售员的名字都是 Julie Smith。可以建立一个数据库,确保每个销售员具有一个惟一的 ID,该 ID 称为惟一的标识符(这样不会混淆两个 Julie);否则,了解谁销售了哪些汽车将是不可能的。其他的数据存储系统,例如文本文件或电子表格,不具有这些类型的检查,很容易存储错误的数据。在后面的章节中,您将了解到如何建立其他的规则,以减少数据出错的风险。例如,可以指明一个雇员的社会保险号在数据库中必须是惟一的。或者如果一辆汽车被售出,并且它是被 ID 为 123 的雇员售出的,则可以添加一个检查,查看数据库表中雇员 123 的全部细节。

一个合理构建的数据库将最小化数据的冗余度。这里再次使用汽车销售示例,可以在数据库中只存储一次关于一个销售员的所有细节,然后使用惟一的 ID 来标识每个销售员。当有其他数据与一个特定的销售员相关时(例如,他们售出了哪些汽车),则可以使用这个惟一的 ID 来搜索数据。这种惟一的 ID 通常是一个数值,该数值比起一个人的全名来说,占用的空间要小一些。

数据库存储原始数据——只是一些事实,也就是说,没有智能。一个汽车销售数据库可能包含每一辆汽车的制造、型号和价格,但通常不存储一个月中汽车的平均销售数目,因为可以根据汽车销售信息(即原始数据)计算该数目。

但是,电子表格可以包含处理过的数据,例如平均数和统计分析。数据库通常只存储数据,将数据的处理留给前端程序,或者用户所看到的接口。前端程序的示例包括能够绘制数据库中数据的 Web 页,或者能够链接到数据库中的数据并允许用户查看它们的程序。

使用数据库使得数据的共享变得更容易。您可以在同一台计算机中的多个用户之间共享数据,或者在通过网络或 Internet 连接的不同计算机中的用户之间共享数据。如果上面示例的汽车销售公司在纽约、华盛顿和波士顿具有分支,则可以在某一个位置放置一台包含数据库的计算机,该计算机能够被所有其他办公室通过网络访问。这不仅是可行的,而且是安全的,因为数据库具有清晰定义的结构,并且具有一些保护所包含数据的规则。它们也允许多个人在同一时间访问数据库并改变所存储的数据;数据库管理系统将处理同时发生的改变。假设在您使用的 Excel 电子表格中存在一个这样的潜在混乱:两个销售员在同时改变数据。您希望同时保存这两个改变,但是最后保存电子表格的人的改变将被存储,它会覆盖前面的所有改变。

数据库比起使用专用数据格式(即一种从属于特定程序、制造商或者操作系统的格式)来说,也使得在不同系统之间共享数据变得更容易。例如,在一台装有 Microsoft Office 的 Windows 计算机上可以容易地读取一个 Excel 电子表格,但是在 UNIX、Macintosh 或 Linux 计算机上读取 Excel 电子表格将是一种挑战,因为这些计算机处理数据的方式不同。即使在一台 Windows

计算机上,您也需要安装 Microsoft Office 软件。您可以在一台中心计算机上放置数据库,将数据库管理系统安装在那里,然后通过局部网络或者 Internet 访问它。

作为数据库的可选项,文本文件和电子表格具有一个很大的优点,该优点也是它们的弱点:灵活性。文本文件没有实际的规则。您可以在任意地方插入任意文本数据。电子表格基本上也是这样的。您可以要求用户以一种预先定义的结构添加数据,但是您无法强制用户遵守这样一种要求。使用数据库限制用户仅能访问数据,不允许用户改变结构。

数据库最后一个重要的优点是安全性。大多数数据库管理系统允许创建用户,以声明各种级别的安全。在访问数据库之前,他或她必须以一个特定的用户登录。每个用户具有各种权限和限制。数据库的维护人员具有所有的权限:编辑数据、改变数据库的结构、添加和删除用户,等等。其他的用户可能仅具有查看数据的权限,但不能改变它们,或者您可能甚至希望限制他们所能够看到的数据。许多数据库管理系统提供各种粒度级别的安全,即它们非常明确地指明一个用户能够做什么。它们是一种“全有或全无(all-or-nothing)”的方法,根据该方法,用户或者具有访问权限或者不具有访问权限。

数据库的使用范围非常广。在计算机开发中,数据处理是一个很重要的部分,即使到当今,也是它们的主要角色之一。日常生活或商业中的每个历程几乎都需要数据库。数据库通常用于个人计算机,以存储内部使用的数据,用于公司网络数据库,以存储并共享公司内的信息。Internet 促进了利用数据库共享信息的发展;大多数在线商店都使用数据库。当您访问任意大小的在线商店时,通常都有一个数据库提供所卖货物的所有信息。通常不是采用手工方法创建每个页面,大多数店主使用模板来显示书或 CD 的细节,并利用 SQL 语句从数据库中检索书的信息。设想如果 Amazon 网站手工创建每个页面,那将需要多少工作量啊!

当需要处理大量数据,这些数据需要被搜索、排序或者被有规律地更新时,数据库都能发挥很大的作用。在下几章中您将发现,组合了 SQL 的数据库允许您以期望的顺序获得所想要的答案。

2. 本书中用到的数据库管理系统

数据库用于存储数据,数据库管理系统提供查询数据的方式,而所提供的软件通常允许查看数据。但是如何在数据库管理软件之外使用数据呢?操作系统,无论是 Windows、UNIX、Linux,还是 Macintosh,都提供连接到数据库管理系统的方式和提取数据的方式。您需要编写程序代码,并将代码放置到一个单独的应用程序中,用户在他们的计算机中运行这个应用程序,或者您可以建立一个 Web 页来提取数据。而且您也不会被限制到特定语言,只要所使用的语言允许您连接到数据库管理软件。

您可以在市面上购买任意数量的不同关系型数据库管理系统,但是本书的目标是介绍 SQL,SQL 是一种标准(在下一节中将介绍更多关于标准的知识),并且大多数 RDBMS 都采用它。但是,也存在这些标准不允许您想做事情的情况。另外,您可能会发现各种 DBMS 开发商实现的 SQL 并不一致。本书提供 MS Access、MS SQL Server 2000、IBM DB2、MySQL 和 Oracle 10g 的一些细节。