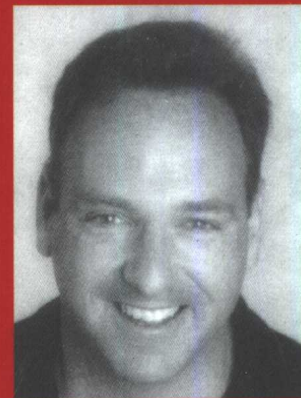
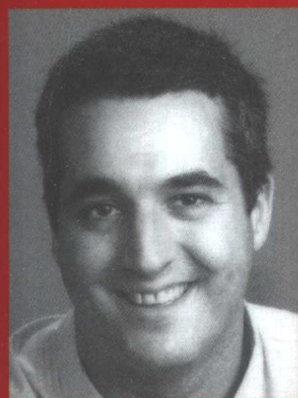
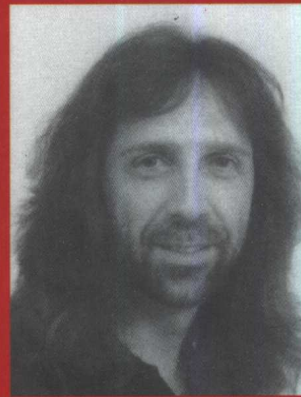
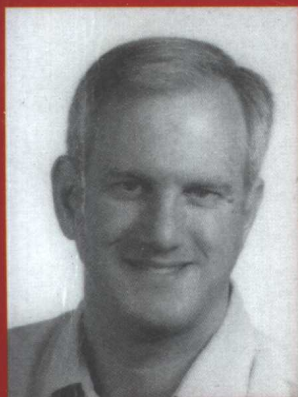




Java数据库 应用程序编程指南

Kevin Mukhar
〔美〕 Todd Lauinger 等著
John Carnell

魏海萍 倪 健 王锐东 等译

Beginning Java Databases

Java数据库应用程序 编程指南

Kevin Mukhar

〔美〕 Todd Lauinger 等著

John Carnell

魏海萍 倪 健 王锐东 等译

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 提 要

Java是一种通用而又稳定的语言,而关系数据库是绝大多数企业所采取的数据存储形式,本书将讲解如何使用这两个强大的技术来建立成功的Java数据库应用程序。首先介绍了SQL及关系数据库的基础知识;接下来专门解释JDBC API的基本原理;最后介绍Java的应用技能,随着Java编程难度的增加,还将讨论如何借助XML和XSLT等其他技术集成Java和数据库。书中还包含有解释清晰、可实际运行的代码,帮助读者迅速掌握所学内容。

本书适用于有一定Java基础的程序设计和开发人员。



Copyright©2001 Wrox Press. All rights reserved. No part of this book may be reproduced, stored in a retrieval system or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embodied in critical articles or reviews.

本书英文版由Wrox公司出版, Wrox公司已将中文版独家版权授予电子工业出版社及北京美迪亚电子信息有限公司。未经许可,不得以任何形式和手段复制或抄袭本书内容。

版权贸易合同登记号: 01-2001-3355

图书在版编目(CIP)数据

Java数据库应用程序编程指南/(美)穆卡尔(Mukhar, K.), (美)劳尔(Lauinger, T.), (美)卡耐尔(Carnell, J.)著;魏海萍等译.-北京:电子工业出版社, 2002.1

书名原文: Beginning Java Databases

ISBN 7-5053-7344-7

I. J… II. ①穆… ②劳… ③卡… ④魏… III. JAVA语言-关系数据库-数据库系统-程序设计 IV. TP312

中国版本图书馆CIP数据核字(2001)第090190号

责任编辑: 春 丽

印 刷: 北京天竺颖华印刷厂

出版发行: 电子工业出版社 <http://www.phei.com.cn>

北京市海淀区万寿路173信箱 邮编: 100036

北京市海淀区翠微东里甲2号 邮编: 100036

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 55.25 字数: 1400千字

版 次: 2002年1月第1版 2002年1月第1次印刷

定 价: 84.00元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换,若书店售缺,请与本社发行部联系。联系电话: (010) 68279077

简介

Java是一种通用而又稳定的编程语言，而且适应各种各样的环境，即使是在中层的应用程序服务器或客户浏览器上。无论用户正在开发何种用途的应用程序，或用户的应用程序将在何种环境中运行，几乎可以肯定的是，这些应用程序都将需要利用存储在某种数据库中的数据。而关系数据库是绝大多数企业所采用的数据存储形式，并且在最近几年中，关系数据库也得到迅猛发展，已经成为强有力的、功能丰富的数据管理系统。

本书的目标是讲解如何使用这两种强有力的技术来建立成功的Java数据库应用程序。本书的基本目标如下：

- 让读者了解关系数据库的设计与操作以及关系数据库所使用的语言SQL。
- 详细介绍JDBC的基本原理。
- 讲解如何开发企业级的、数据稳固的应用程序。我们希望本书在编写有效数据库应用程序方面，能够帮助读者掌握更多的Java知识和提高面向对象设计水平。
- 探讨如何处理对象－关系（Object-Relational）映射及Java映射构架，以便读者能给自己的应用程序提供正确的解决方案。
- 通过实例说明如何将上述内容运用在企业环境中，在这样的环境中，Java程序可能需要与诸如XML之类的其他通用技术交互。

读者对象

虽然这是一本Wrox初学丛书，但严格说来，我们无法从Java，JDBC及数据库这三个核心题目的基础知识开始全面讲起。我们知道，Wrox出版公司的图书通常都很厚，但还没有厚到那个地步！

因此，本书假设读者对Java编程语言的基础知识已有一定的了解，而且阅读目标是建立在此基础上的，所以我们将讲解与数据库编程直接相关的核心Java功能。Java的基础知识是阅读本书的惟一前提。如果读者对关系数据库及JDBC API（Application Programming Interface）仅有一点了解或根本不了解，本书将给读者提供建立有效Java数据库应用程序所需要了解的初中级内容。

因此，在一定程度上，读者可以把本书看作“中高级”初学丛书。对于已完整地阅读过Java基础知识，比如Ivor Horton所著的Beginning Java 2（ISBN：1-861003-66-8），或阅读过类似图书，并且希望把他们的Java技能运用于数据库项目的读者，本书是理想的读物。

如果读者不熟悉Java，但学习过诸如C或C++之类的编程语言，则应该能够读懂本书中所提供的那些代码解释，并且完全可以理解那些代码示例。

内容概述

从概念上讲,本书可以分成五大篇。第一篇为基础篇,介绍SQL及关系数据库的基础知识。第二篇专门解释JDBC API的基本原理。接下来的三篇以前两篇为基础,介绍Java的应用技能,其间我们将逐渐增加Java编程的难度,并探讨如何借助诸如XML和XSLT之类的其他技术来集成Java及数据库。

接下来将简要介绍每一篇内的章节结构。各篇中的每一章都含有解释清晰的代码示例——读者可以实际运行的代码,而不是代码摘录。这些示例将清楚地演示我们正在讨论的问题。

第一篇: Java, SQL及数据库

为了建立有效的Java数据库应用程序,用户必须扎实地了解什么是关系数据库,关系数据库如何工作以及关系数据库使用的语言,即Structured Query Language (SQL, 结构化查询语言)。

第1章为以后的讲解做准备。我们将介绍关系数据库管理系统(RDBMSs)的基本概念,并站在较高的高度上来看一看我们如何以及为什么需要利用SQL和JDBC API来集成Java及关系数据库。我们将描述如何安装Cloudscape RDBMS,以及设置Music Store数据库来存储书中的全部示例数据。

第2章将介绍JDBC技术。我们将描述一些基本的JDBC概念和术语,并介绍JDBC为了能让用户从数据库中检索数据而提供的一些基本Java对象。我们还将介绍代码中所用到的基本错误处理技巧。

至此,我们已经编写了一些基本的SQL语句,用来从数据库表中检索数据,但SQL的威力远远不止这些。第3章和第4章将更深入地探讨SQL。在第3章中,我们将讨论如何使用SQL SELECT语言来查询和操纵数据库中所含有的数据。在第4章中,我们将把叙述重点放在数据库表的创建及修改上,并介绍如何在这些数据库表中创建新的数据行以及如何修改和删除它们。

要想编写成功的应用程序,关键是要有一个设计完善的数据库。应用程序所需要的数据必须得到有效存储,并且必须使用一种保证数据完整性的方式来存储。在第5章中,我们将描述如何把数据组织成数据库表,以及如何定义这些表之间的关系。我们还将站在较高的高度上来看一看可以用在数据库设计中的其他一些数据库对象,包括过程、视图和触发器。

在第6章中,我们将通过设计和建立全书中所用到的Music Store数据库,实际运用前两章所介绍的许多概念。Music Store是我们虚构的一家公司,我们将根据该公司的虚构需求,设计和建立数据库表,并实现一些支持查询、视图以及过程,以便提供公司业务所需要的一些功能。

第二篇: 探索JDBC

通过前一篇的学习,我们已经扎实地掌握了关系数据库的设计,尤其是Music Store示例数据库的设计,而且知道了如何利用SQL编程语言的威力来查询和修改数据。现在,我们

应该开始设计和建立一些能通过JDBC API与数据库（和SQL）交互的Java程序。JDBC提供了能让程序员用来在Java类与数据库之间发送数据的大批Java对象。在本篇中，读者将会学到应用程序开发中将需要使用的核心JDBC类和接口。

第7章是本篇的第一章，将概括性地描述建立Java数据库程序之前将需要做出的一些决策。开发人员将如何建立Java应用程序的模型？将使用什么策略把该模型映射到数据库上？将使用什么JDBC驱动程序？本章将给出关于这些问题的指导，并讨论如何在应用程序中使用数据的关键问题。

第8章是我们真正开始使用JDBC的地方。在应用程序可以做任何事情之前，它将需要建立与数据库的连接。因此，这就是我们开始的地方。我们将讨论JDBC Connection（JDBC连接）接口，并展示如何实现一个ConnectionFactory类，这个类将封装连接代码，并且能用在我们的所有应用程序示例中。

第9章将详细介绍Statement（语句）和ResultSet（结果集）接口。我们将解释如何在SQL中使用Statement对象来执行数据查询和修改命令。然后，我们将讨论ResultSet对象，它用来存储从查询中返回的数据。我们将介绍如何遍历ResultSet数据，以及如何通过ResultSet来更新数据库表中的数据。

在开发应用程序时，我们有时可能没有必要知道应用程序需要使用的数据结构和数据类型。如果希望应用程序能在数据库之间移植，我们就需要一种能查找这些信息的方法。而JDBC就允许用户把元数据（关于数据的信息）用于这一目的。第10章将探讨JDBC所提供的元数据接口。

虽然ResultSet对象对数据检索是一个非常有用的对象，但它们需要维护与数据库的永久性连接。在第11章中，我们将讨论一种代用方案——RowSet（行集）接口。RowSet对象的优点是它能与数据库分开，并能通过网络来发送。我们将介绍开发人员能创建的三个RowSet对象版本：CachedRowSet、JdbcRowSet和WebRowSet，并讨论这3个接口对开发人员来说为什么是有用的。

当把SQL语句发送给数据库时，数据库需要花时间来编译该语句和计算出它的执行“计划”，然后把执行“计划”存储在一个特殊数据区域内。我们经常需要重新发送以前使用过的SQL查询，其中列是相同的，数据库表也是相同的——只有数据是不同的。如果明智，我们将不会让数据库再次执行所有的编译工作和存储一个全新的执行计划。第12章将讨论PreparedStatement（准备语句）接口，该接口可以提供一种创建可重用SQL语句的手段——一旦数据库首次编译了它，它就能一次又一次地使用同一个执行计划。

数据库只不过是一个存储场所——就它的地位来说，它是一个编程平台，程序员可以在这个平台中建立过程和函数来实施应用程序的安全性、数据完整性以及事务规则。在第13章中，我们将讨论如何创建这些存储过程，然后讨论如何使用CallableStatement（可调用语句）接口从Java代码中调用它们。

第14章将讨论一个重要题目：事务处理。我们将介绍什么是事务以及它们必须描述的属性。我们将讨论数据库用来保证数据库中同步发生的事务不相互接口的不同机制。在该上下文中，我们将涉及加锁、并发以及事务隔离等级这几个复杂的问题。我们将包含一些思路清晰的示例，用于讲解如何通过JDBC来控制事务。

第三篇：对象—关系映射

在本篇中，我们开始学习Java编程技能，并检验我们对前两篇中所介绍的那些概念的掌握程度，因此将在Java应用程序中增加难度。我们将介绍当为Java应用程序精心构造的对象模型与关系数据库的设计“不匹配”时我们所面对的困难。我们将看一看怎么才能使用Java框架和JDBC把这些较复杂的应用程序映射到数据库上。

Java是一种通过使用一组对象来完成工作的面向对象软件开发语言。关系数据库使用一种基于表的关系模型。随着Java应用程序变得越来越复杂，读者将会发现对象模型与数据库设计的“不匹配性”。在第15章中，我们将建立一个使用一种“类与表一一映射”的应用程序，然后从概念上给该应用程序增加难度（比如层次对象关系），并讨论我们在映射一个模型到另一个模型时将会遇到的这种问题。

紧跟在前一章的后面，第16章将使用一种称作“Java分层框架”（Java Layered Frameworks）的开放式源框架把Java对象映射到关系数据库上。我们将看一看该框架怎么才能解决前一章所讨论的一些困难。该框架相当复杂，但读者将会从研究它如何工作中学到很多东西。

随着Java数据库变得越来越复杂，记录应用程序的活动日志也随之变得越来越有必要，同样，能够成功地调试应用程序也变得越来越有必要。这也正是第17章的主题。我们将讨论如何把日志记录到System.out和System.err输出流上的一些技巧，它们分别运行本地和交互式调试程序。

第四篇：企业中的Java及数据库

从本质上讲，本篇将让读者领略一下我们怎么才能把新技巧应用到企业级应用程序开发领域中。我们将看一看怎么才能利用一些Java类在Java 2 Enterprise Edition框架内工作，以及怎么才能采用该框架所提供的那些组件和服务。我们将讨论怎么才能建立灵活而精密的Java数据库应用程序，并使之与诸如XML和XSLT之类的数据交换及变换技术相接口。

新发布的JDBC 2.0 API带有DataSource（数据源）对象。由于我们把连接代码存储在一个DataSource对象中，因此可以在同一个地方修改所有的连接参数，同时又不必重新编译应用程序。这可以给代码增添极大的灵活性和可移植性。在第18章中，我们将研究这一功能，并看一看它怎么能使我们在应用程序中使用诸如连接共享和分布式事务之类的高级功能。

第19章将举例说明怎么才能利用迄今为止所学到的技巧来建立适合企业需求的Java数据库应用程序。我们的讲解重点是J2EE（Enterprise Edition of the Java 2）平台中所提供的功能和组件。我们已经精心设计了我们的类，因此把它们用于这个新环境是一项相对轻松的任务。我们将展示如何使用JSP（Java Server Pages）来利用前面所开发的一个Web类。然后，我们将深入研究EJBs（Enterprise JavaBeans）这一高级题目。同样，我们将采用前面所开发的一个类，并把它用作一个EJB。

第20章将展示我们的应用程序怎么才能结合使用XML——XML是一种自描述标记语言，它已彻底地变革了我们在应用程序之间传输和共享数据的方式。在本章中，我们将看一看怎么才能把关系数据映射到XML上。

一旦读者拥有了XML格式的数据，就可以使用许多方法来变换它们，使用何种方法取决于应用程序的需要。我们可以把它们变换成HTML供浏览器使用，把它们变换成文本格式，或把它们变换成一种不同的XML格式。在第21章中，我们将看一看XSLT——处理XML的主要方法。我们将看一看如何实际实现XML和XSLT，以及怎样把它们运用到全书中所使用的Music Store应用程序上。

作为本篇和本书的最后一章，第22章将综述读者迄今为止所学到的大量知识。本章将阐述开发生产性Java数据库应用程序时要考虑的一些问题。首先，我们将用一节来描述应用程序的潜在体系结构。其后，我们将实现一个称作模型-视图-控制器（Model-View-Controller）的框架，并结合开放源框架（比如Tomcat、JBoss、Xalan及Xerces）的使用来说明如何使用Java建立功能齐全的、Web型企业级数据库应用程序。

第五篇：附录

这几个附录的目的是帮助读者加速理解XML和指导读者安装本书所用到的各种软件，而且让读者知道可以在何处获得帮助（如果需要）。

附录A说明如何在各种RDBMSs上安装和访问Music Store数据库，其中包括Oracle、Cloudscape以及Microsoft Access。

附录B含有关于如何安装JBoss应用程序服务器和Tomcat Web服务器的说明和提示。

由于本书最后3章需要一些XML知识，所以我们在附录C中提供了一个XML初级读本。

最后，如果读者在阅读本书时需要帮助和指导，那么附录D将给读者指出正确的方向来获得一些建议。

阅读本书的要求

要想实际运用本书绝大多数章节中所讨论的那些理论，读者所需要的全部东西是JDK（Java Development Kit）和一个带有适当JDBC驱动程序数据库。

我们在本书中使用了JDK 1.3，读者可以从Sun公司的Web站点（<http://java.sun.com>）上获得它。本书中所广泛使用的两个数据库是Cloudscape和Oracle，而且只需一点调整，读者就应该能够得到可在这两个数据库之一上工作的代码。本书的第1章给读者展示了如何获得和安装Cloudscape，而附录A介绍了如何获得Personal Oracle 8i的拷贝，以及如何在计算机上安装它。

随着我们的代码示例变得越来越复杂，以及我们开始丰富主题并含有更多种技术的内容，读者将需要安装和设置新的软件。在本书内的每个相关知识点，我们将给出相关的指导来告诉读者从何处获得软件，以及如何安装并设置它。软件的示例包括开放源、用于对象-关系映射章节的Java分层框架实现、用于XML和XSLT章节的Xalan和Xerces语法分析器以及用于本书结尾部分的Web型企业级应用程序的JBoss和Tomcat服务器。

本书的写作约定

为了帮助读者理解和注意全书中讲解的内容，并为了维持一致性，我们使用不同的字体和排版布局。下面是示例以及它们所代表的含义：

这是一种用来给出除普通讨论之外的某些信息（如建议、提示及其他注释性信息）的样式。

重要信息以黑体形式排版。

练习——读者要试一试的示例

在介绍和学习了某个新概念之后，有一个练习部分用来说明如何使用它。这些代码段是完整示例，并含有如何正确地执行它们的指导。

工作原理

在练习部分的后面，通常有一个“工作原理”的部分，该部分的作用正如该标题所暗示的。在此，我们将解释这段代码如何工作的重要而基础的机制。这是为了帮助读者把刚刚做的与刚刚学的结合起来。

下载源代码

在通读本书内的示例时，读者可能会决定亲自键入全部代码。许多读者喜欢这么做，因为这是熟悉编程技巧的一条上佳途径。

无论读者是否打算键入这些代码，我们已经把本书的全部源代码都放在了下列Web站点上：<http://www.wrox.com/>。单击页面顶端附近的那个Java链接，然后在Java图书清单顶端查找本书。

如果读者愿意亲自键入代码，可以使用我们的文件来检查应该得到的键入结果。如果读者觉得自己可能键入了错误的代码，这些文件应该是读者的第一站。如果读者不打算亲自键入这些代码，可以从上述Web站点中下载这些源代码！

总之，该站点对读者的更新和调试都会有所帮助。

信息反馈

我们一直努力使本书尽可能地准确和令读者阅读愉快，但真正重要的东西是本书所讲解的那些内容。请让我们了解你对本书的看法，读者可以给我们反馈，通过E-mail地址 feedback@wrox.com 跟我们联系。

设置Java环境

在全书中，我们一直使用需要编译和运行的示例。为此，读者将需要保证一些设置。首先，读者需要保证自己的Java安装已经得到正确安装和配置。

安装JDK

对于本书，我们使用了Sun Microsystems JDK 1.3，而且假设它已安装在了C:\jdk1.3目录内。该目录也是我们将要在JAVA_HOME变量中设置的值。如果读者把JDK安装在了一个不同的目录内，则需要保证JAVA_HOME变量指向这个正确的地方。如果还没有该工具箱，读者可以从<http://www.java.sun.com/j2se/>网站上获得它。

设置JAVA_HOME变量

设置JAVA_HOME变量有几种方法，而使用何种方法取决于读者的计算机正运行何种系统。如果读者有Windows 9x，在autoexec.bat文件中包含下列行：

```
SET JAVA_HOME=C:\jdk1.3
```

读者对autoexec.bat文件所做的任何修改只有在重新启动了计算机后才会生效，这是因为autoexec.bat文件只有在计算机启动时才会被读取。如果读者正在运行Windows NT或Windows 2000，可以按下列方式来设置环境变量：

1. 右击桌面上的“我的电脑”图标。
2. 选择“属性”。
3. 选择“高级”选项卡。
4. 单击“环境变量”按钮。
5. 在屏幕的下半部分和“系统变量”的下方，单击“新建”。
6. 现在，把“变量名”设置为JAVA_HOME，并把“变量值”设置成指向JDK安装（如C:\jdk1.3）。
7. 单击“新系统变量”对话框上的“确定”，接着单击“环境变量”对话框上的“确定”，然后单击“系统属性”对话框上的“确定”。

当打开一个新的命令窗口时，读者给环境变量所做的添加将生效。如果已有一个命令窗口处于打开状态，这个新的变量在该窗口中将无效。

接下来将要检查的东西是PATH和CLASSPATH设置。

设置路径

为了能够运行Java编译程序javac和Java运行时java，读者的计算机需要知道在哪里寻找它们。要想保证在键入javac命令时将不会得到下列消息：

```
'javac' is not recognized as an internal or external command, operable
program or batch file.
```

读者需要给PATH变量添加JDK的bin目录。也就是说，当在命令提示符处键入命令时，系统将在JDK安装的\bin目录内查找读者所需要的程序。同样，设置PATH变量有几种方法，使用何种方法取决于读者所运行的操作系统。

如果读者正在使用Windows 9x，可以给自己的autoexec.bat文件添加如下行：

```
SET PATH=%PATH%;%JAVA_HOME%\bin
```

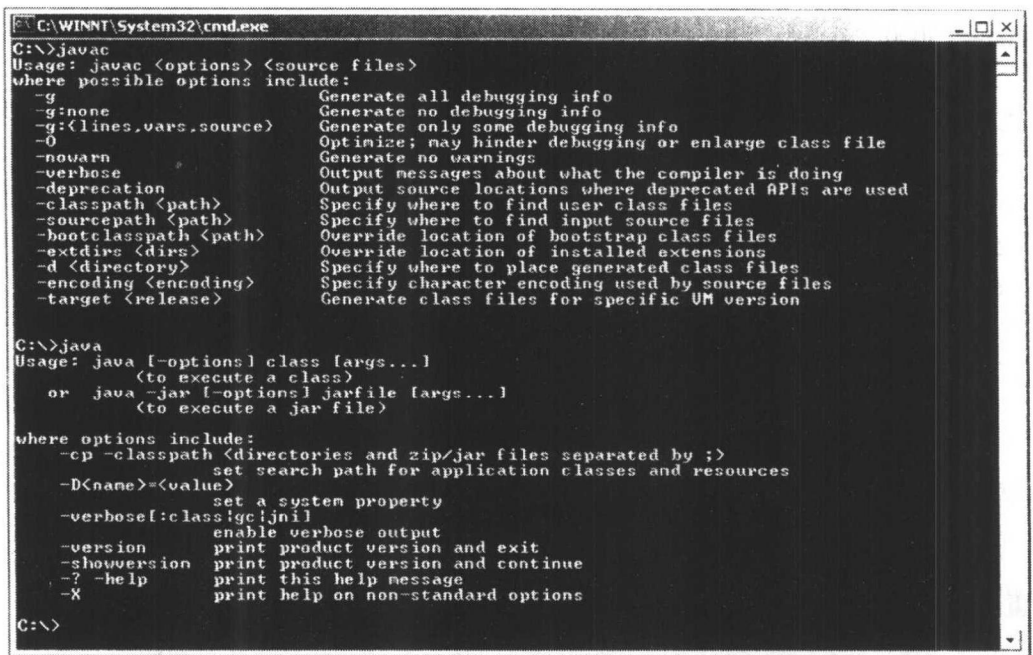
这行代码的效果相当于在PATH变量的尾部添加C:\jdk1.3\bin，也就是说，当我们试图运行命令时，系统将在C:\jdk1.3\bin目录内查找。

要使用分号(;)把值分隔开，以便系统知道一个目录在哪里结束和另一个目录从哪里开始，这一点是非常重要的。

如果读者正在运行Windows 2000，请转到“环境变量”对话框（“我的电脑”！“属性”！“高级”！“环境变量”），选择路径，然后单击“编辑”按钮。滚动到“变量值”框的尾端，然后添加下列代码：

```
;%JAVA_HOME%\bin
```

像前面一样单击了“确定”按钮之后，效果就如同修改了autoexec.bat文件，即给PATH变量添加C:\jdk1.3\bin。现在，当在命令提示符处键入javac或java时，应该得到该命令的一个有效选项清单，如图1中所示。



```
C:\WINNT\System32\cmd.exe
C:\>javac
Usage: javac <options> <source files>
where possible options include:
-g Generate all debugging info
-g:none Generate no debugging info
-g:<lines,vars,source> Generate only some debugging info
-O Optimize; may hinder debugging or enlarge class file
-nowarn Generate no warnings
-verbose Output messages about what the compiler is doing
-deprecation Output source locations where deprecated APIs are used
-classpath <path> Specify where to find user class files
-sourcepath <path> Specify where to find input source files
-bootclasspath <path> Override location of bootstrap class files
-extdirs <dirs> Override location of installed extensions
-d <directory> Specify where to place generated class files
-encoding <encoding> Specify character encoding used by source files
-target <release> Generate class files for specific VM version

C:\>java
Usage: java [-options] class [args...]
        (to execute a class)
or java -jar [-options] jarfile [args...]
        (to execute a jar file)
where options include:
-cp -classpath <directories and zip/jar files separated by ;>
    set search path for application classes and resources
-D<name>=<value>
    set a system property
-verbose[:class[:c1[:jnil
    enable verbose output
-version
    print product version and exit
-showversion
    print product version and continue
-? -help
    print this help message
-X
    print help on non-standard options

C:\>
```

设置CLASSPATH

为了能够编译和运行本书中的许多示例，读者不仅需要有一个Java编译程序，而且还需要各种类文件和JAR文件。同告诉系统在哪里寻找程序的系统变量PATH相类似，CLASSPATH

同样由Java用来查找类。

虽然读者在自己的类路径中所需要的任何JAR文件都必须明确地添加，但是CLASSPATH变量可以既含有命令，又含有JAR文件。使用CLASSPATH变量有两种方法。首先，读者可以设置一个CLASSPATH变量来包含自己所需要的所有文件，这可以通过声明一个CLASSPATH变量（像上述声明JAVA_HOME和PATH变量一样）并把它设置成指向所需目录来实现。对于Windows 9x，读者可以给自己的autoexec.bat文件添加下列代码行：

```
SET CLASSPATH=%CLASSPATH%;%JAVA_HOME%\lib
```

在Windows 2000下，读者可以找到“环境变量”对话框上的CLASSPATH变量，并添加如下代码：

```
;%JAVA_HOME%\lib
```

每当类路径中需要额外的文件时，可以通过上面介绍过的那些方法，或者在命令提示符处进行如下所示的设置，把它们添加到环境变量上：

```
SET CLASSPATH=%CLASSPATH%;c:\java\lib\useful.jar
```

其中，我们正在把C:\java\lib\useful.jar文件添加到类路径上。这种方法在添加常用JAR文件（和目录）到类路径上时是十分有用的。

使用类路径的另一种方法是变化的，取决于我们如何给类路径进行添加。像前面一样声明CLASSPATH变量，这次，不是给类路径进行添加，而是把它用作运行javac或java时-classpath选项的一个起始点：

```
> javac -classpath %CLASSPATH%;c:\java\lib\jndi.jar;. RunSomething.java
```

其中，我们正在给类路径添加两样东西，它们仅用于javac的此次调用：

- C:\java\lib\jndi.jar文件
- 由句点（.）指出的当前目录

虽然使用这种方法的键入速度较慢，但它有一个胜于另一种方法的明显优点——它有助于类的快速编译。这是因为javac在编译一个类时，将查遍所有目录和类路径中的所有JAR文件。读者在自己的路径中有越多的JAR文件，搜索的时间就会越长。

决定怎样设置和使用类路径完全取决于读者本人，只要读者保证自己在类路径中已有那些文件。

示例的目录结构

为了方便起见，本书中的所有示例都将从一个目录（即C:\BegJavaDB）中运行。每一章都将有它自己的子目录，这些目录被命名为Ch01、Ch02，依次类推。

此外，为了方便起见，我们也将把所遇到的全部JAR文件都存储在同一个地方，即c:\java\lib目录中。为了减少键入时间，我们可以设置一个环境变量来指向这个目录，因此把它称作JLIB这样的短语。我们可以使用设置JAVA_HOME的方法来设置它。

假设我们有一个叫作jndi.jar的JAR文件，并且它存储在C:\java\lib目录中，那么我们把我们的JLIB变量设置成指向C:\java\lib目录。使用这个新变量，我们可以在运行时按如下方式

给类路径添加新的JAR文件:

```
> javac -classpath %CLASSPATH%;.;%JLIB%\jndi.jar RunSomething.java
```

像前面一样, 我们正在包含当前目录(这次把它指定在%CLASSPATH%与%JLIB\jndi.jar之间)和c:\java\lib\jndi.jar文件。Javac命令将求解%JLIB%到JLIB变量的内容, 即c:\java\lib目录。

一般情况下, 文中会有一些指导, 用来说明特定代码示例应该存储在C:\BegJavaDB\目录结构内的什么地方。通常, Java代码段的开始处会有package声明。这就意味着相关脚本应该存储在这一章子目录内的文件夹中。例如, 如果某个.java文件的开始处有如下代码段:

```
package results;

import java.sql.*;
import java.io.*;
import connections.*;

public class ScrollingDemo {
    .
    .
    .
}
```

那么, 我们马上就会有这样一个概念: 这个文件应该在这个相关的章文件夹内, 并存在一个名为results的目录中, 且文件名为ScrollingDemo.java。

另外, 本书的一些代码在许多演示中都得到重用。例如, 我们在第8章“数据库连接”中所开发的ConnectionFactory类在许多章节中都会用到。这个类被包含在connections子文件夹中, 并由Java脚本段开始处的import connections.*语句来调用。读者需要尽可能地确保自己的工作目录内有一个名为connections且含有ConnectionFactory类的文件夹; 至少要确保类路径所指向的地方存在一个含有ConnectionFactory类拷贝的connections目录。

作者简介

Kevin Mukhar

Kevin Mukhar从少年时就开始编程，初学时是在电传终端上使用COBOL语言进行编程。他也在HP计算器和第一代家用电脑之一的Commodore 64上做过编程。遗憾的是，他没有意识到自己从编程中享受到了多大的乐趣，因此在读大学期间只选修了一门编程课程。他毕业于华盛顿大学，并获得数学学士学位，然后接受了美国空军的一项任务。在空军服役期间，他继续把编程作为自己的业余爱好，因而学会了包括Pascal、FORTRAN、C、dBase和汇编在内的几种编程语言。这段经历帮助他在离开空军之后获得了一份软件开发的工作。

自从成为一名软件开发人员之后，他一直使用C和C++编写一个基于客户-服务器的培训系统。后来，他开发了一个使用Oracle数据库来管理百亿字节二进制数据的大型数据管理系统。他使用C++和Java语言把该系统的前端开发为一个Web应用程序。这个应用程序的后面是一个用来从分布式数据贮藏库中动态查找和显示数据的纯Java系统。他最近的工作是使用J2EE技术来开发企业内部网Web程序。这个Web应用程序为具有百万用户客户程序的公司提供客户管理。

他心目中的英雄是Steve McConnell (<http://www.construx.com/stevemcc/>)，Philip Greenspun (<http://philip.greenspun.com/>) 和Ron Jeffries (<http://c2.com/cgi/wiki?Ron-Jeffries>)。

Kevin感谢Dave Johnson再次唤醒了他成为一名作家的愿望和提供了初次写作方面的帮助。他还感谢Wrox出版公司的编辑们和审稿人员，谢谢他们帮助他润饰词句和文稿。

Kevin把他撰写的部分献给妻子Anne。

Todd Lauinger

Todd Lauinger是一名自由职业的讲师、顾问、会议演说者和图书作者。他目前受雇为Best Buy有限公司的一名Software Construction Fellow，他在那里的工作是为该企业内的所有Java软件开发活动建立一个公用的基础结构。Todd拥有软件工程硕士学位，同时拥有10多年为工程和商业机构开发大型、关键任务软件系统的经历。工作之余，他喜欢陪伴妻子徒步旅行、骑自行车、打板球和玩排球。

我希望把自己在本书中的成就献给我的Lord和Savior，他们给了我这样的机会和才能。感谢我的爱妻Teri，没有她的支持、鼓励和耐心，我永远无法把那些才能付诸行动。献上我所有的爱！

John Carnell

John Carnell自从12岁开始摆弄他的Commodore 64以来，对计算机一直很着迷，而且着迷程度经久不衰。目前，他正为Workspace（一家提供HR及雇员福利自助解决方案的领先

供应商)工作,职务是高级系统设计师。John喜欢与妻子Janet讨论的题目是基于组件的n层体系结构。John在Microsoft、Oracle和Java n层解决方案方面有着丰富的经验。他与妻子及两只小狗Ladybug和Ginger居住在威斯康辛州的Waukesha市。John的E-mail地址是john_carnell@yahoo.com。

献给我的妻子Janet: 我们一直共同享受着生活的每一刻。我的生活是在我18岁的时候开始的,那时我正和你坐在网球场上,卷缩在我的夹克下面躲避倾盆大雨。就在那个时刻我意识到你十分美妙而又与众不同。11年后,你仍使我惊奇和着迷。谢谢你这些年来给我的爱和支持。你的帮助才使我成为了一个更好的人。我爱你。

James R. de Carli

James R de Carli在计算机行业的经历涉及写作、教学及应用程序开发。

在通过了20多门资格认证考试之后,James获得了几种专业证书,其中包括Microsoft Certified Systems Engineer (MCSE), Microsoft Certified Solutions Developer (MCS D)和Microsoft Certified Database Administrator (MCDBA)。此外,James还拥有计算机与信息科学硕士学位和数据与统计学学士学位。

James还给各种专业杂志写了大量文章,而且有两本合著书籍“Professional JavaScript”和“Professional ASP Data Access”,这两本书都由Wrox出版公司出版。现在,他正受雇于SiteShell公司(一家总部设在康涅狄格州的因特网媒体供应商),职务是网站结构与开发部主任。他还是Fairfield大学的一名副教授,教高年级的软件工程课程。James曾经为一家微软授权的培训与教育中心当了许多年的指导老师,而且还曾经是一家微软解决方案供应商联合公司的高级顾问。

James与妻子Catherine及3个儿子Kenneth、Robert和Daniel居住在康涅狄格州。

Mark Mamone

Mark Mamone是Syntegra(一家全球性的咨询与系统集成商行)的专业设计权威,目前正在为一个全球电子商务系统设计和实现J2EE和Microsoft .NET解决方案。Mark在IT行业从事专业工作达13年,曾经为像Thomas Cook、ABN AMRO和Royal Mail这样的公司工作过。他最近的成果是一家全球性物流公司的一个WAP网关和用于Syntegra商行内各新项目的Microsoft eFramework定义。

我希望把本书献给Julie、James和Cameron,如果没有他们的理解、鼓励和支持,本书将永远都完成不了。我还想感谢我的妈妈和爸爸,没有他们给我买第一台计算机Commodore VIC-20,我的职业可能永远都不会走上这条道路。

Nitin Nanda

Nitin Nanda目前在Quark有限公司的R & D中心做项目副经理,该公司的总部在印度的昌迪加尔市。他在该中心负责管理一套产品的R&D,这些产品的目标是打印目录市场。这些产品使用RMI和CORBA做为中间层开发的基础协议。

他有一本合著书籍“Professional Java Data”，该书由Wrox出版公司出版（中译本《Java数据编程指南》已由电子工业出版社出版）。另外，他也在像Java World、Java Developer's Journal和Java Report之类的第一流Java杂志上发表文章。他把业余时间花费在出版物和图书翻译上。

Nitin希望把他的谢意送给他的Guru（精神导师）Ishwar Devi Gupta，此人教会了他生活的基本真理：要向着使自然科学与精神世界都保持平衡的方向发展！

Damon Payne

Damon去年搬到了威斯康辛州的密尔沃基市，他现在就居住在那里。他现在为Cornerstone Consulting有限公司工作。他的专业技能主要在于使用Java和XML技术开发瘦客户Web应用程序。Damon的其他专业兴趣包括无线应用程序、对象数据库以及开放源软件开发。

当他不在计算机旁边时，Damon喜欢摆弄他的WRX、电子音乐以及玩吉它。他也选修了武术课程，并定期去武馆练习。

我想谢谢John Carnell，他激发了我对专业写作的兴趣，并把我推荐给了Wrox。

Joel Peach

Joel Peach是Consulting Services的副总裁，而且也是俄亥俄州哥伦布市的Tracer Information Systems的合伙创办人。他是一名Java认证程序员，并且在使用J2EE平台和数据库实现分布式Java企业应用程序方面有着丰富的经验。

他的电子邮件地址是jpeach@traceris.com。

Joel想感谢Tom Acree对他的提携，帮助他成为一个开发人员；感谢Sarah Camper和Chet Cady在他说他某一天将成为一个作家时的鼓励、支持和没有嘲笑；感谢Wrox出版公司的Tony Davis、Vel Ilic对他的创作支持。

感谢Gina Lynch，谢谢你酷得足以让我暂时忘怀你，去追求一个几乎和你一样美妙的机会。

Ron Phillips

Ron Phillips从事商业软件工具的设计和开发已有10多年的时间，这些工具包括从IDEs、语言、编译程序和软件设计工具到他的信用卡。他是IT团体内一名积极分子，已经在美国和欧洲的许多行业会议上发表过演讲。他当前的研究工作主要是分布式系统与结构模式挖掘中的松散联合。

David Shanes

David Shanes是Predicate Logic Software Systems有限公司的一名高级软件工程师，这家公司位于加州的San Diego市。在该公司，他的工作与美国海空作战系统中心项目的中间层Java开发有关。David自1996年以来一直与Java打交道，而且在面向对象设计与开发、

Java、数据库、多线程以及网络等编程方面有着丰富的经验。

David于1988从美国海军学院获得计算机科学学士学位，目前正在圣地亚哥州立大学攻读计算机科学硕士学位。

David的兴趣包括骑自行车、溜旱冰、航海以及陪伴妻子和两个孩子。

谢谢我妻子Mindy和我们的孩子Joshua和Jessica，你们始终让我充满快乐，感谢你们对我是如此的支持，让我踌躇满志，以及给我无私的爱。我全心全意地爱着你们。

感谢我妈妈，谢谢她在我成长的时候对我的指导和耐心，感谢我的岳母，谢谢你在我们需要你的时候总是来照顾我们。

最后，我想把本书献给我的爸爸，他为供养我们而劳作了一生。谢谢爸爸为我展示了努力工作和诚实的意义，谢谢爸爸总是握着妈妈的手。我怀念你，爸爸。

Sakhr Youness

Sakhr Youness是Commerce One有限公司的一名高级设计师。他是一名专业工程师（PE），也是一名微软认证解决方案开发员（MCSD），并在数据建模、企业到企业（B2B）应用、客户-服务器以及数据库应用开发方面有10多年的经历。做为一名高级软件设计师和技术顾问，他设计、开发或参与开发了与汽车、银行及工程等行业相关的许多在线B2B及客户-服务器应用程序。

Youness先生是2000年1月出版的Wrox图书“Professional Data Warehousing with SQL Server 7.0 and OLAP Services”的作者。他也是Sams出版公司于1999年11月出版的“SQL Unleashed”一书的主要作者。他还为其他许多图书做过贡献，其中包括Wrox出版公司出版的“Beginning SQL Programming”、“Professional SQL Server 2000 Programming”和Sams出版公司出版的“SQL Server 7.0 Programming Unleashed”。他还参与过Macmillan和Wrox这两家出版公司的10多本图书的技术审读，技术范围涉及Oracle 8、SQL Server 7、Visual Basic 6和Visual Studio 98几个方面的数个主题。

Youness先生喜欢与妻子Nada及两个女儿Maya和Jenna一起外出旅行。他的业余爱好当中有游泳、打篮球、读书以及看体育节目。他的电子邮件地址是syouness@home.com。

我想把本书献给我的家庭成员Nada、Maya和Jenna。我还想感谢Wrox“家庭”，谢谢给了我写作和跟你们合作的机会。