



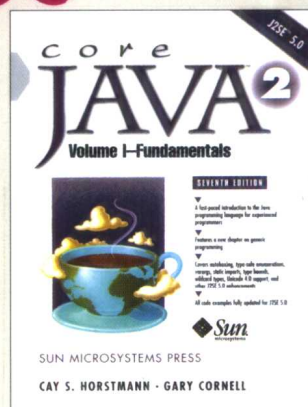
Core Java 2, Volume I — Fundamentals, Seventh Edition

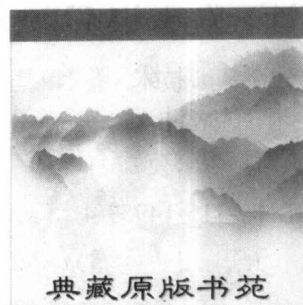
美] Cay S. Horstmann Gary Cornell 著

Java 2 核心技术 卷 I: 基础知识 (第 7 版)(英文版)



- 为有经验的程序员提供 Java 程序设计语言的快速指南。
- 特别增加新的一章介绍泛型程序设计。
- 涵盖自动包、类型安全枚举、变更参数列表、静态导入、类型限定、通配符类型、Unicode 4.0 支持以及其他的 J2SE 5.0 增强功能。
- 全部示例代码均针对 J2SE 5.0 进行了全面更新。





Java 2 核心技术 卷 I: 基础知识

(第7版)(英文版)

Core Java 2, Volume I - Fundamentals, Seventh Edition



人民邮电出版社

图书在版编目 (CIP) 数据

Java 2 核心技术. 卷 I, 基础知识: 第 7 版/ (美) 霍斯特曼 (Horstmann, C. S.), (美) 康奈尔 (Cornell, G.) 著. —北京: 人民邮电出版社, 2006.7
(典藏原版书苑)

ISBN 7-115-14953-4

I. J... II. ①霍... ②康... III. JAVA 语言—程序设计—英文 IV. TP312

中国版本图书馆 CIP 数据核字 (2006) 第 071841 号

版 权 声 明

Original edition, entitled CORE JAVA™ 2, VOLUME I – FUNDAMENTALS, 7th Edition, 0131482025 by Horstmann, Cay; Cornell, Gary, published by Pearson Education, Inc, publishing as Prentice Hall, Copyright © 2005 Sun Microsystems, Inc.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

China edition published by PEARSON EDUCATION ASIA LTD., and POSTS & TELECOMMUNICATIONS PRESS Copyright © 2006.

This edition is manufactured in the People's Republic of China, and is authorized for sale only in People's Republic of China excluding Hong Kong, Macau and Taiwan.

仅限于中华人民共和国境内 (不包括中国香港、澳门特别行政区和中国台湾地区) 销售。

本书封面贴有 Pearson Education (培生教育出版集团) 激光防伪标签。无标签者不得销售。

典藏原版书苑

Java 2 核心技术 卷 I: 基础知识 (第 7 版) (英文版)

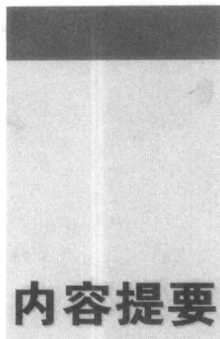
- ◆ 著 [美] Cay S. Horstmann Gary Cornell
责任编辑 李 际
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京顺义振华印刷厂印刷
新华书店总店北京发行所经销
- ◆ 开本: 800×1000 1/16
印张: 49
字数: 1 052 千字 2006 年 7 月第 1 版
印数: 1–4 000 册 2006 年 7 月北京第 1 次印刷

著作权合同登记号 图字: 01-2006-3658 号

ISBN 7-115-14953-4/TP · 5525

定价: 85.00 元

读者服务热线: (010)67132705 印装质量热线: (010)67129223



内容提要

本书是 Java 技术经典参考书，多年畅销不衰。第 7 版在保留以前版本风格的基础上，涵盖 Java 2 开发平台标准版 J2SE 5.0 的基础知识，主要包括面向对象程序设计、反射与代理、接口与内部类、事件监听器模型、使用 Swing UI 工具箱进行图形用户界面设计、异常处理、流输入 / 输出和对象序列化、泛型程序设计等。

本书内容翔实、深入浅出，附有大量程序实例，极具实用价值，是 Java 初学者和 Java 程序员的必备参考书。

Acknowledgments

Writing a book is always a monumental effort, and rewriting doesn't seem to be much easier, especially with continuous change in Java technology. Making a book a reality takes many dedicated people, and it is my great pleasure to acknowledge the contributions of the entire Core Java team.

A large number of individuals at Prentice Hall PTR and Sun Microsystems Press provided valuable assistance, but they managed to stay behind the scenes. I'd like them all to know how much I appreciate their efforts. As always, my warm thanks go to my editor, Greg Doench of Prentice Hall PTR, for steering the book through the writing and production process, and for allowing me to be blissfully unaware of the existence of all those folks behind the scenes. I am grateful to Vanessa Moore for the excellent production support. My thanks also to my coauthor of earlier editions, Gary Cornell, who has since moved on to other ventures.

Thanks to the many readers of earlier editions who reported embarrassing errors and made lots of thoughtful suggestions for improvement. I am particularly grateful to the excellent reviewing team that went over the manuscript with an amazing eye for detail and saved me from many more embarrassing errors.

Reviewers of this and earlier editions include Chuck Allison (Contributing Editor, *C/C++ Users Journal*), Alec Beaton (PointBase, Inc.), Joshua Bloch (Sun Microsystems), David Brown, Dr. Nicholas J. De Lillo (Manhattan College), Rakesh Dhoopar (Oracle), David Geary (Sabreware), Angela Gordon (Sun Microsystems), Dan Gordon (Sun Microsystems), Rob Gordon, Cameron Gregory (olabs.com), Marty Hall (The Johns Hopkins University Applied Physics Lab), Vincent Hardy (Sun Microsystems), Vladimir Ivanovic (PointBase), Jerry Jackson (ChannelPoint Software), Tim Kimmel (Preview Systems), Chris Laffra, Charlie Lai (Sun Microsystems), Doug Langston, Doug Lea (SUNY Oswego), Gregory Longshore, Bob Lynch (Lynch Associates), Mark Morrissey (The Oregon Graduate Institute), Mahesh Neelakanta (Florida Atlantic University), Paul Phillion, Blake Ragsdell, Stuart Reges (University of Arizona), Peter Sander (ESSI University, Nice, France), Paul Sevinc (Teamup AG), Devang Shah (Sun Microsystems), Bradley A. Smith, Christopher Taylor, Luke Taylor (Valtech), George Thiruvathukal, Kim Topley (author of *Core JFC*), Janet Traub, Peter van der Linden (Sun Microsystems), Burt Walsh, Corky Cartwright, Frank Cohen (PushToTest), Chris Crane (devXsolution), Bill Higgins (IBM), Hang Lau (McGill University), Angelika Langer, Mark Lawrence, Dr. Paul Sanghera (San Jose State University and Brooks College), and Steve Stelling (Sun Microsystems).

Cay Horstmann
San Francisco, 2004

前言

致读者

1995 年底, Java 程序设计语言在 Internet 舞台刚一亮相便名声鹊起。其原因在于它有望成为连接用户与信息的“万能胶”, 而不论该信息来自 Web 服务器、数据库、信息提供商, 还是任何其他的信息源。事实上, 就发展前景而言, Java 的地位是独一无二的。它是一种可以完全信赖的程序设计语言, 获得了除微软公司之外的所有厂商的认可。其固有的可靠性与安全性不仅令 Java 程序员放心, 也令使用 Java 程序的用户放心。Java 内建了对网络编程、数据库连接、多线程等高级程序设计任务的支持。

1995 年以来, Sun Microsystems 公司已经发布了 Java 开发工具箱 (Java Development Kit, JDK) 的 6 个主要版本。在过去的 9 年中, 应用程序编程接口 (API) 已经从 200 个类扩展到超过 3000 个类, 并跨越了诸如用户界面构建、数据库管理、国际化、安全性以及 XML 处理等各个不同的领域。2004 年发布的 JDK 5.0 是自 Java 的最初版本之后更新最显著的版本。

本书是《Java 2 核心技术》第 7 版的卷 I。自《Java 2 核心技术》出版以来, 每个新版本都尽可能跟上 Java 开发工具箱发展的步伐, 而且每一版我们都重写部分内容以适应 Java 的最新特性。在这一版中, 将极力展示泛型集合、增强的 for 循环以及一些令人振奋的 JDK 5.0 特性。

和本书的前几版一样, 我们仍然把读者群定位在那些打算将 Java 应用到实际项目中去
的严谨的程序设计人员。我们仍然确保本书不会出现令人沮丧的文字以及莫名其妙的字

符。本书假设读者是具有程序设计语言坚实背景知识的程序设计人员。然而，你可以不必了解 C++或是面向对象程序设计。根据我们收到的本书前几版的反馈意见，我们确信具有使用 Visual Basic、C 以及 COBOL 经历的程序员可以顺利阅读本书（甚至不需要具有建立 Windows、UNIX 或 Macintosh 图形用户界面的任何经验）。

我们假定读者想要：

- 编写实际的代码来解决实际的问题；
- 不希望本书中全部是玩具式样例（如烤面包机、水果或是动物园里的动物）；

在本书中，我们用大量的示例代码演示所讨论的每一种语言和库的特性。我们有意使用简单的示例程序以突出重点，然而，它们中的大部分既不是赝品也没有偷工减料。它们将成为读者自己编写代码的良好开端。

我们假设读者希望（甚至渴望）学习 Java 的所有高级特性。我们将会详细介绍下面内容：

- 面向对象程序设计；
- 反射与代理；
- 接口与内部类；
- 事件监听器模型；
- 使用 Swing UI 工具箱进行图形用户界面设计；
- 异常处理；
- 流输入/输出和对象序列化；
- 泛型程序设计。

随着 Java 类库的爆炸式增长，一本书无论如何也不可能涵盖真正的程序员需要了解的全部 Java 特性。因此，我们决定将本书分为两卷。卷 I（即本书）集中介绍 Java 语言的基本概念以及用户界面编程的基础。卷 II 涉及企业特性以及高级的用户界面编程，其中包括下列内容：

- 多线程；
- 分布式对象；
- 数据库；
- 高级 GUI 组件；

- 本地方法;
- XML 处理;
- 网络编程;
- 集合类;
- 高级图形;
- 国际化;
- JavaBeans。

在编写本书的过程中,难免会出现错误和不准确之处,我们很想知道这些错误。当然,我们也希望同一个问题只被告知一次。我们在网页 <http://www.horstmann.com/corejava.html> 中以列表的形式给出了常见问题、bug 修正和出错位置。在 Bug list 页的最后(这么放置是为了确保你会阅读这一页),提供了用来报告 bug 并提出改进意见的表单。如果我们没有回答每一个问题或没有及时回复,请不要失望。我们会认真阅读所有的 E-mail,感谢你的建议使本书后续版本更清晰、更有指导价值。

我们希望你发现本书生动、有趣且有助于 Java 程序设计。

关于本书

第 1 章概述 Java 与其他程序设计语言不同的性能,解释这种语言的设计初衷,以及在哪些方面达到预期的效果。然后简要叙述 Java 诞生和发展的历史。

第 2 章详细论述如何下载和安装 JDK 以及本书的程序示例。然后,通过编译和运行 3 个典型的 Java 程序(一个控制台应用、一个通行应用、一个 applet),指导读者使用简易的 JDK、可启用 Java (Java-enabled) 的文本编辑器以及一个 Java IDE。

第 3 章开始讨论 Java 语言。这一章涉及的基础知识有变量、循环以及简单的函数。对于 C 或 C++ 程序员来说,学习这一章的内容将会一帆风顺,因为这些语言特性的语法本质上与 C 语言相同。而对于没有 C 编程背景,但使用过其他编程语言(如 Visual Basic)的程序员来说,仔细阅读这一章是非常必要的。

面向对象程序设计(Object-Oriented Programming, OOP)是当今程序设计的主流,而 Java 是完全面向对象的。第 4 章介绍面向对象的两个基本元素中最重要元素——封装,以及 Java 语言实现封装的机制,即类与方法。除了 Java 语言规则之外,我们还对如何正

确地进行 OOP 设计给出了忠告。最后介绍 `javadoc` 工具, 它将代码注释转换为超链接的网页。熟悉 C++ 的程序员可以快速地浏览这一章。而没有面向对象编程背景的程序员应在进一步学习 Java 之前花一些时间了解 OOP 的有关概念。

类与封装仅仅是 OOP 中的一部分, 第 5 章介绍另一部分——继承。继承使程序员可以使用现有的类, 并根据需要进行修改, 这是 Java 程序设计的基础。Java 中的继承机制与 C++ 的继承机制十分相似。C++ 程序员只须关注两种语言的不同之处即可。

第 6 章展示如何使用 Java 的接口。接口可以使你的理解超越第 5 章的简单继承模型。掌握接口的使用可以获得 Java 完全面向对象程序设计的能力。我们还将介绍 Java 的一个很有用的技术特性——内部类, 它可以使代码更清晰、更简洁。

第 7 章开始详细讨论应用程序设计。我们将展示如何制作窗口、如何在窗口中绘图、如何用几何图形绘画、如何用多种字体格式化文本以及如何显示图像。

第 8 章详细讨论 AWT (abstract window toolkit, 抽象窗口工具箱) 的事件模型。我们将介绍如何编写代码来响应诸如鼠标点击或敲击键盘等事件。同时, 我们还将介绍如何处理基本的 GUI 元素, 如按钮和面板。

第 9 章详细讨论 Swing GUI 工具箱。Swing 工具箱允许你建立一个跨平台的图形用户界面。我们将介绍如何建立各种各样的按钮、文本组件、边界、滑块、列表框、菜单以及对话框等。一些更高级的组件将在卷 II 中讨论。

前 9 章讲述编写 applet 需要的基础知识。applet 是可以嵌入网页中的微型程序。第 10 章的主题就是 applet。在这一章中, 我们将展示一些有用且有趣的 applet, 但更重要的是将 applet 看作部署程序的一种方法。然后, 我们将描述如何将应用程序打包到 JAR 文件中, 以及如何使用 Java 的 Web Start 机制在 Internet 上发布应用程序。最后, 解释 Java 程序在部署之后如何存储和得到配置信息。

第 11 章讨论异常处理, 即 Java 的健壮机制, 它用于处理调试好的程序可能出现的意外情况。例如, 网络连接在文件下载过程中可能断开、磁盘可能填满等。异常提供了一种将正常的处理代码与错误处理代码分开的有效的方法。当然, 即使程序具有处理所有异常情况的功能, 但依然可能无法按预计的方式工作。这一章的后半部分给出了大量实用的调试技巧。最后, 我们讲述如何使用各种程序完成一个示例程序, 这些工具包括 JDB 调试器、集成开发环境的调试器、剖析器、代码覆盖率测试工具以及 AWT 自动机。

第 12 章的主题是输入与输出处理。Java 中所有的 I/O 都是通过所谓的流来处理的。流可以让程序员用一致的方式与任何数据源 (如文件、网络连接或内存块) 进行通信。我们

将详细介绍大量读取器和写入器类，使用这些类可以方便地处理 Unicode 代码。另外，我们介绍使用对象序列化机制可能会出现的一些情况，该机制使得对象的存储与加载非常容易。最后，我们还给出已经添加到 JDK 1.4 中的支持高级的、更有效的文件操作的“NEW I/O”类以及正则表达式库。

本书的最后一章是泛型程序设计概述。泛型程序设计是 JDK 5.0 的重要改进，它使程序拥有更好的可阅读性和安全性。在这一章里将展示如何使用强类型机制，而舍弃不安全的强制类型转换。

附录 A 列出了 Java 语言的保留字。

附录 B 介绍如何修改代码示例，使之可以在旧的编译器（JDK 1.4）下编译。

约定

本书使用以下图标表示特定内容。



C++ NOTE: 在本书中有许多关于 C++ 的注释用来解释 Java 与 C++ 的不同。对于没有 C++ 编程背景，或者不擅长 C++ 编程的读者，可以跳过这些注释。



NOTE: 这个图标表示为正文提供的“注意”信息。



TIP: 这个图标表示为正文提供的“提示”信息。



CAUTION: 这个图标表示为正文提供的“警告”信息。

应用程序编程接口（API）

Java 带有一个很大的程序设计库，即应用程序编程接口（Application Programming Interface, API）。第一次调用一个 API 时，我们将会在这一节的结尾给出一个概要描述，并标有 API 图标。这些描述十分通俗易懂，希望能够比联机 API 文档提供更多的信息。我们在每一个 API 注释特性上都标记了版本号，以提示那些不希望使用 Java “风险”版本的读者。

程序源代码按照以下格式列出:

Example 2-4: WelcomeApplet.java

示例代码

本书的网站 <http://www.phptr.com/corejava> 以压缩的形式提供了书中所有示例代码。你可以用相应的解压缩程序或者用 Java 开发工具箱中的 jar 实用程序展开这个文件。有关安装 Java 开发工具箱和示例程序的详细信息请参考第 2 章。

Contents

Chapter 1

An Introduction to Java 1

Java as a Programming Platform	1
The Java "White Paper" Buzzwords	2
<i>Simple</i>	2
<i>Object Oriented</i>	3
<i>Distributed</i>	3
<i>Robust</i>	4
<i>Secure</i>	4
<i>Architecture Neutral</i>	5
<i>Portable</i>	5
<i>Interpreted</i>	5
<i>High Performance</i>	6
<i>Multithreaded</i>	6
<i>Dynamic</i>	6
Java and the Internet	7
A Short History of Java	8
Common Misconceptions About Java	11



Chapter 2

The Java Programming Environment 15

Installing the Java Development Kit	15
<i>Downloading the JDK</i>	15
<i>Setting the Execution Path</i>	16
<i>Installing the Library Source and Documentation</i>	17
<i>Installing the Core Java Program Examples</i>	18
<i>Navigating the Java Directories</i>	18
Choosing a Development Environment	19
Using the Command-Line Tools	20
<i>Troubleshooting Hints</i>	21
Using an Integrated Development Environment	22
<i>Locating Compilation Errors</i>	25
Compiling and Running Programs from a Text Editor	25
Running a Graphical Application	27
Building and Running Applets	30

Chapter 3

Fundamental Programming Structures in Java 35

A Simple Java Program	35
Comments	38
Data Types	39
<i>Integers</i>	39
<i>Floating-Point Types</i>	40
<i>The char Type</i>	41
<i>The boolean Type</i>	42
Variables	42
<i>Initializing Variables</i>	43
<i>Constants</i>	44
Operators	45
<i>Increment and Decrement Operators</i>	46
<i>Relational and boolean Operators</i>	46
<i>Bitwise Operators</i>	47
<i>Mathematical Functions and Constants</i>	47
<i>Conversions Between Numeric Types</i>	48
<i>Casts</i>	49
<i>Parentheses and Operator Hierarchy</i>	50
<i>Enumerated Types</i>	51
Strings	51
<i>Code Points and Code Units</i>	51
<i>Substrings</i>	52
<i>String Editing</i>	52
<i>Concatenation</i>	53
<i>Testing Strings for Equality</i>	54
<i>Reading the On-Line API Documentation</i>	56
Input and Output	58
<i>Reading Input</i>	58
<i>Formatting Output</i>	61



Control Flow	65
Block Scope	65
Conditional Statements	65
Loops	68
Determinate Loops	72
Multiple Selections—The switch Statement	75
Statements That Break Control Flow	77
Big Numbers	79
Arrays	81
The “for each” Loop	82
Array Initializers and Anonymous Arrays	82
Array Copying	83
Command-Line Parameters	84
Array Sorting	85
Multidimensional Arrays	88
Ragged Arrays	90

Chapter 4

Objects and Classes 93

Introduction to Object-Oriented Programming	93
The Vocabulary of OOP	95
Objects	95
Relationships Between Classes	96
OOP Contrasted with Traditional Procedural Programming Techniques	98
Using Predefined Classes	99
Objects and Object Variables	99
The GregorianCalendar Class of the Java Library	102
Mutator and Accessor Methods	103
Defining Your Own Classes	108
An Employee Class	108
Use of Multiple Source Files	111
Dissecting the Employee Class	111
First Steps with Constructors	112
Implicit and Explicit Parameters	113
Benefits of Encapsulation	114
Class-Based Access Privileges	116
Private Methods	116
Final Instance Fields	117
Static Fields and Methods	117
Static Fields	117
Constants	118
Static Methods	118
Factory Methods	119
The main Method	120
Method Parameters	122
Object Construction	127
Overloading	127
Default Field Initialization	128
Default Constructors	128
Explicit Field Initialization	129



Parameter Names	130
Calling Another Constructor	130
Initialization Blocks	131
Object Destruction and the <i>finalize</i> Method	135
Packages	135
Class Importation	136
Static Imports	137
Addition of a Class into a Package	137
How the Virtual Machine Locates Classes	140
Package Scope	143
Documentation Comments	144
Comment Insertion	144
Class Comments	144
Method Comments	145
Field Comments	146
General Comments	146
Package and Overview Comments	147
Comment Extraction	147
Class Design Hints	148

Chapter 5

Inheritance 151

Classes, Superclasses, and Subclasses	151
Inheritance Hierarchies	157
Polymorphism	158
Dynamic Binding	159
Preventing Inheritance: Final Classes and Methods	161
Casting	162
Abstract Classes	164
Protected Access	168
Object: The Cosmic Superclass	169
The <i>equals</i> Method	170
Equality Testing and Inheritance	171
The <i>hashCode</i> Method	173
The <i>toString</i> Method	174
Generic Array Lists	179
Accessing Array List Elements	181
Compatibility Between Typed and Raw Array Lists	185
Object Wrappers and Autoboxing	186
Methods with a Variable Number of Parameters	188
Reflection	189
The <i>Class</i> Class	190
Using Reflection to Analyze the Capabilities of Classes	193
Using Reflection to Analyze Objects at Run Time	197
Using Reflection to Write Generic Array Code	201
Method Pointers!	204
Enumeration Classes	207
Design Hints for Inheritance	209



Chapter 6

Interfaces and Inner Classes 211

- Interfaces 211
 - Properties of Interfaces* 216
 - Interfaces and Abstract Classes* 217
- Object Cloning 218
- Interfaces and Callbacks 223
- Inner Classes 226
 - Use of an Inner Class to Access Object State* 227
 - Special Syntax Rules for Inner Classes* 230
 - Are Inner Classes Useful? Actually Necessary? Secure?* 230
 - Local Inner Classes* 232
 - Anonymous Inner Classes* 234
 - Static Inner Classes* 237
- Proxies 239
 - Properties of Proxy Classes* 243

Chapter 7

Graphics Programming 245

- Introducing Swing 245
- Creating a Frame 249
- Positioning a Frame 251
- Displaying Information in a Panel 256
- Working with 2D Shapes 260
- Using Color 267
 - Filling Shapes* 270
- Using Special Fonts for Text 272
- Doing More with Images 278

Chapter 8

Event Handling 285

- Basics of Event Handling 285
 - Example: Handling a Button Click* 288
 - Becoming Comfortable with Inner Classes* 292
 - Turning Components into Event Listeners* 294
 - Example: Changing the Look and Feel* 296
 - Example: Capturing Window Events* 299
- The AWT Event Hierarchy 302
- Semantic and Low-Level Events in the AWT 304
 - Event Handling Summary* 305
- Low-Level Event Types 307
 - Keyboard Events* 307
 - Mouse Events* 312
 - Focus Events* 320
- Actions 323
- Multicasting 330
- Implementing Event Sources 333



Chapter 9

User Interface Components with Swing 339

- The Model-View-Controller Design Pattern 339
 - A Model-View-Controller Analysis of Swing Buttons* 344
- Introduction to Layout Management 345
 - Border Layout* 347
 - Panels* 348
 - Grid Layout* 350
- Text Input 354
 - Text Fields* 354
 - Labels and Labeling Components* 356
 - Change Tracking in Text Fields* 357
 - Password Fields* 361
 - Formatted Input Fields* 362
 - Text Areas* 376
- Choice Components 380
 - Checkboxes* 380
 - Radio Buttons* 382
 - Borders* 386
 - Combo Boxes* 390
 - Sliders* 393
 - The JSpinner Component* 398
- Menus 406
 - Menu Building* 406
 - Icons in Menu Items* 409
 - Checkbox and Radio Button Menu Items* 410
 - Pop-Up Menus* 411
 - Keyboard Mnemonics and Accelerators* 413
 - Enabling and Disabling Menu Items* 415
 - Toolbars* 419
 - Tooltips* 421
- Sophisticated Layout Management 424
 - Box Layout* 426
 - The Grid Bag Layout* 430
 - The Spring Layout* 440
 - Using No Layout Manager* 449
 - Custom Layout Managers* 449
 - Traversal Order* 453
- Dialog Boxes 455
 - Option Dialogs* 455
 - Creating Dialogs* 465
 - Data Exchange* 469
 - File Dialogs* 475
 - Color Choosers* 486