

《Java核心技术卷I：(第9版·英文版)》

图书基本信息

书名：《Java核心技术卷I：(第9版·英文版)》

13位ISBN编号：9787115319470

10位ISBN编号：7115319472

出版时间：2013-7-1

出版社：人民邮电出版社

作者：[美]Cay S. Horstmann,Gary Cornell

页数：974

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu111.com

内容概要

《Java核心技术 卷I：基础知识(第9版·英文版)(上、下册)》是经典的《Java核心技术 卷I：基础知识》的最新版。这一版针对Java SE 7平台进行了全面更新，以反映Java SE 7的特性。

书中囊括了Java的全部基础知识，提供了大量完整且具有实际意义的应用示例，详细介绍了Java语言基础、面向对象编程、反射与代理、接口与内部类、事件监听器模型、使用Swing GUI工具进行图形用户界面程序设计、打包应用程序、异常处理、登录与调试、泛型编程、集合框架、多线程等内容。

《Java核心技术 卷I：基础知识(第9版·英文版)(上、下册)》适合想将Java应用于实际项目的软件开发人员、高等院校教师和学生参考阅读。

作者简介

Cay S. Horstmann 是Scala for the Impatient的作者，还与人合著了Core JavaServer Faces。他是圣何塞州立大学计算机科学专业的教授，还是一名Java Champion，并经常在很多开发者大会上演讲。Gary Cornell 在编程方面拥有20多年的写作和教育经验。他是Apress的创始人之一，编写了很多与开发相关的畅销书，是Jolt大奖的获奖者之一，还荣获过Visual Basic Magazine的读者选择奖。

书籍目录

Contents	
Preface	v
Acknowledgments	ix
Chapter 1: An Introduction to Java	1
1.1 Java As a Programming Platform	1
1.2 The Java “ White Paper ” Buzzwords	2
1.2.1 Simple	3
1.2.2 Object-Oriented	4
1.2.3 Network-Savvy	4
1.2.4 Robust	5
1.2.5 Secure	5
1.2.6 Architecture-Neutral	6
1.2.7 Portable	7
1.2.8 Interpreted	7
1.2.9 High-Performance	8
1.2.10 Multithreaded	8
1.2.11 Dynamic	8
1.3 Java Applets and the Internet	9
1.4 A Short History of Java	10
1.5 Common Misconceptions about Java	13
Chapter 2: The Java Programming Environment	17
2.1 Installing the Java Development Kit	18
2.1.1 Downloading the JDK	18
2.1.2 Setting the Executable Path	20
2.1.3 Installing the Library Source and Documentation	22
2.1.4 Installing the Core Java Program Examples	23
2.1.5 Navigating the Java Directories	24
2.2 Choosing a Development Environment	24
2.3 Using the Command-Line Tools	25
2.3.1 Troubleshooting Hints	27
2.4 Using an Integrated Development Environment	28
2.4.1 Locating Compilation Errors	32
2.5 Running a Graphical Application	33
2.6 Building and Running Applets	36
Chapter 3: Fundamental Programming Structures in Java	41
3.1 A Simple Java Program	42
3.2 Comments	45
3.3 Data Types	46
3.3.1 Integer Types	47
3.3.2 Floating-Point Types	48
3.3.3 The char Type	49
3.3.4 The boolean Type	51
3.4 Variables	52
3.4.1 Initializing Variables	53
3.4.2 Constants	54
3.5 Operators	55
3.5.1 Increment and Decrement Operators	56

3.5.2	Relational and boolean Operators	57
3.5.3	Bitwise Operators	58
3.5.4	Mathematical Functions and Constants	59
3.5.5	Conversions between Numeric Types	60
3.5.6	Casts	61
3.5.7	Parentheses and Operator Hierarchy	62
3.5.8	Enumerated Types	63
3.6	Strings	64
3.6.1	Substrings	64
3.6.2	Concatenation	64
3.6.3	Strings Are Immutable	65
3.6.4	Testing Strings for Equality	67
3.6.5	Empty and Null Strings	68
3.6.5	Code Points and Code Units	68
3.6.6	The String API	69
3.6.7	Reading the Online API Documentation	72
3.6.8	Building Strings	74
3.7	Input and Output	76
3.7.1	Reading Input	76
3.7.2	Formatting Output	79
3.7.3	File Input and Output	84
3.8	Control Flow	86
3.8.1	Block Scope	86
3.8.2	Conditional Statements	87
3.8.3	Loops	91
3.8.4	Determinate Loops	95
3.8.5	Multiple Selections—The switch Statement	99
3.8.6	Statements That Break Control Flow	102
3.9	Big Numbers	105
3.10	Arrays	107
3.10.1	The “for each” Loop	109
3.10.2	Array Initializers and Anonymous Arrays	110
3.10.3	Array Copying	111
3.10.4	Command-Line Parameters	112
3.10.5	Array Sorting	113
3.10.6	Multidimensional Arrays	116
3.10.7	Ragged Arrays	120
Chapter 4:	Objects and Classes	125
4.1	Introduction to Object-Oriented Programming	126
4.1.1	Classes	127
4.1.2	Objects	128
4.1.3	Identifying Classes	129
4.1.4	Relationships between Classes	129
4.2	Using Predefined Classes	132
4.2.1	Objects and Object Variables	132
4.2.2	The GregorianCalendar Class of the Java Library	136
4.2.3	Mutator and Accessor Methods	138
4.3	Defining Your Own Classes	145
4.3.1	An Employee Class	145

4.3.2	Use of Multiple Source Files	148
4.3.3	Dissecting the Employee Class	149
4.3.4	First Steps with Constructors	150
4.3.5	Implicit and Explicit Parameters	152
4.3.6	Benefits of Encapsulation	153
4.3.7	Class-Based Access Privileges	156
4.3.8	Private Methods	156
4.3.9	Final Instance Fields	157
4.4	Static Fields and Methods	157
4.4.1	Static Fields	157
4.4.2	Static Constants	158
4.4.3	Static Methods	159
4.4.4	Factory Methods	161
4.4.5	The main Method	161
4.5	Method Parameters	164
4.6	Object Construction	171
4.6.1	Overloading	171
4.6.2	Default Field Initialization	172
4.6.3	The Constructor with No Arguments	173
4.6.4	Explicit Field Initialization	174
4.6.5	Parameter Names	175
4.6.6	Calling Another Constructor	176
4.6.7	Initialization Blocks	176
4.6.8	Object Destruction and the finalize Method	181
4.7	Packages	182
4.7.1	Class Importation	182
4.7.2	Static Imports	184
4.7.3	Addition of a Class into a Package	185
4.7.4	Package Scope	188
4.8	The Class Path	190
4.8.1	Setting the Class Path	192
4.9	Documentation Comments	193
4.9.1	Comment Insertion	193
4.9.2	Class Comments	194
4.9.3	Method Comments	195
4.9.4	Field Comments	196
4.9.5	General Comments	196
4.9.6	Package and Overview Comments	197
4.9.7	Comment Extraction	198
4.10	Class Design Hints	199
Chapter 5: Inheritance		203
5.1	Classes, Superclasses, and Subclasses	204
5.1.1	Inheritance Hierarchies	212
5.1.2	Polymorphism	213
5.1.3	Dynamic Binding	214
5.1.4	Preventing Inheritance: Final Classes and Methods	217
5.1.5	Casting	218
5.1.6	Abstract Classes	221
5.1.7	Protected Access	227

5.2	Object: The Cosmic Superclass	228
5.2.1	The equals Method	228
5.2.2	Equality Testing and Inheritance	230
5.2.3	The hashCode Method	234
5.2.4	The toString Method	236
5.3	Generic Array Lists	243
5.3.1	Accessing Array List Elements	246
5.3.2	Compatibility between Typed and Raw Array Lists	249
5.4	Object Wrappers and Autoboxing	251
5.5	Methods with a Variable Number of Parameters	254
5.6	Enumeration Classes	256
5.7	Reflection	258
5.7.1	The Class Class	259
5.7.2	A Primer on Catching Exceptions	261
5.7.3	Using Reflection to Analyze the Capabilities of Classes	263
5.7.4	Using Reflection to Analyze Objects at Runtime	269
5.7.5	Using Reflection to Write Generic Array Code	274
5.7.6	Invoking Arbitrary Methods	278
5.8	Design Hints for Inheritance	282
	Chapter 6: Interfaces and Inner Classes	285
6.1	Interfaces	286
6.1.1	Properties of Interfaces	292
6.1.2	Interfaces and Abstract Classes	294
6.2	Object Cloning	295
6.3	Interfaces and Callbacks	302
6.4	Inner Classes	305
6.4.1	Use of an Inner Class to Access Object State	307
6.4.2	Special Syntax Rules for Inner Classes	311
6.4.3	Are Inner Classes Useful? Actually Necessary? Secure?	312
6.4.4	Local Inner Classes	315
6.4.5	Accessing final Variables from Outer Methods	315
6.4.6	Anonymous Inner Classes	318
6.4.7	Static Inner Classes	322
6.5	Proxies	326
6.5.1	Properties of Proxy Classes	331
	Chapter 7: Graphics Programming	333
7.1	Introducing Swing	334
7.2	Creating a Frame	339
7.3	Positioning a Frame	342
7.3.1	Frame Properties	345
7.3.2	Determining a Good Frame Size	345
7.4	Displaying Information in a Component	350
7.5	Working with 2D Shapes	356
7.6	Using Color	365
7.7	Using Special Fonts for Text	369
7.8	Displaying Images	378
	Chapter 8: Event Handling	383
8.1	Basics of Event Handling	383
8.1.1	Example: Handling a Button Click	386

8.1.2	Becoming Comfortable with Inner Classes	391
8.1.3	Creating Listeners Containing a Single Method Call	394
8.1.4	Example: Changing the Look-and-Feel	395
8.1.5	Adapter Classes	399
8.2	Actions	403
8.3	Mouse Events	411
8.4	The AWT Event Hierarchy	419
8.4.1	Semantic and Low-Level Events	421
Chapter 9: User Interface Components with Swing		425
9.1	Swing and the Model-View-Controller Design Pattern	426
9.1.1	Design Patterns	426
9.1.2	The Model-View-Controller Pattern	428
9.1.3	A Model-View-Controller Analysis of Swing Buttons	432
9.2	Introduction to Layout Management	433
9.2.1	Border Layout	437
9.2.2	Grid Layout	439
9.3	Text Input	443
9.3.1	Text Fields	444
9.3.2	Labels and Labeling Components	446
9.3.3	Password Fields	447
9.3.4	Text Areas	448
9.3.5	Scroll Panes	449
9.4	Choice Components	452
9.4.1	Checkboxes	452
9.4.2	Radio Buttons	454
9.4.3	Borders	458
9.4.4	Combo Boxes	462
9.4.5	Sliders	466
9.5	Menus	473
9.5.1	Menu Building	473
9.5.2	Icons in Menu Items	476
9.5.3	Checkbox and Radio Button Menu Items	477
9.5.4	Pop-Up Menus	479
9.5.5	Keyboard Mnemonics and Accelerators	480
9.5.6	Enabling and Disabling Menu Items	483
9.5.7	Toolbars	488
9.5.8	Tooltips	490
9.6	Sophisticated Layout Management	492
9.6.1	The Grid Bag Layout	494
9.6.1.1	The gridx, gridy, gridwidth, and gridheight Parameters	496
9.6.1.2	Weight Fields	496
9.6.1.3	The fill and anchor Parameters	497
9.6.1.4	Padding	497
9.6.1.5	Alternative Method to Specify the gridx, gridy, idwidth, and gridheight Parameters	497
9.6.1.6	A Helper Class to Tame the Grid Bag Constraints	499
9.6.2	Group Layout	505
9.6.3	Using No Layout Manager	516
9.6.4	Custom Layout Managers	516
9.6.5	Traversal Order	521

9.7	Dialog Boxes	522
9.7.1	Option Dialogs	523
9.7.2	Creating Dialogs	533
9.7.3	Data Exchange	538
9.7.4	File Dialogs	545
9.7.5	Color Choosers	557
Chapter 10:	Deploying Applications and Applets	565
10.1	JAR Files	566
10.1.1	The Manifest	567
10.1.2	Executable JAR Files	568
10.1.3	Resources	569
10.1.4	Sealing	573
10.2	Java Web Start	574
10.2.1	The Sandbox	578
10.2.2	Signed Code	579
10.2.3	The JNLP API	582
10.3	Applets	591
10.3.1	A Simple Applet	591
10.3.1.1	Converting Applications to Applets	595
10.3.2	The applet HTML Tag and Its Attributes	596
10.3.3	The object Tag	599
10.3.4	Use of Parameters to Pass Information to Applets	600
10.3.5	Accessing Image and Audio Files	606
10.3.6	The Applet Context	607
10.3.6.1	Inter-Applet Communication	608
10.3.6.2	Displaying Items in the Browser	609
10.4	Storage of Application Preferences	610
10.4.1	Property Maps	611
10.4.2	The Preferences API	616
Chapter 11:	Exceptions, Assertions, Logging, and Debugging	625
11.1	Dealing with Errors	626
11.1.1	The Classification of Exceptions	628
11.1.2	Declaring Checked Exceptions	630
11.1.3	How to Throw an Exception	632
11.1.4	Creating Exception Classes	634
11.2	Catching Exceptions	635
11.2.1	Catching Multiple Exceptions	637
11.2.2	Rethrowing and Chaining Exceptions	639
11.2.3	The finally Clause	640
11.2.4	The Try-with-Resources Statement	644
11.2.5	Analyzing Stack Trace Elements	646
11.3	Tips for Using Exceptions	649
11.4	Using Assertions	653
11.4.1	Assertion Enabling and Disabling	654
11.4.2	Using Assertions for Parameter Checking	655
11.4.3	Using Assertions for Documenting Assumptions	656
11.5	Logging	657
11.5.1	Basic Logging	658
11.5.2	Advanced Logging	658

11.5.3	Changing the Log Manager Configuration	661
11.5.4	Localization	662
11.5.5	Handlers	663
11.5.6	Filters	667
11.5.7	Formatters	667
11.5.8	A Logging Recipe	668
11.6	Debugging Tips	677
11.7	Tips for Troubleshooting GUI Programs	682
11.7.1	Letting the AWT Robot Do the Work	686
11.8	Using a Debugger	690
Chapter 12: Generic Programming 697		
12.1	Why Generic Programming?	698
12.1.1	Who Wants to Be a Generic Programmer?	699
12.2	Defining a Simple Generic Class	700
12.3	Generic Methods	702
12.4	Bounds for Type Variables	704
12.5	Generic Code and the Virtual Machine	706
12.5.1	Translating Generic Expressions	708
12.5.2	Translating Generic Methods	708
12.5.3	Calling Legacy Code	711
12.6	Restrictions and Limitations	712
12.6.1	Type Parameters Cannot Be Instantiated with Primitive Types	712
12.6.2	Runtime Type Inquiry Only Works with Raw Types	712
12.6.3	You Cannot Create Arrays of Parameterized Types	713
12.6.4	Varargs Warnings	713
12.6.5	You Cannot Instantiate Type Variables	715
12.6.6	Type Variables Are Not Valid in Static Contexts of Generic Classes	717
12.6.7	You Cannot Throw or Catch Instances of a Generic Class	717
12.6.7.1	You Can Defeat Checked Exception Checking	718
12.6.8	Beware of Clashes after Erasure	720
12.7	Inheritance Rules for Generic Types	721
12.8	Wildcard Types	723
12.8.1	Supertype Bounds for Wildcards	725
12.8.2	Unbounded Wildcards	728
12.8.3	Wildcard Capture	728
12.9	Reflection and Generics	731
12.9.1	Using Class Parameters for Type Matching	732
12.9.2	Generic Type Information in the Virtual Machine	733
Chapter 13: Collections 741		
13.1	Collection Interfaces	741
13.1.1	Separating Collection Interfaces and Implementation	742
13.1.2	Collection and Iterator Interfaces in the Java Library	745
13.1.2.1	Iterators	745
13.1.2.2	Removing Elements	748
13.1.2.3	Generic Utility Methods	748
13.2	Concrete Collections	751
13.2.1	Linked Lists	752
13.2.2	Array Lists	762
13.2.3	Hash Sets	763

13.2.4	Tree Sets	767
13.2.5	Object Comparison	768
13.2.6	Queues and Deques	774
13.2.7	Priority Queues	776
13.2.8	Maps	777
13.2.9	Specialized Set and Map Classes	782
13.2.9.1	Weak Hash Maps	782
13.2.9.2	Linked Hash Sets and Maps	783
13.2.9.3	Enumeration Sets and Maps	785
13.2.9.4	Identity Hash Maps	785
13.3	The Collections Framework	787
13.3.1	Views and Wrappers	792
13.3.1.1	Lightweight Collection Wrappers	793
13.3.1.2	Subranges	794
13.3.1.3	Unmodifiable Views	794
13.3.1.4	Synchronized Views	796
13.3.1.5	Checked Views	796
13.3.1.6	A Note on Optional Operations	797
13.3.2	Bulk Operations	799
13.3.3	Converting between Collections and Arrays	800
13.4	Algorithms	801
13.4.1	Sorting and Shuffling	802
13.4.2	Binary Search	805
13.4.3	Simple Algorithms	806
13.4.4	Writing Your Own Algorithms	808
13.5	Legacy Collections	810
13.5.1	The Hashtable Class	810
13.5.2	Enumerations	810
13.5.3	Property Maps	811
13.5.4	Stacks	812
13.5.5	Bit Sets	813
13.5.5.1	The “Sieve of Eratosthenes” Benchmark	814
Chapter 14:	Multithreading	819
14.1	What Are Threads?	820
14.1.1	Using Threads to Give Other Tasks a Chance	827
14.2	Interrupting Threads	833
14.3	Thread States	836
14.3.1	New Threads	836
14.3.2	Runnable Threads	836
14.3.3	Blocked and Waiting Threads	837
14.3.4	Terminated Threads	839
14.4	Thread Properties	839
14.4.1	Thread Priorities	840
14.4.2	Daemon Threads	841
14.4.3	Handlers for Uncaught Exceptions	841
14.5	Synchronization	843
14.5.1	An Example of a Race Condition	843
14.5.2	The Race Condition Explained	848
14.5.3	Lock Objects	850

14.5.4	Condition Objects	854
14.5.5	The synchronized Keyword	859
14.5.6	Synchronized Blocks	864
14.5.7	The Monitor Concept	865
14.5.8	Volatile Fields	866
14.5.9	Final Variables	867
14.5.10	Atomics	868
14.5.11	Deadlocks	868
14.5.12	Thread-Local Variables	871
14.5.13	Lock Testing and Timeouts	873
14.5.14	Read/Write Locks	874
14.5.15	Why the stop and suspend Methods Are Deprecated	875
14.6	Blocking Queues	877
14.7	Thread-Safe Collections	886
14.7.1	Efficient Maps, Sets, and Queues	886
14.7.2	Copy on Write Arrays	888
14.7.3	Older Thread-Safe Collections	888
14.8	Callables and Futures	890
14.9	Executors	895
14.9.1	Thread Pools	896
14.9.2	Scheduled Execution	900
14.9.3	Controlling Groups of Tasks	901
14.9.4	The Fork-Join Framework	902
14.10	Synchronizers	905
14.10.1	Semaphores	906
14.10.2	Countdown Latches	907
14.10.3	Barriers	907
14.10.4	Exchangers	908
14.10.5	Synchronous Queues	908
14.11	Threads and Swing	909
14.11.1	Running Time-Consuming Tasks	910
14.11.2	Using the Swing Worker	915
14.11.3	The Single-Thread Rule	923
	Appendix: Java Keywords	925
	Index	929

编辑推荐

Java经典图书最新版，前一版曾获得第13届Jolt生产效率大奖。Java平台的权威指南，新版做了全面更新，体现了Java SE 7的变化，将对Java SE 7新特性的完整描述融入对Java基本概念的精辟阐述中。Java核心技术 卷I：基础知识(第9版·英文版)(上、下册)专为做实际项目的程序员编写，是一本真实可信、不偏不倚且简单直接的Java教程，书中使用了全面测试过的代码示例来阐述关键的Java语言与库的特性，体现了最佳的编程实践。Java核心技术 卷I：基础知识(第9版·英文版)(上、下册)两位作者均是业内权威人物，有丰富的写作和实战经验。

精彩短评

- 1、算是勉强撸一遍。。
- 2、里面内容很多，讲的很详细。很多地方还特别讲解了和C++的区别。
- 3、书不错，适合系统学习，就是全英文，有些不容易理解，没有英文和C++功底慎购
- 4、书的印刷没的说，内容慢慢啃了
- 5、这本书是有删减还是有部分内容漏印了？上册第200页没有第五条，第四条完了直接第六条了。
- 6、讲的东西很详实，印刷都很棒，今年新版的书。对于初学JAVA的码农应该是很好的。
- 7、经典书，值得一看。同时提高了英文阅读能力
- 8、经典的书，等看完这本还打算买卷II。

精彩书评

1、在上册的4.10 Class Design Hints小节，缺少第5部分。（P200）

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：www.tushu111.com