



图灵计算机科学丛书



Sun公司指定教材

Java™ 教程

(第4版)

The Java™ Tutorial: A Short Course on the Basics
Fourth Edition

[美] Sharon Zakhour Scott Hommel Jacob Royal 著
Isaac Rabinovitch Tom Risser Mark Hoeber
马朝晖 等译



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵计算机科学丛书

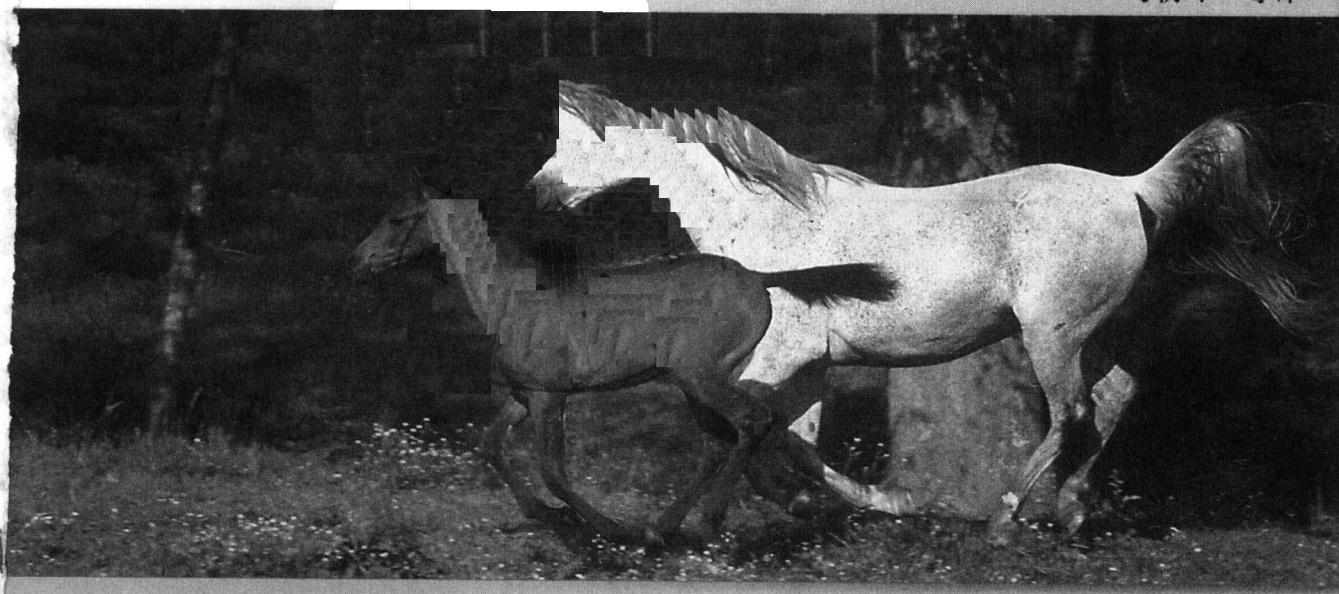
Java™ 教程

(第4版)

The Java™ Tutorial: A Short Course on the Basics

Fourth Edition

[美] Sharon Zakhour Scott Hommel Jacob Royal 著
Isaac Rabinovitch Tom Risser Mark Hoeber
马朝晖 等译



人民邮电出版社
北京

图书在版编目 (CIP) 数据

Java™ 教程: 第 4 版 / (美) 扎克霍尔 (Zakhour, S.) 等著; 马朝晖等译.

—北京: 人民邮电出版社, 2007.9

(图灵计算机科学丛书)

ISBN 978-7-115-16336-3

I. J... II. ①扎...②马... III. JAVA 语言—程序设计—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2007) 第 081151 号

内 容 提 要

本书是Sun公司指定Java教程的最新版。书中除了讲解面向对象的编程概念、语言基础、类和对象、接口和继承、包、数字和字符串、异常、基本I/O、集合、并发、Swing、把程序打包到JAR文件中、applet之外,还介绍了Java Web Start、平台环境和正则表达式,同时涵盖了Java 5.0和6.0引入的新特性,比如泛型和并发、注解。附录B中还给出如何准备Java编程语言认证考试方面的内容。此外,本书配套的网上资源还包含本书的示例以及问题和练习的答案、在线Java SE指南、Java SE开发工具包 (Java SE Development Kit, JDK)、相应的API规范和指导文档等。

本书既适合作为高校相关专业Java程序设计语言课程的教材,也适合各种层次的Java程序员阅读。

图灵计算机科学丛书

Java™ 教程 (第 4 版)

◆ 著 [美] Sharon Zakhour Scott Hommel Jacob Royal
Isaac Rabinovitch Tom Risser Mark Hoeber

译 马朝晖 等

责任编辑 杨海玲

◆ 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: 787 × 1092 1/16

印张: 27

字数: 746 千字

2007 年 9 月第 1 版

印数: 1—4 000 册

2007 年 9 月北京第 1 次印刷

著作权合同登记号 图字: 01-2005-5228 号

ISBN 978-7-115-16336-3/TP

定价: 55.00 元

读者服务热线: (010) 88593802 印装质量热线: (010) 67129223

序

能为《Java教程》的新版写序，我感到非常高兴。

Java平台显著的特征之一就是其文档的质量。这包括教程，JavaDoc方式的API详尽文档，还有讲述Java的很多书籍。有人可能质疑，文档资源的质量对Java平台的广泛采用而言是否具有和语言以及库本身的质量一样的意义。其实，它们是相辅相成的，其质量的好坏都非常关键。

数百万开发人员把他们的职业生涯投入到了Java平台形成的生态系统中。现在有无数的Java标准、API和工具。要学的东西太多了！而且，要了解的信息的数量还在不断增长。虽然我们这些软件专业人员应该感谢这种增长，但对开发人员来说这也是一种挑战。我希望本书能帮助读者迎接这一挑战，并且使读者掌握以至精通Java平台的过程容易一些。

就像Edsger Dijkstra¹多年前说的那样，是否能成为真正“合格的程序员”取决于你自己；但是如果你没能做到，不要归咎于缺乏好的Java教程。很多敬业的专业人士已经为编写良好的资料尽了全力。

我希望新版能够达到甚至超越以前几个版本树立的高度。最终的评判者是你——读者。我认为本书不会令读者失望。

Gilad Bracha²

加利福尼亚州洛斯阿尔托斯市

2006年6月

1. 计算机科学大师，1972年图灵奖得主。——编者注

2. Java顶级专家，Java语言规范的作者之一，曾是Sun公司的杰出工程师，目前供职于Cadence公司。——编者注

作者简介

Sharon Zakhour是本书创作团队的领导，在Sun公司作为高级技术作家任职7年。她编写了*JFC Swing Tutorial*（第2版），并且在Swing和AWT团队工作多年。她毕业于加利福尼亚大学伯克利分校，获得计算机科学学士学位，从事程序员、开发支持工程师和技术作家的的工作已20余年。

Scott Hommel是Sun公司的技术作家，他的工作是为Java SE编写文档。从1999年开始，他参与了JDK每个主要版本的工作，主要的工作是编写API规范说明和核心版本文档。现在他在位于佛蒙特州基灵顿的家中工作。

Jacob Royal是IT专业的硕士和信息系统方向的MBA。他编写过管理员指导、API参考、程序员指导，开发并编写过新的工具和代码，为很多公司（包括朗讯科技和Autodesk公司）编写过标准。他在Sun公司还参与了JSR API规范和Java部署指导的编写工作。

Isaac Rabinovitch是自由技术作家。他为Sun、Borland、SGI和很多其他公司编写过用户手册、程序员指南、管理员手册、API参考、版本说明和支持文档。他喜欢处理技术细节，从事计算机行业多年。

Tom Risser毕业于哈佛大学物理专业（获得学士学位）和加利福尼亚大学伯克利分校（获得博士学位）。他作为计算机行业的技术作家已达15年之久。

Mark Hoeber曾任Sun公司的高级技术作家。他作为技术作家已有12年，致力于为软件开发人员和系统管理人员编写文档。此外，Mark还参与过技术编写团队的内容管理系统的实现工作。

前 言

自从最初的JDK (Java Development Kit, Java开发工具包) 于1995年5月发布以来, Sun公司的工程人员为改进和增强Java平台付出了艰苦的努力。本版与Java SE (Java Platform Standard Edition, Java平台标准版) 6相一致, 并且反映了这一版本的API。

本版介绍了从第3版 (对应Java SE 1.3) 出版以来Java平台添加的新特性, 比如介绍泛型和注解的一章 (第6章)。还有关于Java Web Start (第17章)、平台环境 (第14章) 和正则表达式 (第13章) 的几章。并发 (原来的“线程”) (第12章)、I/O (第10章)、面向对象的编程概念 (第2章) 和语言基础 (第3章) 这几章都完全重新编写了。集合 (第11章) 从附录中提取出来成为独立的一章。新的附录B包含如何准备Java编程语言认证 (Java Programming Language Certification) 考试的信息。

本书的所有资料都经过了Sun公司Java开发人员的仔细审核, 以便确保信息是准确的和最新的。

和前几版一样, 本书以Sun公司网站上的在线教程为基础:

<http://java.sun.com/docs/books/tutorial/>

本书包含的信息 (经常被称为“核心教程”或者“基础”) 针对初级到中级程序设计人员的需要。掌握了本书的内容之后, 读者可以在该网站上学习Java平台的其他知识。

像以往一样, 我们的目的是编写容易阅读的实用程序员指南, 并且用很多示例帮助读者理解程序。

本书面向的读者

本书既适合初学者, 也适合有经验的程序员。

- 新程序员通过从头到尾阅读本书可以得到最大的收获, 包括按照第1章中的分步讲解编译和运行自己的第一个程序。
- 具有过程式语言 (比如C) 经验的程序员可以从Java编程语言的面向对象概念和特性开始学习。
- 经验丰富的程序员可以跳到更高级的主题, 比如关于泛型、并发或者Java Web Start的内容。

本书包含的内容可以满足各种不同经验的程序员的学习需要。

如何使用本书

本书的设计方式使读者既可以通读, 也可以在各主题之间跳转。信息按照逻辑的顺序编排, 并且尽可能避免向前引用。

书中的示例是在6.0版本下编译的。为了编译和运行大多数示例, 读者需要下载这个版本。

本书的网上下载资源包含本书英文原版的内容¹（包括示例，还有问题和练习的答案²），以及在线Java SE教程、Java SE开发工具包（Java SE Development Kit, JDK）³、相应的API规范和指导文档⁴。

在本书中你会看到这样的脚注：

`docs/api/java/lang/Class.html`

和

`tutorial/deployment/applet/examples/TalkServer.java`

根目录包含docs和tutorial目录，所以能在指定的位置找到这些脚注。要想定位到最新的在线页面，在docs脚注前面加上<http://java.sun.com/javase/6/>；在tutorial脚注前面加上<http://java.sun.com/docs/books/>即可：

<http://java.sun.com/javase/6/docs/api/java/lang/Class.html>

<http://java.sun.com/docs/books/tutorial/deployment/applet/examples/TalkServer.java>

本书英文原版的章节在网上下载资源的下列位置：

章	位 置
第1章 入门	<code>tutorial/getStarted</code>
第2章 面向对象的编程概念	<code>tutorial/java/concepts</code>
第3章 语言基础	<code>tutorial/java/nutsandbolts</code>
第4章 类和对象	<code>tutorial/java/javaOO</code>
第5章 接口和继承	<code>tutorial/java/landI</code>
第6章 泛型	<code>tutorial/java/generics</code>
第7章 包	<code>tutorial/java/package</code>
第8章 数字和字符串	<code>tutorial/java/data</code>
第9章 异常	<code>tutorial/essential/exceptions</code>
第10章 基本 I/O	<code>tutorial/essential/io</code>
第11章 集合	<code>tutorial/collections</code>
第12章 并发	<code>tutorial/essential/concurrency</code>
第13章 正则表达式	<code>tutorial/essential/regex</code>
第14章 平台环境	<code>tutorial/essential/environment</code>
第15章 Swing	<code>tutorial/ui</code>
第16章 把程序打包到 JAR 文件中	<code>tutorial/deployment/jar</code>
第17章 Java Web Start	<code>tutorial/deployment/webstart</code>
第18章 applet	<code>tutorial/deployment/applet</code>

我们欢迎读者对本书多提宝贵意见。请使用如下的Tutorial反馈论坛：

http://developers.sun.com/contact/tutorial_feedback.jsp

1. <http://java.sun.com/docs/books/tutorial/>

2. 为了方便读者参考，图灵网站（www.turingbook.com）还提供了本书“问题和练习”部分的中文版答案。

——编者注

3. <http://java.sun.com/javase/downloads/index.jsp>

4. <http://java.sun.com/javase/6/docs/api>

致谢

许多因特网读者帮助我们保持并提高了本书的质量，他们给我们发电子邮件，热情地指出我们的输入错误和无效的链接，以及本书中可能引起误解的地方和需要重写的地方。

许多现任和前任的Java开发人员和文档编写人员给我们提了许多建议，解答了我们的许多问题，审阅了我们的内容，甚至参与了本书部分内容的编写。这个名单很长，其中我们特别要感谢Eric Armstrong、David Bristor、Martin Buchholz、Mandy Chung、Iris Garcia Clark、Margarita Fischer、Amy Fowler、Andy Herrick、Shannon Hickey、David Holmes、Jim Holmlund、Patrick Keegan、Peter Korn、Masayoshi Okutsu、Scott Seligman、Xueming Shen、Sundar Sivasubramanian、Pete Soper、Scott Violet和Kathy Walrath作出的贡献。我们还要特别感谢Sun公司其他高水平的作者和程序员，他们作为特邀作者参与了在线教程的编写。

我们要感谢以前的同事Joshua Bloch¹（“集合”一章的原作者），他在百忙之中为我们提供了最新的信息。我们还要感谢特邀的审稿人Doug Lea和Brian Goetz的长期支持。

感谢我们的经理Debra Scott给予我们鼓励，并且给我们的工作提供了必要的支持。

我们的程序经理Mary Lautner和Ian Hardie跟踪进度、分担事务性工作，并以其幽默的天性使工作变得轻松。

插图设计者Michael Quillman和Dwayne Wolff制作了专业的图片，工作又快又好。

我们还要感谢SQE团队成员提供了有价值的反馈。

最后，感谢Java SE部门的经理让他们的工程师和插图设计者帮助编写本书，特别感谢：Uday Dhanikonda、Frances Ho、James Hsieh、Janet Koenig和Keith Yarwood。

1. Bloch曾是Java平台设计的灵魂人物，Java集合框架的设计者，现供职于Google公司。他的《Java解惑》一书（人民邮电出版社出版）是Java进阶最佳读物之一。——编者注

目 录

第1章 入门.....	1	3.1.4 变量小结.....	33
1.1 Java技术现象.....	1	问题和练习: 变量.....	34
1.1.1 关于Java技术.....	1	3.2 操作符.....	34
1.1.2 Java技术能做什么.....	3	3.2.1 赋值、算术和一元操作符.....	35
1.1.3 Java技术将如何改变我们的 生活.....	3	3.2.2 相等、关系和条件操作符.....	37
1.2 “Hello World!”应用程序.....	4	3.2.3 位操作符和移位操作符.....	39
1.2.1 使用NetBeans IDE编写 “Hello World!”.....	4	3.2.4 操作符小结.....	40
1.2.2 在Microsoft Windows平台 上编写“Hello World!”.....	9	问题和练习: 操作符.....	41
1.2.3 在Solaris OS和Linux平台 上编写“Hello World!”.....	12	3.3 表达式、语句和块.....	42
1.3 分析“Hello World!”应用程序.....	15	3.3.1 表达式.....	42
1.3.1 源代码注释.....	16	3.3.2 语句.....	42
1.3.2 HelloWorldApp类定义.....	16	3.3.3 块.....	43
1.3.3 main方法.....	16	问题和练习: 表达式、语句和块.....	43
1.4 常见问题(和解决方案).....	17	3.4 控制流语句.....	44
1.4.1 编译器问题.....	17	3.4.1 if-then和if-then-else 语句.....	44
1.4.2 运行时问题.....	18	3.4.2 switch语句.....	45
问题和练习: 入门.....	19	3.4.3 while和do-while语句.....	47
第2章 面向对象的编程概念.....	21	3.4.4 for语句.....	48
2.1 对象是什么.....	21	3.4.5 分支语句.....	49
2.2 类是什么.....	22	3.4.6 控制流语句小结.....	52
2.3 继承是什么.....	24	问题和练习: 控制流语句.....	52
2.4 接口是什么.....	24	第4章 类和对象.....	54
2.5 包是什么.....	25	4.1 类.....	54
问题和练习: 面向对象的编程概念.....	25	4.1.1 声明类.....	55
第3章 语言基础.....	27	4.1.2 声明成员变量.....	55
3.1 变量.....	27	4.1.3 定义方法.....	57
3.1.1 命名.....	28	4.1.4 为类提供构造器.....	58
3.1.2 基本数据类型.....	28	4.1.5 把信息传递给方法或构造器.....	59
3.1.3 数组.....	30	4.2 对象.....	62
		4.2.1 创建对象.....	63
		4.2.2 使用对象.....	66
		4.3 关于类的更多介绍.....	67
		4.3.1 从方法返回值得.....	68

4.3.2 使用关键字this	69	5.2.9 Object作为超类	101
4.3.3 控制对类成员的访问	70	5.2.10 编写最终类和方法	104
4.3.4 了解实例成员和类成员	72	5.2.11 抽象方法和抽象类	104
4.3.5 初始化字段	75	5.2.12 继承小结	106
4.3.6 创建和使用类和对象的小结	76	问题和练习: 继承	106
问题和练习: 类	76	第6章 泛型	108
问题和练习: 对象	77	6.1 简介	108
4.4 嵌套类	78	6.2 泛型类型	109
4.4.1 为什么使用嵌套类	78	6.3 泛型方法和构造器	111
4.4.2 静态嵌套类	79	6.4 受限类型参数	112
4.4.3 内部类	79	6.5 子类型化	113
4.4.4 内部类示例	80	6.6 通配符	114
4.4.5 嵌套类小结	81	6.7 类型擦除	115
问题和练习: 嵌套类	82	6.8 泛型小结	116
4.5 枚举类型	83	问题和练习: 泛型	117
问题和练习: 枚举类型	85	第7章 包	118
4.6 注解	86	7.1 创建和使用包	118
4.6.1 文档	86	7.1.1 创建包	119
4.6.2 编译器使用的注解	87	7.1.2 命名包	120
4.6.3 注解处理	88	7.1.3 使用包成员	121
问题和练习: 注解	88	7.1.4 管理源文件和类文件	123
第5章 接口和继承	90	7.1.5 创建和使用包小结	124
5.1 接口	90	问题和练习: 创建和使用包	124
5.1.1 Java中的接口	90	第8章 数字和字符串	126
5.1.2 接口作为API	91	8.1 数字	126
5.1.3 接口和多重继承	91	8.1.1 数字类	126
5.1.4 定义接口	91	8.1.2 格式化数字输出	128
5.1.5 实现接口	92	8.1.3 高级算术功能	131
5.1.6 使用接口作为类型	93	8.1.4 数字小结	134
5.1.7 重新编写接口	94	问题和练习: 数字	134
5.1.8 接口小结	95	8.2 字符	136
问题和练习: 接口	95	8.3 字符串	137
5.2 继承	95	8.3.1 创建字符串	138
5.2.1 Java平台类层次结构	96	8.3.2 字符串长度	138
5.2.2 继承示例	96	8.3.3 连接字符串	139
5.2.3 在子类中能进行什么操作	97	8.3.4 创建格式化字符串	139
5.2.4 子类中的私有成员	98	8.3.5 在数字和字符串之间进行 转换	140
5.2.5 对象的类型转换	98	8.3.6 处理字符串中的字符	141
5.2.6 覆盖和隐藏方法	99		
5.2.7 隐藏字段	100		
5.2.8 使用关键字super	100		

8.3.7 比较字符串和部分字符串	145	10.1.7 对象流	182
8.3.8 StringBuilder类	146	10.2 文件I/O	185
8.3.9 字符和字符串小结	149	10.2.1 File对象	185
问题和练习: 字符和字符串	150	10.2.2 随机访问文件	187
第9章 异常	151	10.3 新I/O包	189
9.1 异常是什么	151	10.4 小结	189
9.2 捕获或者指定需求	152	问题和练习: 基本I/O	189
9.2.1 异常的三种类型	152	第11章 集合	190
9.2.2 绕过捕获或者指定	153	11.1 集合简介	190
9.3 捕获和处理异常	153	11.1.1 集合框架是什么	190
9.3.1 try块	154	11.1.2 Java集合框架的优势	190
9.3.2 catch块	154	11.2 接口	191
9.3.3 finally块	155	11.2.1 Collection接口	192
9.3.4 结合使用	156	11.2.2 Set接口	195
9.4 指定方法抛出的异常	159	11.2.3 List接口	198
9.5 如何抛出异常	159	11.2.4 Queue接口	204
9.5.1 throw语句	160	11.2.5 Map接口	206
9.5.2 Throwable类及其子类	160	11.2.6 对象排序	212
9.5.3 Error类	161	11.2.7 SortedSet接口	216
9.5.4 Exception类	161	11.2.8 SortedMap接口	219
9.5.5 链式异常	161	11.2.9 接口小结	219
9.5.6 创建异常类	162	问题和练习: 接口	220
9.6 关于不可控异常的争论	163	11.3 实现	221
9.7 异常的优点	164	11.3.1 Set实现	222
9.7.1 优点一: 把错误处理代码和 “常规”代码分离开	164	11.3.2 List实现	223
9.7.2 优点二: 把错误沿调用栈向上 传递	165	11.3.3 Map实现	224
9.7.3 优点三: 对错误类型进行分组 和区分	166	11.3.4 Queue实现	225
9.8 小结	167	11.3.5 包装器实现	226
问题和练习: 异常	167	11.3.6 简便实现	227
第10章 基本I/O	169	11.3.7 实现小结	228
10.1 I/O流	169	问题和练习: 实现	229
10.1.1 字节流	170	11.4 算法	229
10.1.2 字符流	171	11.4.1 排序	229
10.1.3 缓冲流	174	11.4.2 混序	231
10.1.4 扫描和格式化	174	11.4.3 常规数据操作	231
10.1.5 命令行I/O	179	11.4.4 搜索	231
10.1.6 数据流	180	11.4.5 组合	232
		11.4.6 查找极值	232
		11.5 定制集合实现	232
		11.5.1 编写实现的原因	232
		11.5.2 如何编写定制实现	233

11.6 互操作性	234	13.4 字符类	264
11.6.1 兼容性	235	13.5 预定义字符类	268
11.6.2 API设计	236	13.6 量词	270
第12章 并发	238	13.6.1 零长度匹配	270
12.1 进程和线程	238	13.6.2 使用量词的捕获组和 字符类	273
12.1.1 进程	238	13.6.3 greedy、reluctant和possessive 量词的区别	273
12.1.2 线程	238	13.7 捕获组	274
12.2 Thread对象	239	13.7.1 编号	274
12.2.1 定义和启动线程	239	13.7.2 反向引用	275
12.2.2 使用sleep暂停执行	240	13.8 边界匹配器	275
12.2.3 中断	240	13.9 Pattern类的方法	277
12.2.4 join	241	13.9.1 创建带有标志的模式	277
12.2.5 SimpleThreads示例	242	13.9.2 嵌入标志表达式	278
12.3 同步	243	13.9.3 使用matches(String, CharSequence)方法	279
12.3.1 线程干扰	243	13.9.4 使用split(String) 方法	279
12.3.2 内存一致性错误	244	13.9.5 其他工具方法	280
12.3.3 同步方法	245	13.9.6 java.lang.String中和 Pattern等同的方法	280
12.3.4 内在锁和同步	246	13.10 Matcher类的方法	280
12.3.5 原子访问	247	13.10.1 索引方法	280
12.4 活性	247	13.10.2 研究方法	281
12.4.1 死锁	247	13.10.3 替换方法	281
12.4.2 饿死和活锁	248	13.10.4 使用start和end方法	281
12.5 保护块	249	13.10.5 使用matches和 lookingAt方法	282
12.6 不可变对象	252	13.10.6 使用replaceFirst (String)和replaceAll (String)	283
12.6.1 同步类示例	252	13.10.7 appendReplacement (StringBuffer,String) 和appendTail(String- Buffer)方法	284
12.6.2 定义不可变对象的策略	253	13.10.8 java.lang.String中 和Matcher等同的方法	284
12.7 高级并发对象	255	13.11 PatternSyntaxException 类的方法	284
12.7.1 Lock对象	255	13.12 小结	286
12.7.2 执行器	257	13.13 附加资源	287
12.7.3 并发集合	258		
12.7.4 原子变量	259		
12.8 推荐阅读	260		
问题和练习: 并发	260		
第13章 正则表达式	262		
13.1 简介	262		
13.1.1 正则表达式是什么	262		
13.1.2 这个包如何表示正则 表达式	262		
13.2 测试示例	263		
13.3 字符串字面量	263		

问题和练习: 正则表达式	287	16.1.4 更新JAR文件	319
第 14 章 平台环境	288	16.1.5 运行JAR打包的软件	321
14.1 配置工具	288	16.2 处理清单文件: 基础	322
14.1.1 属性	288	16.2.1 了解默认的清単	322
14.1.2 命令行参数	291	16.2.2 修改清单文件	322
14.1.3 环境变量	292	16.2.3 设置应用程序的入口点	323
14.1.4 其他配置工具	293	16.2.4 把类添加到JAR文件的类 路径	324
14.2 系统工具	294	16.2.5 设置包版本信息	324
14.2.1 命令行I/O对象	294	16.2.6 在JAR文件中密封包	326
14.2.2 系统属性	294	16.3 JAR文件的签名和验证	326
14.2.3 安全管理器	296	16.3.1 了解签名和验证	327
14.2.4 系统中的杂项方法	297	16.3.2 签名JAR文件	329
14.3 PATH和CLASSPATH	297	16.3.3 验证签名的JAR文件	330
14.3.1 更新PATH变量 (Microsoft Windows NT / 2000 / XP)	298	16.4 使用和JAR相关的API	330
14.3.2 更新PATH变量 (Solaris 和Linux)	298	16.4.1 示例——应用程序 JarRunner	330
14.3.3 检查CLASSPATH变量 (所有 平台)	299	16.4.2 JarClassLoader类	331
问题和练习: 平台环境	299	16.4.3 JarRunner类	333
第 15 章 Swing	301	问题和练习: JAR文件	334
15.1 Swing包概述	301	第 17 章 Java Web Start	335
15.1.1 Swing是什么	301	17.1 运行Java Web Start应用程序	336
15.1.2 Swing演示程序	302	17.1.1 从浏览器运行Java Web Start 应用程序	336
15.2 Swing的特性	305	17.1.2 从Java缓存查看器运行Java Web Start应用程序	336
15.2.1 Swing组件的可视化 指导	306	17.1.3 从桌面运行Java Web Start应用 程序	337
15.2.2 可插拔的界面外观	310	17.2 部署Java Web Start应用程序	337
15.2.3 拖放和数据传输	311	17.2.1 设置Web服务器	337
15.2.4 国际化和本地化	311	17.2.2 创建JNLP文件	337
15.2.5 可访问性	312	17.2.3 把应用程序放到服务器上	342
15.2.6 和桌面集成	312	17.2.4 创建Web页面	342
15.2.7 系统托盘图标支持	313	17.3 开发Java Web Start应用程序	342
问题和练习: 图形化用户界面	313	17.3.1 把应用程序打包到JAR 文件中	342
第 16 章 把程序打包到 JAR 文件中	314	17.3.2 读取JAR文件中的资源	343
16.1 使用JAR文件: 基础	314	17.3.3 不被信任的应用程序	343
16.1.1 创建JAR文件	315	17.4 JNLP API	343
16.1.2 查看JAR文件的内容	318	17.5 Java Web Start和安全	344
16.1.3 提取JAR文件的内容	318	17.5.1 Java Web Start安全基础	344

17.5.2	Java Web Start部署的签名 JAR文件	345
17.5.3	安全和JNLP文件	345
17.5.4	HTTPS证书的动态下载	345
17.6	常见的Java Web Start问题	345
	问题和练习: Java Web Start	346
第 18 章	applet	347
18.1	applet入门	348
18.1.1	扩展Applet或者JApplet	349
18.1.2	applet的生存周期	349
18.1.3	里程碑方法	351
18.1.4	用于绘制和事件处理的方法	352
18.1.5	用于添加UI组件的方法	354
18.1.6	applet能做什么,不能做什么	357
18.2	利用applet API的优势	357
18.2.1	查找和加载数据文件	358
18.2.2	显示简短状态字符串	359
18.2.3	在浏览器中显示文档	359
18.2.4	向其他applet发送消息	360
18.2.5	播放声音	363
18.2.6	定义和使用applet参数	365
18.2.7	使用APPLET标记	367
18.3	编写applet时的考虑事项	370
18.3.1	安全限制	370
18.3.2	创建用户界面	371
18.3.3	获得系统属性	373
18.3.4	applet中的线程	374
18.3.5	处理服务器端应用程序	376
18.4	完成applet	378
18.4.1	发布applet前的工作	378
18.4.2	圆满完成的applet	379
18.5	部署applet	380
18.5.1	一般的部署考虑事项	380
18.5.2	使用applet标记	380
18.5.3	使用object标记	380
18.5.4	使用embed标记	381
18.5.5	在混合浏览器环境中部署 applet	382
18.6	解决常见的applet问题	383
	问题和练习: Java applet	384
附录 A	Java 语言关键字	385
附录 B	Java 编程语言认证考试的 准备	386
索引		394

入 门

1.1 Java 技术现象

到处都在谈论Java技术，但是Java技术到底是什么呢？下面的章节将说明Java技术为何既是一种编程语言又是一个平台，并且概述这种技术能为你做什么。

1.1.1 关于 Java 技术

Java技术既是一种编程语言，也是一个平台。

1. Java编程语言

Java编程语言是一种高级语言，其特点可以用下面几个词语来归纳：

- 简单；
- 平台独立；
- 面向对象；
- 可移植；
- 分布式；
- 高性能；
- 多线程；
- 健壮；
- 动态；
- 安全。

在James Gosling和Henry McGilton撰写的*Java Language Environment*¹一文中，对上面的每个词语都给了解释。

在Java编程语言中，首先把所有源代码都写入到纯文本文件中，以.java扩展名结束。然后，使用javac编译器把这些源文件编译为.class文件。.class文件不包含对应于处理器的本机代码，而是包含字节码（bytecode）——Java虚拟机（Java Virtual Machine, Java VM）的机器语言²。然后java启动器工具通过Java虚拟机的一个实例运行应用程序（见图1-1）。

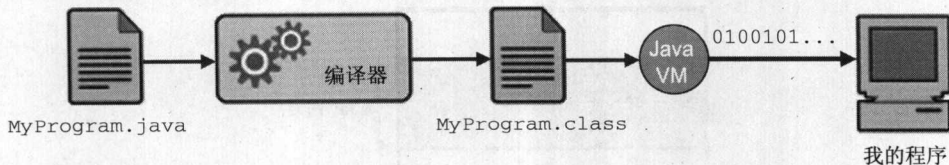


图1-1 编译和运行应用程序

1. <http://java.sun.com/docs/white/langenv/>

2. 术语“Java虚拟机”和“JVM”都表示Java平台上的虚拟机（Virtual Machine）。

因为在很多不同的操作系统上都有Java VM，所以相同的.class文件可以运行在Microsoft Windows、Solaris OS、Linux或者Mac OS上。一些虚拟机，比如Java HotSpot虚拟机¹，在运行时还会执行额外的步骤，以提高应用程序的性能。这些步骤包括各种任务，比如查找性能瓶颈和将频繁使用的代码段落重编译（为本机代码）（见图1-2）。

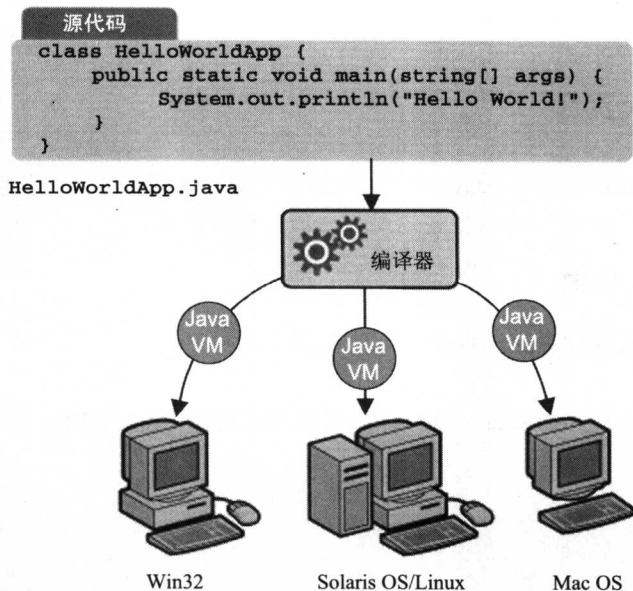


图1-2 通过Java VM，同一个应用程序能够运行在多种平台上

2. Java平台

平台（platform）是程序在其中运行的硬件或软件环境。我们已经提到了一些最流行的平台，比如Microsoft Windows、Linux、Solaris OS和Mac OS。大多数平台可以被描述为操作系统和底层硬件的组合。Java平台与大多数其他平台的不同之处在于，它是一种运行在其他基于硬件平台之上的纯软件平台。

Java平台有两个组成部分：

- Java虚拟机（Java Virtual Machine, Java VM）。
- Java应用程序编程接口（Application Programming Interface, API）。

我们已经介绍了Java虚拟机，它是Java平台的基础，已经被移植到了各种硬件平台上（见图1-3）。

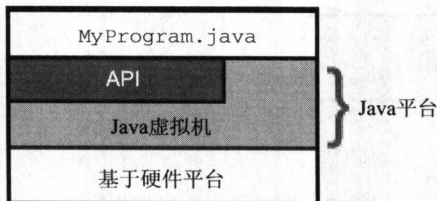


图1-3 API和Java虚拟机把程序和底层硬件隔离开

API是已经开发好的软件组件的大型集合，它们提供了许多有用的功能。API按相关性类和

1. <http://java.sun.com/javase/technology/hotspot.jsp>

接口分为多个库，这些库被称为包（package）。下一节重点介绍API提供的一些功能。

作为独立于平台的环境，Java平台比本机代码稍慢一些。但是，编译器和虚拟机技术的发展已经使Java平台的性能与本机代码接近，而且不会有损于可移植性。

1.1.2 Java 技术能做什么

通用的高级Java编程语言是一个功能强大的软件平台。Java平台的每个完整实现都能提供如下特性：

- **开发工具**——开发工具提供进行编译、运行、监视、调试应用程序和编写应用程序文档所需的所有功能。作为开发新手，你将使用的主要工具是javac编译器、java启动器和javadoc文档工具。
- **API**——API提供Java编程语言的核心功能。它提供大量已编写好的类，可以用于你自己的应用程序。核心API非常大；要大致了解API包含的内容，请参考Java SE开发工具包6（Java SE Development Kit 6，JDK 6）文档¹。
- **部署技术**——JDK软件提供标准机制，比如Java Web Start软件和Java Plug-In软件，用于把应用程序部署给最终用户。
- **用户界面工具包**——Swing和Java 2D工具包使创建复杂的图形用户界面（Graphical User Interface，GUI）成为可能。
- **集成库**——集成库，比如Java IDL API、JDBC API、Java命名和目录接口（Java Naming and Directory Interface，JNDI）API、Java RMI和Java RMI-IIOP技术（Java Remote Method Invocation over Internet Inter-ORB Protocol Technology），支持数据库访问和远程对象的操纵。

3
?
4

1.1.3 Java 技术将如何改变我们的生活

我们不能保证如果你学习了Java编程语言就能够得到荣誉、财富，甚至工作。但是，它很可能使你的程序更好，而需要付出的努力却比其他语言少。我们相信Java技术会给你带来以下好处：

- **入门迅速**——尽管Java编程语言是一种强大的面向对象语言，但是它容易学，特别是对于熟悉C或C++的程序员。
- **编写的代码更少**——程序度量（比如类的数量、方法的数量等）的对比表明，用Java编程语言写的程序比用C++写的相同程序小4倍。
- **编写的代码更好**——Java编程语言鼓励良好的编程实践，而且它的自动垃圾回收功能有助于避免内存泄漏。它的面向对象性质、JavaBeans组件架构以及易于扩展的大量API，使你可以重用已有的、经过测试的代码并且减少错误。
- **开发程序更快**——Java编程语言比C++简单，因为这个原因，使用Java进行开发的速度可能要快上一倍。你的程序所需的代码行数也更少。
- **平台无关性**——因为不需要使用由其他语言编写的库，可以保持程序的可移植性。
- **一次编写，处处运行**——因为Java程序被编译为与机器无关的字节码，所以它们可以在任何Java平台上以一致的方式运行。
- **发布软件更容易**——使用Java Web Start软件，用户只需通过点击鼠标就可启动应用程序。启动时的自动版本检查使用户总能更新到软件的最新版本。如果有可用的更新，那么Java Web Start软件就会自动更新安装。

5

¹ docs/index.html