

Advanced Tricks, Tips, and Techniques for Java Programming

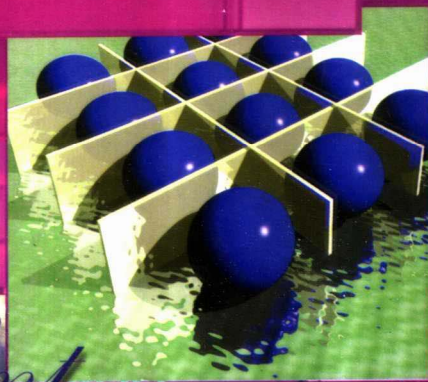
Java

案例精粹 150 例

张怀庆 编著



Excellent



冶金工业出版社

Java

案例教程

150

第2版



Java 案例精粹 150 例

张怀庆 编著

北 京

冶金工业出版社

2005

内 容 简 介

本书通过 150 个实例全面介绍了 Java 编程基础、多线程编程、输入/输出编程、Applet 编程、Swing 图形界面编程、多媒体和图形图像处理编程、网络编程、数据库编程、B/S 结构编程、Web Service 编程、EJB 编程、安全性编程、J2ME 编程、Java 其他编程以及开发环境配置等。这些实例典型、简洁，主要功能突出，所涉及的技术可以解决同类问题。读者可以通过本书实例提供的技术或技巧去实现更为复杂的功能，并应用到更广的方面。

本书例题典型、结构合理、思路清晰，既可作为大专院校相关专业及 Java 培训班的教材，也可作为具有一定 Java 编程知识的自学者的参考用书，同时还可以作为从事软件设计的工程人员的参考用书。

图书在版编目 (CIP) 数据

Java 案例精粹 150 例 / 张怀庆编著. —北京: 冶金工业出版社, 2005.2
ISBN 7-5024-3708-8

I. J... II. 张... III. JAVA 语言—程序设计
IV. TP312

中国版本图书馆 CIP 数据核字 (2004) 第 142919 号

出版人 曹胜利 (北京沙滩嵩祝院北巷 39 号, 邮编 100009)

责任编辑 程志宏

湛江蓝星南华印务公司印刷; 冶金工业出版社发行; 各地新华书店经销

2005 年 3 月第 1 版, 2005 年 3 月第 1 次印刷

787mm × 1092mm 1/16; 21.75 印张; 501 千字; 338 页

39.00 元

冶金工业出版社发行部 电话: (010) 64044283 传真: (010) 64027893

冶金书店 地址: 北京东四西大街 46 号 (100711) 电话: (010) 65289081

(本社图书如有印装质量问题, 本社发行部负责退换)

前 言

一、关于 Java

Java(tm)是由美国升阳电脑公司(Sun Microsystems, Inc)所发展出的第一种能在国际互联网络(Internet)上具有「硬件/软件中立性」交互能力的程序语言。Java一推出即引起世界各种媒体的广泛注意与讨论,如Time、Business Weekly、CNN等,可见其功能及魅力,事实上,以Java的功能,势必在信息界掀起另一番高潮,对于网络的发展,又将迈入另一个新的纪元。Java是一种软件技术,在Sun的Java语言白皮书中明白地说明Java语言的特点:简单的(Simple)、面向对象的(Object-Oriented)、分布式的(Distributed)、健壮的(Robust)、安全的(Secure)、结构中立的(Architecture Neutral)、可移植的(Portable)、解释的(Interpreted)、高效能的(High Performance)、多线程的(Multi threaded)、动态的(Dynamic)。

1998年,Sun发布了Java 2平台。在Java的发展史上,Java 2是一个里程碑式的产品。Java 2不但在性能上有很大提高,还提供了一个安全、灵活的新模式,以及一套完整的应用软件开发接口,同年,Sun在Javaone大会上发布了EJB1.0规范,为建立分布式商务对象系统提供了坚实的结构体系基础。从此Java开始走向企业级应用平台。Sun确定的战略目标是将Java发展成为三个系统:针对嵌入式设备的J2ME、针对个人PC的J2SE、企业级服务器端平台的J2EE。

2000年Sun发布了J2ME。J2ME是针对嵌入式的Java平台,随后Sun又发布了专门针对移动通讯工具的MIDP开发规范以作为J2ME的补充。到2004年,J2ME技术、电子邮件、HTML衍生产品已经成为手持移动设备的主要开发平台。

2001年1月Java的企业版J2EE发布,J2EE是分布式企业软件组件架构的规范,具有Web性能、灵活性、简化的集成性、便捷性以及J2EE服务器之间的互操作性。

2001年9月J2SDK1.4Beta2发布。Sun制定Java的发展方向后,JDK1.2以后的版本均称为J2SDK,这些版本都支持Java 2平台标准,同时为了支持业界的开放源代码运动,Sun公布了部分版本的源代码,可以在网上直接下载。Sun还发布了J2SDK1.5版本,在这个版本中Sun在语言和技术都作了重大改进,Java的功能变得更为强大。

二、本书结构

本书内容结构安排如下:

第1章:Java编程初步。介绍了Java面向对象编程的基础知识及Java数据结构的编程。

第2章:Java多线程编程。介绍了利用Java的多线程功能编程,并介绍了处理多个任务的应用程序,还介绍了几个多线程的著名问题的实现程序。

第3章:Java输入/输出编程。介绍了Java的输入、输出类的应用,包括文件的读写、压缩以及流、管道等的操作,目录示例器、对象次第读写等编程知识。

第4章:Java Applet编程。介绍了Applet小程序的开发,包括简单的Applet程序、文

本域、多行文本域和它们的触发事件等。

第 5 章：Java Swing 图形界面编程。介绍了利用 Swing 组件进行界面设计。

第 6 章：Java 多媒体和图形图像处理编程。介绍了 Java 在多媒体方面对音频和图像的处理。

第 7 章：Java 网络编程。介绍了 Java 获取网络资源、套接字、服务器/客户机。

第 8 章：Java 数据库编程。介绍了利用 JDBC 进行数据库连接和数据库编程的 7 个实例。

第 9 章：Java B/S 结构编程。通过 14 个实例介绍了 JSP、JavaBean、Servlet 等技术的编程应用。

第 10 章：Java 的 Web Service 编程。通过 6 个实例介绍了 XML 和 SOAP 编程。

第 11 章：EJB 编程。通过 7 个实例介绍了各种 EJB 的编程，如无状态会话 Bean、有状态会话 Bean、银行存取 BMP 等。

第 12 章：Java 安全性编程。介绍了各种加、解密和各种安全机制编程技术。

第 13 章：J2ME 编程。介绍了 J2ME 的框架和简单应用，包括 Command 对象应用、查看内存利用情况、菜单设计、记录存储、使用定时器、无线 HTTP 连接等。

第 14 章：Java 其他编程。介绍了 JNI 和一些与系统有关的编程技术的应用。

第 15 章：Java 工具和服务器配置与集成。介绍了 Java 的各种开发环境的安装和配置。

三、本书特点

本书内容丰富、结构合理，叙述简洁明了，有很强的实用性，以实例的形式向读者展示了 Java 2 编程技术的精髓，既注重面向对象软件设计的基础，又体现了 Java 2 的新技术。

四、本书适用对象

本书例题典型、结构合理、思路清晰，既可作为大专院校相关专业及 Java 培训班的教材，也可作为具有一定 Java 编程知识的自学者的参考用书，同时还可以作为从事软件设计的工程人员的参考用书。

本书针对 Java 2 的各种技术进行了全面的介绍，要求读者初步掌握 Java 程序设计的基础知识，可将此书作为 Java 入门与提高的参考书，可以帮助读者迅速掌握 Java 的各种技术的应用方法。本书的实例都通过编译并给出了运行结果，如有问题，敬请读者批评指正。

由于编者水平有限，编写时间仓促，缺点和错误在所难免，希望广大读者批评指正。

虽然经过严格的审核、精细的编辑，本书在质量上有了一定的保障，但我们的目标是力求尽善尽美，欢迎广大读者和专家对我们的工作提出宝贵建议，联系方式如下：

电子邮件：service@cnbook.net

网址：www.cnbook.net

本书涉及的源代码可在该网站的下载中心免费下载。此外，该网站还有一些其他相关书籍的介绍，可以方便读者选购参考。

编 者

2004 年 12 月

目 录

第 1 章 Java 编程初步	1
实例 1 Hello world!	1
实例 2 Integer 类	2
实例 3 显示最大值	3
实例 4 计算某年某月的天数.....	4
实例 5 数值运算	5
实例 6 日历表	6
实例 7 Character 类	7
实例 8 String 类	8
实例 9 StirngBuffer 类	9
实例 10 数组	10
实例 11 类	11
实例 12 Class 类	12
实例 13 观察垃圾回收过程	14
实例 14 继承	17
实例 15 动态绑定	17
实例 16 接口	18
实例 17 异常处理	19
实例 18 冒泡排序	20
实例 19 Hanio 塔问题	21
实例 20 链表	21
实例 21 用 LinkedList 实现栈	24
实例 22 Vector 类	26
实例 23 Properties 类	27
实例 24 ArrayList 类	28
实例 25 线程初探	29
小结	30
第 2 章 Java 多线程编程	32
实例 26 简单的多线程应用	32
实例 27 简单动画	34
实例 28 模拟接力	35
实例 29 线程树	37

实例 30 生产者 - 消费者问题	39
实例 31 死锁	42
实例 32 哲学家就餐问题	43
小结	46
第 3 章 Java 输入/输出编程	47
实例 33 文件的读取	47
实例 34 写入文件	48
实例 35 标准 I/O 重定向	49
实例 36 流操作	49
实例 37 管道操作	54
实例 38 目录列示器	55
实例 39 目录的检查与修改	56
实例 40 压缩文件	58
实例 41 对象次第读写	59
小结	61
第 4 章 Java Applet 编程	62
实例 42 简单的 Applet 程序	62
实例 43 文本域	64
实例 44 文本域触发事件	65
实例 45 多行文本域	66
实例 46 TextArea 触发事件	67
实例 47 画红点	68
实例 48 渐显文字	70
小结	72
第 5 章 Java Swing 图形界面编程	73
实例 49 鼠标画线	73
实例 50 使用画框	75
实例 51 JButton 示例	76
实例 52 JScrollPane 示例	78
实例 53 JCheckBox 实例	80
实例 54 JList 示例	82
实例 55 JTree 示例	84
实例 56 JTable 示例	88
实例 57 Java Application 启动画面制作	91

实例 58 带进度条的启动窗口	92
实例 59 文件选择对话框	95
实例 60 菜单	95
实例 61 对话框	97
实例 62 计算器	102
实例 63 TicTacToe 游戏	107
实例 64 文本编辑器	109
小结	112
第 6 章 Java 多媒体与图形图像处理编程	113
实例 65 画图	113
实例 66 填充图形	116
实例 67 动态填充	118
实例 68 字体	119
实例 69 数字时钟	121
实例 70 纹理与色彩	124
实例 71 模拟时钟	125
实例 72 椭圆形按钮	129
实例 73 裁剪效果	131
实例 74 Applet 音频播放器	135
实例 75 音频播放	139
小结	141
第 7 章 Java 网络编程	142
实例 76 访问并获得 URL 资源	142
实例 77 UDP 传送	143
实例 78 代理服务器	146
实例 79 网页浏览器	147
实例 80 Telnet	149
实例 81 扫描应用端口	150
实例 82 聊天工具	151
实例 83 上传文件	154
实例 84 IP 多组传播	159
实例 85 邮件发送	163
实例 86 FTP 浏览器	165
小结	173
第 8 章 Java 数据库编程	174

实例 87	JDBC-ODBC 桥	174
实例 88	数据库的常见操作	175
实例 89	使用 JDBC 驱动程序连接 SQL Server	178
实例 90	显示数据库的内容	180
实例 91	数据库查询和修改	183
实例 92	事务使用	187
实例 93	设置保存点	188
小结		190
第 9 章	Java B/S 结构编程	191
实例 94	HelloWorld	191
实例 95	读写文件	192
实例 96	网站计数器	193
实例 97	汉字处理	194
实例 98	文件的上传	196
实例 99	获取客户机浏览器的信息	199
实例 100	收发邮件	202
实例 101	JSP 连接数据库	206
实例 102	用 JavaBean 连接数据库	207
实例 103	网络定餐	209
实例 104	简单的 Servlet 程序	218
实例 105	在 Servlet 中实时地创建图像	220
实例 106	简单购物车	221
实例 107	用户注册登录系统	224
小结		235
第 10 章	Java 的 Web Service 编程	236
实例 108	读写 XML 文件	236
实例 109	DOM 树	239
实例 110	web.xml 中的 Servlet 配置	242
实例 111	简单的 SOAP 调用	245
实例 112	访问 SOAP RPC 服务	248
实例 113	查询 UDDI 信息	249
小结		251
第 11 章	EJB 编程	252
实例 114	JNDI 的使用	252

实例 115 构建一个简单的 EJB	253
实例 116 无状态会话 Bean	256
实例 117 简单的有状态会话 Bean	260
实例 118 银行存取 BMP	263
实例 119 MessageDrivenBean	271
实例 120 JSP 调用 EJB	274
小结	276
第 12 章 Java 安全性编程	277
实例 121 一个简单的加密和解密程序——凯撒密码	277
实例 122 DES 加密和解密	278
实例 123 RSA 加密和解密	279
实例 124 数字签名	281
实例 125 创建数字证书	283
实例 126 简单的 SSL 通信	284
实例 127 HTTPS 服务	286
实例 128 密码输入框	289
小结	290
第 13 章 J2ME 编程	292
实例 129 Hello World!	292
实例 130 Command 对象	295
实例 131 查看内存利用	296
实例 132 菜单设计	298
实例 133 记录存储	303
实例 134 使用定时器	308
实例 135 无线 HTTP 连接	310
小结	312
第 14 章 Java 其他编程	313
实例 136 JNI 调用本地 C 语言	313
实例 137 获取本机的 Mac 地址	314
实例 138 获取 Java 的系统信息	315
实例 139 贪食蛇游戏	316
小结	323
第 15 章 Java 工具和服务器配置与集成	324
实例 140 J2SE SDK 的安装和配置	324

实例 141	使用第三方包	325
实例 142	Apache 安装和配置	326
实例 143	Tomcat 的安装和配置	328
实例 144	整合 Tomcat 和 Apache	329
实例 145	整合 Tomcat 和 IIS	331
实例 146	WebLogic 的安装和配置	332
实例 147	整合 Jbuilder 与 WebLogic	333
实例 148	Jboss 的安装与配置	334
实例 149	Eclipse 开发 J2EE 配置	335
实例 150	安装、配置和执行 MIDlet	337
小结		337
参考文献		338

第 1 章 Java 编程初步

本章通过 25 个实例介绍了 Java 编程的基础知识,包括几种数据类型 Integer、Character、String、StringBuffer 的使用,类、继承、动态绑定、接口等面向对象编程的基础,还有 LinkedList、Vector、Properties、ArrayList 等 Java 的数据结构的编程。

实例 1 Hello world!

1. Java 的环境配置

Java 开发工具 (JDK) 是一个由 Sun 公司开发的免费软件包。这个软件包包括了所有编写 Java 程序和运行 Java 程序所需要的工具。它包括构成 Java 环境的所有基本组件,包括 Java 编译器、Java 解释器、小应用程序浏览器 (不用打开兼容 Java 的网页浏览器就可以观察小应用程序) 以及一些用于开发 Java 应用程序的程序。

JDK 可以使用户在编写程序时所要做的工作尽可能地减少。许多设计师可能已经习惯了使用集成的开发环境,但它不是集成的开发环境,它没有完全包括开发 Java 程序的所有工具,JDK 的工具是命令行方式的,它没有漂亮的图形用户界面,如 Visual C++6.0 或 Borland 的 C++builder, JDK 工具是在命令提示符下使用的 (就像 Windows 95 和 Windows NT 里的 DOS 窗口那样)。JDK 包含源代码的文件就是用文本编辑器编辑的普通 ASCII 码文本文件,这种文本编辑器可以是 NotPad (Win32 系统提供)、vi (UNIX 系统提供) 或者 BBEdit (Macintosh 提供)。

本实例介绍了在不用任何 Java 编程工具的情况下只用 JDK 和一般的文本编辑工具完成 Java 程序的编写和执行。先在 Sun 公司的主页 (<http://java.sun.com>) 上下载 JDK, 有多个版本, 选择 Java 2 SDK sel.4.2 版本下载并安装。

1) Path 环境变量的配置

JDK 安装完毕后, 在 Windows XP 下进行配置。

(1) 在“我的电脑”上单击右键进入“系统属性”, 单击“高级”标签项, 再选择“环境变量”, 在“系统变量(S)”下查找 Path 环境变量, 找到后选择“编辑”。

(2) 在“变量值”的文本框中, 将光标移动到最前方, 输入安装的 J2SDK1.4.2 所在的目录下 bin 目录的位置, 例如: 笔者在机器上安装的是 J2SDK1.4.0, 安装到了 D 盘上, 其 bin 目录的完整路径为 D:\j2sdk1.4.2\bin, 注意加上“;”作为分隔符号。如果在“系统变量(S)”下没有找到 Path 环境变量, 那么就选择“新建”。

(3) 在“变量名”中输入“Path”, 在变量值中输入安装的 J2SDK1.4.2 所在的目录下 bin 目录的位置, 然后单击“确定”按钮。

2) CLASSPATH 的配置

在“系统变量(S)”下, 选择“新建”, 在“变量名”的文本框中输入“CLASSPATH”, 在变量值中输入“.”, 单击“确定”按钮。

最后在“环境变量”对话框中单击“确定”按钮, 结束“Path”和“CLASSPATH”环境变量的配置。

2. Hello world 程序的实现

先建立一个文本文件 HelloWorld.java (仔细检查大小写, Java 是大小写敏感的), 然后开始编写这个简单的程序, 程序代码如下:

```
public class HelloWorld
{
    public static void main(String args[])
    {
        System.out.println("Hello,World!");
    }
}
```

在 DOS 命令提示符下进入 HelloWorld.java 文件的目录, 然后输入 javac HelloWorld.java。机器应该暂停一会儿, 然后退回到提示符下。如果出现“javac 找不到”这种错误, 是因为 PATH 变量中没有设置 JAVA/BIN 目录。那么就应该回到前面检查一下 JDK 环境变量配置。如果未出现任何错误, 就可以开始运行用户的第一个 Java 程序了。在命令提示符下输入下列内容:

```
java HelloWorld
```

这时用户将会看到下列结果:

```
Hello,World!
```

Java 环境通过在编译器和计算机之间增加一个名为 Java 虚拟机 (JVM) 的中介来解决跨平台的问题。Java 编译器 javac 不是直接面向某种计算机类型的计算机编译器, 而是将文本文件中人可读懂的 Java 源代码编译成 Java 虚拟机能够理解的低级 Java 字节码。Java 虚拟机接管这些字节码并对它解释, 这样 Java 程序就可以在装有 Java 虚拟机的计算机上运行了。

Java 开发工具包 (JDK) 中包含以下命令行工具:

javac: 基于 Java 编程语言的编译器。

java: Java 应用程序的启动器。

jar: 管理 Java 归档 (JAR) 文件。

javadoc: 提取代码中的注释, 并根据这些注释生成 HTML 文档。

appletviewer: 无需浏览器运行和调试 applet。

实例 2 Integer 类

本实例演示 Integer 类的使用方法。

【编程要点】

Integer 类在某对象中打包了原始类型为 int 的值。Integer 类型对象包含 int 型的单个域。此外, 此类提供了许多方法, 可以将 int 型转换为 String 型, 也可以将 String 型转换为 int 型, 还包含处理 int 类型时的其他有用常量和方法。

【程序实现】

IntegerDemo.java 文件内容如下:

```
public class IntegerDemo{
    public IntegerDemo(){
    }
    public static void main(String[] args){
        Integer[] array = {new Integer(20),new Integer(40), new Integer("110")};
        for(int i=0; i<array.length; i++){
            System.out.print(Integer.toBinaryString(array[i].intValue())+"\t");
            System.out.print(Integer.toHexString(array[i].intValue())+"\t");
            System.out.print(Integer.toOctalString(array[i].intValue())+"\n");
        }
    }
}
```

```
    }
}
```

【运行结果】

```
10100    14    24
101000   28    50
1101110  6e   156
```

【分析与总结】

整数的不同进制的表示方法如下：

toBinaryString(int i): 创建一个整数参数以 2 为基数的无符号整数的字符串表示。

toHexString(int i): 创建一个整数参数的以 16 为基数的无符号整数的字符串表示。

toOctalString(int i): 创建一个整数参数以 8 为基数的无符号整数的字符串表示。

实例 3 显示最大值

本实例显示 Java 常见数据类型的最大值。

【编程要点】

Java 的各种数据类型 int、float、double、byte 等在 java.lang 包里都有各自的类分别和它们对应。其中每个类都有个属性 MAX_VALUE 表示这种类型数据的最大值。

【程序实现】

MaxVariables.java 文件内容如下：

```
public class MaxVariables {                                //java 中的一切都可以用类表示
    public static void main(String args[]) {               //每个程序必须有 main() 函数
        //整数类型
        byte largestByte = Byte.MAX_VALUE;                //定义一个 byte 类型的变量
        short largestShort = Short.MAX_VALUE;              //定义一个 short 类型的变量
        int largestInteger = Integer.MAX_VALUE;            //定义一个 int 类型的变量
        long largestLong = Long.MAX_VALUE;                 //定义一个 long 类型的变量
        //实数类型
        float largestFloat = Float.MAX_VALUE;              //定义一个 float 类型的变量
        double largestDouble = Double.MAX_VALUE;           //定义一个 double 类型的变量
        //其他基本类型
        char aChar = 'S';                                   //定义一个字符
        boolean aBoolean = true;                            //定义一个布尔类型的变量
        //在屏幕上显示对应类型的最大值
        System.out.println("最大的 byte 值是: " + largestByte);
        System.out.println("最大的 short 值是: " + largestShort);
        System.out.println("最大的 integer 值是: " + largestInteger);
        System.out.println("最大的 long 值是: " + largestLong);
        System.out.println("最大的 float 值是: " + largestFloat);
        System.out.println("最大的 double 值是: " + largestDouble);
        if (Character.isUpperCase(aChar)) {                 //判断字符是否大写
            System.out.println("字符" + aChar + "是大写的字符");
        } else {
            System.out.println("字符" + aChar + "是小写的字符");
        }
        System.out.println("布尔型变量的值是: " + aBoolean);
    }
}
```

【运行结果】

最大的 byte 值是: 127

最大的 short 值是: 32767
 最大的 integer 值是: 2147483647
 最大的 long 值是: 9223372036854775807
 最大的 float 值是: 3.4028235E38
 最大的 double 值是: 1.7976931348623157E308
 字符 S 是大写的字符
 布尔型变量的值是: true

【分析与总结】

程序中还用到了 Character 类的 isUpperCase 方法, 用于判断字符是不是大写。

实例 4 计算某年某月的天数

本实例实现计算某年指定月份的天数。

【编程要点】

月份可分为大月和小月, 而二月份的天数跟是否闰年有关, 所以对二月份要特殊处理。

【程序实现】

DayShower.java 文件内容如下:

```
class DayShower { //定义一个 DayShower 类
    //本程序的 main() 方法
    public static void main(String[] arguments) { //定义一个 String 对象的数组, 用来存
放参数
        //初始化变量
        int yearIn = 2004;           //定义一个整数类型的变量, 并且赋初值
        int monthIn = 2;             //定义一个整数类型的变量, 并且赋初值
        //检验命令行参数的数量
        if (arguments.length > 0)
            monthIn = Integer.parseInt(arguments[0]);
        if (arguments.length > 1)
            yearIn = Integer.parseInt(arguments[1]);
        //在屏幕上输出得到的结果
        System.out.println(yearIn + "年" + monthIn + "月有"
            + countDays(monthIn, yearIn) + "天");
    }
    //countDays() 方法, 可以被 DayShower 调用
    static int countDays(int month, int year) { //用 switch 语句来计算某个月的天数
        int count = -1;
        switch (month) {
            case 1:
            case 3:
            case 5:
            case 7:
            case 8:
            case 10:
            case 12:
                count = 31;
                break;
            case 4:
            case 6:
            case 9:
            case 11:
                count = 30;
                break;
            case 2:
                if (year % 4 == 0)
                    count = 29;
                else
                    count = 28;
                if ((year % 100 == 0) & (year % 400 != 0))
```



```
        count = 28;
    }
    return count; //返回 count 的值
}
}
```

【运行结果】

2004 年 2 月有 29 天

【分析与总结】

程序中处理命令行参数时用到了 Integer 类的一个方法:

```
public static int parseInt(String s) throws NumberFormatException
```

该方法将字符串参数作为带符号十进制整数来分析。除第一个字符为 ASCII 字符中减号 '-' 表示的负数, 字符串中的字符都必须是十进制数。若该串不包含一个可分析的整数, 则抛出 NumberFormatException 异常。

从本例我们应该掌握条件语句 switch case 的使用。

实例 5 数值运算

本实例演示如何利用 Java 的 API 进行数值运算。

【编程要点】

java.lang 包的 Math 类包含基本数学操作, 如指数、对数、平方根和三角函数。

【程序实现】

MathDemo.java 文件内容如下:

//演示如何调用 Java API 完成数值运算

```
public class MathDemo {
    public static void main(String[] args) {
        //演示求绝对值和求最大、最小值函数的用法
        System.out.println("abs(-5) = " + Math.abs(-5));
        System.out.println("max(2.72, 3.14) = " + Math.max(2.72, 3.14));
        System.out.println("min(256, 285) = " + Math.min(256, 285));
        //演示四舍五入函数的用法
        System.out.println("round(3.8) = " + Math.round(3.8));
        System.out.println("round(-3.8) = " + Math.round(-3.8));
        //演示求平方根和求幂函数的用法
        System.out.println("sqrt(2) = " + Math.sqrt(2));
        System.out.println("pow((1+2.25/100), 5) = " + Math.pow((1 + 2.25/ 100), 5));
        //演示指数与对数函数的用法
        System.out.println("E = " + Math.E);
        System.out.println("exp(2) = " + Math.exp(2));
        System.out.println("log(2) = " + Math.log(2));
        //演示天花板与地板函数的用法
        System.out.println("ceil(3.14) = " + (int) Math.ceil(3.14));
        System.out.println("floor(3.14) = " + (int) Math.floor(3.14));
        //演示三角函数的用法
        System.out.println("Pi = " + Math.PI);
        System.out.println("sin(Pi / 2) = " + Math.sin(Math.PI / 2));
        System.out.println("cos(0) = " + Math.cos(0));
    }
}
```

【运行结果】

```
abs(-5) = 5
max(2.72, 3.14) = 3.14
min(256, 285) = 256
round(3.8) = 4
round(-3.8) = -4
sqrt(2) = 1.4142135623730951
pow((1+2.25/100), 5) = 1.1176776934618162
E = 2.718281828459045
```