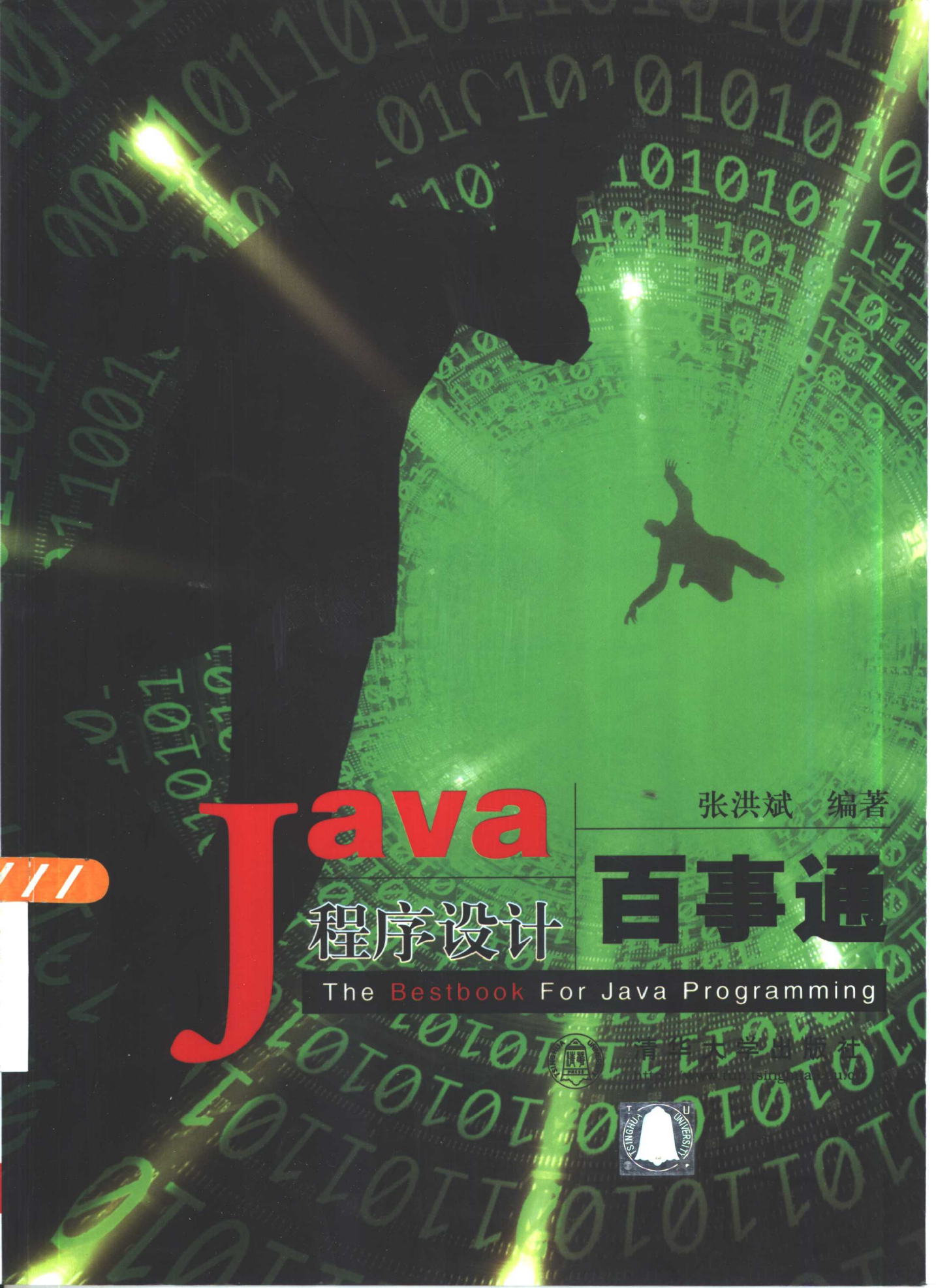




Java

程序设计

张洪斌 编著

百事通

The Bestbook For Java Programming



Java 程序设计百事通

The Bestbook For Java Programming

张洪斌 编著

清华大学出版社

(京)新登字 158 号

内 容 摘 要

Java 是目前真正跨平台、纯粹的面向对象、适合单机和 Internet 开发的编程语言。Java 问世较晚,是一种新型的程序设计语言,在国外十分流行,有网络时代 C++ 之美誉。本书详细讲解了面向对象的理论在 Java 中的实现和常见的程序设计主题。本书的开发环境为 JDK1.3。本书可供高等院校在校学生,网络技术开发人员,希望了解和详细学习面向对象技术和 Java 实现技术的人员,Java 程序设计新手等选用。

本书作者在计算机图书写作方面具有较高的技巧和先进的教育思想,读者将会发现这是目前写得最容易阅读、收获最大的 Java 程序设计入门与提高的著作。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

Java 程序设计百事通/张洪斌著. - 北京:清华大学出版社,2001

ISBN 7-302-04308-6

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(2001)第 12769 号

出 版 者:清华大学出版社(北京清华大学学研大厦 A 座,邮编 100084)

责任编辑:宋 韬

印 刷 者:北京市清华园胶印厂

发 行 者:新华书店总店北京科技发行所

开 本:787×1092 1/16 印张:23.25 字数:537 千字

版 次:2001 年 4 月第 1 版 2001 年 8 月第 2 次印刷

书 号:ISBN 7-302-04308-6/TP·2532

印 数:5001 - 10000

定 价:39.00 元

读者选购指南

选购一本电脑书从来就不是件轻松的事,因为同类书实在是琳琅满目,难以取舍。购电脑书的第一项是看看书的起点或者说是读者对象,不要买太浅或者太深的书。本书面向有基本的过程式程序设计经验(如 Basic)的读者,他们不了解或者不太了解面向对象的理论和相关的程序设计语言如 C++、VB,对 Java 语言也没有经验或者经验很少。如果这是您的起点,那么您就已经具备了看懂本书所需要的预备知识,而如果您在找一本面向对象的理论和 Java 程序设计的入门教材,那么本书就是您理想的选择。

现在的程序设计语言主要是面向对象的,内容十分庞大(这是功能强大的结果),任何一本书不可能全面详细地讲述,而如果有这样一本书,它一定有至少几千页之厚。几乎所有的电脑书都可以分为入门和深入两大类,Java 也不例外。它的较深入的著作大部分都是面向某一专题的。作为入门书,本书在内容的选择上颇费思量,最后把它划分为三大部分,第一部分是前两章,第 1 章用很少的篇幅让您对 Java 有个大体上的了解。第 2 章讲述 Java 的基本语法。第 3 章到第 4 章讲述面向对象的理论在 Java 中的实现技术,这是相当详细、深入和完整的内容,作者所以这样写是因为注意到了 Java 的专题大部头著作在这方面不会过多涉及,而一般的入门书又讲得太浅,所以,既然本书把入门与深入的内容全部讲述到了,用句套话说就是填补了市场空白,满足了读者的需要。

第 5 章到第 13 章讲述了常见的程序设计主题,或者说是讲述了有关的类库(类似其他程序设计语言中的函数库)。考虑到本书设定的读者群没有太多面向对象的程序设计经验,因此所选的例子大都短小精悍并且讲解得很详尽。对相似的内容或者功能(例如三角函数),通常只详细讲解其中的一二(如正弦函数),这样本书就能最大限度地让读者了解 Java 语言的全貌,从而为阅读大部头著作打下一个坚实的基础。

第 14 章即最后一章介绍了 Java 在程序设计语言方面的强势地位,即 Java 好在哪里。本章的内容通常是其他参考书的第 1 章,如果希望了解 Java 的功能,读者也可以先读读本章。

本书可作为高校在校的学生、网络技术开发人员,希望了解和详细学习面向对象技术和 Java 实现技术的人员,Java 程序设计的新手的选择。甚至也可作为希望学习 C++ 的读者的预备读物,因为 Java 源自 C++,但比 C++ 更容易理解和学习。

上面讲述的是本书的读者对象和内容概貌,接着作者要解释的是本书如何组织和讲解内容,即本书的写作原则或者指导思想。

任何一本书的作者写书时都有指导思想,无论他是否觉察到或者这个指导思想是否恰当。本书除了具备科技著作的起码要求即技术讲解正确、内容全面完整外,全书的写作思想和具体的写作方法尤其具有与众不同的特色,使读者不仅看得懂、看得轻松,还看得愉快。本书的指导思想是 Know Unknown Through Known(通过已知知未知),这是法国教师 Piggysong 的名言和教育思想,意思是人类了解未知的最好方式是通过已知。

本书的具体写作策略是站在读者的角度,按照普通人的学习习惯而不是知识本身的固有体系布局谋篇,全书章节都是按照基本——常用——深入来安排内容结构,突破了传统电脑书不分主次和轻重缓急的字典式写作惯例。因为每当提到一个名词、概念或者其他知识点,作者要问自己三个问题:“读者可能学过吗?前面讲过吗?应该在这里讲吗?”,尽量保证每一句话、每一个段落的知识只涉及前面的内容而不涉及后面的内容,使读者从第一页看起,基本没有看不懂的地方,这就是不用未知解释另外一个未知。万一要提及后面才能详细讲解的知识,就使用这样的说法:“先照葫芦画瓢,后面还要详细讲解”,以避免读者产生没有讲清楚的错觉,仅此一点,就可保证读者看得不累。

本书对知识点,不管是一个专业名词还是一章一节的内容,大多评价其实用效果,以便读者知道如何在实践中运用知识,使读者为应用而学习,而不是为知识而学习。这样的例子也比比皆是,例如“评论”图标后的内容。

本书对知识点基本采用先举例,从已知知识切入或者先以比喻的方式引入,这些都是读者已知或者很容易“已知”的,作者以为,这是最好的知识点引入方式,类似的例子也不胜枚举。

本书在语言方面,尽量地运用比喻来讲解知识点,这在科技著作中极为罕见,没有学过本书内容的读者都可以很容易地在书中找到证据。

除了运用比喻外,本书的语言叙述尽量平白、简洁、生动又不乏幽默,因为作者厌恶以“专业”术语唬人或者自我炫耀,反喜欢“化神奇为腐朽”,把艰深干涩的电脑知识通俗化、生动化、趣味化,使读者在阅读时,不仅读得懂、读得轻松,还读得愉快,甚至会发出“Java不过如此!”的感叹。

上面所述仅是本书的几个突出特点而已,其他特色就不必一一列出了,读者在阅读时,自然能够体会到。作者相信,读者只要在书店里站着读几分钟,就能够发现本书的一些特色。

本书书名为“百事通”,作者已经出版的著作也是用同一名字。“百事”说明本书的内容广泛坚实,知识含量高。书名中有一个字是“通”,正是作者写作思想所追求的目标,因为一本书,无论讲的是什么,如果读者看不懂或者很难看懂,也如废纸一般。

读者可能会注意到作者没有用“通俗易懂”、“循序渐进”、“快速入门”、“轻松上手”等来形容本书,因为所有的电脑书都这样自称,而尽管本书当之无愧,却只想用一个“通”字。对于这句“广告词”,作者还拿出诸多可判定的证据来证明读者读了后,会真“通”而不是假“通”。

其实,书名中的“百事通”源自英文 Bestbook 的发音,因为作者在刚写作时,就希望创作出最好的电脑书奉献给读者,因此以 Bestbook 作为英文书名,鞭策自己精心构思、认真写作,不辱此名。

本书的全部程序也在作者的网站上,读者可以下载下来。如果读者对本书有什么批评和建议,既可以给作者发送电子邮件,也可以在网站上进行交流。

张洪斌

bestbooks@netease.com

<http://bestbookedu.top263.net>

自序兼谈本书应用的教育思想

Know Unknown Through Known 是 19 世纪法国教师 Piggysong 的名言,意思是人类了解未知的最好方式是通过已知,即 The best way to know what is unknown is through what is known。我是在上大学时,从图书馆借的一本英文名人传中知道这句话的,记得那是一本发黄的书,好像是解放前的馆藏。我能把这句话记到现在不是因为我当时明白了其中的深邃思想,而是这句英文的优美特点:只有 4 个单词,并且其中的 3 个都是 Know。我曾竭力地想把它翻译成旗鼓相当的中文,可是至今都没有完全成功,我所做的最好译文是“通过已知知未知”,意思表现出来了,而且与英文一样也有 3 个“知”,但却用了 7 个汉字(一般认为汉语译文比英文原文篇幅要少)。

那本英文传记介绍说,作为教师,Piggysong 非常善于教学,他的学生总是能以最快的速度掌握知识,他用的就是 Through Known 的原则。例如,他大量地使用比喻,把生活中学生已经了解的事物与要学习的专业知识比较。在做试验时不是像现在的教师那样先讲原理,而是先做试验,再讲原理。在教学顺序上,他按照内容的深浅程度而不是知识固有的体系安排,虽然被同行批评教授的知识显得缺少体系和章法,但是他的学生学得快学得好却是不争的事实,不然怎么上了名人传,让我这个几百年以后的学生也知道他的大名呢?

当时我在看传记时,对他的思想并没有留下太多的感触,因为自己当初并非教育工作者,只是刚上大学的学生而已,况且他的教学方法就是由简到繁,由浅入深,由具体到抽象之类,这些大家不都知道吗?很 plain(平白)吗!我学的是计算机专业,得听课还得看书,自然地感到同样的内容,有些老师和有些书就比别的老师和教材讲得好,讲得更容易懂。工作以后,主要是通过看书来提升自己,逐渐感到现在的电脑书越来越难看了,一开始认为也许是电脑技术进步的“恶果”,可是后来逐渐发现并非完全如此。例如很多书的第一章都是告诉你这个软件有什么强大的功能,新版本比老版本有什么改进,其中会涉及一大堆后面章节才应讲解的术语,看得人头皮发麻。试问,还没有学它如何明白它的强大之处?没有用过旧版本何以明白新版本高明的地方?

笔者至今还记得自己刚学习“文件”这个概念时的艰难历程。当时笔者刚学电脑,书中告诉我:文件用于记录电脑信息(大意)。我不知道文件是什么,当然也不知道信息是什么,用我不理解的名词来解释另一个不理解的名词岂不是雪上加霜?接着开始讲解文件的命名法,我不认识老虎却要认老虎这两个字干吗?然后书中又告诉我,文件有可执行文件、文本文件等类型,既然我不知道什么是老虎又怎么知道什么是东北虎、华南虎呢?书中讲完了全部的文件概念才开始学习 DOS 命令,这时才明白什么是文件,可是都已经“学”了很多页了。如果一边学习 DOS 命令一边讲解文件的概念岂不是轻松得多,例如要讲解可执行文件,只要运行一个游戏文件就可以了。作者也注意到目前很多 Windows 的书也是先讲文件的概念,再讲文件操作的。

我在专业上学习的同时,也在研究电脑书的写作方法,慢慢地发现电脑书写得差的根本原因就是:应该在后面讲解的知识在讲解前出现,就像笔者在前面举的例子一样。于是我就想起了 Piggysong 和他的名言,尽管我当时认为多么 plain,多么 simple,可是不是有很多教育工作者包括他们写的书都没有运用吗?为了具体地实践 Piggysong 的思想,我也写了本电脑书,即《Internet 网虫百事通》,并且“大言不惭”地用 bestbook 作为英文书名,即 The Bestbook For Growing Of Internet Spiders。在清华大学出版社出版后销路很好,已经重印数次并且登上了北京图书大厦(<http://www.bjbb.com>)的排行榜,现在已经近一年了,仍然在榜上。这给了我极大的鼓励,不仅还敢于把这本书叫 bestbook,也通过本书,正式公布 Piggysong 的教育思想,期望能对其他电脑书作者和教育工作者能有参考价值,促进书店里出现更多看得懂,看得轻松,看得愉快的好书。

Piggysong 不是个太有名的人,甚至不能作为教育家流传后世,也许是因为他不太“善于”总结自己的思想,如果把自己的思想叫做“XXX 思想”或者“XXX 主义”(例如目前最流行的“建构主义”教育理论),名字最好如雷贯耳又莫名其妙,并且拌有枯燥加令人困惑的长篇大论,这样后人才会不知疲倦,兴致勃勃地研究和探讨。他的 Know Unknown Through Known 太白了,太容易理解了,根本不需要什么人做进一步的发挥和解释。不过,笔者认为,这正是一个教育工作者的追求:授业的目的不正是解惑,并且解得越快越好吗?

最后,作者谨把此书献给赵嵩,她的鼓励和建议始终是作者从事写作和教育研究的动力,也是本书得以问世的主要原因。

张洪斌
2001 年 2 月

目 录

第1章 Java 快速入门	(1)
1.1 Java 的故事	(1)
1.2 JDK 的安装和设置	(2)
1.3 使用文本编辑器输入和编辑源程序	(2)
1.4 编译	(3)
1.5 独立应用程序的运行	(4)
1.6 Applet 小程序的运行和测试	(4)
1.6.1 在网页上运行小应用程序	(5)
1.6.2 测试小应用程序	(6)
1.7 Windows 应用程序的运行	(7)
1.8 Java 程序结构分析	(8)
1.9 本章小结	(9)
第2章 Java 基本语法	(10)
2.1 注释	(10)
2.2 标识符和保留字	(11)
2.3 基本数据类型	(11)
2.3.1 整数类型和整数变量的声明	(11)
2.3.2 浮点型和浮点变量的声明	(12)
2.3.3 字符型和字符变量的声明	(12)
2.3.4 布尔型和布尔变量的声明	(13)
2.4 符号常量	(13)
2.5 运算符	(14)
2.5.1 数学运算符	(14)
2.5.2 赋值运算	(15)
2.5.3 类型转换	(16)
2.5.4 自增与自减运算符	(17)
2.5.5 比较运算符	(18)
2.5.6 逻辑运算符	(19)
2.5.7 位运算符	(21)
2.5.8 移位运算符	(21)
2.6 程序流程控制语句	(21)
2.6.1 if 语句	(21)

2.6.2	switch 语句	(22)
2.6.3	while 语句	(24)
2.6.4	do while 语句	(24)
2.6.5	for 语句	(25)
2.7	本章小结	(26)
第3章	面向对象的程序设计(I)	(27)
3.1	类的定义和使用	(27)
3.1.1	最简单的类	(27)
3.1.2	类的数据	(28)
3.1.3	类的函数(方法)	(28)
3.1.4	完整的类的例子	(29)
3.1.5	创建类的对象和运行程序	(30)
3.1.6	程序代码的书写格式	(32)
3.1.7	小结	(32)
3.2	对象、引用和作用域	(33)
3.2.1	对象和引用	(33)
3.2.2	变量的作用域	(37)
3.3	基本数据类型和对象的初始化	(39)
3.3.1	基本数据类型成员的初始化	(39)
3.3.2	初始化方法中的变量	(43)
3.3.3	对象和引用的初始化	(44)
3.4	类的构造方法和重载	(47)
3.4.1	类的构造方法	(48)
3.4.2	重载方法与默认构造方法	(49)
3.5	数组与字符串	(52)
3.5.1	数组基础	(52)
3.5.2	对象数组	(54)
3.5.3	多维数组	(57)
3.5.4	字符串	(58)
3.5.5	main 方法中的参数	(59)
3.6	类的继承、组合与重置方法	(60)
3.6.1	类的继承和重置方法	(60)
3.6.2	超类与子类不在一个目录的情况	(62)
3.6.3	在一个文件中继承类	(62)
3.6.4	为每个类添加 main 方法	(64)
3.6.5	类与其超类的构造方法的调用	(66)
3.6.6	不用继承使用另外一个类的功能	(69)
3.6.7	类作为另外一个类的成员;类的组合	(71)

3.6.8 滥用重置	(72)
3.6.9 重置与重载	(74)
3.7 本章小结	(76)
第4章 面向对象的程序设计(Ⅱ)	(77)
4.1 类的转换:超类与继承类的关系	(77)
4.2 import 语句和软件包	(78)
4.2.1 建立软件包	(79)
4.2.2 使用软件包	(79)
4.3 static 关键字	(80)
4.4 对象的初始化	(82)
4.4.1 非静态对象的初始化	(82)
4.4.2 静态对象的初始化	(84)
4.4.3 继承时的构造方法	(87)
4.4.4 静态变量初始化块	(89)
4.4.5 非静态变量初始化块	(89)
4.5 this 和 super 关键字	(90)
4.6 存取控制:public、private 和 protected 关键字	(92)
4.7 多态性	(94)
4.8 抽象类和抽象方法	(99)
4.9 final 关键字	(100)
4.10 接口 interface 与多重继承	(103)
4.10.1 接口基础	(103)
4.10.2 多重继承	(105)
4.10.3 接口的嵌入	(109)
4.10.4 接口的转换	(110)
4.11 类的嵌入(内部类)	(111)
4.11.1 内部类导引	(111)
4.11.2 继承内部类	(115)
4.12 Java 的存储结构	(116)
4.13 本章小结	(117)
第5章 通过 Java 文档了解类的使用	(118)
5.1 Java 文档的使用	(118)
5.1.1 Overview 页	(120)
5.1.2 Package 页面	(120)
5.1.3 Class 页面	(121)
5.1.4 Use 页面	(123)
5.1.5 Tree 页面	(123)
5.1.6 Deprecated API 页面	(124)

5.1.7 Index 页面	(124)
5.2 实例分析	(125)
5.2.1 实例分析一	(125)
5.2.2 实例分析二	(126)
5.3 JDK 自带的演示程序	(127)
5.4 使用 JavaDoc 生成自己的程序文档	(128)
5.5 本章小结	(131)
第6章 程序的异常处理	(132)
6.1 异常入门	(132)
6.2 基本的异常处理	(133)
6.2.1 try 和 catch	(134)
6.2.2 获得异常的有关信息	(136)
6.2.3 抛出异常	(137)
6.2.4 异常类的构造方法	(139)
6.3 在其他地方处理异常	(140)
6.3.1 在其他方法中处理异常	(140)
6.3.2 重新抛出异常	(143)
6.4 建立自己的异常类	(147)
6.5 不捕获异常时的情况	(151)
6.6 异常类的其他主题	(154)
6.6.1 Throwable 和 Error 类	(154)
6.6.2 捕获多个可能发生的异常	(156)
6.6.3 finally 关键字	(159)
6.6.4 Java 软件包中抛出异常的方法	(162)
6.7 本章小结	(162)
第7章 文件管理与输入输出	(163)
7.1 文件管理	(163)
7.2 输入与输出	(167)
7.2.1 InputStream 类	(168)
7.2.2 OutputStream 类	(171)
7.2.3 Reader 和 Writer 类	(172)
7.2.4 System 类	(173)
7.2.5 标准输入和输出的重定向	(175)
7.2.6 缓冲	(176)
7.2.7 RandomAccessFile 类	(177)
7.3 压缩类	(181)
7.4 本章小结	(185)

第8章 Java 小应用程序和多媒体程序设计	(186)
8.1 Applet 类和小应用程序的运行机制	(186)
8.2 HTML 文件与小应用程序	(188)
8.3 Java 图形设计	(189)
8.3.1 图形坐标系	(190)
8.3.2 文本输出	(190)
8.3.3 使用颜色	(191)
8.3.4 绘图	(192)
8.3.5 得到当前字体的属性	(193)
8.3.6 使用指定的字体	(195)
8.3.7 得到系统上可用的字体	(196)
8.4 显示图像与 URL 类	(198)
8.5 声音的播放	(201)
8.5.1 网页打开时自动播放声音	(201)
8.5.2 声音播放控制	(202)
8.6 本章小结	(204)
第9章 Java 构件	(205)
9.1 构件和 JApplet 类	(205)
9.2 建立构件和选择构件的布局	(207)
9.3 布局管理器	(210)
9.3.1 BorderLayout 类	(211)
9.3.2 FlowLayout 类	(212)
9.3.3 GridLayout 类	(213)
9.3.4 BoxLayout 类	(214)
9.4 Java 构件一览	(217)
9.4.1 Button 按钮	(217)
9.4.2 JCheckBox 复选框	(219)
9.4.3 ButtonGroup 按钮组	(220)
9.4.4 JTextField 文本框	(221)
9.4.5 JSlider 标尺	(222)
9.4.6 JProgressBar 进度指示器	(224)
9.4.7 文本区域构件和滚动条	(226)
9.4.8 JComboBox 下拉列表	(230)
9.4.9 JList 列表框	(231)
9.4.10 JTree 树型图	(232)
9.4.11 JTable 表格	(233)
9.4.12 JTabbedPane 标签面板和 JPanel 面板构件	(235)
9.4.13 JMenuBar 菜单	(238)

9.4.14	JPopupMenu 弹出菜单	(241)
9.4.15	JOptionPane 弹出式窗口	(242)
9.5	Java 构件的装饰	(247)
9.5.1	带图像和背景的构件	(247)
9.5.2	构件的提示文字	(248)
9.5.3	构件的边框	(249)
9.5.4	构件上使用 HTML 标记	(249)
9.5.5	在构件中绘图	(251)
9.6	本章小结	(253)
第10章	Java 构件的容器和 Windows 应用程序	(254)
10.1	JPanel 类(面板)	(254)
10.2	JWindow 和 JFrame 窗口	(255)
10.3	JDialog 对话框窗口	(258)
10.4	Windows 应用程序的设计	(261)
10.4.1	设计 Windows 应用程序	(261)
10.4.2	设计也可作为小应用程序运行的 Windows 应用程序	(263)
10.4.3	只能在 Windows 应用程序中使用的类:JFileChooser 类	(265)
10.5	本章小结	(271)
第11章	交互与事件处理	(272)
11.1	事件和交互基础	(272)
11.2	键盘事件	(276)
11.3	焦点事件	(279)
11.4	鼠标事件	(280)
11.4.1	MouseMotionListener 接口	(281)
11.4.2	MouseListener 接口	(283)
11.5	选择适当的事件和确定引起事件的构件	(284)
11.5.1	选择适当的事件类型	(285)
11.5.2	确定事件发生的构件	(289)
11.6	构件事件实例	(292)
11.6.1	窗口事件	(292)
11.6.2	单选按钮组事件	(296)
11.6.3	复选框事件	(297)
11.6.4	下拉列表事件	(299)
11.6.5	标尺事件	(300)
11.6.6	标签面板事件	(303)
11.6.7	菜单事件	(305)
11.6.8	文本事件	(308)
11.6.9	绘图事件	(310)

11.7 本章小结.....	(312)
第12章 线程	(313)
12.1 线程入门.....	(313)
12.2 图形用户界面程序中的线程.....	(318)
12.3 Runnable 接口	(321)
12.4 线程的同步.....	(322)
12.5 线程的调度和优先级.....	(329)
12.6 本章小结.....	(332)
第13章 网络编程基础	(333)
13.1 小应用程序和 URL 类	(333)
13.1.1 在小应用程序中打开一个网页	(333)
13.1.2 从 URL 上读入文件的内容	(335)
13.2 IP 地址与 InetAddress 类	(337)
13.2.1 计算机名和 IP 地址.....	(337)
13.2.2 InetAddress 类中的方法	(340)
13.3 Socket 网络编程.....	(343)
13.3.1 建立连接	(344)
13.3.2 Socket 的通讯	(347)
13.3.3 多线程的服务器程序	(350)
13.4 本章小结.....	(355)
第14章 Java 的优势	(356)

第 1 章 Java 快速入门

作为全书的第 1 章,本章首先简单介绍 Java 的历史,然后是编辑、编译、运行 Java 程序的方法,最后通过几个小例子,演示几种 Java 程序。本章的目标是让初学者对设计 Java 程序有个完整的(虽然是粗略的)感知认识。

1.1 Java 的故事

几乎每个使用电脑的人,都知道 Internet 是当今的时尚,也或多或少地知道 Java 语言在 Internet 开发上的重要地位,但是也许不知道,Java 最初的用途并不是在 Internet 上,甚至 Java 这个名字也不是原名。

Java 的产生大约在 1990 年左右,当时用 C++ 开发家电产品的软件时,发现这个编程语言的“大哥大”并不能胜任,于是开发者以 C++ 为基础设计了一种新的程序设计语言,命名为 Oak。Oak 先被用于家用电器控制系统中,可以通过手持设备控制家庭中的电视、录像机等电器,后来又被用于 VOD 视频点播控制系统中,慢慢地 Oak 系统成熟了起来,不过却发现 Oak 这个名字已经被注册了,于是 Oak 的开发者 Sun 公司就起了 Java 这个名字。以后随着 Internet 的普及,Sun 公司发现 Java 非常适合 Internet 编程,就提出小应用程序 Applet 的概念,这是可以嵌入到网页中运行的程序。随着 Java 的不断完善,Sun 公司在 1996 年前后,发布了 Java 语言的正式版本。Java 以其强大的功能迅速成为网络时代的首选编程语言,被称为网络时代的 C++,在国外非常受宠。

实际上,Java 不仅在网络开发上十分擅长,设计一般的应用程序也丝毫不比其他语言逊色。您可以在其他 Java 参考书中找到很多描述 Java 特点的形容词:面向对象、平台无关性、分布式、健壮、安全、可扩展、多线程等等。对于初学者而言,要在还没有学会 Java 前明白这些特点的含义是不可能的,作者也不做这种无谓的尝试。只是希望读者记住这样一句话:Java 是真正跨平台、纯粹的面向对象,适合单机和 Internet 开发的编程语言。跨平台指它可以在不同操作系统上运行。而面向对象则是目前程序设计的主流思想,因为它是新型语言,所以能够成为纯粹的面向对象的语言,像 C++ 这种“历史悠久”的语言因为要不得不与以前的版本兼容,所以也就不得不包含进很多非面向对象的内容,从而使程序的设计和维护较困难。

在国内,Java 受到较大的关注也是最近一年左右的事情,以此可想象的到国内 IT 与国外的差距之大。

 **提醒:**本书的最后一章介绍了 Java 的特点,如果愿意,读者现在也不妨看看。

1.2 JDK 的安装和设置

使用任何语言前,必须要安装有关的开发软件,虽然能够开发 Java 的工具不止一个,但是一般建议初学者最好用最正宗并且还是免费的软件,这就是 Sun 自己的开发软件。

Sun 的开发软件称为 JDK(Java Developing Kit),可以从 Sun 的网站<http://java.sun.com>上免费下载,目前可下载的最新版本为 JDK1.3 或者 JDK1.2.2。

下载完成后,像安装其他软件那样安装它,可以使用默认文件夹即 c:\jdk1.3 或者 c:\jdk1.2.2。安装完成后,顺便浏览一下文件夹中的内容,其中的 bin 文件夹存放的是 Java 可执行文件,建议读者把它的路径(例如 c:\jdk1.3\bin)加入到 autoexec.bat 文件的 path 路径中。

其他的开发工具还有微软的 Visual J++、Imprise 的 J++ Builder 等。

1.3 使用文本编辑器输入和编辑源程序

像您学过的任何高级语言一样,Java 源程序也是文本文件,它的后缀为 java,下面是一个最简单的程序,文件名是 MyFirstProgram.java,请注意代码中的黑体字与文件名是相同的(包括大小写),这是 Java 语法的规定,详情我们会在后面的章节中描述:

```
//c1:MyFirstProgram.java
//author:ZhangHongbin
//This program is protected by copyright laws.
//The first DOS program.
public class MyFirstProgram
{
    public static void main(String[] args)
    {
        System.out.println("这是我写的第一个程序");
        //下面的程序段是用来暂停程序的。
        try
        {
            System.in.read();
        }
        catch(Exception e) {}
    }
}
```

读者可以用 Windows 的记事本程序输入该程序,不过很不方便,作者建议使用 Ultra-Edit 文本编辑器或者其他功能强大的文本编辑器。如果您自己用键盘输入这段小程序,请一定与本书保持一致,尤其要注意大小写、单词的拼写、分号和各种括号。本程序(实际

上是本书的所有程序清单)都可以从作者的网站上下载。

本程序的输出结果是在 DOS 窗口中显示“这是我写的第一个程序”后,按一下回车键,程序才能结束。

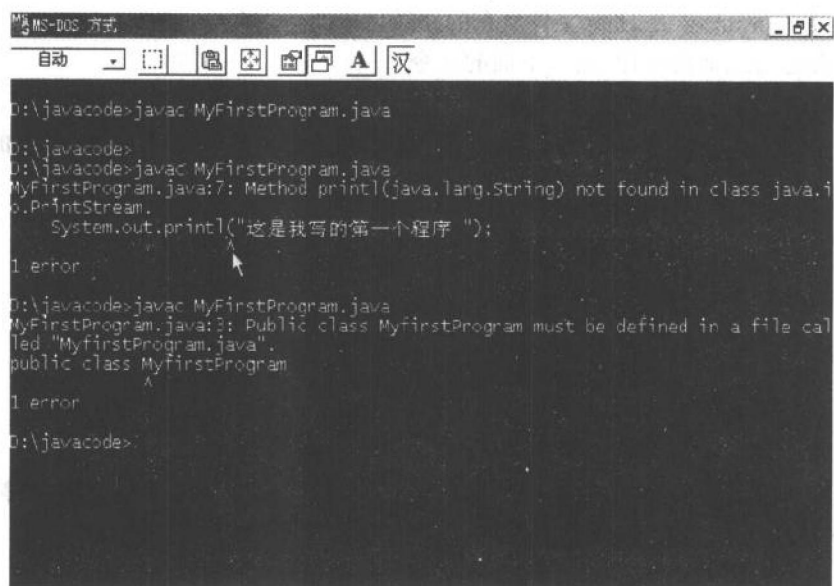
1.4 编译

源程序输入完毕后,可仔细检查一下,然后进行编译。

首先进入 MS-DOS 方式,然后切换到源程序所在的文件夹如 d:\javacode,用键盘输入下面的命令:

```
javac MyFirstProgram.java
```

javac.exe 是 JDK 的编译程序,其中 javac 中的字母 c 代表 compile(编译),一定要注意文件名的大小写不能错。图 1.1 所示为几次编译程序的结果。



```
MS-DOS 方式
自动
D:\javacode>javac MyFirstProgram.java
D:\javacode>
D:\javacode>javac MyFirstProgram.java
MyFirstProgram.java:7: Method println(java.lang.String) not found in class java.i
o.PrintStream.
    System.out.println("这是我写的第一个程序 ");
1 error
D:\javacode>javac MyFirstProgram.java
MyFirstProgram.java:3: Public class MyfirstProgram must be defined in a file cal
led "MyfirstProgram.java".
public class MyfirstProgram
1 error
D:\javacode>
```

图 1.1 编译结果

如果程序没有编译错误,那么不会产生任何输出,直接出现提示符。

图 1.1 中共进行了 3 次编译,第一次没有错误。

第二次将 println 的字母 n 去掉,结果出现一个错误。错误输出依次为:

- 文件名
- 数字 7 是错误发生的行号
- 有关该错误的说明
- 显示错误所在的语句
- 错误总数

第三次编译出现的错误是把 MyFirstProgram 中的 F 小写了,因此就与文件名不一致。