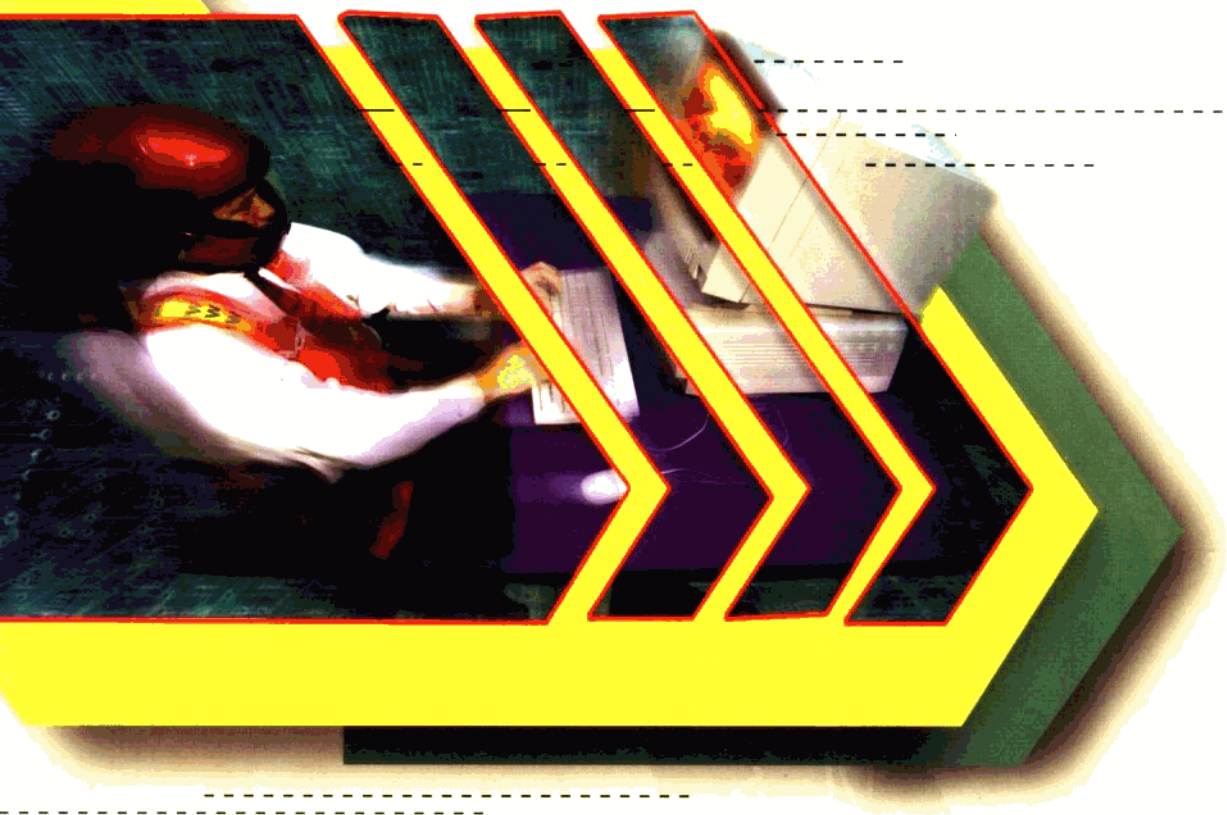


高职高专计算机系列教材

主编 谭浩强

Java 语言 程序设计 (第二版)

邵丽萍 邵光亚 张后扬 编著



清华大学出版社



编辑委员会

《高职高专计算机系列教材》

主 任 谭浩强
副 主 任 陈 明 丁桂芝

委 员 (按姓氏笔画排序)

王智广	刘 星	刘荫铭	安志远
安淑芝	孙 慧	李文英	李叶紫
李 琳	李雁翎	秦建中	陈 强
邵丽萍	尚晓航	张 玲	侯冬梅
郝 玲	赵丰年	莫治雄	袁 玫
訾秀玲	薛淑斌	谢树煜	



《高职高专计算机系列教材》

进入 21 世纪,计算机将成为人类常用的现代工具,每一个有文化的人都应当了解计算机,学会使用计算机来处理面临的事物。

学习计算机知识有两种不同的方法:一种是侧重知识的学法,从原理入手,注重理论和概念;另一种是侧重应用,从实际入手,注重掌握其应用的方法和技能。不同的人应根据其具体情况选择不同的学习方法。对多数人来说,计算机是作为一种工具来使用的,应当以应用为目的,以应用为出发点。对于高职和高专的学生,显然应当采用后一种学习方法。

传统的理论课程采用以下的三部曲:提出概念——解释概念——举例说明,这适合前面第一种方法。对于侧重应用的学习者,我们在教学实践中摸索出新的三部曲:提出问题——介绍解决问题的方法——最后归纳出一般规律或概念。实践证明这种方法是行之有效的,减少了初学者在学习上的困难。传统的方法是:先理论后实际,先抽象后具体,先一般后个别。我们采用的方法是:从实际到理论,从具体到抽象,从个别到一般,从零散到系统。我们认为这种教学方法更适合于高职高专和成人高校。

本套教材是针对高职和高专的特点组织编写的,包括了高职高专的计算机专业和非计算机专业的教材和参考书。不同专业可以从中选择所需的部分。本系列教材包含的内容比较广,除了可作为正式教材外,还可作为某些专业的选修课或指定自学的教材。

应当指出:检查学习好坏的标准,不是“知道不知道”,而是“会不会用”,学习的目的全在于应用。因此希望读者一定要重视实践环节,多上机练习,千万不要满足于“上课能听懂、教材能看懂”。有一些问题,别人讲半天也不明白,自己一上机就清楚了。教材中有些实践性比较强的内容,不一定在课堂上由老师讲授,而应指定学生通过上机掌握这些内容。这样做可以培养学生的自学能力,启发学生的求知欲望。

本系列教材自 1999 年出版以来,已陆续出版了 30 余种,受到全国各高校广大师生的欢迎,认为对提高高职高专的教学质量起了积极的作用。由于近

年来高职教育发展迅速,各地区、各学校的情况差别较大,根据各校老师的要求,我们对本套教材进行了重新安排,包括3个系列:(1)继续组织好原有教材的编写出版,对象是基础较好、要求较高的学校和专业;(2)组织编写出版一批“实用教程”,以满足理论要求相对低一些或学时较少的学校的需要;(3)组织若干种“实训教材”,其特点是:侧重实践环节,引导学生通过自己的实践(而不是通过理论讲授)去获取知识,掌握应用。我们认为这应该是高职高专教学改革的一个重要方面。

本套教材是由一些普通高校和高职高专的老师们编写的,他们对高职高专的教学特点有较多的了解,也有较多的实践经验,保证了教材的质量,使本套教材受到全国各校的好评。

由于我国的高职教育正在蓬勃发展,许多问题有待深入讨论,新的经验将会层出不穷,对如何进行高职教育将会有更新更深入的认识,本套教材的内容将会根据新的形势不断丰富和调整。我们只是为了满足许多高职高专学校对教材的急需,编写了这套教材,以期抛砖引玉。教材中肯定会有不足之处,请专家和读者不吝指正。

全国高等院校计算机基础教育研究会会长

谭浩强

2004年1月

第二版前言

Java 是目前推广速度最快的程序设计语言,它采用面向对象编程技术,功能强大而又简单易学,深受广大程序设计人员的喜爱。Java 伴随着 Internet 问世,随着 Internet 的发展而成熟。Java 是精心设计的语言,它具有简单性、面向对象性、平台无关性、安全性和健壮性等诸多特点,内置了多线程和网络支持能力,可以说它是网络世界的通用语言。为了迎接信息时代的挑战,学习和掌握 Java 语言无疑会带来更多的机遇。

面向对象编程技术为软件开发带来了一场深刻的变革,但由于涉及的内容较多,学习起来也困难得多。与传统的面向过程编程语言相比,面向对象编程语言一方面提出了很多新概念,需要学习者了解和掌握;另一方面提供了大量的类库和方法,程序设计人员只有在熟悉它们之后,编写程序时才能得心应手。Java 是标准的又是大众化的面向对象编程语言,其简单易学性体现在对象的设计和方法的使用上,初学者很容易接受面向对象的概念和设计方法,可以很快编写出合格的面向对象程序来解决一些简单问题。一些抽象的很难理解的内容,如继承、多态、异常、多线程等,在 Java 中都得到了明显的简化。因此,无论是作为面向对象编程语言还是作为网络编程语言来学习,Java 都是最佳选择,不仅适合初学编程者也适合具有一定编程经验者学习。

由于本书第一版受到读者的欢迎,而且 Java 在不断发展,为了体现 Java 的新特点,满足读者的需要,所以对本书进行改写。作为一本教材,本书第二版对 Java 语言的基本内容做了详细介绍,并在有关章节内穿插介绍了一些常用的类库和方法。在内容的编排上体现了新的计算机教学思想和方法,以“提出问题——解决问题的方法——归纳必要的结论和概念”的方式介绍 Java 编程思路,通过大量的实例和插图,使读者尽可能快地熟悉基本概念和掌握基本编程方法。

在内容上第二版有了很大改变,对原来的内容进行重新整理,并添加了新的例子,所有的程序都是使用最新的 Java 开发工具完成的。特别增加了第 12 章综合实例,以帮助读者提高开发 Java 程序的能力。第 1 章是 Java 快速入门,全面介绍了 Java 语言的基本情况,引导读者进入 Java 世界,指导读

者使用最新的 Java 工具。第 2 章和第 3 章介绍了 Java 语法和语句构成,这是必须要掌握的基本内容。第 4 章和第 5 章是有关 Java 面向对象编程的内容,深入浅出地介绍了面向对象编程的核心:对象、类以及相关概念。第 6 章介绍了包、接口和异常,读者可以了解到这些抽象概念并不是难以理解的。第 7 章介绍了常用系统类,因为面向对象编程语言难学的一个原因就是有太多的类库和方法,要花很大精力来掌握。第 8 章和第 9 章介绍了图形用户界面的构件,可利用这些构件进行程序界面设计,编写方便适用的窗口和界面。第 10 章介绍了多线程技术,使用多线程技术可以编写许多实用的程序。第 11 章是 Java 轻松的一面,读者将在这里学习如何画图、如何显示图像、如何设计动画以及如何播放数字音频。第 12 章介绍了 Java 的一些综合实例,体现了前面各章内容的综合使用,会对读者学习起到承上启下的作用。学完之后,读者将会发现这些看上去神秘的东西原来并不难掌握。

本书由邵丽萍编写第 5、第 10、第 11、第 12 章,邵光亚编写第 6、第 8、第 9 章,张后扬编写第 3、第 4 章,陆军编写第 1、第 2 章,帅零编写第 7 章。最后由邵丽萍统稿。

作者

2004 年 4 月

目录

第 1 章 Java 快速入门	1
1.1 Java 简介	1
1.1.1 Java 的起源与发展	1
1.1.2 Java 语言特点	1
1.1.3 Java 的用途	4
1.2 Java 的运行环境	5
1.2.1 SDK 的下载与安装	6
1.2.2 配置 Java 运行环境	9
1.3 初步了解 Java 程序	14
1.3.1 编写第一个 Java 源程序	14
1.3.2 Java 程序的编译	15
1.3.3 运行 Java 程序	15
习题	17
第 2 章 Java 基本语法	18
2.1 Java 语言的基本元素	18
2.1.1 标识符	19
2.1.2 关键字	20
2.1.3 分隔符	20
2.2 数据类型	21
2.2.1 基本数据类型	22
2.2.2 直接量与常量	22
2.2.3 变量	24
2.3 运算符与表达式	27
2.3.1 赋值运算符	27

2.3.2	算术运算符	27
2.3.3	关系运算符	28
2.3.4	条件运算符	29
2.3.5	逻辑运算符	29
2.3.6	位运算符	30
2.3.7	其他运算符	32
2.3.8	运算优先级	32
2.4	数组	33
2.4.1	一维数组的声明	33
2.4.2	一维数组的创建与赋值	33
2.4.3	多维数组	35
	习题	35

第3章 Java 语句及其控制结构 38

3.1	Java 程序结构	38
3.1.1	Java 程序构成	38
3.1.2	Java 语句	39
3.2	选择语句	40
3.2.1	if 语句	40
3.2.2	if...else 语句	41
3.2.3	if...else 复合结构	41
3.2.4	switch 开关语句	43
3.3	循环语句	46
3.3.1	for 循环语句	46
3.3.2	while 循环语句	47
3.3.3	do...while 循环语句	49
3.3.4	循环语句的嵌套	50
3.3.5	循环语句小结	50
3.4	跳转语句	51
3.4.1	break 语句	51
3.4.2	带标号的 break 语句	52
3.4.3	continue 语句	52
3.4.4	带标号的 continue 语句	53
3.4.5	return 语句	55
	习题	56

第 4 章 面向对象编程	58
4.1 面向对象基本概念	58
4.1.1 对象	58
4.1.2 封装	59
4.1.3 消息	59
4.1.4 类	59
4.1.5 继承	60
4.1.6 接口	60
4.2 Java 的类与对象	62
4.2.1 类的创建	63
4.2.2 对象的创建	66
4.3 成员变量与封装	71
4.3.1 成员变量的声明	71
4.3.2 成员变量的修饰	72
4.4 成员方法	77
4.4.1 成员方法的设计	77
4.4.2 成员方法的声明与修饰	78
4.4.3 方法体	81
4.4.4 消息传递	82
4.4.5 Java 编程小结	84
习题	85
第 5 章 类的继承性与多态性	88
5.1 类的继承	88
5.1.1 父类和子类	88
5.1.2 成员变量的继承和隐藏	89
5.1.3 成员方法的覆盖	91
5.1.4 this 和 super	93
5.2 类的多态	96
5.2.1 成员方法的重载	96
5.2.2 构造方法的重载	98
习题	100

第 6 章 包、接口和异常	103
6.1 包	103
6.1.1 Java 的系统包	104
6.1.2 包的引用	107
6.1.3 声明自定义包	107
6.2 接口	110
6.2.1 实现系统提供的接口	110
6.2.2 创建自定义接口	112
6.2.3 接口的继承关系	113
6.3 异常处理	114
6.3.1 什么是错误与异常	114
6.3.2 异常发生的原因	115
6.3.3 异常处理机制	115
6.3.4 异常对象类型	116
6.3.5 异常的处理	117
6.3.6 throw 语句	120
6.3.7 throws 子句	121
6.3.8 finally 语句	123
6.3.9 编译时对异常情况的检查	124
6.3.10 创建自己的异常类	125
习题	126
第 7 章 常用系统类	129
7.1 Applet 类	129
7.1.1 Applet 的特点	129
7.1.2 Java 和 HTML 的关系	130
7.1.3 Applet 的生命周期与主要活动	131
7.1.4 HTML 和 Applet 的参数传递	133
7.1.5 Applet 标签属性	135
7.2 字符串类	137
7.2.1 字符串与字符串类	137
7.2.2 字符串类的构造方法	139
7.2.3 String 类的应用	140
7.2.4 StringBuffer 类的应用	143
7.3 输入输出流类	145

7.3.1	数据流概念	145
7.3.2	Java 的标准数据流	146
7.3.3	java.io 包中的数据流类及文件类	147
7.3.4	数据流的应用	150
7.4	其他常用类	152
7.4.1	数学函数类 Math	152
7.4.2	日期类	153
7.4.3	随机数类 Random	155
7.4.4	向量类 Vector	156
7.4.5	Class 类与 Runtime 类	158
	习题	159

第 8 章 图形用户界面..... 161

8.1	组件	161
8.1.1	标签	161
8.1.2	按钮	162
8.1.3	选项框	163
8.1.4	复选框和选项按钮	164
8.1.5	列表框	167
8.1.6	文本框	168
8.1.7	文本区	169
8.2	组件布局管理	170
8.2.1	顺序布局	170
8.2.2	边界布局	170
8.2.3	卡片布局	171
8.2.4	网格布局	172
8.2.5	面板的使用	173
8.2.6	手工布局	175
8.3	事件处理方法	176
8.3.1	事件与事件处理方法	176
8.3.2	Java 的事件处理机制	176
8.3.3	事件处理实例	178
8.3.4	事件适配器类与类的包容	184
	习题	186

第 9 章	窗口、菜单和对话框	187
9.1	窗口	187
9.1.1	创建可关闭窗口	187
9.1.2	窗口事件适配器	188
9.1.3	匿名类	189
9.1.4	在窗口中加入组件	190
9.1.5	多重窗口	191
9.2	菜单	192
9.2.1	为窗口加入菜单	192
9.2.2	菜单综合应用	194
9.2.3	弹出式菜单	196
9.3	对话框	198
9.3.1	自定义对话框	198
9.3.2	创建自定义对话框实例	198
9.4	Swing 简介	201
9.4.1	Swing 按钮与标签	201
9.4.2	Swing 工具栏	203
	习题	205
第 10 章	多线程机制	206
10.1	多线程的概念	206
10.1.1	程序、进程和多任务	206
10.1.2	线程	206
10.1.3	多线程	207
10.1.4	线程的生命周期与 Java 的多线程机制	207
10.2	创建线程	208
10.2.1	通过继承 Thread 类创建线程	208
10.2.2	通过 Runnable 接口创建线程	209
10.3	线程的优先级	212
10.4	线程的调度与控制	213
10.4.1	线程类的方法	213
10.4.2	控制线程的状态	214
10.5	线程的同步机制与共享资源	215
10.6	何时使用多线程及注意问题	220
	习题	221

第 11 章 图形与多媒体处理	222
11.1 基本图形	222
11.1.1 直线	222
11.1.2 矩形	223
11.1.3 椭圆	225
11.1.4 圆弧	225
11.1.5 多边形	226
11.2 画布	227
11.2.1 画布的作用	227
11.2.2 创建画布实例	227
11.3 文字输出	228
11.3.1 字符串、字符和字节输出	228
11.3.2 字体控制	230
11.4 颜色控制	231
11.4.1 为不同对象设置颜色	231
11.4.2 不同颜色的文字	233
11.5 图像处理	234
11.5.1 图像种类	234
11.5.2 图像的显示	234
11.5.3 图像的缩放显示	235
11.6 动画处理	236
11.6.1 动画原理	236
11.6.2 用线程实现动画	238
11.7 加载声音文件	240
11.7.1 添加背景音乐	240
11.7.2 随时播放声音文件	241
习题	243
第 12 章 综合实例	244
12.1 数值变换运算	244
12.2 幻灯机效果	245
12.3 利用滚动条改变背景颜色	246
12.4 Applet 与 Application 合并运行的程序	247
12.5 对象的克隆	249

12.6	正弦曲线	251
12.7	四叶玫瑰曲线	252
12.8	模拟网上购物结算	253
12.9	数字和图形时钟	256
12.10	在画布上手工画图	258
12.11	电闪雷鸣的动画	260
12.12	文字的动画显示	263
12.13	控制移动的文字	264
12.14	水中倒影	266
12.15	访问其他网站	268
12.16	输入保存信息窗口	269
12.17	简单的文本编辑器	272
	提示	275
	习题	275

iv	参考文献	277
----	------	-----

第1章

Java 快速入门

Java 是非常具有吸引力的面向对象编程语言,又是当前最流行的网络编程语言。Java 的出现引起了软件开发的变革,为迅速发展的 IT 业增添了新的活力。近两年 Sun 公司推出的 J2ME、J2SE、J2EE 三大平台显现了 Java 语言强大的应用能力,从高性能计算,到移动计算(手机通信)、智能卡,到处都有 Java 的身影。

本章的主要任务是了解 Java 的基本特点和用途,掌握如何下载 Java SDK 软件包,如何设置 Java 程序的开发环境 SDK,如何运行 Java 程序,了解 Java 的概貌,为后续各章的学习做好准备。

1.1 Java 简介

1.1.1 Java 的起源与发展

Java 是一种计算机程序语言,用 Java 既可以编写嵌入在 Web 网页中运行的 Java Applet 小程序,也可以编写独立运行的 Java Application。Java 是当今十分流行的网络编程语言。

Java 是 Sun 公司于 20 世纪 90 年代初开发的。最初并不是为了用于 Internet,而是作为一种小家用电器的编程语言,用来解决诸如电视机、电话、闹钟、烤面包机等家用电器的控制和通信问题,命名为 Oak。由于这些智能化家用电器的市场需求当时没有预期的高,Sun 放弃了该项计划。就在 Oak 几近夭折之时,Internet 异常火爆起来。Sun 看到了 Oak 在计算机网络上的广阔应用前景,他们改造了 Oak,于是 Java 诞生了。

1995 年 5 月 Sun 正式发布了 Java。由于 Internet 上存在着巨大的商业利益,Java 的出现引起了商界的极大兴趣。作为专为商业用途设计的程序语言,Java 伴随着 Internet 的迅猛发展而发展,逐渐成为重要的 Internet 编程语言。由于 Java 提供了强大的图形、图像、动画、音频、视频、多线程及网络交互能力,使它在设计交互式、多媒体网页和网络应用程序方面大显身手,成为当今推广速度最快的一门计算机程序语言。

1.1.2 Java 语言特点

按照 Java 设计者的解释,Java 是一个简单、面向对象、网络适用、解释型、健壮、安全、

结构中立、可移植、高性能、多线程、动态的计算机程序语言。

1. 简单性

设计 Java 语言的出发点就是容易编程,不需要深奥的知识。Java 语言的风格十分接近 C++ 语言,但要比 C++ 简单得多。Java 舍弃了一些不常用的、难以理解的、容易混淆的成分,如运算符重载、多继承等,增加了自动垃圾搜集功能,用于回收不再使用的内存区域。这不但使程序易于编写,而且大大减少了由于内存分配而引发的问题。

简单性还体现在小上。Java 解释器、系统模块和运行模块都比较小,适合在小型机器上运行,也适合从网上下载。

2. 面向对象

面向对象编程是一项有关对象设计和对象接口定义的技术,或者说是一项如何定义程序模块才能使它们“即插即用”的技术。Java 程序可以看成是一个对象,对象通过继承和重定义,成为解决新问题的新程序模块,使程序代码重用有了可能。

3. 网络适用性

Java 提供了大量的系统模块支持基于 TCP/IP 协议的编程,这使得 Java 建立网络连接要比 C/C++ 容易得多。Java 程序通过 URL 访问网络资源像存取本地文件系统一样简单。

4. 健壮性

Java 程序的健壮性从多方面得到了保证。Java 提供早期的编译检查和后期的动态(运行期)检查,大量消除了引发异常的条件。Java 程序在没有授权的情况下是不能访问内存的。所有这些措施,使 Java 程序员不用再担心内存的崩溃,因为根本就不存在这样的条件。

5. 安全性

Java 主要用于网络和分布式环境,采取了很多措施来加强系统的安全性。Java 可以组建病毒无法入侵和无法篡改的系统,其数字验证使用了基于公共密钥的技术。

安全性和健壮性密切相关。Java 的指针类技术杜绝了非法存取数据结构或关键对象属性的可能,关闭了病毒发起攻击的大门。

6. 平台无关性

Java 最重要的特点就是独立于硬件设备的平台无关性,使 Java 程序无须任何修改就可运行在支持 Java 的任何计算机上。平台无关性是 Java 设计伊始的指导思想,也使 Java 相对于其他成熟语言发生了革命性变化,为 Java 能够在多平台的网络上运行创造了可行条件。

7. 可移植性

结构中立构成了程序可移植性的基础。另一方面,很多语言的基本数据类型长度都

有平台依赖性,而 Java 则采取固定长度。例如整数类型 int 的长度固定为 32 位,双精度类型 double 的长度固定为 64 位。

Java 的类库提供了可移植的接口。例如,类库中有一个抽象类 Window,它适用于 UNIX、Windows 和 Macintosh 等系统。

8. 解释型

Java 是解释执行的。程序运行时,字节码通过 Java 虚拟机被直接翻译成本地机器指令,中间没有存储。由于模块连接是步进的和多线程的,执行速度可以很快。

9. 高性能

设计字节码时已经把机器码的翻译问题考虑进去了,所以实际翻译过程非常简单,编译器在对程序进行优化后生成高性能的字节码。程序运行时,字节码将被快速翻译成当前 CPU 的指令,在某种程度上相当于将最终机器指令的产生放在动态加载器中进行。在 Sun Microsystems SPARCStation 10 计算机上进行的一项 30 万个方法调用的实验,证明解释型字节码翻译成机器代码的速度和 C/C++ 几乎相当。

10. 多线程

现实世界中,每时每刻都有很多事情在我们身边同时发生。多线程的概念和这种情况差不多,就是让计算机同时运行多个程序段。编写一个能同时处理多个任务的程序要比编写一个单线程程序困难得多。

Java 提供了一套复杂的线程同步化机制,程序员可以方便地使用基于这种机制设计的方法,编写出健壮的多线程程序。

11. 动态性

Java 的动态性表现在它可以设计成能适应环境变化的语言。Java 属于运行加载,Java 的类库(即插即用模块集)可以自由添加方法和属性而不会影响到用户程序。因此,Java 的动态性可以更好地适应不断变化的执行环境。

12. 丰富的类库

Java 代码清晰、结构合理、简明流畅,并提供丰富的类库,使程序员可以很方便地建立自己的系统。

13. 嵌入浏览器运行

Java 的 Applet 小应用程序可嵌入浏览器运行,能够完成动画、游戏、交互式事务处理等功能,使 Web 页面具有动感,而不只是被动地展现文本和图形,提高了 Web 页面的智能性和灵活性。

Applet 提供动态下载程序运行的技术基础,程序在网络上传输而不受 CPU 和环境的限制。Applet 嵌入 Web 页面运行的能力使 Java 能够最大限度地利用网络,为 Java 的发展创造了新机遇。