

普通高等院校计算机类专业规划教材·精品系列

Java程序设计

Java CHENGXU SHEJI

(第二版)

杨厚群 主编

陈 静 王业统 副主编



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内容简介

普通高等院校计算机类专业规划教材·精品系列

Java 程序设计

(第二版)

图灵社区会员专享 (CIP) 目录

Java 程序设计 (第二版) 杨厚群 主编

杨厚群 主编

陈静 王业统 副主编

靳婷 邢治杏 符发 参编

中国版本图书馆 CIP 数据核字 (2015) 第 078312 号

作者：杨厚群 主编
编者：陈静 王业统 副主编
参编者：靳婷 邢治杏 符发

0830-7

责任编辑：周燕燕
封面设计：周燕燕
封面制作：周燕燕
责任校对：周燕燕
责任印制：李仕

出版发行：中国铁道出版社（北京市西城区右安门西街 8 号）邮政编码 100045

网址：http://www.95542.com

2015 年 8 月第 1 次印刷

2015 年 8 月第 2 版

开本：787mm×1092mm 1/16 印张：19 字数：453 千字

书号：ISBN 978-7-113-20823-0

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

地址：北京市西城区右安门西街 8 号

电话：(010) 51873629

内 容 简 介

本书是《Java 程序设计》的第二版, 继续保持了原教材的特点——注重理论传承和实用为先。本书拓展了面向对象程序设计知识, 重新编排了所有例题, 并对部分内容做了调整, 增加了新知识和新例题。本书重点讲解 Java 程序设计知识及其编程方法, 包括 Java 概述、Java 基本编程结构、字符串和数组、对象和类、继承与多态、异常处理、图形编程、Java Swing 与事件处理、Applet 基础、多线程、输入/输出流及文件、Java 的网络编程等。本书有配套的习题与实验指导书。

本书适合作为高等院校计算机类专业的基础教材, 也可作为使用 Java 语言的工程技术人员和科技工作者的自学参考书。

图书在版编目 (CIP) 数据

Java 程序设计/杨厚群主编. 2 版—北京: 中国铁道出版社, 2015. 8

普通高等院校计算机类专业规划教材·精品系列

ISBN 978-7-113-20553-9

I. ①J… II. ①杨… III. ①JAVA 语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2015) 第 152966 号

书 名: Java 程序设计 (第二版)

作 者: 杨厚群 主编

策 划: 周海燕

读者热线: 400-668-0820

责任编辑: 周海燕 徐盼欣

封面设计: 穆 丽

封面制作: 白 雪

责任校对: 汤淑梅

责任印制: 李 佳

出版发行: 中国铁道出版社 (北京市西城区右安门西街 8 号, 邮政编码 100054)

网 址: <http://www.51eds.com>

印 刷: 北京海淀五色花印刷厂

版 次: 2009 年 1 月第 1 版 2015 年 8 月第 2 版 2015 年 8 月第 1 次印刷

开 本: 787mm×1092mm 1/16 印张: 19 字数: 423 千

书 号: ISBN 978-7-113-20553-9

定 价: 42.00 元

版权所有 侵权必究

凡购买铁道版图书, 如有印制质量问题, 请与本社教材图书营销部联系调换。电话: (010) 63550836

打击盗版举报电话: (010) 51873659

南燕，融参袋料，杏奇雅，朝薄，融主隔升融业王，韩潮，融主升籍早融由许本
秀油板，就支代大丁千能巨融即许本板对融即融，融市融融升融承对融对融学大

前言 (第二版)



本书在第一版的基础上，全面讲解了 Java 的基础内容和编程方法，在内容的深度和广度方面都给予了仔细考虑，在类、对象、继承、接口等重要的基础知识上侧重深度，在实用类的讲解上侧重广度。在第二版中增加了泛型等面向对象的知识，对部分内容如多态、算法的讲解以及各章例题内容做了调整，并根据 JDK 版本的演进增加了一些新的知识内容和例题。

通过本书的学习，读者可以掌握 Java 面向对象编程的思想和 Java 在网络编程中的一些重要技术。

全书共分 12 章。第 1 章主要介绍 Java 语言的特点和开发环境，读者可以了解到 Java 是如何做到跨平台的。第 2 章介绍 Java 基本编程结构，包括 Java 程序的基本构成和结构化编程的内容。第 3 章介绍最常用的字符串和数组知识。第 4 章和第 5 章是本书的重点内容之一，讲述了类、对象、继承、接口、多态及泛型等面向对象程序设计的重要知识。第 6 章介绍异常处理，包括了检查型和非检查型异常的知识和异常处理的技巧。第 7 章讲述的是图形编程，介绍了常用的框架和容器，涉及 Color、Font 和 FontMetrics 等类的使用。第 8 章介绍 Java Swing 与事件处理，主要包括布局管理器和组件的使用，事件处理的原理和如何掌握窗口事件、鼠标事件、键盘事件的应用。第 9 章为 Applet 基础，介绍 Applet 的运行原理及其多媒体应用。第 10 章讲述多线程技术，这是较难掌握的一部分内容，在这一章通过许多有启发性的例子来帮助读者理解多线程编程。第 11 章讲解 Java 中的输入/输出流技术。第 12 章讲解 Java 在网络编程中的一些重要技术，涉及 URL、Socket、InetAddress、DatagramPacket、UDP 等重要网络概念。

本书的章节编排与内容以人们学习与认知过程为基础，保持“注重实验”的风格，尽可能将面向对象编程和 Java 自身的特点紧密结合，在内容编排上充分反映 Java 语言这些年来的发展，并兼顾项目管理和环境的要求；注重内容的可读性和可用性，与实际需求相匹配，帮助读者全面理解 Java 程序设计的整体情况与近期发展情况。本书内容力求简明，许多例题都经过精心设计，既能帮助理解知识，又具有启发性；每章都包含了图、表、例程以及类和接口的相关内容，希望读者能在轻松和愉快的氛围之中迅速地理解与掌握 Java 程序设计的知识和方法，并将其应用到实践中去。

本书的例题全部在 JDK 1.7 环境下编译通过。本书还配套了习题与实验指导书，对应主教材的每一章都有练习题，通过完成练习题可以使读者加深对知识的理解。



由于 Java 程序设计覆盖面广, 发展又很迅速, 加之编者水平有限, 书中疏漏与不妥之处在所难免, 诚恳希望读者不吝指正。

目 录

第1章	Java 概述	1
1.1	Java 发展简史	1
1.2	Java 的特点	2
1.3	Java 和 Internet	4
1.4	Java Application 程序	5
1.5	Java Applet 程序	6
1.6	图形界面与字符界面输入/输出	7
1.7	JDK 开发工具	11
1.8	Eclipse 集成开发环境	13
1.8.1	安装	13
1.8.2	界面介绍	13
1.8.3	创建 Java 项目并运行	14
1.8.4	Java 程序调试	15
第2章	Java 基本编程结构	17
2.1	简单的 Java 程序	17
2.2	注释	18
2.3	基本数据类型	19
2.4	变量	20
2.4.1	声明变量	21
2.4.2	变量的使用	22
2.4.3	变量的作用域	23
2.5	常量	24
2.6	操作符	25
2.6.1	赋值运算符	25
2.6.2	算术运算符	25
2.6.3	关系运算符	27
2.6.4	逻辑运算符	28
2.6.5	位运算符	29
2.6.6	其他运算符	30
2.6.7	运算符优先级与结合性	31
2.7	控制语句	32
2.7.1	分支语句	32
2.7.2	循环语句	36
2.7.3	与程序转移有关的跳转语句	39
第3章	字符串和数组	42
3.1	字符串	42
3.1.1	String 类	42



3.1.2	StringBuffer 类	45
3.1.3	StringTokenizer 类	46
3.1.4	Character 类	47
3.2	数组	48
3.2.1	一维数组	48
3.2.2	多维数组	51
3.3	排序	53
3.3.1	选择排序	53
3.3.2	插入排序	54
3.3.3	冒泡排序	54
3.4	查找	56
3.4.1	线性查找	56
3.4.2	二分查找	57
第 4 章	对象和类	58
4.1	面向对象程序设计	58
4.1.1	面向对象方法学的形成	58
4.1.2	面向对象的基本概念	60
4.1.3	UML 静态视图简介	63
4.2	创建用户类	66
4.2.1	类的定义	66
4.2.2	成员变量的定义与初始化	68
4.2.3	成员方法的定义	70
4.2.4	成员方法的重载	73
4.2.5	构造方法的定义与重载	74
4.2.6	将消息传递给方法或构造器	75
4.2.7	嵌套的类	77
4.3	对象实例化	79
4.3.1	创建对象	79
4.3.2	使用对象	80
4.3.3	清除对象	81
4.4	访问属性控制	82
4.4.1	默认访问属性	82
4.4.2	public	83
4.4.3	private	84
4.4.4	protected	87
4.5	静态成员	87
4.5.1	静态成员变量	87
4.5.2	静态成员方法	90
4.6	final、this 和 null	91
4.6.1	final	92
4.6.2	this	93

121	4.6.3 null	93
121	4.7 包	94
121	4.7.1 包的声明	94
121	4.7.2 包的使用	95
121	4.7.3 常用系统包简介	96
121	4.8 综合应用示例	96
102	第 5 章 继承与多态	102
102	5.1 类的继承	102
102	5.1.1 子类的定义	102
104	5.1.2 子类的构造方法	104
106	5.2 类成员的隐藏与重载	106
106	5.2.1 类成员的继承	106
106	5.2.2 成员变量的隐藏	106
108	5.2.3 成员方法的重载与覆盖	108
110	5.2.4 构造方法的覆盖	110
111	5.3 多态性	111
111	5.3.1 多态概念	111
112	5.3.2 多态的应用	112
115	5.4 Object 类和 Class 类	115
115	5.4.1 Object 类	115
116	5.4.2 Class 类	116
117	5.5 抽象类与接口	117
117	5.5.1 抽象类	117
119	5.5.2 接口	119
122	5.6 泛型	122
122	5.6.1 泛型声明	122
123	5.6.2 泛型类	123
124	5.6.3 泛型方法	124
125	5.6.4 通配符泛型	125
126	5.7 对象克隆	126
128	5.8 对象转型和类的设计原则	128
131	5.9 综合应用示例	131
140	第 6 章 异常处理	140
140	6.1 异常和异常类	140
143	6.2 已检查和未检查的异常	143
144	6.3 异常处理	144
144	6.3.1 try...catch...finally 语句	144
145	6.3.2 再次抛出异常	145
146	6.4 异常处理技巧	146
147	6.5 创建自己的异常类	147
148	6.6 综合应用示例	148



第7章 图形编程	151
7.1 Swing 概述	151
7.2 框架	152
7.2.1 创建并显示框架	153
7.2.2 给框架定位	154
7.2.3 在框架中创建组件	154
7.3 在面板中显示信息	155
7.4 颜色	157
7.5 绘制几何图形	158
7.5.1 绘制图形	158
7.5.2 写字	160
7.6 文本和字体	161
7.6.1 Font 类	162
7.6.2 Fontmetrics 类	162
7.7 图像	165
7.7.1 加载图像并显示图像	165
7.7.2 图标	166
7.8 综合应用示例	167
第8章 Java Swing 与事件处理	170
8.1 布局管理介绍	170
8.1.1 顺序布局 (FlowLayout)	171
8.1.2 网格布局 (GridLayout)	172
8.1.3 边框布局 (BorderLayout)	174
8.1.4 箱式布局 (BoxLayout)	175
8.2 文本输入	177
8.2.1 文本框 JTextField	177
8.2.2 JPasswordField	178
8.2.3 文本域 JTextArea	178
8.3 按钮与标签	180
8.3.1 按钮	180
8.3.2 标签	181
8.4 选择组件	182
8.4.1 复选框	182
8.4.2 单选按钮	182
8.4.3 列表	186
8.4.4 下拉列表和组合框	186
8.4.5 选项卡	188
8.4.6 滚动条	189
8.4.7 多个窗口	191
8.5 菜单	191
8.6 复杂的布局管理	194
8.6.1 卡片布局 (CardLayout)	194
8.6.2 网格袋布局 (GridBagLayout)	196

125	8.7	对话框	197
125	8.8	事件处理基础	201
125	8.8.1	事件和事件源	201
125	8.8.2	事件注册监听和处理	201
125	8.8.3	事件处理	202
125	8.9	AWT 事件继承层次	203
125	8.10	AWT 的语义事件	203
125	8.11	低级事件类型	203
125	8.11.1	窗口事件	203
125	8.11.2	鼠标事件	204
125	8.11.3	键盘事件	207
125	8.12	综合应用示例	209
第9章 Applet 基础			214
125	9.1	Applet 运行原理	214
125	9.1.1	运行原理	214
125	9.1.2	关于 repaint()方法和 update(Graphics g)方法	216
125	9.2	Applet 的 HTML 标记和属性	216
125	9.2.1	Applet 定位属性	217
125	9.2.2	Applet 代码属性	218
125	9.2.3	用于非 Java 兼容浏览器的 Applet 属性	219
125	9.2.4	向 Applet 传递消息	219
125	9.3	多媒体应用	220
125	9.3.1	在 Applet 中播放声音	220
125	9.3.2	在 Applet 中绘制图形和图像	221
125	9.3.3	在 Applet 中显示图像	225
125	9.4	JAR 文件	227
125	9.5	综合应用示例	229
第10章 多线程			233
125	10.1	Java 中的线程	233
125	10.2	线程的生命周期	234
125	10.3	线程的优先级和调度管理	236
125	10.4	扩展 Thread 类创建线程	238
125	10.5	实现 Runnable 接口创建线程	239
125	10.6	常用方法	241
125	10.7	线程同步	242
125	10.8	线程组	244
125	10.9	综合应用示例	245
第11章 输入/输出流及文件			248
125	11.1	Java 输入/输出类库	248
125	11.1.1	流的概念	248
125	11.1.2	基本输入/输出流类	249
125	11.1.3	其他输入/输出流类	250

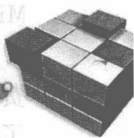


11.1.4	标准输入/输出	251
11.2	字符的输入与输出	253
11.2.1	输入字符	253
11.2.2	输出字符	254
11.3	数据输入/输出流	255
11.4	Java 程序的文件与目录	257
11.4.1	创建 File 类对象	258
11.4.2	获取文件或目录属性	258
11.4.3	文件或目录操作	258
11.4.4	顺序文件的访问	260
11.4.5	随机文件的访问	262
11.5	综合应用示例	266
第 12 章 Java 的网络编程		269
12.1	网络基础知识	269
12.1.1	IP 地址	269
12.1.2	端口	270
12.1.3	客户机与服务器	270
12.1.4	URL 概念	271
12.1.5	TCP/IP 网络参考模型	272
12.2	Java 网络编程概述	272
12.3	Java 网络类和接口	273
12.4	基于 URL 的网络编程	274
12.4.1	URL 类和 URL 对象	274
12.4.2	使用 URL 读取网络资源	276
12.4.3	通过 URLConnection 连接网络	277
12.5	基于 Socket 的网络编程	279
12.5.1	Socket 类	279
12.5.2	ServerSocket 类	279
12.5.3	Socket 通信的过程	280
12.5.4	客户端 Socket	280
12.5.5	服务器 Socket	280
12.5.6	C/S 环境下 Socket 的应用	281
12.6	数据报通信的应用	284
12.6.1	数据报概述	284
12.6.2	发送和接收工作流程	285
12.6.3	利用数据报通信的客户机/服务器程序	285
12.7	综合应用示例	288
12.7.1	HTTP 协议的作用原理	288
12.7.2	Web 服务器功能实现过程	288
12.7.3	Web 服务器实现程序代码	288
12.7.4	运行 Java 服务器	291
参考文献		293



第 1 章

Java 概述



Java 作为一种优秀的语言，具备面向对象、体系结构中立、安全、稳定和多线程等优良特性，是目前软件设计中功能极为强大的编程语言。Java 不仅可以开发大型的应用程序，而且特别适合 Internet 的应用开发。Java 具备了“一次编写，到处运行”的特点，因此，它已成为网络时代最重要的语言之一。也许现在还无法评估 Java 为整个 IT 行业带来的影响，但是有一点却毋庸置疑：Java 将不可避免地影响一代又一代的程序员。

本章要点

- Java 发展简史。
- Java 的特点。
- Java 和 Internet。
- 安装 JDK 开发环境。
- 使用命令行工具。
- 使用集成开发环境。

1.1 Java 发展简史

Java 最初是由 James Gosling 领导的小组在 Sun 公司开发的。该公司以其产品 Sun 工作站而闻名于世。Sun 公司于 1991 年投资启动了一个内部研究项目，代号为 Green。项目的副产品便是诞生了一种类似 C++ 的语言，当时 James Gosling 把它命名为 Oak（橡树），名字起源于其办公室窗外的一棵橡树，后来发现已经有一种计算机语言叫作 Oak。当一天一群 Sun 公司编程人员在咖啡馆里喝着 Java（爪哇）咖啡时，有人灵机一动，举荐了 Java 这个名称，得到了其他人的赞赏，于是 Java 这个名字就传开了，并沿用至今。

面向对象的 Java 具备“一次编写、到处运行”的特点，成为服务提供商和系统集成商用以支持多种操作系统和硬件平台的首选解决方案。Java 作为软件开发的一种革命性的技术，其地位已确定。



1996 年 1 月, Sun 公司发布 Java 1.0, 各大公司(包括 IBM、Apple、HP、Oracle、Microsoft 等)相继从 Sun 公司购买了 Java 技术许可证, 开发相应的产品。

1998 年 12 月, Sun 公司发布 Java 2 平台。该平台不仅可以驾驭智能卡和小型消费类设备, 还可以驾驭大型数据中心服务器等一系列系统。这一开发工具极大地简化了编程人员编制企业级 Web 软件的工作, 把“一次编写、到处运行”的特点应用到服务器领域。

1999 年 6 月, Sun 公司发布 Java 企业平台——J2EE, 成为开发商创建电子商务应用的事实标准。J2EE 平台作为一种可扩展的、全功能的平台, 可以将关键的企业应用扩展到任何 Web 浏览器上, 并可适合各种不同的 Internet 数据流。

2002 年 2 月, Sun 公司推出了 JDK 历史上最成熟的版本 JDK 1.4。自此, Java 在企业平台上大放异彩, 基于 Java 创建的开源框架, 如 Spring、Struts、Hibernate 等涌现出来, 大量企业应用服务器也开始涌现, 如 WebLogic、WebSphere、JBoss 等。

2004 年 10 月, JDK 1.5 推出, 并改名 Java SE 5, 与此同时, J2ME、J2EE 分别改名为 Java ME、Java EE。这个版本里添加了泛型类型, 还添加了几个来源于 C# 的实用语言特性, 如 foreach 循环、自动打包和元数据。

2009 年初推出版本 6, 经过版本 5 的大改, 版本 6 中没有对语言方面再做改进, 而是改进了其他性能, 并增强了一些实用类库。同年 4 月, Oracle 公司宣布收购 Sun 公司, 经过长时间的等待后, 该公司于 2011 年推出了版本 7, 这是本书所使用的版本, 也是目前较新的 Java SE 版本。与前一版相比, Java SE 7 对语言方面做了一些小改进, 使得 Java 语言更加灵活、方便。

目前, Oracle 公司已经推出了 Java SE 8, 在不久的将来还会继续推出 Java SE 9 等。



1.2 Java 的特点

Java 目前非常流行。Java 的迅速发展和被广泛接受归功于它的设计和程序特征。正如 Sun 公司在 Java 白皮书开始所说: Java 是简洁的、面向对象的、分布式的、解释型的、健壮的、安全的、体系结构中立的、可移植的、高效的、多线程的、动态的。

1. Java 是简洁的

大多数计算机语言是复杂的, 但与 C++ 相比, Java 的语法实际上是 C++ 语法的“缩减”版本。Java 摒弃了头文件、指针算法、结构、联合、操作符重载、虚基类等, 并用接口(interface)的简单语言概念取代了 C++ 的多重继承。

Java 还采用自动内存分配和回收, 而 C++ 则要求程序员手动去完成这项工作。语言概念变少了, 再加上清晰的语法, 使得 Java 程序容易编写和阅读。

Java 是简洁的, 它的基本解释器和类支持的大小仅约为 40 KB。增加基本的标准库和线程支持, 大约仅需要增加 175 KB。

2. Java 是面向对象的

简单来说, 面向对象设计是一种把重点放在数据(等于对象)和对象接口的编程

技术,因为世界上的任何事物都可以抽象为对象,如一个人是对象,一个窗口也是对象;Java 以对象为模型描述现实世界,进行对象创建、对象处理,并使对象协调工作。

Java 的面向对象特性和 C++ 相似。两者的主要区别在于对多重继承的处理(Java 使用单继承和接口技术解决),Java 元类模型、反射机制和对象序列化特性使得实现持久对象和 GUI 构建器更为简单和方便。

3. Java 是分布式的

分布式计算涉及多个计算机通过网络协同工作。Java 的设计使分布式计算变得简单起来,这是由于 Java 带有一个扩展例程库,用以处理 HTTP 和 FTP 等 TCP/IP 协议族。Java 应用程序能够通过 URL 打开和访问网络上的对象,其方便程度如同访问本地文件一样。由于 Java 的网络能力强大且易于使用,远程方法调用机制能够进行分布式对象间的通信。

4. Java 是解释型的

运行 Java 程序需要一个解释器。Java 程序编译成 Java 虚拟机(Java Virtual Machine, JVM)字节码,如图 1.1 所示。字节码是独立于计算机的,也就是说,只需要编译一次源代码,编译器生成 Java 字节码,Java 解释器可以在任何移植了 Java 解释器的机器上执行 Java 字节码。

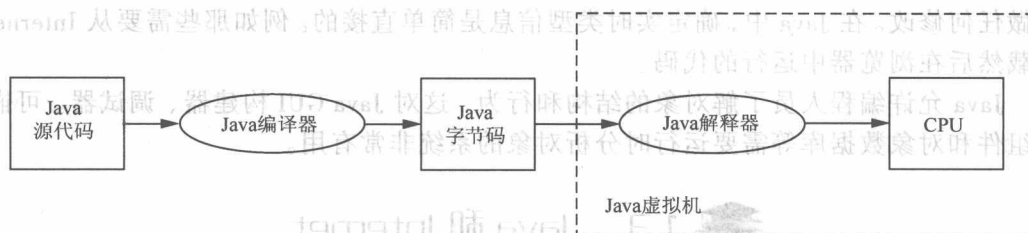


图 1.1 Java 解释器执行 Java 字节码

5. Java 是健壮的

Java 非常重视进行早期问题和后期动态(运行时)的检查,以及消除致错状态。Java 编译器可以查出许多其他语言运行时才能发现的错误。Java 抛弃了其他语言中容易引起错误的某些程序概念类型,如它不支持指针,避免了内存分配错误,以及必须预防内存泄漏。

Java 具有实时异常处理的功能,这有助于提高程序的健壮性。Java 强制程序员编写处理异常的代码,使其能够捕获并响应异常情况,从而使程序在发生运行时错误时能够继续正常执行错误处理代码。

6. Java 是安全的

作为 Internet 程序设计语言,Java 用于网络/分布式环境。Java 执行多层安全机制用于保护系统不受恶意程序破坏。Java 安全机制禁止 Java 程序进行一些操作。

(1) 禁止运行时堆栈溢出,避免如蠕虫所做的破坏。

(2) 禁止在自己的处理空间外破坏内存。

(3) 禁止通过安全控制类装载器来读写本地文件,即当用户下载并运行一个 Java



Applet 时,它不会损害本地系统。

7. Java 是体系结构中立的 传统的编译式语言,源代码必须能被编译成二进制代码或机器代码的可执行形式。而 Java 源代码不会针对一个特定平台进行编译,而是被转换成一种中间格式——字节码,字节码无关体系结构,可在任何运行 Java 虚拟机的计算机上运行,而 Java 虚拟机是与平台相关的。由此可见,Java 程序在 Java 虚拟机上运行,而 Java 虚拟机又在操作系统上运行。Java 虚拟机用来解释和执行 Java 字节码。

使用 Java,软件开发商无须为了适应多个平台为产品开发多个版本,只需编写一个版本就能在各种平台上运行。

8. Java 是多线程的

Java 通过流控制来执行程序流,程序中单个顺序的流控制称为线程,多线程则指的是在单个程序中可以同时运行多个不同的线程,执行不同的任务。多线程意味着一个程序的多行语句可以看上去几乎在同一时间同时运行。Java 将线程支持与语言运行环境结合在一起,提供了多任务并发执行的能力。

9. Java 是动态的

Java 能够适应变化的环境,类库中允许增加新的方法以及实例变量,而客户端无须做任何修改。在 Java 中,确定实时类型信息是简单直接的。例如那些需要从 Internet 下载然后在浏览器中运行的代码。

Java 允许编程人员了解对象的结构和行为,这对 Java GUI 构建器、调试器、可嵌入组件和对对象数据库等需要运行时分析对象的系统非常有用。



1.3 Java 和 Internet

Internet 的发展实现了资源共享,为人们的工作、学习和生活带来了极大的方便;网络应用要求程序代码必须具备安全、可靠的特点。同时,还要求能运行于不同平台和机器,Java 凭借它在语言上无法比拟的优势,如安全性、平台无关性、硬件结构无关性、语言简洁、面向对象等特性成为网络编程的首选语言之一。

Internet 的飞速发展伴随着新技术、新概念和新产品的不断产生,这些新生事物不仅仅是 Internet 发展的产物,也是 Internet 进行再次飞跃的直接推动力。在 Internet 的发展史中,Java 影响非常深刻,是对计算机产业发展促进作用极大的一种新兴技术。

Java 根植于网络,网络的发展促进了 Java 的规范。Internet 的服务种类丰富,应用最广泛的有 WWW (World Wide Web) 服务、Gopher 服务、文件传输服务、远程终端服务、E-mail 服务、网络论坛、各电子公告栏信息服务、网上购物等。Web 页是发布消息、相互交流的重要方式之一。Web 页由网络浏览器装载。由 Java 编写的 Applet 程序代码可以嵌入 Web 页中在浏览器上运行,可以轻松地实现动画、人机交互和事件处理等功能,Java 与 Web 联系十分紧密,Java 在 Web 上的应用充分显现出它的强大功能。

Intranet (内部网) 通常指企业或部门内部,利用 Internet,特别是 Web 技术构造

的供内部工作人员进行信息共享、分布计算的网络环境。Intranet 一般通过“防火墙”(Firewall)与 Internet 相连接,从而使得内部人员可以通过 Intranet 访问 Internet 的资源,而外界对 Intranet 网络资源的访问则受到一定的限制。由于 Intranet 技术是一个企业或部门内部计算机互连网络的基础,利用它可以提高产品设计、制造和营销方面的效率和效益,因此受到了产业界和用户的普遍重视,众多大型软、硬件供应商都提出了自己的 Intranet 构造和解决方案。随着 Java 的广泛应用,Java 技术对 Intranet 的影响日益扩大,使得 Intranet 的构造、应用和管理方式都发生了相应的变化。



1.4 Java Application 程序

Java 程序可分为 Application (应用程序) 和 Applet (小程序) 两种类型,这两类程序的开发原理是相同的,但是运行环境有所不同。Application 程序有以下主要特点:

- (1) Application 程序是独立完整的程序。
- (2) 在命令行调用独立的解释器即可运行 Application 程序。
- (3) Application 程序的主类必须有一个名为 main 的方法,作为程序的入口。
- (4) Application 程序的图形界面需要在程序中自己构建。

例 1.1 是一个简单的 Application 程序,它的功能是在字符界面上输出“This is my first Java program!”。

【例 1.1】 Application 程序示例: MyFirstJavaProgram.java。

```
import java.io.*;
public class MyFristJavaProgram
{
    public static void main(String[] args)
    {
        System.out.println("This is my first Java Program!");
    }
}
```

集成开发环境适合开发含多个源文件的大程序。对于简单程序来说,在编辑完源文件后,可打开一个 Shell 或终端窗口,使用 SDK 自带的 Java 命令实现编译、运行或监视等操作,使得这类程序的开发更加容易快捷。

Java 程序开发包括源代码编写、编译成字节码文件以及运行。由于 J2SDK 本身没有提供编写源代码的编辑工具,所以可使用任何文本编辑器来编写源代码。所有的 Java 程序都是由类或者说类的定义组成。

在编辑 Java 源程序定义类的时候,需要注意如下问题:

- (1) Java 是对大小写敏感的语言,关键字的大小写不能搞错,如果将 class 写成 Class 或者 CLASS,都会导致错误。
- (2) 在一个类内部不能定义其他的类(内部类除外),类与类是平行的,而非嵌套的关系。

- (3) 一个程序内可有一个或多个类,但只能有一个主类。



“谢 (4) 源程序编辑完成后, 应该以主类名命名文件名, 以 java 为其扩展名, 保存在磁盘上。

个一保存好文件后, 执行编译的步骤, 即在命令行下运行 Java 的编译器 javac.exe。下面的语句将编译 MyFirstJavaProgram.java 程序:

```
C:\>javac MyFirstJavaProgram.java
```

编译成功后, 将生成一个或多个字节码文件, 每个字节码文件对应源程序中定义的一个类, 该文件名就是所对应的类的名字, 并以 class 为统一的扩展名。

运行一个编译好的 Java 字节码程序, 需要调用 Java 的解释器工具 java.exe。解释器负责字节码载入、代码校验和解释执行的工作。下面的语句运行已生成的 MyFirstJavaProgram.class 文件:

```
C:\>java MyFirstJavaProgram
```

当程序有多个字节码文件时, 运行时只需要指出主类名即可。



1.5 Java Applet 程序

Java Applet 是另一类非常重要的 Java 程序, 它有如下特点:

(1) Applet 程序不是完整的独立程序, 而更像是一个已经构建好的框架程序中的一个模块。

(2) Applet 程序只能在 WWW 浏览器环境下运行, 因此还必须建立一个 HTML 文件来调用 Applet 程序。

(3) Applet 程序的主类必须是 java.applet.Applet 类的子类。

(4) Applet 程序不用在程序中构建图形界面, 而是直接利用 WWW 浏览器提供的图形用户界面。

【例 1.2】Applet 程序示例, 在 WWW 图形界面上输出 “This is my first Java Applet!”: MyFirstJavaApplet.java。

```
import java.applet.Applet;
import java.awt.*;
public class MyFirstJavaApplet
{
    public void paint(Graphics g)
    {
        g.drawString("This is my first Java Applet!", 10, 10);
    }
}
```

因为 Applet 程序不能单独运行, 必须要有一个 HTML 页面来调用使它在 WWW 浏览器上运行, 可以通过记事本编写以下的 HTML 文件内容, 文件保存为 Applet.html, 与 MyFirstJavaApplet.java 的存放路径一致。

```
<html>
<head><title>SimpleGraphicsInOut</title></head>
```