

21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

Java程序设计 简明教程

Beginning Java for Developer

李永杰 陈鑫伟 编著

- 无需程序设计基础
- 精选典型应用实例
- 掌握Java开发技术



精品系列

 人民邮电出版社
POSTS & TELECOM PRESS

21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

TP312/2901

2008

Java程序设计 简明教程

Beginning Java for Developer

李永杰 陈鑫伟 编著



精品系列

人民邮电出版社

北京

图书在版编目(CIP)数据

Java 程序设计简明教程 / 李永杰, 陈鑫伟编著.

北京: 人民邮电出版社, 2008.7

21 世纪高等学校计算机规划教材. 精品系列

ISBN 978-7-115-17919-7

I. J… II. ①李…②陈… III. Java 语言—程序设计—
高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2008) 第 046914 号

内 容 提 要

本书循序渐进地介绍了 Java 语言的语法基础、开发工具、程序控制语句、面向对象的程序设计、图形用户界面编程、Java 高级编程等内容。每章附有习题, 供读者复习使用, 同时针对每章内容, 在人民邮电出版社网站上提供了上机指导, 供读者选用。

本书通过附加精选实例的方法, 并穿插少量的设计细节和注意事项, 按照实际开发流程对 Java 语言的程序设计进行讲解, 简明易懂, 实用性强。本书可作为普通高等院校计算机及理工类专业 Java 相关课程的教材, 同时也可供 Java 语言的初学者参考使用。

21 世纪高等学校计算机规划教材——精品系列

Java 程序设计简明教程

- ◆ 编 著 李永杰 陈鑫伟
责任编辑 蒋 亮
- ◆ 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>
北京艺辉印刷有限公司公司印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 14.75
字数: 385 千字
印数: 1—3 000 册
- 2008 年 7 月第 1 版
2008 年 7 月北京第 1 次印刷

ISBN 978-7-115-17919-7/TP

定价: 25.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223

反盗版热线: (010)67171154

计算机应用能力已经成为社会各行业最重要的工作要求之一，而计算机教材质量的好坏会直接影响人才素质的培养。目前，计算机教材出版市场百花争艳，品种急剧增多，要从林林总总的教材中挑选一本适合课程设置要求、满足教学实际需要的教材，难度越来越大。

人民邮电出版社作为一家以计算机、通信、电子信息类图书与教材出版为主的科技教育类出版社，在计算机教材领域已经出版了多套计算机系列教材。在各套系列教材中涌现出了一批被广大一线授课教师选用、深受广大师生好评的优秀教材。老师们希望我社能有更多的优秀教材集中地呈现在老师和读者面前，为此我社组织了这套“21世纪高等学校计算机规划教材——精品系列”。

“21世纪高等学校计算机规划教材——精品系列”具有下列特点。

(1) 前期调研充分，适合实际教学需要。本套教材主要面向普通本科院校的学生编写，在内容深度、系统结构、案例选择、编写方法等方面进行了深入细致的调研，目的是在教材编写之前充分了解实际教学的需要。

(2) 编写目标明确，读者对象针对性强。每一本教材在编写之前都明确了该教材的读者对象和适用范围，即明确面向的读者是计算机专业、非计算机理工类专业还是文科类专业的学生，尽量符合目前普通高等教学计算机课程的教学计划、教学大纲以及发展趋势。

(3) 精选作者，保证质量。本套教材的作者，既有来自院校的一线授课老师，也有来自IT企业、科研机构等单位的资深技术人员。通过他们的合作使老师丰富的实际教学经验与技术人员丰富的实践工程经验相融合，为广大师生编写出适合目前教学实际需求、满足学校新时期人才培养模式的高质量教材。

(4) 一纲多本，适应面宽。在本套教材中，我们根据目前教学的实际情况，做到“一纲多本”，即根据院校已学课程和后续课程的不同开设情况，为同一科目提供不同类型的教材。

(5) 突出能力培养，适应人才市场要求。本套教材贴近市场对于计算机人才的能力要求，注重理论技术与实际应用的结合，注重实际操作和实践动手能力的培养，为学生快速适应企业实际需求做好准备。

(6) 配套服务完善，共促提高。对于每一本教材，我们在教材出版的同时，都将提供完备的PPT课件，并根据需要提供书中的源程序代码、习题答案、教学大纲等内容，部分教材还将在作者的配合下，提供疑难解答、教学交流等服务。

在本套教材的策划组织过程中，我们获得了来自清华大学、北京大学、人民大学、浙江大学、吉林大学、武汉大学、哈尔滨工业大学、东南大学、四川大学、上海交通大学、西安交通大学、电子科技大学、西安电子科技大学、北京邮电大学、北京林业大学等院校老师的大力支持和帮助，同时获得了来自信息产业部电信研究院、联想、华为、中兴、同方、爱立信、摩托罗拉等企业 and 科研单位的领导或技术人员的积极配合。在此，人民邮电出版社向他们表示衷心的感谢。

我们相信，“21世纪高等学校计算机规划教材——精品系列”一定能够为我国高等院校计算机课程教学做出应有的贡献。同时，对于工作欠缺和不妥之处，欢迎老师和读者提出宝贵的意见和建议。

前 言

Java 语言具有许多优秀的特性,如简单性、面向对象、分布式、健壮性、可移植性、安全性等,这些优秀的特性得到业界广泛认可,从而使其成为当前最为流行的程序设计语言之一。

与此同时,越来越多的高校将 Java 语言作为程序设计教学时主要的编程工具。一方面,随着互联网应用和 Web 服务的不断发展,计算机科学与技术、网络工程、软件工程等传统的计算机类专业加强了对于 Java 语言开发能力的要求,这些专业或将 Java 课程与 C++课程平行开设供学生选择,或在 C++课程之后增加一门精炼的 Java 课程;另一方面,随着移动应用和嵌入式应用的不断推广,通信、电子信息、自动化、微电子等理工类专业也纷纷开设了 Java 课程,以适应用人单位对于专业人才的能力要求。

本书作为一本 Java 语言的简明教程,主要是向读者介绍 Java 语言的基础知识和实用技术,目的是使读者能够快速了解和掌握 Java 语言的相关内容。读者在学习本书之前,不需要具备程序设计的学习经验。如果读者在学习本书之前有过程式程序设计语言(例如 C 语言)的学习经验,则可以从 Java 语言的面向对象概念和特性部分开始重点学习。

本书介绍内容时注重将基础理论知识的讲解与实践应用相结合。全书共分 9 章。其中,第 1 章和第 2 章对 Java 的基础知识,包括 Java 语言的发展历史、面向对象程序设计的基本思想、Java 开发工具、Java 语言的基本语法等进行介绍。第 3 章主要介绍了 Java 语言程序控制语句,包括选择语句、循环语句、跳转语句、断言等。第 4 章和第 5 章为本书学习的难点,详细讲解了 Java 语言面向对象编程的特性。第 6 章介绍了 Java 语言的类库,从 Object 类开始,介绍了封装类、数组、String 类以及集合框架等。第 7 章介绍了如何使用 Java Swing 进行图形界面程序编程,对其中常用组件、事件处理、布局管理等都作了详细介绍,并通过丰富实例说明其使用方法。第 8 章主要介绍了 Java Applet 的基础知识与应用,并通过综合实例进一步说明 Java Applet 的使用。第 9 章为 Java 高级编程,主要是为读者进一步学习 Java 语言做好准备,内容包括 Java 语言中的异常处理、多线程、I/O 流与文件处理、网络编程等。

本书可作为普通高等院校计算机及理工类专业 Java 相关课程的教材,同时也可供 Java 语言的初学者参考使用。

针对书中各章内容,编者设计了相应的上机指导,在人民邮电出版社网站供老师和学生免费下载实践练习使用。本书中所有例题和代码都已调试通过,并且针对本书制作了多媒体课件,供老师教学时参考使用。

本书由李永杰、陈鑫伟共同编写。参与本书修改、整理和示例编写调试的还有陈晨、李洋、刘振清、宋昕、姜海英、许泽、杨晓强等,在此,编者对以上人员致以诚挚的谢意!

由于时间仓促和作者的水平有限,书中错误和不妥之处在所难免,敬请读者批评指正。

编 者

2008 年 6 月

目 录

第 1 章 Java 语言概述.....1

1.1 Java 语言简介	1
1.1.1 Java 发展的历程	1
1.1.2 Java 的不同平台	2
1.2 Java 语言的特点	2
1.3 Java 开发工具箱	3
1.3.1 系统要求	3
1.3.2 下载 JDK	4
1.3.3 安装 JDK	4
1.3.4 JDK 目录	6
1.3.5 设置环境变量	6
1.4 一个简单的 Java 程序	8
1.4.1 简单 Java 程序说明	8
1.4.2 编译和运行 Java 源程序	9
1.4.3 程序解析	9
1.5 Java 开发工具概述	9
1.5.1 文本编辑器 UltraEdit 和 EditPlus	10
1.5.2 Borland 公司的 JBuilder	11
1.5.3 Eclipse 详细介绍	11
1.6 简单的 Java 小应用程序	15
小结	16
习题	16
上机指导	17

第 2 章 Java 语法基础.....18

2.1 Java 程序的构成	18
2.2 Java 基本数据类型	19
2.2.1 整数类型	19
2.2.2 浮点类型	21
2.2.3 字符类型 (char)	21
2.2.4 布尔类型 (boolean)	22
2.2.5 类型之间的转换	22
2.3 标记	24
2.3.1 空白分隔符	24

2.3.2 注释 (comments)	24
2.3.3 关键字	27
2.3.4 标识符 (identifier)	28
2.3.5 分隔符	28
2.4 常量和变量	28
2.4.1 常量 (literal)	28
2.4.2 变量 (variable)	29
2.5 表达式	30
2.5.1 赋值	30
2.5.2 算术运算符	31
2.5.3 自增自减运算	36
2.5.4 关系运算符	37
2.5.5 逻辑运算符	39
2.5.6 三元运算符	41
2.5.7 位运算	42
2.5.8 移位运算	43
2.5.9 括号及运算符间的优先级关系	46
2.6 常用数学工具包——java.lang.Math 类	46
2.6.1 数学常量	47
2.6.2 常用数学函数	47
小结	49
习题	49
上机指导	49

第 3 章 Java 语言程序控制语句.....50

3.1 选择语句	50
3.1.1 if 语句	50
3.1.2 switch 语句	52
3.2 循环语句	58
3.2.1 while 语句	58
3.2.2 do-while 语句	59
3.2.3 for 语句	61
3.3 跳转语句	63
3.3.1 break 语句	64
3.3.2 continue 语句	65

3.3.3 return 语句	67	5.1.1 什么是内部类	92
3.3.4 异常处理语句	67	5.1.2 使用内部类	92
3.4 断言	67	5.1.3 什么是匿名类	94
3.4.1 断言的定义	67	5.1.4 使用内部类需要注意的问题	94
3.4.2 断言的开启/关闭	68	5.2 抽象类	95
3.4.3 防止断言滥用	70	5.2.1 抽象类和抽象方法	95
3.5 Java 简单程序示例	71	5.2.2 如何定义和使用抽象类	95
小结	72	5.3 继承	97
习题	73	5.3.1 子类	97
上机指导	73	5.3.2 覆盖父类中的方法	99
第 4 章 Java 面向对象编程基础	74	5.3.3 this 关键字的使用	101
4.1 Java 的类	74	5.3.4 super 关键字的使用	102
4.1.1 类的基本结构	74	5.3.5 继承的层次结构	102
4.1.2 类的定义	75	5.4 包	103
4.1.3 Java 源文件结构	75	5.5 接口	103
4.1.4 main() 入口方法	75	5.5.1 接口概述	103
4.2 类的成员变量与方法	76	5.5.2 成员变量在接口中的使用	104
4.2.1 访问修饰符: public, private, protect 和 default	77	5.5.3 方法在接口中的使用	106
4.2.2 构造方法	77	5.5.4 接口与抽象类	113
4.2.3 定义变量	78	5.5.5 instanceof 的使用	116
4.2.4 定义成员方法	78	小结	120
4.2.5 参数的传递	80	习题	120
4.2.6 方法的返回值	81	上机指导	121
4.2.7 成员方法重载与过载	81	第 6 章 Java 的类库	122
4.2.8 static 静态成员	82	6.1 语言基础类库	122
4.2.9 final 关键字	83	6.1.1 Object 类	122
4.3 对象概述	84	6.1.2 封装类——数据类型类	126
4.3.1 对象的声明及创建	84	6.1.3 利用封装类进行数据转换功能	129
4.3.2 调用对象的方法	86	6.2 数组	132
4.3.3 对象的生存周期	87	6.2.1 数组的创建	133
4.3.4 对象的清理	87	6.2.2 用数组保存对象	134
4.4 简单范例	88	6.2.3 数组的排序	137
小结	90	6.2.4 二维数组	138
习题	90	6.3 字符串	139
上机指导	91	6.3.1 String 类	140
第 5 章 Java 面向对象编程进阶	92	6.3.2 StringBuffer 类	143
5.1 内部类和匿名类	92	6.3.3 正则表达式	146
		6.4 Java 中集合框架	150
		6.4.1 集合类的结构层次	150

6.4.2 常用集合类的使用-List 的使用	150	7.7.2 鼠标事件	184
6.4.3 常用集合类的使用-Vector 的 使用	152	7.7.3 鼠标移动事件	184
6.4.4 常用集合类的使用: Map 概述	153	7.7.4 对话框事件	185
6.4.5 常用集合类的使用: Hashtable 的 使用	156	7.7.5 键盘事件	185
小结	158	7.8 布局管理器	185
习题	158	7.8.1 概述	185
上机指导	159	7.8.2 边框布局 BorderLayout	186
第 7 章 Java Swing	160	7.8.3 流动布局 FlowLayout	186
7.1 Java Swing 概述	160	7.8.4 卡片布局 CardLayout	186
7.2 一个简单的 Swing 程序	161	7.8.5 网格布局 GridLayout	187
7.3 Swing 和 AWT 的对比	162	7.9 图形用户界面编程综合案例	187
7.3.1 Swing 与 AWT 之间的关系	163	小结	190
7.3.2 Swing 与 AWT 的控件的 混用建议	163	习题	190
7.4 界面图形的绘制以及文字显示与 颜色控制	164	上机指导	190
7.4.1 图形绘制	165	第 8 章 Java Applet 编程	191
7.4.2 文本显示	168	8.1 Applet 基础知识	191
7.4.3 动画效果	169	8.1.1 Applet 的特点	191
7.5 对话框与面板	171	8.1.2 Applet 的安全机制	191
7.5.1 顶级容器概述	171	8.1.3 Applet 的基本工作原理	192
7.5.2 JFrame 主对话框	171	8.1.4 如何在网页中嵌入 Applet	192
7.5.3 JDialog 对话框	172	8.2 Applet 的生命周期和四种基本方法	194
7.5.4 JPanel 面板的使用	173	8.3 向 Applet 传递参数	195
7.6 常用的组件	173	8.4 Applet 综合示例	197
7.6.1 按钮	173	8.4.1 示例—使用 Applet 显示当前 日期及时间	197
7.6.2 文本组件	174	8.4.2 示例—使用 Applet 实现计算器 功能	198
7.6.3 单选按钮 JRadioButton	176	8.4.3 示例—在 Applet 小应用程序中 显示图像	202
7.6.4 复选框 JCheckBox	177	小结	203
7.6.5 下拉列表 JComboBox	178	习题	203
7.6.6 滑块 JSlider	178	上机指导	204
7.6.7 进度条 JProgressBar 及 Timer 组件	179	第 9 章 Java 高级编程	205
7.6.8 菜单 JMenu	180	9.1 异常处理	205
7.6.9 工具栏 JToolBar	181	9.1.1 异常类的层次结构	205
7.7 Java 中的事件处理	181	9.1.2 如何捕获异常	206
7.7.1 事件处理概述	182	9.1.3 如何抛出异常	207
		9.1.4 自定义异常	207

9.1.5 使用 finally 语句段	208
9.2 Java 多线程机制	208
9.2.1 Java 中的线程	208
9.2.2 创建线程的两种方式	210
9.2.3 线程的启动与停止	210
9.2.4 线程的优先级	211
9.2.5 线程的同步	211
9.3 I/O 数据流与文件处理	211
9.3.1 流概述	211
9.3.2 文件类(File)的使用	213
9.3.3 随机访问文件	215

9.4 网络编程	216
9.4.1 网络通讯概述	216
9.4.2 使用 URL 类	217
9.4.3 使用套接字 Socket 类	219
9.5 数据库操作	220
9.5.1 使用 JDBC 连接数据库	220
9.5.2 执行 SQL 语句	226
小结	227
习题	228
上机指导	228

第 1 章

Java 语言概述

Java 语言是由 Sun 公司于 1995 年推出的一种新的编程语言，是一种跨平台、适合于分布式计算环境的纯面向对象语言。

1.1 Java 语言简介

Java 的历史要追溯到 1991 年，Sun MicroSystem 公司的 Jame Gosling、Bill Joe 等人，为在家用消费类电子产品上进行交互式操作而开发了一个名为 Oak 的软件，但当时并没有引起人们的关注。后来发现 Oak 是一种已有计算机语言的名字，于是将其更名为 Java。



Java 是爪哇岛出产的美味咖啡，用其命名表示给每个使用 Java 进行开发的程序员一杯美味的咖啡。

1.1.1 Java 发展的历程

Java 从诞生到现在主要经历了如下过程。

(1) 1996 年初，Sun 公司发布了 Java 1.0，但是很快人们就发现其缺陷很严重。虽然后来 Java 1.1 的发布弥补了其中大部分明显的缺陷，为 GUI 编程增加了新的事件处理模型，但仍然具有很大的局限性。

(2) 1998 年 Sun 公司推出 Java 1.2。该版本更加接近了“一次开发，随处运行”的承诺。后来的 1.3 及 1.4 版本扩展了标准类库，提高了系统性能，同时为 Java 语言添加了很多新的特性。在这期间随着网络的发展，Java 也在各个领域得到了广泛的应用。

从 Java 1.2 开始 Java 也称为 Java 2，并且分 J2SE、J2EE、J2ME 3 个版本。

(3) 2004 年 Java 5.0 发布，为 Java 添加了很多新的特性，如泛型、安全枚举类型、程序注解等，大大简化了 Java 的开发。



5.0 版原来被命名为 1.5 版，在 2004 年的 JavaOne 会议之后，版本数字升为 5.0。

(4) 2006 年底 Java 6.0 发布，不但进一步提升了性能，还加入了对各种脚本语言嵌入式开发的支持等特性，大大提升了 Java 的开发能力和易用性。

本书将基于 Java SE 6.0 向读者进行介绍。

1.1.2 Java 的不同平台

根据不同的开发用途，将 Java 分为如下 3 个不同的平台版本。

(1) Java SE (Java Standard Edition): Java 的标准版，主要用来进行桌面以及普通网络、数据库等程序的开发。

(2) Java EE (Java Enterprise Edition): Java 的企业版，提供了企业级开发需要的各种技术支持，如 RMI, JNDI, EJB 等。主要用来进行企业级应用的开发。

(3) Java ME (Java Micro Edition): Java 的微型版，提供了移动、嵌入式开发需要的各种技术支持，如 CDC, CLDC 等。主要用来进行移动、嵌入式应用的开发。

在 Java SE 5.0 以前这 3 个平台分别称为 J2SE(Java 2 Standard Edition), J2EE(Java 2 Enterprise Edition), J2ME (Java 2 Micro Edition)。

J2SE 是 Java2 平台的标准版，J2SE 包含构成 Java 语言核心的类。如：数据库连接、接口定义、输入/输出、网络编程。J2SE 适用于桌面系统，提供了 CORBA 标准的 ORB 技术，结合 Java 的 RMI 支持分布式互操作环境，运行在 Java 虚拟机上。J2SE 是可移植性、异构性的实现环境和健壮平台，也是实现可伸缩性、分布式异构互操作应用软件开发的标准平台，现在已经更名为 Java SE。

J2EE 平台主要用于 Java 技术企业级应用，可以使程序员迅速、快捷地开发和分布式企业级应用程序。由于 J2EE 平台作为一种可扩展的、全功能的平台，可以将企业应用扩展到任何 Web 浏览器上。目前，很多厂家都对 J2EE 给予支持，并将 J2EE 技术作为大型 B2B 市场和海量交易处理的安全稳定的端到端平台。

J2EE 技术的基础就是 J2SE 标准版，巩固了标准版中的许多优点。其最终目的是能够使企业开发者大幅缩短投放市场时间。J2EE 为灵活配置各种多层企业应用软件，特别是 B2B、B2C 等电子商务应用，提供了强大的服务功能。

J2ME 被称为 Java2 微型版，被使用在许多消费电子产品上，如智能卡、手机、PDA、电视机顶盒等方面。同样，J2ME 也秉持了 Java 语言一贯的特性，那就是跨平台和安全网络传输。J2ME 使用了一系列更小的类包，而且 Javax.microedition.io 为 J2SE 包的子集。

1.2 Java 语言的特点

从 Java 诞生以来，就以其在网络应用开发上独特的魅力以及“一次开发，随处运行”的可移植性引起了人们极大的兴趣。自 1994 年以来，Internet 与万维网迅速发展，这大大促进了以网络开发见长的 Java 的发展，使其逐渐成为 Internet 上最受欢迎的编程平台。下面从几个不同的方面介绍 Java 的一些特点，并且通过与 C++ 的对比进一步指出其在开发方面的优势。

1. 简单性

与 C++ 语法相比，Java 语法是一个“升华”版，其剔除了 C++ 中许多易混淆、难理解的特性，但在设计 Java 的时候还是尽可能地接近 C++，以便系统更易于理解。在 Java 中，略去了头文件、指针运算、操作符重载、多重继承等概念，并且通过自动垃圾收集大大简化了程序员的内存管理工作。

另外，Java 也适于在嵌入式系统上运行，其基本解释器以及类库的支持仅为 40KB 左右，加上多线程的支持也只有 175KB 左右，这是一个杰出的成就。

2. 面向对象

在相当长时间里,面向对象已经证明了其自身的价值,现代程序设计语言不使用面向对象技术让人难以置信。Java 的面向对象特性与 C++ 旗鼓相当,实现了模块化和信息隐藏,类提供了对象的原型,子类通过继承机制可以使用父类所提供的数据与方法。Java 语言的设计集中于对象及其接口上。Java 与 C++ 不同的是取消了多重继承,取而代之的是接口概念,使 Java 更加易用。

3. 分布式

Java 在网络方面的强大易用是其他语言无法比拟的,可以说 Java 是面向网络的语言。通过其提供的类库可以方便地处理各种网络协议,方便地进行传统的套接字网络开发,例如 RMI、CORBA、Web 服务等现在流行的网络开发。

4. 健壮性

Java 在编译和运行时,都会对程序可能出现的问题进行检查。并将出错信息报告给程序员。同时其提供垃圾收集机制来自动管理内存,避免了程序员很多无心的错误和恶意的攻击。而 C++ 需要程序员自己编写代码管理内存,这对大多数程序员来说并不是一项愉快的工作。对曾经花费很多时间来检查悬浮指针引起内存冲突的程序员,一定非常喜欢 Java 的这一特性。

5. 可移植性

与 C++ 不同,Java 规范中没有“依赖具体实现”的地方。与平台无关的特性可以方便地将 Java 部署到任何不同平台的机器上。同时 Java 的类库封装了不同平台上的实现,为其提供统一的接口,这使得同样的类库可以在不同的平台上使用。这也就意味着用 Java 开发的应用可以“一次开发,随处运行”。

6. 安全性

在安全性方面 Java 也是首屈一指的,其是目前极少的语言本身具有安全性的开发平台。而 JCP 也大力支持对 Java 安全性的研究,制定了使用规范,实现了虚拟机安全机制和安全库,添加了数字签名类的概念。从一开始 Java 就被设计为有防范各种病毒、袭击的能力,举例如下:

- (1) 一切对内存的访问都必须通过对象的实例引用来实现;
- (2) 禁止破坏自己处理空间之外的内存;
- (3) 禁止运行时堆栈溢出;
- (4) 未经授权禁止读写文件。



JCP 是 Java Community Process 的缩写,是制定 Java 标准的一个委员会,由很多大的厂商和机构参加,如 Sun、IBM、BEA、Oracle 等都是 JCP 的成员。

1.3 Java 开发工具箱

JDK 是 Java Development Kit 的缩写,即 Java 开发工具箱,Java 开发工具包是免费下载和使用的。在这一节中,将向读者介绍如何获得开发 Java 程序所需要的环境,以及环境的安装和使用前的设置。在一切准备好之后,1.5 小节将用最简单的 HelloWold.java 作为第一个 Java 程序,向大家介绍 Java 程序最基本的编写、编译和运行的方式,使大家对 Java 程序有一个初步的认识。

1.3.1 系统要求

浏览器方面,Java 支持 Solaris 8 (SPARC 和 x86) 上的 Netscape, Windows 平台上的 Internet

Explorer。操作系统方面,Java 支持 Intel Architecture (32 位)的系统,可以用于 Windows 95、98、NT4.0 (Pack 5)、ME、2000、XP、2003 Server 等的操作系统,某些新操作系统可能仍然用的是 Beta 版本。

1.3.2 下载 JDK

Java 开发工具包是免费下载和使用的,需要的读者可以到相关的网站上进行搜索,也可以根据笔者提供的 URL 地址下载相应的 JDK。JDK 当前的最新版本为 jdk 1.6.0_01-ea,可以根据不同的操作系统平台,下载相应的 JDK。本书将以基于 Windows 平台的 32 位机为例,介绍下载的具体过程,步骤如下。

(1) 在浏览器地址栏输入“<http://java.sun.com/javase/downloads/index.jsp>”,过一会可以进入如图 1-1 所示的下载网页。

(2) 将滚动条向下拉,寻找写有“JDK6”的一项,单击后边的“Download”按钮,进入如图 1-2 所示的 Java (TM) SE Development Kit 6 的下载页面。

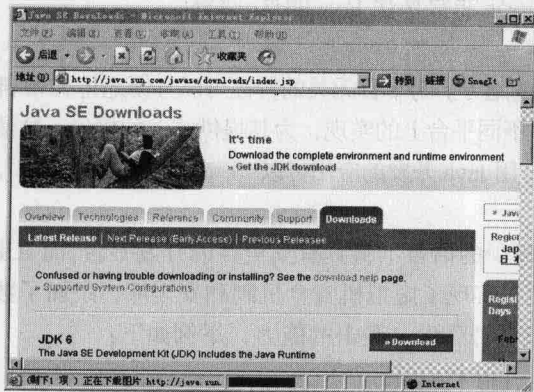


图 1-1 Java SE 官方下载网页

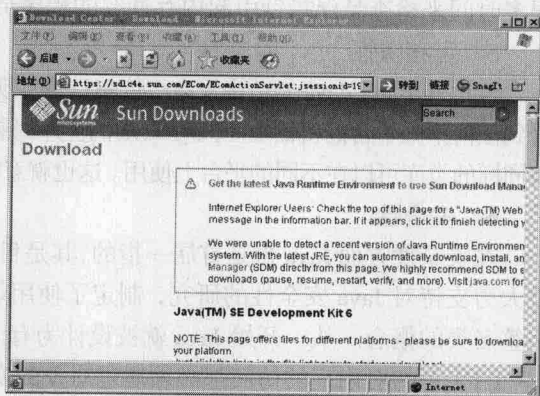


图 1-2 Java (TM) SE Development Kit 6 下载页面

(3) 将滚动条向下拉,选中“Accept License Agreement”单选按钮,页面会自动刷新,开始下载 JDK。

(4) 在页面中寻找如图 1-3 所示的写有“Windows Platform - Java (TM) SE Development Kit 6”的表格,在此表中找到第 2 列写有“jdk-6-windows-i586.exe”的一行,并单击此行“Windows Offline Installation, Multi-language”的超链接进行下载。

Windows Platform - Java(TM) SE Development Kit 6		
↓ Windows Offline Installation, Multi-language	jdk-6-windows-i586.exe	53.16 MB
↓ Windows Online Installation, Multi-language	jdk-6-windows-i586-iftw.exe	361.62 KB

图 1-3 Windows Platform - Java (TM) SE Development Kit 6 下载选项



基于其他操作系统平台的 JDK 下载方法与上边介绍的类似,读者可参照上边的介绍自行操作。同时本书光盘中也附有常用操作系统平台下的 JDK 安装程序。

1.3.3 安装 JDK

下载完成之后,即可进行安装。下面介绍在 Windows 操作系统下安装 JDK 的方法。

(1) 进入存放 Java SE 6.0 软件包安装程序的目录, 双击 jdk-6-windows-i586.exe, 运行 Java SE6.0 的安装程序。

(2) 在经过短暂的初始化工作后, 进入安装界面, 如图 1-4 所示。

(3) 单击【接受】按钮, 接受许可证协议, 继续下面的安装, 此时安装界面如图 1-5 所示。

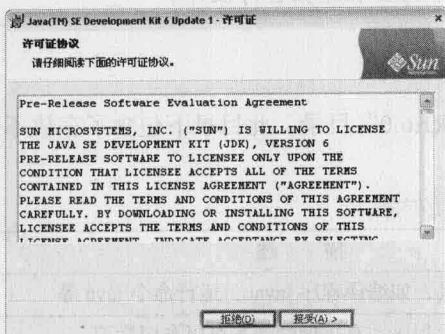


图 1-4 接受许可协议

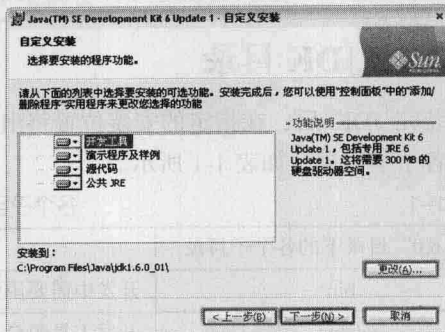


图 1-5 选择安装功能与目录

(4) 在此处选择安装的功能组件, 单击【更改】按钮选择安装目录。完成设置后, 单击【下一步】按钮继续安装, 安装界面如图 1-6 所示。

(5) 在安装完功能组件后, 会弹出图 1-7 所示的对话框, 在此处配置 Java 运行时环境 (JRE) 安装。

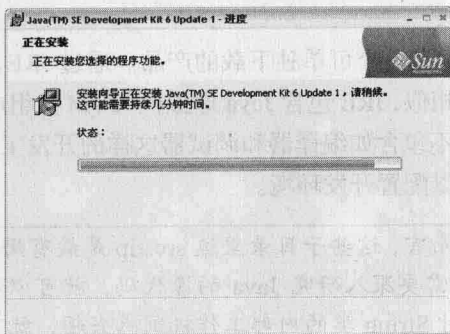


图 1-6 【安装进度】对话框

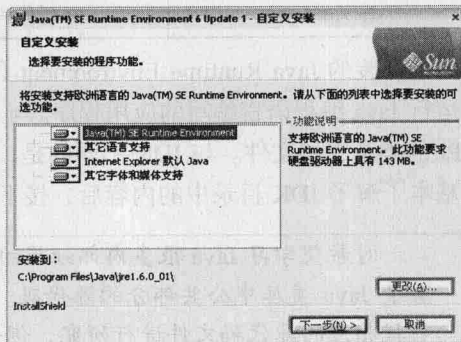


图 1-7 安装 JRE

(6) 选择安装语言支持等功能, 并单击【更改】按钮选择安装目录。完成设置后, 单击【下一步】按钮, 程序进入图 1-8 所示的界面。

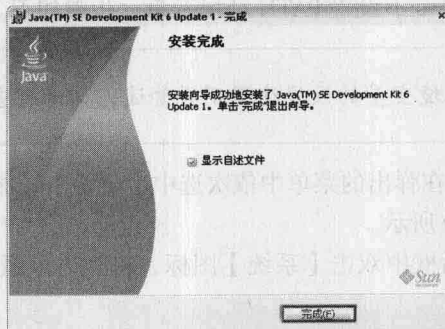


图 1-8 完成安装

(7) 在图 1-8 所示的对话框中选择是否显示自述文件, 完成选择后, 单击【完成】按钮退出。这样就完成了 JDK 的安装。



在其他操作系统 (Solaris、Linux) 中安装 Java SE 6.0 的过程与在 Windows 下的安装过程基本相同, 读者可以参照 Windows 下的安装过程自行操作。

1.3.4 JDK 目录

安装完 JDK 后, 在指定的安装位置将出现 “jdk1.6.0” 目录, 此目录下包含了存放不同方面内容的各个子目录, 如表 1-1 所示。

表 1-1

各个子目录的存放内容

“jdk1.6.0” 目录下的各个子目录	描 述
bin	开发中需要用到的工具, 如编译程序 javac, 运行命令 java 等
demo	开发工具包自带的演示程序, 有兴趣的读者可以参照学习
db	使用嵌入式数据库 Derby 开发所需要的资源以及一些案例
include	用于编译本地方法的 C++ 头文件
jre	Java 运行时环境
lib	开发工具包的类库文件
sample	开发工具包自带的示例程序, 有兴趣的读者可以参照学习
src.zip	类库 API 源代码文件

而可选安装的 Java Runtime Environment (JRE) 是一个可单独下载的产品。通过 JRE, 程序员可以运行 Java 编程语言编写的应用程序。与 JDK 相似, JRE 包含 Java 虚拟机 (JVM)、组成 Java 平台 API 的类及支持文件。与 JDK 不同的是, JRE 不包含如编译器和调试器这样的开发工具。

在基本了解了 JDK 目录中的内容后, 接下来可以配置开发环境。



对希望学习 Java 很多内部细节的读者而言, 这些子目录里边 src.zip 是最有用的, 包含了 Java 类库中公共部分的源代码。如果需要深入研究 Java 的源代码, 就可以在其中寻找相应的源代码文件进行研究, 例如, 对 String 类的内部工作机制感兴趣, 就可以查看其中的 src/java/lang/String.java 文件。

1.3.5 设置环境变量

下面将介绍如何在 Windows 下设置 PATH 环境变量, 步骤如下。



其他操作系统环境变量的设置这里不再赘述, 读者可查阅其他相关资料自行设置。

(1) 单击【开始】按钮, 在弹出的菜单中依次选中【设置】|【控制面板】命令。系统将弹出【控制面板】对话框, 如图 1-9 所示。

(2) 在【控制面板】对话框中双击【系统】图标, 将弹出【系统属性】对话框, 如图 1-10 所示。

(3) 在弹出的【系统属性】对话框中, 单击【高级】标签, 如图 1-11 所示。



图 1-9 【控制面板】对话框

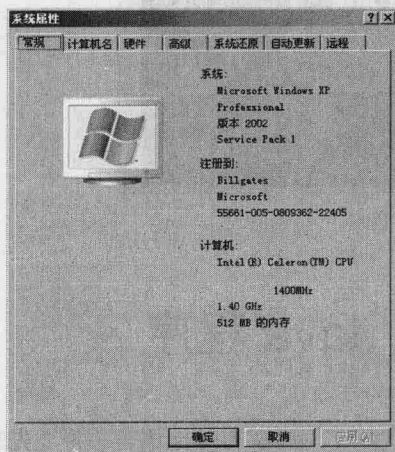


图 1-10 系统属性对话框

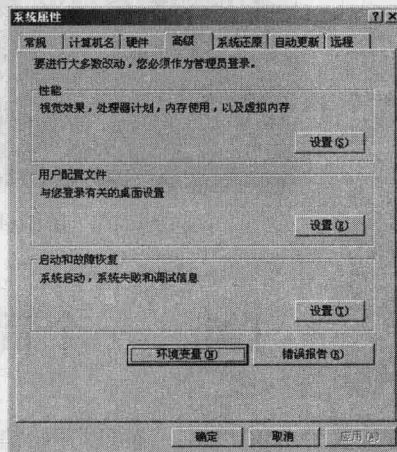


图 1-11 高级标签

(4) 在高级标签对话框中单击【环境变量】按钮，进入【环境变量】设置对话框，如图 1-12 所示。

(5) 在【环境变量】对话框中的【系统变量】列表框中找到【Path】选项并选中，然后单击【编辑】按钮进入【编辑系统变量】对话框，如图 1-13 所示。

(6) 在【变量值】文本框内容的最前边插入 JDK 目录下 bin 目录所在的路径，如“C:\Program Files\Java\jdk1.6.0_01\bin”，并用分号与后边原来的内容相隔。



读者可以根据自己机器 JDK 的安装情况进行设置。

(7) 完成编辑后，逐步单击【确定】按钮退出。

设置完成后，测试环境变量是否设置成功；打开 Windows 的【命令提示符】对话框，输入 java-version，然后按下 Enter 键，如果是出现如图 1-14 所示的提示信息，显示了读者目前安装的版本是 1.6.0_01 等详细信息；键入 java、javac 等命令同样都出现提示信息；则表明读者安装 JDK 成功。读者还可以使用一个简单的 Java 实例进一步测试 JDK。

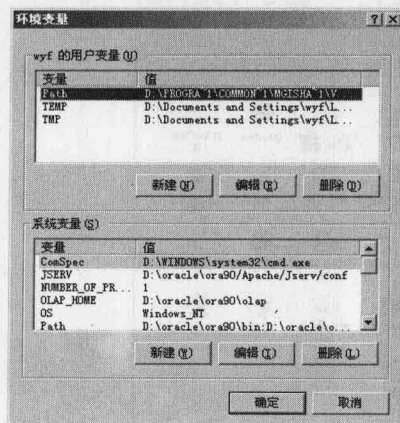


图 1-12 【环境变量】设置对话框

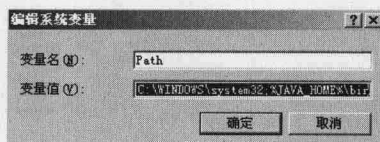


图 1-13 【编辑系统变量】对话框

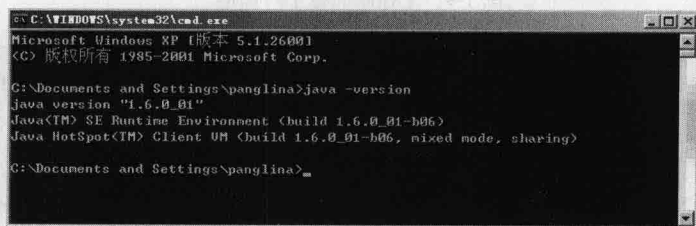


图 1-14 JDK 安装版本

1.4 一个简单的 Java 程序

安装并配置了 JDK 以后，就可以开始编写和运行代码了。和学习其他语言一样，先来考虑一个最简单的应用程序：HelloWorld.java，从而对 Java 代码有一个初步的印象。

1.4.1 简单 Java 程序说明

如代码 1-1 所示，完成了在屏幕上输出 “HelloWorld!” 一行文本。

【代码 1-1】 HelloWorld 代码

```
class HelloWorld
{
    public static void main(String argv[])           //类中主方法，程序的入口点
    {
        System.out.println("Hello World!");          //在屏幕输出字符串 “Hello World!”
    }
}
```

代码编写步骤如下。

(1) 可以在桌面上创建一个文本文档 “新建文本文档.txt”，在其中输入代码内容，如图 1-15 所示。

(2) 在 “新建文本文档.txt” 中编写好代码后，单击【文件】|【另存为】命令，出现文件存取对话框，并选择路径为 “C:\” 下，并将文件名改为 “HelloWorld.java”，保存类型设置为 “所有文件”，如图 1-16 所示。