

计算机专业“十三五”规划教材

# Java程序设计

主编◎袁明兰 王晓鹏 孔春丽



北京希望电子出版社  
Beijing Hope Electronic Press  
[www.bhp.com.cn](http://www.bhp.com.cn)

计算机专业“十三五”规划教材

# Java 程序设计

主 编 袁明兰 王晓鹏 孔春丽

副主编 梁兴波 胡雪松 戴 琴 陈 明 刘德政



北京希望电子出版社  
Beijing Hope Electronic Press  
www.bhp.com.cn

## 内 容 简 介

本书以 Java 语言的基础程序设计、面向对象程序设计和事件处理为三大主线,采用浅显易懂的语言和丰富简单的实例,全面系统地介绍了 Java 语言的特点及应用技术,以及 Java 语言面向对象程序设计的重点和难点。本书共 10 章,主要内容包括 Java 语言概述、Java 语言运行环境及常用工具、Java 编程、JSP 编程、数据库访问、Servlet 编程、JavaBean 编程、自定义标签、MVC 模式和 Struts 应用。

本书既可作为应用型本科院校、职业院校计算机专业的教材,也可作为各学校程序设计公共选修课的教材,同时还可用作职业教育的培训用书和 Java 语言初学者的入门教材,或为具有一定 Java 语言编程经验的开发人员学习使用。

### 图书在版编目 (CIP) 数据

Java 程序设计 / 袁明兰, 王晓鹏, 孔春丽主编. --  
北京: 北京希望电子出版社, 2018.11

ISBN 978-7-83002-651-6

I. ①J… II. ①袁… ②王… ③孔… III. ①JAVA 语  
言—程序设计 IV. ①TP312.8

中国版本图书馆 CIP 数据核字 (2018) 第 255694 号

出版: 北京希望电子出版社  
地址: 北京市海淀区中关村大街 22 号  
        中科大厦 A 座 10 层  
邮编: 100190  
网址: www.bhp.com.cn  
电话: 010-82626270  
传真: 010-62543892  
经销: 各地新华书店

封面: 赵俊红  
编辑: 李小楠  
校对: 薛海霞  
开本: 787mm×1092mm 1/16  
印张: 16.5  
字数: 428 千字  
印刷: 廊坊市广阳区九洲印刷厂  
版次: 2018 年 12 月 1 版 1 次印刷

定价: 48.00 元

# 前言

随着信息化的发展，网络技术水平不断提高，带动了 Java 语言在金融、通信、制造、电子政务、移动设备及消费类电子产品等领域日益广泛的应用，Java 语言已经和我们的日常生活息息相关。近年来，市场对 Java 语言开发人才的需求旺盛，激发了广大开发人员学习 Java 语言的兴趣。对于很多 Java 语言的初学者来说，如何选择适合自己的教材，从而快速提高 Java 语言的编程水平，是很重要的事情。

本书作者具有丰富的教学经验及软件开发经历，在编写本书时，自始至终贯穿着“面向对象”的编程思想，以 Java 语言为实现方式，力求内容设计切合实际，尽量缩小计算机专业毕业生的技能水平与人才需求之间的差距，同时引导读者深入理解 Java 语言程序设计，少走弯路。本书具有以下几个特点。

(1) 本书内容新颖，是根据当前社会发展的需要而确定的，符合计算机科学技术的发展和教学改革的要求。

(2) 本书以应用为出发点，强调实用性，适合初学者。本书作者都是长期在第一线从事高校计算机基础教育的教师，对学生的基础、特点和认识规律有深入的研究，在教学实践中积累了丰富的经验。

(3) 在教材的写法上，既注意概念的严谨和清晰，又特别注意采用读者容易理解的方法阐明看似深奥难懂的道理，做到例题丰富、通俗易懂、便于自学。

(4) 本书强调 Java 语言的实践性，提供大量实用性很强的编程实例，生动、完整，连贯性强。

(5) 本书从实际出发，知识更新及时，反映了 Java 语言和程序设计的发展成果。

本书共 10 章，主要内容包括 Java 语言概述、Java 语言运行环境及常用工具、Java 编程、JSP 编程、数据库访问、Servlet 编程、JavaBean 编程、自定义标签、MVC 模式和 Struts 应用。本书由重庆应用技术职业学院的袁明兰、河南中医药大学王晓鹏和保山中医药高等专科学校的王应邦任主编，由江西工业职业技术学院的梁兴波、装甲兵工程学院的胡雪松、吉安职业技术学院的戴琴、重庆市育才职业教育中心的陈明和南昌大学共青学院的刘德政任副主编。本书的相关资料和售后服务可通过扫本书封底的微信二维码或与 QQ (2436472462) 联系获得。

本书既可作为应用型本科院校、职业院校计算机专业的教材，也可作为各学校程序设计公共选修课的教材，同时还可用作职业教育的培训用书和 Java 语言初学者的入门教材，或为具有一定 Java 语言编程经验的开发人员学习使用。

本书难免有疏漏和不当之处，敬请各位专家及读者不吝赐教。

编者

# 目 录

<b>第 1 章 Java 语言概述</b> .....	1
1.1 Java 语言的基本知识 .....	1
1.2 Java 语言的特点和 Java 语言程序的类型 .....	3
本章小结 .....	5
本章习题 .....	5
<b>第 2 章 Java 语言运行环境及常用工具</b> .....	6
2.1 构建 Java 运行环境 .....	6
2.2 Java 语言的常用工具 .....	19
本章小结 .....	23
本章习题 .....	23
<b>第 3 章 Java 编程</b> .....	24
3.1 Java 编程基础 .....	24
3.2 流程控制结构 .....	36
3.3 Java 面向对象程序设计技术 .....	49
3.4 例外处理 .....	62
3.5 线程 .....	67
3.6 输入/输出流的常用方法 .....	76
本章小结 .....	88
本章习题 .....	89
<b>第 4 章 JSP 编程</b> .....	93
4.1 JSP 基本知识 .....	93
4.2 JSP 的指令元素 .....	96
4.3 JSP 的动作元素 .....	100
4.4 JSP 的内置对象 .....	110
本章小结 .....	134
本章习题 .....	134



<b>第 5 章 数据库访问</b> .....	135
5.1 MySQL 的安装和配置 .....	135
5.2 创建数据库和表 .....	140
5.3 SQL 基础 .....	142
5.4 与数据库建立连接 .....	143
5.5 操作数据库 .....	145
本章小结 .....	148
本章习题 .....	148
<b>第 6 章 Servlet 编程</b> .....	149
6.1 Servlet 的基本知识 .....	149
6.2 Servlet 的应用 .....	153
本章小结 .....	164
本章习题 .....	164
<b>第 7 章 JavaBean 编程</b> .....	165
7.1 JavaBean 的基本知识 .....	165
7.2 JavaBean 的使用 .....	168
本章小结 .....	179
本章习题 .....	179
<b>第 8 章 自定义标签</b> .....	180
8.1 自定义标签的基本知识 .....	180
8.2 创建标签 .....	181
本章小结 .....	192
本章习题 .....	192
<b>第 9 章 MVC 模式</b> .....	193
9.1 MVC 的基本知识 .....	193
9.2 MVC 下其他框架的应用 .....	196
本章小结 .....	197
本章习题 .....	197
<b>第 10 章 Struts 应用</b> .....	198
10.1 Struts 与 Struts 2 的基本知识 .....	198
10.2 表单数据处理 .....	209
10.3 文件上传 .....	220
10.4 Struts 2 程序的国际化 .....	230
10.5 Struts 2 标签的使用 .....	237
本章小结 .....	257
本章习题 .....	257
<b>参考文献</b> .....	258

# 第 1 章 Java 语言概述

## 本章导读

Java 是一种可以撰写跨平台应用软件的面向对象的程序设计语言。Java 技术具有卓越的通用性、高效性、平台移植性和安全性,被广泛应用于个人计算机、数据中心、游戏控制台、科学超级计算机、移动电话和互联网,同时拥有全球最大的开发者专业社群。

## 本章目标

- 了解 Java 语言的基本知识
- 掌握 Java 语言编程的特点和 Java 语言程序的类型

## 1.1 Java 语言的基本知识

Java 是由 Sun Microsystems 公司推出的“Java 面向对象程序设计语言”(以下简称“Java 语言”)和“Java 平台”的总称。Java 由 James Gosling 和同事们共同研发,并在 1995 年正式推出。Java 最初被称为“Oak”,是 1991 年为消费类电子产品的嵌入式芯片而设计的;1995 年更名为“Java”,并被重新设计用于开发互联网应用程序。用 Java 实现的 HotJava 浏览器(支持 Java Applet)显示了 Java 的魅力:跨平台、动态 Web、互联网计算。从此,Java 被广泛接受并推动了 Web 的迅速发展,常用的浏览器均支持 Java Applet。另一方面,Java 技术也在不断更新。

Java 自面世后就非常流行,其发展迅速,对 C++ 语言形成有力冲击。在全球云计算和移动互联网的产业环境下,Java 更具备了显著优势和广阔前景。2010 年,Oracle 公司收购 Sun Microsystems。

### 1.1.1 Java 语言的技术优势

与传统程序不同,Sun 公司在推出 Java 之际就将其作为一种开放的技术。全球数以万



计的 Java 开发公司被要求所设计的 Java 软件必须相互兼容。“Java 语言靠群体的力量而非公司的力量”是 Sun 公司的口号之一,并获得了广大软件开发商的认同,这与微软公司所倡导的注重精英和封闭式的模式完全不同。

Sun 公司对 Java 编程语言的解释是:Java 编程语言是种简单、面向对象、分布式、解释性、健壮、安全与系统无关、可移植、高性能、多线程和动态的语言。

Java 平台是基于 Java 语言的平台,这样的平台非常流行。因此,微软公司推出了与之竞争的 .NET 平台及模仿 Java 的 C#语言。至此,Java 的应用已十分广泛。

Java 是功能完善的通用程序设计语言,可以用来开发可靠的、要求严格的应用程序。

Java 的用途:Java 是成熟的产品,已经有 10 年的历史。80% 以上的高端企业级应用都使用 Java 平台(电信、银行等)。

自从 1995 年 Sun 公司正式发布 Java1.0 版以来,在全球范围内引发了经久不衰的 Java 热潮,Java 的版本也不断更新到 v1.1、v1.2、v1.3、v1.4,同时其内容有了巨大的改进和扩充,还出现了标准版、企业版、服务器版等满足不同需要的版本。另外,有了迅速发展的 JavaBean,其他的 Java 编译器和集成开发环境等第三方软件。

### 1.1.2 Java 语言的基本含义

(1) 抽象类:规定一个或多个抽象方法的类别本身必须被定义为“abstract”,抽象类只是用来派生子类,而不能用它来创建对象。

(2) 最终类:又称“final 类”,它只能用来创建对象,而不能被继承;与抽象类刚好相反,而且抽象类与最终类不能同时修饰同一个类。

(3) 包:Java 中的包是相关类和接口的集合,创建包必须使用关键字“package”。

(4) 继承:Java 作为面向对象编程语言,支持“继承”这一基本概念。但 Java 只支持单根继承,java.lang.Object 是所有其他类的基类。

(5) 多态类:在 Java 中,对象变量是多态的,而 Java 不支持多重继承。

(6) 接口:Java 中的接口是一系列方法的声明,是一些方法特征的集合,一个接口只有方法的特征而没有方法的实现,因此,这些方法可以在不同的地方被不同的类实现,而这些实现可以具有不同的行为。

(7) 通用编程:任何类类型的所有值都可以用 Object 类型的变量来代替。

(8) 封装:把数据和行为结合在一个包中,并对对象使用者隐藏数据的实现过程,一个对象中的数据被称为它的“实例字段”(instance field)。

(9) 重载:当多个方法具有相同的名字而含有不同的参数时,便发生重载。编译器必须挑选出调用哪个方法进行编译。

(10) 重写:也可被称为方法的“覆盖”。在 Java 中,子类可继承父类中的方法,而不需要重新编写相同的方法。但有时子类并不想原封不动地继承父类的方法,而是想作一定的修改,这就需要采用方法的重写。值得注意的是,子类在重新定义父类已有的方法时,应保持与父类完全相同的方法头声明。

(11) Class 类:Object 类中的 getClass 方法返回 Class 类型的一个实例,程序启动时包含



在 main 方法的类会被加载,虚拟机要加载其需要的所有类,每一个加载的类都要加载其需要的类。

### 1.1.3 Java 语言的基本语法

编写 Java 程序时,应注意以下几点。

(1) 大小写敏感:Java 是大小写敏感的,这就意味着标识符“Hello”与“hello”是不同的。

(2) 类名:对于所有的类来说,类名的首字母应该大写。如果类名由若干单词组成,那么每个单词的首字母应该大写。例如,MyFirstJavaClass。

(3) 方法名:所有的方法名都应该以小写字母开头。如果方法名含有若干单词,则后面的每个单词的首字母大写。例如,myFirstJavaClass。

(4) 源文件名:源文件名必须和类名相同。当保存文件的时候,应该使用类名作为文件名保存(切记 Java 是大小写敏感的),文件名的后缀为 .java。如果文件名和类名不相同则会导致编译错误。

(5) 主方法入口:所有的 Java 程序由 public static void main (String[ ] args) 方法开始执行。

## 1.2 Java 语言的特点和 Java 语言程序的类型

### 1.2.1 Java 语言的特点

Java 是一种被广泛使用的网络编程语言,是一种新的计算概念。首先,作为一种程序设计语言,它简单、面向对象,具有分布式、解释性、可移植性、健壮性、安全性等特点,并且提供了多线程、动态的机制,具有很高的性能。下面对这些特点作简要介绍。

#### 1. 简单

首先,Java 语言是一种面向对象的语言,它通过提供最基本的方法来完成指定的任务,只需理解一些基本的概念,就可以用它编写出适合于各种情况的应用程序。Java 虽然来源于 C++,但它省去了 C++ 中模糊复杂的概念,如头文件、指针、结构体、单元、运算符重载、多重继承等,并且通过实现自动垃圾收集,大大简化了程序设计者的内存管理工作。其次,Java 把编程时经常用到的一些功能封装成类库提供给程序员,这样程序员就避免了重复编写相同代码的麻烦。另外,Java 也适合于在小型机上运行,它的基本解释器及类的支持只有 40 KB 左右,加上标准类库和线程的支持也只有 215 KB 左右。

#### 2. 面向对象

Java 语言的设计集中于对象及其接口,即提供了简单的类机制及动态的接口模型。对象中封装了它的状态变量及相应的方法,实现了模块化和信息隐藏;而类则提供了一类对象



的原型,并且通过继承机制,子类可以使用父类所提供的方法,实现了代码的复用。

### 3. 分布式

Java 是面向网络的语言,通过它提供的类库可以处理 TCP/IP、HTTP、FTP 等协议,用户可以通过 URL 地址在网络上很方便地访问其他对象。

### 4. 解释性

Java 解释器直接对 Java 字节码进行解释执行。字节码本身携带了许多编译时的信息,使连接过程更加简单。运行 Java 应用程序时,由解释器根据字节码一边解释一边执行,并根据需要载入相应类库。

### 5. 高性能

用标准的 Java 解释器转换字节码来运行 Java 程序,其速度明显不够快,目前采用另一种编译方式,即准实时(Just-In-Time)编译器,或者被称为“JIT 编译器”。该技术把字节码转换成机器码并缓存,对其进行适当的优化,当再次运行该字节码时可以直接调用缓存中的代码,因此,可以实现 10~20 倍的速度提升,明显快于传统的 Java 编译器。

### 6. 可移植性

与平台无关的特性使 Java 程序可以方便地被移植到网络上的不同机器,同时,Java 的类库中也实现了与不同平台的接口,使这些类库可以被移植,这样在任何平台的任何 Java 解释器上的数据类型都是一致的。另外,Java 编译器是由 Java 语言实现的,Java 运行时系统由标准 C 实现,这使得 Java 系统本身也具有可移植性。

### 7. 健壮性

Java 在编译和运行程序时,都要对可能出现的问题进行检查,以消除错误的产生。它提供自动垃圾收集以进行内存管理,防止程序员在管理内存时出现错误,从而避免了程序运行时覆盖或修改某些重要数据。

### 8. 安全性

用于网络、分布环境下的 Java 必须要防止病毒和黑客的入侵,为此,它建立了严格的安全检查机制。Java 不支持指针,一切对内存的访问都必须通过对象的实例变量来实现,这样就防止了程序员使用“特洛伊”木马等欺骗手段访问对象的私有成员,同时也避免了指针操作中容易产生的错误。

### 9. 多线程

多线程机制使应用程序能够并行执行,更重要的是,多线程提高了程序的交互性能和实时响应性能,而且同步机制保证了对共享数据的正确操作。通过使用多线程,程序设计者可以分别用不同的线程完成特定的行为,而不需要采用全局的事件循环机制,这样就很容易地实现网络上的实时交互行为。

### 10. 动态

Java 的设计使它适合于一个不断发展的环境,在类库中可以自由地加入新的方法和实例变量而不会影响用户程序的执行,这个特性是 C++ 等其他编译器语言所无法实现的。

### 1.2.2 Java 语言程序的类型

按照实现环境的不同,Java 语言程序大致可以分为以下五种类型。

(1) Java Application:一种独立的 Java 应用程序。

(2) Java Applet:Java 小应用程序,通常在用户浏览器中运行。它是动态、安全、跨平台的网络应用程序。Java Applet 嵌入 HTML 语言,通过主页被发布到互联网。网络用户访问服务器的 Applet 时,这些 Applet 从网络上进行传输,然后在支持 Java 的浏览器中运行。由于 Java 语言的安全机制,用户一旦载入 Applet,就可以放心地来生成多媒体的用户界面或完成复杂的计算而不必担心病毒的入侵。虽然 Applet 可以和图像、声音、动画等一样从网络上下载,但它不同于这些多媒体的文件格式,它可以接收用户的输入,动态地进行改变,而不仅仅是动画的显示和声音的播放。

(3) Java Servlet:Java 服务器小程序,实质上是一个 Java 类,运行于 Web 服务器端,接收客户端的请求,并自动生成动态网页返回到客户端。

(4) JSP(Java Server Page):一种用于生成动态网页的技术,类似 ASP,基于 Servlet 技术,可实现程序与页面格式控制的分离。JSP 能够快速开发出基于 Web、独立于平台的应用程序。JSP 程序同样运行于 Web 服务器端。

(5) JavaBeans:可重用的、独立于平台的 Java 程序组件,使用相应的开发工具,可将它直接插入其他的 Java 应用程序中。



#### 本章小结

本章主要讲解了 Java 语言的基本知识、Java 语言的特点和 Java 语言程序的类型等。通过对本章的学习,读者可以了解 Java 语言的技术优势和基本语法;掌握 Java 语言的基本含义;了解 Java 语言简单、面向对象,具有分布式、解释性、高性能、可移植性、健壮性、安全性等特点,并提供多线程和动态等机制;知道 Java 语言程序的 Java Application、Java Applet、Java Servlet、JSP 和 JavaBeans 五种类型。



#### 本章习题

1. 简述 Java 语言的基本含义。
2. 简述 Java 语言的基本语法。
3. 简述 Java 语言的特点。
4. 简述 Java 语言程序常见的五种类型。

## 第2章 Java 语言运行环境及常用工具

### 本章导读

编写的 Java 应用程序要想运行起来,必须预先配置好 Java 的运行环境,即编写 Java 语言的计算机应用程序需要一个编辑、编译和运行的环境。这里将详细介绍 Java 高级语言的开发工具及相应的环境设置,读者在熟练掌握这些知识的基础上,可以编写简单的 Java 程序。

### 本章目标

- 了解 Java 运行环境的构建
- 掌握 Java 语言的常用工具

## 2.1 构建 Java 运行环境

### 2.1.1 安装 JDK

JDK 是 Java 开发工具包(Java Development Kit)的缩写,它是一种用于构建在 Java 平台上发布的应用程序、applet 和组件的开发环境。编写 Java 程序必须得有 JDK,它提供了编译 Java 和运行 Java 程序的环境,Java 语言的初学者一般都采用这种开发工具。本书采用的是 JDK v6 版本,可到网址 <http://java.sun.com/javase/downloads/index.jsp> 免费下载。

选择相应平台的 JDK 并下载后,双击安装文件,进入安装界面(见图 2-1),安装过程只需按照安装向导一步步进行即可。

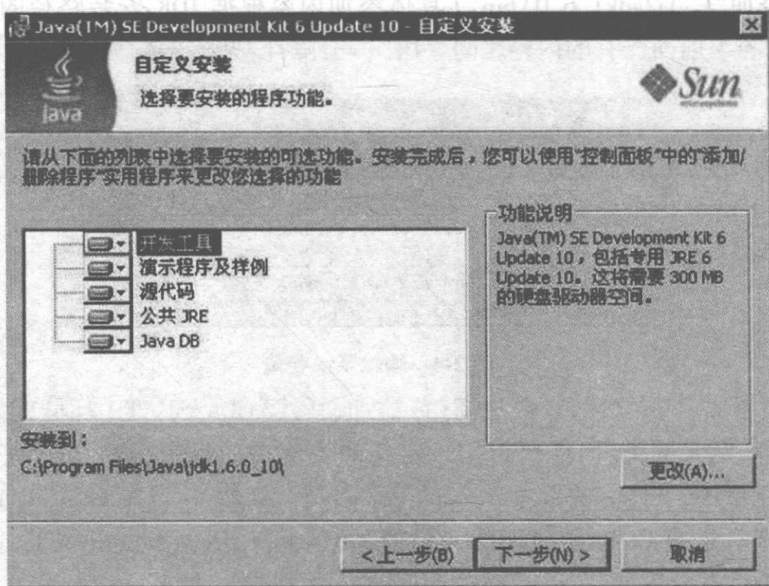


图 2-1 JDK 安装界面

为了在命令行方式使用 JDK 工具时,系统能够自动找到 JDK 工具的位置,还需要修改环境变量 Path 和设置 CLASSPATH,在 Path 的最前面增加 Java 的路径,这样 JDK 便安装好了。

在 Windows 环境下设置 Path 的步骤如下。

步骤 1:选择“开始”→“设置”→“控制面板”→“系统”命令,在“系统属性”对话框中选择“高级”选项卡,见图 2-2。

步骤 2:单击“环境变量”按钮,弹出“环境变量”对话框,见图 2-3。

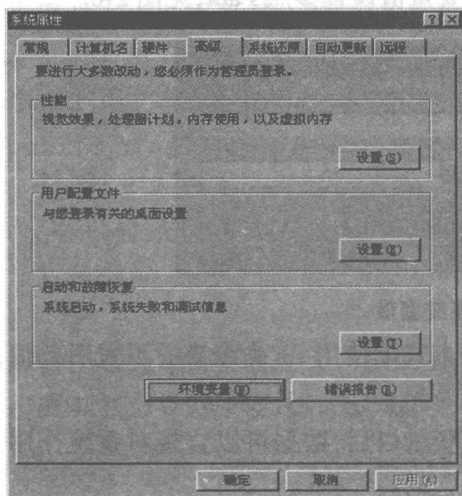


图 2-2 “系统属性”对话框

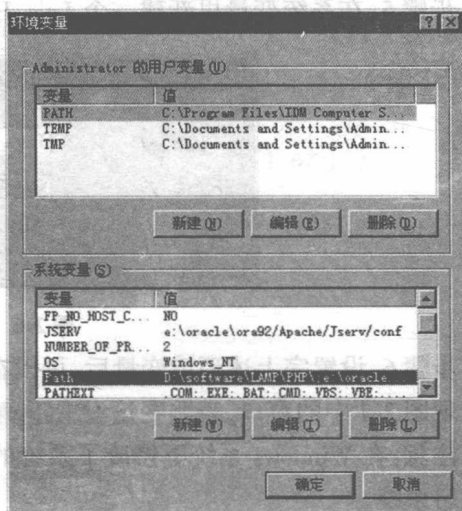


图 2-3 “环境变量”对话框

步骤 3:在系统变量中选择“Path”,单击“编辑”按钮,弹出“编辑系统变量”对话框,在

“变量值”中添加“C:\jdk1.6.0\bin”(具体添加内容根据 JDK 安装路径的不同而不同), 注意添加的内容与前面已有的内容之间要用“;”分隔开, 见图 2-4。

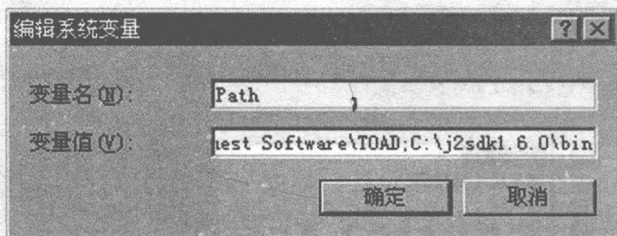


图 2-4 编辑系统变量

步骤 4: 设置完 Path 之后, 还要更新(或添加、新建) CLASSPATH 环境变量, 这是为了使系统能找到用户定义的类, 而需要将用户定义的类所在的目录(通常是当前目录)放入到 CLASSPATH 变量中, 具体方法参考 Path 的修改过程。当然了, 设置 CLASSPATH 是可选的, 如果不存在, 则新建的 CLASSPATH 的变量值为“. ; Java\_Home% \lib; % Java\_Home% \lib \ tools. jar”;如果没有设置, 则在运行 Java 程序时必须显式指明 CLASSPATH。

注意: 分号前面的点一定要保留, 见图 2-5。

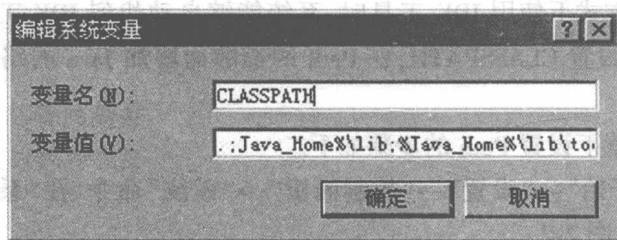


图 2-5 更新环境变量

步骤 5: 在系统变量里新建一个 Java\_Home, 变量值设置参照 Path, 见图 2-6。

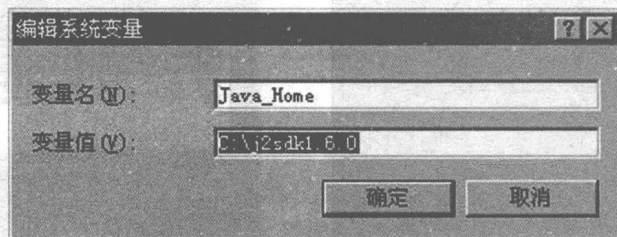
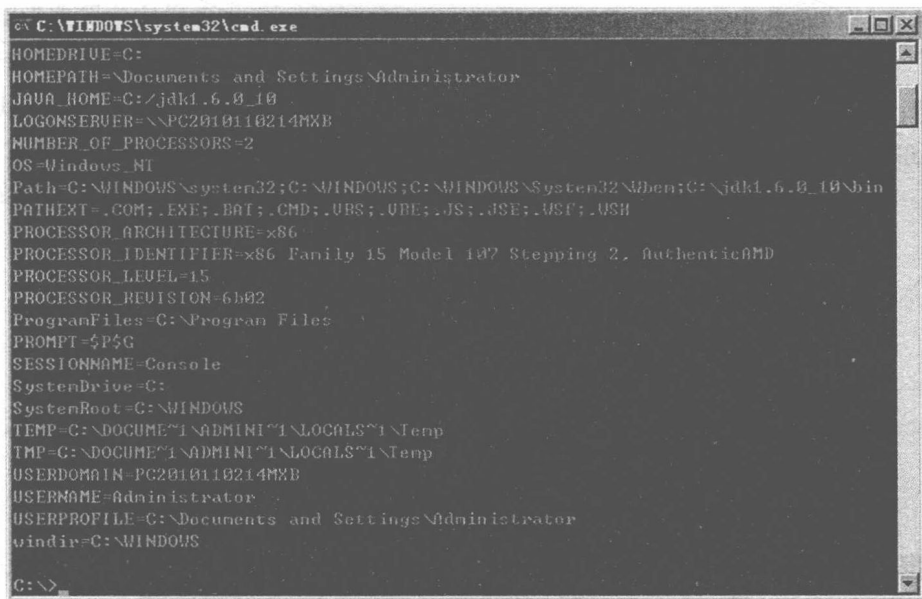


图 2-6 新建系统变量

步骤 6: 设置完上述环境变量后, 可以在命令方式下使用 set 命令查看设置后的值, 此时会出现图 2-7 中的窗口, 表明变量设置正确。

注意: 在 Windows 系统下修改了系统的环境变量以后, 数据可以直接对系统作用, 而无需重启系统。

读者可以到 Sun 公司的网站上查找最新的版本及信息。



```

C:\WINDOWS\system32\cmd.exe
HOMEDRIVE=C:
HOMEPATH=\Documents and Settings\Administrator
JAVA_HOME=C:\jdk1.6.0_10
LOGONSERVER=\\PC2010110214MXB
NUMBER_OF_PROCESSORS=2
OS=Windows_NT
Path=C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\bin;C:\jdk1.6.0_10\bin
PATHEXT=.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH
PROCESSOR_ARCHITECTURE=x86
PROCESSOR_IDENTIFIER=x86 Family 15 Model 147 Stepping 2, AuthenticAMD
PROCESSOR_LEVEL=15
PROCESSOR_REVISION=6b02
ProgramFiles=C:\Program Files
PROMPT=$P$G
SESSIONNAME=Console
SystemDrive=C:
SystemRoot=C:\WINDOWS
TEMP=C:\DOCUMENT1\ADMINI1\LOCALS1\Temp
TMP=C:\DOCUMENT1\ADMINI1\LOCALS1\Temp
USERDOMAIN=PC2010110214MXB
USERNAME=Administrator
USERPROFILE=C:\Documents and Settings\Administrator
windir=C:\WINDOWS

C:\>

```

图 2-7 查看环境变量的设置结果

## 2.1.2 代码编辑工具

JDK 中的所有工具都是基于命令行方式的。相比那些高级的开发环境, JDK 的这种方式反而显得简单明了, 易于学习。它可以通过任何文本编辑器来编写 Java 源文件, 然后在命令方式下通过 `javac` 和 `java` 命令来执行。下面介绍几种在 Java 开发时常用的文本编辑器。

### 1. 记事本

先编写一个简单的 Java 测试例子。

步骤 1: 在 C:\ 下新建一个文本文件, 并命名为“Hello.java”, 文件的扩展名必须是“.java”。

步骤 2: 用文本编辑器(记事本)打开该文件, 输入如下内容。

```

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

```

步骤 3: 在命令行中输入“C:\javac Hello.java”。

步骤 4: 在命令行中输入“java Hello”, 则该程序被执行, 见图 2-8。

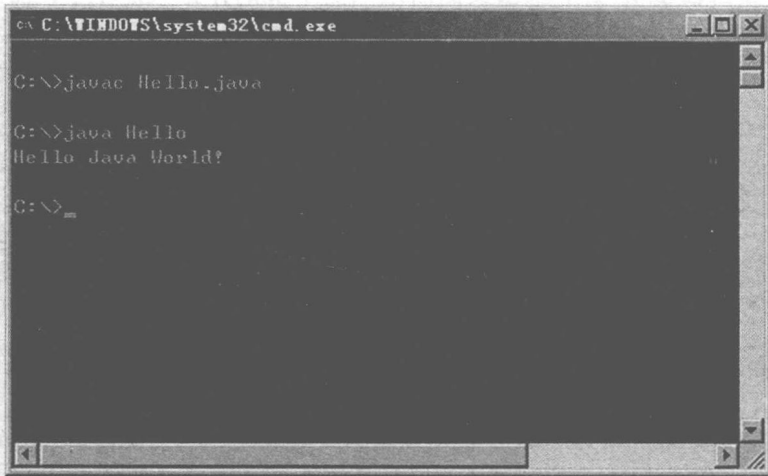


图 2-8 Hello.java 的执行结果

根据上面的例子可以看出,开发 Java 程序总体上可以分为以下三个步骤:

- (1) 编写 Java 源文件。它是一种文本文件,扩展名为 .java。
- (2) 编译 Java 源文件。也就是将 Java 源文件编译(Compile)成 Java 类文件(扩展名为 .class)。
- (3) 运行 Java 程序。Java 程序可以分为 Java Application(Java 应用程序)和 Java Applet(Java 小应用程序)。其中,Java Application 必须通过 Java 解释器(Java.exe)来解释执行其字节码文件,而 Java Applet 必须使用支持它的浏览器(如 IE)运行。

## 2. UltraEdit

UltraEdit 是一套功能强大的文本编辑器,可以编辑文本、十六进制数值、ASCII 码,完全可以取代记事本(如果计算机配置足够强大),内建英文单词检查、C++及 VB 指令突显,可同时编辑多个文件,而且即使打开很大的文件其速度也不会慢。该软件附有 HTML 标签颜色显示、搜寻替换及无限制的还原功能,一般用它来修改 EXE 或 DLL 文件,是能够满足用户一切编辑需要的编辑器。最值得称道的是,它可以对各种源代码进行语法着色,使用户能够清晰地分辨代码中的各种成分。同时,也可以用它来开发 HTML、JSP、ASP 等。它不仅可以区分其中的 scriptlet(小脚本)和 HTML 代码,对它们进行很好的着色,而且提供了几乎全部的 HTML TAG 和特殊字符,方便轻松查到。当它对 HTML 进行着色时,可以做到对 TAG、PROPERTY 和 VALUE 进行不同的着色。

它的配置很简单,步骤如下。

步骤 1:运行 UltraEdit,选择“高级”→“工具配置”命令,打开“工具配置”对话框,在“命令行”文本框中输入“javac.exe %n%e”,在“工作目录”文本框中输入“%p”,在“菜单项名称”文本框中输入“Java 编译”(见图 2-9);在“选项”选项卡中勾选“先保存所有文件”复选框;在“输出”选项卡中勾选“输出到列表框”和“捕捉输出”复选框,然后单击“插入”按钮。

步骤 2:再次单击“插入”按钮,插入“Java 运行”。“Java 运行”的设置与“Java 编译”大致相同。在“命令行”文本框中输入“java.exe %n”,在“工作目录”文本框中输入“%p”,在“菜单项名称”文本框中输入“Java 运行”;在“选项”选项卡中勾选“先保存所有文件”复选

框;在“输出”选项卡中勾选“输出到列表框”和“捕捉输出”复选框,然后单击“应用”按钮和“确定”按钮(见图 2-10)。

配置成功之后,在“高级”菜单项下面会生成配置好的菜单项,并带有快捷方式,默认添加的第一个菜单项的快捷方式为 Ctrl+Shift+0,第二个菜单项的快捷方式为 Ctrl+Shift+1,以此类推(见图 2-11)。

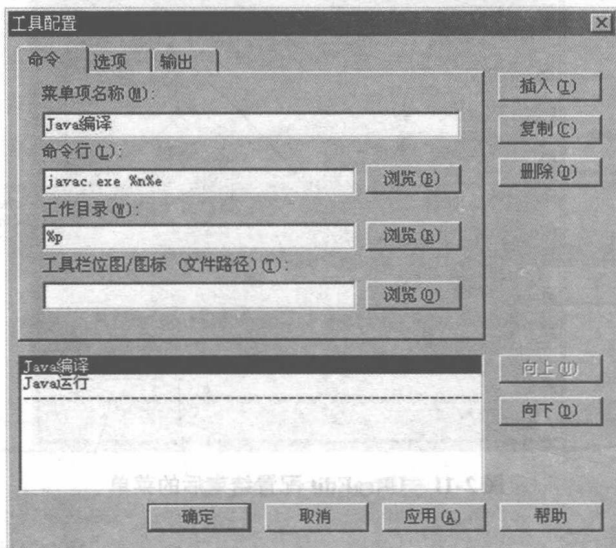


图 2-9 “Java 编译”的设置

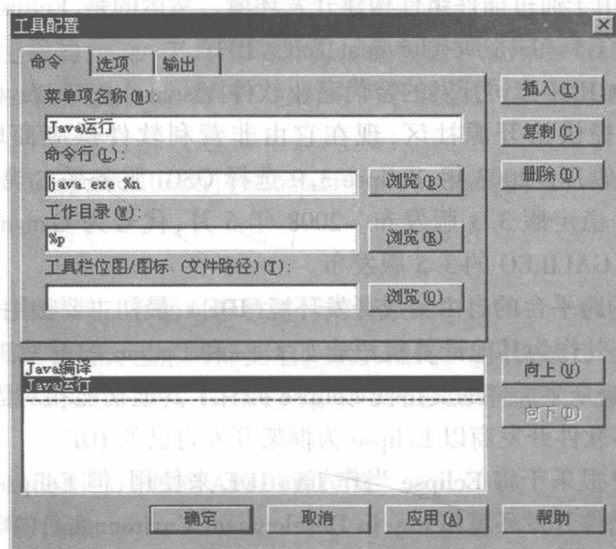


图 2-10 “Java 运行”的设置