

高等学校计算机专业教材精选·算法与程序设计

Java 程序设计教程

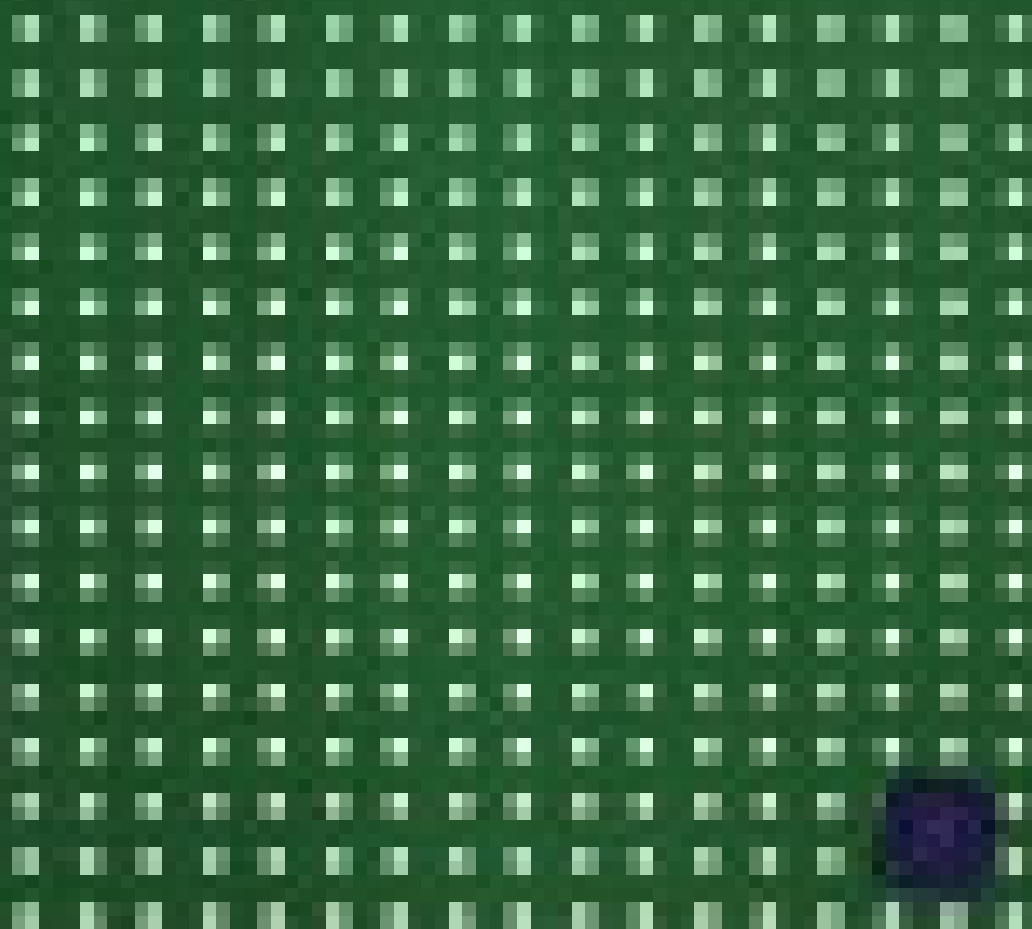
孙燮华 编著



清华大学出版社

Java 程序设计教程

清华大学出版社



TP312/2664

2008

高等学校计算机专业教材精选·算法与程序设计

Java 程序设计教程

孙燮华 编著

清华大学出版社
北京

内 容 简 介

由于 Internet、无线网络和移动通信的迅速发展,手机游戏软件和嵌入式家用电器软件的开发等,使 Java 语言得到普遍应用,成为首选的“网络”开发工具。

本书分为两部分。第一部分是 Java 核心基础,由前 10 章组成,其中包括 Java 语言基础、面向对象程序设计、图形用户界面设计、Applet、图形 GUI 设计和 Java 数据流等基本知识和技术。内容完全覆盖 Java 二级程序员考试大纲。第二部分为 Java 应用技术,由第 11 章和第 12 章组成,内容包括 Java 多线程和 JDBC 数据库技术。

本书写作严谨,所有算法原理和相应的程序都经过核对,并编译通过运行。本书同时特别注重培养学生创新的思想,为此写入了一些能发挥学生创新的题材,结合书中的创新题,让学生沿着实践—设计—创新的方向得到训练。

本书选配了较多的习题。除了通常的选择题和编程题外,还加入了实践题和创新题。实践题是为学生在计算机上输入程序,反复体会程序中使用的有关语句、算法和技术。在实践题的选择上还注意到趣味性和应用性。相信这部分实践题会得到学生们的大量选用练习。

本书通俗易懂、深入浅出,适合计算机专业和非计算机专业的教学和自学者使用,也适合软件开发人员及其他有关人员学习。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13501256678 13801310933

图书在版编目(CIP)数据

Java 程序设计教程/孙夔华编著. —北京:清华大学出版社,2008.1

(高等学校计算机专业教材精选·算法与程序设计)

ISBN 978-7-302-16104-2

I. J… II. 孙… 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(2007)第 141466 号

责任编辑:张 民 李 晔

责任校对:白 蕾

责任印制:何 芊

出版发行:清华大学出版社

<http://www.tup.com.cn>

c-service@tup.tsinghua.edu.cn

社 总 机:010-62770175

投稿咨询:010-62772015

地 址:北京清华大学学研大厦 A 座

邮 编:100084

邮购热线:010-62786544

客户服务:010-62776969

印 刷 者:北京市人民文学印刷厂

装 订 者:三河市李旗庄少明装订厂

经 销:全国新华书店

开 本:185×260 印 张:19.5

字 数:466 千字

版 次:2008 年 1 月第 1 版

印 次:2008 年 1 月第 1 次印刷

印 数:1~5000

定 价:28.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。
联系电话:010-62770177 转 3103 产品编号:027045-01

出版说明

我国高等学校计算机教育近年来迅猛发展,应用所学计算机知识解决实际问题,已经成为当代大学生的必备能力。

时代的进步与社会的发展对高等学校计算机教育的质量提出了更高、更新的要求。现在,很多高等学校都在积极探索符合自身特点的教学模式,涌现出一大批非常优秀的精品课程。

为了适应社会的需求,满足计算机教育的发展需要,清华大学出版社在进行了大量调查研究的基础上,组织编写了《高等学校计算机专业教材精选》。本套教材从全国各高校的优秀计算机教材中精挑细选了一批很有代表性且特色鲜明的计算机精品教材,把作者们对各自所授计算机课程的独特理解和先进经验推荐给全国师生。

本系列教材特点如下。

(1) 编写目的明确。本套教材主要面向广大高校的计算机专业学生,使学生通过本套教材,学习计算机科学与技术方面的基本理论和基本知识,接受应用计算机解决实际问题的基本训练。

(2) 注重编写理念。本套教材作者群为各校相应课程的主讲,有一定经验积累,且编写思路清晰,有独特的教学思路和指导思想,其教学经验具有推广价值。本套教材中不乏各类精品课配套教材,并力图努力把不同学校的教学特点反映到每本教材中。

(3) 理论知识与实践相结合。本套教材贯彻从实践中来到实践中去的原则,书中的许多必须掌握的理论都将结合实例来讲,同时注重培养学生分析、解决问题的能力,满足社会用人要求。

(4) 易教易用,合理适当。本套教材编写时注意结合教学实际的课时数,把握教材的篇幅。同时,对一些知识点按教育部教学指导委员会的最新精神进行合理取舍与难易控制。

(5) 注重教材的立体化配套。大多数教材都将配套教师用课件、习题及其解答,学生上机实验指导、教学网站等辅助教学资源,方便教学。

随着本套教材陆续出版,相信能够得到广大读者的认可和支持,为我国计算机教材建设及计算机教学水平的提高,为计算机教育事业的发展做出应有的贡献。

清华大学出版社

前 言

Java 语言是“网络语言”。在当今网络时代,随着 Internet 和无线网络通信的日益扩大和普及,Java 语言的使用也随之得到扩大和普及。许多计算机、通信和信息技术公司在进行涉及网络、手机、游戏或嵌入式产品开发时,已离不开 Java 语言了。因此,可以说,Java 语言是程序员必须掌握的语言。

Java 语言也是一种优秀的语言。其优越性可以从 Microsoft 公司推出的最新一代语言 C#.NET 中看出。C#.NET 语言虽然属于 C 系列语言——C/C++/C#,但它更像是 Java 语言。它与 Java 语言一样取消了指针和类的多继承。总之,它更多地采用 Java 语言的体制。因此,学习和掌握 Java 语言对于学习最新一代 C#.NET 语言也是有意义的。

本教程共分为两部分。第一部分 Java 核心基础,由前 10 章组成,其内容完全覆盖 Java 二级程序员考试大纲(修订)。对于计算机专业的教学,尤其是学习过 C++ 程序设计语言的学生,可以选学或跳过第 2 章~第 4 章的部分内容。第二部分 Java 应用技术,共有 2 章。主要供计算机专业教学和非计算机专业学生和自学者选学。

作者认为,要熟练掌握一门编程语言,不经过大量的编程和实践是不可能的。因此,本书选配了较多的习题。除了通常的选择题和编程题外,还加入了实践题。实践题是为学生在计算机上输入程序,反复体会程序中使用的有关语句、算法和技术。在实践题的选择上还注意到趣味性和应用性,相信这部分实践题会得到学生们的大量选用练习。

作者在教学过程中,发现一些学生不满足于对 Java 语言的一般性了解,渴望有更为深入的学习。这正是一个青年学生的最难能可贵之处——一种独立探索问题,寻求新主意和新动机。一本好书应该能够激发学生的进一步求知、求真和创新的欲望。为培养学生的创新、探索和开发实践能力,本书首次在第 4 章、第 7 章和第 9 章习题中布置了创新题。为这部分学生提供了进一步思考 and 研究的题材。这部分题材也可以作为学生课外科技活动或毕业设计的题材,供大学生撰写习作性的研究论文之用。

本书选配的习题很多,有些习题选自国内外的习题集。大量的习题和其内容完全覆盖 Java 二级程序员考试大纲,其目的是使本书不仅适用于计算机专业教学,也适用于非计算机专业的教学和自学者使用。

博士生姚伏天参加了本书第 5 章和习题的编写,其余均由孙夔华完成。

本书的全部程序均在 J2SDK1.4 和 J2SDK 1.5 下编译通过,重要之处还增加了较为详细的注释。虽然本书在成书前经过多次教学检验和修改,在出版前又进行多次审核校对,但在书中难免还有一些错误,恳请读者指正。

最后,作者对本书写作过程中参考和引用过的一些参考书的作者们表示感谢。对本书出版前作为讲义试用期间提出过宝贵意见和提供过帮助的所有同事和学生表示感谢。

作 者

2007 年 11 月

目 录

第一部分 Java 核心基础

第 1 章 概论	1
1.1 初识 Java	1
1.1.1 第一个 Java 独立应用程序	1
1.1.2 第一个 Java 小应用程序	2
1.1.3 第一个 MIDlet 手机程序	3
1.2 为什么要学习 Java	5
1.3 Java 的开发运行环境	6
1.3.1 Java 虚拟机(JVM)	7
1.3.2 Java 运行系统	8
1.4 Java 程序开发	9
1.4.1 Java API	9
1.4.2 JDK 的操作命令	9
1.4.3 Java 程序的编写与编译过程	10
习题 1	12
第 2 章 Java 语言基础	14
2.1 标识符与关键词	14
2.1.1 标识符	14
2.1.2 关键词	15
2.2 Java 的数据类型	15
2.2.1 基本数据类型	16
2.2.2 数据类型的转换	18
2.2.3 变量、说明和赋值	18
2.2.4 复合数据类型	19
2.3 表达式与语句	20
2.3.1 运算符与表达式	20
2.3.2 语句	28
2.4 程序控制流	29
2.4.1 if 语句	30
2.4.2 switch 语句	31
2.4.3 while 和 do-while 语句	32
2.4.4 for 语句	35

2.4.5	跳转语句	35
习题 2	38
第 3 章	数组与字符串	43
3.1	数组	43
3.1.1	一维数组	43
3.1.2	多维数组	48
3.2	数组作为参数或返回值的方法调用	52
3.2.1	传递数组	52
3.2.2	返回值为数组类型的方法	52
3.3	数组元素的排序与查找等算法	54
3.3.1	数组元素排序算法	54
3.3.2	数组元素查找算法	57
3.3.3	Java API 中的排序与查找方法	59
3.4	字符串和字体	62
3.4.1	String 类	62
3.4.2	StringBuffer 类	68
3.4.3	Character 类	70
3.4.4	字体类 Font	72
习题 3	75
第 4 章	Java 的面向对象特性	78
4.1	概述	78
4.2	类和对象	79
4.2.1	类的定义	79
4.2.2	对象的创建、初始化和使用	81
4.2.3	构造方法	82
4.2.4	成员变量和成员方法	84
4.2.5	成员方法的递归和递推	87
4.2.6	方法的重载	89
4.2.7	抽象类	92
4.2.8	封装	93
4.3	类的继承和多态	93
4.3.1	继承的概念	93
4.3.2	继承的实现	94
4.3.3	单重继承	95
4.3.4	多态性	95
4.3.5	方法和域的覆盖	96
4.3.6	运算符 instanceof	99

4.4	包与接口	100
4.4.1	Java 的包	100
4.4.2	接口	101
4.5	Java 程序的结构	103
4.5.1	访问控制	103
4.5.2	Java 程序的结构	106
	习题 4	108
第 5 章	Java 的异常处理	112
5.1	Java 的异常处理机制	112
5.2	异常举例	113
5.3	异常的处理	114
5.4	异常的分类	116
5.4.1	3 类意外	116
5.4.2	常用的异常	117
5.5	抛出异常	117
5.6	创建自定义异常	118
	习题 5	119
第 6 章	Java 的图形界面设计与事件处理	124
6.1	文本框与文本域	124
6.1.1	文本框 TextField	124
6.1.2	文本域 TextArea	126
6.2	标签、按钮与事件	128
6.2.1	标签 Label	128
6.2.2	按钮 Button	129
6.2.3	AWT 控件与 Swing 控件	131
6.3	面板与框架	134
6.3.1	面板 Panel	134
6.3.2	框架 Frame	136
6.4	事件与监视器接口	138
6.4.1	事件类 Event	139
6.4.2	事件的起源	142
6.4.3	键盘事件举例	142
	习题 6	145
第 7 章	布局管理器	148
7.1	网格布局管理器	148
7.1.1	带有间隔的网格布局管理器	148

7.1.2	整数类 Integer、浮点数类 Float 和数学类 Math	149
7.1.3	简单加法器的网格布局	151
7.2	边界布局管理器 BorderLayout	153
7.3	网袋布局管理器 GridBagLayout	156
7.3.1	关于 Java 控件的自然尺寸和它的显示区域	159
7.3.2	关于参数 weightx 和 weighty	160
7.3.3	控件间的相互影响	163
7.4	不使用布局管理器	166
	习题 7	168
第 8 章	小应用程序 Applet	171
8.1	Applet 基础	171
8.1.1	Applet 类的定义与成员方法	171
8.1.2	Applet 的生命周期	173
8.1.3	独立应用程序与 Applet 的转换	174
8.1.4	确保 Applet 的正常运行	177
8.2	HTML 与标记 <APPLET>	177
8.2.1	<APPLET> 标记的属性	178
8.2.2	利用标记参数向 Applet 传递信息	181
8.3	Applet 的应用	184
8.3.1	Applet 与图形用户界面(GUI)	184
8.3.2	Applet 应用举例	185
	习题 8	191
第 9 章	图形 GUI 设计	194
9.1	Graphics 类与 Color 类	194
9.1.1	Graphics 类	194
9.1.2	Color 类	195
9.2	建立绘图程序	197
9.2.1	Point 类及其应用	197
9.2.2	布尔标志的设计与绘图程序	198
9.3	图形与图像的显示	202
9.4	Java2D 与图形变换	205
9.4.1	Graphics2D 类的几何变换	206
9.4.2	用 GeneralPath 类画直线与多边形	210
9.4.3	封装类及其用法	211
9.5	XOR 绘图模式	214
	习题 9	216

第 10 章 Java 数据流与数据传输	221
10.1 基本输入/输出流类	221
10.1.1 InputStream 类	221
10.1.2 OutputStream 类	224
10.1.3 Reader 类和 Writer 类	228
10.2 文件的输入/输出	230
10.2.1 File 类	230
10.2.2 FileInputStream 类和 FileOutputStream 类	234
10.2.3 字节文件输入/输出流的读/写	235
10.2.4 FileReader 类和 FileWriter 类	237
10.3 RandomAccessFile 类	240
10.4 数据传输	243
习题 10	246

第二部分 Java 应用技术

第 11 章 多线程	249
11.1 关于线程	249
11.1.1 不调用多线程和调用多线程比较试验	249
11.1.2 线程的状态	253
11.1.3 与线程有关的类	253
11.2 创建线程	257
11.2.1 扩展线程类	257
11.2.2 利用 Runnable 接口	258
11.3 多线程的分组管理	265
11.4 线程同步	267
11.4.1 线程不同步产生的问题	267
11.4.2 同步线程	271
习题 11	272
第 12 章 JDBC 与数据库	274
12.1 关系数据库与 SQL 语言	274
12.1.1 关系数据库的基本概念	274
12.1.2 数据定义语言	275
12.1.3 数据操纵语言	275
12.1.4 数据查询语言	276
12.2 使用 JDBC 连接数据库	276
12.2.1 JDBC 结构	276
12.2.2 4 类 JDBC 驱动程序	278
12.2.3 JDBC 编程要点	278

12.2.4	常用的 JDBC 类与方法	279
12.2.5	安装 ODBC 驱动程序示例	282
12.3	JDBC 编程实例	285
12.3.1	创建和删除数据表	285
12.3.2	插入记录	287
12.3.3	更新数据	288
12.3.4	删除记录	290
12.3.5	查询数据库	291
习题 12		293
参考文献		296

第一部分 Java 核心基础

第 1 章 概 论

什么是 Java 语言? Java 语言有什么用处? 学过其他语言的读者还会问“已掌握了一门语言,为什么还要学习 Java 语言?”本章作为概论将要回答这些问题,并将介绍 Java 的开发包 J2SDK 及其应用。

1.1 初识 Java

传统的第一个 Java 程序是“Hello Java!”。但是,随着 Java 的发展,Java 语言已经成为“第一网络语言”,仅提供一个简单的“Hello Java!”程序已经不能满足目前读者的需要了。当今的“网络”概念不仅仅是连结计算机的 Internet 网络,而是一个连结包括手机以及各种家用电器等在内的相当广泛的网络。因此,这里提供了 3 个程序: Java 独立应用程序(Application)、Java 小应用程序(Applet)和 MIDlet 手机程序。

1.1.1 第一个 Java 独立应用程序

独立应用程序(Application)就是带有 main()方法,如同 C 语言程序能独立运行的 Java 程序。下面的程序将在 DOS 屏幕上显示几行文字。

例 1-1 第一个 Java 独立应用程序。

```
//Welcome.java
public class Welcome{
    public static void main(String[] args){
        String[] greeting=new String[2]; //定义字符串数组 greeting
        greeting[0]="Welcome to Java!";
        greeting[1]="Hi, Java!";
        for (int i=0; i<greeting.length; i++){
            System.out.println(greeting[i]);
        }
    }
}
```

程序运行结果如下:

```
Welcome to Java!
Hi, Java!
```

1.1.2 第一个 Java 小应用程序

小应用程序即 Applet。Applet 是由单词 Application 开头的 3 个字母和英语后缀-let (小)创造的。Java 还创造了 Servlet(小服务器程序)、MIDlet(小 MID 程序)等名词。下面的例 1-2 就是一个 MIDlet。科学和工程上似乎有一个以“-let”命名的“趋势”，可能是由 wavelet(小波)这个名词引发的。

下面来看第一个“小”应用程序。它不能独立运行，只能嵌入在 Web 页面的 HTML 文件中运行。因此，它与独立应用程序的主要区别是必须有下面的语句：

```
import java.applet.Applet; //或使用通配符"*"语句 import java.applet.*;
public class AppletFileName extends Applet{
```

要在 Web 中运行 Java 程序，还需要相应的 HTML 文件。

例 1-2 第一个 Java Applet 源程序及其 HTML 文件。

```
//hello.java
import java.awt.*;
import java.applet.Applet;
public class hello extends Applet{
    public void paint(Graphics g){
        g.setFont(new Font(" ", Font.BOLD, 36));
        g.drawString("Welcome!", 60, 60);
        g.drawString("您好，来自 Java 的问候!", 60, 100);
    }
}
//hello.html，嵌入在网页中的 html 文件
<html>
    <applet code=hello.class width=500 height=300>
    </applet>
</html>
```

在 J2SDK1.4.0 的模拟浏览器中显示的结果如图 1-1 所示。



图 1-1 在 J2SDK1.4.0 中显示的结果

在 Internet Explorer 中的实际显示如图 1-2 所示。中文在 Java 中的显示有时会出现乱码，其主要原因是 Internet Explorer 中没有安装 JRE (Java 运行时环境, Java Runtime Environment)。读者可以到网站 <http://java.sun.com> 上免费下载 JRE。



图 1-2 在 Internet Explorer 中显示 Applet

1.1.3 第一个 MIDlet 手机程序

用 Java 编写手机程序的集成开发环境较多。下面以 WKToolkit1.04 为例,介绍手机程序的编写、编译和运行。图 1-3 是 WKToolkit1.04 开发环境界面。

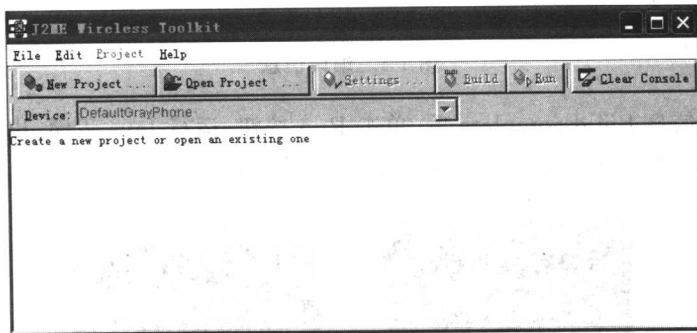


图 1-3 WKToolkit1.04 界面

例 1-3 第一个手机程序,用 J2ME 编写。

```
package com.mycom;
import javax.microedition.midlet.*;
import javax.microedition.lcdui.*;

public class HelloWorld extends MIDlet{
    Display display;
    Form form;
    Image image;

    public HelloWorld(){
        display=Display.getDisplay(this);
        form=new Form("我的第一个 MIDP 程序");
        form.append("Hello World!");
        try{
            image=Image.createImage("/com/mycom/_dukeok8.png");
```

```

    form.append(image);
} catch (Exception ex) {}
}
public void startApp() {
    display.setCurrent(form); System.out.println("Hello World!");
}
public void pauseApp() {}
public void destroyApp(boolean conditional) {}
}

```

单击 Build 按钮,进行编译,结果如图 1-4 所示。

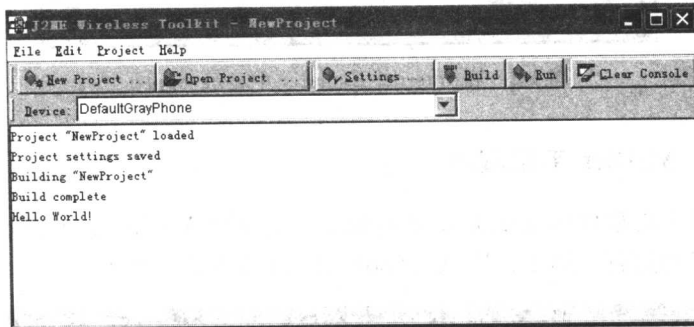


图 1-4 编译信息

在出现提示信息 Build complete 后,单击 Run 按钮,运行 Java 程序,出现如图 1-5(a)所示的模拟手机图。在手机上单击 Launch 按钮后,结果如图 1-5(b)所示。



图 1-5 模拟手机图

1.2 为什么要学习 Java

通过以上 3 个例子,我们已经初步认识了 Java,也了解了 Java 的某些应用。现在,回答第二个问题,为什么要学习 Java? 尤其是当学过 C 语言、C++、Visual Basic、Delphi 等之后,为什么还要学习 Java 语言? 这要从 Java 的诞生谈起。

Java 语言诞生于 1991 年。它是由 Sun Microsystems 的一个开发小组在开发 Green 项目时完成的。该项目本意是开发一种用于消费类电子产品并且与平台无关的软件技术。到 1994 年,Sun 的两个开发人员创建了 HotJava 的第一个版本。当时称为 WebRunner,这就是现在在 Web 上使用的图形浏览器。此后,又被称为 Java。1995 年 5 月,Sun 公司在 SunWorld95 大会上正式推出了 Java 语言。Java 给出的最重要的承诺是将用户同信息连接到一起,无论那些信息来自 Web 服务器、数据库、信息供应商还是能够想象到的其他任何信息源。其实 Java 是当时能真正实现这些承诺的唯一一种编程语言。

由于 Internet 的飞速发展,Java 语言立即成为全球信息网上的一颗明星。

Java 语言是一种网络编程语言,它最大限度地利用了网络资源。用 Java 可以编制如下程序:

- 独立应用程序(Application)。
- 嵌入 Web 页的 Java 小应用程序(Applet)。
- 用于 Web 页的小服务器程序(Servlet)。

Applet 可以跨平台、跨操作系统、跨网络运行。此外,由于 Applet 代码短小,易于在网络上快速下载和发送,而且具有不需要修改应用程序就可以增加 Web 页新功能的特性,因此,Java 在 Internet/Intranet 中得到了广泛应用。此外,Java 配有丰富的类库,为用户编程提供了极大的方便。

Java 语言与其他主要语言相比较,其独有的特点是平台无关性。

1. 平台无关性

众所周知,Internet 是由世界范围内数以万计的各种各样的计算机系统组成。这些系统中的计算机软件和硬件千差万别。要让应用软件在网络上的任何一台计算机系统中正常运行,就像是专门为该系统设计的一样,就必须使软件具有平台无关性。也就是说,软件本身不受计算机硬件和操作系统的限制,软件代码可以在不同的计算机环境中良好运行。

长期以来,软件的平台无关性一直是软件发展和编程人员追求的目标,而 Java 就是一种具有平台无关性的编程语言。它在源程序级保证了基本数据类型与平台无关。Java 源程序经编译后产生的二进制代码是一种与具体机器指令无关的指令集合,通过 Java 虚拟机(Java Virtual Machine,JVM),可以在不同的平台上运行。也就是说,JVM 为我们隔离了复杂的外部网络世界。只要计算机安装了 JVM,就可以运行 Java 程序。这就是说,只要安装了支持 Java 的浏览器,如 Internet Explorer、Netscape Navigator 等,Applet 就可运行于 Windows 2000/XP/Vista、Mac 或 UNIX 等不同的平台上。

Java 语言与平台无关性相应的特点是安全性。

2. 安全性

在网络和分布环境下,防止病毒入侵和恶意攻击是必须重视的重大问题。作为 Web 编