

● 高等学校 21 世纪计算机教材

JAVA

程序设计与实践

汤 庸 主编
冯智圣 申建芳 肖 炜 编著



冶金工业出版社

清华大学出版社

JAVA

程序设计—实践

第 1 版



高等学校 21 世纪计算机教材

JAVA 程序设计与实践

汤 庸 主编

冯智圣 申建芳 肖 炜 编著

北 京

冶金工业出版社

2003

内 容 简 介

Java 的跨平台兼容性以及其优良的性能使之成为 Web 开发中首选的工具。本书循序渐进, 系统、全面地介绍了 Java 的各项功能, 并结合面向对象、可视化编程等方法, 选取常用的例子加深读者的理解。本书共分为 13 章, 从 Java 的基本知识、编程基础和面向对象编程讲起, 详细介绍了 Java 的程序和开发环境、文件系统和流处理、异常、用户界面技术、线程及多线程技术、多媒体技术、数据库技术、网络编程等方面的内容, 还介绍了包括 JavaBean、J2EE、XML 等技术的 Java 高级应用, 最后以一个物料管理系统为例, 综合运用全书所有知识, 帮助读者巩固所学内容。本书内容新颖, 实例丰富, 实用性强, 书中的所有例子都在中文 Windows 2000/XP、JBuilder 8.0 环境下编译通过。

本书不仅可以作为 Java 自学者和从事 Java 程序开发的技术人员的学习和参考用书, 也可以作为各大中专院校相关专业和 Java 培训班的参考用书。

图书在版编目 (CIP) 数据

JAVA 程序设计与实践 / 汤庸主编. —北京: 冶金工业出版社, 2003.11
ISBN 7-5024-3373-2

I. J... II. 汤... III. JAVA 语言—程序设计
IV. TP312

中国版本图书馆 CIP 数据核字 (2003) 第 092126 号

出版人 曹胜利 (北京沙滩嵩祝院北巷 39 号, 邮编 100009)

责任编辑 程志宏

广东惠阳印刷厂印刷; 冶金工业出版社发行; 各地新华书店经销

2003 年 11 月第 1 版, 2003 年 11 月第 1 次印刷

787mm × 1092mm 1/16; 24.75 印张; 572 千字; 386 页; 1-5000 册

39.00 元

冶金工业出版社发行部 电话: (010) 64044283 传真: (010) 64027893

冶金书店 地址: 北京东四西大街 46 号 (100711) 电话: (010) 65289081

(本社图书如有印装质量问题, 本社发行部负责退换)

前 言

一、关于 Java

Java 是 Sun 公司推出的面向对象、跨平台的网络编程语言,随着网络的飞速发展,Java 作为一种网络开发语言越来越受到广泛关注,在许多方面改变了 Web 和 Web 程序设计,使 Web 界面更生动华丽。

本书主要包括 Java 应用程序编程及设计领域的许多内容,将引导读者在 Java 世界中遨游。Java 是面向对象的语言,读者最好具有面向对象程序设计的经验。本书的目的是通过浅显易懂的语言,详尽的实例帮助读者全面、完整地理解 Java 的基本原理及使用技巧。这些原理和实践将贯穿全书,并且结合公共 Java 标准库中的一系列例子进行说明。

二、本书结构

本书共分为 13 章,各章内容安排如下:

第 1 章:Java 概述。主要介绍了 Java 的基本知识、编程环境的设置和简单的 Java 程序编辑等内容。

第 2 章:Java 编程基础。主要介绍了 Java 的语法基础、数据类型、基本语句和流程控制、类型转换等内容。

第 3 章:面向对象编程。主要介绍了面向对象的程序设计、常用术语、类和继承、接口和包、常见问题和解决方法等内容。

第 4 章:Java 程序和开发环境。主要介绍了创建应用程序和小应用程序、Java 环境和工具以及 JDBC 等内容。

第 5 章:文件系统和流处理。主要介绍了文件系统和流系统等内容。

第 6 章:异常。主要介绍了异常的概念、分类、处理和定义自己的异常类等内容。

第 7 章:Java 用户界面技术。主要介绍了基本的 AWT 组件、Swing 组件和事件处理等内容。

第 8 章:Java 线程及多线程技术。主要介绍了线程的基本知识、临界区、信号量、锁、阻塞、优先级以及两种特殊的线程和线程间的通讯等内容。

第 9 章:Java 多媒体技术。主要介绍了基本的图形、图像、声音和动画的处理等内容。

第 10 章:Java 数据库技术。主要介绍了 ODBC 和 JDBC 的相关知识、使用 JDBC 桥 JDBC-ODBC 桥和数据库相连、Java 对数据库的基本操作和定义性操作以及 JBuilder 数据库工具类的应用等内容。

第 11 章:Java 网络编程。主要介绍了 Socket 的基础知识、单线程服务器和客户程序、通用的多用户服务器和客户程序等内容。

第 12 章:高级应用和新技术。主要介绍了 JavaBean、J2EE 和 XML 等内容。

第 13 章:物料清单管理系统制作实例。主要介绍了物料管理系统的设计、制作方案及程序代码等内容。

三、本书的特点

本书在编写过程中始终贯彻面向对象的方法，将 Internet 编程、多媒体编程、MDI 等有机地结合起来，力图详细地介绍面向对象方法的概念及原则机制等。

本书内容丰富、知识涵盖面广、结构合理、且结合大量的实例和直观的图示来介绍 Java 程序设计的原理和方法。通过本书的学习，读者不但能掌握 Java 语言的理论知识，还能迅速学会 Java 程序设计的方法和技巧。

为帮助读者巩固所学的知识，本书的每章末都配有练习题，在书末还附有习题参考答案，以供读者参考。

四、本书的适用对象

本书不仅可以作为 Java 自学者和从事 Java 程序开发的技术人员的学习和参考用书，也可以作为各大中专院校相关专业和 Java 培训班的参考用书。

全书由汤庸主编，第 1 章至第 4 章由申建芳编写，第 5 章和第 6 章由肖炜编写，第 7 章至第 12 章由冯智圣编写，第 13 章由陈杰贺、黄俊健、冯智圣、高敬明、聂湘共同编写，其中李峥对第 11 章提供了程序的源代码。本书在编写过程中，参阅了大量文献资料和网上资料，同时得到了中山大学协同软件研究开发中心人员的大力支持，在此一并表示衷心的感谢！

由于作者水平有限，编写时间仓促，书中错漏之处在所难免，敬请广大读者批评指正。读者如有好的意见或建议可发 E-mail 至 service@cnbook.net，也可登陆网站 <http://www.cnbook.net>，在该网站的论坛进行探讨（本书所有源代码也可从该网站下载）。

编者

2003 年 9 月

目 录

第1章 Java 概述..... 1

1.1 Java 简介..... 1

1.1.1 Java 的特点..... 1

1.1.2 Java 的发展历史..... 2

1.1.3 Java 的应用..... 3

1.1.4 支持 Java 的 Web 浏览器..... 3

1.1.5 Java 和 JavaScript 区别..... 4

1.1.6 Java 与 C、C++ 的联系与区别..... 4

1.2 Java 编程环境的设置..... 6

1.2.1 JDK 应用简介..... 6

1.2.2 JBuilder 8.0 的环境配置..... 7

1.3 简单的 Java 程序编辑..... 11

1.3.1 在 JDK 环境下编辑 Java 小应用 程序 (Java Applet)..... 11

1.3.2 在 JBuilder 8.0 集成环境下 编辑 Java..... 14

1.4 Java 的程序结构..... 16

小结..... 17

综合练习一..... 17

一、填空题..... 17

二、判断题..... 18

三、简答题..... 18

四、实践题..... 18

第2章 Java 编程基础..... 19

2.1 Java 语法基础..... 19

2.1.1 变量..... 19

2.1.2 标识符..... 21

2.1.3 关键字..... 21

2.1.4 运算符..... 21

2.1.5 分隔符..... 27

2.1.6 表达式..... 27

2.1.7 包..... 28

2.2 数据类型..... 29

2.2.1 基本类型..... 29

2.2.2 扩展类型..... 30

2.3 基本语句和流程控制..... 30

2.3.1 基本语句..... 30

2.3.2 流程控制..... 31

2.4 类型转换..... 36

2.4.1 整型、浮点型、字符型变量间的 相互转换..... 37

2.4.2 Java 的包装类..... 37

2.4.3 简单类型变量和包装类之间的 相互转换..... 37

2.4.4 String 类和其他数据类型的 相互转换..... 38

2.4.5 将字符型变量转换为数值型变量..... 38

2.4.6 Date 类与其他数据类型的相互转换..... 38

小结..... 39

综合练习二..... 39

一、填空题..... 39

二、判断题..... 39

三、简答题..... 39

四、实践题..... 40

第3章 面向对象编程..... 41

3.1 面向对象的程序设计..... 41

3.1.1 面向对象的产生和发展..... 41

3.1.2 Java 中面向对象的特点..... 41

3.1.3 面向对象程序设计与结构化程序 设计的主要区别..... 42

3.2 常用的术语 C..... 42

3.2.1 对象..... 42

3.2.2 消息..... 42

3.2.3 类..... 43

3.2.4 继承..... 43

3.2.5 封装..... 44

3.2.6 多态..... 44

3.2.7 接口..... 45

3.3 类和继承	45	4.3.6 Java 调试器 (jdb)	60
3.3.1 类的声明和实例化	45	4.3.7 API 文件生成器 (javadoc)	61
3.3.2 类实体	46	4.4 JDBC	62
3.3.3 类中的数据成员	46	4.4.1 JDBC 示意图	62
3.3.4 类中的方法	46	4.4.2 JDBC 类的功能	63
3.3.5 类的继承	47	4.4.3 JDBC 调用过程及方法	63
3.3.6 创建新类	47	小结	63
3.4 接口和包	48	综合练习四	63
3.4.1 接口的定义	48	一、填空题	63
3.4.2 预定义包	48	二、判断题	63
3.5 常见的问题及解决的方法	51	三、简答题	64
3.5.1 编译器 (javac) 找不到类	51	四、实践题	64
3.5.2 解释器 (java) 不能找到其中 一个类	52	第 5 章 文件系统和流处理	65
3.5.3 程序不能工作的原因	52	5.1 文件系统	65
小结	52	5.1.1 File 类及其应用	65
综合练习三	53	5.1.2 RandomAccessFile 及其应用	68
一、填空题	53	5.1.3 FileInputStream 和 FileOutputStream	73
二、判断题	53	5.1.4 FileReader 和 FileWriter	79
三、简答题	53	5.2 流系统	79
四、实践题	53	5.2.1 流系统简介	79
第 4 章 Java 程序和开发环境	54	5.2.2 字节流	80
4.1 创建应用程序	54	5.2.3 字符流	87
4.1.1 编辑 Java 应用程序 (Application) 的源程序	54	5.2.4 其他流	88
4.1.2 编译源文件	54	小结	92
4.1.3 解释执行程序	55	综合练习五	93
4.2 创建小应用程序	55	一、填空题	93
4.2.1 Applet 的工作原理	55	二、判断题	93
4.2.2 插入 Java Applet 的格式与步骤	56	三、简答题	93
4.2.3 Applet 应用举例	56	四、实践题	93
4.2.4 Applet 源文件的分类	57	第 6 章 异常	94
4.3 Java 环境和工具	57	6.1 异常的概念	94
4.3.1 小应用程序浏览器 (appletviewer)	57	6.2 异常的分类	95
4.3.2 解释器 (java)	58	6.3 异常的处理	96
4.3.3 编译器 (javac)	59	6.3.1 Java 的异常处理机制	96
4.3.4 头文件生成器 (javah)	59	6.3.2 try-catch-finally 语句	98
4.3.5 反汇编器 (javap)	60	6.3.3 非运行时异常简介	99
		6.4 定义自己的异常类	105

小结.....	109	二、判断题.....	175
综合练习六.....	109	三、简答题.....	175
一、填空题.....	109	四、实践题.....	175
二、判断题.....	109		
三、简答题.....	109		
四、实践题.....	109		
第7章 Java 用户界面技术.....	110	第8章 Java 线程及多线程技术.....	176
7.1 基本 AWT 组件.....	110	8.1 线程概述.....	176
7.1.1 Button (按钮).....	110	8.1.1 线程简介.....	176
7.1.2 CheckBox (复选框) 和		8.1.2 线程的四种状态.....	176
CheckboxGroup (选择框组).....	115	8.1.3 创建线程.....	177
7.1.3 Scrollbar (滚动条).....	123	8.2 线程的临界区、信号量和锁.....	181
7.1.4 List (列表).....	128	8.3 线程的阻塞.....	182
7.2 基本 Swing 组件.....	133	8.4 线程的优先级.....	183
7.2.1 版面布局.....	133	8.5 两种特殊的线程.....	185
7.2.2 JPanel.....	136	8.5.1 主线程.....	185
7.2.3 JLabel.....	137	8.5.2 守护线程.....	186
7.2.4 JButton.....	138	8.6 线程间的通讯.....	187
7.2.5 JRadioButton 和 JButtonGroup.....	138	8.6.1 生产者和消费者简介.....	187
7.2.6 JToggleButton.....	140	8.6.2 Buffer.....	189
7.2.7 JTextField、JPasswordField 和		8.6.3 生产者和消费者的结合.....	190
JTextArea.....	141	小结.....	191
7.2.8 JList.....	142	综合练习八.....	191
7.2.9 JComboBox.....	147	一、填空题.....	191
7.2.10 JFileChooser.....	149	二、判断题.....	191
7.2.11 JScrollPane.....	152	三、简答题.....	191
7.2.12 JProgressBar.....	154	四、实践题.....	191
7.2.13 JTree.....	157		
7.2.14 JTable.....	160	第9章 Java 多媒体技术.....	192
7.3 事件处理.....	165	9.1 基本的图形处理.....	192
7.3.1 事件监听器简介.....	165	9.1.1 颜色处理.....	192
7.3.2 创建事件监听器.....	166	9.1.2 文本处理.....	195
7.3.3 监听器适配器类.....	170	9.1.3 画直线.....	198
7.3.4 把事件源连接到事件监听器.....	172	9.1.4 画矩形.....	198
7.3.5 从事件监听器断开事件源.....	172	9.1.5 画椭圆和圆.....	201
小结.....	175	9.1.6 画弧形.....	203
综合练习七.....	175	9.1.7 画多边形.....	205
一、填空题.....	175	9.2 基本的图像处理.....	208
		9.3 基本的声音处理.....	212
		9.4 基本的动画处理.....	217
		小结.....	223

综合练习九.....	223	11.2 单线程的服务器 (Server) 和客户 程序 (Client)	276
一、填空题.....	223	11.2.1 单线程服务器.....	276
二、判断题.....	223	11.2.2 单线程客户程序	278
三、简答题.....	223	11.2.3 执行结果.....	279
四、实践题.....	223	11.3 通用的多用户服务器和客户程序	280
第 10 章 Java 数据库技术.....	224	11.3.1 通用多用户服务器	280
10.1 ODBC 与 JDBC 简介.....	224	11.3.2 通用多用户客户程序	297
10.1.1 体系结构.....	224	11.3.3 公共参数包.....	309
10.1.2 运行环境.....	225	11.3.4 执行过程.....	310
10.2 使用 JDBC 桥、JDBC-ODBC 桥和 数据库相连	226	小结.....	311
10.2.1 与 Microsoft Access 的连接.....	226	综合练习十一	311
10.2.2 与 SQL Server 2000 的连接.....	229	一、填空题.....	311
10.2.3 与 MySQL 的连接.....	234	二、判断题.....	311
10.2.4 与 Oracle 的连接.....	239	三、简答题.....	311
10.3 Java 对数据库的基本操作	241	四、实践题.....	311
10.3.1 插入数据.....	241	第 12 章 高级应用和新技术.....	312
10.3.2 对数据库数据的查询、修改、 删除.....	243	12.1 JavaBean	312
10.3.3 程序说明.....	251	12.1.1 JavaBean 简介	312
10.3.4 发送 SQL 语句.....	252	12.1.2 JavaBean 的应用.....	312
10.3.5 ResultSet	257	12.2 J2EE	317
10.4 Java 对数据库的定义性操作	259	12.2.1 J2EE 简介	317
10.4.1 创建和删除数据表	260	12.2.2 J2EE 的应用.....	319
10.4.2 创建和删除视图	263	12.3 XML.....	328
10.5 JBuilder 数据库工具类的应用.....	267	12.3.1 XML 的简介	328
小结.....	272	12.3.2 XML 的应用	329
综合练习十.....	272	小结.....	330
一、填空题.....	272	综合练习十二	330
二、判断题.....	273	一、填空题.....	330
三、简答题.....	273	二、判断题.....	330
四、实践题.....	273	三、简答题.....	330
第 11 章 Java 网络编程.....	274	四、实践题.....	330
11.1 Socket 简介.....	274	第 13 章 物料清单管理系统制作实例.....	331
11.1.1 Socket 的概念.....	274	13.1 系统设计	331
11.1.2 基本的通信流程	275	13.1.1 制造标准管理业务分析	331
11.1.3 Java 对通信流程的封装.....	275	13.1.2 制造标准管理业务数据流程图.....	331
		13.1.3 实体关系设计 (E-R 关系图)	331

13.1.4 模块功能设计	332
13.2 制作方案	332
13.2.1 系统功能	332
13.2.2 专业术语	332
13.2.3 具体操作	333
13.3 程序代码	338
13.3.1 Bom 包的文件	338
13.3.2 Dababase 包的文件	356
13.3.3 XML 文件	360
13.3.4 JSP 文件	364
13.3.5 HTML 文件	368
小结	369
综合练习十三	369
一、填空题	369
二、判断题	369
三、简答题	369
四、实践题	369

参考答案	370
-------------------	------------

第 1 章	370
第 2 章	370
第 3 章	371
第 4 章	371
第 5 章	372
第 6 章	374
第 7 章	376
第 8 章	377
第 9 章	378
第 10 章	380
第 11 章	382
第 12 章	383
第 13 章	384

参考文献	386
-------------------	------------

第 1 章 Java 概述

Java 是 Sun Microsystems 公司在 1995 年发布的一种面向对象的、多线程的解释型编程语言，它是网络时代应用最广泛的开发工具。本章从 Java 的特点、历史发展、应用领域及最新版本等入手，向大家介绍其强大的功能。

本章的主要内容有：

- (1) Java 简介。
- (2) Java 编程环境的设置。
- (3) 简单的 Java 程序编辑。
- (4) Java 的程序结构。

1.1 Java 简介

Java 最初用于在网页上创建简单的程序，是一种简单的面向对象的分布式的解释型的健壮的安全的多线程语言。

1.1.1 Java 的特点

1. 简单

Java 是一种简单明了的编程语言。其简单性主要体现在以下三个方面：

(1) Java 的风格类似于 C++，因此 C++ 程序员会感到非常熟悉。从某种意义上讲，Java 语言是 C 及 C++ 语言的一个变种，因此，C++ 程序员可以很快就掌握 Java 编程技术。

(2) 当程序运行的时候，Java 会检查并自动回收那些不再使用的内存，程序员不必清楚内存指针以及如何手动分配内存。这一重要手段使得程序崩溃的几率大大降低，而且内存也不可能被故意地滥用、误用，因此 Java 摒弃了 C++ 中容易引发程序错误的地方，如指针和内存管理。

(3) Java 提供了丰富的类库

由于 Java 开发工具 (JDK) 上绑定了丰富的包，这些包类似于 C++ 中的类库，用户不需要开发已经提供的功能，并且这些包中的源代码适用于任何应用程序，因此用户可快速地编写和运行 Java 程序。

2. 面向对象

面向对象可以说是 Java 最重要的特性。Java 语言的设计完全是面向对象的，和其他面向对象的编程语言有很大程度的不同，Java 重在创建可以被程序访问和操作的对象（数据结构或行为），Java 支持静态和动态风格的代码继承及重用，同时具有适用于分布式计算环境的特性。

3. 分布式

Java 包括一个支持 HTTP 和 FTP 等基于 TCP/IP 协议的子库。因此，Java 应用程序可凭借 URL 打开并访问网络上的对象，其访问方式与访问本地文件系统几乎完全相同，非常适合分布环境的应用。

4. 健壮性

Java 检查程序编译和运行时的错误。主要是针对内存和指针方面的错误。类型检查帮助检查出许多开发早期出现的错误。Java 自己操纵内存减少了内存出错的可能性。Java 还实现了真数组, 避免了覆盖数据的可能。这些功能特征大大缩短了开发 Java 应用程序的周期。

5. 平台无关性

Java 作为网络的一个整体, Java 将它的程序编译成一种结构中立的中间文件格式。只要有 Java 运行系统的机器都能执行这种中间代码, 即可以在大量不同的设备和计算平台上使用, 所以它设计为与平台无关的。Java 源程序被编译成一种高层次的机器无关的字节码, 这种语言被设计在虚拟机上运行, 由机器相关的运行调试器实现执行。

6. 安全性

Java 的安全性可从两个方面得到保证。一方面, 在 Java 语言里, 像指针和释放内存等 C++ 功能被删除, 避免了非法内存操作; 另一方面, 当 Java 用来创建浏览器时, 语言功能和浏览器本身提供的功能结合起来, 使它更安全。Java 语言在机器上执行前, 要经过很多次的测试。它经过代码校验, 检查代码段的格式, 检测指针操作、对象操作是否过分以及试图改变一个对象的类型。

7. 解释型

Java 虚拟机又称为解释器 (运行系统) 能直接运行目标代码指令。链接程序通常比编译程序所需资源少, 然而, 解释程序有时确实会导致一些严重的性能问题, 与由编译语言生成的针对平台的机器码相比, Java 字节码的执行速度要慢一些。

8. 多线程

对多线程最基本的理解就是同时对多个任务加以控制, 线程是一个进程里分出来的独立的小进程。在操作系统中引入进程的目的是增加程序的并发性, 从此充分地利用系统资源。Java 多线程功能使得在一个程序里可同时执行多个小任务。因为 Java 实现的多线程技术, 所以比其他语言更健壮、高效。多线程带来的更大的好处是更好的交互性能和实时控制性能, 同时在开发难易程度和性能上都比单线程要好。

1.1.2 Java 的发展历史

Java 是一种编程语言, 它适合于设计与 Internet 有关的软件, 同时也是一种面向对象的编程语言, 具有面向对象的封装、多态和继承等特性。另外, 它还是一种跨平台的语言, 可以在多种操作系统上运行。

Java 的前身是 Oak 语言, 是一种用于网络的精巧而安全的语言, 它保留了熟悉的 C++ 语法, 但避免了由于明确的资源引用、指针算法、操作符重载等带来的系统潜在的危险。Oak 将内存管理直接加入到语言中, 程序员只需要专心于所要完成的任务即可。当 Oak 逐渐成熟时也是万维网急剧增长的时期, 因此特别适合于 Internet 应用程序开发。“连 Internet, 用 Java 编程”, 成为技术人员的一种时尚也成为软件开发的一种革命性的技术。1996 年, 许多大型的软件和硬件公司都从 SUN 取得了 Java 的许可, 如 Adobe、IBM、Borland、Oracle, 这些公司和其他的一些 Java 的授权的公司都将 Java 加入到其桌面产品、操作系统以及其他产品中。

在今天,最容易找到 Java 的地方是 WWW, Web 是 Java 的方便的传输机制, Web 的无所不在使 Java 作为 Internet 工具闻名于世。目前较为常用的 Java 的集成开发环境有多种,如 JBuilder、Visual J++、VisualAge for Java 等,其中 JBuilder 8.0,由 Borland 软件公司(Nasdaq 纳思达克股票代码为 BORL)于 2002 年 6 月 21 日在北京发布。

Borland JBuilder 是用于构建企业级 Java 应用程序的跨平台开发环境,使用双向可视化设计器加速 EJBTM、Web、XML 和数据库应用程序的开发,并快速分发到领先的 J2EE 平台应用服务器,包括 BEA WebLogic、IBM WebSphere、iPlanet Application Server、Oracle9i Application Server 和集成的 Borland Enterprise Server。

JBuilder 8.0 采用最新 Java 标准,缩短了开发周期。JBuilder 8.0 将 UML 代码直观化、单元测试,以及对多个平台和版本控制系统的支持与改进后的重构技术组合到一起。由于有工具栏可以访问 Optimizeit Suite 和 TeamSource DSP, JBuilder 8.0 将应用开发生命周期的主要内容与开发环境无缝连接到了一起。JBuilder 8.0 内含 Web 服务功能,并且符合当前的移动应用程序标准。JBuilder 8.0 可以满足客户两个主要的业务目标:提供最高水准的服务和技术并降低开发成本,并将 JBuilder 8.0 集成到开发过程中,使开发组织更高效而且更快地建立应用程序。

JBuilder 8.0 的三个版本分别为个人版、专业版和企业版。JBuilder Personal (个人版)是用于那些想学习 Java 开发的学生以及建立个人的、非商业用途应用者的 Java 开发环境。对于那些希望使 Java 开发更为简单容易的开发者, JBuilder 8.0 SE (专业版)是最好的解决方案。JBuilder SE 包括节约时间工具如重构、高级调试、JDK 切换、Javadoc 工具和团队开发等。JBuilder Enterprise (企业版)使用完整的开发解决方案将 Java 开发带到更高的阶段,这个完整的解决方案包括设计、编码、分发及数据库应用的企业级 Java 应用软件。

1.1.3 Java 的应用

Java 语言有着广泛的应用前景,大体上可以从以下几个方面来考虑其应用:

- (1) 所有面向对象的应用开发,包括面向对象的事件描述、处理、综合等。
- (2) 计算过程的可视化、可操作化的软件的开发。
- (3) 动态画面的设计,包括图形图像的调用。
- (4) 交互操作的设计(选择交互、定向交互、控制流程等)。
- (5) Internet 的系统管理功能模块的设计,包括 Web 页面的动态设计、管理和交互操作设计等。
- (6) Intranet (企业内部网)上的软件开发(直接面向企业内部用户的软件)。
- (7) 与各类数据库连接查询的 SQL 语句实现。
- (8) 其他应用类型的程序。

总之,Java 是最流行的网络编程语言,主要用于开发网络应用程序和 Web 浏览。

1.1.4 支持 Java 的 Web 浏览器

Web 浏览器除了可以作为 WWW 浏览工具外,还可以作为开发人员调试小程序的运行环境。目前,几种主要的浏览器 HotJava、Internet Explorer 和 Netscape Navigator 等都具有 Java 浏览功能。

由 Sun 公司开发的 HotJava 是第一个支持 Java 的浏览器。

Navigator 是 Netscape 通信公司的起家产品。目前，它占据了浏览器的大份额市场。

Microsoft 的 Web 浏览器是 Internet Explorer，目前的最新版本也是 6.0。作为浏览器中的后起之秀，Explorer 正在逐渐占据更多的市场份额，动摇 Netscape 的主导地位。Explorer 最大的优点是它是 Windows 系列操作系统内嵌的部分。

1.1.5 Java 和 JavaScript 区别

二者的区别体现在：

(1) 它们是分别由两个公司开发的不同的两个产品，Java 是 SUN 公司推出的新一代面向对象的程序设计语言，特别适合于 Internet 应用程序开发；而 JavaScript 是 Netscape 公司的产品，其目的是为了扩展 Netscape Navigator 功能，而开发的一种可以嵌入 Web 页面中的基于对象和事件驱动的解释性语言。

(2) JavaScript 是基于对象的，而 Java 是面向对象的，即 Java 是一种真正的面向对象的语言，即使是开发简单的程序，都必须设计对象。JavaScript 是种脚本语言，它可以用来制作与网络无关的，与用户交互作用的复杂软件，它是一种基于对象和事件驱动的编程语言，因此它本身提供了非常丰富的内部对象供设计人员使用。

(3) 两种语言在其浏览器中所执行的方式不一样。Java 的源代码必须经过编译再传递到客户端执行，客户端上必须具有相应平台上的仿真器或解释器，它可以通过编译器或解释器实现独立于某个特定的平台编译代码的束缚。JavaScript 是一种解释性编程语言，其源代码在发往客户端执行之前不需经过编译，而是将文本格式的字符代码发送给客户，由浏览器解释执行。

(4) 两种语言所采取的变量是不一样的。Java 采用强类型变量检查，即所有变量在编译之前必须作声明。JavaScript 中变量声明，采用弱类型，即变量在使用前不需作声明，而是解释器在运行时检查其数据类型。

(5) 代码格式不一样。Java 是一种与 HTML 无关的格式，必须通过像 HTML 中引用外媒体那么进行装载，其代码以字节代码的形式保存在独立的文档中。JavaScript 的代码是一种文本字符格式，可以直接嵌入 HTML 文档中，并且可动态装载。编写 HTML 文档就像编辑文本文件一样方便。

(6) 嵌入方式不一样。在 HTML 文档中，两种编程语言的标识不同，JavaScript 使用 `<script>...</script>` 来标识，而 Java 使用 `<applet> ... </applet>` 来标识。

(7) 静态绑定和动态绑定。Java 采用静态绑定，即 Java 的对象引用必须在编译时的进行，以使编译器能够实现强类型检查。JavaScript 采用动态绑定，即 JavaScript 的对象引用在运行时进行检查，如不经编译就无法实现对象引用的检查。

1.1.6 Java 与 C、C++ 的联系与区别

C 语言是微软用来和 Java 抗衡的武器，对于变量声明、参数传递、操作符、流控制等，Java 使用了和 C、C++ 相同的传统，因此二者在很大程度上有着惊人的相似。同时，两者不同的地方也很多，Java 摒弃了 C 和 C++ 中许多不合理的内容，下面对他们的相同点和不同点分别列举如下：

1. 二者之间的相同点

(1) 语法定义相同。Java 语言基本上是从 C 语言上发展过来的。Java 的大部分语法都是从 C 或 C++中继承过来的。

(2) 变量的命名与规则相同。

(3) 表达式, 程序流程的控制语句基本相同。

(4) 都具有接口(interface)的概念。

2. 二者之间的不同点

1) 全局变量

Java 程序中, 不能在所有类之外定义全局变量, 只能通过在一个类中定义公用、静态的变量来实现一个全局变量。例如:

```
ClassGlobalVar{public static global_var;}
```

在类 GlobalVar 中定义变量 global_var 为 public static, 使得其他类可以访问和修改该变量。Java 对全局变量进行了更好的封装。而在 C 和 C++中, 依赖于不加封装的全局变量常常造成系统的崩溃。

2) Goto

Java 不支持 C、C++中的 goto 语句, 而是通过例外处理语句 try, Catch, final 等来代替 C、C++中用 goto 来处理遇到错误时跳转的情况, 使程序更可读且更结构化。

3) 指针

指针是 C、C++中最灵活, 也是最容易产生错误的数据类型。由指针所进行的内存地址操作常会造成不可预知的错误, 同时通过指针对某个内存地址进行显式类型转换后, 可以访问一个 C++中的私有成员, 从而破坏安全性, 造成系统的崩溃。而 Java 对指针进行完全的控制, 程序员不能直接进行任何指针操作, 例如把整数转化为指针, 或者通过指针释放某一内存地址等。同时, 数组作为类在 Java 中实现, 良好地解决了数组访问越界问题, 这在 C、C++中是不作检查的错误。

4) 内存管理

在 C 中, 程序员通过库函数 malloc()和 free()来分配和释放内存, C++中则通过运算符 new 和 delete 来分配和释放内存。再次释放已释放的内存块或未被分配的内存块, 会造成系统的崩溃; 同样, 忘记释放不再使用的内存块也会逐渐耗尽系统资源。而在 Java 中, 所有的数据结构都是对象, 通过运算符 new 为它们分配内存堆。通过 new 得到对象的处理权, 而实际分配给对象的内存可能随程序运行而改变, Java 对此自动地进行管理并且进行垃圾收集, 有效防止了由于程序员的误操作而导致的错误, 并且更好地利用了系统资源。

5) 数据类型的支持

在 C、C++中, 对于不同的平台, 编译器对于简单数据类型如 int, float 等分别分配不同长度的字节数, 例如: int 在 IBMPC 中为 16 位, 在 VAX-11 中为 32 位, 这导致了代码的不可移植性, 但在 Java 中, 对于这些数据类型总是分配固定长度的位数, 如 int 型, 它总占 32 位, 这就保证了 Java 的平台无关性。

6) 类型转换

在 C、C++中, 可以通过指针进行任意的类型转换, 常常带来不安全性, 而 Java 中, 运行时系统对对象的处理要进行类型相容性检查, 以防止不安全的转换。

7) 头文件

C、C++中用头文件来声明类的原型以及全局变量、库函数等，在大的系统中，维护这些头文件是很困难的。而 Java 不支持头文件，类成员的类型和访问权限都封装在一个类中，运行时系统对访问进行控制，防止对私有成员的操作。同时，Java 中用 import 语句来与其他类进行通讯，以便使用它们的方法。

8) 结构和联合

C、C++中的结构和联合中所有成员均为公有，这就带来了安全性问题。Java 中不包含结构和联合，所有的内容都封装在类中。

9) 预处理

C、C++中用宏定义来实现的代码给程序的可读性带来了困难。在 Java 中，不支持宏，它通过关键字 final 来声明一个常量，以实现宏定义中广泛使用的常量定义。

1.2 Java 编程环境的设置

如果用户安装了 Microsoft Windows 系列的操作系统，可能已经安装了可以运行 Java 程序的 Java 解释程序，因为这个解释程序是 Web 浏览器的一部分，而且仅仅能运行小程序。为开发 Java 应用程序，不仅仅需要解释器，而且需要编译器和一些其他的工具用于创建、运行和调试程序。

Java 语言有两个开发环境，一个是免费的 JDK，是命令行方式的。还有一个是 JBuilder、Visual J++ 等高级语言，是开发 Java 程序的集成环境。

1.2.1 JDK 应用简介

Java 开发工具 (JDK) 是许多 Java 专家最初使用的开发环境。尽管许多编程人员已经使用第三方的开发工具，但 JDK 仍被当作 Java 开发的重要工具，目前流行的版本是 JDK 1.4。Java 的源文件的扩展名为 .java，是一种与平台无关的文档格式，并且兼容性好，不管是在 Windows 操作系统还是 Unix 操作系统都可运行，并且在 IE 和 Netscape 之间也是兼容的。Java 的源文件可以用文本编辑器或字处理软件编辑，要保存为文本文件。

JDK 由一个标准类库和一组建立测试及建立文档的 Java 实用程序组成。其核心 Java API 是一些预定义的类库，开发人员需要用这些类来访问 Java 语言的功能。Java API 包括一些重要的语言结构以及基本图形，网络 and 文件 I/O。一般来说，Java API 的非 I/O 部分对于运行 Java 的所有平台是相同的，而 I/O 部分则仅在通用 Java 环境中实现。作为 JDK 实用程序，工具库中有七种主要程序。

(1) javac: Java 编译器，将 Java 源代码转换成字节码。

(2) java: Java 解释器，直接从类文件执行 Java 应用程序字节代码。

(3) appletviewer: 小应用程序浏览器，一种执行 HTML 文件上的 Java 小程序的 Java 浏览器。

(4) javadoc: 根据 Java 源码及说明语句生成 HTML 文档。

(5) jdb: Java 调试器，可以逐行执行程序，设置断点和检查变量。

(6) javah: 产生可以调用 Java 过程的 C 过程，或建立能被 Java 程序调用的 C 过程的头文件。