

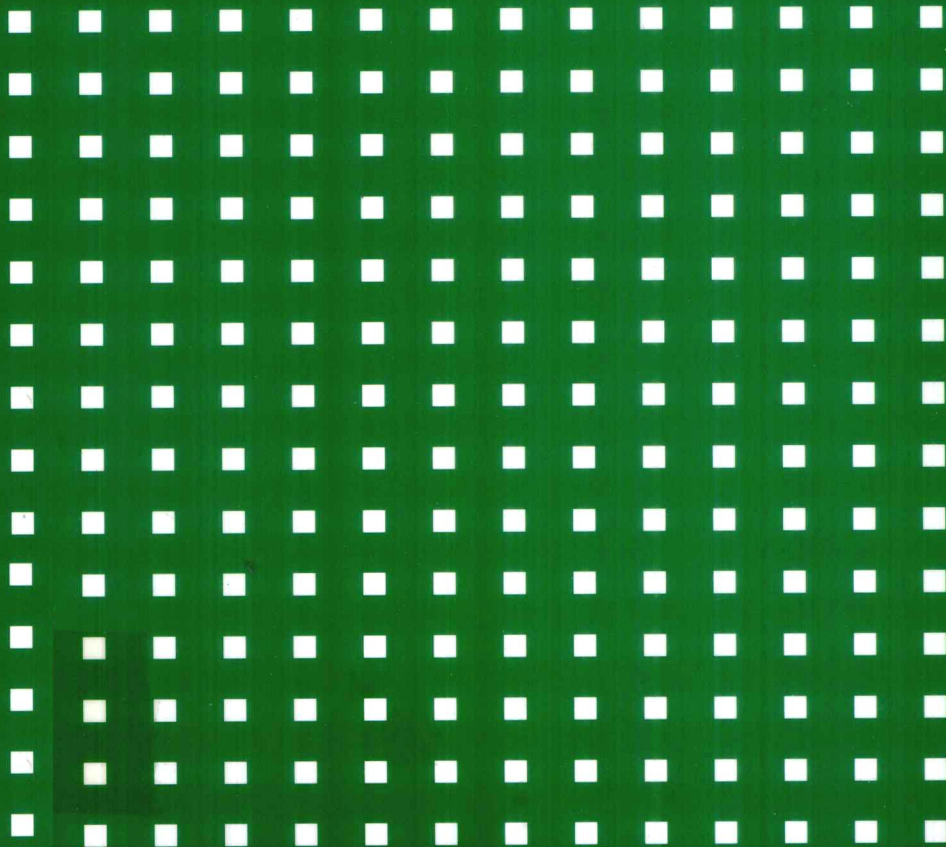
高等学校计算机类规划教材

宁波市高校特色教材

- Java入门与实际动手能力并重
- 强调系统开发中结构和模式的应用
- 遵循“提出问题—知识学习—案例实现”的体例，注重实用性
- 免费提供配套PPT电子课件及习题答案

Java语言实用教程

杨晓燕 主编



清华大学出版社

Java语言实用教程

杨晓燕 主编

清华大学出版社
北京

内 容 简 介

Java 是纯面向对象的程序设计语言,具有完全面向对象、简单高效、与平台无关、安全、支持多线程等特点,目前非常流行,且发展前景广阔。本书遵循“案例分析—知识学习—案例实现”和“每章案例—综合案例”的编写思路,以 Java 入门和架构应用并重为原则进行编写,内容主要包括 Java 的渊源和特点、Java 开发工具入门、Java 的输入/输出、Java 流程控制、类及类的构造、共享包的定义和引用、多态的实现和管理、异常处理、GUI 图形界面、图形处理及 Java 数据库连接等。

本书既可作为大中专院校 Java 程序设计课程的教材,也可作为 Java 自学者的入门用书。

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 13701121933

图书在版编目(CIP)数据

Java 语言实用教程/杨晓燕主编. —北京:清华大学出版社,2012.2

ISBN 978-7-302-27699-9

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

中国版本图书馆 CIP 数据核字(2011)第 268170 号

责任编辑:朱英彪

封面设计:傅瑞学

版式设计:文森时代

责任校对:张彩凤

责任印制:杨 艳

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者:北京鑫海金澳胶印有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:22 字 数:505 千字

版 次:2012 年 2 月第 1 版 印 次:2012 年 2 月第 1 次印刷

印 数:1~5000

定 价:36.00 元

产品编号:038458-01

特色教材序言

宁波市为了推进自主创新,建设创新型城市,在 2005 年提出了构建“服务型教育体系”的重要决策,在市委、市政府的大力推动下,在各级教育机构的努力实践下,全市的教育服务理念不断强化,教育与经济社会的结合更为密切,教育对经济发展的服务、贡献能力日益提高。作为中国“智慧城市建设”的先驱,宁波市非常重视各类人才的引进与培养,通过“人才公寓”和“人才驿站”计划,引进了大批急需的大学毕业生、研究生以及高水平海归人才;教育主管部门设立专项资金,以立项的方式对在甬高校人才培养模式改革、服务型重点专业建设给予大力扶持,主要包括与地方经济发展相适应的高素质应用型人才培养基地,符合地方支柱和主导产业发展需要的服务型教育重点专业,引进和开发面向地方应用型人才培养、培训的特色教材,引进和培育国内外高端培训机构等。本套教材即是 2009 年宁波市首批特色教材建设项目。

浙江万里学院自 2006 年开始实施“以合作式学习为特征的研究性教学方法改革”,从 8 门课程的“点”到 4 个专业作为实验区的“线”(计算机专业是实验区之一),直至目前有 400 余门课程实践的“面”,取得了丰硕的改革实践成果,学生的创新思维、自主学习能力和综合素养普遍有了大幅度提升,教师们的改革热情、教学艺术和研究水平不断提高。“研究性教学”不仅在本校落地生根,开花结果,而且得到了省内外专家、同行的充分认可,每年有近百所高校来校交流取经,浙江省教育厅分别在 2008 年和 2011 年两次组织全省高校领导与教师来校召开现场会。因改革成效显著,浙江万里学院计算机科学与技术专业在 2011 年被教育部委托浙江省列为“计算机类专业培养服务外包人才试点”专业。本套教材由该专业教师承担编写任务,有以下特点:

(1) 理念先进

本教材建立在 IPR-CDIO (Interest、Perseverance、Responsibility- Conceive、Design、Implement、Operate) 工程教育理念的基础上,力求将学生兴趣激发、责任与毅力培养融入项目化教学过程中,以利于学生的综合素质、实践能力和创新意识的培养。

(2) 特色鲜明

作为 IT 类软件系列主干课程教材,本套教材内容紧紧围绕学生的软件核心能力培养编写,通过鲜活的案例或实用的工程项目引导,配以具有思考与探究意义的小课题,促进学生小组共同学习、训练与探究,充分体现了“研究性教学”的特色。根据课程特点与性质,各教材在体例上各具风格。

(3) 实践性强

本教材从培养高素质应用型人才的需求出发,兼顾了社会培训要求以及自学者的兴趣。教材知识结构完整,实验项目由浅入深,工程项目由单一到完整,学习者只需要参考、模拟,即可得到系统训练,学以致用,稍有改进与创新,即可收到意想不到的效果,大大增

强学习者的成就感。

本批教材包括《数据结构与算法》、《数据库技术及应用》、《Java 语言实用教程》和《实用计算机图形学》，属于 IT 类软件基础课程的配套教材。每本教材都已在学校改革创新实验区中历经了几轮的试用实践，是作者们多年教学经验的展示，所有程序代码均经过测试，读者从模拟开始，可逐渐达到熟练应用的目标。

本批教材得到了宁波市教育局的大力支持，除了给予经费资助，还组织教育界与出版社联合专家组进行评审，得到了评审专家的高度赞同，在此一并表示衷心的感谢！

真诚期望广大读者提出宝贵意见或建议，以促进本套教材不断优化完善。



2012 年 1 月

前 言

Java 语言诞生于 1995 年,是目前最为流行的面向对象程序设计语言,它具有简单高效、跨平台、安全、支持多线程的特点。面向对象技术具有模拟现实世界的思维方式,其数据与操作相捆绑的程序风格符合现代大规模软件开发的要求,是目前计算机应用开发领域的主流趋势。不仅如此,Java 的跨平台性造就了它在 Internet 上无可比拟的应用前景,使其成为当今 Internet 上最流行、最受欢迎的一种程序开发语言。原 Sun 公司总裁兼首席运营官 Jonathan Schwartz 说:“Java 技术正在成为全球网络应用的事实标准,它将大大加快和简化提供移动、消费和企业市场的服务。”现在,Java 平台仍在继续为 Java 经济注入活力,并驱动着全球企业在桌面系统和服务器领域进行着大量的技术创新。

当前,许多高校都开设了 Java 程序设计课程,并将其作为高校计算机基础教学的骨干课程之一。在教育部计算机基础课程教学指导委员会制定的白皮书中,Java 课程也被列为核心课程之一。

本书具有以下特色:

- ☑ 每章都由“案例分析”开篇,“案例实现”收尾,案例大小适中,能够使读者快速入门。
- ☑ 本书遵循“案例提出问题—知识学习—案例实现”和“章节案例—综合案例”的编写思路,为便于理解,又把综合案例分解为可独立运行的子案例。
- ☑ 课后练习分为习题和问题探究,习题是积木式的内容重建,问题探究则在知识广度和深度上做了进一步拓展。习题中的 SCJP 题为学生打开一扇认证之窗。
- ☑ 四化设计,即核心知识案例化、抽象概念形象化、复杂问题分解化和综合知识项目化。
- ☑ 为了便于教师讲解和理解,为主要程序代码增加了行号。
- ☑ 本书不仅注重一般概念和理论的叙述,而且注重系统开发过程中结构和模式的研究。
- ☑ 本书编写的初衷是重在应用。每章后面的习题、问题探究及 SCJP 题提供有参考答案。

本书由杨晓燕主编,参与编写的老师还有姜遇姬教授、张梁斌和邓芳等。在本书顺利完成之际,特别感谢宁波市教育局在宁波市高校特色教材项目的大力支持,感谢参与编写教材的长辈和同事,特别是尊敬的已故教授姜遇姬老师的耐心、认真的指导及高水平的编写,也感谢我的同事刘臻。在此一并感谢我的学生邱臻豪和陈潇俊提供的支持,感谢清华大学出版社朱英彪编辑的热心付出和敬业指导。

本书 PPT、习题答案参考及实验安排资料可通过编者邮箱 yangxy3225@163.com 索要,也可以在清华大学出版社的网站下载,或向朱英彪编辑(邮箱:zhuyingbiao@126.com)

索要。

由于编者水平所限，书中难免存在一些疏漏和不足之处，希望读者批评指正。联系方式为：yangxy3225@163.com。

编 者

目 录

第 1 章 Java 概述	1
1.1 Java 的崛起	2
1.2 Java 与 C、C++	3
1.2.1 Java 与 C++	3
1.2.2 Java 与 C	3
1.3 Java 语言的特点	5
1.4 Java 程序的类型及其不同的编程模式	6
1.5 Java 程序开发过程	8
1.5.1 开发过程简介	8
1.5.2 创建 Java Application 程序示例	8
1.5.3 创建 Java Applet 程序示例	11
1.5.4 良好的编程习惯	13
1.6 Java 编程环境与开发工具	14
1.6.1 JDK 的下载与安装	14
1.6.2 环境变量介绍和配置	17
1.6.3 JDK 开发工具简介	19
1.6.4 Java 程序开发步骤	19
1.7 本章小结	20
习题	21
问题探究	21
第 2 章 Java 语言基础	23
2.1 标识符、关键字和分隔符	23
2.1.1 标识符和关键字	23
2.1.2 分隔符	24
2.2 数据类型	25
2.2.1 基本数据类型	26
2.2.2 变量	28
2.3 运算符与表达式	29
2.3.1 算术运算符	29
2.3.2 赋值运算符	30
2.3.3 关系运算符	31
2.3.4 逻辑运算符	32

2.3.5 条件运算符.....	32
2.3.6 其他运算符.....	33
2.3.7 运算符的优先级.....	33
2.4 字符串.....	34
2.4.1 创建 String 对象.....	34
2.4.2 创建 StringBuffer 对象.....	35
2.5 案例实现.....	37
2.6 本章小结.....	37
习题.....	38
问题探究.....	38
第 3 章 Java 输入/输出.....	40
3.1 标准输入/输出方法.....	40
3.1.1 标准输出方法.....	41
3.1.2 Scanner 键盘输入类.....	42
3.1.3 read()方法的使用.....	44
3.2 命令行参数输入法的应用.....	45
3.3 流式交互输入/输出的应用.....	46
3.3.1 应用示例.....	46
3.3.2 Java I/O 基本模型.....	48
3.4 文件数据的读/写.....	50
3.5 JOptionPane 对话框输入法.....	53
3.6 案例实现.....	54
3.7 本章小结.....	56
习题.....	57
问题探究.....	57
第 4 章 程序流程控制结构和方法.....	58
4.1 语句和程序流程控制结构.....	58
4.2 选择结构 (selection structure).....	59
4.2.1 选择语句 (selection statements).....	59
4.2.2 多选择结构 switch 语句.....	63
4.3 循环结构 (loop structure).....	66
4.3.1 3 种循环语句.....	67
4.3.2 循环程序结构小结.....	71
4.3.3 循环嵌套和 continue、break 语句.....	72
4.4 算法设计.....	75
4.4.1 迭代算法.....	76
4.4.2 穷举算法.....	76

4.4.3 复杂程序解决方案和方法.....	78
4.5 案例实现.....	84
4.6 本章小结.....	85
习题.....	86
问题探究.....	87
第5章 数组.....	88
5.1 数组的基本概念.....	88
5.2 一维数组.....	89
5.2.1 一维数组的声明.....	89
5.2.2 一维数组的内存申请.....	89
5.2.3 一维数组的初始化.....	90
5.2.4 测定数组的长度.....	92
5.2.5 foreach 语句与数组.....	93
5.3 二维数组.....	93
5.3.1 二维数组简介.....	93
5.3.2 二维数组的声明与创建.....	94
5.3.3 二维数组元素的初始化.....	95
5.3.4 二维数组的引用.....	96
5.4 案例实现.....	98
5.5 本章小结.....	101
习题.....	101
问题探究.....	102
第6章 Java 类和对象.....	104
6.1 面向对象编程.....	104
6.2 类的描述.....	107
6.2.1 类的定义.....	108
6.2.2 成员变量的访问控制符.....	109
6.2.3 成员方法.....	111
6.2.4 成员变量和局部变量.....	112
6.2.5 final 变量.....	113
6.3 对象的创建与使用.....	114
6.3.1 对象的创建.....	114
6.3.2 对象的比较.....	116
6.3.3 对象的使用.....	117
6.3.4 释放对象.....	121
6.3.5 Java 变量内存分配.....	121
6.3.6 匿名对象.....	122

6.4	构造方法.....	123
6.4.1	构造方法的作用和定义.....	123
6.4.2	this 引用.....	125
6.5	static 变量及 static 方法.....	128
6.5.1	static 变量.....	128
6.5.2	static 方法.....	130
6.6	对象初始化过程.....	132
6.7	成员方法.....	135
6.7.1	方法调用与参数传递方式.....	135
6.7.2	方法重载.....	138
6.7.3	final 最终方法和 abstract 方法.....	140
6.8	类的继承.....	141
6.8.1	继承的概念.....	141
6.8.2	创建子类.....	142
6.8.3	关于父类的构造方法.....	143
6.8.4	成员变量的隐藏和成员方法的重构.....	146
6.8.5	类与类之间的数据类型转换.....	147
6.8.6	抽象类.....	148
6.9	实用工具类.....	150
6.9.1	向量类 Vector.....	151
6.9.2	日期类 Date、Calendar 与 DateFormat.....	152
6.10	案例实现.....	154
6.11	本章小结.....	158
	习题.....	158
	问题探究.....	160
第 7 章	Java 接口、包和异常.....	164
7.1	接口 (interface).....	164
7.1.1	接口概述.....	165
7.1.2	接口的定义.....	165
7.1.3	实现接口的类定义.....	166
7.1.4	接口的多态性.....	169
7.2	包 (package).....	171
7.2.1	创建包.....	171
7.2.2	类的包外引用.....	173
7.2.3	jar 命令打包与引用.....	175
7.3	异常处理.....	178
7.3.1	异常的基本概念.....	178

7.3.2 异常处理机制.....	182
7.3.3 自定义异常类.....	186
7.3.4 GUI 应用程序的异常处理.....	187
7.4 案例实现.....	189
7.5 本章小结.....	192
习题.....	192
问题探究.....	192
第 8 章 图形处理	193
8.1 Java 坐标系 (coordinate system)	193
8.2 图形的颜色控制.....	194
8.3 Graphics 类的基本图形	197
8.3.1 绘制直线和矩形.....	197
8.3.2 绘制圆弧.....	200
8.3.3 绘制多边形.....	201
8.4 文字输出.....	202
8.5 绘图模式控制.....	204
8.6 案例实现.....	205
8.7 本章小结.....	206
习题.....	207
问题探究.....	207
第 9 章 图形用户界面	208
9.1 图形用户界面概述.....	209
9.1.1 图形用户界面路线.....	209
9.1.2 组件分类.....	209
9.1.3 常用容器类的应用.....	211
9.2 事件处理.....	215
9.2.1 基本概念.....	215
9.2.2 事件处理机制.....	217
9.2.3 事件处理的实现方式.....	218
9.2.4 适配器类.....	223
9.3 一般组件.....	225
9.3.1 标签——JLabel 类.....	225
9.3.2 按钮——JButton 类.....	226
9.3.3 文本框——JTextField 类与 JPasswordField 类.....	226
9.3.4 文本区——JTextArea 类.....	227
9.3.5 列表组件——JComboBox 类和 JList 类.....	229
9.3.6 滚动窗格——JScrollPane 类.....	231

9.3.7 复选框和单选按钮——JCheckBox 类和 JRadioButton 类	233
9.3.8 滑动条——JSlider 类	235
9.3.9 多事件处理示例	237
9.4 菜单与对话框	240
9.4.1 为窗口加入菜单	240
9.4.2 弹出式菜单	244
9.4.3 对话框	245
9.5 布局管理器	251
9.5.1 顺序布局——FlowLayout	251
9.5.2 边界布局——BorderLayout	251
9.5.3 网格布局——GridLayout	252
9.5.4 卡片布局——CardLayout	253
9.5.5 手工布局	255
9.6 JApplet 的使用	256
9.7 Java 事件类方法列表	258
9.8 案例实现	259
9.9 本章小结	263
习题	264
问题探究	267
第 10 章 多线程	268
10.1 多线程概述	268
10.1.1 程序、进程、多任务、线程和多线程	269
10.1.2 线程的状态与生命周期	271
10.1.3 线程的调度与优先级	273
10.2 创建和执行线程	273
10.2.1 利用 Thread 类创建线程	273
10.2.2 用 Runnable 接口创建线程	276
10.3 线程间的数据共享	278
10.4 多线程的同步控制	281
10.4.1 线程同步的相关概念	281
10.4.2 synchronized 应用	283
10.4.3 synchronized 的进一步说明	286
10.5 案例实现	286
10.6 本章小结	289
习题	289
问题探究	291
第 11 章 JDBC: Java 数据库访问技术	292
11.1 数据库及数据库关系系统	292

11.2	JDBC 基础	293
11.2.1	JDBC 工作区域	294
11.2.2	JDBC 连接方式的选择	294
11.2.3	JDBC API 提供的常用类和接口	295
11.2.4	JDBC 数据库访问步骤	296
11.3	使用 Microsoft Access 数据库管理系统	298
11.3.1	数据库的创建	298
11.3.2	连接数据库的 ODBC 数据源	299
11.4	运用 JDBC 进行数据库的简单查询	301
11.4.1	常用 SQL 语句	301
11.4.2	运用 JDBC-ODBC 进行数据库查询示例	302
11.5	运用 JDBC 进行数据库插入、删除和更新	304
11.6	案例实现	305
11.7	本章小结	309
	习题	309
	问题探究	310
第 12 章	综合案例: 聊天通信	311
12.1	聊天效果与代码	311
12.1.1	聊天效果	311
12.1.2	完整代码	311
12.2	应用程序框架分解	315
12.2.1	Socket 连接的建立	316
12.2.2	基于 TCP 的 Socket 数据通信架构	318
12.2.3	图形界面与事件处理界面设计	321
12.3	知识点	324
12.3.1	网络通信的层次	324
12.3.2	通信端口	325
12.3.3	Java 语言网络编程中主要使用的类和可能产生的异常	325
12.3.4	Socket 通信模式	325
12.3.5	Socket 类、ServerSocket 类的构造方法与常用方法	326
12.3.6	DataInputStream 和 DataOutputStream 类的继承层次与应用	327
12.3.7	多线程处理机制	330
	思考题	331
附录	实验安排	332
	参考文献	336

第 1 章 Java 概述

目前,全球拥有 450 万 Java 开发者,中国大约有 200 多万 Java 程序员的空缺,近年来,不论是就业市场需求还是薪酬方面,掌握 Java 技术的人一直领先。为什么继 C/C++ 语言之后,Java 语言能在业界得到快速普及和广泛认可,掌握 Java 语言的人在就业时拥有更多优势呢?让我们循序渐进,走进 Java,学习 Java,掌握 Java。

本章主要内容包括:

- ☑ Java 与 C、C++
- ☑ Java 语言的特点
- ☑ Java 程序的类型及其不同编程模式
- ☑ Java 编程环境与开发工具

【案例分析】

使用面向对象方法,描述现实世界中的一个实体——售报亭,如图 1.1 所示。

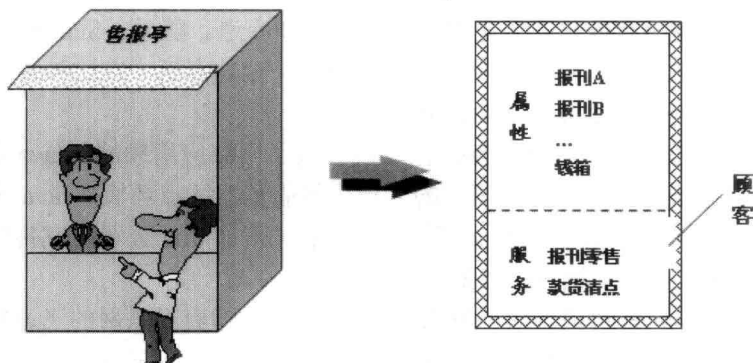


图 1.1 售报亭的对象封装

面向对象方法解决问题的思路是从现实世界中的客观对象(如人和事物)入手,尽量运用人类的自然思维方式来构造软件系统。

在面向对象方法中,把一切都看成是对象。把对象的属性和服务操作结合成一个独立的系统单位,其属性与操作刻画了事物的性质和行为,并尽可能隐蔽对象的内部细节,向外部只是提供接口。软件对象是数据和方法的封装体,如图 1.1 所示,属性对应软件对象的数据,服务对应软件对象的方法。

在面向对象系统中,无论是系统的构成成分,还是通过这些成分之间的关系而体现的系统结构,都可直接地映射问题域。这使得运用面向对象方法有利于正确理解问题域及系统责任。

1.1 Java 的崛起

1991 年, 美国 Sun Microsystems 公司启动了名为 Green Project 的研究项目, 旨在解决家用电器的智能通信和控制问题。开发小组最初想以当时颇为流行的 C++ 语言来进行开发, 但是后来, 由于 C++ 语言本身的复杂性、安全性以及平台移植方面的障碍与问题, 项目组最后决定基于 C++ 重新开发一套语言系统, 这样 Java 语言就产生了。

 **提示:** 无论使用哪种编程语言开发的应用程序, 都需要经过操作系统和处理器才能完成程序的运行, 因此, 这里所指的平台由操作系统 (OS) 和处理器 (CPU) 所构成。平台无关、可移植及跨平台的意思相同, 指的是软件的运行不因操作系统、处理器的变化而导致程序不能被识别和正确运行。

Java 语言的始创者是 Sun 公司的 James Gosling。最初, 他根据办公室窗外的一棵橡树 (Oak), 将这种语言命名为 Oak 语言。申请注册时, 因为命名冲突问题, 将其改名为 Java。

之所以命名为 Java 语言, 有两种说法: 其一, 印度尼西亚有一个重要的岛屿——爪哇岛, 该岛盛产咖啡, 开发人员起名 Java 寓意为世人端上一杯热腾腾的咖啡; 其二, Java 说法为美洲俚语——咖啡之意。

Java 语言可以称得上是一种精巧而安全的语言。然而, Java 的发展并非一帆风顺。一开始, Sun 公司就遭遇了智能化家用电器市场的萧条和冷遇。随后, Sun 公司以 Java 语言投标一个自认为乐观的交互式电视项目, 也未能成功, 无功而返。在这种情况下, Java 语言似乎生不逢时, Green 项目几乎陷入了绝境。

1993 年, 万维网空前流行起来, Java 的发展转向了网络应用领域。Java 语言具有平台无关性, 因此可以适应 Internet 上不可控的多样化的服务器站点环境。Java 程序既可以在 Windows 平台, 也可以在 UNIX、Linux 等平台上运行, 正如 Sun 公司所宣传的那样 “Write Once, Run Anywhere” (一次编写, 随处运行)。

1995 年 5 月, Sun 公司正式对外发布了 Java 语言。随着互联网的飞速发展, Java 语言逐渐确定了自己网络编程语言的地位, 并被美国著名杂志 *PC Magazine* 评为十大优秀科技产品之一。1996 年 1 月, JDK 1.0 发布。

目前, Java 平台主要包括 Java SE (J2SE)、Java EE (J2EE) 和 Java ME (J2ME)。Java SE 称为 Java 标准版或 Java 标准平台; Java EE 称为 Java 企业版或 Java 企业平台, 用于构建企业级的服务应用; Java ME 为移动开发平台。其中, Java SE 是 Java 开发的基础。

在计算机领域中, 从来没有出现过像现在所发生的 Internet/WWW/Java 这样的 “火爆” 情况。

延伸阅读: 万维网和因特网

WWW 是环球信息网 (World Wide Web) 的缩写, 简称为 Web, 其中文名字为万维网。万维网包括 WWW 服务器和 WWW 浏览器。它是一个资源空间, 由 “统一资源标识符” (URL) 标识。这些资源通过超文本传输协议 (Hypertext Transfer Protocol) 传送给使用者, 而用户通过单击链接来获得资源。

因特网 (Internet) 是当前全球最大的、开放的、由众多网络相互联结而成的计算机网络。

万维网常常被视作因特网的同义词, 其实万维网只是依赖因特网运行的一项服务。万维网是基于因特网的, 被广泛应用于因特网之上。

1.2 Java 与 C、C++

随着程序规模的不断扩大, 在 20 世纪 60 年代末期出现了软件危机。当时的程序设计范型无法克服随着代码增多错误也呈级数般扩大的问题, 这时出现了一种新的程序设计范型——面向对象程序设计。

1.2.1 Java 与 C++

Sun 微系统公司的 Java 开发小组汲取了 C++ 的精华, 将其组合到 Java 中, 舍弃了 C++ 中低效率和不利于程序员使用的特性, 而且创造了一些新的特性——基于 Internet 应用程序开发所必需的动态性。

开发 Java 的目的并不是为了改进并最终取代 C++。Java 和 C++ 这两种语言是用来解决不同问题的, Java 用来开发需要共存于不同机器上的应用程序, 通常是基于 Internet 的应用程序; 相反, C++ 用来开发在一台特定机器上运行的程序, 尽管 C++ 程序被重新编译后也能够其他机器上运行。

Java 语言的许多基本结构与 C++ 是相似的, 有些甚至是相同的。例如, Java 是一种面向对象编程语言, 它用类来创建对象的实例。类具有数据成员和方法成员, 这和 C++ 中的类是相似的。

但是 Java 没有指针, 而在 C/C++ 编程语言中, 指针是一个基石。在 C++ 中, 正确使用指针能使程序更有效率, 但是指针难以掌握, 如果使用不当会导致运行错误。

Java 带有自动的垃圾收集器, 这是 C/C++ 中没有的功能。垃圾收集器是一个常规程序, 收集程序中不再使用的内存, 程序员不必编写代码来释放之前使用的内存。

在不同的平台上使用 C/C++ 程序时, 系统会对每种数据类型依平台的不同分配不同的字节数。而在 Java 中, 会为各种数据类型分配合理的固定位数, 在每种平台上都不改变, 这样便保证了 Java 的平台无关性。

C++ 中支持多重继承, 一个类可以有多个父类, 这种方式使 C++ 中的类可以使用多个父类的属性和方法, 但结构较混乱。而在 Java 中, 一个类只能有一个父类, 但是可以实现多个接口, 这样既达到多重继承的目的, 又保证了结构比多重继承更加清晰。

除此之外, Java 中不支持结构和联合、宏定义、头文件及友元, 从而大大保证了 Java 程序的安全性。

1.2.2 Java 与 C

C 语言是面向过程的程序设计语言, 在程序设计过程中倾向于面向行为。C 语言中, 程序设计的单元是函数, C 编程人员着重于编写函数。执行同一任务的一系列动作构成函