



高等教育应用型人才培养系列教材

——科瑞国际校企合作指定教材

Java 程序设计教程

徐翠娟 李德有 主编 左晓英 主审 (入门篇)



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

高等教育应用型人才培养系列教材

——科瑞国际校企合作指定教材

Java 程序设计教程

（入门篇）

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书在介绍 Java 基础语法的同时,由浅入深地引入面向对象编程的概念,能够让读者快速入门,理解 Java 程序设计语言。本书共分 6 章。第 1 章介绍了 Java 基础的起源与发展、Java 的特征以及 Java 技术的体系结构。第 2、3 章介绍了 Java 语言基础语法,包括标识符、关键字、数据类型、运算符、表达式、语句等。第 4 章介绍了字符串及数组。第 5 章介绍 Java 面向对象的特征,包括封装、继承、多态等。第 6 章详细介绍了面向对象中继承的概念及应用等。本书中提供了大量的经典实例,从而达到快速理解知识点的目的。

本书层次清晰,结构严谨,便于理解,注重应用,既可以作为高等院校相关专业的教材,也可作为从事软件开发工作的专业技术人员的参考书。

图书在版编目(CIP)数据

Java 程序设计教程:入门篇 / 徐翠娟, 李德有
主编. —北京: 中国铁道出版社, 2010. 9
高等教育应用型人才培养系列教材 科瑞国际校企合作指定教材
ISBN 978-7-113-11791-7

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

中国版本图书馆 CIP 数据核字 (2010) 第 164895 号

书 名: Java 程序设计教程 (入门篇)

作 者: 徐翠娟 李德有 主编

策划编辑: 秦绪好 刘彦会

责任编辑: 鲍 闻

封面设计: 付 巍

版式设计: 于 洋

读者热线电话: 400-668-0820

封面制作: 李 路

责任印制: 李 佳

出版发行: 中国铁道出版社 (北京市宣武区右安门西街 8 号 邮政编码: 100054)

印 刷: 三河市华业印装厂

版 次: 2010 年 9 月第 1 版

2010 年 9 月第 1 次印刷

开 本: 787mm×1092mm 1/16 印张: 11.75 字数: 281 千

书 号: ISBN 978-7-113-11791-7

定 价: 20.00 元

版权所有 侵权必究

凡购买铁道版图书, 如有印制质量问题, 请与本社计算机图书批销部联系调换。

Java 是由 Sun Microsystems 公司于 1995 年 5 月推出的 Java 程序设计语言和 Java 平台的总称。用 Java 实现的 HotJava 浏览器（支持 Java Applet）显示了 Java 的魅力：跨平台、动态的 Web、Internet 计算。从此，Java 被广泛接受并推动了 Web 的迅速发展，常用的浏览器现在均支持 Java Applet。2003 年，有 5.5 亿台桌面计算机上运行 Java 程序，75% 的开发人员将 Java 作为主要的开发工具。至今还保持着编程语言第一的市场份额，可以毫不夸张地说，Java 在软件界占据半壁江山。

每一种语言都会有出生、成长、壮大、衰落几个阶段，Java 也不例外，既然有它的黄金时期，就有它的衰落时期。不过我们大可放心，因为 Java 不只是一门语言，更是一门技术，Java 所代表的这一类语言将不会消失。人们需要跨平台、开源的技术，人们需要功能强大但使用简单的技术。我们认为未来高级编程语言将会更清晰的分为三大类：一类适合底层开发，追求效率，就像 C 语言；一类适合傻瓜式的开发，目标是使人人都能够编程，就像 Delphi；第三类则是在效率与简单之间寻找平衡，就像 Java。

本书共分 6 章。第 1 章介绍 Java 基础的起源与发展、Java 的特征以及 Java 技术的体系结构；第 2 章以及第 3 章介绍 Java 语言基础语法，包括标示符、关键字、数据类型、运算符、表达式、语句等；第 4 章介绍 Java 的数据类型中常用的字符串及数组；第 5 章开始介绍 Java 面向对象的特征，包括封装、继承、多态等；第 6 章详细介绍了面向对象中继承的概念及应用等。

该书是作者多年的教学经验及项目实战经验积累的结晶，是对 Java 应用技术及学生项目实战演练中出现的问题进行分析、总结的成果。分析结果表明：学生在学习 Java 技术中出现问题根本原因是由于找不到一本便于理解的、由浅入深出的教材，本套书有两册：《Java 程序设计教程（入门篇）》和《Java 程序设计教程（进阶篇）》。前者主要介绍 Java 核心基础技术，使读者了解 Java 语言的一些关键特性；后者主要介绍编程人员在开发时需要用到的高级应用技术。因此，两者搭配使用能够使读者快速掌握使用 Java 语言进行软件开发的技术。在这里我们将竭诚为 Java 爱好者提供帮助。如果您是一个有着丰富经验的开发人员，并且能够灵活应用像枚举和泛型这样的高级语言特性，那么您就不必学习完“入门篇”再学习“进阶篇”，不过在“进阶篇”中根据内容需要适当参考“入门篇”中的有关内容。两本书配合使用能满足读者学习 Java 语言的需求，使得读者不必再徘徊在 Java 技术的大门外，使其能够走进 Java 语言的美妙世界。

作者根据 Java 技术知识的掌握及运用情况，系统地重新规划了 Java 基础课程中所必须掌握的知识点，而且对于理解能力较好的读者，本书也提供了较高层次的知识。另外本教材结合案例教学方式，对 Java 实用技术进行了全面讲解，使读者能在最短的时间内熟练掌握

握 Java 技术。

本书由徐翠娟、李德有任主编，杨扬、孙首梅、郭景辉任副主编。第 1、2 章由徐翠娟编写，第 3、4 章由李德有编写，第 5 章由杨扬编写，第 6 章由郭景辉编写，孙首梅、何鑫、于震、马军、宋博伟、宿佳宁、宋磊、周洋也参加了本书的编写工作，全书由左晓英主审。

由于编者水平有限，书中难免有疏漏和不妥之处，敬请各位专家、老师和广大读者不吝指正。

编者

2010 年 6 月

第 1 章	Java 课程概述	1
1.1	Java 的由来	1
1.2	Java 的历史	2
1.3	Java 的特性	4
1.3.1	Java 语言的优点	4
1.3.2	Java 语言的关键特性	4
1.4	Java 程序运行机制	6
1.5	Java 虚拟机	7
1.6	JDK 的安装与配置	8
1.7	编写第一个 Java 程序	13
1.8	程序的检测	14
1.8.1	语法错误	14
1.8.2	语义错误	15
1.9	提高程序的可读性	16
	本章小结	17
第 2 章	标识符、关键字、数据类型	18
2.1	标识符与关键字	18
2.1.1	标识符	18
2.1.2	关键字	19
2.2	变量	19
2.3	基本数据类型概述	22
2.3.1	基本数据类型	22
2.3.2	整型类型	23
2.3.3	字符类型	27
2.3.4	单精度浮点型与双精度浮点型	29
2.3.5	布尔类型	31
2.3.6	基本数据类型的默认值	31
2.3.7	数据类型的转换	32
	本章小结	34
第 3 章	运算符、表达式、语句	36
3.1	运算符、表达式与语句	36
3.1.1	运算符与表达式	36
3.1.2	赋值运算符	36
3.1.3	运算符的优先级	45

3.2 表达式	46
3.3 循环与选择性语句	49
3.3.1 程序的结构设计	49
3.3.2 选择结构	51
3.3.3 多重选择语句——switch 语句	55
3.3.4 while 循环	57
3.3.5 do...while 循环	59
3.3.6 for 循环	60
3.3.7 循环嵌套	61
3.3.8 循环的跳出	63
3.3.9 局部变量	66
本章小结	66
第 4 章 字符串和数组	68
4.1 String、StringBuffer 和 StringBuilder	68
4.1.1 String 类	68
4.1.2 StringBuffer 类	68
4.1.3 StringBuilder 类	69
4.2 数组	69
4.2.1 一维数组	69
4.2.2 二维数组	75
4.3 多维数组	78
本章小结	79
第 5 章 类的基本形式	80
5.1 面向对象程序设计的基本概念	80
5.1.1 对象	80
5.1.2 类	81
5.1.3 封装性	81
5.1.4 继承性	82
5.1.5 多态性	82
5.2 类与对象	82
5.2.1 类的声明	83
5.2.2 创建新的对象	84
5.2.3 创建多个新对象	86
5.3 类的封装性	88
5.4 在类内部调用方法	93
5.5 引用数据类型的传递	95
5.6 匿名对象	98
5.7 构造方法	99
5.8 对象的比较	103

5.9 this 关键字的使用	107
5.10 static 关键字的使用	112
5.10.1 静态变量	112
5.10.2 静态方法	116
5.10.3 理解 main()方法	118
5.10.4 静态代码块	119
5.11 私有构造方法	121
5.12 对象数组的使用	122
5.13 内部类	124
5.13.1 在类外部引用内部类	128
5.13.2 在方法中定义内部类	128
5.14 Java 文档注释	131
5.14.1 javadoc 标记	132
5.14.2 文档注释的一般形式	134
5.14.3 javadoc 的输出	134
本章小结	138
第 6 章 类的继承	140
6.1 继承的基本概念	140
6.1.1 子类对象的实例化过程	143
6.1.2 super 关键字的使用	146
6.1.3 限制子类的访问	149
6.1.4 覆写	151
6.2 抽象类	155
6.3 Object 类	159
6.4 final 关键字	161
6.5 接口 (interface)	162
6.6 对象多态性	166
6.6.1 instanceof 关键字的使用	170
6.6.2 覆写 Object 类中的 equals()方法	171
6.6.3 接口对象的实例化	173
6.7 匿名内部类	176
本章小结	179

第 1 章

Java 课程概述

在本章的开始部分，我们首先为大家介绍 Java 的发展史，作为使用 Java 程序设计语言编程的程序员有必要先来了解一下这些内容。很多读者在学习的时候并不重视这个部分的内容，因为这些“陈芝麻，烂谷子”好像对编程本身并没有什么帮助。其实这种想法是不对的。我国著名的哲学家孟子曾言：“颂其诗，读其书，不知其人可乎？是以论其世也。”意思就是说文学作品和作家本人的生活思想以及时代背景有着极为密切的关系，如果你不了解李白的为人，就不会知道为什么他能写出“君不见黄河之水天上来”的千古佳句，如果你不了解杜甫生活的时代和他悲天悯人的情怀，你就无法理解什么是“朱门酒肉臭，路有冻死骨”。而我们学习 Java 也是一样的，如果你不了解 Java 产生的背景，就不可能知道 Java 语言的优势在哪里，以及 Java 未来的发展方向是什么？而这些对于一个 Java 程序员来说都是至关重要的知识。好了，请读者跟我回到上个世纪的 90 年代，那个互联网风起云涌、英雄辈出的时代，回顾 Java 语言从发展到壮大的整个过程。

1.1 Java 的由来

Larry Wall 说，优秀程序员应有的三个特点：懒惰、急躁和傲慢。Java 就是诞生在一群懒惰、急躁而傲慢的程序天才之中。

1990 年 12 月，Sun 的工程师 Patrick Naughton 由于忍受不了当时糟糕的 Sun C++ 工具，威胁要离开 Sun 转投当时在 Steve Jobs 领导之下的 NeXT 公司。Sun 公司的领导层为了留住这个程序天才，给他一个机会，启动了一个叫做 Stealth（秘密行动）的项目。随着 James Gosling 等人的加入，这个项目更名为 Green。其目标是使用 C++ 为嵌入式设备开发一种新的基础平台技术，James Gosling 本人负责开发一个 SGML 编辑器。正如人们事后分析的那样，这位天才的程序员太懒惰，所以没有把 C++ 学好，开发中到处碰壁；太急躁——所以不愿意停下来读读 Scott Meyers 的新书《Effective C++》；太傲慢——所以轻易地决定开发一种新的编程语言。他把这种语言命名为 C++++-，意思是 C++ “加上一些好东西，减去一些坏东西”。显然这个冗长的名字不可能“长命百岁”，很快这种颇受同伴喜爱的语言被命名为 Oak。到了 1992 年 9 月，Oak 语言连同 Green OS 和一些应用程序一起发布在 Start 7 的小设备上，从而使之有了第一次精彩的亮相。随后，Sun 开了一家名为 FirstPerson 的公司，整个团队被转移到这家公司里研发机顶盒，以投标时代华纳公司的一个项目。这些天才被技术所鼓舞，开发出了一个高交互性的设备，结果没想到时代华纳公司和有线电视服务商并不愿意用户拥有那么大的控制权，从而在竞标之战中败给了 SGI。Oak 的锋芒之锐，竟然把客户都给吓懵了。Sun 沮丧地关闭了 FirstPerson，召回了整个团队。事实证明，传统行业中那些保守主义者是腐朽没落的。回去！回到激情澎湃的

IT 产业, 抓住互联网的大潮, 这才是出路! 1994 年, Oak 被命名为 Java, James Gosling 参加了在硅谷召开的“技术、教育和设计大会”。他抱着试一试的心情向与会者演示了 Java 的功能。他点击了网页上的一个静止的分子结构图标, 一条命令通过互联网送到了几百英里外的网站, 下载了一段 Java 小程序, 在本地工作站上开始执行。在几秒的时间里, 原本静止的网页上, 一个分子图像以三维动画的方式转了起来。“哇!” James Gosling 听到了全场观众的赞叹声。不久之后, 硅谷最大的报纸《圣荷西信使报》在头版刊登了一篇专文《为什么 Sun 认为一杯热咖啡能让你鼓足精神》。从文章见报开始, 负责 Java 市场人员的电话响个不停。不久, 全美国的主要报纸杂志都刊登了 Java 的报道。

1.2 Java 的历史

Java 来自于 Sun 公司的一个叫 Green 的项目, 其原先的目的是为家用电子消费产品开发一个分布式代码系统, 这样就可以把 E-mail 发送到电冰箱、电视机等家用电器的控制系统中, 对其进行控制以及信息交流。开始他们准备采用 C++, 但 C++ 太复杂、安全性差, 最后基于 C++ 开发了一种新的语言 Oak (Java 的前身)。Oak 是一种用于网络的精巧而安全的语言, Sun 公司曾以此投标一个交互式电视项目, 但结果被 SGI 打败。于是 Oak 几乎“无家可归”, 恰巧这时 Mark Andreessen 开发的 Mosaic 和 Netscape 启发了 Oak 项目组成员, 他们用 Java 编制了 Hot Java 浏览器, 得到了 Sun 公司首席执行官 Scott McNealy 的支持, 开启了 Java 进军 Internet 的大门。

Java 技术是由美国 Sun 公司倡导和推出的, 它包括 Java 语言和 Java Media APIs、Security APIs、Management APIs、Java Applet、Java RMI、JavaBeans、JavaOS、Java Servlet、Java Server Page 以及 JDBC 等。Java 技术的发展历程简述如下:

☐ 1990 年, Sun 公司 James Gosling 领导的小组设计了一种平台独立的语言 Oak, 主要用于为各种家用电器编写程序。

☐ 1995 年 1 月, Oak 更名为 Java。

☐ 1995 年 5 月 23 日, Sun 公司在 Sun World 95 上正式发布 Java 和 Hot Java 浏览器。

☐ 1995 年 8 月—12 月, Netscape 公司、Oracle 公司、Borland 公司、SGI 公司、Adobe 公司、IBM 公司、AT&T 公司、Intel 公司获得 Java 许可证。

☐ 1996 年 1 月, Sun 公司宣布成立新的业务部门——Java 软件部, 开发、销售并支持基于 Java 技术的产品, 由 Alan Baratz 任总裁。同时推出 Java 开发工具包 JDK (Java Development Kit) 1.0, 为开发人员提供编制 Java 应用软件所需的工具。

☐ 1996 年 2 月, Sun 公司发布 Java 芯片系列, 包括 PicoJava、MicroJava 和 UltraJava, 并推出 JDBC (Java Database Connectivity)。

☐ 1996 年 3 月, Sun 公司推出 Java Workshop。

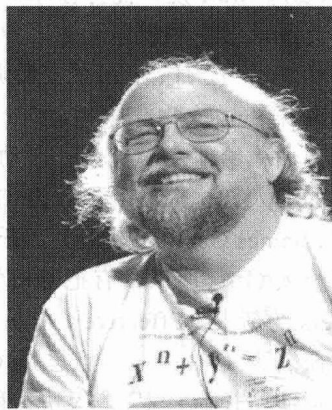


图 1-1 “懒惰”、“急躁”和“傲慢”的程序天才——James Gosling

☐ 1996年4月, Microsoft公司、SCO公司、苹果电脑公司(Apple)、NEC公司等获得Java许可证。Sun公司宣布允许苹果电脑、HP、日立、IBM、Microsoft、Novell、SGI、SCO、Tandem等公司将Java平台嵌入到其操作系统中。

☐ 1996年5月, HP公司、Sybase公司获得Java许可证。北方电讯公司宣布把Java技术和Java微处理器应用到其下一代电话机中的计划。5月29日, Sun公司在旧金山举行第一届JavaOne世界Java开发者大会, 业界人士踊跃参加。Sun公司在大会上推出一系列Java平台新技术。

☐ 1996年8月, Java Workshop成为Sun公司通过互联网提供的第一个产品。

☐ 1996年9月, Addison-Wesley和Sun公司推出Java虚拟机规范和Java类库。

☐ 1996年10月, 德州仪器等公司获得Java许可证。Sun公司提前完成JavaBeans规范并发布, 同时发布第一个Java JIT(Just-In-Time)编译器, 并打算在Java Workshop和Solaris操作系统中加入JIT。10月29日, Sun公司发布Java企业计算技术, 包括Java Station网络计算机、65家公司发布的85个Java产品及应用、7个新的Java培训课程及Java咨询服务、基于Java的Solstice互联网邮件软件、新的Java开发者支持服务、Hot Java Views演示、Java Tutor、Java Card API等。Sun公司宣布完成Java Card API规范, 这是智能卡使用的第一个开放API。Java Card规范将把Java能力“赋予”全世界亿万张智能卡。

☐ 1996年11月, IBM公司获得JavaOS和Hot Java许可证。Novell公司获得Java Workshop许可证。Sun公司和IBM公司宣布双方就提供Java化的商业解决方案达成一项广泛协议, IBM公司同意建立第一个Java检验中心。

☐ 1996年12月, Xerox等公司获得Java或JavaOS许可证。Sun公司发布JDK1.1、Java商贸工具包、JavaBeans开发包及一系列Java APIs; 推出一个新的JavaServer产品系列, 其中包括Java Web Server、Java NC Server和JavaServer Toolkit。Sun公司发布100%纯Java计划, 得到百家公司的支持。

☐ 1997年1月, SAS等公司获得Java许可证。Sun公司交付完善的JavaBeans开发包, 这是在确定其规范后不到8个月内完成的。

☐ 1997年2月, Sun公司和ARM公司宣布同意使JavaOS运行在ARM公司的RISC处理器架构上。Informix公司宣布在其Universal Server和其他数据库产品上支持JDK1.1。Netscape公司宣布其Netscape Communicator支持所有Java化的应用软件和核心API。

☐ 1997年3月, HP公司获得Java Workshop许可证, 用于其HP-UX操作系统。西门子、AG公司等获得Java许可证。日立半导体公司、Informix公司等获得JavaOS许可证。Novell公司获得Java Studio许可证。Sun公司发售JavaOS 1.0操作系统, 这是一种在微处理器上运行Java环境的代价最小、最快的方法, 可提供给JavaOS许可证持有者使用。Sun公司发售Hot Java Browser 1.0, 这是一种Java浏览器, 可以方便地按需编制专用的信息应用软件, 如客户自助台和打上公司牌号的网络应用软件。

☐ 1996年6月, Sun公司发布JSP1.0, 同时推出JDK1.3和Java Web Server 2.0。

☐ 1999年11月, Sun公司发布JSP1.1, 同时推出JSWDK1.0.1和Java Servlet 2.2。

☐ 2000年9月, Sun公司发布JSP1.2和Java Servlet 2.3 API。

☐ 2004 年，Sun 公司发布 JDK1.5。

☐ 2005 年 6 月，JavaOne 大会召开，Sun 公司公开 Java SE 6。此时，Java 的各种版本已经更名，以取消其中的数字“2”：J2EE 更名为 Java EE，J2SE 更名为 Java SE，J2ME 更名为 Java ME。

☐ 2006 年 12 月，Sun 公司发布 JRE6.0。

目前，Java 技术的架构主要包括以下三个方面：

☐ Java EE (Java Platform Enterprise Edition) 企业版，是以企业为环境而开发应用程序的解决方案。

☐ Java SE (Java Platform Standard Edition) 标准版，是桌面开发和低端商务应用的解决方案。

☐ Java ME (Java Platform Micro Edition) 小型版，是致力于消费产品和嵌入式设备的最佳解决方案。

1.3 Java 的特性

首先让我们来看一下官方宣传 Java 的“广告词”：

Java 是一种简单的面向对象的、分布式的、解释的、健壮的、安全的、结构中立的可移植的性能很优异的多线程的动态语言。

1.3.1 Java 语言的优点

Java 语言是一种优秀的编程语言。它最大的优点就是与平台无关，在 Windows、Solaris、Linux、Mac OS 以及其他平台上，都可以使用相同的代码。“一次编写，到处运行”的特点，使其在互联网上被广泛采用。

由于 Java 语言的设计者们十分熟悉 C++ 语言，所以在设计时借鉴了 C++ 语言。可以说，Java 语言是一种比 C++ 语言“还面向对象”的一种编程语言。Java 语言的语法结构与 C++ 语言的语法结构十分相似，这使得 C++ 程序员学习 Java 语言更加容易。

当然，如果仅仅是对 C++ 改头换面，那么就不会有今天被众人追捧的 Java 了。Java 语言提供的一些有用的新特性，使得使用 Java 语言比 C++ 语言更容易写出“无错代码”。

这些新特性包括：

☐ 提供了对内存的自动管理，程序员无须在程序中进行分配、释放内存，那些可怕的内存分配错误不会再打扰设计者了。

☐ 去除了 C++ 语言中的令人费解、容易出错的“指针”，用其他方法来进行实现。

☐ 避免了赋值语句（如 `a=3`）与逻辑运算语句（如 `a==3`）的混淆。

☐ 取消了多重继承这一复杂的概念。

Java 语言的规范是公开的，可以在 <http://www.sun.com> 上找到它，阅读 Java 语言的规范是提高技术水平的好方法。

1.3.2 Java 语言的关键特性

Java 语言有许多有效的特性，吸引着程序员们，最主要的有以下几个：

1. 简洁有效

Java 语言是一种相当简洁的“面向对象”程序设计语言。Java 语言省略了 C++ 语言中所有的难以理解、容易混淆的特性，例如头文件、指针、结构、单元、运算符重载、虚拟基础类等。它更加严谨、简洁。

2. 可移植性

对于一个程序员而言，写出来的程序如果无须修改就能够同时在 Windows、MacOS、UNIX 等平台上运行，简直就是美梦成真的好事！而 Java 语言就让这个原本遥不可及的事越来越近现实了。使用 Java 语言编写的程序，只要做较少的修改，甚至有时根本无须修改就可以在不同平台上运行。

3. 面向对象

可以这么说，“面向对象”是软件工程学的一次革命，大大提升了人类的软件开发能力，是一个伟大的进步，是软件发展的一个重大的里程碑。

在过去的 30 年间，“面向对象”有了长足的发展，充分体现了其自身的价值，到现在已经形成了一个包含“面向对象的系统分析”、“面向对象的系统设计”、“面向对象的程序设计”的完整体系。所以作为一种现代编程语言，是不能够偏离这一方向的，Java 语言也不例外。

4. 解释型

Java 语言是一种解释型语言，相对于 C/C++ 语言来说，用 Java 语言写出来的程序效率低，执行速度慢。但它正是通过在不同平台上运行 Java 解释器，对 Java 代码进行解释，来实现“一次编写，到处运行”的宏伟目标。为了达到目标，牺牲效率还是值得的，况且，现在的计算机技术日新月异，运算速度也越来越快，用户是不会感到太慢的。

5. 适合分布式计算

Java 语言具有强大的、易于使用的联网能力，非常适合开发分布式计算程序。Java 应用程序可以像本地文件系统一样通过 URL 访问远程对象。

使用 Java 语言编写 Socket 通信程序十分简单，使用它比使用任何其他语言都简单。而且它还十分适用于公共网关接口（CGI）脚本的开发，另外还可以使用 Java 小应用程序（Java Applet）、Java 服务器页面（Java Server Page, JSP）、Servlet 等手段来构建更丰富的网页。

6. 拥有较好的性能

正如前面所述，由于 Java 是一种解释型语言，所以它的执行效率相对就会慢一些，但由于 Java 语言采用了两种手段，使得其性能还是不错的。

Java 语言源程序编写完成后，先使用 Java 伪编译器进行伪编译，将其转换为中间码（也称为字节码），然后再进行解释。

提供了一种“准实时”（Just-in-Time, JIT）编译器，当需要更快的速度时，可以使用 JIT 编译器将字节码转换成机器码，然后将其进行缓冲，这样速度就会更快。

7. 健壮、防患于未然

Java 语言在伪编译时，做了许多早期潜在问题的检查，并且在运行时又做了一些相应的检查，可以说是一种最严格的“编译器”。

它的这种“防患于未然”的手段将许多程序中的错误扼杀在摇篮之中。经常有许多在其他

语言中必须通过运行才会暴露出来的错误，在编译阶段就被发现了。

另外，在 Java 语言中还具备了许多保证程序稳定、健壮的特性，有效地减少了错误，这样使得 Java 应用程序更加健壮。

8. 具有多线程处理能力

线程，是一种轻量级进程，是现代程序设计中必不可少的一种特性。多线程处理能力使得程序能够具有更好的交互性、实时性。

Java 在多线程处理方面性能超群，具有让设计者惊喜的强大功能，而且在 Java 语言中进行多线程处理很简单。

9. 具有较高的安全性

由于设计 Java 语言时，在安全性方面考虑得很仔细，做了许多探究，使得 Java 语言成为目前最安全的一种程序设计语言。

尽管 Sun 公司曾经许诺：“通过 Java 可以轻松构建出防病毒、防黑客的系统”，但“世界上没有绝对的安全”这一真理是不会因为某人的许诺而失效的。

就在 JDK (Java Development Kit) 1.0 发布不久后，美国 Princeton (普林斯顿) 大学的一组安全专家发现了 Java 1.0 安全特性中的第一例错误。从此，Java 安全方面的问题开始被关注。不过至今所发现的安全隐患都是微不足道的，而且 Java 开发组还宣称，他们对系统安全方面的 Bug 非常重视，会对这些发现的 Bug 立即进行修复。

而且由于 Sun 公司开放了 Java 解释器的细节，所以有助于通过各界力量，共同发现、防范这些安全隐患。

10. Java 是一种动态语言

Java 是一种动态的语言，这表现在以下两个方面：

- ① 在 Java 语言中，可以简单、直观地查询运行时的信息；
- ② 可以将新代码加入到一个正在运行的程序中去。

11. 具有中性结构

“Java 编译器生成的是一种中性的对象文件格式。”也就是说，Java 编译器通过伪编译后，将生成一个与任何计算机系统无关的“中性”的字节码。

这种中性结构其实并不是 Java 首创的，在 Java 出现之前 UCSD Pascal 系统就已在一种商业产品中做到了这一点，另外在 UCSD Pascal 之前也有这种方式的先例，在 Nicklaus Wirth 实现的 Pascal 语言中就采用了这种降低一些性能，换取更好的可移植性和通用性的方法。

经过精心设计的 Java 字节码，能够很好地与当今大多数流行的计算机系统兼容，在任何机器上都易于解释，易于动态翻译成为机器代码。

1.4 Java 程序运行机制

前面我们已经为大家介绍了 Java 的历史和特性，我们可以看出：Java 是一门优秀的面向对象编程语言。要想掌握好这样一门语言，我们就必须了解 Java 程序的运行机制。Java 的运行机制与其他语言相比是比较特殊的，因为它在运行的过程当中既需要编译，也需要解释，所以说 Java 是一门编译解释型的语言，这和我们以前所学习的 C 语言有很大的差别。造成这种差别的

主要原因是 Java 使用了虚拟机技术。

在 20 世纪的 90 年代互联网发展的初期, Java 的创造者敏锐地发现随着互联网以及计算机的高速发展, 应用程序需要在不同的计算机上运行。如果为不同的计算机开发不同的程序, 成本太高, 也不利于程序的使用和传播。为了解决这样的问题, 就必须提升编程语言的可移植性。也就是说, 应用程序可以在不做任何修改的前提下在不同的计算机上运行。Java 语言正是以此作为前提设计出来的。在 Java 当中使用了当时极为先进的虚拟机技术, 大幅度的提升了程序的可移植性, 这也是 Java 取得巨大成功的重要原因之一。

在图 1-2 中, *.java 是用 java 程序设计语言编写的源程序, 通过 javac 命令编译成 *.class 文件, 在操作系统之上是由 Java 虚拟机来运行以 .class 为结尾的文件。这里的 class 文件在 Java 中我们称之为可执行性类文件; 或者对于虚拟机来说其实就是字节码文件, 也就是 0、1 代码, 即计算机能直接读懂的二进制代码。

通过图 1-3 我们还可以明显地看出, Java 语言在执行的过程当中有一个解释 Java 字节码文件的过程, 这个过程是由 Java 虚拟机完成的。Sun 公司为不同的计算机准备了不同的虚拟机, Java 字节码被这些虚拟机解释成为当前的硬件和操作系统可以执行的机器代码。Java 虚拟机的知识将在 1.5 节进行介绍。

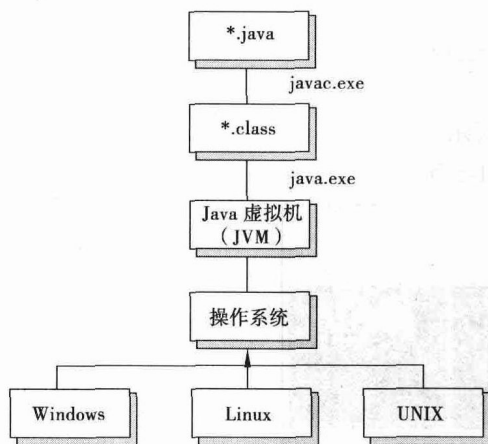


图 1-2 Java 程序运行机制

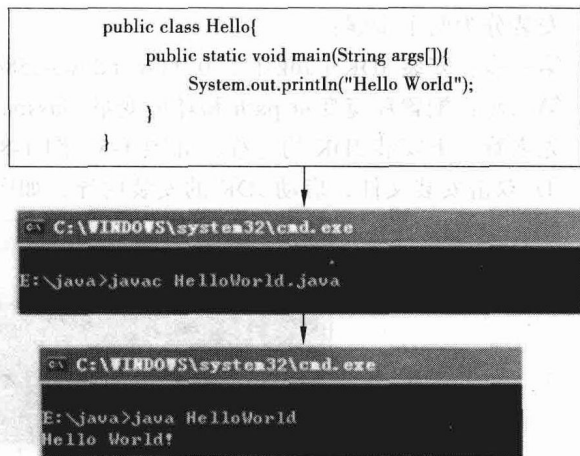


图 1-3 编译命令、执行命令

1.5 Java 虚拟机

Java 虚拟机 (JVM) 是可运行 Java 代码的假想计算机。只要根据 JVM 规范描述, 将解释器移植到特定的计算机上, 就能保证经过编译的任何 Java 代码可以通过解释器被运行, 即使 Java 代码在该系统上运行。

从图 1-4 中不难看出: Java 可以实现可移植性的原因——只要在操作系统上 (Windows、Linux) 植入 JVM, Java 程序就具有可移植性, 也符合 Sun 公司提出的口号 “Write Once, Run Anywhere” (“一次编写, 处处运行”)。

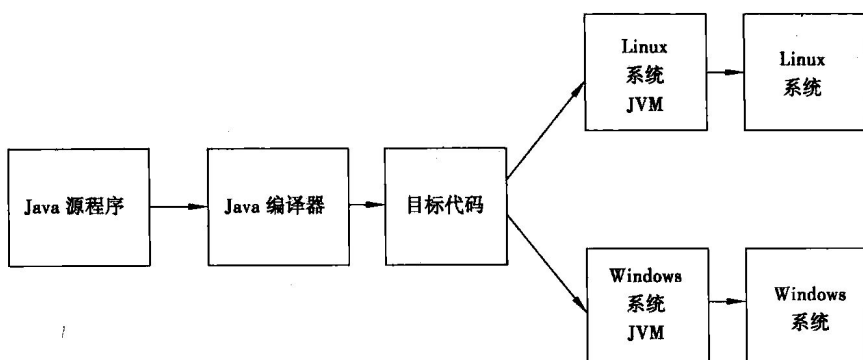


图 1-4 Java 虚拟机运行机制

1.6 JDK 的安装与配置

要开发 Java 程序首先必须配置好环境变量，而 Java 的运行环境的配置比较麻烦，相信有些读者也会有这种体会，下面来看一下 JDK 的安装过程。在这里 JDK 选用的是 J2SDK1.5 版本。

安装分为两个步骤：

第一步：安装 JDK (Jdk-1_5_0_11-windows-i586-p.exe)。

第二步：配置环境变量 path 和环境变量 classpath。

先来看一下安装 JDK 的过程，如图 1-5～图 1-8 所示。

① 双击安装文件，启动 JDK 的安装向导，如图 1-5 所示。



图 1-5 启动 JDK 安装向导

② 设置 JDK 的安装路径，如图 1-6 所示。

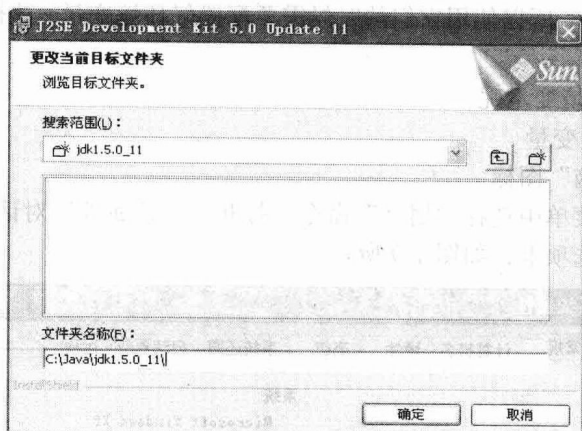


图 1-6 设置 JDK 安装路径

③ 设置 JRE 的安装路径并完成安装，如图 1-7、图 1-8 所示。

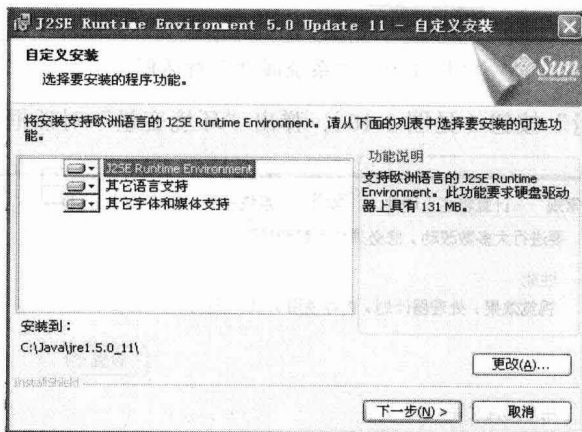


图 1-7 设置 JRE 安装路径



图 1-8 安装完成