



普通高等教育“十二五”规划教材

# Java程序设计

刘卫国 主编

中国铁道出版社  
CHINA RAILWAY PUBLISHING HOUSE



普通高等教育“十二五”规划教材

# Java 程序设计

主 编 刘卫国

副主编 彭 胜 黎 明

参 编 姚昱旻 柳小文 郭 美

肖 敏 李首洪 李 博

欧阳佳 文碧望 石 玉



中国铁道出版社  
CHINA RAILWAY PUBLISHING HOUSE

## 内 容 简 介

本书讲解了 Java 语言的基本知识及程序设计的基本方法,使读者掌握面向对象程序设计的基本概念,从而具有利用 Java 语言进行程序设计的能力,为将来从事软件开发,特别是 Web 应用系统开发打下良好基础。全书共分 10 章,从内容上大致分为三个部分:第一部分为第 1 章~第 3 章,介绍 Java 程序设计的基础知识,包括 Java 语言概述、Java 语言基础以及算法与程序控制结构。第二部分为第 4 章~第 6 章,介绍 Java 面向对象程序设计的基本方法与技术,这是 Java 的核心与特色内容,包括类与对象、封装、继承与多态以及异常处理与输入/输出。第三部分为第 7 章~第 10 章,介绍 Java 的实际应用,包括多线程、网络程序设计、数据库应用以及图形用户界面开发技术。

本书内容讲解详细,程序代码均经过调试,案例实用。

本书适合作为高等院校计算机程序设计课程的教材,也可作为具有一定程序设计基础和经验的读者的参考用书。

### 图书在版编目(CIP)数据

Java 程序设计 / 刘卫国主编. —北京:中国铁道出版社, 2012. 4

普通高等教育“十二五”规划教材

ISBN 978-7-113-14002-1

I. ①J… II. ①刘… III. ②JAVA 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 (2011) 第 001190 号

书 名: Java 程序设计

作 者: 刘卫国 主编

策 划: 吴宏伟

读者热线: 400-668-0820

责任编辑: 杜 鹃

编辑助理: 何 佳

封面设计: 刘 颖

封面制作: 白 雪

责任印制: 李 佳

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

网 址: <http://www.51eds.com>

印 刷: 航远印刷有限公司

版 次: 2012 年 4 月第 1 版 2012 年 4 月第 1 次印刷

开 本: 787mm×1092mm 1/16 印张: 19.25 字数: 460 千

印 数: 1~3 000 册

书 号: ISBN 978-7-113-14002-1

定 价: 34.00 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,请与本社教材研究开发中心批销部联系调换。电话:(010) 63550836

打击盗版举报电话:(010) 63549504

# 普通高等教育“十二五”规划教材

顾 问：桂卫华 中南大学  
主 任：邹北骥 中南大学  
副 主 任：施荣华 中南大学  
刘任任 湘潭大学  
沈 岳 湖南农业大学  
彭小宁 怀化学院

委 员：(按姓氏音序排列)

|     |          |     |        |
|-----|----------|-----|--------|
| 陈国平 | 吉首大学     | 傅自钢 | 湖南农业大学 |
| 高守平 | 湘南学院     | 龚德良 | 湘南学院   |
| 郭观七 | 湖南理工学院   | 黄同成 | 邵阳学院   |
| 蒋加伏 | 长沙理工大学   | 江世明 | 邵阳学院   |
| 李长云 | 湖南工业大学   | 李实英 | 湖南大学   |
| 李勇帆 | 湖南第一师范学院 | 林 华 | 湖南科技学院 |
| 刘 宏 | 湖南师范大学   | 刘华富 | 长沙学院   |
| 刘卫国 | 中南大学     | 罗 帅 | 湖南农业大学 |
| 骆嘉伟 | 湖南大学     | 石良武 | 湖南商学院  |
| 谭骏珊 | 中南林业科技大学 | 谭敏生 | 南华大学   |
| 谢建全 | 湖南财政经济学院 | 徐建波 | 湖南科技大学 |
| 徐蔚鸿 | 长沙理工大学   | 杨长兴 | 中南大学   |
| 羊四清 | 湖南人文科技学院 | 张新林 | 湖南科技学院 |

湖南省高等教育学会计算机教育专业委员会与中国铁道出版社长期合作，致力于计算机基础教育系列教材的编写、出版和发行。自 2005 年合作出版《大学计算机基础》、《大学计算机基础实验教程》、《C 语言程序设计》、《C 语言程序设计实验教程》、《Visual Basic 程序设计》、《Visual Basic 程序设计实验教程》、《Visual FoxPro 程序设计》和《Visual FoxPro 程序设计实验教程》四套八本教材以来，无论在教材编写、出版的质量方面，还是在推广发行方面都取得了巨大成功。迄今为止，这一系列教材发行情况较好，学生受益面不仅在湖南省各个高校，而且遍布全国其他地区高校。它不仅凝聚了全体作者的智慧和辛勤的劳动，也包含了为这一系列教材进行编辑、加工、出版和发行的出版社同仁们所作出的不懈努力。

计算机技术水平和应用能力是当代大学生的基本素质之一。随着计算机的普及和广泛应用，特别是计算机网络技术和软件技术的迅速发展，人们的工作和生活方式发生了彻底的变化。人们离不开计算机，计算机的各种软件如雨后春笋一般呈现在人们眼前，这就要求大家不断地学习，不断地使用各种软件，以适应社会的发展和需要。特别是当今的大学生，仅仅学习《大学计算机基础》和《程序设计》两门课程已很难满足现实的需要。因此，各个大学除了开设《大学计算机基础》和《程序设计》两门基础课程以外，还根据各个专业的需要，开设了《计算机网络技术》、《计算机辅助设计技术》、《多媒体技术》等系列选修课程。正是在这样一种背景下，我们组织各高校优秀的教师，编写了这套系列教材，分别是：《Java 程序设计》、《Java 程序设计实践教程》、《SQL Server 数据库应用技术》、《Access 数据库应用技术》、《网络技术及应用》、《网站建设》、《动态网页设计》、《多媒体技术及应用》、《单片机原理与应用》、《计算机组装与维护》和《计算机辅助设计》共十部。

我们真诚地希望这套教材的出版发行，能够促进计算机基础教学水平的提高，能够让全体学生通过使用这套教材，学到真正所需要的知识和技能。我们也真诚地欢迎使用这套教材的老师们，给我们提出宝贵的修改意见，以便在今后的再版中更加完善。

湖南省高等教育学会计算机教育专业委员会 邹北骥

2010 年 11 月 22 日

计算机程序设计是一门重要的计算机基础课程。通过本课程的学习,可使学生掌握利用计算机解决问题的基本思路和方法即算法,了解程序设计语言的组成要素和基本特征,具有编写程序和调试程序的基本技能。在程序设计中,算法是灵魂,语言是实现算法的工具。算法要通过语言描述才成为程序。由于不同学校、不同专业对学生程序设计能力的要求不尽相同,所以程序设计课程通常采用不同的教学语言。但程序语言只是一种载体,是一种学习程序设计的工具,无论选用哪种语言,都应掌握程序设计的基础知识与基本技术,学习的重点应落在程序设计的方法与应用开发能力上。

Java 语言是目前流行的程序设计语言之一。随着 Internet 的发展,特别是 Web 应用开发技术的发展,Java 语言越来越受到业界的青睐,越来越多的企业会选择使用 Java 及其相关技术来构建它们的应用系统。一方面是因为 Java 语言具有面向对象、平台无关性、可移植性、健壮性和安全性等优良特性,另一方面也因为 Java 技术对各个层次的分布式应用(包括嵌入式应用、桌面应用与企业级应用)都提供了全面系统的方法与技术。因此,目前 Java 已不仅仅是一种单纯的设计语言,而变成了一种支撑 Internet 应用的庞大技术体系,是一种开发软件的标准、规范与架构的统称。事实上,语言只是整个 Java 体系构架中的一部分,学习 Java 本身也不仅仅在于学习如何使用 Java 语言的语法,更多的时候是在学习如何应用 Java 所提供的资源与各种标准,以开发出架构更好、更容易维护的软件。

本书讲解了 Java 语言的基本知识及程序设计的基本方法,使读者掌握面向对象程序设计的基本概念,从而具有利用 Java 语言进行程序设计的能力,为将来从事软件开发,特别是 Web 应用系统开发打下良好基础。

全书共分 10 章,其内容体系大致分为三个部分:第一部分为第 1 章~第 3 章,这一部分从对 Java 语言的初步认识入手,介绍了 Java 程序设计的基础知识,包括语言概述、语言基础以及算法与程序流程控制。介绍了 Java 语言的基本数据类型与运算、程序的三种基本结构及其实现方法。第二部分为第 4 章~第 6 章,这一部分介绍面向对象程序设计的概念和基本方法,介绍 Java 的核心与特色内容,包括类与对象、封装、继承与多态以及异常处理与输入/输出。第三部分为第 7 章~第 10 章,这一部分内容是对基础知识的扩充和发展,介绍 Java 的实际应用,包括多线程、网络程序设计、数据库应用以及图形用户界面开发技术。

本书注重 Java 基础知识的介绍,结合 Java 流程控制语句介绍了程序设计的基本知识和一些常用算法,因此本书可以作为第一门程序设计教材供学生使用。同时,本书也强调面向对象程序设计的核心技术和基本应用技术,因此也可供具有一定程序设计基础和经验的读者阅读参考。

为了方便教学和读者上机操作练习,作者还组织编写了《Java 程序设计实践教程》一书,作为与本书配套的教学参考书。《Java 程序设计实践教程》与本教材相互配套,又是本教材很好的补充,其中一些内容可作为读者课外阅读材料,达到巩固与提高学生程序设计能力的目的。另外,还有与本书配套的教学课件与相关教学资源,供教师教学参考。

本书由刘卫国任主编，彭胜、黎明任副主编，参加编写的有姚昱旻、柳小文、郭美、肖敏、李首洪、李博、欧阳佳、文碧望、石玉，参与程序调试与资料整理的有欧鹏杰、胡勇刚等。邹北骥教授、施荣华教授、彭小宁教授对本书的编写给予了许多指导，在此表示诚挚的感谢。

由于作者学识水平有限，书中疏漏或不足之处在所难免，恳请广大读者批评指正。

作 者

2011 年 10 月

|       |                       |    |
|-------|-----------------------|----|
| 第 1 章 | Java 语言概述             | 1  |
| 1.1   | Java 语言的发展历史与应用       | 1  |
| 1.1.1 | Java 语言的发展历史          | 1  |
| 1.1.2 | Java 语言的应用            | 3  |
| 1.2   | Java 语言的特点            | 3  |
| 1.3   | Java 程序的运行机制          | 7  |
| 1.3.1 | Java 程序的分类            | 7  |
| 1.3.2 | Java 程序的运行过程          | 7  |
| 1.3.3 | Java 虚拟机              | 8  |
| 1.4   | Java 程序的运行环境          | 9  |
| 1.4.1 | Java 平台的版本选择          | 9  |
| 1.4.2 | Java 开发工具——JDK        | 9  |
| 1.4.3 | Java 集成开发环境——NetBeans | 12 |
| 1.4.4 | 简单的 Java 程序           | 12 |
|       | 小结                    | 17 |
|       | 思考与练习                 | 18 |
| 第 2 章 | Java 语言基础             | 20 |
| 2.1   | Java 符号集              | 20 |
| 2.1.1 | Java 中的标识符            | 20 |
| 2.1.2 | Java 的关键字             | 21 |
| 2.1.3 | Java 的分隔符             | 21 |
| 2.2   | 常量和变量                 | 22 |
| 2.2.1 | 常量的概念与使用              | 22 |
| 2.2.2 | 变量的概念与使用              | 23 |
| 2.3   | 基本数据类型                | 24 |
| 2.3.1 | 整型                    | 25 |
| 2.3.2 | 浮点型                   | 26 |
| 2.3.3 | 字符型                   | 26 |
| 2.3.4 | 布尔型                   | 27 |
| 2.3.5 | 基本数据类型值间的转换           | 27 |
| 2.3.6 | 基本数据类型类               | 28 |
| 2.4   | 常用数值计算方法              | 29 |
| 2.5   | 运算符和表达式               | 31 |
| 2.5.1 | 赋值运算                  | 31 |
| 2.5.2 | 算术运算                  | 32 |
| 2.5.3 | 关系运算与逻辑运算             | 33 |



|       |                       |    |
|-------|-----------------------|----|
| 2.5.4 | 位运算符 .....            | 35 |
| 2.5.5 | 其他运算符 .....           | 37 |
| 2.5.6 | 运算符的优先级及结合性 .....     | 38 |
| 2.6   | 字符串 .....             | 39 |
| 2.6.1 | String 类 .....        | 39 |
| 2.6.2 | StringBuffer 类 .....  | 41 |
| 2.7   | 数组 .....              | 43 |
| 2.7.1 | 数组的概念与创建 .....        | 43 |
| 2.7.2 | 数组的引用 .....           | 45 |
| 小结    | .....                 | 46 |
| 思考与练习 | .....                 | 47 |
| 第 3 章 | 算法与程序控制结构 .....       | 51 |
| 3.1   | 算法 .....              | 51 |
| 3.1.1 | 算法的概念及描述 .....        | 51 |
| 3.1.2 | 程序的三种基本结构 .....       | 53 |
| 3.2   | 顺序结构程序 .....          | 53 |
| 3.2.1 | Java 语句 .....         | 53 |
| 3.2.2 | 基本输入/输出 .....         | 54 |
| 3.2.3 | 顺序结构程序举例 .....        | 56 |
| 3.3   | 选择结构程序 .....          | 57 |
| 3.3.1 | 单分支条件语句 .....         | 57 |
| 3.3.2 | 双分支条件语句 .....         | 58 |
| 3.3.3 | 多分支条件语句 .....         | 59 |
| 3.3.4 | 开关语句 switch .....     | 61 |
| 3.3.5 | 分支结构程序举例 .....        | 63 |
| 3.4   | 循环结构程序 .....          | 65 |
| 3.4.1 | for 循环语句 .....        | 65 |
| 3.4.2 | while 循环语句 .....      | 67 |
| 3.4.3 | do...while 循环语句 ..... | 68 |
| 3.4.4 | 嵌套循环 .....            | 69 |
| 3.4.5 | 循环结构程序举例 .....        | 70 |
| 3.5   | 跳转语句 .....            | 72 |
| 3.5.1 | break 语句 .....        | 72 |
| 3.5.2 | continue 语句 .....     | 73 |
| 小结    | .....                 | 74 |
| 思考与练习 | .....                 | 75 |
| 第 4 章 | Java 类与对象 .....       | 81 |
| 4.1   | 面向对象的基本概念 .....       | 81 |
| 4.1.1 | 对象、类和消息 .....         | 81 |
| 4.1.2 | 面向对象的特性 .....         | 82 |

|              |                         |            |
|--------------|-------------------------|------------|
| 4.2          | Java 类 .....            | 83         |
| 4.2.1        | 类的基本结构 .....            | 83         |
| 4.2.2        | 类的成员 .....              | 84         |
| 4.2.3        | 构造方法 .....              | 86         |
| 4.3          | Java 对象 .....           | 87         |
| 4.3.1        | 对象的创建和初始化 .....         | 87         |
| 4.3.2        | 对象的引用和清除 .....          | 89         |
| 4.3.3        | 方法的调用 .....             | 93         |
| 4.4          | 包 .....                 | 97         |
| 4.4.1        | 包的概念 .....              | 97         |
| 4.4.2        | 包的创建与使用 .....           | 97         |
|              | 小结 .....                | 98         |
|              | 思考与练习 .....             | 99         |
| <b>第 5 章</b> | <b>封装、继承与多态 .....</b>   | <b>104</b> |
| 5.1          | 封装 .....                | 104        |
| 5.1.1        | 封装的概念 .....             | 104        |
| 5.1.2        | 访问控制符 .....             | 105        |
| 5.2          | 继承 .....                | 106        |
| 5.2.1        | 继承的概念 .....             | 106        |
| 5.2.2        | 继承的应用 .....             | 107        |
| 5.2.3        | this 与 super .....      | 112        |
| 5.2.4        | 继承的终止 .....             | 116        |
| 5.3          | 多态 .....                | 117        |
| 5.3.1        | 方法重载 .....              | 117        |
| 5.3.2        | 方法覆盖与变量多态 .....         | 118        |
| 5.3.3        | 抽象类 .....               | 119        |
| 5.4          | 接口 .....                | 123        |
| 5.4.1        | 接口的概念 .....             | 123        |
| 5.4.2        | 接口的应用 .....             | 124        |
| 5.4.3        | 抽象类与接口的比较 .....         | 126        |
|              | 小结 .....                | 126        |
|              | 思考与练习 .....             | 127        |
| <b>第 6 章</b> | <b>异常处理与输入/输出 .....</b> | <b>133</b> |
| 6.1          | 异常处理 .....              | 133        |
| 6.1.1        | 异常的概念 .....             | 134        |
| 6.1.2        | 异常的处理 .....             | 134        |
| 6.1.3        | 异常类的层次结构 .....          | 136        |
| 6.1.4        | 自定义异常 .....             | 137        |
| 6.1.5        | 异常处理的使用原则 .....         | 138        |
| 6.2          | Java 的 I/O 处理 .....     | 139        |

|       |                                             |     |
|-------|---------------------------------------------|-----|
| 6.2.1 | I/O 流的概念.....                               | 139 |
| 6.2.2 | 基本的 I/O 类.....                              | 140 |
| 6.2.3 | 实体流与过滤流.....                                | 143 |
| 6.2.4 | 标准输入/输出对象.....                              | 144 |
| 6.3   | 字节流应用.....                                  | 144 |
| 6.3.1 | InputStream 和 OutputStream.....             | 144 |
| 6.3.2 | 常用实体流.....                                  | 147 |
| 6.3.3 | 常用过滤流.....                                  | 148 |
| 6.3.4 | 使用示例.....                                   | 149 |
| 6.4   | 字符流应用.....                                  | 151 |
| 6.4.1 | Reader 和 Writer.....                        | 151 |
| 6.4.2 | 字符流中的实体流和过滤流.....                           | 152 |
| 6.4.3 | InputStreamReader 和 OutputStreamWriter..... | 153 |
| 6.4.4 | 使用示例.....                                   | 153 |
| 6.5   | 文件处理.....                                   | 154 |
| 6.5.1 | File 类.....                                 | 155 |
| 6.5.2 | 文件的顺序读/写.....                               | 156 |
| 6.5.3 | 文件的随机访问.....                                | 161 |
|       | 小结.....                                     | 164 |
|       | 思考与练习.....                                  | 164 |
| 第 7 章 | 多线程.....                                    | 170 |
| 7.1   | 进程与线程.....                                  | 170 |
| 7.1.1 | 进程的概念.....                                  | 170 |
| 7.1.2 | 线程的概念.....                                  | 171 |
| 7.2   | 线程的创建.....                                  | 172 |
| 7.2.1 | 继承 Thread 类.....                            | 172 |
| 7.2.2 | 实现 Runnable 接口.....                         | 173 |
| 7.2.3 | 两种创建方式的比较.....                              | 174 |
| 7.2.4 | 线程的生命周期.....                                | 174 |
| 7.3   | 线程的调度与同步机制.....                             | 175 |
| 7.3.1 | 线程调度.....                                   | 175 |
| 7.3.2 | 线程的同步.....                                  | 181 |
| 7.4   | 线程的通信.....                                  | 184 |
| 7.5   | 多线程应用实例.....                                | 187 |
|       | 小结.....                                     | 190 |
|       | 思考与练习.....                                  | 191 |
| 第 8 章 | 网络程序设计.....                                 | 199 |
| 8.1   | 网络程序设计基础知识.....                             | 199 |
| 8.1.1 | TCP/IP 简介.....                              | 199 |
| 8.1.2 | 地址、域名与端口号.....                              | 200 |
| 8.1.3 | InetAddress 类.....                          | 200 |

|        |                               |     |
|--------|-------------------------------|-----|
| 8.2    | URL 编程.....                   | 201 |
| 8.2.1  | URL 类.....                    | 201 |
| 8.2.2  | URLConnection 类.....          | 204 |
| 8.3    | Socket 编程.....                | 206 |
| 8.3.1  | Socket 的概念.....               | 206 |
| 8.3.2  | Socket 编程.....                | 207 |
| 8.4    | Datagram 编程.....              | 211 |
| 8.4.1  | Datagram 的概念.....             | 211 |
| 8.4.2  | Datagram 编程的方法.....           | 212 |
|        | 小结.....                       | 220 |
|        | 思考与练习.....                    | 221 |
| 第 9 章  | 数据库应用开发.....                  | 225 |
| 9.1    | JDBC 概述.....                  | 225 |
| 9.1.1  | 数据库的基本概念.....                 | 225 |
| 9.1.2  | JDBC 的原理.....                 | 227 |
| 9.2    | 数据库访问.....                    | 228 |
| 9.2.1  | JDBC 的常用接口和类.....             | 228 |
| 9.2.2  | JDBC 数据库连接.....               | 230 |
| 9.2.3  | JDBC 数据库操作.....               | 233 |
|        | 小结.....                       | 244 |
|        | 思考与练习.....                    | 244 |
| 第 10 章 | Java 图形用户界面设计.....            | 248 |
| 10.1   | 图形用户界面设计概述.....               | 248 |
| 10.1.1 | Java 图形用户界面工具.....            | 248 |
| 10.1.2 | Java 图形用户界面的构成.....           | 250 |
| 10.1.3 | Java 事件响应及处理机制.....           | 251 |
| 10.2   | 容器和布局管理器.....                 | 252 |
| 10.2.1 | Swing 的常用容器.....              | 252 |
| 10.2.2 | 常用的布局管理器.....                 | 261 |
| 10.3   | Swing 的常用控件.....              | 265 |
| 10.3.1 | Swing 标签和文本框.....             | 265 |
| 10.3.2 | Swing 命令按钮.....               | 268 |
| 10.3.3 | Swing 的组合框和列表框.....           | 269 |
| 10.3.4 | Swing 复选框和单选按钮.....           | 270 |
| 10.3.5 | Swing 中的高级控件.....             | 274 |
| 10.4   | 使用 Netbeans 开发简单的 GUI 应用..... | 282 |
| 10.4.1 | Netbeans 的 GUI 生成器.....       | 282 |
| 10.4.2 | Netbeans 的 GUI 开发案例.....      | 283 |
|        | 小结.....                       | 286 |
|        | 思考与练习.....                    | 287 |
|        | 参考文献.....                     | 292 |

# 第 1 章 Java 语言概述

## 引言

在经历了以大型机为代表的集中计算模式和以 PC 为代表的分散计算模式之后, Internet 的出现使得计算模式进入了网络计算时代。网络计算模式的一个特点是计算机是异构的, 即计算机的类型和操作系统是不一样的。网络计算模式的另一个特点是代码可以通过网络在各种计算机上进行迁移, 这就迫切需要一种跨平台的程序设计语言, 使得用其编写的程序能够在网络中的各种计算机上正常运行, Java 就是在这种需求下应运而生的。

随着 Internet 的发展, 特别是 Web 应用开发技术的发展, Java 语言越来越受到业界的青睐, 越来越多的企业会选择使用 Java 及其相关技术来构建它们的应用系统。如果进行 Internet 相关的程序开发, Java 语言是最佳选择。本章主要介绍 Java 语言的发展历史和应用、Java 语言的特点和优越性、Java 程序的运行机制以及 Java 语言的开发环境等内容。

## 学习目标

通过对本章内容的学习, 读者应该能够:

- 了解 Java 语言的发展历史和现状。
- 理解 Java 语言的自身特点和 Java 程序的运行机制。
- 掌握 Java 开发工具 (JDK) 的安装与配置, 能使用文本编辑器和 Java 集成开发环境 (IDE) 编写、运行最基本的 Java 程序。

## 1.1 Java 语言的发展历史与应用

Java 语言是美国 Sun 公司推出的新一代面向对象程序设计语言, 而 Java 本是印度尼西亚的一个小岛——爪哇岛的名字, 爪哇岛盛产咖啡, 将两者联系在一起有着很有趣的故事。当初 Java 语言还不为世人所熟知, 几位项目组的成员在讨论给这个新的语言取什么名字时, 他们正在喝一种来自 Java 岛的咖啡, 有人灵机一动说就叫 Java 怎样, 很快得到了其他人的赞赏, 于是 Java 这个名字就这样传开了, 寓意着为世人端上一杯热咖啡。现在看来, Java 这杯咖啡已经飘香世界各地。

### 1.1.1 Java 语言的发展历史

Java 语言来源于 Sun 公司的 Green 项目。早在 1991 年, Sun 公司启动了 Green 项目, 该项目最初的目的是为了开发一种分布式系统架构, 使其能够在各种智能家电上运行。由于该项目组的成员都具有深厚的 C++ 开发背景, 所以一开始准备使用 C++ 语言来完成这个项目, 但很快便意识到 C++ 过于复杂和庞大且安全性差, 并不适用于这个项目, 因此有必要为这个项目专门研发一种

新的程序设计语言，于是对一门新语言的研究就这样展开了。最后，项目组成员基于 C++ 开发了一种新的语言 Oak（橡树），这就是 Java 语言的前身。

Oak 语言是一种用于网络的精巧而安全的语言。该语言采用了许多 C++ 语言的语法，提高了安全性，并且是面向对象的语言，但是 Oak 语言在商业上并未获得成功。直至 1994 年，针对 Internet 的迅猛发展和 WWW 应用的快速增长，Sun 公司发现 Oak 语言所具有的跨平台、面向对象、安全性高等特点非常符合 Internet 的需要，于是改进了该语言的设计，实现了如下几个目标：

- （1）创建一种面向对象的程序设计语言，而不是面向过程的语言。
- （2）提供一个解释执行的程序运行环境，使程序代码独立于计算机系统平台。
- （3）吸收 C 和 C++ 的优点，使程序员更容易掌握。
- （4）剔除 C 和 C++ 中影响程序健壮性的部分，使程序更安全，例如指针、内存申请和释放等。
- （5）实现多线程，使得程序能够同时执行多个任务。
- （6）提供动态下载程序代码的机制。
- （7）提供代码校验机制以保证安全性。

最终，更为完善的 Oak 版本被重新起了一个更响亮的名字——Java。1995 年春，Sun 公司发表了 Java 的完整技术规范，立即得到包括 Netscape 公司在内的各 WWW 厂商的广泛支持。不久，许多著名的大公司，如 IBM、Novell、Oracle、SGI 和 Borland 公司等相继购买了 Java 的使用权，Java 得到了广泛的支持。Bill Gates 在反复仔细研究了 Java 的技术规范之后，认为“Java 确实是最卓越的程序设计语言”，继而 Microsoft 公司购买了 Java 的使用权并很快推出了集成化的 Java 开发环境——Visual J++，至此 Java 在软件界的地位不可动摇。

1995 年 Java 公开亮相后，经过一年的试用和改进，Java 1.0 版终于在 1996 年年初正式发表。这个版本包括两部分：Java 运行环境（Java Runtime Environment, JRE）和 Java 开发工具（Java Development Kit, JDK）。在 Java 运行环境中包括了核心 API（Application Program Interface）、集成 API、用户界面 API、发布技术、Java 虚拟机（Java Virtual Machine, JVM）5 部分，而 Java 开发工具还包括了编译 Java 程序的编译器（即 javac）。在 JDK 1.0 时代，JDK 除了 AWT（一种用于开发图形用户界面的 API）外，其他的库并不完整。

1997 年 2 月发布了 JDK 1.1，1998 年 12 月发布了 JDK 1.2 版本，1999 年 6 月 Sun 公司发布 Java 的 3 个版本：标准版（Java 2 Standard Edition, J2SE）、企业版（Java 2 Enterprise Edition, J2EE）和微型版（Java 2 Micro Edition, J2ME），表明了 Java 开始向桌面应用、企业应用和移动设备应用三大领域渗透。

2000 年 5 月，JDK 1.3 和 JDK 1.4 先后发布。2001 年 9 月，J2EE 1.3 发布。2002 年 2 月，J2SE 1.4 发布，自此 Java 的计算能力有了大幅提升。2004 年 9 月，J2SE 1.5 发布，该版本扩充了大量的 API 类库并且引进了诸多新的特性，它的发布是 Java 语言发展史上的又一里程碑事件。为了表示这个版本的重要性，J2SE 1.5 更名为 J2SE 5.0。

2005 年 6 月，Sun 公司公布 Java SE 6，该版本主要的改进是性能的提升，并且强化了对各种脚本语言的支持。此时，Java 的各种版本已经更名，已取消其中的数字“2”，J2SE 更名为 Java SE，J2EE 更名为 Java EE，J2ME 更名为 Java ME。

2006 年 11 月，Sun 公司宣布 Java 全线采纳 GNU General Public License Version 2，从而开始有步骤地公开 Java 的源代码。2007 年 12 月，Sun 公司正式发布 Java 开源集成开发环境（Integrated Development Environment, IDE）——NetBeans 6.0。

2008 年 9 月，第一届 JVM 语言峰会的召开标志着 Java 从一个以语言为中心的技术正在转变

成为一个以平台为中心的技术，基于 JVM 的动态语言达 240 多种，这其中以 Jython、Ruby 的发展最为迅猛。

2009 年 4 月，Oracle 公司以 74 亿美元收购了 Java 的缔造者——Sun 公司。Oracle 公司负责产品开发的执行副总裁 Thomas Kurian 称 Java 为王冠上的一颗明珠，他示意将会增强和扩展 Java 编程模型，支撑各种新兴的应用开发模式。

### 1.1.2 Java 语言的应用

任何一种程序设计语言的诞生都有两个不可缺少的条件，一是行业的需要，二是语言本身的优势。Java 是一种基于 Internet 的网络程序设计语言，具有十分广阔的市场前景，它改变了传统的计算机应用方式，在安全性能和与硬件平台无关性等方面具有极大的优势。面向对象的编程环境、对多媒体信息的全面支持、多线程技术的使用都促进了 Java 语言的普及，目前它在各个重要的行业部门得到了广泛的应用，而且出现在各种各样的设备、计算机和网络中。从笔记本式计算机到数据中心，从游戏控制台到科学计算超级计算机，从手机到 Internet，Java 无处不在。

从权威的 Tiobe 全球程序设计语言使用和关注度排行来看，在最近 10 年的绝大部分时间里 Java 以较大优势位居所有程序设计语言的首位。2010 年 10 月份排行榜榜单中，Java、C、C++ 依然是前三甲（见 <http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html>）。

Java 语言有着广泛的应用前景，大体上可以分成以下几个方面：

- (1) 所有面向对象的应用程序开发，包括面向对象的事件描述、处理和综合等。
- (2) 计算过程可视化以及可操作化软件的开发。
- (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，包括选择交互、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) 网络通信与移动通信以及网络集成。

## 1.2 Java 语言的特点

Java 语言自产生以来，Sun 公司致力于将 Java 设计成为一种简单的、面向对象的、分布式的、解释执行的、健壮的、安全的、平台无关的、可移植的、高性能的、多线程的和动态的语言，正是这些特点，使得 Java 成为目前使用最为广泛的网络程序设计语言。

### 1. 简单

这里的简单当然不是功能上的简单，而是指 Java 语言在学习、使用和维护上比较方便。Java 的简单性是与 C/C++ 比较得出的结论，Java 是在 C/C++ 的基础上开发的，它继承了 C/C++ 的许多特性，但同时也舍弃了 C/C++ 中烦琐的、难以理解的或存在安全隐患的功能，比如指针、运算符重载和多重继承等，并对经常使用的操作进行了简化，比如字符串的操作和网络编程等。由于 Java 在设计时参考了许多 C/C++ 的语法和特性，使得无论是对程序语言的初学者还是对有一定基础的 C/C++ 程序员来说，Java 都是易学、易用和易维护的。

## 2. 面向对象

现实世界是由各种各样的实体（事物、对象）所组成的，每种对象都有自己的内部状态和运动规律，不同对象间的相互联系和相互作用就构成了各种不同的系统，并进而构成整个客观世界。人们为了更好地认识客观世界，把具有相似内部状态和运动规律的实体（事物、对象）综合在一起，称为类。类是具有相似内部状态和运动规律的实体的抽象，进而人们抽象地认为客观世界是由不同类的事物间相互联系和相互作用所构成的一个整体。计算机软件的目的就是为了模拟现实世界，使各种不同的现实世界系统在计算机中得以实现，进而为人们工作、学习、生活提供帮助。

面向对象（Object-Oriented, OO）其实是现实世界模型的自然延伸，它是按人们认识客观世界的系统思维方式，采用基于对象（实体）的概念建立模型、模拟客观世界分析、设计和实现软件的办法。通过面向对象的理论使计算机软件系统能与现实世界中的系统一一对应。现实世界中的对象均有属性和行为，反映到计算机程序上，属性则表示对象的数据，行为表示对象的方法（其作用是处理数据或同外界交互）。如果说传统的面向过程程序设计语言是以过程为中心、以算法为驱动的话，面向对象程序设计语言则是以对象为中心、以消息为驱动。用公式表示，面向过程程序设计语言为：程序=算法+数据；面向对象程序设计语言为：程序=对象+消息。

关于面向对象程序设计有以下基本概念：

（1）对象（Object）。对象是指现实世界中各种各样的实体，它可以指具体的事物，也可以指抽象的事物。例如，学生李木子、苹果、飞机、音乐会、窗体等。每个对象皆有自己的内部状态和运动规律，如学生李木子具有学号、姓名、性别、出生年月、籍贯等内部状态，具有吃饭、上课、打球、借书、演讲等运动规律。在面向对象概念中，把对象的内部状态称为属性，运动规律称为方法。

（2）类（Class）。类是具有相同属性和方法的对象的集合。例如，学生是一个类，而张三、李四、王五等一个个具体的学生，是人这个类的一个个对象。

（3）消息（Message）。消息是指对象间相互联系和相互作用的方式。一个消息主要由 5 部分组成：发送消息的对象、接收消息的对象、消息传递办法、消息内容（参数）和反馈。

（4）类的特性。类的定义决定了类具有以下 3 个基本特性：封装、继承和多态。

- 封装：面向对象的类是封装良好的模块，类定义将其说明（用户可见的外部接口）与实现（用户不可见的内部实现）显式地分开，其内部实现按其具体定义的作用域提供保护。类是封装的最基本单位。封装防止了程序相互依赖而带来的变动影响。在类中定义的接收对方消息的方法称为类的接口。
- 继承：继承是类不同抽象级别之间的关系。父类是子类更高级别的抽象。子类可以继承父类的所有内部状态和运动规律。在计算机软件开发中采用继承性，提供了类的规范的等级结构；通过类的继承关系，使公共的特性能够共享，提高了软件的重用性。
- 多态：多态性是指同种方法可在不同的类中具有不同的运动规律。不同的子类可以演绎出不同的运动规律。例如，动物都会奔跑，而羊和狼奔跑的速度和动作都不一样。

Java 语言是面向对象的程序设计语言，其中的绝大语法元素都是对象。另外，Java 与 C++ 相比取消了复杂的多重继承，取而代之的接口概念使得 Java 更加容易使用。



### 3. 分布式

分布式包括数据分布和操作分布。数据分布是指数据可以分散在网络的不同主机上，操作分布是指把一个计算分散在不同主机上处理。Java 支持 WWW 客户机/服务器 (Client/Server) 计算模式，因此，它支持这两种分布性。对于前者，Java 提供了一个称做 URL 的对象，利用这个对象，可以打开并访问具有相同 URL 地址的对象，访问方式与访问本地文件系统相同。对于后者，Java 的 Applet 小程序可以从服务器下载到客户端，即部分计算在客户端进行，提高了系统执行效率。Java 提供了一整套网络类库，开发人员可以利用类库进行网络程序设计，方便地实现 Java 的分布式特性。

### 4. 解释执行

Java 是解释执行的语言。高级语言程序必须转换为机器语言程序才能执行，但不同的计算机系统所使用的机器语言不同。Java 为了实现“编写一次，随处运行” (Write Once, Run Anywhere) 的目标，Java 的源程序在编译时，并不直接编译成特定的机器语言程序，而是编译成与机器结构无关的字节码 (Byte Code)，然后，这种字节码在任何 Java 的运行环境中由 Java 虚拟机 (JVM) 解释执行。

JVM 能直接在任何机器上执行，为字节码提供运行环境。当 JVM 解释执行 Java 程序时，Java 实时编译器 (Just In Time, JIT) 会将字节码译成目标平台对应的机器语言的指令代码。

### 5. 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。它提供自动垃圾回收机制 (Garbage Collection) 来进行内存管理，防止程序员在管理内存时容易产生的错误。通过集成的面向对象的异常 (Exception) 处理机制，在编译时，Java 提示出可能出现但未被处理的异常，帮助程序员正确地进行选择以防止系统崩溃。另外，在编译时 Java 还可捕获类型声明中许多常见的错误，防止动态运行时异常或错误不匹配问题的出现，保证程序更加可靠。

### 6. 安全性

由于 Java 主要用于网络应用程序开发，因此对安全性有较高的要求。如果没有安全保证，用户从网络下载程序执行就非常危险。Java 通过自己的安全机制防止了病毒程序的产生和下载程序对本地系统的威胁破坏。Java 的安全性可从两个方面得到保证：一方面，Java 不支持指针和内存释放等 C++ 所具有的功能，一切对内存的访问都必须通过对象的实例变量来实现，这样就避免了非法内存操作和指针操作时容易产生的错误。另一方面，类装载时可将本机类和网络资源类的名称分开，这样，调入类时总要经过检查，可以防止程序员使用“特洛伊”木马等欺骗手段访问对象的私有成员，从而保证了安全性。

### 7. 平台无关性

所有软件都有其运行的特定环境，即所谓的平台。平台无关有两种：源代码级和目标代码级。C/C++ 具有一定程度的源代码级平台无关，表明用 C/C++ 写的应用程序不用修改，只需重新编译即可在不同平台上运行，而 Java 的平台无关性是指用 Java 写的应用程序不用修改就可在不同的软硬件平台上运行，这主要靠 Java 虚拟机 (JVM) 在目标代码级实现平台无关性。JVM 是一种抽象机器，它附着在具体操作系统之上，本身具有一套虚拟机器指令，并有自己的栈、寄存器组等。JVM 是 Java 平台无关的基础，也是 Java 程序可移植性的保证，在 JVM 上，有一个 Java 解释器用来解释 Java 编译器编译后的程序。Java 编程人员在编写完软件后，通过 Java 编译器将 Java 源程序编