21世纪高等职业教育计算机系列规划教材

Java程序设计教程

刘甫迎 主 编
郑显举 陈振梁 樊婷婷 编 著

- 把课程教学目标纳入专业技能培养目标中，目标具体明确
- 本书突出实践动手能力和实用性，突出案例，并配有实验指导书、习题、模拟试题

配备课件

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

21 世纪高等职业教育计算机系列规划教材

Java 程序设计教程

刘甫迎　主　编

郑显举　陈振梁　樊婷婷　编　著

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

Java 程序设计语言是由 Sun 公司开发的面向对象的语言，其既可开发一般的商务程序，又可用于 Web 编程，是 Java EE 企业级编程的语言基础。可以编译跨平台、跨语言的代码，去掉了 C 语言中的指针和多继承，简单易学且功能强大，越来越受到人们的青睐。

本书有 15 章和 3 个附录。详述了 Java 及其开发环境，Java 的基本数据，方法、类和对象，数组和结构，继承、重载、接口与多态，异常处理，图形用户界面（GUI），程序设计（Swing、AWT），文件的输入和输出，多线程，数据库操作，Applet，Java 的多媒体技术等。本书一开始便是以面向对象程序设计来展开的，内容实用，注重对读者编程能力的培养。

本书有案例、习题、教学大纲、实验指导书和模拟试题，便于学习与教学。可作为高等学校及软件学院的教材，也适于从事软件开发和应用的人员参考。

图书在版编目（CIP）数据

Java 程序设计教程/刘甫迎主编. —北京：电子工业出版社，2010.12
（21 世纪高等职业教育计算机系列规划教材）
ISBN 978-7-121-12087-9

Ⅰ. ①J… Ⅱ. ①刘… Ⅲ. ①JAVA 语言－程序设计－高等学校：技术学校－教材 Ⅳ. ①TP312

中国版本图书馆 CIP 数据核字（2010）第 207881 号

策划编辑：徐建军
责任编辑：徐建军　特约编辑：方红琴
印　　刷：北京市李史山胶印厂
装　　订：
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　邮编 100036
开　　本：787×1 092　1/16　印张：23.5　字数：602 千字
印　　次：2010 年 12 月第 1 次印刷
印　　数：4 000 册　定价：38.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

前　言

Java 程序设计语言是由 Sun Microsystems 公司开发的一种面向对象的语言，它的诞生给 IT 产业带来了一次变革，从某些意义上讲对人们的日常生活也产生了深远的影响。Java 的结构类似 C/C++，不仅用于开发一般的商务程序，而且与互联网发展紧密结合，已成为互联网和计算机应用的主流，它是 Java EE 企业级编程的语言基础。

近年来，Java 程序设计语言之所以如此流行是因为它具有以下优势：安全特性和平台无关性，这意味着用 Java 程序设计语言编写的程序可以在任何一个平台上运行。它去掉了 C 语言中的指针和多继承，简单易学且功能强大，越来越多地受到人们的青睐。

目前，国内高校讲授该类课程大多采用传统的教学内容和教学方式，导致学生的编程设计和应用能力不够。教育部新近启动实施了“卓越工程师教育培养计划”，提出行业企业深度参与培养过程、学校按通用标准和行业标准培养工程人才、强化培养学生的工程能力和创新能力。笔者认为讲授此课程主要应培养学生基于工程的 Java 项目开发的实践动手能力，分析问题、解决问题的能力。很有必要改变课程的教学内容和教学方式，用先进的教学理念和方法培养一流人才。本教材就是在此改革的思路下编写的，其特点是：

（1）把课程教学目标纳入专业技能培养目标中，目标具体明确，学生学习兴趣大。例如，对“计算机软件”等专业来说，《Java 程序设计》课程是该专业的一门专业基础课，本书重点对本专业分解到该课程的结构化程序设计、面向对象程序设计、C/S 模式编程、图形用户界面（GUI）程序设计等能力模块进行教学。

（2）本书一开始便是用 Java 的面向对象的程序来展开的，便于学生们学习。

（3）选材上，讲述的理论“以必需够用为度”，减轻学生负担。另外，书中例子使用了许多经典算法，弥补了有些读者数据结构知识不足的弱点。

（4）突出实践动手能力和实用性，突出案例（有各章例子及综合实例等），配有实验指导书、习题、模拟试题、教学大纲等便于学习与教学，力图使学生学习本书后便基本可以编制应用程序。

本书可作为高等院校及软件学院的教材，也适于从事软件开发和应用的人员参考。本书由刘甫迎、郑显举、陈振梁、樊婷婷编著。刘甫迎编写第 1 章、第 2 章、第 8 章、第 9 章和附录 A、附录 B；郑显举编写第 6 章、第 7 章、第 10 章、第 12 章、第 15 章和附录 C；陈振梁编写第 3 章、第 4 章和第 11 章；樊婷婷编写第 5 章、第 13 章和第 14 章。全书由刘甫迎教授统稿，谢春、李朝蓉、杨雅志、刘焱、党晋蓉、李琦、王蓉、龚茗茗、饶斌等做了不少工作。在编写和出版的过程中，电子工业出版社的编辑给予了很大的帮助，在此表示感谢。

为了方便教师教学，本书配有电子教学课件，请有此需要的教师登录华信教育资源网（www.hxedu.com.cn）免费注册后进行下载，有问题时请在网站留言板留言或与电子工业出版社联系（E-mail:hxedu@phei.com.cn）。

由于编者水平有限和时间仓促，书中难免存在疏漏之处，欢迎广大读者批评指正。

编　者

目　录

第1章 Java及其开发环境

本章介绍什么是程序设计（结构化程序设计和面向对象程序设计）与Java语言，如何编写、修改和运行一个Java程序，以及Java开发环境等，以便对Java及其开发、运行环境有一个基本的了解。

1.1 程序设计与Java语言

1.1.1 什么是程序设计

一个计算机程序只不过是一组人为编写的用来告诉计算机做什么的指令集合。计算机是由很多小的开关组成的电路构建而成的，所以可以顺着下面的命令行来编写计算机程序：

第一个开关处于开状态
第二个开关处于关状态
第三个开关处于关状态
第四个开关处于开状态

程序可以为几千个开关像这样不断地写下去，这种模式的程序是用机器语言编写的，机器语言是最基本的电路语言。用这种语言编写程序的问题在于：在编写任何一个有价值的任务时需要明白开关当前的状态及当程序不能按照预期效果运行时，如何去发现那些出错的开关。另外，不同的计算机的开关数量和位置是不同的，这就意味着需要为每种想在它上面运行程序的机器定制一种机器语言。

幸运的是，由于高级程序设计语言的发展，编程已经变得较容易了。高级程序设计语言提供了如“read”，“write”和“add”等易于理解的术语，取代了完成这些任务所需的开关的一系列组合。高级程序设计语言还允许定义直观的计算机存储单元的名字，像“hoursWorked”，“rateOfPay”，而不是必须记住那些值的存储单元（开关号）。

每一种高级程序设计语言都有它自己的语法，或是语言的规则。例如：取决于特定的高级语言，可能会用动词“print”或“write”来产生输出。所有的语言都有一个特定的有限词汇表及使用那个词汇表的一组特定的规则。程序员使用一个叫编译器的计算机程序把其用高级语言编写的语句翻译为机器代码。每当程序员错误使用编程语言时，编译器就会发出一个错误提示信息；接着，程序员纠正这个错误并且尝试再次编译该程序。学习Java，C++或C#这样的程序设计语言时，实际上是在学习那种语言的词汇表和语法规则。

除了要学习一种特殊语言的正确语法之外，程序员还必须理解计算机程序设计逻辑。任何一个程序背后的逻辑都包括了以正确的顺序执行各种各样的语句和方法来产生所期望得到的结果。比如：你可能会在一个乐器上弹奏优秀的乐章，但是如果不能按正确的顺序来弹奏它们（当期望F高调时却弹成B降半音），那么就不能弹出悦耳的音乐。类似地，你可能能够正确地使用计算机语言的语法，但是程序的逻辑结构不正确。逻辑错误的例子包括了当你打算把两个值相除时却把它们相乘了，或者在获得适当的输入之前就产生了输出。

因此，程序设计的主要步骤是：

（1）分析问题。给定的有哪些数据，需要输出什么样的数据，需要进行哪种处理，即分析用户需求。需要用到哪些硬件和软件，即分析运行环境，进行可行性分析，做到心中大体有数。

（2）确定算法。算法是解题的过程。首先需要将一个物理过程或工作状态用数学形式表达出来，即确定解题最合适的过程，或确定合适的处理方案。对同一个问题处理方案的不同，决定了不同的处理步骤，效率则不同。

（3）画出程序流程图。用规定的基本图形来描述解题步骤。它表达了算法，是编写程序的依据。

（4）编写程序。根据流程图表达的步骤，用程序设计语言逐句、逐行地写出程序。

（5）调试程序。主要包括排错和测试两部分，排错是指查出在程序执行过程中出现的语法错误和逻辑错误，并加以改正；测试是指确认程序在各种可能的情况下正确、可靠地运行，输出结果准确无误。排错和测试常常是交叉进行的，直到结果满意为止。

（6）建立健全文档资料。文档资料是计算机软件的重要组成部分，从接受用计算机解题任务开始就应注意加强文档资料的建立，解题任务完成时，文档资料也应建立完毕。

上面介绍的 6 个步骤中，最关键的是第 2 个步骤，即“算法设计”。只要算法是正确的，写程序就不会有太多困难。一个程序员应该掌握如何设计一个问题的算法或者采用已有的现成算法。

所谓“算法”，粗略地讲，是为解决一个特定问题而采取的确定的有限步骤。广义地说，做任何事都需要先确定算法，然后去实现这个算法以达目的。我们这里所说的算法，是指计算机能执行的算法。

计算机算法可以用流程图来表示。

1.1.2　面向对象程序设计

编写计算机程序有两种常用的方法：面向过程的程序设计和面向对象的程序设计。

面向过程的程序设计（或者叫结构化程序设计）包括用程序语言建立存放值的存储单元并且编写对那些值进行运算的一系列步骤或操作。计算机存储单元被称为变量，因为它们所保存的值可以变化。例如：某公司的工资程序中有变量 rateOfPay ，这个变量存放的存储器单元在不同时间内可以有不同的值（公司中每个员工对应不同的值）。当执行工资程序时，对存储在 rateOfPay 中的值可对应多种操作，比如，从输入设备中输入该值，与表示时间的变量相乘，在打印纸上输出。为方便起见，一个计算机程序的各个操作通常被组合成逻辑单元，称为过程。比如，把确定个人所得税的四到五步之比较和计算可以合成一个过程，称为 caculateFederalWithholding 。面向过程的程序定义了可变的存储单元，然后调用或引用一些过程对这些单元中的值进行输入、操作和输出。一个面向过程的程序通常包括成百上千的变量和过程调用。

为了提高程序的易读性（容易理解），保证程序质量，降低软件成本，荷兰学者 Dijkctra 等提出了“结构化程序设计”的方法。“结构化程序设计”的要点是：

（1）程序的质量标准是“清晰第一，效率第二”。

（2）要求程序设计者按一定规范书写。

（3）规定用三种基本结构来组成程序（顺序结构、选择结构、循环结构）。

（4）自顶向下、逐步细化、模块化的程序设计方针。

一个综合型的复杂软件，被分为若干个相对独立的部分，每个部分又称为一个模块。根据

具体情况，各个模块还可以再细分为几个小部分，每个小部分又称为一个子模块。这样一层层分解，逐步细化，使整个程序结构层次清晰，各个子模块任务简单明确。软件编写人员可以按模块进行分工合作，互相之间容易接口，不易发生遗漏，给编写程序和调试程序都带来方便。特别是在查找程序错误方面，缩小了范围，节省了时间和精力。

面向对象程序设计是面向过程程序设计（结构化程序设计）的一种扩展，在编写程序时采用的方法有一些不同。用面向对象的方法考虑问题时首先把程序元素看成是与现实世界中的具体对象相似的对象，然后对这些对象进行操作以得到期望的结果。编写面向对象的程序包括创建对象和创建使用这些对象的应用程序。

如果你曾经使用过命令行操作系统（如 DOS）的计算机，也使用过图形用户界面的操作系统（如 Windows）的计算机，那么你已经了解了面向过程的程序和面向对象的程序的区别。如果想将几个文件从软盘复制到硬盘，可以在提示符或命令行输入命令，或者是在一个图形环境下用鼠标来完成这个任务。两者的区别在于是否用一系列的命令按顺序移动 3 个文件，或者是拖动表示文件的图标从屏幕上一个位置移到另一个位置，这如同在办公室里将文件从一个档案柜拿到另一个档案柜一样。两者之中的任何一种操作系统都能移动相同的 3 个文件，但图形用户界面系统允许你像对现实世界中的文件复制品那样对文件进行操作。换句话说，图形用户界面系统允许把文件当做对象处理。

在现实世界和面向对象程序设计中的对象都由状态和方法组成。一个对象的状态也称为属性。比如，汽车有一些属性是制造单位、型号、生产日期和购买价格，另一些属性包括汽车目前是否在行驶、车速和是否干净。所有汽车都有相同的属性，当然这些属性值不会相同。类似的，你的狗有品种、名字、年龄、杂色是否流行等属性。你的红色 Chevrolet 汽车是所有汽车类中的一个实例。你的名叫 Goldie 的金黄色的狗是所有狗类中的一个实例。把某些项目看成是类的实例允许你把类的常识运用到类的个体成员中，这些对象的特定实例继承了一般化类的属性。如果你的朋友购买了一辆汽车，你知道了它的型号，如果你的朋友喂养了一只狗，你知道了狗的品种。你可能不知道你朋友的汽车的速度或狗的杂色的精确内容或当前的状态，但你确实知道这一类汽车和这类狗的属性。同样，在图形用户操作环境中，你希望每一个组件都有特殊的、一致的属性，如相同的菜单栏和标题栏，因为每一个组件都继承了图形用户界面组件通用类的成员属性。

注意：按规定，程序员使用 Java 程序语言定义类名以大写字母开头。因此定义一个汽车的属性和方法的类就可能被命名为 Automobile，并且“狗”类可能就被命名为 Dog。然而，生成一个可行的程序并不要求遵循这个规定。

除属性外，对象能使用方法来完成任务。例如：汽车能够向前行驶和向后倒车、能装满汽油或被清洗，这些都是改变它们某些属性的方法。方法是为了弄清某些属性，如汽车目前的速度和汽油缸当前的状态。类似的，一条狗能走或者跑，吃东西和洗澡，并且也有判断这只狗有多饿的一些方法。图形用户界面操作系统组件能被最大化、最小化和拖动。与面向过程的程序一样，面向对象程序有变量（属性）和过程（方法），但是这些属性和方法是被封装在对象中的。封装是一种技术，把对象属性整合成一个互相依赖的单元，从而作为一个不可分割的整体使用。程序员有时把封装看成是一个“黑盒”，或者可以使用的设备，而不用考虑其内部的装置。

如果一个对象的方法编写好了，用户可以不知道其如何执行的底层细节。在这种情况下，用户必须简单理解方法与对象间的接口或交互作用。例如：能把汽车的汽油加满，是因为你知

道汽油泵塞和汽车缸之间的接口如何打开，而不必了解泵如何工作或者汽油缸实际装在汽车的哪个部位。如果你能读取速度表，而显示的数字是如何计算的就不重要了。事实上，如果生产了更先进、精确的速度测定设备，并安装到汽车中，只要它与原来的设备接口相同，你不必了解或关心它如何操作。同样的道理可应用到面向对象程序设计中的创建对象上。

1.1.3　Java 的由来和发展

Java 语言是目前流行的一种编程语言，它的面向对象、跨平台、健壮安全、多线程和分布应用等特点给编程人员带来一种崭新的计算概念，使 WWW 由最初的单纯提供静态信息发展到现在的提供各种各样的动态服务。Java 不仅能够编写嵌入网页中具有声音和动画功能的小应用程序，而且还能够应用于独立的大中型应用程序，其强大的网络功能可能把整个 Internet 作为一个统一的运行平台，极大地拓展了传统单机或 Client/Server 模式应用程序的外延和内涵。从 1995 年正式问世以来，Java 逐步从一种单纯的高级编程语言发展为一种重要的 Internet 开发平台，并进而引发带动了 Java 产业的发展和壮大，成为当今计算机业界不可忽视的力量和重要的发展潮流与方向。

Java 的开发环境有不同的版本，如 Sun 公司的 Java Developers Kit，简称 JDK。后来微软公司推出了支持 Java 规范的 Microsoft Visual J++ Java 开发环境，简称 VJ++。

1. Java 语言的起源

当 1995 年 Sun 推出 Java 语言之后，全世界的目光都被这个神奇的语言所吸引。那么 Java 到底有何神奇之处呢？

最早 Java 语言的出现是源于独立开发平台语言的需要，当时人们希望能编写出嵌入到各种家用电器等设备的芯片上，且易于维护的程序。它的出现是为了弥补当时的编程语言，如 C、C++等只能对特定的 CPU 芯片进行编译的缺陷。Java 的设计者们就大胆设想让更换芯片的电器还是能够正确运行，无须重新编译芯片，因此 Sun 公司于 1990 年成立了由 James Gosling 领导的开发小组，开始致力于开发一种可移植的、跨平台的语言，该语言能生成正确运行于各种操作系统、各种 CPU 芯片上的代码。经过他们的精心钻研和努力，1991 年促成了 Java 语言的前身、起初被称为 OAK 的语言诞生。在网络出现之前，OAK 可以说是默默无闻，甚至差点夭折。但是，网络的出现改变了 OAK 的命运。

在 Java 出现以前。Internet 上的信息内容都是一些乏味死板的 HTML 文档。这对于那些迷恋于 Web 浏览的人们来说是不可容忍的。他们迫切希望能在 Web 中看到一些交互式的内容，开发人员也极希望能够在 Web 上创建一类无须考虑软硬件平台就可以执行的应用程序，当然这些程序还要有极大的安全保障。对于用户的这种要求，传统的编程语言显得无能为力，而 Sun 的工程师敏锐地察觉到了这一点，从 1994 年起，他们开始将 OAK 技术应用于 Web 上，并且开发出了 HotJava 的第一个版本。当 Sun 公司 1995 年正式以 Java 这个名字推出时，几乎所有的 Web 开发人员都意识到：这正是自己想要的，Java 成了一颗耀眼的明星。

2. Java 语言的发展历程

Java 语言（简称 Java）的诞生对 IT 产业带来了一次变革，从某些意义上讲对人们的日常生活也产生了深远的影响。Java 的结构虽类似 C/C++，但因其平台无关性和与互联网发展紧密结合，预计未来必定成为互联网和计算机应用的主流。Java 当之无愧地被纽约时报评为 1995 年的十大科技成果之一，并将 Java 作为一项重大发明载入科技史册。微软总裁比尔·盖茨曾在观察了一段时间后，十分惭愧地说："Java 是长时间以来最卓越的程序设计语言"，并确定微

软整个软件开发的战略从 PC 时代向着以网络为中心的时代转移，而购买 Java 则是他的重大战略决策的实施部署。当然微软与 Sun 也曾为纯 Java 对峙法庭，微软必将直接或间接将 Java 技术融入其产品体系中。Sun Microsoft 公司的总裁 Scott McNealy 认为 Java 为 Internet 和 WWW 开辟了一个崭新的时代。环球信息网 WWW 的创始人 Berners-Lee 说：计算机事业发展的下一个浪潮就是 Java，并且将很快发生。所以，使用 Java 已成大势所趋。

Microsoft 和 IBM 两大公司都在 Internet 上销售用 Java 编写的软件，IBM 著名 Java 开发集成环境 Visual Age For Java、网站集成平台 Webspere，Microsoft 的 VJ++都是目前主要常见的 Java 开发环境或产品。另外，Apple、HP、IBM、Microsoft、Novell、SGI、SCO、Tandem 等公司均在各自开发的操作系统中加入了 Java 开发运行环境，而负责开发并推广 Java 技术的 Javasoft 公司（这是 Sun 下属的一个子公司），通过颁发许可证的办法来允许各家公司把 Java 虚拟机和 Java 的 Applets 类库嵌入他们开发的操作系统，这样各类开发人员就能更容易地选择多种平台来使用 Java 语言编程，不同的用户也就可以脱离 Web 浏览器来运行 Java 应用程序，这无疑是很受广大用户欢迎的，也为 Java 语言的应用开拓了极为广阔的前景。

1996 年 6 月 7 日，由 Sun 公司和台湾经济事务部信息发展局、台湾信息技术研究所等单位牵头，成立了一个“台湾 Java 联盟”，有多个台湾著名计算机公司参与，并在台北建立“Java 开发中心”，在新竹建立“Java 语言实验室”，以掀起台湾开发与应用 Java 语言的热潮。香港则在 1996 年 4 月就举行了全岛的 Java 杯比赛，在计算机界掀起了学习 Java 的热潮（尤其是在大学生中，出现了一批 Java 迷）。1996—1998 年，Sun 公司与国内大陆的清华大学、北京大学的著名高校合作，成立 10 个“Java 教育中心”；与中软、邮电部等机构合作成立培训中心、开发中心；在北京、天津和上海成立研究开发基地。每年一度的 JavaOne 大会几乎都会推出让 Java 更受人欢迎的新技术，曾经有人预言：Java 将是网络上的“世界语”，将会被未来所证实。

1998 年，由于 Java 在安全、性能等方面到了一个关键阶段，甚至 Java 被列入了 20 世纪末十大必然死亡的技术之一，十大科技成果被预言必将死亡实在是一个幽默。1998 至今，随着 Java2 一系列新技术如 JAVA2D、JAVA3D、SWING、JAVA SOUND、EJB、SERVLET、JSP、CORBA、XML、JNDI、J2EE、Java EE 等的引入，以及 JVM 自身的安全策略完善、效率提高，新一轮 Java 热潮再次被掀起。目前，国内电子商务、金融、证券、邮电、电信等行业的大部分系统几乎都正在或者准备采用部分 Java 技术来实现。此外国内也出现了致力于 Java 技术推广和开发的非盈利组织——中国“Java 阵线联盟”（javaunion.org）及一批作为该组织成员的个人 Java 技术网站，预计一两年内 Java 应用将被国内广大行业用户所接受，Java 技术水平也将逐步与世界同步。

3. Java 语言的发展前景

在 2005 年的 Java One 开发者大会上，James Gosling 做了题为“Java 技术下一个 10 年贡献”的演讲，James Gosling 认为，Java 技术提高了计算的“流动性”，就如同货币的发明提高了商品的流动性一样。无所不在的网络丰富了每个人的信息量，就如同可以兑换的货币产生了财富一样。由于从前的网络速度是很慢的，所以计算被束缚在特定的计算机上，而这种情况将一去不复返了。

目前，全球 Java 开发人员已经超过 500 万，因此 Java 社区是一个充满活力和创新精神的团队，这正是 Java 更加繁荣的保障。为了保持 Java 的增长和推进 Java 社区的参与，Sun 在 Java One 开发者大会上宣布开放 Java 核心源代码，以鼓励更多的人参与到社团活动中来，这是 Sun

为推进社团发展和维护 Java 技术兼容性而迈出的重要一步，同时也是 Java 技术在创新和社会进步上继续发挥重要作用的标志。

随着 Java 的开源，在未来的十年里，Java 的应用范围将变得更广。数字媒体将是 Java 的下一个目标，同时，Java 将教育和健康作为未来 Java 发展过程中的两大重点应用领域。

1.1.4　Java 程序设计语言的特点

Java 语言具有能独立于平台而运行、面向对象、可对动态画面进行设计与操作、坚固性等特点，又具有多线程、内置校验器用来防止病毒入侵等功能，所以用来在 Internet 上研制与开发软件时，特别受到用户的欢迎。

Java 语言的优点主要表现在：简单、面向对象、自动的内存管理、分布计算、稳定、安全、解释执行、结构中立、平滑移植、多线程，以及异常处理等方面。

1. 简单

由于 Java 的结构类似于 C 和 C++，所以一般熟悉 C 与 C++语言的编程人员稍加学习就不难掌握 Java 的编程技术了。并且 Java 所具有的自动内存管理机制也大大简化了 Java 程序设计开发。Java 程序设计语言模仿了 C++程序设计语言。尽管两种语言在首次推出时都认为不易于阅读或理解，但 Java 语言去除了一些在 C++理解上最难的特性。对于 C++程序员来说，指针和多重继承是两个最难掌握的特性。Java 程序设计语言除去了 C++语言中这些令人感到麻烦的特性。

2. 面向对象

Java 的面向对象机制实际上可以看做是 C++面向对象机制的延伸。Java 提供了简单的类机制和动态的构架模型，对象中封装了它的状态变量和方法（函数、过程），实现了模块化和信息隐藏；而类则提供了一类对象的原型，通过继承和重载机制，子类可以使用或者重新定义父类或者超类所提供的过程，从而实现代码的复用。

3. 自动内存管理

Java 的自动无用内存回收集（Auto Garbage Collectino）实现了内存的自动管理，因此简化了 Java 程序开发的工作，早期的 GC（Garbage Collectino）对系统资源抢占太多而影响整个系统的运行，Java2 对 GC 进行的改良使 Java 的效率有了很大提高。GC 的工作机制是周期性地自动回收无用存储单元。Java 的自动内存回收机制简化程序开发的同时，提高了程序的稳定性和可靠性。

4. 分布计算

Java 为程序开发提供了 Java.net 包，该包提供了一组使程序开发者可以轻易实现基于 TCP/IP 的分布式应用系统。此外，Java 还提供了专门针对互联网应用的类库，如 URL、Java mail 等。

5. 稳定性

人们最常见的应用程序错误就是“非法访问 xxx 内存”，其实质是程序指针使用出错。Java 拥有一种指针（Pointer）模型，能够排除发生内存被覆盖和毁损数据的可能性。Java 不采用指针算术法，而是提供真正的数组（Array），运行程序下标检查；另外，它也不会发生有对象类型转换将一个任意数转换成指针的情形。Java 的自动内存管理在减少编程工作的同时，大大减少了运行态错误。

6．安全性

Java 的设计目的是提供一个用于网络/分布式的计算环境。因此，Java 强调安全性，如确保无病毒、小应用程序运行安全控制等。Java 的验证技术是以公钥（Public-key）加密算法为基础，而且从环境变量、类加载器、文件系统、网络资源和名字空间等方面实施安全策略。

7．解释执行

Java 解释器（Interpreter）可以直接在任何已移植的解释器的机器上解释、执行 Java 字节代码，不需重新编译。当然，其版本向上兼容，因此如果是高版本环境下编译的 Java 字节码，在低版本环境下运行时也许会有部分问题。

8．跨异构环境

Java 是网络空间的“世界语”，编译后的 Java 字节码是一种“结构中立性（Architecture Neutral）的目标文件格式，可以在所有提供 Java 虚拟机（JVM）的多种不同主机、不同处理器上运行。具有平台无关性，这意味着用 Java 程序设计语言编写的程序可以在任何一个平台上运行。运行 Java 程序的机器只需要有一个称为解释器的特殊程序来为主机翻译程序。用 Java 编写的程序被编译成 Java 虚拟机代码，称为字节代码。编译后的字节代码最后在执行程序的机器上被解释。任何编译后的程序都可以在带有 Java 语言解释器的任何机器上运行。

9．平滑移植

“Write once,run every where!”也许是 Java 最诱人的特点。用 Java 开发而成的系统的移植工作几乎为零，一般情况下只需对配置文件、批处理文件作相应修改即可实现平滑移植。相比之下，当使用其他的程序设计语言时，软件的开发商通常要生产同一产品的多种版本（DOS、Windows 98、Windows 2000 和 Windows XP 等版本），这样所有的用户才能使用这个程序。有了 Java 程序语言，一个程序版本可以在所有这些平台上运行。

10．多线程

Java 的多线程（Multithreading）机制使程序可以并行运行。Java 还有一组同步化基本单元，它们是以广泛使用的 C.A.R.Hoare 监视器与条件变量图为基础的。同步机制保证了对共享数据的正确操作。多线程使程序设计者可以用不同的线程分别实现各种不同的行为，而不需要采用全局的事件循环机制，因此，使用 Java 语言可以非常轻松地实现网络上的实时交互行为。

11．异常处理

C 语言程序员大都有使用 goto 语句来做条件跳转，Java 编程中不支持 goto 语句。Java 采用异常模型使程序的主流逻辑变得更加清晰明了，并且能够简化错误处理工作。

12．可扩充

Java 目前发布的 Java EE 标准主要为采用 Java 技术为企业提供全面解决方案提供了一个技术规范框架，规划了一个利用现有和未来各种 Java 技术整合解决企业应用远景蓝图。

Java 语言的特点，对软件开发技术有巨大的影响。曾有人预言：“Java 语言的出现，将会引起一场软件革命”，这是因为 Java 语言能在执行码（二进制码）上兼容，这样以前所开发的软件就能运行在不同的机器上，只要所用的机器能提供 Java 语言解释器即可。时至今日，Java 的优势已经不再仅限于跨平台和动态页面显示，更多的优势逐步在支持 Web 服务器后端处理及应用整合、图形处理与多媒体等方面得到体现。Java2 及以后版本提供了 Swing、2D、3D、Sound 等图形、图像和多媒体支持类库，弥补了早期 Java 版本在此方面的不足。Java 的媒体处理能力使程序开发的动画效果远比 GUI 技术更加逼真，尤其是利用 WWW 提供的巨大动画资源空间，可以共享全世界的动态画面的资源。