

中国计算机学会教育专业委员会 推荐
全国高等学校计算机教育研究会 出版
高等学校规划教材

Java 语言程序设计 习题解答与实践教程

姜志强 编著

计 算 机 学 科 教 学 计 划



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

高等学校规划教材

Java 语言程序设计 习题解答与实践教程

姜志强 编著



电子工业出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

本书是普通高等教育“十一五”国家级规划教材《Java 语言程序设计》的配套教材。全书共分 5 章，第 1 章为 Java 语言学习指导，主要为 Java 语言初学者进行 Java 语言学习的方向性、方法性指导；第 2 章为 Java 语言常见问题解答，集中解答 Java 语言学习者在学习过程中遇到的一些容易混淆的概念性问题和教学过程中的常见问题；第 3 章为习题解答，给出主教材各章所配习题的参考答案；第 4 章为实验教程，提供了一些实验题目，供从事 Java 语言教学的教师在教学过程中选用；第 5 章为课程设计与综合应用，给出了几个有一定难度的综合性程序设计题目，供从事 Java 语言教学的教师在教学过程中作为课程设计的内容使用，也可供对 Java 语言程序设计有比较浓厚兴趣的读者使用。

本书与主教材配套，适合作为普通高等院校计算机及相关专业 Java 课程的教材，也可供其他专业的本科生、研究生及计算机专业的技术人员参考。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有，侵权必究。

图书在版编目 (CIP) 数据

Java 语言程序设计习题解答与实践教程/姜志强编著. —北京：电子工业出版社，2009.3

高等学校规划教材

ISBN 978-7-121-08214-6

I. J… II. 姜… III. JAVA 语言—程序设计—高等学校—教学参考资料 IV. TP312

中国版本图书馆 CIP 数据核字 (2009) 第 013189 号

策划编辑：童占梅

责任编辑：余 义

印 刷：北京市海淀区四季青印刷厂

装 订：北京鼎盛东极装订有限公司

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：787×1092 1/16 印张：11.75 字数：301 千字

印 次：2009 年 3 月第 1 次印刷

印 数：4000 册 定价：19.00 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 zltz@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

本书是普通高等教育“十一五”国家级规划教材《Java 语言程序设计》的配套教材，其编写目标是为选用该书作为“Java 语言程序设计”课程教材的广大师生提供一个教学辅助工具，以便任课教师能更好地使用教材，并且在此基础上，更方便地完成实验教学和课后教学等环节的工作。

全书共分 5 章，第 1 章为 Java 语言学习指导，其中的内容是为 Java 语言初学者提供一个指导，以便于学生把握学习方向，安排学习内容；第 2 章为 Java 语言常见问题解答，其中的内容集中回答了在 Java 语言程序设计教学过程中学生经常遇到的比较频繁出现的问题；第 3 章为习题解答，给出了主教材各章后面所配全部习题的参考答案，是主教材习题的集中解答；第 4 章为实验教程，提供了实验课程的教程；第 5 章为课程设计与综合应用，给出了几个课程设计方面的题目，供对 Java 语言有更浓厚兴趣的读者在程序设计练习时使用。

关于第 4 章和第 5 章的内容还要进行如下说明。

第 4 章实验教程的内容是为配合主教材的讲授进度而设计的，各个实验都与主教材的有关章节直接对应，所以在安排这些实验内容的时候，最好能够配合课堂讲授的进度。每个实验的内容大致需要用 2 学时的时间，所以建议安排 24 个学时完成这些实验内容。在设计这些内容的时候，编者依然秉承《Java 语言程序设计》一书的指导思想，坚持强调要学习和掌握 Java 语言 API 文档，所以在每个实验中都安排了阅读 API 文档的实验内容。编者坚定地认为，了解和掌握 API 文档，了解其中对各种 Java 类及每个 Java 类中的字段和方法的定义，对于 Java 语言的学习者而言是十分重要的，其重要性甚至不亚于了解 Java 语言的基本语法。另外，编者没有为《Java 语言程序设计》一书第 10 章到第 13 章的内容给出配套的实验，这是因为编者认为 JDBC、Servlet、JSP、JavaBean 等内容本来是 J2EE 的组成内容，通常情况下都是在一个比较大的 J2EE 项目中才能够完整地应用实施这些内容，并且在一个项目中这些内容都是综合应用、协同工作的，所以在一般情况下，实验课堂的课时难以完成这样的大题目。编者建议对这部分内容比较感兴趣的读者，在学习完 J2EE 的内容之后再着手进行有关的程序设计。

第 5 章中编者设计了几个具有一定难度和综合性的题目，其目的是为那些对 Java 语言程序设计有比较浓厚的学习兴趣的学生提供一个深入学习和综合运用的锻炼机会，可利用这几个题目编写工作量比较大一点、难度比较高一点的程序。不过，由于这几个题目难度比较大，需要的时间也比较多，通常并不是每个学生都能够达到这样的要求，所以编者只希望将其作为选学内容，而不应该将其纳入日常教学的要求中。

本书的内容大多是在编者多年教学工作过程中积累下来的教案和设计题目的基础上整理编写而成的，并不一定就适合其他各个学校的课堂教学和实验教学的实际使用要求。所以，编者建议使用《Java 语言程序设计》一书和本书作为教材的教师，不必拘泥于本书的章节，在使用过程中一定要密切结合本校的实际情况，根据教学需要、学生的学习基础和接受能力等实际情况，对这些教学内容进行相应的取舍，把本书作为安排教学内容的参考书，在此基础上设计出适合自身教学要求的教学内容来。

编者真诚地欢迎广大师生将教学和学习过程中使用本书的感受、心得体会、需求等告知编者，也希望把好的意见和建议与作者进行交流，以便编者在今后的工作中对本书进行进一步的改进和完善。由于写作时间仓促和编者水平有限，书中难免会存在一些缺点和错误，恳请专家和读者批评指正。

编者电子邮件: javajiangzhiqiang@163.com。

编 者

2008年10月于吉林大学

反侵权盗版声明

电子工业出版社依法对本作品享有专有出版权。任何未经权利人书面许可，复制、销售或通过信息网络传播本作品的行为；歪曲、篡改、剽窃本作品的行为，均违反《中华人民共和国著作权法》，其行为人应承担相应的民事责任和行政责任，构成犯罪的，将被依法追究刑事责任。

为了维护市场秩序，保护权利人的合法权益，我社将依法查处和打击侵权盗版的单位和个人。欢迎社会各界人士积极举报侵权盗版行为，本社将奖励举报有功人员，并保证举报人的信息不被泄露。

举报电话：(010) 88254396；(010) 88258888

传 真：(010) 88254397

E-mail : dbqq@phei.com.cn

通信地址：北京市万寿路 173 信箱

电子工业出版社总编办公室

邮 编：100036

目 录

第 1 章	Java 语言学习指导	(1)
1.1	为什么要学习 Java 语言?	(1)
1.2	学习 Java 语言应该抓住哪些关键点?	(2)
1.3	学习 Java 语言的基本步骤有哪些?	(5)
第 2 章	Java 语言常见问题解答	(7)
2.1	基本语法概念问题	(7)
2.2	面向对象程序设计、图形用户界面概念问题	(10)
2.3	异常处理、输入/输出、多线程概念问题	(14)
2.4	J2EE 有关概念问题	(17)
第 3 章	习题解答	(19)
3.1	第 0 章习题解答	(19)
3.2	第 1 章习题解答	(20)
3.3	第 2 章习题解答	(21)
3.4	第 3 章习题解答	(31)
3.5	第 4 章习题解答	(59)
3.6	第 5 章习题解答	(60)
3.7	第 6 章习题解答	(91)
3.8	第 7 章习题解答	(112)
3.9	第 8 章习题解答	(128)
3.10	第 9 章习题解答	(139)
3.11	第 10 章习题解答	(148)
3.12	第 11 章习题解答	(152)
3.13	第 12 章习题解答	(153)
3.14	第 13 章习题解答	(159)
第 4 章	实验教程	(161)
4.1	实验 1: Java 语言开发工具 JDK 安装练习	(161)
4.2	实验 2: 基本运算符和基本语句练习	(163)
4.3	实验 3: 类定义与类继承练习	(164)
4.4	实验 4: 异常发生和异常处理练习	(165)
4.5	实验 5: 图形用户界面布局练习	(166)
4.6	实验 6: 事件处理与监听器使用练习	(168)
4.7	实验 7: 组件使用与菜单设计练习	(168)
4.8	实验 8: Applet 程序设计练习	(169)

4.9	实验 9: Applet 多媒体程序设计练习	(170)
4.10	实验 10: 输入/输出练习	(171)
4.11	实验 11: 多线程程序设计练习	(171)
4.12	实验 12: 网络程序设计练习	(172)
第 5 章	课程设计与综合应用	(174)
5.1	综合应用 1: 排球比赛计分系统	(174)
5.2	综合应用 2: 小学生数学练习题目自动生成系统	(175)
5.3	综合应用 3: 二维平面反弹球模拟和自然落体运动模拟	(175)
5.4	综合应用 4: 文件内容合并	(176)
5.5	综合应用 5: 测试统计分析系统 (单机版)	(177)
5.6	综合应用 6: 人事档案管理信息系统 (网络版)	(179)

第1章 Java 语言学习指导

1.1 为什么要学习 Java 语言?

在教学实践的过程中经常有学生提出这样的问题：“老师，我们已经学习了 C 语言和 C++ 语言，还学习了使用 Visual C++ 开发工具，为什么还要学习 Java 语言啊？”可以说，诸如此类的问题是几乎所有 Java 语言初学者们都会想到的。关于这些问题，我们可以给出很多个答案，比如说“Java 语言是简单易学的程序设计语言”，“Java 语言是目前最为流行的开发平台”，“别的人都在学，所以我们也应该尝试一下”，“现在互联网这么普及，学会了 Java 语言写程序就更方便了”，等等。应该讲这些都是我们学习 Java 语言的原因，但是如果要真正说清楚这个问题，确实还要涉及很多方面的事情。要真正理解为什么要学习 Java 语言，恐怕还需要经过一些时间的学习才能寻求到答案。

要回答为什么要学习 Java 语言的问题，需要从计算机科学与技术宏观发展的角度去理解和认识，要认识到 Java 语言的出现是计算机科学与技术历史发展的一个必然，要从“Java 语言是什么”和“Java 语言对于计算机开发设计的作用”去理解。主要的原因大概有以下几个。

1. Java 语言是一种全新的面向对象的程序设计语言

众所周知，计算机程序设计语言发展到今天，面向对象的设计思想已经成为主流的程序设计指导思想。Java 语言实现了完全的面向对象程序设计，可以说是引领了程序设计语言发展的趋势和潮流。特别是在一些具体细节上，Java 语言实现了语言的优化设计，可以让软件开发人员更快更好地开发出功能复杂且完善的软件。可以说，Java 语言是程序设计语言和软件工程最新思想的集中体现者，其中所蕴涵的知识和理念代表了计算机科学与技术发展的最新成就。所以，无论是作为一种程序设计语言，还是作为一种开发工具，Java 语言都非常值得计算机业界人士学习和研究。对于从事软件开发的技术人员，可以作为一种专业技术培训来学习；对于从事计算机理论研究和教学的专业人员，可以作为一种理论修养来钻研。

2. Java 语言是一种有着巨大的优势和发展潜力的程序设计语言

正如比尔·盖茨所评价的那样，“Java 是长时间以来最优秀的程序设计语言”。Java 语言具有以下一些其他开发工具所无法比拟的优势：免费、跨平台、在学术界获得了强有力的支持、简单性和稳固性、拥有企业应用的成熟平台、可以进行无线应用、开放的 Java、获得了开源组织或者非营利机构的支持，这些构成了 Java 语言在推广应用和技术进步等方面的巨大推动力，也使得 Java 语言在今后的日子里具有巨大的发展潜力。

Java 语言的最大特点就是其跨平台性。那么，什么是平台呢？所谓平台就是软件的运行环境，通常包括两层含义，即硬件平台和软件平台。硬件平台指的是计算机硬件系统的设计规范和设计标准，包括 CPU 设计、总线设计等；软件平台一般指的是操作系统平台，比如 Windows 操作系统、Linux 操作系统、手机操作系统等。由于计算机科学自身的特点及计算机发展历史的原因，从计算机出现一直到目前为止，几十年来在计算机行业内部形成了多足鼎立的局面。无论是硬件平台，还是软件平台，都形成了多种标准和产品并存，各自独立，

相互之间兼容性差的局面。在一定历史条件下,这种局面有利于计算机技术的创新和发展,但是同时也在相当程度上阻碍了计算机技术的进一步发展和完善。程序设计语言的移植问题,软件开发工具在各种平台上的共享问题,以及软件的重复开发等问题早已有之,在计算机进入互联网时代的今天,这种矛盾日益突出。程序设计语言的移植和软件的移植问题曾经是计算机科学一个重要的研究领域,可以说从计算机出现以来,应用软件和开发工具的兼容性和可移植性问题一直是困扰计算机科学与技术发展的一个关键性问题。

随着 Internet 技术的普及和发展,逐渐形成了一个全新的技术标准和计算平台,各种应用技术和产品逐渐开始向 Internet 平台靠拢。Java 语言的出现又为 Internet 平台提供了一个统一的开发设计语言,这件事情无论是对于 Internet 技术的发展还是开发设计技术的发展,都有着巨大的意义。由于 Java 语言的问世,不同的硬件平台和软件平台之间的差异和互不兼容不再是困扰软件工程师的困难,人们终于找到了一种从程序设计语言层次上解决程序移植问题的方案。从这一点上看,Java 语言的诞生是计算机程序设计技术方面的一次革命,其前途不可限量。

3. Java 语言是一种严谨的、统一的、规范的程序设计语言

Java 语言在制定其语言标准的时候就充分考虑了语言的跨平台问题,其语言的设计、虚拟机的实现都采用了统一的技术标准,从语言成分定义到类库定义都完全统一。学习 Java 语言的最大的好处是,学习者只需要专注于程序设计语言本身的学习,而不必再像以往那样考虑运行环境问题,学习者也不必再像学习 C 语言和 C++ 语言那样在不同的平台上开发软件时需要熟悉不同的语言定义和库函数,而只需要学习“一种 Java 语言”,所编写的程序代码也可以在不同的平台上实现“一次编写,处处运行”。

4. Java 语言是多数程序员需要掌握的一门专业技术

Java 语言问世之后短短的几年时间,就以前所未有的速度得到普及。到目前为止,已经成为计算机行业中一种最主要的开发技术,在软件开发设计行业领域中所占的份额早在几年前以前就超过了一半,并且由于其开放性的原则,其市场份额有望在今后的日子里得到进一步的扩展。从它这种普及推广的速度和趋势上可以得出一个结论:Java 语言应该是多数从事软件开发设计工作的程序员必须掌握的一门专业技术。从这个意义上看,对于那些对软件开发设计有着浓厚兴趣并有志于从事这项工作的青年人而言,学习和掌握 Java 语言是必要的,也是相当重要的。

1.2 学习 Java 语言应该抓住哪些关键点?

在学习 Java 语言的时候要注意以下几个方面:首先要注意学习 Java 语言的创新特点,最好能够通过与其他程序设计语言的比较找到 Java 语言中有什么新的内容、新的思想。其次要注意学习 Java 语言的设计思想,也就是其语言系统的设计实现方式,尤其要抓住 Java 语言与平台无关的特点,注意其编译系统的实现方式。接着,要注意学习 Java 语言的基本概念和基本语法,这是学习的主要内容,其重要性毋庸置疑。另外,在学习 Java 语言的基本概念和基本语法的同时,还要注意掌握程序设计技术,要知道掌握程序设计技术是学习程序设计语言的最终目的。

学习 Java 语言的目的是为了掌握和使用 Java 语言进行应用软件的开发,所以光学习 Java

语言本身,即只掌握语法和概念是远远不够的,还必须要掌握一些必要的计算机专业知识,如软件工程理论、面向对象程序设计理论、数据结构理论、计算机操作系统理论、计算机体系结构技术、计算机网络技术、数据库技术、Internet 技术,等等。事实上,在 Java 语言中已经融合了大量计算机专业技术理论和应用知识,特别是软件工程方面的最新知识,要想充分理解和消化 Java 语言技术的全部内容,上述一些基础性的知识是必不可少的。对于经过严格的计算机专业系统训练的人来说,这或许还不是一个太大的问题,而对于一些非计算机专业的 Java 语言爱好者们来说,就应该在学习 Java 语言的同时注意充实相关的知识。

建议初学者抓住以下几个关键,深入理解和学习 Java 语言。

1. 要充分认识到 Java 是一种提供了强大功能的开发设计平台

正如 Bruce Eckel 在其广受好评的著作《Thinking in Java》中所一再谈到的,Java 语言之所以如此成功的原因就是因为它是针对当今程序开发者们所面临的诸多问题而设计的,其最主要的目标就是要提高软件的产出效率。SUN 公司在订立 Java 技术的总体设计目标时,有一个专门的降低 Java 程序员难度的目标。Java 语言将诸如多线程和网络编程那样的复杂问题归结到语言本身的特性或类库中,从而大大简化了任务。Java 语言的前景被一致看好的一个原因就是 Java 语言可以提高软件生产能力,从而让我们可以编写出功能更为强大的程序。

实际上,在 Java 语言的设计实现过程中将很多功能通过类库实现,程序员需要将更多的注意力放到对 Java 语言“可以做什么”的了解上,而不是更多地关注“怎样做什么”上。这样一来,在编写 Java 程序时就可以在程序中写上“要做什么”,而不去关注“是怎样做的”。正是因为有了以 Java 类库为基础的强大的功能支持,才能够让 Java 语言的使用者可以在短时间内迅速开发出大型的功能复杂的软件项目。

“Java”这个词首先代表一种程序设计语言,同时它还代表一种开发设计平台,还代表一种软件技术标准,还代表一种开放式标准指导下的一系列工具和产品。认识到这一点,对于已经具备了一定计算机专业训练基础的人尤其重要。也就更容易理解为什么在 Java 语言中使用几条简单的语句就可以实现诸如网络连接、数据库访问那样复杂的功能,也就更容易理解为什么在使用 Java 产品时可以从多家公司的产品中根据自己的需要去选择。

2. 要完整准确地理解 Java 语言中蕴涵的程序设计思想

了解 Java 语言优势,学习 Java 语言编程思想是学习 Java 语言和 Java 相关技术的一个十分重要的内容。Java 语言之所以成为 Java 语言,其重要的原因就是在它当中融合了很多程序设计方法和软件工程方面的新思想、新方法。很多为 Java 语言做出过重要贡献的人都是软件设计方法和软件工程方面的专家,如《Effective Java》一书的作者 Joshua Bloch、被称为“世界上对 Java 影响力最大的个人”的纽约州立大学 Oswego 分校计算机科学系的 Doug Lea、轻量级容器 Spring 的创建者 Rod Johnson 等。

面向对象程序设计方法是将需要解决的问题的诸多要素与解决问题的表述组合成“对象”,使得程序设计代码能够更好地描述实际问题。其核心思想是从要解决问题的角度,而不是从计算机的角度来描述问题。可以说,Java 语言中很多概念都建立在面向对象的基础上,其要点是通过向程序中添加新的对象,使程序能够适应问题的各种变化。例如,其中的继承机制就是在不断地定义新的类的过程中不断加深对实际问题的描述,逐步加深问题的清晰程度。可以说,Java 语言的这种面向对象程序设计方法,也是一种新的程序设计方法,是

一种软件工程思想的实施。在 Java 语言中,对象是用来描述客观事物的基本方式和手段,可以说对象是程序设计的核心,正因为如此,才将这种程序设计方法称为“面向对象”的程序设计方法。Java 语言中的一切面向对象的概念都是围绕“对象”定义的,所以抓住这一个核心概念,其他概念和理论就都容易理解。

对象机制使得在程序设计过程中构建复杂的程序代码成为可能,这主要是通过两种方式实现的:一种是通过已有的对象来创建新的对象,一种是通过类的继承机制。面向对象程序设计允许将已经存在的类的对象定义为新的类的成员,亦即类的成员可以是其他类的对象,这使得程序有可能通过这种方式构造出复杂的程序实体。由此,将软件开发的复杂性化解在类定义和对对象生成的简单性中。通过类的继承机制,可以利用接口和抽象类定义出一些共有的属性,可以利用在类中实现接口和在已有的类的基础上定义子类等手段实现代码的复用,并且由此构造出功能强大的程序实体。再加上重载、重写等机制还可以实现程序的多态。

3. 注意跟踪 Java 技术的新内容和新变化

Java 技术目前还正处于成长期,其内容随着新版本的推出在不断发展和变化,有的概念在低版本和高版本中的定义会有一些差异,有些新的功能也会在新的版本中出现,同时新的 Java 产品和 Java 工具也会不断推陈出新。应该说,这些变化对于一个发展中的事物是十分正常的,也是可以理解的。Java 的学习者要注意努力跟踪 Java 技术的发展,随时掌握 Java 技术的新变化和新内容,与时俱进,不断更新自己的知识。

还有一件事情对于 Java 来说也很重要,并且十分值得关注,那就是 Java 的开源式发展。目前最有影响力的 Java 开源软件项目是 JBoss 和 Eclipse。可以说,几乎所有的 Java 开发人员都或多或少地听到过或接触过它们。开源这种方式是一种新的软件产品的开发方式,对于软件产品的更新和软件市场的划分都有着十分重要的影响,可以让软件开发者和客户最大限度地享受高质量、高可靠性的 Java 开源软件所带来的低成本。这种方式对于未来软件业的影响和促进究竟有多大,在这种方式下软件行业的发展和市场利润的划分究竟会发生怎样的变化,都是非常值得关注的问题。

4. 多思考、多动手、多实践,还要平心静气,不能急躁,更不能浮躁

学习计算机技术光靠听课和看书是远远不够的,很多人都有这样的体会,就是很多知识当时弄明白了,但是考试过后很容易就忘记了,特别是学习程序设计语言,不亲手编写几个程序,不通过敲敲键盘编写和调试程序是绝对不行的。对于语言中的概念,都是通过不断对程序进行调试改错,通过无数次犯错误才真正认识 and 理解的。程序设计不仅是一种技术,而且还是一项技能,是需要学习者熟练掌握的一项技能。学习程序设计需要学习者在学习的过程中不断思考,在编写程序代码的过程中不断深化对语言的语法内容的认识和理解,达到掌握程序设计技能这个目的。所以,学习者最好的学习方式是写程序,做项目。程序设计是一项需要高度熟练的技术和技能,只有通过大量的练习才能达到这个目的。

另外,还要提醒 Java 学习者在学习保持一个正确的目的和正确的态度,不能为了“赶时髦”而学习 Java。最近几年由于 Java 的迅速普及和推广,也由于网络在社会生活中的影响越来越大,所以有很多人网络趋之若鹜,争相学习 Java,这本是一种好的现象,无可厚非。只是在学习过程中一定要注意脚踏实地,一步一个脚印,认认真真地学好 Java。现在的社会很浮躁,IT 行业更浮躁,在浮躁的心态下是不能学好东西的。

1.3 学习 Java 语言的基本步骤有哪些?

建议按照以下的学习步骤学习 Java 语言和 Java 技术。

1. 从开始学习的时候就要安装好自己的 Java 语言开发环境

最好是在开始学习 Java 语言的时候就在计算机上为自己建立一套 Java 语言的开发环境。初学者可以安装最简单的 JDK 和 API 文档资料,有兴趣的学习者还可以安装功能更为复杂完善的 NetBeans、JBuilder 或 Eclipse 开发工具。安装 Java 语言开发环境一方面可以在学习过程中随时编写程序,巩固学习成果,通过编写和调试程序辨别遇到的问题,解答学习上的疑问;另一方面还可以在学习过程中随时查阅 Java 语言的 API 文档资料,增加自己对 Java 语言和 Java 平台的了解。

2. 学习 Java 语言的基本语法

学习 Java 语言还是要从基本的语言成分学起,包括字符集、标识符、语句等。对于有 C 语言和 C++语言基础的人,这部分内容的学习会比较轻松,而且会有轻车熟路的感觉,并且由于 Java 语言简洁的语法,相比于 C++语言的烦琐,让人还有一种如释重负的感觉。一般来说,在学习这部分内容的时候不会有太大的困难,需要注意的是把握几个关键问题。

3. 学习 Java 语言的面向对象的编程特性

正如前面已经讲过的,面向对象的编程是 Java 语言的核心内容,对于已经经历了面向对象方面的训练,或者是已经学习过 C++语言的人来说,这部分也不是十分困难的事情。如果没有这样的基础,那么这部分知识就是需要全力以赴去跨越的一道难关。

4. 熟悉 Java 语言的类库,阅读 Java API 文档

Java 语言的很多重要成分都被定义在其类库中,所以在学习 Java 语言的过程中熟悉类库中的内容,阅读 Java API 文档是必不可少的。

Java API 文档资料中几个重要的包必须要了解,如 java.lang、java.io、java.awt、java.applet、java.util、java.sql、javax.swing 等,应当对其中的类的种类、用途、主要的属性和方法等有一个了解和认识,对于最核心的几个类要做到耳熟能详,这是掌握 Java 语言基本性能的一个很好的方法。要了解 Java 语言给我们提供了哪些类,每个类用在什么场合,当遇到问题的时候,知道哪个类,或者哪几个类的组合使用可以解决遇到的问题。实际上在写程序的时候,通常只要知道该用哪个类来完成工作就足够了。

在阅读 Java API 文档时要注意把握一个“度”,由于 Java API 文档的内容繁杂,信息量十分丰富,要想全都记住是不可能的,也是没有必要的。对于常用的类要多熟悉一些,不常用的类可以暂时先不看,类中的常用属性、方法要仔细地看,不常用的属性、方法可以简略地看。Java 语言的开发者们为 Java 用户提供了一套十分全面、十分好用的 API 文档,通过阅读这些文档资料,可以十分容易地了解大量的 Java 知识,掌握很多 Java 编程技术。

5. 学习 Java Web 编程

如果要在学完 Java 语言基本语法的基础上继续学习 Java Web 编程,即 Servlet/JSP 编程,

就必须要对 HTTP 协议有充分的了解。很多人发现,当经过简单的 Java 语法学习之后,开始学习 Java Web 编程时,会突然有一种不知所措的感觉,一时之间好像什么都不会了,也不知道该干些什么了。此时最应该做的就是详细了解 Servlet/JSP 的工作原理,认真学习 Servlet 和 JSP 编程时使用的几个常用类,同时最好能够查阅一下 HTTP 协议的有关类,经过这些步骤,学会 Java Web 编程就是一件水到渠成的事。

6. 学习 J2EE 编程

J2EE 实际上是一系列平台、规范和协议的总称,是面向企业级的开发应用而设计的一整套开发设计规范。J2EE 的学习是比较复杂的,主要表现在相关的一系列平台、规范和协议上。有经验的 Java 程序员都知道,只掌握 Java 语言本身很难开发 J2EE 应用程序。J2EE 的工作模式可以说是专业水平的,一般要求学习者具备相当的计算机理论方面的基本训练。对于非计算机专业的人士,建议在学习 J2EE 之前先进行一些理论方面的知识准备。J2EE 包括 Servlet、JSP、EJB、JDBC、JNDI、JMS、RMI、JTA 等内容,还涉及 UML 和设计模式,提倡框架式设计。初学者应该把学习的重点放在设计模式和框架的学习上。

7. 学习一些 Java 工具产品的使用

由于 Java 开放式的标准,因此很多优秀的 Java 工具由不同的软件公司提供,而不是像微软公司那样由其一家提供。这带来的一个变化是 Java 使用者有可能在不同的产品之间进行比较,挑选自己最满意的开发工具。学习使用市面上一些流行的 Java 开发工具,既是 Java 学习者的一项基本任务,也可以让自己在各个公司的产品之间从容选择。在“精其技,利其器”的同时也让自己使用到最好的开发工具。

8. 学习 Java 语言的设计模式和编程思想

在 Java 语言的发展历程中,吸收了很多新的软件设计思想,通常是一些解决复杂的程序设计问题的优化方法,一般将其称为模式(pattern)。如果已经开始学习 Java 技术并打算采用 Java 技术开发设计软件,特别是设计比较大型的软件系统,那么学习和理解这些模式是必需的。在 Java 中,MVC 程序设计模式是在 J2EE 软件开发领域应用非常广泛的设计模式。在这一新的设计思想的指导下,出现了很多新的设计框架和工具,如基于 J2EE 平台的已经成为创建 Web 数据库应用的一个最流行的 MVC 框架 Struts,还有 Hibernate、Spring 等。可以说,掌握 Java 技术也是掌握这些新的程序设计思想的过程。

第2章 Java 语言常见问题解答

2.1 基本语法概念问题

1. Java 语言是一门非常优秀的语言，它的缺点有哪些？

如同其他任何事情一样，Java 语言也是有缺点的，主要表现在以下两个方面：

第一，Java 语言的运行速度慢，有些场合下甚至非常慢。众所周知，Java 语言程序在运行时要依赖于 Java 虚拟机，在操作系统中既要加载 Java 虚拟机，还要存储 Java 字节码，会占用更多的内存空间和处理器运行时间。因为它不是直接执行机器码，所以相比于汇编语言、C 语言和 C++ 语言等其他语言编写的程序，其运行速度明显要慢。

第二，Java 语言远离操作系统的底层。因为 Java 语言在设计时要充分考虑跨平台性，并且取消了指针操作，所以不像 C 语言那样灵活。这既是 Java 语言的优点，同时又是 Java 语言的缺点，因此它不能像汇编语言、C 语言那样更接近操作系统，也就不能和操作系统的底层打交道。尽管可以通过 Java 语言的 JNI 即 Java 本地接口技术解决这一问题，但也只是解决了一部分问题。

基于上述缺陷，Java 语言一般不直接用于建立大型项目。相信随着计算机硬件的发展，Java 语言运行速度慢的问题会迎刃而解，至于 Java 语言远离操作系统底层的问题，应该看做是 Java 语言所倡导的计算机技术的一个发展趋势，这很可能会导致未来计算机技术的一次飞跃性发展。

2. 编写 Java 语言程序时，怎样才能生成能在 Windows 下运行的 exe 文件，或者直接双击就能运行的执行文件？

回答这个问题首先要搞清楚 Windows 下具有 .exe 扩展名的可执行文件是怎么回事。具有 .exe 扩展名的可执行文件是 Microsoft 公司制定的 DOS 和 Windows 操作系统下的可执行文件格式。在 Windows 环境下，只要用鼠标双击文件的图标就可以启动执行文件。Microsoft 公司所提供的各种开发工具都可以将源程序编译为这种可执行文件。Java 语言是一种跨平台的程序设计语言，并不依赖于某一种具体的操作系统平台，所以不支持 Microsoft 公司的这种可执行文件格式，因而在 Java 语言中并不能够生成这种文件，同样也不能生成其他操作系统下的可执行文件格式，这也是 Java 语言平台无关性特点的一个典型体现。但是，在 Java 语言中支持将程序代码生成一种具有 .jar 扩展名的压缩执行包，可以在操作系统环境下通过双击直接运行。另外，某些 Java 语言程序开发工具提供把 Java 语言源程序转换为 .exe 可执行文件的功能，比如在 JBuilder 开发工具中就可以生成具有 .exe 扩展名的可执行文件。

3. ASCII 字符集与 Unicode 字符集的区别是什么？

ASCII 字符集中的字符采用 1 字节 8 位存储方式，最多可以表达 256 个字符；而 Unicode 字符集中的字符采用 2 字节 16 位存储方式，最多可以表达 65536 个字符。由于 Unicode 字符集能够表达的字符数量多，所以从理论上讲可以包括各个国家的文字字符，例如中文字符。

4. “goto”和“const”到底是不是Java语言的关键字？Java语言中有没有goto语句？

“goto”和“const”这两个字符串是Java语言中被保留但却不再使用的两个关键字。在Java语言中没有goto语句。

5. “int”和“Integer”有什么区别？

“int”是基本数据类型中的32位整型量的专用标识关键字；而“Integer”是基本数据类型int的封装类，该类保存在java.lang包中，其对象实例中包装了一个基本类型int的值。同样的情形还有“boolean”和“Boolean”、“char”和“Character”、“byte”和“Byte”、“short”和“Short”、“long”和“Long”、“float”和“Float”、“double”和“Double”等。

在Java语言中提供了两种处理基本数据的方式：基本数据类型定义和封装类型引用。例如，“int”是基本数据类型定义，而“Integer”是一个在类库中定义的封装了int型数据的封装类。在java.lang包中为8种基本数据类型中的每一种都提供了封装类。基本数据类型定义和封装引用类型二者之间的差别是很大的，它们具有不同的语义，它们的行为也完全不同，并且具有不同的特征和用法，包括存储数据结构，用做某个类的实例数据时所指定的默认值。当在程序中使用了引用类型的变量时，其默认值为“null”。另外，使用封装引用类型时可以通过定义在封装类中的方法处理数据，具有相当大的灵活性，特别是采用这种方式可以实现多态性。

对应于8种基本数据类型的封装类都是最终类，不可以被继承。

6. “String”是基本的数据类型吗？

在Java语言中只定义了4类8种基本数据类型，“String”不包含在其中，不是基本的数据类型。事实上“String”是一个类，该类可以用来处理字符串型对象实例，所以在Java语言中的字符串都是对象实例引用。一般在进行字符串处理时，都会采用更为灵活的StringBuffer类，因为StringBuffer类的对象实例是可以修改的，而String类的对象实例一旦被创建之后不能进行修改。所以在程序设计过程中一般都是创建一个StringBuffer类的对象实例来动态构造字符串数据。

7. “char”型变量中能不能存储一个中文汉字？

原则上讲，“char”型变量中可以存储一个中文汉字。因为在Java语言中采用unicode编码，任何一个字符都是16位存储空间，所以在一个char型变量中存放一个中文汉字字符是没有问题的。

8. 运算符“&”和“&&”的区别是什么？运算符“|”和“||”的区别是什么？

运算符“&”是按位与运算符，是二元位运算符，其两个操作数是整数类型。该运算符还可以作为逻辑与运算符，也是二元位运算符，其两个操作数是逻辑类型。

运算符“&&”是条件逻辑与运算符，也是二元运算符，其两个操作数是逻辑类型，该运算符在计算逻辑值之前先考察左运算数的逻辑值，如果左运算数的逻辑值为“false”，则不再考察右运算数的逻辑值，而直接得到结果“false”。

运算符“&”和“&&”的区别在于：前者既是位运算符，又是逻辑运算符，既可以操作整型数，也可以操作逻辑型数；后者是逻辑运算符，只能操作逻辑型数，在进行逻辑运算时，可能会比前者节省运算时间。

运算符“|”和“||”的区别与之类似。

9. “sizeof”是Java语言的运算符吗？

“sizeof”这个字符串不是Java语言的运算符。

由于很多Java语言的初学者在学习Java语言之前都学习过C语言，Java语言与C语言有很多相似之处，所以经常有人误以为Java语言把C语言的运算符体系都继承下来了，于是想当然地认为“sizeof”也是Java语言的运算符。

在C语言中，定义“sizeof”这个运算符的意义和作用在于需要经常测试各种数据类型的数据的大小，这是因为在不同平台上所定义的C语言编译系统对于同样的数据类型量的数据长度的定义是不尽相同的。由于Java语言在这一点上进行了明确的改进，统一规定了各种数据类型量的数据长度，所以只要知道具体的数据类型也就知道了其大小，因此类似C语言中的“sizeof”这样的运算符也就失去了存在的意义，于是在Java语言中就没有保留这个运算符。这一点也是Java语言平台无关性和可移植性的一个具体体现。

10. 强制类型转换算得上是运算符吗？

严格地说，强制类型转换算得上是Java语言的运算符，并且这个运算符还有自己的优先级，正因为如此，才将其编入Java语言运算符优先级和结合性表中。

11. for循环语句、while循环语句和do-while循环语句三者之间有何差别？

for循环语句的循环次数通常是可以通过循环控制变量的初始值、增量和终止条件判断出来的。while循环语句可以不使用循环控制变量，一般不能事先判断出循环次数，其循环体可能一次也不执行，也可能执行若干次。do-while循环语句也可以不使用循环控制变量，但是其循环体将至少被执行一次。在很多场合下，这三种循环语句可以互相转换，但对于有些情形，只能用其中的一种形式来实现功能。

12. switch语句中的控制表达式是否可以是“byte”类型，是否可以是“long”类型，是否可以是“String”类型？

switch语句中的控制表达式可以是“byte”类型，但是不能是“long”类型，也不能是“String”类型。

在switch语句中，其控制表达式只能是“byte”、“short”、“int”或者“char”类型，其他的基本数据类型是不可以的，其他的引用数据类型更是不能使用的。

13. 数组有没有“length()”这个方法？String有没有“length()”这个方法？

数组中没有“length()”这个方法，但是有length这个属性。在String类中定义有“length()”这个方法。

14. 表达式“k=k+1”和“k+=1”有什么不同？

从表面上看，这两个表达式都是将变量k的值增加1，但是其运算过程有很大的不同。

如果变量k是“int”型变量，则这两个表达式是完全一样的，但是如果变量k是“byte”或“short”类型，则情况就不一样了。因为常数1被看成“int”型，在采用第一种表达式进行加法运算时，“byte”或“short”类型变量k将被自动地进行数据类型的提升，先提升为“int”类型，再进行加法运算，之后再将运算结果赋值给变量k，所以对于前一个表达式，变量k