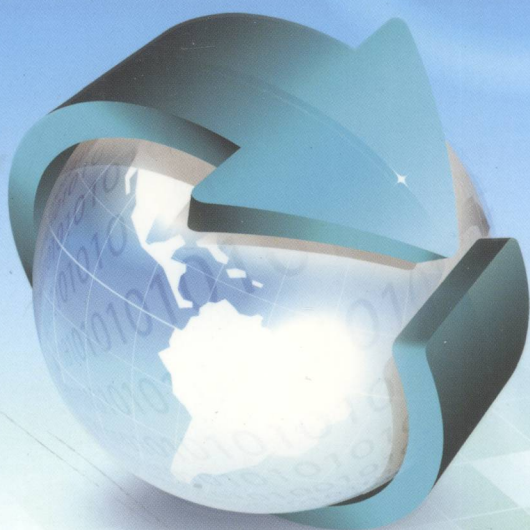




高职高专工作过程·立体化创新规划教材

——计算机系列



Java程序设计 与应用教程

刘新娥 罗晓东 主 编

赠送
电子课件

- 以培养技能型创新人才为目标，用丰富的版块合理安排全文，突出实用性和可操作性。
- 以工作过程为导向，全面展示案例实施的全过程，提炼技术要点，即学即用，面向就业。
- 以强化实际操作技能为主线，答疑解惑，解决工作中的常见问题。

清华大学出版社



高职高专工作过程·立体化创新规划教材——计算机系列

Java 程序设计与应用教程

刘新娥 罗晓东 主 编

清华大学出版社

北 京

内 容 简 介

Java 是一种很优秀的编程语言,具有面向对象、与平台无关、安全、稳定和多线程等特点,是目前软件设计中极为健壮的编程语言。Java 语言不仅可以用来开发大型的应用程序,而且特别适合于在 Internet 上应用开发,Java 已成为网络时代最重要的编程语言之一。

本书注重可读性和实用性,每章都从应用出发,通过工作场景来引导读者,配备大量的例题,这些例题都经过精心的挑选,既能帮助读者理解知识,又具有启发性。本书通俗易懂,便于自学,针对较难理解的问题,例子都是从简单到复杂,内容逐步深入,便于读者掌握 Java 编程的技巧。每章的末尾均提供案例实训和关于该章主题内容的常见问题,以及作者给出的答案,章末的测验题可测试读者对内容的理解程度。

全书共分 12 章,分别介绍 Java 的简单数据类型、运算符、表达式、流程控制、类、数组、字符串、异常处理、容器、Java I/O 系统、并发、图形化用户界面、Applet 程序设计和 Java 的应用等内容。

本书不仅可以作为高职高专院校相关专业的教材,也适合自学者及软件开发人员参考使用。

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 程序设计与应用教程/刘新娥,罗晓东主编. --北京:清华大学出版社,2011.5
(高职高专工作过程·立体化创新规划教材——计算机系列)
ISBN 978-7-302-25288-7

I. ①J… II. ①刘… ②罗… III. ①JAVA 语言—程序设计—高等职业教育—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2011)第 059479 号

责任编辑:章忆文 宋延清

装帧设计:山鹰工作室

责任校对:王 晖

责任印制:杨 艳

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

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

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 刷 者:北京密云胶印厂

装 订 者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185×260 印 张:22.5 字 数:542 千字

版 次:2011 年 5 月第 1 版 印 次:2011 年 5 月第 1 次印刷

印 数:1~4000

定 价:39.00 元

产品编号:035616-01



读者回执卡

欢迎您立即填写回函

您好！感谢您购买本书，请您抽出宝贵的时间填写这份回执卡，并将此页剪下寄回我公司读者服务部。我们会在以后的工作中充分考虑您的意见和建议，并将您的信息加入公司的客户档案中，以便向您提供全程的一体化服务。您享有的权益：

- ★ 免费获得我公司的新书资料；
- ★ 寻求解答阅读中遇到的问题；
- ★ 免费参加我公司组织的技术交流会及讲座；
- ★ 可参加不定期的促销活动，免费获取赠品；

读者基本资料

姓 名 _____ 性 别 ☐男 ☐女 年 龄 _____
 电 话 _____ 职 业 _____ 文化程度 _____
 E-mail _____ 邮 编 _____
 通讯地址 _____

请在您认可处打√ (6至10题可多选)

- 您购买的图书名称是什么：_____
- 您在何处购买的此书：_____
- 您对电脑的掌握程度：

☐ 不懂	☐ 基本掌握	☐ 熟练应用	☐ 精通某一领域
-----------------------------	-------------------------------	-------------------------------	---------------------------------
- 您学习此书的主要目的是：

☐ 工作需要	☐ 个人爱好	☐ 获得证书	
-------------------------------	-------------------------------	-------------------------------	--
- 您希望通过学习达到何种程度：

☐ 基本掌握	☐ 熟练应用	☐ 专业水平	
-------------------------------	-------------------------------	-------------------------------	--
- 您想学习的其他电脑知识有：

☐ 电脑入门	☐ 操作系统	☐ 办公软件	☐ 多媒体设计
☐ 编程知识	☐ 图像设计	☐ 网页设计	☐ 互联网知识
☐ 书名	☐ 作者	☐ 出版机构	☐ 印刷、装帧质量
☐ 内容简介	☐ 网络宣传	☐ 图书定价	☐ 书店宣传
☐ 封面、插图及版式	☐ 知名作家（学者）的推荐或书评	☐ 其他	
- 您比较喜欢哪些形式的学习方式：

☐ 看图书	☐ 上网学习	☐ 用教学光盘	☐ 参加培训班
------------------------------	-------------------------------	--------------------------------	--------------------------------
- 您可以接受的图书的价格是：

☐ 20元以内	☐ 30元以内	☐ 50元以内	☐ 100元以内
--------------------------------	--------------------------------	--------------------------------	---------------------------------
- 您从何处获知本公司产品信息：

☐ 报纸、杂志	☐ 广播、电视	☐ 同事或朋友推荐	☐ 网站
--------------------------------	--------------------------------	----------------------------------	-----------------------------
- 您对本书的满意度：

☐ 很满意	☐ 较满意	☐ 一般	☐ 不满意
------------------------------	------------------------------	-----------------------------	------------------------------
- 您对我们的建议：_____

1 0 0 0 8 4

北京100084—157信箱

读者服务部

收

邮政编码：□□□□□□

贴 票
邮 处

请剪下本页填写清楚，放入信封寄回，谢谢！

丛 书 序

高等职业教育强调“以服务为宗旨，以就业为导向，走产学结合发展道路”。服务社会、促进就业和提高社会对毕业生的满意度，是衡量高等职业教育是否成功的重要指标。坚持“以服务为宗旨，以就业为导向，走产学结合发展道路”体现了高等职业教育的本质，是其适应社会发展的必然选择。为了提高高职院校的教学质量，培养符合社会需求的高素质人才，我们计划打破传统的高职教材以学科体系为中心，讲述大量理论知识，再配以实例的编写模式，设计一套突出应用性、实践性的丛书。一方面，强调课程内容的应用性。以解决实际问题为中心，而不是以学科体系为中心；基础理论知识以应用为目的，以“必需、够用”为度。另一方面，强调课程的实践性。在教学过程中增加实践性环节的比重。

2009年5月，我们组织全国高等职业院校的专家、教授组成了“高职高专工作过程·立体化创新规划教材”编审委员会，全面研讨人才培养方案，并结合当前高职教育的实际情况，历时近两年精心打造了这套“高职高专工作过程·立体化创新规划教材”丛书。我们希望通过对这一套全新的、突出职业素质需求的高质量教材的出版和使用，促进技能型人才培养的发展。

本套丛书以“工作过程为导向”，强调以培养学生的职业行为能力为宗旨，以现实的职业要求为主线，选择与职业相关的教学内容组织开展教学活动和过程，使学生在学习和实践中掌握职业技能、专业知识及工作方法，从而构建属于自己的经验和知识体系，以解决工作中的实际问题。

本丛书首推书目

- 计算机应用基础
- 办公自动化技术应用教程
- 计算机组装与维修技术
- C++语言程序设计与应用教程
- C 语言程序设计
- Java 程序设计与应用教程
- Visual Basic 程序设计与应用开发
- Visual C# 2008 程序设计与应用教程
- 网页设计与制作
- 计算机网络安全技术
- 计算机网络规划与设计
- 局域网组建、管理与维护实用教程
- 基于.NET 3.5 的网站项目开发实践
- Windows Server 2008 网络操作系统
- 基于项目教学的 ASP.NET(C#)程序开发设计
- SQL Server 2008 数据库技术实用教程
- 数据库应用技术实训指导教程(SQL Server 版)

- 单片机原理及应用技术
- 基于 ARM 的嵌入式系统接口技术
- 数据结构实用教程
- AutoCAD 2010 实用教程
- C# WEB 数据库编程

丛书特点

(1) 以项目为依托,注重能力训练。以“工作场景导入”→“知识讲解”→“回到工作场景”→“工作实训营”为主线编写,体现了以能力为本位的教育模式。

(2) 内容具有较强的针对性和实用性。丛书以贴近职业岗位要求、注重职业素质培养为基础,以“解决工作场景”为中心展开内容,书中每一章节都涵盖了完成工作所需的知识和具体操作过程。基础理论知识以应用为目的,以“必需、够用”为度,因而具有很强的针对性与实用性,可提高学生的实际操作能力。

(3) 易于学习、提高能力。通过具体案例引出问题,在掌握知识后立刻回到工作场景解决实际问题,使学生很快上手,提高实际操作能力;每章末的“工作实训营”板块都安排了有代表意义的实训练习,针对问题给出明确的解决步骤,并给出了解决问题的技术要点,且对工作实践中常见问题进行分析,使学生进一步提高操作能力。

(4) 示例丰富、由浅入深。书中配备了大量经过精心挑选的例题,既能帮助读者理解知识,又具有启发性。针对较难理解的问题,例子都是从简单到复杂,内容逐步深入。

读者定位

本系列教材主要面向高等职业技术学院和应用型本科院校,同时也非常适合计算机培训班和编程开发人员培训、自学使用。

关于作者

丛书编委会特聘执教多年且有较高学术造诣和实践经验丰富的名师参与各册之编写。他们长期从事有关的教学和开发研究工作,积累了丰富的经验,对相应课程有较深的体会与独特的见解,本丛书凝聚了他们多年的教学经验和心血。

互动交流

本丛书保持了清华大学出版社一贯严谨、科学的图书风格,但由于我国计算机应用技术教育正在蓬勃发展,要编写出满足新形势下教学需求的教材,还需要我们不断的努力实践。因此,我们非常欢迎全国更多的高校老师积极加入到“高职高专工作过程·立体化创新规划教材——计算机系列”编审委员会中来,推荐并参与编写有特色、有创新的教材。同时,我们真诚希望使用本丛书的教师、学生和读者朋友提出宝贵意见和建议,使之更臻成熟。联系信箱: Book21Press@126.com。

丛书编委会

前 言

Internet 极大地影响着计算机世界，同时也影响着人们的生活，它已经成为全球范围最大、资源最为丰富的网络。而美国的 Sun Microsystems 公司推出的 Java 语言是一种令人激动的新型语言，它具有面向对象、与平台无关、可移植性强、安全、支持分布式等诸多特征，特别适合开发 Internet 网络程序。Sun Microsystems 的合作创办者 Bill Joy 在说明这种新语言的时候，并没有正面说明它的优势。他只是宣布“这代表了最近历年以来试图为建立更加简单和更加可靠的软件而开发更好的编程语言和环境的最终结果”。网络是多台计算机的互联集合，Java 则正是网络中通用的世界语。

Java 已经广泛用于桌面应用程序、Internet 服务器、中间件、个人数字助理、嵌入式设备等软件的开发，特别是计算机网络方面的软件，如大型软件 ERP 等。目前越来越多的人开始学习并使用 Java 语言，全世界已经掀起了一股 Java 热潮。Java 2 的推出使 Java 的用途更加广泛，它适用于各种应用开发，尤其是网络应用和嵌入式系统开发。

1. 本书的特点

现有的大量教材都比较注重理论知识的讲解，忽略了学生对实际操作知识的需要，使得学生在学完一门课程之后，面对实际问题还是很难动手解决，这是一个很大的弊端。考虑到这点，本书以“工作场景导入”→“知识讲解”→“回到工作场景”→“工作实训营”为主线编写，理论与操作并重，使学生在学的过程中可以提高实际操作能力。

本书的主要特点如下。

(1) 结构清晰，模式合理。以工作场景引导出理论知识，在进行详细讲解后，再运用所学到的知识解决工作场景中的问题。

(2) 示例丰富、由浅入深。本书配备了大量经过精心挑选的例题，既能帮助读者理解知识，又具有启发性。针对较难理解的问题，例子都是从简单到复杂，内容逐步深入，便于读者掌握 Java 编程的技巧。

(3) 上手快、提高快。通过具体案例引出问题，在掌握知识后立刻回到工作场景解决实际问题，使学生能够很快上手，提高实际操作能力；每章末的“工作实训营”板块都安排了有代表意义的实训练习，并给出了解决问题的技术要点，另外还提供了实践及面试中常见问题解析，能够进一步拓展学生的知识，应对将来的挑战。

(4) 包含大量的范例代码。其中绝大部分都是完整的程序。

2. 本书的内容

本书共 12 章，具体安排如下。

第 1 章 介绍 Java 语言的发展简史，以及 Java 开发环境的安装，并向读者展示如何运行自己的第一个 Java 程序。

第 2 章 详细地讲解 Java 的 8 种基本数据类型和其对应的封装类型。

第 3 章 主要讲解 Java 常用的运算符和表达式的使用。

第4章 主要讲解 Java 的流程控制。通过使用流程控制语句,能更好地完成程序上的各种逻辑操作,编写出复杂的应用程序。

第5章 详细地讲解 Java 的复杂类型——类、数组和字符。从这一章开始体现 Java 的面向对象编程。

第6章 主要讲解 Java 的异常处理,即在程序中出现错误时,应如何处理。

第7章 通过实例详细地讲解 Java 容器的使用。

第8章 详细地讲解 Java 的 I/O 系统。主要包括几种输入/输出和文件上的操作。

第9章 主要讲解 Java 如何通过多线程实现并发机制。

第10章 详细讲解 Java 的图形化用户界面,通过对 GUI 的学习,可以很容易地开发出我们所需要的桌面化应用程序。

第11章 主要讲解 Java Applet 小程序,Applet 可以增加 HTML 文件的有关的内容,使网页更加富有生气,如添加声音、动画等。

第12章 主要讲解面向 Java 的应用。内容包括 JDBC、数据库的连接和操作及网络 Socket 编程,并简单地向读者介绍 J2ME、J2SE 和 J2EE。

3. 适合的读者

本书可作为高职高专院校相关专业的教材,同时也适用于编程新手、已经了解 Java 1.0 或者 1.1 的人和对其他语言(例如 C 语言、Visual C++ 或者 Visual Basic 等)有经验的开发人员,将学会如何开发小程序以及应用程序。如果已经从事 Java 编程多年,可以使用本书作为技术手册,同样会受益匪浅。

当读者完成本书的学习后,将能够处理 Java 语言中的任何常见问题,而且对 Java 应用开发相当熟悉,可以在编程项目中(在 Web 上或在 Web 下)一展身手。

4. 编者与致谢

本书由刘新娥、罗晓东任主编。在本书编写过程中,汪大锋、姚昌顺、许勇、杨明、杨萍、赵传审、李海、赵明、张伍荣、范荣钢、钱阳勇、陈芳等同志给予了很大的帮助。限于作者水平,书中难免存在不当之处,恳请广大读者批评指正。

编 者

目 录

第 1 章 Java 语言概论..... 1

1.1 工作场景导入..... 2

1.2 Java 语言简介..... 2

1.2.1 什么是 Java..... 2

1.2.2 Java 的特点..... 3

1.3 Java 开发环境的构建..... 4

1.3.1 JDK 的安装..... 4

1.3.2 环境变量的设定..... 5

1.4 第一个 Java 应用程序..... 6

1.4.1 编写源代码..... 6

1.4.2 编译源代码..... 7

1.4.3 运行程序..... 7

1.5 使用 Eclipse 开发工具..... 8

1.5.1 Eclipse 简介与安装..... 8

1.5.2 使用 Eclipse 开发 Java 项目..... 8

1.6 回到工作场景..... 10

1.7 工作实训营..... 11

1.7.1 训练实例..... 11

1.7.2 工作实践和面试中常见问题 解析..... 13

习题..... 14

第 2 章 简单数据类型..... 17

2.1 工作场景导入..... 18

2.2 概述..... 18

2.2.1 变量..... 18

2.2.2 标识符..... 20

2.2.3 保留字..... 20

2.3 简单数据类型..... 20

2.3.1 整型数据..... 21

2.3.2 浮点型数据..... 21

2.3.3 布尔型数据..... 22

2.3.4 字符型数据..... 23

2.4 数据类型之间的转换..... 23

2.4.1 类型的自动转换规则..... 23

2.4.2 类型的强制转换..... 23

2.5 Java 类库中的包装类..... 24

2.6 回到工作场景..... 25

2.7 工作实训营..... 27

2.7.1 训练实例..... 27

2.7.2 工作实践与面试中常见问题 解析..... 28

习题..... 29

第 3 章 运算符和表达式..... 31

3.1 工作场景导入..... 32

3.2 概述..... 32

3.2.1 运算符..... 32

3.2.2 表达式..... 33

3.3 算术运算符和算术表达式..... 33

3.4 关系运算符和关系表达式..... 35

3.5 布尔逻辑运算符和布尔逻辑表达式..... 36

3.6 位运算符和位运算表达式..... 36

3.6.1 位逻辑运算符..... 36

3.6.2 移位运算符..... 37

3.7 运算符和优先级..... 37

3.8 回到工作场景..... 38

3.9 工作实训营..... 40

3.9.1 训练实例..... 40

3.9.2 工作实践和面试中常见问题 解析..... 44

习题..... 44

第 4 章 流程控制..... 47

4.1 工作场景导入..... 48

4.2 概述..... 48

4.3 分支语句..... 49

4.3.1 条件语句..... 49

4.3.2 多分支语句..... 50

4.4 循环语句.....	51	5.10.1 一维数组.....	78
4.4.1 while 循环.....	51	5.10.2 多维数组.....	80
4.4.2 do-while 循环.....	52	5.10.3 应用实例.....	80
4.4.3 for 循环.....	52	5.11 字符串.....	81
4.5 跳转语句.....	53	5.11.1 字符串的定义.....	81
4.5.1 break 和 continue.....	53	5.11.2 字符串运算.....	82
4.5.2 return 语句.....	55	5.11.3 调用字符串方法.....	82
4.6 循环语句与分支语句的嵌套.....	55	5.12 回到工作场景.....	83
4.7 递归.....	55	5.13 工作实训营.....	85
4.8 回到工作场景.....	56	5.13.1 实例训练.....	85
4.9 工作实训营.....	58	5.13.2 工作实践和面试中常见问题 解析.....	88
4.9.1 训练实例.....	58	习题.....	90
4.9.2 工作实践和面试中常见问题 解析.....	61	第6章 异常处理.....	95
习题.....	61	6.1 工作场景导入.....	96
第5章 类、数组和字符串.....	63	6.2 概述.....	96
5.1 工作场景导入.....	64	6.3 异常类型.....	97
5.2 概述.....	64	6.3.1 异常情形.....	97
5.3 类.....	65	6.3.2 声明异常.....	98
5.3.1 类的定义.....	65	6.3.3 抛出异常.....	99
5.3.2 类的实例化.....	66	6.4 处理异常.....	99
5.3.3 方法重载.....	67	6.4.1 try 语句和 catch 语句.....	99
5.3.4 构造方法.....	67	6.4.2 finally 语句.....	100
5.4 接口.....	68	6.4.3 异常处理的原则.....	101
5.5 包.....	71	6.5 回到工作场景.....	101
5.6 类成员修饰符.....	71	6.6 工作实训营.....	103
5.6.1 访问控制修饰符.....	71	6.6.1 训练实例.....	103
5.6.2 static 修饰符.....	72	6.6.2 工作实践和面试中常见问题 解析.....	105
5.6.3 final 修饰符.....	72	习题.....	106
5.6.4 abstract(抽象)修饰符.....	72	第7章 容器.....	109
5.7 继承.....	72	7.1 工作场景导入.....	110
5.8 类库.....	74	7.2 概述.....	110
5.9 内部类.....	75	7.3 List.....	111
5.9.1 成员内部类.....	75	7.3.1 ArrayList.....	111
5.9.2 局部内部类.....	76	7.3.2 迭代器.....	112
5.9.3 静态内部类.....	76	7.3.3 LinkedList.....	113
5.9.4 匿名内部类.....	77		
5.10 数组.....	78		

7.4	Set	114	9.3	线程的概念	154
7.5	Map	114	9.3.1	什么是线程	154
7.6	Vector	115	9.3.2	Java 中的线程模型	155
7.7	回到工作场景	116	9.4	线程的创建	155
7.8	工作实训营	118	9.4.1	定义任务	155
7.8.1	训练实例	118	9.4.2	Thread 类	157
7.8.2	工作实践和面试中常见问题 解析	121	9.4.3	使用 Executor	157
习题	122		9.5	线程的生命周期	158
第 8 章	Java I/O 系统	123	9.5.1	创建线程	159
8.1	工作场景导入	124	9.5.2	启动线程	159
8.2	概述	124	9.5.3	线程运行	159
8.3	File 类	125	9.5.4	线程阻塞	159
8.3.1	创建文件	125	9.5.5	终止线程	160
8.3.2	File 类的方法	126	9.6	线程的调度与线程控制	160
8.3.3	目录列表器	128	9.6.1	线程优先级与线程调度 策略	160
8.4	字节流	130	9.6.2	线程的基本控制	162
8.4.1	InputStream 类型	130	9.7	线程同步	165
8.4.2	OutputStream 类型	132	9.7.1	对象锁	167
8.5	字符流	134	9.7.2	可重入锁	168
8.6	使用流	135	9.7.3	notifyAll 和 wait 方法	168
8.6.1	随机访问文件 (RandomAccessFile)	135	9.8	回到工作场景	170
8.6.2	读取器和写出器	137	9.9	工作实训营	173
8.6.3	过滤流	139	9.9.1	训练实例	173
8.6.4	管道流	141	9.9.2	工作实践和面试中常见问题 解析	174
8.6.5	标准 I/O	141	习题	175	
8.7	对象序列化	142	第 10 章	图形化用户界面	179
8.8	回到工作场景	144	10.1	工作场景导入	180
8.9	工作实训营	147	10.2	概述	180
8.9.1	训练实例	147	10.3	Swing 基础	181
8.9.2	工作实践和面试中常见问题 解析	150	10.3.1	一个显示框架	182
习题	151		10.3.2	面板	183
第 9 章	并发	153	10.3.3	创建按钮	185
9.1	工作场景导入	154	10.3.4	捕获事件	186
9.2	概述	154	10.3.5	控制布局	187
			10.4	Swing 事件模型	196
			10.4.1	事件处理简介	196

10.4.2 事件处理实例.....	196	11.4.3 Applet 中的事件处理.....	263
10.4.3 使用事件适配器.....	200	11.5 Applet 的多媒体支持.....	264
10.4.4 AWT 事件继承关系.....	202	11.5.1 显示图像.....	264
10.5 Swing 组件一览.....	203	11.5.2 动画制作.....	265
10.5.1 标签.....	203	11.5.3 播放声音.....	268
10.5.2 文本输入类组件.....	205	11.6 回到工作场景.....	269
10.5.3 选择类组件.....	212	11.7 工作实训营.....	274
10.5.4 菜单类组件.....	218	11.7.1 训练实例.....	274
10.5.5 对话框.....	225	11.7.2 工作实践和面试中常见问题 解析.....	279
10.5.6 表格.....	231	习题.....	279
10.6 回到工作场景.....	236	第 12 章 Java 的应用	281
10.7 工作实训营.....	242	12.1 工作场景导入.....	282
10.7.1 训练实例.....	242	12.2 JDBC 的概念及利用 JDBC 访问 数据库.....	282
10.7.2 工作实践和面试中常见问题 解析.....	249	12.2.1 JDBC 的概念.....	282
习题.....	250	12.2.2 利用 JDBC 访问数据库.....	284
第 11 章 Applet 程序设计	253	12.2.3 实例训练.....	285
11.1 工作场景导入.....	254	12.2.4 执行 SQL 语句.....	290
11.2 Applet 概述.....	254	12.2.5 使用 PreparedStatement.....	292
11.2.1 Applet 的概念.....	254	12.2.6 事务处理.....	294
11.2.2 Applet 的生命周期概念.....	255	12.3 J2ME 平台.....	295
11.2.3 Applet 的类层次结构.....	257	12.4 J2SE 平台.....	298
11.2.4 Applet 类 APL 概述.....	257	12.5 J2EE 平台.....	299
11.2.5 Applet 的关键方法.....	258	12.6 回到工作场景.....	301
11.2.6 显示 Applet.....	259	12.7 工作实训营.....	314
11.3 编写 Applet.....	259	12.7.1 训练实例.....	314
11.3.1 Applet 编写的步骤.....	259	12.7.2 工作实践和面试中常见问题 解析.....	317
11.3.2 用户 Applet 类的定义.....	259	习题.....	318
11.3.3 在 HTML 页中包含 Applet.....	260	附录 各章习题参考答案	323
11.4 Applet 中的图形化用户界面 GUI.....	260	参考文献	347
11.4.1 基于 AWT 构件的 Applet 用户界面.....	260		
11.4.2 基于 Swing 的 Applet 用户界面.....	262		

第 1 章

Java 语言概论



本章要点

- Java 的发展简史及特点
- J2SDK 的下载与安装
- Java 应用程序的编写
- Eclipse 的下载及使用



技能目标

- 正确安装使用 J2SDK
- 使用记事本开发 Java 应用程序
- 在 Eclipse 上建立 Java 项目(Project)



1.1 工作场景导入

【工作场景】

为系统设计欢迎提示界面，要求如下：显示“HelloWorld”欢迎语，并显示当前的时间，结果如图 1.1 所示。

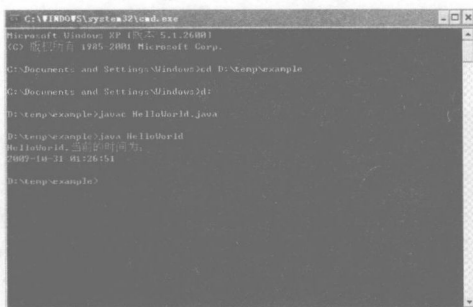


图 1.1 显示 HelloWorld 及时间

【工作场景分析】

该应用程序首先需要有输出“HelloWorld, 当前的时间为:”的欢迎语，然后另起一行输出当前的时间。所以我们的工作为：先编写源程序，然后编译运行 Java 应用程序。

【引导问题】

- (1) 什么是 Java?
- (2) 如何下载和安装 J2SDK?
- (3) 怎样用记事本开发 Java 程序?
- (4) Eclipse 是什么? 怎样使用它来编写程序?
- (5) 如何编译和运行自己的 Java 程序?



1.2 Java 语言简介

1.2.1 什么是 Java

Java 是一种简单的、跨平台的、面向对象的、分布式的、解释的、健壮的、安全的、结构的、中立的、可移植的、性能很优异的、多线程的、动态的语言。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?

Java 语言其实最早诞生于 1991 年，起初被称为 Oak 语言，是 Sun 公司为开发一些消费性电子产品而设计的一个通用环境。最初的目的只是为了开发一种独立于平台的软件技术，

而且在网络出现之前, 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死板乏味的 HTML 文档。这对于那些迷恋于 Web 浏览的人们来说简直不可容忍。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 上创建一类无需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 年正式推出 Java 的时候, 几乎所有的 Web 开发人员都想到: 噢, 这正是我想要的。于是 Java 成了一颗耀眼的明星, 丑小鸭一下子变成了白天鹅。

1.2.2 Java 的特点

1. 平台无关性

平台无关性是指 Java 能运行于不同的平台。Java 引进虚拟机原理, 并运行于虚拟机中, 实现在不同平台之间, 使用 Java 编写的程序能在世界范围内共享。Java 的数据类型与机器无关, Java 虚拟机(Java Virtual Machine)是建立在硬件和操作系统之上的, 它实现了 Java 二进制代码的解释执行功能, 提供了不同平台的接口。

2. 安全性

Java 的编程类似 C++, 学习过 C++ 的读者将很快掌握 Java 的精髓。Java 舍弃了 C++ 的指针对存储器地址的直接操作, 程序运行时, 内存由操作系统分配, 这样可以避免病毒通过指针侵入系统。Java 对程序提供了安全管理器, 防止程序的非法访问。

3. 面向对象

Java 吸取了 C++ 面向对象的概念, 将数据封装于类中, 利用类的优点, 实现了程序的简洁性和便于维护性。类的封装性、继承性等有关对象的特性, 使程序代码只需一次编写, 然后可通过上述特性反复利用。程序员只需把主要精力用在类和接口的设计和应用上。Java 提供了众多的一般对象的类, 通过继承即可使用父类的方法。在 Java 中, 类的继承关系是单一的、非多重的, 一个子类只有一个父类, 子类的父类又有一个父类。Java 提供的 Object 类及其子类的继承关系如同一棵倒立的树形, 根类为 Object 类, Object 类功能强大, 在程序开发中经常会使用到它及其派生的子类。

4. 分布式

Java 建立在扩展 TCP/IP 网络平台上。库函数提供了用 HTTP 和 FTP 协议传送和接受信息的方法。这使得程序员使用网络上的文件如同使用本机文件一样容易。

5. 健壮性

Java 致力于检查程序在编译和运行时的错误。类型检查能够帮助检查出开发早期出现的许多错误。Java 自己操纵内存, 减少了内存出错的可能性。Java 还实现了真数组, 避免

了覆盖数据的可能, 这些功能特征大大缩短了开发 Java 应用程序的周期。并且 Java 还提供了 Null 指针检测、数组边界检测、异常出口、Byte Code 校验等功能。



1.3 Java 开发环境的构建

1.3.1 JDK 的安装

Sun 公司免费提供了 Java 开发工具包(Java Development Kit, JDK)。该工具包包含了编译、运行以及调试程序所需要的工具, 此外, 还提供了大量的基础类库。基础类库是应用开发中的砖和瓦, 开发人员只要灵活地使用, 就可建造出各种各样的建筑物——应用软件。

Sun 公司为不同的操作系统(如 Windows、Linux/Unix、Mac OS)提供了相应的 Java 开发包安装程序。本书中使用 Windows 操作系统环境下的 Java 开发包。读者可以登录 Sun 公司的网站(<http://www.sun.com>)获取免费的 Java 开发包安装程序。本书中所给出的例子程序均在版本为 1.6.0_10 的 JDK 下运行通过。具体安装过程如下。

(1) 在得到 Java 开发包后首先需要进行安装。双击 Java 开发包安装程序, 出现的安装界面如图 1.2 所示。

(2) 安装程序会弹出许可证协议阅读界面, 如图 1.3 所示, 阅读相应条款后, 单击“接受”按钮继续安装。

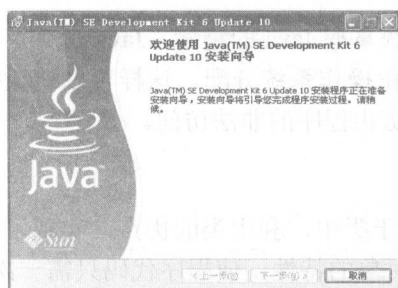


图 1.2 安装启动界面

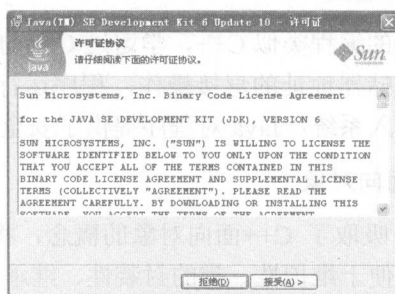


图 1.3 许可证协议阅读界面

(3) 用户可以选择安装开发包的部分或全部内容, 如图 1.4 所示。

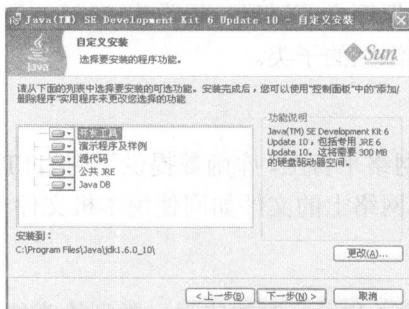


图 1.4 安装内容选择

(4) 如果用户想要更改开发包的安装路径,则单击“更改”按钮,弹出如图 1.5 所示的对话框。更改了安装路径后,单击“确定”按钮可关闭该对话框。

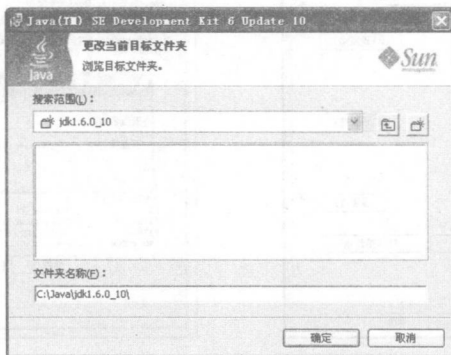


图 1.5 安装路径选择

(5) 在选定了安装内容和安装路径后,单击“下一步”按钮进入正式安装。其他版本的开发包安装过程中除了显示界面略有不同,别的方面类似。

1.3.2 环境变量的设定

设定环境变量的目的是为了能够正常使用所安装的开发包。配置的 Java 环境变量包括 JAVA_HOME、Path 和 CLASSPATH,其中 JAVA_HOME 为 JDK 安装路径(Java 安装目录),CLASSPATH 为 Java 加载类(Class 或 Lib)路径,Path 为 Java 编译程序和运行程序的目录。

设置环境变量的方法如下。

- (1) 在桌面上右击“我的电脑”图标,在弹出的快捷菜单中选择“属性”命令,打开“系统属性”对话框,切换至“高级”选项卡,如图 1.6 所示。
- (2) 然后单击“环境变量”按钮,打开“环境变量”对话框。
- (3) 新建 JAVA_HOME 环境变量:单击“新建”按钮,弹出“新建用户变量”对话框,设置变量名为“JAVA_HOME”,变量值为“C:\Program Files\Java\jdk1.6.0_1”,然后单击“确定”按钮。
- (4) 新建 CLASSPATH 变量:单击“新建”按钮,弹出“新建用户变量”对话框,设置变量名为“CLASSPATH”,变量值为“.;%JAVA_HOME%\lib;%JAVA_HOME%\lib\tools.jar”,然后单击“确定”按钮。
- (5) 编辑 Path 变量。在“系统变量”列表中双击变量名为 path 的变量,弹出“编辑系统变量”对话框,在“变量值”文本框中把“%JAVA_HOME%\bin;%JAVA_HOME%\jre\bin”添加在原变量值的开头处,如图 1.7 所示。

Java 环境变量已经设置好了,现在我们可以测试一下。选择“开始”→“运行”命令,在弹出的“运行”对话框中输入“cmd”,单击“确定”按钮,即可进入 DOS 控制台窗口,如图 1.8 所示。需要注意的是,用这种方式进入 DOS 控制台窗口后,当前目录为系统默认目录。如果想转换到其他盘符,例如 D 盘,可以在提示符后输入“D:”,然后按 Enter 键即可。