

• 从初学到精通系列 •

Broadview®
www.broadview.com.cn

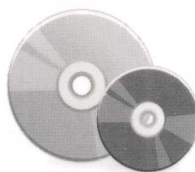
- ❶ 循序渐进：从入门开始，由浅入深，逐步深入地讲解
- ❷ 实例丰富：讲解技术点时列举了大量实例，实用性强
- ❸ 注释详细：书中的实例源代码给出了大量注释，便于阅读
- ❹ 案例精讲：提供了3个综合案例帮助读者提高开发水平
- ❺ 视频讲解：配大量多媒体教学视频进行讲解，学习效率高

Java

从初学到精通



辛立伟 张帆 等编著



53小时
多媒体教学视频



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

• 从初学到精通系列 •

Java

从初学到精通

辛立伟 张帆 等编著

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

Java 不但是最为流行的计算机语言，而且是优秀的纯面向对象语言。本书针对初学者和自学者的特点，以通俗易懂的语言介绍 Java 语言，力求做到深入浅出，将复杂的概念用简洁浅显的语言娓娓道来。

全书内容分为 5 篇，共计 27 章，涵盖了 Java 程序设计语言基础、Java 面向对象基础、Java 图形化界面编程技术、Java 网络应用程序开发技术、Java 数据库应用程序开发技术等方面的内容。为了便于读者学习和实际应用，本书最后一篇通过对几个完整项目案例的分析，详细讲解了使用 Java 语言开发游戏程序、网络程序和数据库应用程序的完整过程，有利于初学者锻炼使用 Java 语言解决实际问题的能力。

本书提供有大量翔实的编程实例和丰富的项目案例，通过对项目的实现和讲解，以提高读者解决实际问题的能力，使读者逐步具备利用 Java 语言来开发应用程序的能力，特别适合于想学习 Java 语言的初学者和大中专院校学生阅读，也可作为有一定 Java 基础的软件开发人员和其他有关人员的参考资料。

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 从初学到精通 / 辛立伟等编著. —北京：电子工业出版社，2010.6

（从初学到精通系列）

ISBN 978-7-121-10648-4

I. ①J… II. ①辛… III. ①JAVA 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字（2010）第 058032 号

责任编辑：高洪霞

印 刷：北京东光印刷厂

装 订：三河市皇庄路通装订厂

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：38.75 字数：1013 千字

印 次：2010 年 6 月第 1 次印刷

印 数：4000 册 定价：69.00 元（含 DVD 光盘 1 张）

凡所购买电子工业出版社图书如有缺损问题，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。

从此初学，走向精通

*The book is the nourishment in the whole world.
Life the inside have no book, seem to have no sunlight;
The inside of the intelligence has no book, seem the bird have no wing.*

书籍是全世界的营养品。生活里没有书籍，
就好像没有阳光；智慧里没有书籍，就好像鸟儿没有翅膀。

——莎士比亚



为什么要写这样一本书

Java 语言自 1995 年诞生以来，经过十多年的发展和应用，技术日益成熟，现已有 Java SE、Java EE 和 Java ME 三大平台。Java 已成为当今网络计算平台的标准，也成为全球程序员的首选开发平台。今天，Java 技术已无处不在，小至信用卡，大到企业级应用系统、美国宇航系统等。到 2008 年，全球已有超过 15 亿台手机和手持设备应用 Java 技术。Java 技术正显示出高速的增长力，Java 技术所驱动的业务每年都在 1000 亿美元以上。同时，Java 技术以其跨平台特性和良好的可移植性，以及平缓的学习曲线，成为广大软件开发技术人员的挚爱。

为了方便初学者学习 Java 语言，笔者花费半年多时间写作这本书。本书全面地介绍了 Java 语言基础，以实例介绍了 Java 语言的编程技术，以实际项目案例讲解了 Java 应用程序的开发过程。学完本书之后，力求让读者既掌握到 Java 语言的精髓，又有项目实践的本领。

本书的特点

1. 细致体贴的讲解

本书假定读者没有任何的编程技能基础，讲解时尽可能地站在读者的角度，循序渐进，由浅入深，用准确的语言总结概念►用直观的图示演示过程►用详细的注释解释代码►用形象的比方帮助记忆。效果如下：

① **知识点介绍** 全面、准确、清晰是其显著特点，一般放在每一节开始位置，让零基础的读者了解相关概念，顺利入门。

② **范例** 本书精心选择的编程示例，可实践性强，注意让读者能从实例运用中举一反三，既有利于初学者对基本知识点的理解和掌握，又锻炼了读者的编程能力。

③ **范例代码** 与范例编号对应，层次清楚、语句简洁、注释丰富，体现了代码优美的原则，有利于读者养成良好的代码编写习惯。读者可以直接根据这些源程序快速编写程序，直接切入相关应用。

④ **运行结果** 对范例给出运行结果和对应图示，帮助读者更直观地理解范例代码、验证结果。

⑤ **代码解析** 本书对范例代码中的关键代码配以图示并进行详细解释，有助于读者掌握相关概念和知识。

⑥ 贴心的提示 为了便于读者阅读，全书还穿插着一些技巧、提示等小贴士，体例约定如下：

- 提示：通常是一些贴心的提醒，让读者加深印象或提供建议，或者解决问题的方法。
- 注意：提出学习过程中需要特别注意的一些知识点和内容，或者相关信息。
- 警告：对操作不当或理解偏差将会造成的灾难性后果做警示，以加深读者印象。

⑦ 小结 每章最后提供对本章所介绍知识的简单总结，有利于读者加深印象，巩固总结。

经作者多年的培训和授课证明，以上讲解方式是最适合读者学习的方式，读者按照这种方式，会非常轻松、顺利地掌握本书知识。

1

2

3

4

5

Java从初学到精通

第19章 异常处理

19.4 自定义异常

尽管 Java 的内置异常能够处理大多数常见错误，但有时还可能出现系统没有考虑到的异常，因此我们仍然希望建立自己的异常类型，来处理所遇到的特殊情况。通过继承 Exception 类或它的子类，实现自定义异常类。对于自定义异常，必须采用 throw 语句抛出异常。这种类型的异常不会自行产生。

19.4.1 创建自定义异常类

使用 Java 内置的异常类可以描述在编程时出现的大部分异常情况。除此之外，用户还可以自定义异常。所有的用户自定义异常类都必须由 Exception 类或者 Exception 类的子类派生，所以必须显示指明异常类的基类。自定义异常的基本形式如下所示。

【范例 19-2】本范例演示了如何查看 student 的约束。

代码 19-2 带脚本的 HTML 文件：script.html

<html>
<head>
<script language="javascript">
<!--
alert(new Date());
-->
</script>
<title>带脚本的 HTML</title>
</head>
<body>
<script>
</body>
</html>

<!-- 脚本代码 -->

当用浏览器来浏览网页的时候，会弹出一个显示时间的小对话框，如图 19.19 所示。

图 19.19 JavaScript 文件的运行结果

【代码解析】

在上述代码中，<script></script>标签对中的代码为脚本代码，函数 alert 作用就是在浏览器窗口中弹出一个对话框，而代码 new Date（即计算机时间）则是对对话框上显示的内容。网页中有三个地方编写 JavaScript 脚本代码，它们分别是：
① 网页文件的 <script></script> 标签对中直接编写脚本程序代码。
② 网页文件中引用一个脚本程序文件，而脚本代码却单独放置在引用文件中。

406

2

3

4

5

6

7

将脚本代码作为某个元素的事件属性或超链接的 href 属性值。

【例 19-8】使用自定义异常示例。

import java.util.*;
public class Exp12 {
 public static void main(String[] args) {
 Scanner in=new Scanner(System.in); //创建一个对象，用于读取用户输入
 int number1=in.nextInt(); //从键盘获得输入
 int number2=in.nextInt(); //从键盘获得输入
 try {
 int result = avg(number1, number2); //调用方法 avg()
 System.out.println(result);
 } catch (MyException e) {
 System.out.println(e);
 }
 }
 super(ErrorMessage);
}

该程序运行结果如图 19.15 所示。

图 19.15 例 19.8 的运行结果

如果 path 变量在“系统变量”列表中已经存在，这时候就需要选择该变量名，然后单击“编辑”按钮就会弹出“新建系统变量”对话框，最后在“变量值”文本框中末尾添加一个“%”符号才可以把“%JAVA_HOME%bin”添上。

在上面的程序中，自定义了一个异常类 MyException，它是 Exception 的子类。在方法 avg() 的定义中，使用关键字 throws 声明该方法可能会抛出 MyException 类型的异常，并且在 avg() 方法内部，当一定的条件满足时（“number1<0||number2<0”或“number1>100||number2>100”），使用关键字 throw 创建一个 MyException 类型的异常对象并抛出。在 main() 方法中调用了 avg() 方法，所以必须处理 avg() 方法可能抛出的异常。在这里为使用 catch 语句捕获它并输出异常信息。

19.5 小结

本章主要向读者介绍了 Java 中的异常处理机制。异常处理机制是保证 Java 程序正常运行、具有较高安全性的重要手段。初学良好的编程习惯是非常重要的。读者在学习本章内容时，把重点放在对 Java 程序中的异常处理上。而在向读者介绍 Java 程序中的输入和输出方法。

407

2. 实用超值的 DVD 光盘

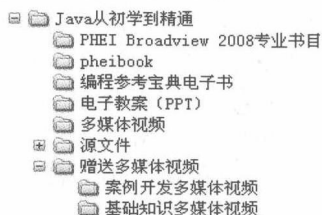
为了帮助读者比较直观地学习，本书附赠 DVD 光盘，内容包括多媒体视频、电子教案（PPT）、编程参考宝典电子书、各章习题答案和实例源代码等。

● 多媒体视频

配有长达 12 小时手把手教学视频，讲解关键知识点界面操作和书中的一些综合练习题。作者亲自配音、演示，手把手教会读者使用。此外，还随书免费赠送 41 小时相关知识多媒体视频，帮助读者巩固和加深对所学内容的理解和掌握。

● 电子教案（PPT）

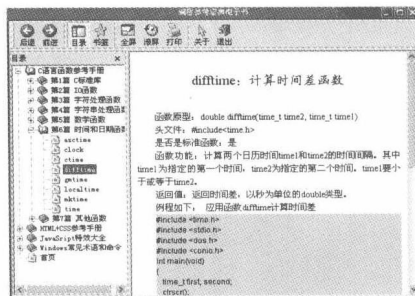
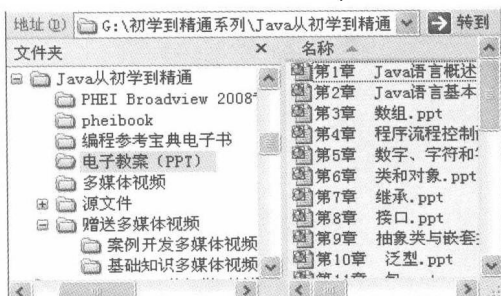
本书可以作为高校相关课程的教材或课外辅导书，所以笔者特别为本书制作了电子教案（PPT），以方便老师教学使用。





● 编程参考宝典电子书

为方便广大读者学习，特别制作了编程开发参考电子书，供读者查阅和参考。



3. 提供完善的技术支持

本书提供了论坛：<http://www.rzchina.net>，读者可以在上面提问交流。另外，论坛上还有一些小的教程、视频动画和各种技术文章，可帮助读者提高开发水平。

4. 丰富的额外素材下载

相关的开发素材文件，在 www.broadview.com.cn 提供下载。

本书适合哪些读者阅读

- 想从事软件开发的入门者
- 有一点经验的开发人员
- 从其他语言迁移过来的开发人员
- 大中专院校的学生
- 社会培训学生

本书作者

本书由辛立伟、张帆主笔编写，其他参与编写的人员有昊燃、方振宇、陈冠佐、傅奎、陈勤、梁洋洋、毕梦飞、陈庆、柴相花、陈非凡、陈华、陈嵩、承卓。

编者

目 录

第 1 篇 Java 语言基础篇

第 1 章 Java 语言概述.....	30
----------------------	----

本章介绍了 Java 语言的背景、特点、开发环境、开发过程及开发工具的使用。在学习本章内容时，要重点理解 Java 技术的两个含义：Java 程序设计语言与 Java 平台。重点掌握如何搭建 Java 程序的开发环境，包括下载及安装 Java 开发工具包 JDK、配置环境变量及 JDK 环境测试等。

1.1 Java 语言简介.....	30
1.1.1 Java 语言的起源.....	30
1.1.2 使用 Java 语言的优点.....	31
1.2 Java 语言与 Java 平台.....	32
1.2.1 Java 程序设计语言.....	32
1.2.2 Java 平台.....	32
1.3 搭建 Java 程序开发环境.....	33
1.3.1 系统要求.....	33
1.3.2 下载 Java 程序开发工具包 JDK.....	34
1.3.3 安装 JDK.....	35
1.3.4 在 Windows 系统下配置 JDK.....	36
1.4 开发第一个 Java 应用程序.....	37
1.4.1 创建第一个 Java 应用程序源文件.....	37
1.4.2 将 HelloWorldApp.java 源文件编译为.class 文件.....	38
1.4.3 运行 HelloWorldApp 应用程序.....	39
1.4.4 Java 应用程序的基本结构.....	40
1.5 小结.....	41

第 2 章 Java 语言基本语法.....	42
------------------------	----

本章深入学习了 Java 语言的基本语法，重点对常量与变量及数据类型和运算符做了详细的讲述。在本章的学习过程中，可能对 Java 数据类型的转换难以理解，因此读者一定要多做练习，通过上机练习体会和掌握 Java 的基本语法。

2.1 标识符和关键字.....	42
2.1.1 标识符.....	42
2.1.2 关键字.....	42
2.1.3 标识符命名规则.....	43

2.2	常量与变量	43
2.2.1	常量的概念及常量声明	43
2.2.2	变量的概念及变量声明	44
2.3	基本数据类型	45
2.3.1	数据类型	45
2.3.2	整数类型	45
2.3.3	浮点类型	47
2.3.4	字符类型	47
2.3.5	逻辑类型	49
2.3.6	不同数据类型间的转换	49
2.4	运算符与表达式	51
2.4.1	算术运算符与算术表达式	51
2.4.2	赋值运算符与赋值表达式	52
2.4.3	自增和自减运算符	53
2.4.4	关系运算符和关系表达式	54
2.4.5	逻辑运算符和逻辑表达式	54
2.4.6	位运算符	57
2.4.7	三元运算符	59
2.4.8	运算符的优先级	60
2.5	小结	60
第3章	数组	61
本章深入学习了 Java 语言的数组，了解了 Java 语言中数组的概念，学习了如何声明、创建、访问和修改数组。本章重点是一维数组和二维数组。其中对于数组的创建及二维数组的创建和使用是本章学习的难点，建议读者多上机练习。		
3.1	数组的概念与特点	61
3.1.1	什么是数组	61
3.1.2	Java 语言中数组的特点	62
3.2	一维数组	62
3.2.1	声明一维数组变量	62
3.2.2	创建一维数组对象	63
3.2.3	访问一维数组元素	63
3.2.4	修改一维数组元素	64
3.3	多维数组	65
3.3.1	声明二维数组变量	65
3.3.2	创建二维数组对象	66
3.3.3	访问二维数组元素	67

3.3.4 修改二维数组元素	67
3.4 小结	68
第4章 程序流程控制语句	69
本章深入介绍了有关程序流程控制的语句，主要包括选择语句、循环语句和跳转语句。其中的 while 循环语句因其灵活方便，读者应重点掌握。本章最后讲到的跳转语句主要用于提高循环语句的灵活性，特别是标签 break 语句，是本章掌握的难点。	
4.1 选择语句	69
4.1.1 if 选择语句	69
4.1.2 if-else 双分支选择语句	70
4.1.3 if-else-if 多分支选择语句	71
4.1.4 选择语句的嵌套	74
4.1.5 switch 多分支开关语句	76
4.1.6 if 选择语句与 switch 开关语句的区别	78
4.2 循环语句	78
4.2.1 while 循环语句	78
4.2.2 do-while 循环语句	80
4.2.3 for 循环语句	83
4.2.4 循环语句的嵌套	85
4.2.5 使用增强型 for 语句遍历一维数组中的元素	87
4.2.6 对一维数组进行排序	88
4.2.7 使用增强型 for 语句遍历二维数组中的元素	89
4.3 跳转语句	90
4.3.1 break 跳转语句	90
4.3.2 continue 跳转语句	93
4.4 实例	94
4.4.1 实例 1：译密码	95
4.4.2 实例 2：打印九九乘法表	96
4.5 小结	97
第5章 数字、字符和字符串处理	98
本章深入介绍了 Java 语言中有关数字、字符和字符串的处理的相关知识，重点讲解了 java.lang 包中用来处理数字的 Number 及其子类、专门用来处理数学运算的 Math 类、专门用来处理字符的 Character 类，以及专门用来创建和操作字符串的类 String。	
5.1 数字	98
5.1.1 Numbers 数字包装器类	98
5.1.2 数字的格式化输出	100

5.1.3	Math 数学运算处理类	104
5.2	字符	107
5.2.1	Character 字符包装器类	107
5.2.2	转义字符序列	108
5.3	使用 String 类	109
5.3.1	创建字符串对象	109
5.3.2	获取字符串的长度	110
5.3.3	连接字符串	110
5.3.4	字符串与字符数组	112
5.3.5	在一个字符串中操纵字符	113
5.3.6	字符串比较和字符串部分内容比较	117
5.4	使用 StringBuilder 类	120
5.4.1	认识 StringBuilder 类	120
5.4.2	StringBuilder 对象的长度和容量	121
5.4.3	StringBuilder 类提供的操作方法	121
5.5	实例	123
5.5.1	实例 1: 用户登录验证程序	123
5.5.2	实例 2: 敏感词过滤程序	125
5.6	小结	126

第 2 篇 Java 面向对象篇

第 6 章	类和对象	128
-------	------	-----

本章主要学习了类和对象的相关概念,包括类的声明、类的实例(即对象)。本章所讲的知识是 Java 面向对象语言中最基本的概念,所涉及的类和声明、对象的构造、访问控制修饰符都是读者学习的重点。其中的构造器的工作过程、访问控制和 this 关键字的使用是本章学习的难点。

6.1	面向对象简介	128
6.1.1	什么是对象	128
6.1.2	什么是类	130
6.1.3	什么是继承	131
6.1.4	什么是接口	131
6.1.5	什么是包	132
6.2	定义类	132
6.2.1	类的基本结构	132
6.2.2	声明类的成员变量	134
6.2.3	声明类的成员方法	135
6.2.4	成员方法命名	136
6.2.5	方法重载	137

6.2.6	调用成员方法	137
6.2.7	为类提供构造器	139
6.3	向方法或构造器内传递信息	140
6.3.1	使用参数	140
6.3.2	实参与形参的关系	141
6.3.3	参数类型	143
6.3.4	传递任意数量的参数	143
6.3.5	参数名称	144
6.3.6	传递原始数据类型参数	144
6.3.7	传递引用数据类型参数	145
6.4	对象	145
6.4.1	对象实例	145
6.4.2	创建对象	147
6.4.3	使用对象	150
6.5	方法的返回值	152
6.5.1	返回值	152
6.5.2	返回类或接口	153
6.6	使用 <code>this</code> 关键字	153
6.6.1	对字段使用 <code>this</code>	154
6.6.2	对构造器使用 <code>this</code>	154
6.7	控制对类的成员的访问	155
6.8	理解实例成员和类成员	156
6.8.1	类变量	156
6.8.2	类方法	158
6.8.3	常量	158
6.8.4	实例: <code>AutoCar</code> 类	158
6.9	初始化字段	159
6.9.1	静态初始化块	159
6.9.2	初始化实例成员	160
6.10	枚举类型	161
6.11	标注	164
6.11.1	标注的用法	164
6.11.2	文档标注	164
6.11.3	预定义标注	165
6.11.4	标注处理	168
6.12	小结	168

第7章 继承 169

本章主要学习了类的继承。这是面向对象语言的三大特征之一。本章学习的重点是继承中方法的覆盖和方法的重写、super 关键字的使用及构造器链。读者在学习本章内容时对 super 关键字的使用及构造器链的调用要细心体会，这是本意学习的难点。

7.1 继承概述	169
7.1.1 继承的概念	169
7.1.2 Java 平台中的类层次	169
7.1.3 一个继承的示例	170
7.1.4 在子类中可以做的事情	171
7.2 从子类中访问父类的私有成员	172
7.3 对象类型转换	173
7.3.1 隐式对象类型转换	173
7.3.2 强制对象类型转换	174
7.3.3 使用 instanceof 运算符	174
7.4 覆盖和隐藏超类中的方法	174
7.4.1 覆盖超类中的实例方法	175
7.4.2 协变覆盖	175
7.4.3 隐藏超类中的类方法	176
7.4.4 方法覆盖和方法隐藏后的修饰符	178
7.4.5 方法覆盖和方法隐藏总结	178
7.5 隐藏超类中的字段	179
7.6 使用 super 关键字访问超类中的成员	180
7.6.1 调用超类中被覆盖的方法	180
7.6.2 访问超类中被隐藏的字段	180
7.7 使用 super 关键字调用超类的构造器	181
7.7.1 调用超类的无参构造器	181
7.7.2 调用超类的带参构造器	183
7.7.3 构造器链	184
7.8 Object 类	185
7.8.1 作为超类的 Object 类	185
7.8.2 Object 类的对象克隆方法 clone()	186
7.8.3 Object 类的对象比较方法 equals()	186
7.8.4 Object 类的对象清除方法 finalize()	187
7.8.5 Object 类的获取对象类信息的方法 getClass()	187
7.8.6 Object 类的获得对象内存地址的方法 hashCode()	188
7.8.7 Object 类的对象字符串表示方法 toString()	188
7.9 编写 final 类和 final 方法	188
7.9.1 使用 final 关键字声明 final 方法	188

7.9.2 使用 final 关键字声明 final 类	189
7.10 小结	189
第 8 章 接口	190
本章主要介绍了接口的概念、接口的声明及接口的实现。在学习本章内容时，读者重点理解接口的概念和学习如何实现接口。在将接口作为类型使用上，读者可能会感到难以理解，需用心体会。	
8.1 理解接口	190
8.1.1 一般接口的概念	190
8.1.2 Java 中的接口	191
8.1.3 在 Java 中定义接口	191
8.1.4 在 Java 中使用接口	191
8.1.5 作为 API 的接口	192
8.1.6 接口和多继承	192
8.2 定义接口	193
8.2.1 接口声明	193
8.2.2 接口体	193
8.3 实现接口	194
8.3.1 实现接口的语法	194
8.3.2 接口实例：Relatable	195
8.3.3 实现接口 Relatable	195
8.3.4 编译并运行带有接口的实例	196
8.3.5 实现多个接口时的常量和方法冲突问题	197
8.4 将接口作为类型使用	198
8.5 改写接口	199
8.6 小结	200
第 9 章 抽象类与嵌套类	201
本章介绍了两种特殊的类型：抽象类和嵌套类。抽象类是实现多态的一种途径，嵌套类主要应用在事件监听器的构建上。其中抽象类和接口的异同是读者学习的一个重点和难点。嵌套类的定义也是学习中容易迷惑的地方。建议读者在学习到这部分内容时，如果暂时不能理解的话，不妨先放一放。等学过后面的事件监听机制以后，再回来学习这部分内容，相信就容易理解了。	
9.1 抽象类和抽象方法	201
9.1.1 抽象方法	201
9.1.2 抽象类	201
9.1.3 抽象类与接口对比	202
9.1.4 抽象类实例	202
9.1.5 抽象类实现接口	203

9.1.6	抽象类的类成员	204
9.2	嵌套类	204
9.2.1	为什么要使用嵌套类	205
9.2.2	静态嵌套类	205
9.2.3	内部类	205
9.2.4	内部类实例	206
9.2.5	局部内部类和匿名内部类	207
9.2.6	内部类的修饰符	207
9.3	小结	207
第 10 章	泛型	208
本章主要描述了什么是泛型、为什么要使用泛型及如何使用泛型。读者在学习本章时，重点理解什么是泛型及为什么要使用泛型。读者可能在学习本章的过程中对如何使用泛型还存在迷惑，没有关系，完全可以先不去理会它。在具体使用泛型最多的地方一般是在 Java 的集合框架部分，所以读者可以在学习 Java 集合框架部分内容时，再回来结合本章的内容来掌握泛型的使用。		
10.1	泛型简介	208
10.1.1	为什么要使用泛型	208
10.1.2	一个简单的 Box 类	208
10.2	泛型类型	210
10.2.1	Box 类的泛型版本	210
10.2.2	类型参数命名惯例	212
10.3	泛型方法和泛型构造器	212
10.4	限定的类型参数	213
10.5	泛型子类型	214
10.6	使用通配符	215
10.7	类型擦除	216
10.8	小结	217
第 11 章	包	218
本章主要介绍了 Java 语言中包的概念及如何创建和使用包。包是 Java 组织资源的一种方式，读者学习起来应该不会感到吃力。只是如何将包对应到具体的物理路径上，读者应该细心体会。因为 Java 平台的许多实现依赖于分组的文件系统来管理源文件和 class 文件的，包名往往对应着系统中实际的目录名，这一点读者一定要注意。		
11.1	使用包	218
11.2	创建包	219
11.2.1	创建包的语句	219
11.2.2	命名包	220

11.2.3 包命名惯例	220
11.3 使用包中的成员	221
11.3.1 通过全限定名引用包成员	221
11.3.2 导入包中的成员	221
11.3.3 导入整个包	222
11.3.4 包的层级	222
11.3.5 包成员名称冲突	223
11.3.6 静态导入语句	223
11.4 管理源文件和 class 类文件	224
11.4.1 管理实现策略	224
11.4.2 设置 CLASSPATH 系统变量	225
11.5 小结	225
第 12 章 集合	226

本章主要学习了 Java 集合框架，包括接口、实现和算法。其中核心集合接口是 Java 集合框架的基础。实现是用于存储集合的数据对象，是实现了集合接口的类。

12.1 Java 集合框架概述	226
12.1.1 Java 集合框架	226
12.1.2 使用 Java 集合框架的益处	227
12.2 核心集合接口	227
12.3 Collection 接口	229
12.3.1 转换构造器	229
12.3.2 Collection 接口的定义	229
12.3.3 Collection 接口的基本操作	230
12.3.4 遍历 Collection 接口	230
12.3.5 Collection 接口的批量操作	231
12.3.6 Collection 接口的数组操作	231
12.4 Set 接口	232
12.4.1 Set 接口的定义	232
12.4.2 Set 接口的基本操作	233
12.4.3 Set 接口的批量操作	234
12.4.4 Set 接口的数组操作	235
12.5 List 接口	235
12.5.1 List 接口的定义	235
12.5.2 List 接口与 Vector 的比较	236
12.5.3 从 Collection 继承的操作	236
12.5.4 按位置访问和查找操作	237

12.5.5	List 迭代器	238
12.5.6	子集合操作	240
12.6	Queue 接口	242
12.6.1	Queue 接口的定义	242
12.6.2	Queue 接口的操作	242
12.6.3	Queue 接口应用	243
12.7	Map 接口	243
12.7.1	Map 接口的定义	243
12.7.2	Map 接口的基本操作	244
12.7.3	Map 接口的批量操作	245
12.7.4	集合视图	245
12.8	SortedSet 接口	246
12.8.1	SortedSet 接口的定义	246
12.8.2	SortedSet 接口的操作	247
12.8.3	SortedSet 接口的标准构造器	247
12.8.4	子集合操作	247
12.8.5	端点操作	248
12.9	SortedMap 接口	248
12.9.1	SortedMap 接口的定义	249
12.9.2	SortedMap 接口的操作	249
12.9.3	SortedMap 接口的标准构造器	249
12.9.4	与 SortedSet 接口的比较	249
12.10	实现	250
12.10.1	实现的类型	250
12.10.2	Set 接口的实现	251
12.10.3	List 接口的实现	252
12.10.4	Map 接口的实现	252
12.10.5	Queue 接口的实现	253
12.11	算法	253
12.11.1	Collections 类	254
12.11.2	排序	254
12.11.3	乱序	254
12.11.4	常规数据处理	255
12.11.5	查找	255
12.11.6	组成	255
12.11.7	查找极值	255
12.12	小结	256