



H

V

更好理解、更易上手、更适合新手的Java入门宝典

◎ 宁楠 著

Java 零基础 实战



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS

◎ 宁楠 著

Java 实战

零基础

坚持学习 持续精进



宁楠

人民邮电出版社
北京

图书在版编目 (C I P) 数据

Java零基础实战 / 宁楠著. -- 北京 : 人民邮电出版社, 2019. 5

ISBN 978-7-115-50895-9

I. ①J… II. ①宁… III. ①JAVA语言—程序设计
IV. ①TP312.8

中国版本图书馆CIP数据核字(2019)第037865号

内 容 提 要

Java 是一门经典的面向对象的编程语言, 具有很强的通用性、平台可移植性和安全性, 并且一直在编程语言排行榜上稳居前列。本书旨在通过浅显易懂的语言, 讲清 Java 的特性, 帮助读者掌握面向对象编程的核心思想; 同时, 通过丰富实用的代码示例, 帮助读者快速上手, 轻松掌握 Java 语言。

本书分 11 章, 带领读者从零开始认识 Java, 了解 Java, 会用并且用好 Java。书中不仅对 Java 的运行机制、体系结构和基本的安装配置进行了讲解, 还对面向对象编程的思想进行了深刻的剖析和总结, 同时作者以丰富的代码示例演示了基本的 Java 编程。除此之外, 本书还对 Java 的集合框架、多线程、实用类、IO 流、反射机制和 Web 编程等进行了讲解, 让读者在掌握基本编程技巧的基础上, 进一步探索 Java 的强大功能。

本书由资深 Java 开发老手编写, 汇集了丰富的实践经验和实用的编程思想, 非常适合想入门 Java 的新手学习, 也适合作为计算机相关专业的辅助教程, 其他非科班出身的想学习 Java 编程的读者也可以参考学习。

-
- ◆ 著 宁楠
责任编辑 胡俊英
责任印制 焦志炜
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
三河市祥达印刷包装有限公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 21.5
字数: 505 千字
印数: 1-3 000 册
- 2019 年 5 月第 1 版
2019 年 5 月河北第 1 次印刷
-

定价: 69.80 元

读者服务热线: (010)81055410 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京东工商广登字 20170147 号



作者简介

宁楠，资深Java开发工程师，拥有多年软件研发、系统架构经验，历任高级开发工程师、项目经理、技术总监。

热爱技术交流和知识分享，在多个平台输出技术博客、图文课程和视频课程，公众号原创博主、知识星球嘉宾、慕课网讲师、GitChat认证作者，对Java编程有着丰富的经验和独到的见解。

欢迎关注微信公众号“Java大联盟”，阅读技术文章，与作者和同道中人共话技术交流。



Java大联盟

也欢迎大家加入读者QQ群（群号：688588534），作者会在群内解疑答惑，还会适时在群里发布重磅超值的学习资料。



读者QQ群

本书由浅入深地对Java核心基础进行了解析，实用性强，内容通俗易懂，示例代码丰富，能够帮助零基础的读者快速掌握Java的核心概念，轻松上手Java编程。如果你想从零开始学习Java编程，这本书就是非常明智的选择。

——张福民

美国亚马逊公司工程部总监
清华大学计算机科学学士&硕士
美国克莱姆森大学计算机工程博士

Java语言在服务器开发、无线客户端开发以及分布式架构开发等方面有着诸多应用，本书由浅入深地介绍了Java开发中各个环节的关键知识，覆盖了Java开发的基础内容，是Java入门的佳品。

——李博

《机器学习实践应用》作者
公众号“凡人机器学习”博主

本书内容深入浅出、知识点全面，结合了作者丰富的开发经验，具有较强的实战指导意义，可以帮助零基础的读者快速入门Java编程，值得一读。

——李艳鹏

《可伸缩服务架构：框架与中间件》作者
《分布式服务架构：原理、设计与实战》作者

作为一名工作多年的老程序员，有一个比较深刻的认识：一个程序员的基础有多牢固，决定了他能够走多远。虽然现在热门类库、框架有很多，但万变不离其宗，底层的Java基础知识是非常重要的。宁楠的这本书从零开始讲述Java编程技术，涉及Java核心概念的方方面面，内容通俗易懂，是编程新手和初级程序员巩固基础的优选读物之一。

——纯洁的微笑

知名技术博主、自媒体人

Java 编程学习之道

我大学读的是软件工程专业，在校期间学校开设了很多高级编程语言课程，如 C、C++、C#、Java 等。经过一段时间的学习实践和对比，我决定专攻 Java，原因有三点：一、Java 易上手且功能强大，经过 20 多年的发展，Java 生态体系已经非常成熟；二、应用面广，无论是 Web 端、移动端，还是企业级项目，都可以使用 Java 进行开发；三、Java 开发工程师的市场需求量很大，并且待遇也不错。基于以上三点原因，我非常坚定地选择了 Java 作为自己主要的编程语言。当然这并不是说其他语言不好，每种语言都有自己的优势和劣势，没有绝对的好与坏，要结合自己的实际情况选择一门编程语言学习。在我看来，对于零基础的初学者，选择相对容易上手且成熟的编程语言来入门，效率会更高，做到“先有再精”，后期可根据工作情况和个人选择来学习其他的高级编程语言，有了基础以后，学起来就会很快。编程思想都是相通的，拿我本人来说，除了继续深入学习 Java，我也在学习其他编程技术，毕竟选择从事编程行业的工作就注定了要不断地学习。

我从第一次接触 Java 到现在也有将近 10 年的时间了，从初级开发工程师做起，一路进阶到高级开发、项目经理、技术总监。在职业生涯中的每个阶段，我对 Java 编程都会有新的认识，甚至很多时候对于过去的理解有颠覆性的改变。这是因为学习技术需要有一个沉淀和积累的过程，由量变完成质变。毕竟，写过一万行代码和写过 10 万行代码的人对技术的认知是不一样的，不同阶段对编程的理解也是不一样的，我们在工作和学习中需要不断总结思考，通过实际项目研发去夯实基础，建立自己的知识体系。

对于初入职场的新人来说，什么是最重要的？不是理论知识是否扎实，更不是你懂多少新技术，而是解决问题的能力，公司花钱雇你来上班，那你最基本的职责就是要在规定的时间内完成规定的任务。项目研发的不可预期性很高，你不知道什么时候会遇到哪些技术难点，解决这些难点需要花费多少时间和精力，这就要求你具备快速解决问题的能力。能在短时间内把问题搞定，能保证项目按时交付，你才是一名合格的程序员。在工作中，我们常常会遇到痴迷于技术的程序员，这本身没有问题，每个程序员都应该痴迷于技术，但是需要明确的是技术应为项目服务，也就是说能按时交付合格的项目是最重要的。应用的技术是否高端并不是最重要的，如果不能按时完成项目，连最基本的要求都无法达到，何谈技术优化升级？所以我们在实际工作中不要过于追求技术，一切要以实践为主。我曾经遇到过这样的同事，技术很厉害，喜欢在项目中炫技，把自认为好的技术强行加入项目中。他不考虑是否合适，是否会影响项目进度，想方设法也要把自认为好的技术加进去，

最终的结果往往是辛苦加班很多天也没有实现需要的功能，而别人使用相对传统的技术很快就能完成任务。这就是本末倒置，得不偿失。技术固然很重要，但是应该以项目为重，为项目选择最合适的技术，而不是你认为最新最好的技术。公司对项目的要求首先是能按时交付，在这个基础之上，再去考虑技术的迭代和优化。

看到这里，有的人可能会说，那编程工作就很简单了，只需要掌握一些基本的技术，能完成工作就可以了，不需要学习什么新技术呀。我上面说的是对一个开发者最基本的要求，或者说是最标准，如果你希望自己的事业节节高升，就需要不断学习新技术，不断更新优化自己的知识体系，并且记住一点，那就是只学有用的。什么是有用的呢？一方面是你当前工作所需要具备的技能，可以更好地帮你完成工作。另一方面是前瞻性的技术，比如未来两三年你可能需要掌握的，或者是大趋势所指向的技术。

当你从初级开发者成长为中高级开发者时，就不能仅限于实现基本的业务功能了，这样的工作没有多少含金量，干一年和干3年不会有太大的区别。打个比方，当搬砖对你来说已经是驾轻就熟，你就需要去思考怎么设计房子了，也就是从 CRUD 基本操作到软件设计架构的进阶。怎样提高自己的软件架构能力呢？首先你要具备扎实的基础知识，其次要有足够的项目经验，还要视野开阔，在技术领域的涉猎面要广。提高编程能力最直接也有效的方法就是看源码，学习源码是有一定门槛的，刚开始看的时候可能会遇到很多问题，或者根本就看不懂。这个时候也不能放弃，要逼着自己看源码，第一遍第二遍完全不懂没关系，坚持看到第三遍、第四遍的时候会发现自己好像懂了那么一点，继续坚持下去，你就会理解源码的思路了。

学习的阶段性很重要，找准自己当前所处的位置，学自己最应该学习的东西，慢慢提升自己的水平，不要跟风，不要别人学什么你就学什么。不同的阶段需要有针对性地部署学习计划来不断地提高和完善自己。假如你是刚入行的初级开发者，你首先应该考虑如何提高自己解决问题的能力，学习业务知识，更加高效地完成工作，而不是去学习软件架构设计。连地基都没有，何来万丈高楼？学习编程是一个从无到有、从有到精、从精到广的过程，想要做好 Java 软件开发的工作，需要不断地总结、思考、学习。

欢迎大家加入 QQ 群“Java 零基础实战”（群号：688588534）来参与互动学习，我会在群里答疑解惑，还会适时分享实用且超值的学习资料。

宁楠

2019年1月于西安

致 谢

感谢一起工作过的同事们，感谢你们在工作中给予我的帮助。感谢人民邮电出版社胡俊英编辑，她非常专业的指导和建议使得本书的内容更加完善。感谢我的家人，因为有了你们的理解和支持，我才能顺利完成本书的写作。

作者简介

宁楠，资深 Java 开发工程师，历任高级开发工程师、项目经理、技术总监。同时也是公众号原创博主、知识星球嘉宾、慕课网讲师、GitChat 认证作者，对 Java 编程有着丰富的经验和独到的见解，热衷于分享技术干货。欢迎大家搜索微信公众号“Java 大联盟”关注我的技术文章，也期待与大家有更多的技术交流。

前言

写作本书的目的

我个人在学习 Java 的过程中有这样一种感受，很多书本上的知识偏重于理论，没有太多的实践案例，不利于快速上手。网上的一些博客教程偏重于实践，没有太多晦涩难懂的理论知识，能快速上手进行开发，往往学习效率更高。但是博客也有不好的地方，一方面是不够系统，往往只是一些单一的技术点，并没有形成完整的知识体系。另一方面是不够严谨，错误也比较多，找到一篇优质且技能点完整的博客不是一件容易的事情。那么一本以实践为主，理论为辅，能快速上手的 Java 系统性入门图书，对于零基础的初学者来讲就显得尤为重要，甚至可以帮助他们达到事半功倍的效果。我写作本书的目的就是为所有对 Java 感兴趣的零基础读者提供一本可以帮助他们快速入门，以实际开发为导向的书。

一直以来我都有做笔记和写总结的习惯，并且坚持以实践为主，用实践去验证理论。也正是因此，我的笔记和总结实用性很强，我通过各种网络渠道分享过一些自认为比较好的笔记，反响还不错，尤其是很多初学者很喜欢看我写的教程，觉得通俗易懂，很适合他们。我在工作之余录制过一些视频课程，也做过在线直播授课，久而久之发现初学者的很多问题和困惑都是相同的，就集中在那么几个关键点上，所以我在写教程的时候就会侧重于这些技术点的讲解，并且尽可能用通俗易懂的语言去讲解，抓住了关键点，自然会得到受众群体的青睐。

我将多年来的心得体会和技术笔记进行汇总，历时数月，历经反复推敲修改，最终整理成本书。目标群体定位为零基础的初学者，让他们可以通过本书实现 Java 编程的快速入门，并且能将书中所讲的知识学以致用，写出功能完善的 Java 项目。

读者对象

本书适合以下几类读者阅读：

- 计算机相关专业的学生；
- 有兴趣从事 Java 开发工作的人群；
- 初级 Java 开发工程师。

如何阅读本书

本书的结构是按照 Java 知识体系的进阶来编写的，全书分为 4 个部分共 11 章。

第 1 部分是 Java 基础，也就是第 1~3 章，介绍了 Java 的历史、功能、运行机制和体系结构，以及 Java 环境的安装。重点讲解了 Java 的基本语法、变量、内存模型、运算符、流程控制、循环和数组等内容。

第 2 部分是 Java 面向对象，包括第 4~5 章，主要介绍了面向对象编程思想，以及在 Java 中的应用，包括类与对象、封装、继承、抽象、多态、包装类和异常。

第 3 部分是 Java 高级应用，包括第 6~9 章，主要讲解了 Java 的集合框架、多线程、实用类和 IO 流，学完这部分内容，就可以使用 Java 开发出功能完善的程序了。

第 4 部分是底层扩展，包括第 10~11 章，主要讲解了 Java 的反射机制和网络编程，这两块内容对于初学者可能有些难度，但是都非常重要。要搞清楚 Java 企业级开发组件和框架的底层原理，这两部分知识必不可少。

大家可以根据自己的需求选择阅读的侧重点，不过初学者最好能够按照顺序来阅读，这样可以有一个循序渐进的过程，对整个技术结构有一个清晰的梳理，有助于读者建立自己的知识体系。

本书示例代码说明

本书共有 300 多段示例代码，为了让阅读更加精简，书中只摘取了代码的核心部分，其余部分做了省略，省略的部分包括包的引入、导入其他包的类、成员变量的 getter、setter 方法以及 try-catch 等常规操作，完整的代码片段如下所示。

```
package com.southwind.io;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.OutputStream;
public class IOTest {
    private String path;
    public String getPath() {
        return path;
    }
    public void setPath(String path) {
        this.path = path;
    }
    public static void main(String[] args) {
        OutputStream outputStream = null;
        try {
            outputStream = new FileOutputStream("...");
            outputStream.write(99);
        }
    }
}
```

```
    } catch (IOException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    } finally {
        try {
            outputStream.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
}
```

精简之后的代码片段如下所示。

```
public class IOTest {
    private String path;
    //getter、setter 方法

    public static void main(String[] args) {
        OutputStream outputStream = null;
        try {
            outputStream = new FileOutputStream("...");
            outputStream.write(99);
        } catch (IOException e) {
        } finally {
            try {
                outputStream.close();
            } catch (IOException e) {
            }
        }
    }
}
```

读者在复现书中示例时需要注意补全代码，完整的源码可到异步社区的官网下载。

关于勘误

作者水平有限，加之时间仓促。虽然我花费了大量的精力反复审读，但书中仍难免会有一些错误和纰漏。如果读者发现了任何问题，恳请反馈给我，可以通过邮箱 ningnan9801@163.com 与我取得联系，也可以提交到异步社区。

资源与支持

本书由异步社区出品，社区 (<https://www.epubit.com/>) 为您提供相关资源和后续服务。

配套资源

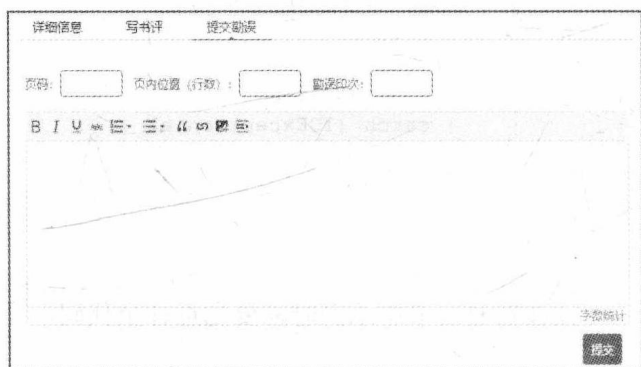
本书提供配套代码资源，要获得该配套资源，请在异步社区本书页面中单击 **配套资源**，跳转到下载界面，按提示进行操作即可。注意：为保证购书读者的权益，该操作会给出相关提示，要求输入提取码进行验证。

如果您是教师，希望获得教学配套资源，请在社区本书页面中直接联系本书的责任编辑。

提交勘误

作者和编辑尽最大努力来确保书中内容的准确性，但难免会存在疏漏。欢迎您将发现的问题反馈给我们，帮助我们提升图书的质量。

当您发现错误时，请登录异步社区，按书名搜索，进入本书页面，单击“提交勘误”，输入勘误信息，单击“提交”按钮即可。本书的作者和编辑会对您提交的勘误进行审核，确认并接受后，您将获赠异步社区的100积分。积分可用于在异步社区兑换优惠券、样书或奖品。



The screenshot shows a web form titled "提交勘误" (Submit勘误) with three tabs: "详细信息" (Detailed Information), "写书评" (Write a Review), and "提交勘误" (Submit勘误). The form includes input fields for "页码:" (Page Number), "页内位置 (行数):" (Page Position (Line Number)), and "勘误内容:" (勘误内容:). Below these fields is a rich text editor with a toolbar containing icons for bold (B), italic (I), underline (U), strikethrough (ABC), bulleted list, numbered list, link, and unlink. At the bottom right of the form, there is a "字数统计" (Character Count) label and a "提交" (Submit) button.

扫码关注本书

扫描下方二维码，您将会在异步社区微信服务号中看到本书信息及相关的服务提示。



与我们联系

我们的联系邮箱是 contact@epubit.com.cn。

如果您对本书有任何疑问或建议，请您发邮件给我们，并请在邮件标题中注明本书书名，以便我们更高效地做出反馈。

如果您有兴趣出版图书、录制教学视频，或者参与图书翻译、技术审校等工作，可以发邮件给我们；有意出版图书的作者也可以到异步社区在线提交投稿（直接访问 www.epubit.com/selfpublish/submission 即可）。

如果您是学校、培训机构或企业，想批量购买本书或异步社区出版的其他图书，也可以发邮件给我们。

如果您在网上发现有针对异步社区出品图书的各种形式的盗版行为，包括对图书全部或部分内容的非授权传播，请您将怀疑有侵权行为的链接发邮件给我们。您的这一举动是对作者权益的保护，也是我们持续为您提供有价值的内容的动力之源。

关于异步社区和异步图书

“异步社区”是人民邮电出版社旗下 IT 专业图书社区，致力于出版精品 IT 技术图书和相关学习产品，为作译者提供优质出版服务。异步社区创办于 2015 年 8 月，提供大量精品 IT 技术图书和电子书，以及高品质技术文章和视频课程。更多详情请访问异步社区官网 <https://www.epubit.com>。

“异步图书”是由异步社区编辑团队策划出版的精品 IT 专业图书的品牌，依托于人民邮电出版社近 30 年的计算机图书出版积累和专业编辑团队，相关图书在封面上印有异步图书的 LOGO。异步图书的出版领域包括软件开发、大数据、AI、测试、前端、网络技术 etc。



异步社区



微信服务号

目 录

第 1 部分 Java 基础

第 1 章 Java 初体验 2

- 1.1 Java 概述 2
 - 1.1.1 什么是 Java 2
 - 1.1.2 Java 的运行机制 4
 - 1.1.3 Java 三大体系 5
- 1.2 搭建 Java 开发环境 6
 - 1.2.1 安装配置 Java10 6
 - 1.2.2 Java 程序开发步骤 11
- 1.3 小结 13

第 2 章 Java 入门 14

- 2.1 开发第一个 Java 程序 14
 - 2.1.1 使用 Eclipse 开发程序 14
 - 2.1.2 编码规范 21
 - 2.1.3 注释 24
 - 2.1.4 关键字 27
- 2.2 变量 29
 - 2.2.1 什么是变量 29
 - 2.2.2 如何使用变量 30
- 2.3 基本数据类型 32
- 2.4 数据类型转换 33
 - 2.4.1 自动转换 33
 - 2.4.2 强制转换 35
- 2.5 运算符 36
 - 2.5.1 赋值运算符 36
 - 2.5.2 基本算术运算符 37
 - 2.5.3 复合算术运算符 40

- 2.5.4 关系运算符 41
- 2.5.5 逻辑运算符 43
- 2.5.6 条件运算符 46
- 2.5.7 位运算符 47
- 2.6 小结 50

第 3 章 Java 进阶 51

- 3.1 流程控制 51
 - 3.1.1 if-else 51
 - 3.1.2 多重 if 53
 - 3.1.3 if 嵌套 55
 - 3.1.4 switch-case 56
- 3.2 循环 58
 - 3.2.1 while 循环 58
 - 3.2.2 do-while 循环 61
 - 3.2.3 for 循环 63
 - 3.2.4 while、do-while 和 for 这 3 种循环的区别 64
 - 3.2.5 双重循环 65
 - 3.2.6 终止循环 69
- 3.3 数组 70
 - 3.3.1 什么是数组 70
 - 3.3.2 数组的基本要素 71
 - 3.3.3 如何使用数组 71
 - 3.3.4 数组的常用操作及方法 74
 - 3.3.5 二维数组 78
- 3.4 综合练习 80
- 3.5 小结 86

第 2 部分 Java 面向对象

第 4 章 面向对象基础 88

- 4.1 什么是面向对象 88
- 4.2 类与对象 89
 - 4.2.1 类与对象的关系 89
 - 4.2.2 定义类 90
 - 4.2.3 构造函数 91
 - 4.2.4 创建对象 92
 - 4.2.5 使用对象 93
 - 4.2.6 this 关键字 93
 - 4.2.7 方法重载 94
 - 4.2.8 成员变量和局部变量 95
- 4.3 封装 98
 - 4.3.1 什么是封装 98
 - 4.3.2 封装的步骤 99
 - 4.3.3 static 关键字 101
- 4.4 继承 105
 - 4.4.1 什么是继承 105
 - 4.4.2 子类访问父类 106
 - 4.4.3 子类访问权限 109
 - 4.4.4 方法重写 111
 - 4.4.5 方法重写 VS 方法重载 114
- 4.5 多态 114
 - 4.5.1 什么是多态 114
 - 4.5.2 多态的使用 117
 - 4.5.3 抽象方法和抽象类 119
- 4.6 小结 122

第 5 章 面向对象进阶 123

- 5.1 Object 类 123
 - 5.1.1 认识 Object 类 123
 - 5.1.2 重写 Object 类的方法 125
- 5.2 包装类 131
 - 5.2.1 什么是包装类 131
 - 5.2.2 装箱与拆箱 132
- 5.3 接口 135
 - 5.3.1 什么是接口 135

- 5.3.2 如何使用接口 136
- 5.3.3 面向接口编程的实际应用 138

- 5.4 异常 142
 - 5.4.1 什么是异常 142
 - 5.4.2 异常的使用 142
 - 5.4.3 异常类 146
 - 5.4.4 throw 和 throws 147
 - 5.4.5 自定义异常类 151
- 5.5 综合练习 152
- 5.6 小结 157

第 3 部分 Java 高级应用

第 6 章 多线程 160

- 6.1 进程与线程 160
- 6.2 Java 中线程的使用 163
 - 6.2.1 继承 Thread 类 163
 - 6.2.2 实现 Runnable 接口 165
 - 6.2.3 线程的状态 166
- 6.3 线程调度 167
 - 6.3.1 线程休眠 167
 - 6.3.2 线程合并 169
 - 6.3.3 线程礼让 171
 - 6.3.4 线程中断 173
- 6.4 线程同步 175
 - 6.4.1 线程同步的实现 175
 - 6.4.2 线程安全的单例模式 183
 - 6.4.3 死锁 187
 - 6.4.4 重入锁 189
 - 6.4.5 生产者消费者模式 194
- 6.5 综合练习 196
- 6.6 小结 198

第 7 章 集合框架 199

- 7.1 集合的概念 199
- 7.2 Collection 接口 200
 - 7.2.1 Collection 接口的定义 200

7.2.2	Collection 的子接口	201
7.3	List 接口	201
7.3.1	List 接口的定义	201
7.3.2	List 接口的实现类	202
7.4	Set 接口	207
7.4.1	Set 接口的定义	207
7.4.2	Set 接口的实现类	208
7.5	Map 接口	214
7.5.1	Map 接口的定义	214
7.5.2	Map 接口的实现类	215
7.6	Collections 工具类	220
7.7	泛型	222
7.7.1	泛型的概念	222
7.7.2	泛型的应用	224
7.7.3	泛型通配符	226
7.7.4	泛型上限和下限	227
7.7.5	泛型接口	228
7.8	综合练习	229
7.9	小结	232

第 8 章 实用类 233

8.1	枚举	233
8.2	Math	236
8.3	Random	237
8.4	String	238
8.4.1	String 实例化	238
8.4.2	String 常用方法	242
8.5	StringBuffer	244
8.6	日期类	246
8.6.1	Date	246
8.6.2	Calendar	248
8.7	小结	249

第 9 章 IO 流 250

9.1	File 类	250
9.2	字节流	251
9.3	字符流	257
9.4	处理流	265

9.5	缓冲流	267
9.5.1	输入缓冲流	268
9.5.2	输出缓冲流	274
9.6	序列化和反序列化	279
9.6.1	序列化	279
9.6.2	反序列化	280
9.7	小结	281

第 4 部分 底层扩展

第 10 章 反射 284

10.1	Class 类	284
10.2	获取类结构	287
10.2.1	获取类的接口	288
10.2.2	获取父类	289
10.2.3	获取构造函数	290
10.2.4	获取方法	292
10.2.5	获取成员变量	294
10.3	反射的应用	296
10.3.1	反射调用方法	296
10.3.2	反射访问成员变量	299
10.3.3	反射调用构造函数	302
10.4	动态代理	303
10.5	小结	309

第 11 章 网络编程 310

11.1	IP 与端口	312
11.1.1	IP	312
11.1.2	端口	314
11.2	URL 和 URLConnection	314
11.2.1	URL	314
11.2.2	URLConnection	317
11.3	TCP 协议	318
11.4	UDP 协议	322
11.5	多线程下的网络编程	324
11.6	综合练习	326
11.7	小结	328