

高等院校信息技术规划教材

Java面向对象 程序设计

孙连英 刘畅 彭涛 编著



清华大学出版社

高等院校信息技术规划教材

Java面向对象 程序设计

孙连英 刘畅 彭涛 编著

清华大学出版社
北京

内 容 简 介

本书基于 Java 语言介绍面向对象程序设计理念,引入 UML 工具解释设计过程;用丰富的案例阐述面向对象程序设计的基本理论和方法,详细介绍面向对象的基本特性、基本技术,知识点与应用实例相结合。全书分为三篇:第 1 篇,编程基础,介绍 Java 的编程基础;第 2 篇,Java 面向对象,介绍面向对象的封装性、继承性和多态性;第 3 篇,高级面向对象设计,介绍 UI、多线程、网络编程等内容。本书内容从简单到复杂,阶梯式递进,读者可以根据需要选读。

本书介绍面向对象思想,注重理论联系实际,既可作为高等院校软件工程、计算机相关专业的本科生教材,也可作为研究生的参考资料。

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 面向对象程序设计/孙连英,刘畅,彭涛编著. —北京:清华大学出版社,2017
(高等院校信息技术规划教材)
ISBN 978-7-302-48907-8

I. ①J… II. ①孙… ②刘… ③彭… III. ①JAVA 语言—程序设计—高等学校—教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字(2017)第 286295 号

责任编辑:张 玥 赵晓宁
封面设计:常雪影
责任校对:时翠兰
责任印制:杨 艳

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 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

课件下载: <http://www.tup.com.cn>, 010-62795954

印 装 者:北京富博印刷有限公司

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:185mm×260mm

印 张:18

字 数:412千字

版 次:2017年12月第1版

印 次:2017年12月第1次印刷

印 数:1~1500

定 价:45.00元

产品编号:067883-01

前言

foreword

背景

随着“互联网+”国家战略的实施和大数据、云计算、物联网等技术的不断发展,移动互联网逐渐成为人们日常交流、通信和娱乐的基本工具。人们通过移动终端获取信息已经成为一种普遍的现象,教育、汽车、医疗、金融、旅游、生活服务等细分领域移动 APP 将迎来飞跃式发展。目前主流的移动应用操作系统主要有 Android、IOS、Windows Phone 等。Android 系统是一个开发平台,与其他智能操作系统不同的是,Android 是一款基于 Linux 平台的开源操作系统,从而避开了阻碍市场发展的专利壁垒。由于 Android 系统是一款完全免费的智能手机平台,Android 移动终端应用市场占有率逐年增长,由此 Java 移动应用软件开发人员有较大的社会需求。

Java 语言的基本编程思想是面向对象,面向对象的程序设计已经成为软件编程技术中一项非常关键的技术。Java 语言吸收了其他语言的各种优点,设计简洁而优美,使用方便而高效。特别是跨平台性,使其在大型应用系统和嵌入式开发中都占有重要地位。本教材基于 Java 语言介绍面向对象的思想设计,运用 UML 建模,具有丰富的实例,体现面向对象程序设计的应用性。

Java 世界的巅峰永远都在技术人员的追求中不断升高,每个人都有不同的路,有效的参考教材是每一位程序员必不可少的工具。本书的内容是按照企业技能培训的模式来安排的,主要从应用开发的逻辑路径组织内容,注重完成基本功能的思路和步骤分析说明,没有从试图穷尽所有的知识面角度来撰写,如果读者想就某一个具体的技术点(例如某个控件的特殊用法)进行突破则需要参阅其他资料来完成。

本书特色

任何技术的目的都是为了应用。本书不仅结合实例详细讲解 Java 的基础知识,同时还就 Java 的主要应用进行实例讲解。全书共

分 14 章,从基本概念的引入,到典型案例的分析,使读者更形象地理解面向对象思想,掌握 Java 编程技术。

本书特点如下:

- (1) 由案例引入,从具体问题的分析入手,由浅入深。
- (2) 注重具体问题的分析、设计。案例中给出解决思路,有助于提高读者分析问题和解决问题的能力。
- (3) 突出软件开发的设计与实现过程,将面向对象分析与 Java 语言开发相结合,使学生掌握软件开发的基本技能。
- (4) 每章后配有实验,注重程序开发能力的锻炼。

读者对象

本书可作为高等院校软件工程、计算机科学与技术等相关专业本科教材,也可作为相关学科的研究生参考资料,同时还可作为学习 Java 开发、移动应用开发、Java EE 开发的职业技能培训教材。

本书作者

本书受到北京联合大学规划教材建设项目资助,由北京联合大学教师团队与北京尚嘉悦成科技有限公司企业团队合作完成。参加本书编写工作的有北京联合大学的孙连英、刘畅、彭涛和北京尚嘉悦成科技有限公司的毛英勇、刘磊。其中,第 1~第 4 和第 9 章由孙连英编写,第 5~第 8 和第 10 章由刘畅编写,第 11~第 14 章由彭涛编写。本书中的案例由毛英勇、刘磊提供。全书由孙连英、刘畅统稿。在编写过程中得到刘小安、李刚、任运贵、张启秀、李琳等的帮助,在此表示感谢。

对于本书实例开发中的程序源代码,读者可以在清华大学出版社网站上免费下载。

由于作者水平有限,书中遗漏之处在所难免,敬请读者批评指正。

编者

2017 年 6 月

目录

Contents

第 1 篇 编程基础

| | |
|--------------------------------|----|
| 第 1 章 概述 | 3 |
| 1.1 面向对象编程思想 | 3 |
| 1.1.1 面向对象编程语言 | 3 |
| 1.1.2 类和对象 | 4 |
| 1.1.3 面向对象的特征 | 4 |
| 1.1.4 Java 语言的特点 | 6 |
| 1.2 编写 Java 程序的步骤 | 8 |
| 1.2.1 准备 Java 开发环境 | 8 |
| 1.2.2 准备编程工具 | 12 |
| 1.2.3 编写第一个 Java 程序 | 12 |
| 1.3 标准 I/O | 16 |
| 1.3.1 标准输入流 | 16 |
| 1.3.2 通过 Scanner 类实现输入输出 | 18 |
| 习题 1 | 19 |
| 编程练习 | 19 |
| 第 2 章 基础知识 | 20 |
| 2.1 标识符 | 20 |
| 2.2 数据类型 | 21 |
| 2.2.1 基本数据类型 | 22 |
| 2.2.2 类型转换 | 23 |
| 2.3 变量和赋值 | 24 |
| 2.4 常量 | 25 |
| 2.5 运算符和表达式 | 25 |
| 2.5.1 基本赋值运算 | 26 |

| | | |
|--------------|-------------------------|-----------|
| 2.5.2 | 算术运算 | 26 |
| 2.5.3 | 关系运算 | 28 |
| 2.5.4 | 布尔运算 | 29 |
| 2.5.5 | 位运算 | 30 |
| 2.5.6 | 条件运算 | 30 |
| 2.5.7 | 复合赋值运算符 | 30 |
| 2.6 | 字符串 | 31 |
| 2.7 | if 语句 | 32 |
| 2.7.1 | if 语句的三种形式 | 32 |
| 2.7.2 | 使用 if 分段显示 | 34 |
| 2.8 | switch 语句 | 35 |
| 2.8.1 | 使用 switch 分段显示 | 35 |
| 2.8.2 | switch 语句详解 | 35 |
| 2.9 | for 循环 | 36 |
| 2.9.1 | for 循环语法 | 36 |
| 2.9.2 | 求和运算实例 | 37 |
| 2.10 | while 循环 | 38 |
| 2.10.1 | while 循环语句 | 38 |
| 2.10.2 | while 循环实现输入控制 | 38 |
| 2.11 | do...while 循环 | 39 |
| 2.11.1 | do...while 语句 | 39 |
| 2.11.2 | do...while 实现退出操作 | 39 |
| 2.12 | break 与 continue | 39 |
| 2.12.1 | break 语句 | 39 |
| 2.12.2 | continue 语句 | 40 |
| 2.13 | 综合实例：十进制与二进制转换 | 41 |
| 2.13.1 | 问题分析 | 41 |
| 2.13.2 | 算法设计 | 42 |
| 2.13.3 | 主程序 | 42 |
| | 习题 2 | 43 |
| | 编程练习 | 44 |
| 第 3 章 | 方法 | 46 |
| 3.1 | 定义方法 | 46 |
| 3.2 | 调用方法 | 46 |
| 3.3 | 方法实例 | 48 |
| 3.4 | 方法应用 | 50 |
| 3.4.1 | 单个参数的方法 | 50 |

| | |
|-------------------------|-----------|
| 3.4.2 多个参数的方法 | 51 |
| 3.4.3 递归方法 | 53 |
| 3.4.4 多个返回值的方法 | 54 |
| 3.5 方法重载 | 54 |
| 习题 3 | 55 |
| 编程练习 | 56 |
| 第 4 章 数组 | 58 |
| 4.1 创建数组 | 58 |
| 4.2 初始化一维数组 | 59 |
| 4.2.1 静态初始化 | 60 |
| 4.2.2 动态初始化 | 60 |
| 4.2.3 创建数组实例 | 61 |
| 4.3 数组名的使用 | 62 |
| 4.4 数组作为方法的参数和返回值 | 63 |
| 4.4.1 数组作为方法参数 | 63 |
| 4.4.2 数组作为返回类型 | 64 |
| 4.5 增强的 for 循环 | 65 |
| 4.6 多维数组 | 66 |
| 4.6.1 二维数组定义 | 67 |
| 4.6.2 初始化二维数组 | 67 |
| 4.6.3 二维数组实例 | 68 |
| 习题 4 | 69 |
| 编程练习 | 70 |

第 2 篇 Java 面向对象

| | |
|-------------------------|-----------|
| 第 5 章 类和对象 | 75 |
| 5.1 类与对象的关系 | 75 |
| 5.2 类的设计与 UML 建模 | 75 |
| 5.3 类 | 76 |
| 5.3.1 汽车类实例 | 76 |
| 5.3.2 定义类 | 77 |
| 5.4 对象 | 78 |
| 5.4.1 创建对象 | 78 |
| 5.4.2 使用对象 | 79 |
| 5.5 成员变量与局部变量 | 80 |

| | | |
|--------------|-------------------------|------------|
| 5.6 | 构造方法 | 81 |
| 5.6.1 | 为什么需要构造方法 | 81 |
| 5.6.2 | 构造方法重载 | 82 |
| 5.7 | this 关键字 | 84 |
| 5.7.1 | this 表示类的成员 | 84 |
| 5.7.2 | this 调用构造方法 | 85 |
| 5.7.3 | this 表示当前对象 | 86 |
| 5.8 | static 修饰符 | 87 |
| 5.8.1 | 静态变量 | 87 |
| 5.8.2 | 静态方法 | 89 |
| 5.8.3 | main 方法详解 | 91 |
| 5.8.4 | 静态成员特点 | 91 |
| 5.9 | String 类 | 92 |
| 5.10 | 对象参数实例 | 95 |
| 5.11 | 聚集类 | 97 |
| 5.11.1 | 聚集类实例 | 97 |
| 5.11.2 | 银行操作主界面 | 101 |
| 5.12 | 内部类 | 105 |
| 5.12.1 | 类中定义的内部类 | 105 |
| 5.12.2 | 内部类被外部引用 | 106 |
| 5.12.3 | 方法中定义的内部类 | 107 |
| | 习题 5 | 108 |
| | 编程练习 | 109 |
| 第 6 章 | 封装 | 112 |
| 6.1 | 类的抽象与封装 | 112 |
| 6.2 | Java 中的包 | 113 |
| 6.2.1 | package 语句 | 113 |
| 6.2.2 | import 语句 | 115 |
| 6.2.3 | Calendar 与 DateFormat 类 | 116 |
| 6.3 | 类的成员的访问控制 | 118 |
| 6.3.1 | 包访问权限 | 118 |
| 6.3.2 | public: 接口访问权限 | 118 |
| 6.3.3 | private: 类内部访问权限 | 119 |
| 6.3.4 | protected: 继承访问权限 | 119 |
| 6.4 | 类的访问权限 | 119 |
| 6.5 | 链表实例 | 120 |
| 6.5.1 | 链表结点 | 120 |

| | | |
|--------------|-------------------|-----|
| 6.5.2 | 链表类 | 121 |
| 6.5.3 | 测试类 | 125 |
| 习题 6 | | 125 |
| 编程练习 | | 126 |
| 第 7 章 | 继承 | 128 |
| 7.1 | 继承的定义与实现 | 128 |
| 7.1.1 | 继承实例 | 128 |
| 7.1.2 | 继承的定义 | 128 |
| 7.1.3 | 继承的实现 | 129 |
| 7.1.4 | 继承的结果 | 132 |
| 7.2 | 方法重写 | 132 |
| 7.3 | super 关键字 | 135 |
| 7.3.1 | 使用 super 访问父类成员 | 135 |
| 7.3.2 | 使用 super 调用父类构造方法 | 137 |
| 7.3.3 | 构造子类对象 | 137 |
| 7.3.4 | super 与 this | 138 |
| 7.4 | 抽象类 | 139 |
| 7.4.1 | Employee 抽象类 | 139 |
| 7.4.2 | 抽象类实例 | 140 |
| 7.4.3 | 抽象类的子类实例 | 140 |
| 7.5 | 接口 | 142 |
| 7.5.1 | 定义接口 | 143 |
| 7.5.2 | 接口实例 | 143 |
| 7.6 | final 修饰符 | 144 |
| 7.6.1 | final 修饰类 | 145 |
| 7.6.2 | final 修饰方法 | 145 |
| 7.7 | Object 类 | 145 |
| 7.7.1 | toString() 方法 | 146 |
| 7.7.2 | equals() 方法 | 147 |
| 7.8 | 类的关系 | 147 |
| 7.8.1 | 依赖 | 147 |
| 7.8.2 | 关联 | 148 |
| 7.8.3 | 聚合 | 148 |
| 7.8.4 | 组合 | 149 |
| 7.9 | 综合实例: 组装计算机 | 150 |
| 习题 7 | | 152 |
| 编程练习 | | 153 |

| | |
|----------------------------|-----|
| 第 8 章 多态 | 155 |
| 8.1 多态实例 | 155 |
| 8.1.1 句柄引用对象实例 | 155 |
| 8.1.2 父类句柄引用子类对象 | 156 |
| 8.2 动态绑定 | 157 |
| 8.3 对象的多态性 | 159 |
| 8.3.1 向上转型 | 161 |
| 8.3.2 向下转型 | 161 |
| 8.3.3 instanceof 运算符 | 161 |
| 8.4 多态应用实例 | 162 |
| 8.4.1 接口作为参数实例 | 162 |
| 8.4.2 父类作为方法返回类型实例 | 164 |
| 8.4.3 面向接口编程 | 168 |
| 习题 8 | 168 |
| 编程练习 | 169 |

第 3 篇 高级面向对象设计

| | |
|-------------------------------|-----|
| 第 9 章 图形用户界面设计 | 173 |
| 9.1 Swing 和 AWT | 173 |
| 9.1.1 AWT 组件 | 173 |
| 9.1.2 Swing 组件 | 173 |
| 9.1.3 容器类组件 | 174 |
| 9.2 创建一个基本 GUI 程序 | 175 |
| 9.2.1 使用 JFrame 类创建一个框架 | 175 |
| 9.2.2 在框架中添加组件 | 176 |
| 9.2.3 设置界面布局 | 177 |
| 9.2.4 事件处理 | 183 |
| 9.3 常用的事件及其相应的监听器接口 | 185 |
| 9.3.1 Java 中事件 | 185 |
| 9.3.2 Windows 事件处理 | 186 |
| 9.3.3 键盘事件处理 | 189 |
| 9.3.4 鼠标事件处理 | 192 |
| 9.4 菜单 | 193 |
| 9.4.1 菜单的设计与实现 | 193 |
| 9.4.2 实现菜单项事件处理代码 | 195 |

| | |
|----------------------------------|------------|
| 习题 9 | 198 |
| 编程练习 | 199 |
| 第 10 章 异常处理 | 200 |
| 10.1 Java 异常处理 | 200 |
| 10.2 Exception 类 | 201 |
| 10.3 使用异常处理 | 202 |
| 10.3.1 try/catch/finally 块 | 202 |
| 10.3.2 catch 块的顺序 | 203 |
| 10.3.3 抛出异常 | 205 |
| 10.4 自定义异常 | 205 |
| 习题 10 | 207 |
| 编程练习 | 208 |
| 第 11 章 集合类 | 209 |
| 11.1 Iterable 接口 | 210 |
| 11.2 Collection 接口 | 210 |
| 11.3 List 接口 | 212 |
| 11.3.1 LinkedList 类 | 212 |
| 11.3.2 ArrayList 类 | 212 |
| 11.3.3 Vector 类 | 213 |
| 11.3.4 Stack 类 | 213 |
| 11.4 Set 接口 | 214 |
| 11.4.1 Set 接口常用方法 | 214 |
| 11.4.2 Set 接口实例 | 215 |
| 11.5 Map 接口 | 216 |
| 11.5.1 Map 常用方法 | 216 |
| 11.5.2 HashMap 管理网络名和密码 | 217 |
| 11.6 泛型 | 218 |
| 11.7 集合类实例：书籍管理 | 219 |
| 习题 11 | 223 |
| 编程练习 | 223 |
| 第 12 章 输入输出流及文件处理 | 225 |
| 12.1 I/O 流 | 225 |
| 12.2 字节流 | 226 |
| 12.2.1 InputStream 类常用方法 | 226 |

| | | |
|---------------|--|------------|
| 12.2.2 | OutputStream 类常用方法 | 227 |
| 12.2.3 | 文件数据流 | 228 |
| 12.2.4 | 实例：输入信息保存到文件 | 228 |
| 12.2.5 | 读取并显示文件 | 230 |
| 12.2.6 | 文件复制 | 231 |
| 12.3 | 字符流 | 232 |
| 12.3.1 | InputStreamReader 和 OutputStreamWriter | 233 |
| 12.3.2 | 字符流实例 | 233 |
| 12.4 | 文件输入输出 | 234 |
| 12.4.1 | 文件基本操作 | 234 |
| 12.4.2 | 文件随机读写 | 236 |
| 12.5 | 对象序列化 | 237 |
| 12.5.1 | 存储对象 | 237 |
| 12.5.2 | Car 对象序列化实例 | 238 |
| 12.5.3 | 存储和读取序列化对象信息 | 238 |
| 12.6 | 正则表达式 | 240 |
| 12.6.1 | 正则表达式的基本概念 | 240 |
| 12.6.2 | 在 Java 中使用正则表达式 | 242 |
| | 习题 12 | 247 |
| | 编程练习 | 247 |
| 第 13 章 | 多线程 | 248 |
| 13.1 | 线程概述 | 248 |
| 13.1.1 | 线程的概念 | 248 |
| 13.1.2 | 多线程 | 248 |
| 13.2 | 创建线程 | 249 |
| 13.2.1 | 继承 Thread 类 | 249 |
| 13.2.2 | 实现 Runnable 接口 | 250 |
| 13.3 | 线程的状态 | 251 |
| 13.3.1 | 线程的 5 种状态 | 251 |
| 13.3.2 | 线程的调度 | 252 |
| 13.4 | 资源共享与线程同步 | 254 |
| 13.4.1 | 资源共享 | 254 |
| 13.4.2 | 线程同步 | 256 |
| 13.4.3 | 等待与通知 | 257 |
| 13.4.4 | 死锁 | 260 |
| | 习题 13 | 261 |
| | 编程练习 | 263 |

| | |
|---------------------------|-----|
| 第 14 章 网络编程 | 264 |
| 14.1 Java 网络编程基础 | 264 |
| 14.1.1 IP 地址 | 264 |
| 14.1.2 端口号 | 265 |
| 14.2 InetAddress 类 | 265 |
| 14.3 使用 URL 类访问网络资源 | 267 |
| 14.4 Socket 套接字编程 | 269 |
| 14.4.1 使用 TCP 通信 | 269 |
| 14.4.2 使用 UDP 通信 | 272 |
| 习题 14 | 273 |
| 编程练习 | 273 |
| 参考文献 | 274 |

第 1 篇

编程基础

概 述

20世纪80年代以来,面向对象的程序设计方法越来越引人注目。与传统的程序设计方法比较,面向对象的程序设计方法最显著的特点是它更接近人们通常的思维规律。随着互联网技术的发展,尤其是移动操作系统 Android 的广泛应用,使得 Java 面向对象编程语言成为主流之一。本章将以 Java 语言为例,讨论面向对象程序设计方法的一些基本概念。

1.1 面向对象编程思想

1.1.1 面向对象编程语言

Java 是一种典型的面向对象编程语言,除此以外还有 C++、C# 等编程语言。Java 语言应用广泛,因此本书以 Java 语言为例来阐述面向对象的概念和编程思路。

Java 语言起源于 1991 年 Sun 公司的 Green 计划,Green 计划主要致力于智能型家电的研究。要为家用电子设备编程,就需要一种编程语言。由于家用电子设备的软件要在不同的芯片上运行,当时的计算机语言(包括 C、C++)都不能满足要求。因此,Green 计划负责人 James Gosling 设计了一种新的编程语言,将其命名为 Oak。后来,由于 Oak 已经是另一种程序设计语言的名称了,因此将 Oak 改名为 Java。

1996 年 Sun 公司推出 Java 开发工具包 JDK1.0 (Java Development Kit),为开发人员提供编写 Java 应用软件的工具。1998 年 JDK1.2 发布。1999 年 Java 技术被分成 J2SE、J2EE 和 J2ME 三个部分,Java Server Pages (JSP)技术也随之发布。2004 年,J2SE1.5 发布,为了表示此版本的重要性,更名为 J2SE5.0。2005 年发布了 Java SE 6。同年 Java 的各种版本更名,取消其中的数字 2,J2EE 更名为 Java EE,J2SE 更名为 Java SE,J2ME 更名为 Java ME。2010 年 1 月,Oracle 公司收购了 Sun 公司及其产品,现在 Java 由 Oracle 公司控制。2011 年 7 月,Java SE 7.0 版本发布,这个版本距离上次版本发布达 5 年之久,也是到目前为止最新的版本,主要特点是支持动态语言、多重异常处理等。

Java 程序能在各种不同的计算机平台上运行,非常适合编写网络应用程序。随着移动互联网技术和移动智能终端的发展,在 Android 操作系统的智能终端上的开发语言也是 Java。