



高等学校软件工程专业校企深度合作系列实践教材

Java 项目开发实践

JAVAXIANGMUKAIFASHIJIAN

总主编 / 周清平 主编 / 覃遵跃

副主编 / 陈园琼 张彬连 彭耶萍 王新峰



中南大学出版社
www.csupress.com.cn

高等学校软件工程专业校企深度合作系列实践教材

Java 项目开发实践

总主编	周清平	
主 编	覃遵跃	
副主编	陈园琼	张彬连
	彭耶萍	王新峰



中南大学出版社
www.csupress.com.cn

图书在版编目(CIP)数据

高等学校软件工程专业校企深度合作系列实践教材/周清平总主编.

Java 项目开发实践/覃遵跃主编. —长沙:中南大学出版社,2015. 2

ISBN 978 - 7 - 5487 - 1397 - 5

I . J . . . II . ①周 . . . ②覃 . . . III . JAVA 语言 - 程序设计

IV . TP312

中国版本图书馆 CIP 数据核字(2015)第 035338 号

Java 项目开发实践

覃遵跃 主编

☐责任编辑 韩 雪

☐责任印制 易建国

☐出版发行 中南大学出版社

社址:长沙市麓山南路

邮编:410083

发行科电话:0731-88876770

传真:0731-88710482

☐印 装 湖南地图出版社印刷厂

☐开 本 787 × 1092 1/16 ☐印张 16.5 ☐字数 405 千字

☐版 次 2015 年 4 月第 1 版 ☐2015 年 4 月第 1 次印刷

☐书 号 ISBN 978 - 7 - 5487 - 1397 - 5

☐定 价 39.00 元

图书出现印装问题,请与经销商调换

内容简介

Introduction

Java 语言已经成为互联网时代主流开发语言。本书以现实生活中的 7 个典型项目为实训案例，使学生系统掌握不同应用背景下完成一个真实 Java 应用程序开发所具备的专业知识，熟练使用 Java 应用程序开发关键技术和工具。

本书章节的内容顺序按照 Java 应用程序实际的开发流程编排，将每个项目开发拆分为“项目描述、项目目标、项目实施、项目小结和拓展”四个部分。每个项目侧重的知识点不同，避免了实例罗列和知识点的重复，第 2 章“简单计算器开发”涉及的主要知识点是界面设计，第 3 章“日历记事本开发”主要涉及各种常用类及组件的使用以及事件处理，第 4 章“简易画图板开发”主要涉及抽象类、接口、继承以及画图工具的使用等知识点，第 5 章“简易职员管理系统开发”主要涉及树形导航条以及表格控件的使用，第 6 章“基于文件的学籍管理系统开发”主要涉及 I/O 系统，第 7 章“简单聊天室开发”主要涉及网络编程和多线程编程。每个实训任务的设计都围绕提高 Java 实践能力和创新能力，学生通过自主学习即可完成 7 个 Java 实训项目的开发。

作者简介

About the Author

总主编：

周清平，男，1966年3月出生，湖南省张家界人，土家族，教授，博士后，现任中国服务贸易协会专家委员会副理事长，全国服务外包技能考试专家委员会副理事长，吉首大学软件服务外包学院院长，长期从事软件工程专业课程教学和开发，主要研究方向为量子信息、软件信息系统，主持国家自然科学基金、中国科学院科学研究基金、中国博士后基金、教育部科学研究重点项目、湖南省景区信息化专项等科研项目，主持国家级工程实践教育中心、软件工程综合改革试点专业、福特II国际合作项目、湖南省教育信息化专项等教研教改项目，获中国服务外包人才培养最佳实践新锐奖、湖南省自然科学奖、湖南省自然科学优秀学术论文奖，在 *Springer: Quant. Inform. Proces.*, *phys. Leet. A* 等国内外高级学术期刊发表SCI论文二十余篇。

本书主编：

覃遵跃，男，1974年4月出生，湖南省张家界人，土家族，副教授、博士研究生、国家高级程序员，现任吉首大学教学督导组组员。长期从事软件工程专业课程教学与研究，主讲Java语言、C语言和数据库技术等专业课程。主要研究方向为Web数据库技术、软件开发与设计。主持省级课题2项，参与国家级、部省级等项目4项，主编《利用案例轻松学习Java语言》等教材。发表学术论文40余篇，获国家级、部省级学科竞赛优秀指导教师、吉首大学教学能手等称号。

编审委员会

Editorial Committee

高等学校软件工程专业
校企深度合作系列实践教材

顾问

王志英 李仁发 陈志刚 唐振明

主任

周清平

副主任

徐洪智 颜一鸣 成 焕

编委(按姓氏笔画排序)

马庆槐	王建峰	王晓波	王新峰	宁小浩	刘 彬
闫茂源	李 刚	李长云	杨燕萍	沈 岳	张晋华
张彬连	陈生萍	陈园琼	钟 键	贾 涛	郭 鑫
唐伟奇	黄 云	黄 伟	覃遵跃	彭耶萍	曾明星
赖 炜	蔡国民				

总序

Preface

企业专业实训是在真实的企业工作环境中，以项目组的工作方式实现完整的项目开发过程，是实现高素质软件人才培养的重要实践教学环节，是集中训练学生的科学研究能力、工程实践能力和创新能力的必要一环，是对学生综合运用多学科的理论、方法、工具和技术解决实际问题的真实检验，对全面提高教育教学质量具有重要意义。

近年来，吉首大学大力践行“整体渗透、优势互补、人才共育、过程共管、资源共享、责任共担、利益共生、合作共赢”的校企深度合作办学模式，先后与中软国际、青软实训、苏软培训等知名企业开展专业共建，在沉浸式实训模式创新、课程研发、实践教学资源建设等方面取得了显著成效，本次编写出版的“高等学校软件工程专业校企深度合作系列实践教材”就是其中一项重要成果。

本系列教材包括《C 语言项目开发实践》《数据库项目开发实践》《Java 项目开发实践》《Web 前端项目开发实践》《Java EE 项目开发实践》《. Net 项目开发实践》《Android 项目开发实践》《嵌入式 ARM 体系结构编程项目开发实践》，共 8 本。校企双方教师、技术专家联合组成了教材编写委员会，他们深入生产实际、把握主流技术、遵循教学规律，摆脱了传统教材“理论知识 + 实训案例”的简单模式，将实训内容项目化、专业化和职业化，以真实的企业项目案例为载体，循序渐进地引导学生完成实训项目开发流程，使其专业知识得到巩固，专业技能得到提升，综合分析和解决实际问题的能力、项目开发能力、项目管理能力和创新精神得到强化，同时，在项目执行力、职业技能与素养诸方面得到有效锻炼。

本套教材内容覆盖了软件工程专业主要能力点，精选了一定数量的软件项目案例，从项目描述、项目目标、项目实施、项目小结与拓展等方面介绍，

均符合各自相关的项目开发规范，项目实施遵循软件生命周期模型，给出了软件设计思想、开发过程和开发结果。学生通过项目需求分析、系统设计、编码实现、系统测试与系统部署等环节，不断积累项目开发经验。本套丛书构思设计之巧、涉猎领域之广、推广应用之实，无不反映了吉首大学的教育教学改革已经转型到以学生发展为中心、以能力培养为核心的全面综合素质教育上来，是推行校企深度合作办学基础上微创新教学改革成果的集中展示。

“一分耕耘，一分收获”，吉首大学的老师们致力于耕耘，期待着收获。站在第一读者的角度，我更期待本套教材能成为高等院校软件工程专业、职业培训和软件从业人员最具实用价值的实训教材和参考书，用书中所蕴含的智慧创造更多的财富。

是为序。

 教授

联合国教科文组织产学合作教席理事会理事
教育部软件工程专业教学指导委员会副主任
国家示范性软件学院建设工作办公室副主任
北京交通大学软件学院院长、博士生导师

2014 年 6 月

前言

Foreword

由于 Java 具有“一次编写，处处运行”而不依赖具体平台、可靠性高以及安全性好等特点，并且随着 Java 语言在移动互联网以及智能电子等领域的深入应用，使它成为信息时代最重要的程序设计语言之一。目前很多高等院校都将 Java 语言作为程序设计入门语言，掌握 Java 语言已经成为人们的共识。

在学习了 Java 基本知识以后，如何综合利用 Java 各知识点提高 Java 项目开发实战能力是学习者普遍关注的问题，通过典型案例开发是提高 Java 编程能力的有效途径。

本书设计了 6 个典型案例，所有案例包含功能分析、项目设计目标、项目具体实施步骤(包括类设计、代码实现、系统发布和系统测试)、项目小结和项目拓展。每个案例互相独立，并且侧重的知识点不同，读者可以根据实际需要学习相关案例。简单计算器和日历记事本侧重 GUI 编程，画图板侧重绘图和菜单，职员管理系统侧重树形导航结构和数据库编程，学籍管理侧重输入输出和集合编程，聊天室侧重网络和多线程编程。读者可以根据实际需要参考这些案例进行 Java 项目开发。本书适合于高导院校《Java 程序设计》实践教学指导教材，也可作为 Java 程序设计人员以及自学者的参考用书。

本书由覃遵跃任主编，陈园琼、张彬连、彭耶萍、王新峰任副主编。所有案例在 JDK 1.6 运行环境下调试通过。由于作者水平有限，书中难免存在不足之处，敬请读者批评指正。

编者

2014 年 6 月

目录

C O N T E N T S

第 1 章 Java 项目开发基础	(1)
1.1 Java 项目开发实训目标	(2)
1.1.1 实训知识目标	(2)
1.1.2 实训能力目标	(2)
1.1.3 实训素质目标	(2)
1.2 Java 项目开发技术	(2)
1.2.1 图形用户界面编程技术	(2)
1.2.2 输入输出编程技术	(3)
1.2.3 网络编程技术	(4)
1.2.4 多线程编程技术	(4)
1.2.5 数据结构编程技术	(5)
1.2.6 Java 绘图编程技术	(5)
1.2.7 JDBC 数据库编程技术	(5)
1.2.8 本书各项目采用技术	(6)
1.3 Java 项目开发工具	(6)
1.3.1 JDK 开发工具包	(6)
1.3.2 SQL Server 数据库系统	(8)
1.3.3 JCreator	(10)
1.3.4 Eclipse	(10)
1.4 Java 项目开发规范	(11)
1.4.1 Java 项目开发实训工作流程	(11)
1.4.2 Java 项目开发流程	(12)
1.4.3 Java 项目开发代码规范	(12)
1.4.4 文档与源码提交规范	(15)
1.5 小结	(16)
第 2 章 简易计算器项目设计与开发	(17)
2.1 项目描述	(17)

2.2 项目目标	(17)
2.2.1 系统功能	(17)
2.2.2 功能描述	(18)
2.3 项目实施	(18)
2.3.1 类及 UML 设计	(18)
2.3.2 代码实现	(23)
2.3.3 系统发布	(41)
2.3.4 系统测试	(42)
2.4 项目小结与拓展	(44)
2.4.1 项目小结	(44)
2.4.2 项目拓展	(44)
第3章 日历记事本项目设计与开发	(45)
3.1 项目描述	(45)
3.2 项目目标	(45)
3.2.1 系统功能	(45)
3.2.2 功能描述	(45)
3.3 项目实施	(46)
3.3.1 类及 UML 设计	(46)
3.3.2 代码实现	(47)
3.3.3 系统发布	(61)
3.3.4 系统测试	(62)
3.4 项目小结与拓展	(64)
3.4.1 项目小结	(64)
3.4.2 项目拓展	(64)
第4章 简易画图板项目设计与开发	(65)
4.1 项目描述	(65)
4.2 项目目标	(65)
4.2.1 系统功能	(65)
4.2.2 功能描述	(65)
4.3 项目实施	(67)
4.3.1 类及 UML 设计	(67)
4.3.2 代码实现	(71)
4.3.3 系统发布	(95)
4.3.4 系统测试	(97)
4.4 项目小结与拓展	(100)
4.4.1 项目小结	(100)
4.4.2 项目拓展	(100)

第5章 简易职员管理系统设计与开发	(101)
5.1 项目描述	(101)
5.2 项目目标	(101)
5.2.1 系统功能	(101)
5.2.2 功能描述	(102)
5.3 项目实施	(105)
5.3.1 数据库设计	(105)
5.3.2 类及UML设计	(106)
5.3.3 代码实现	(112)
5.3.4 系统发布	(166)
5.3.5 系统测试	(167)
5.4 项目小结与拓展	(172)
5.4.1 项目小结	(172)
5.4.2 项目拓展	(172)
第6章 基于文件的学籍管理系统设计与开发	(173)
6.1 项目描述	(173)
6.2 项目目标	(173)
6.2.1 系统功能	(173)
6.2.2 功能描述	(173)
6.3 项目实施	(174)
6.3.1 数据结构设计	(174)
6.3.2 类及UML设计	(174)
6.3.3 代码实现	(178)
6.3.4 系统发布	(202)
6.3.5 系统测试	(203)
6.4 项目小结与拓展	(206)
6.4.1 项目小结	(206)
6.4.2 项目拓展	(206)
第7章 简单聊天室设计与开发	(207)
7.1 项目描述	(207)
7.2 项目目标	(207)
7.2.1 系统功能	(207)
7.2.2 功能描述——服务器端	(209)
7.2.3 功能描述——客户端	(209)
7.3 项目实施	(209)
7.3.1 类及UML设计——服务器端	(209)

7.3.2 类及 UML 设计——客户端	(212)
7.3.3 代码实现	(214)
7.3.4 系统发布	(243)
7.3.5 系统测试	(245)
7.4 项目小结与拓展	(246)
7.4.1 项目小结	(246)
7.4.2 项目拓展	(246)
参考文献	(247)

第 1 章

Java 项目开发基础

1995 年 Sun 公司推出了 Java 语言,该语言与其他编译执行的计算机语言和解释执行计算机语言不同,Java 语言首先将源代码编译成二进制字节码,然后依赖各种不同平台上的虚拟机来解释执行字节码,从而实现了“一次编译、到处执行”的跨平台特性,但是每次编译执行需要消耗一定的时间,在一定程度上降低了 Java 程序的运行效率,不过随着 JIT 即时编译技术的不断进步,使 Java 程序的执行效率接近 C++ 程序。

Java 语言是一种具有跨平台、纯面向对象、分布式、健壮安全、可移植、性能优异、支持多线程等特征的动态语言,它的编程风格十分接近 C++ 语言,它继承了 C++ 语言面向对象技术的核心,但舍弃了 C++ 语言中容易引起错误的指针(以引用取代)、运算符重载、多重继承(以接口取代)等特性,增加了垃圾回收器功能用于回收不再被引用的对象所占据的内存空间,使得程序员不用再为内存管理而担忧。在 Java SE 1.5 版本中,Java 又引入了泛型编程、枚举、不定长参数和自动装/拆箱等语言特性。

Java 语言是互联网时代主流程序设计语言,在最新的 TIOBE 编程语言社区排行榜中,Java 语言排名第一。目前,Java 语言主要应用在四个领域:①行业和企业信息化领域。由于 Sun、IBM、Oracle、BEA 等国际厂商相继推出各种基于 Java 技术的应用服务器以及各种应用软件,带动了 Java 在金融、电信、制造等领域日益广泛的应用。如东方科技的 Tong Web、金碟的 Apusic、中创的 Inforweb 等 J2EE 应用服务器及和佳 ERP 和宝信 ERP 等 ERP 产品,已在许多企业得到应用。②电子政务及办公自动化。东方科技、金碟、中创等开发的 J2EE 应用服务器在电子政务及办公自动化中也得到应用,如金碟的 Apusic 在民政部,东软电子政务架构 EAP 平台在社会保险、公检法、税务系统得到应用,中创的 Inforweb 等系列中间件产品在国家海事局、中国建设银行、民生银行等金融系统应用。③嵌入式设备及消费类电子产品。J2ME 是用于嵌入式设备的 Java 软件平台开发技术,专门用于开发消费电子产品的应用,如智能手机、PDA 等,此外无线手持设备、医疗设备、信息家电(如数字电视、机顶盒、电冰箱)、汽车电子设备等也是 Java 的重要应用领域。④电子商务。电子商务是当今最活跃的新商务模式,电子商务系统要求程序代码安全、可靠,同时要求能与运行在不同平台的机器的全世界客户开展业务,由于 Java 具有强安全性、平台无关性、硬件结构无关性、语言简洁同时面向对象等特点,在网络编程语言中占据无可比拟的优势,因此成为实现电子商务系统的首选语言。

随着云计算和移动互联网的快速发展,Java 语言更显示出了强大的技术优势和广阔的应用前景。

1.1 Java 项目开发实训目标

本书设计了简易计算器、简单日历记事本、简易画图板程序、简易职员管理系统、基于数据结构的学籍管理系统和简单聊天室等六个真实项目作为实训案例，通过完成这六个案例达到运用 Java 技术解决实际问题的能力。

1.1.1 实训知识目标

- (1) 了解 Java 语言的开发技术与开发工具。
- (2) 了解 Java 语言开发项目的基本规范。
- (3) 掌握 Java 语言的封装、继承和多态的面向对象编程特征。
- (4) 掌握 Java GUI 外观和事件处理编程方法。
- (5) 掌握 Java 常用类的使用方法。
- (6) 掌握 Java 数据库编程方法。
- (7) 掌握 Java 网络编程方法。
- (8) 掌握 Java 多线程编程方法。
- (9) 掌握 Java I/O 编程方法。
- (10) 学会运用各种 Java 开发与测试工具进行辅助开发。

1.1.2 实训能力目标

- (1) 根据实训项目业务背景，建立需求模型的能力。
- (2) 根据项目需求模型设计相应类的能力。
- (3) 综合运用 Java 技术实现项目目标的能力。

1.1.3 实训素质目标

- (1) 用户至上的开发理念。
- (2) 良好的项目开发规范意识。
- (3) 良好的团队合作意识。
- (4) 良好的沟通与表达能力。
- (5) 良好的自主学习能力。
- (6) 良好的创新意识。
- (7) 诚实守信，责任感强。

1.2 Java 项目开发技术

Java 开发技术是一个庞大的系统，包含的内容非常丰富，不同的应用采用的开发技术不同，下面介绍本书采用的主要 Java 开发技术。

1.2.1 图形用户界面编程技术

Java 平台的图形用户界面编程分为 GUI 外观设计和事件处理。对于 GUI 外观开发，Java

语言提供了两个图形用户界面设计的包，它们是 `java.awt` 包中的类和 `javax.swing` 包中的类，利用这两个包中的类可以完成各种复杂的图形界面设计。

`java.awt` 包提供的图形界面的各种元素和成分分为三种，分别是容器(container)、布局管理器(layout manager)和控制组件(component)。容器用来组织或者容纳其他界面成分和元素的组件，一个容器可以容纳多个组件，并且容器也可以作为组件放进另一个容器中。布局管理器可以使容器中的组件按指定位置进行摆放，并且如果容器大小改变了，布局管理器也可以进行相应调整来适应这种改变。控制组件是图形界面的最小单位，它里面不包含其他组件，例如文本框、单选按钮、下拉列表、菜单等都属于控制组件。

AWT 是基于对等体来实现 GUI，利用这种方法编写简单的程序时效果很好，但如果要编写高质量可移植的图形界面，其缺陷非常明显。因为不同平台例如 Windows 和 Solaris 的菜单、滚动条和文本框等界面成分有细微的差别，因此试图通过对等体的方式给用户一致的用户体验难以做到。针对这个问题，JDK2.0 创建了新的图形用户界面库 Swing。Swing 是轻量级的组件，它利用 Java 语言实现(AWT 利用 C 语言实现)，它不是基于对等体的 GUI，使用 Swing 能够更轻松容易构建图形用户界面。本书中的 GUI 外观设计采用 Swing 组件实现。

利用 Swing 设计了应用系统的外观，然后需要完成图形用户界面与用户的交互功能，例如通过单击 JButton 组件执行一个文件复制操作等。Java 语言提供了一组事件类来处理不同对象(组件)产生的事件从而实现用户交互功能，事件处理的类在 `java.awt` 包和 `java.util` 包中。

Java 的事件处理采用 MVC 模式，即模型(model) - 视图(view) - 控制器(controller)模式，这种模式将业务逻辑、数据、界面显示分离的方法组织代码，将业务逻辑聚集到一个部件里面，在改进和个性化定制界面及用户交互的同时，不需要重新编写业务逻辑。其中 model(模型)是应用程序中用于处理应用程序数据逻辑的部分，通常该模型对象负责在数据库中存取数据；view(视图)是应用程序中处理数据显示的部分，通常视图是依据模型数据创建的；controller(控制器)是应用程序中处理用户交互的部分，通常控制器负责从视图读取数据，控制用户输入，并向模型发送数据。

1.2.2 输入输出编程技术

在 Java 语言中，输入输出使用流来实现，包括输入流和输出流。输入流(InputStream)表示 Java 程序从外部数据源读入数据，输出流(OutputStream)表示 Java 程序向外部写出数据。数据的流动如图 1-1 所示。

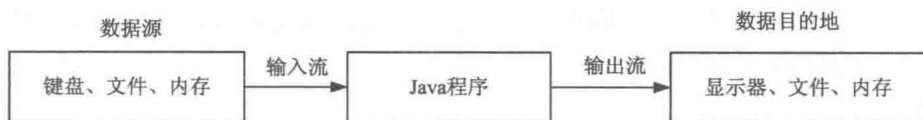


图 1-1 Java 的 I/O 处理模型

在 Java 中，I/O 流类可按读/写数据的不同类型分为字节流和字符流。字节流处理的最小数据单元是字节，字节流类可分为表示字节输入流的 InputStream 类及其子类，表示字节输

出流的 `OutputStream` 类及其子类。字符流处理的最小数据单元是字符，它包括表示字符输入流的 `Reader` 类及其子类，表示字符输出流的 `Writer` 类及其子类。

在本书的第 4、6、7 三个章节所开发的项目中，需要把创建的对象保存在文件中，在需要的时候进行恢复。Java 语言提供的对象序列化（有些书称为对象持久化）可以实现对象的传输和存储。对象的序列化是指把对象变为二进制数据写入到一个输出流中，对象的反序列化是指从一个输入流中读取一个对象。

在 Java 语言中，只有实现了 `java.io.Serializable` 接口的类的对象才能被序列化和反序列化。对象序列化和反序列化需要依靠对象输出流 `ObjectOutputStream` 和对象输入流 `ObjectInputStream`，在对象反序列化过程中，为了读出正确的对象，必须保证向对象输出流写入对象的顺序与从对象输入流读取对象的顺序一致。

1.2.3 网络编程技术

计算机网络将地理位置不同的计算机及其外部设备通过通信线路连接起来，在网络操作系统以及网络通信协议的管理和协调下，实现了资源共享和信息传递。Java 语言的 `java.net` 包提供了三种网络程序开发模式，分别是 URL 通信模式、TCP 通信模式和 UDP 通信模式。本书的第 7 章主要采用了网络编程技术。

URL 通信模式中，Java 提供了 `URL` 类、`URLConnection` 类访问网络上的资源，例如访问远程计算机的一个网页或者本地计算机的一个文件。

在 Java 中采用套接字（Socket）完成 TCP 程序的开发。Socket 套接字是当前最常用的网络通信应用程序接口之一，它采用 TCP 协议，通过提供面向连接的服务，实现客户/服务器之间双向、可靠的、点对点的通信连接。`java.net` 包中的 `Socket` 类用于建立客户端通信对象、`ServerSocket` 类用于建立服务器端通信对象。

利用 Socket 进行通信分为三个步骤：

（1）建立 Socket 连接。在通信开始之前由通信双方确认身份，然后在客户端和服务端建立一条虚拟连接线路。

（2）数据通信。利用已经建立的虚拟线路传送信息和接受信息。

（3）关闭通信线路。通信结束后将所建立的虚拟连接关闭并释放资源。

套接字编程采用 TCP 协议，在通信过程中，通信双方需要建立可靠的连接，这样浪费了大量的系统资源，降低了系统性能。UDP 通信采用的通信协议是数据报通信协议 UDP（User Datagram Protocol），通信时服务器与客户端不需要建立可靠的连接，数据以独立的包为单位发送，缺点是数据包可能丢失延误，因此 UDP 通信是不可靠的通信，但 UDP 通信速度快，因此常常被应用在要求实时交互，准确性要求不高，但传输速度要求较高的场合。目前 UDP 通信在各种聊天工具中被广泛使用。

1.2.4 多线程编程技术

进程是程序的一次动态执行过程，每个进程都有自己独立的内存空间。一个应用程序可以同时启动多个进程，线程是进程中的一个执行流程，一个进程可以由多个线程组成，即一个进程中可以同时运行多个不同的线程，每个线程完成不同的任务。一个进程内的若干个线程同时运行时，称为线程的并发运行。在 Java 语言中，通过继承 `Thread` 类和实现 `Runnable`