

21 世纪高职高专规划教材
软件专业系列

Java 高级编程实用教程

仵 博 池瑞楠 张丽涓 编著

清华大学出版社
北 京

内 容 简 介

本书是一本介绍 Java 高级编程的实用教程,面向具有一定 Java 编程基础的开发人员。本书通过对“项目”的分析、实现和讲解,使读者在实践的过程中,逐步掌握 Java 高级编程的技巧和方法。

全书分 10 章,第 1 章至第 4 章详细介绍面向对象技术和 Java 类的高级特性;第 5 章介绍数组,主要涉及数组的创建和使用;第 6 章介绍图形用户界面和事件处理;第 7 章讲述高级 I/O 处理技术;第 8 章主要介绍多线程技术,包括线程的两种创建方法、线程的基本控制、线程的同步和线程的交互;第 9 章以网络聊天室为例,以 JBuilder 9.0 为开发环境,详细介绍图形用户界面设计、事件处理方法、Socket 编程技术和多线程编程技术;第 10 章以员工管理系统为例,详细讲述如何使用 JDBC 访问数据库。附录 A 讲述 Java 编程规范,对于开发者来说具有较高的参考价值。

本书主要是面向高职高专院校高年级学生,也可作为本科学生和程序开发者参考用书。

版权所有,翻印必究。举报电话:010-62782989 13501256678 13801310933

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。

本书防伪标签采用特殊防伪技术,用户可通过在图案表面涂抹清水,图案消失,水干后图案复现;或将表面膜揭下,放在白纸上用彩笔涂抹,图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

Java 高级编程实用教程/仵博,池瑞楠,张丽涓编著. —北京:清华大学出版社,2006.7

(21 世纪高职高专规划教材·软件专业系列)

ISBN 7-302-12922-3

I. J… II. ①仵… ②池… ③张… III. Java 语言—程序设计—高等学校:技术学校—教材
IV. TP312

中国版本图书馆 CIP 数据核字(2006)第 039856 号

出 版 者:清华大学出版社

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

社 总 机:010-62770175

地 址:北京清华大学学研大厦

邮 编:100084

客户服务:010-62776969

责任编辑:宋 方

印 刷 者:北京密云胶印厂

装 订 者:三河市新茂装订有限公司

发 行 者:新华书店总店北京发行所

开 本:185×230 印张:21.75 字数:445 千字

版 次:2006 年 7 月第 1 版 2006 年 7 月第 1 次印刷

书 号:ISBN 7-302-12922-3/TP·8215

印 数:1~5000

定 价:28.00 元

出版说明

高职高专教育是我国高等教育的重要组成部分,担负着为国家培养并输送生产、建设、管理、服务第一线高素质技术应用型人才的重任。

进入 21 世纪后,高职高专教育的改革和发展呈现出前所未有的发展势头,学生规模已占我国高等教育的半壁江山,成为我国高等教育的一支重要的生力军;在办学理念上,“以就业为导向”成为高等职业教育改革与发展的主旋律。近两年来,教育部召开了三次产学研交流会,并启动四个专业的“国家技能型紧缺人才培养项目”,同时成立了 35 所示范性软件职业技术学院,进行两年制教学改革试点。这些举措都表明国家正在推动高职高专教育进行深层次的重大改革,向培养生产、服务第一线真正需要的应用型人才的方向发展。

为了顺应当前我国高职高专教育的发展形势,配合高职高专院校的教学改革和教材建设,进一步提高我国高职高专教育教材质量,在教育部的指导下,清华大学出版社组织出版了“21 世纪高职高专规划教材”。

为推动规划教材的建设,清华大学出版社组织并成立了“高职高专教育教材编审委员会”,旨在对清华版的全国性高职高专教材及教材选题进行评审,并向清华大学出版社推荐各院校办学特色鲜明、内容质量优秀的教材选题。教材选题由个人或各院校推荐,经编审委员会认真评审,最后由清华大学出版社出版。编审委员会的成员皆来源于教改成效大、办学特色鲜明、师资实力强的高职高专院校、普通高校以及著名企业,教材的编写者和审定者都是从事高职高专教育第一线的骨干教师和专家。

编审委员会根据教育部最新文件和政策,规划教材体系,比如部分专业的两年制教材;“以就业为导向”,以“专业技能体系”为主,突出人才培养的实践性、应用性的原则,重新组织系列课程的教材结构,整合课程体系;按照教育部制定的“高职高专教育基础课程教学基本要求”,教材的基础理论以“必要、够用”为度,突出基础理论的应用和实践技能的培养。

本套规划教材的编写原则如下:

- (1) 根据岗位群设置教材系列,并成立系列教材编审委员会;
- (2) 由编审委员会规划教材、评审教材;
- (3) 重点课程进行立体化建设,突出案例式教学体系,加强实训教材的出版,完善教学服务体系;
- (4) 教材编写者由具有丰富教学经验和多年实践经历的教师共同组成,建立“双师

型”编者体系。

本套规划教材涵盖了公共基础课、计算机、电子信息、机械、经济管理以及服务等大类的主要课程,包括专业基础课和专业主干课。目前已经规划的教材系列名称如下:

• 公共基础课

公共基础课系列

• 计算机类

计算机基础教育系列

计算机专业基础系列

计算机应用系列

网络专业系列

软件专业系列

电子商务专业系列

• 电子信息类

电子信息基础系列

微电子技术系列

通信技术系列

电气、自动化、应用电子技术系列

• 机械类

机械基础系列

机械设计与制造专业系列

数控技术系列

模具设计与制造系列

• 经济管理类

经济管理基础系列

市场营销系列

财务会计系列

企业管理系列

物流管理系列

财政金融系列

国际商务系列

• 服务类

艺术设计系列

本套规划教材的系列名称根据学科基础和岗位群方向设置,为各高职高专院校提供“自助餐”形式的教材。各院校在选择课程需要的教材时,专业课程可以根据岗位群选择系列;专业基础课程可以根据学科方向选择各类的基础课系列。例如,数控技术方向的专业课程可以在“数控技术系列”选择;数控技术专业需要的基础课程,属于计算机类课程的可以在“计算机基础教育系列”和“计算机应用系列”选择,属于机械类课程的可以在“机械基础系列”选择,属于电子信息类课程的可以在“电子信息基础系列”选择。依此类推。

为方便教师授课和学生学习,清华大学出版社正在建设本套教材的教学服务体系。本套教材先期选择重点课程和专业主干课程,进行立体化教材建设,加强多媒体教学课件或电子教案、素材库、学习盘、学习指导书等形式的制作和出版,开发网络课程。学校在选用教材时,可通过邮件或电话与我们联系获取相关服务,并通过与各院校的密切交流,使其日臻完善。

高职高专教育正处于新一轮改革时期,从专业设置、课程体系建设到教材编写,依然是新课题。希望各高职高专院校在教学实践中积极提出意见和建议,并向我们推荐优秀选题。反馈意见请发送到 E-mail:gzgz@tup.tsinghua.edu.cn。清华大学出版社将对已出版的教材不断地修订、完善,提高教材质量,完善教材服务体系,为我国的高职高专教育出版优秀的高质量的教材。

前言

Java 高级编程实用教程

Java 一推出,“一次编译、到处运行”的特性就讓它在 Internet 领域占领了显著的地位,越来越多的公司开始支持 Java。Sun 公司推出三种 Java 版本,J2EE™(The Java™ 2 Platform, Enterprise Edition)是目前比较流行的多层企业应用技术。它的优点之一是能够适应不同类型的客户端,如 Web 浏览器、Java Applets 和 Java 应用程序,这些客户端能够很容易地部署在笔记本电脑、台式机和工作站上,实现跨平台应用。J2SE™(The Java™ 2 Platform, Standard Edition)提供在桌面电脑上开发一般的系统,在实际中应用并不是很广泛。J2ME™(The Java™ 2 Platform, Micro Edition)是当前使用比较多的基于移动设备的开发技术。它提供了一种新型的企业客户端类型,如手机、PDA 等手持设备。

不管以上的哪种版本,都要以 Java 知识作为基础,因此学好 Java 的一些基础和高级特性,对于开发者来说是学习 Java 的第一步。本书是一本介绍 Java 高级编程的实用教程,面向具有一定 Java 编程基础的开发人员。本教材不注重讲述基础知识,而是着重讲述面向对象技术、数组、异常处理、高级 I/O 处理、多线程技术、网络编程技术和数据库开发技术。面向对象技术、数组、异常处理、高级 I/O 处理和多线程技术是 Java 语言中的高级特性,本教材本着“够用”原则,以实际工程中的应用为出发点,深入浅出地进行讲述,其中的项目精练、重点和难点突出,有助于学生对 Java 本质特性的掌握。网络编程技术和数据库开发技术是实际工程中应用较多的技术,同时又是对上述几种技术的综合应用,为巩固学生所学知识,并将这些知识应用到实际项目中,本教材在最后开发出两个较大的工程项目:“网络聊天室”和“员工管理系统”,这两个项目是所学知识的总结和提高。

全书分 10 章,第 1 章至第 4 章详细介绍面向对象技术和 Java 类的高级特性,读者应该通过这 4 章的学习,掌握如何根据 Java 类的高级特性来创建类,并能够灵活使用它。这一部分是本书的第一个重点,掌握面向对象技术是开发 Java 程序的第一步。第 5 章介绍一种线性结构——数组,主要涉及数组的创建和使用。第 6 章介绍图形用户界面和事件处理。第 7 章讲述高级 I/O 处理技术。第 8 章主要介绍多线程技术,这一部分是本书的第二个重点,包括线程的两种创建方法、线程的基本控制、线程的同步和线程的交互。

读者通过本章的学习,应该初步掌握 Java 多线程编程的思路。第 9 章以网络聊天室为例,以 JBuilder 9.0 为开发环境,详细介绍项目开发的一些方法、图形用户界面设计、事件处理方法、Socket 编程技术和多线程编程技术。第 10 章以员工管理系统为例,详细讲述如何使用 JDBC 访问数据库。附录 A 讲述 Java 编程规范,对于开发者来说具有较高的参考价值。

在本书的编写过程中,刘兴东高级工程师担任本书的主审工作,在教材的内容安排、结构组织和内容深浅取舍上给予编者大量的帮助,在此表示衷心的感谢。同时感谢以下同事对本书所提出的宝贵建议和帮助,他们是:孙湧高级工程师、蔡学军高级工程师、邹润生高级工程师、石淑华副教授、王隆杰老师、梁广民老师、齐治文老师、石光华老师、刘平老师、杨名川老师、冯延蓬老师、杨旭副教授、张喜生老师和孙宏伟老师等。

编 者

2006 年 3 月

目 录

Java 高级编程实用教程

第 1 章 面向对象程序设计概述	1
1.1 面向过程和面向对象	1
1.1.1 面向过程程序设计语言	2
1.1.2 面向对象程序设计语言	2
1.1.3 面向过程和面向对象程序设计	3
1.2 类	3
1.2.1 对象和类	4
1.2.2 类的声明	5
1.2.3 属性的声明	6
1.2.4 方法的声明	6
1.2.5 类的使用	7
1.2.6 构造器	8
1.2.7 信息隐藏和封装	12
1.2.8 【项目 1-1】的参考程序	13
1.3 Java 文件的组织	16
1.3.1 Java 源文件结构	16
1.3.2 包(package)	16
1.3.3 文件组织层次	18
1.4 本章小结	19
习题	19
第 2 章 类的设计	21
2.1 类的继承	22
2.1.1 类继承的基本使用方法	22

2.1.2 单继承	24
2.1.3 访问控制	24
2.2 方法的重写	33
2.3 多态	35
2.3.1 编译时的多态性	35
2.3.2 运行时的多态性	36
2.4 重载	43
2.4.1 普通方法的重载	43
2.4.2 构造器的重载	46
2.5 【项目 2-1】的参考程序	49
2.6 本章小结	52
习题	52
第 3 章 类的高级特性	55
3.1 static 和 final 关键字	55
3.1.1 static 关键字	56
3.1.2 【项目 3-1】的参考程序——单子(Singleton)设计模式	58
3.1.3 静态块	60
3.1.4 final 关键字	61
3.2 抽象类和接口	63
3.2.1 抽象类	63
3.2.2 接口	65
3.2.3 抽象类和接口的多态性	68
3.2.4 【项目 3-2】——抽象类和接口的综合应用	69
3.3 内部类	70
3.4 本章小结	77
习题	77
第 4 章 异常处理	79
4.1 异常的概念及其分类	80
4.1.1 Java 异常的概念	80
4.1.2 Java 异常的分类	81
4.2 Java 异常处理机制	82
4.2.1 使用 try-catch-finally 捕获异常	82

4.2.2 使用 throws 声明抛出异常	89
4.2.3 使用 throw 人工抛出异常	92
4.3 用户自定义异常	93
4.4 本章小结	94
习题	95
第 5 章 数组	97
5.1 数组的概念	98
5.2 一维数组的声明和内存分配	98
5.2.1 声明数组	99
5.2.2 内存分配	100
5.2.3 数组的初始化	100
5.3 二维数组	104
5.3.1 二维数组的声明和初始化	105
5.3.2 二维数组的引用	105
5.4 【项目 5-1】的参考程序	106
5.5 本章小结	109
习题	109
第 6 章 图形用户界面(GUI)设计	111
6.1 GUI 概述	112
6.1.1 java.awt 包	112
6.1.2 javax.swing 包	112
6.2 容器	113
6.2.1 Frame	113
6.2.2 Panel	113
6.3 Swing 组件	113
6.3.1 JFrame	113
6.3.2 JPanel	114
6.3.3 标签	114
6.3.4 列表框的应用	114
6.4 容器布局	119
6.4.1 FlowLayout 布局管理器	119
6.4.2 BorderLayout 布局管理器	120

6.4.3	GridLayout 布局管理器	120
6.5	事件处理概述	127
6.5.1	什么是事件	127
6.5.2	事件源与事件处理器	127
6.6	事件模型	127
6.6.1	分层模型(JDK 1.0)	127
6.6.2	委托模型(JDK 1.1)	128
6.7	事件类型	130
6.8	事件适配器	132
6.9	【项目 6-1】的参考程序	138
6.10	本章小结	144
	习题	144
第 7 章	I/O 处理	146
7.1	标准输入输出	146
7.2	文件和文件 I/O	148
7.2.1	文件类 File	148
7.2.2	随机访问文件 RandomAccessFile 类	149
7.3	流	157
7.3.1	输入输出流概念	157
7.3.2	输入输出类	158
7.3.3	字节流 InputStream 和 OutputStream 类	158
7.3.4	字符流 Reader 和 Writer 类	160
7.4	本章小结	168
	习题	168
第 8 章	多线程编程	169
8.1	线程概述	169
8.1.1	什么是线程	169
8.1.2	线程构成	170
8.2	创建线程	172
8.2.1	继承 Thread 类创建线程	172
8.2.2	实现 Runnable 接口创建线程	173
8.2.3	两种创建线程方法的优缺点	174

8.3	线程的基本控制	174
8.3.1	启动线程	175
8.3.2	延迟线程	176
8.3.3	停止线程	185
8.3.4	其他控制线程的方法	186
8.4	线程的同步	187
8.4.1	堆栈问题	187
8.4.2	多线程同步问题	191
8.4.3	多线程同步中的死锁问题	197
8.5	线程的交互	198
8.5.1	wait()和 notify()	199
8.5.2	生产者—消费者问题	200
8.5.3	suspend()和 resume()方法	204
8.6	本章小结	205
	习题	205
第9章	网络编程	207
9.1	项目概要设计	207
9.1.1	任务概述	207
9.1.2	对功能需求的规定	208
9.1.3	网络中的数据	209
9.2	创建聊天室界面	209
9.2.1	创建工程	209
9.2.2	创建应用程序	210
9.2.3	布局聊天室界面	213
9.2.4	应用程序执行流程	215
9.3	客户端程序	216
9.3.1	Socket 和数据流的创建和连接	216
9.3.2	添加发送昵称和性别的代码	218
9.3.3	添加退出聊天室的代码	219
9.3.4	添加发送聊天信息的代码	219
9.3.5	客户端接收服务器发送的数据	220
9.4	服务器端程序	222
9.4.1	服务器端程序分析与设计	222

9.4.2	服务器端程序	223
9.4.3	服务器端多线程	224
9.4.4	运行效果图	228
9.5	【项目 9-1】的参考程序	230
9.5.1	客户端完整代码	230
9.5.2	服务器端完整代码	239
9.6	基础理论知识	242
9.6.1	TCP/IP 四层结构模型	242
9.6.2	Socket 通信模型	243
9.6.3	基于 TCP 的 Socket 通信	245
9.6.4	基于 UDP 的 Socket 通信	250
9.7	本章小结	254
	习题	254
第 10 章	数据库编程	258
10.1	项目概要设计	258
10.1.1	任务概述	258
10.1.2	对功能需求的规定	259
10.2	使用 JDBC 建立与数据库的连接	260
10.2.1	JDBC 的两个组成部分	260
10.2.2	使用 JDBC 建立与数据库的连接	260
10.2.3	JDBC API 中的重要接口	261
10.3	用户登录模块	268
10.3.1	创建工程	268
10.3.2	创建应用程序	269
10.3.3	用户登录模块设计	272
10.3.4	应用程序执行流程	273
10.3.5	访问数据库	277
10.4	密码修改模块	279
10.4.1	创建密码修改窗体	279
10.4.2	密码修改程序	281
10.5	管理模块	283
10.5.1	创建管理模块窗体	283
10.5.2	员工查询、修改、删除程序	287

10.6	添加员工数据模块.....	290
10.6.1	员工数据添加程序.....	290
10.6.2	菜单事件处理程序.....	292
10.7	【项目 10-1】的参考程序	293
10.7.1	主类完整代码.....	293
10.7.2	MainFrame 类完整代码	295
10.7.3	ManageDialog 类完整代码	303
10.7.4	LoginDialog 类完整代码	313
10.7.5	changePasswordDialog 类完整代码.....	318
10.8	本章小结.....	321
	习题.....	321
附录 A	Java 编程规范	324
参考文献		333

面向对象程序设计概述

学习目标

- 理解面向过程和面向对象的概念。
- 掌握面向对象程序的核心概念——对象和类,以及它们之间的区别和联系。
- 掌握声明类、属性和方法的办法。
- 掌握创建对象和访问对象成员的方法。
- 掌握构造器的声明和使用方法。
- 理解类的封装的优点,并掌握如何根据封装的原则正确使用类的属性和方法。
- 理解 Java 源程序的布局和层次结构,并掌握 package 语句、import 语句的使用方法。

本章在介绍面向对象程序设计基本概念的基础上,通过一个项目的设计,学习如何根据面向对象的思想设计一个类(包括类中的属性、方法和构造器)、使用一个类(创建类的对象、访问对象成员)。其中,封装是一个重要的面向对象程序设计特征,是设计类的一个基本思想。

对于实际项目的开发,如何组织众多的 Java 文件(包括源文件和类文件)是一个很重要的问题,它关系到各个项目的开发是否能够顺利进行。1.3 节将对这个问题进行讨论。

本章的内容是面向对象程序设计的基础,也是学习后面章节的基础。如果读者已经具备面向对象程序设计的基本知识,本章内容可以适当跳过。

1.1 面向过程和面向对象

在软件开发过程中,存在面向过程(Process-Oriented, PO)和面向对象(Object-Oriented, OO)两种程序开发方法,相应地,我们将计算机语言分为面向过程程序设计语

言和面向对象程序设计语言两种。

1.1.1 面向过程程序设计语言

在 C 语言和其他面向过程的程序设计语言中,编程趋向于面向动作,程序员关心的是程序功能的实现。因此,C 程序员专注于编写函数。完成特定功能的几组动作组成函数,函数最终又组成了程序,函数是 C 语言的编程单元。在 C 语言中,数据当然是重要的,但是,数据的作用主要是支持函数执行的动作,完成函数的功能。面向过程的程序设计强调的是模块化、结构化的程序设计方法——通过函数实现。

1.1.2 面向对象程序设计语言

Java 是一种完全面向对象的程序设计语言。面向对象的编程思想力图使计算机语言对事务的描述与现实世界中该事务的本来面目尽可能地一致,这样可以增强程序的可理解性,并使得程序在稳定性、可维护性和可重用性方面更有优势。面向对象是对现实世界模型的自然体现和延伸,它认为现实世界的各种实体(甚至是虚构的事务)都可以看成对象,对象之间通过消息相互作用。

我们来看一个生活中的例子。当程序中要处理人员的生日信息时,如果采用 C 语言这样的面向过程程序设计语言时,可能需要为每个人定义三个整型变量(例如 `int year1, month1, day1`),用于保存他的生日信息。如果要保存 n 个人的生日信息,可能需要使用 $n \times 3$ 个数据(即使使用数组,也需要这么多个数组元素)。值得一提的是,这些数据本身不能体现相互之间的联系——到底哪三个数据组合在一起表示一个人的生日信息,而只能靠程序员硬性规定并记录下来。就算使用了结构体,把一个人的出生年、月、日作为结构体的成员保存在一起,也无法将对生日信息的处理(例如将某人的生日日期改为 1980 年 1 月 1 日)和生日信息有机地结合起来。由此可以看到,使用面向过程的程序设计语言设计程序时,不能很好地体现现实世界中事务的本来面目,丢失了数据本来应该携带的一些逻辑信息,增加了程序维护和重用的难度。而如果采用面向对象的程序设计语言来设计,则可以把一个人的生日信息看成一个对象,其中包括生日完整的年、月、日信息,更重要的是,这个对象中还包括对有关数据的操作功能,并且能够在数据操作时增加必要的校验和控制功能(例如可以校验所设置的日期是否是一个合法的日期),这样就大大简化了处理和传递数据的复杂程度。

从上面的例子可以看出,一个对象需要由两部分来描述:第一部分是它的“数据”,包括该对象的特性、状态等静态信息;第二部分是它的“行为”,包括对该对象的操作、对象的功能等能动信息。数据和行为是对象的两个基本特征。

例如,现实世界中的一辆汽车就可以看作一个对象,它可以行驶,这就是它的行为,另外,汽车的品牌、载客的数量、行驶公里数、行驶状态,都是它的特性和状态信息,也就是它的数据。与“汽车”相对应的是“司机”,一位司机也可以看成一个对象,司机可以驾驶汽

车,这是司机的行为;另外,司机具有人的所有属性(例如姓名、性别、年龄等),这些是司机这个对象的数据。“汽车”和“司机”这两个对象是通过行驶和驾驶这样的行为相互作用的。因为司机有驾驶汽车的技能,而汽车有行驶的本领。所以当司机在驾驶的行为中发出行驶的命令以后,汽车的行驶状态就变成“行驶中”,行驶公里数也开始逐渐增加。这样,就在对象和对象之间产生了联系。

因此,我们说,面向对象程序员关注的焦点是对象而非函数(在 Java 中称为方法),包括对象中的数据和對数据的操作——行为。

1.1.3 面向过程和面向对象程序设计

程序设计的根本任务是通过合适的计算机语言描述问题和解决问题。在以前学过的程序设计基础相关的课程中,我们学过“程序=数据结构+算法”这个著名的公式。其中,数据结构是用来描述问题的,而算法是用来解决问题的,这是面向过程程序设计的基本思想,如图 1-1 所示。

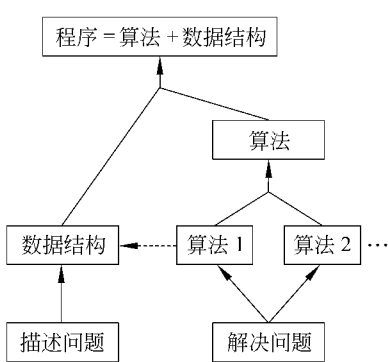


图 1-1 面向过程程序设计示意图

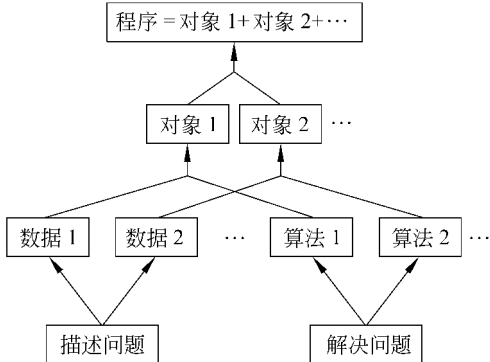


图 1-2 面向对象程序设计示意图

而面向对象的程序设计,其程序是由一个个对象组成的,对象包括数据以及作用于这些数据上的操作两个部分。因此,我们可以把面向对象的程序设计表示为如图 1-2 所示。

1.2 类

【项目 1-1】 某城市正在进行人口普查工作,需要对该城市所有人员的出生年月日信息进行记录,同时要求工作人员应该能够随时读取、修改某个人的出生年月日,并将其显示在屏幕上。

项目分析

根据【项目 1-1】的要求,我们可以按功能将其划分成以下三个模块。

模块一：记录被普查人员的出生年月日信息(设计类)；

模块二：为工作人员提供读取、修改某个人的出生信息的接口(设计类的方法)；

模块三：工作人员能够将某个人的生日信息显示在屏幕上(类的使用)。

要完成这三个功能模块的要求，需要用到类的相关知识。

1.2.1 对象和类

对于现实世界中存在的类似的多个实体，我们可以将其抽象概括为某一类事务。类似地，为了减少麻烦，对于在数据和行为上类似的多个对象(object)，可以将其抽象为一个类(class)来表示。类是对象的抽象。

正如建筑设计师可以先绘制出建筑物的图纸，然后再根据这个图纸(蓝图)建出很多相同的建筑物一样，类是对象的蓝图，可以根据实际的需要实例化出多个相互独立的对象。又例如，“人”是一个类，它描述了能够直立行走、能使用语言和生活工具、具有思维的一类高级动物，它是一个抽象的概念。而“张三”是“人”这个类的一个具体存在的对象，是实实在在的一个人，例如张三是一个高 170cm、黑眼睛、黄皮肤、25 岁的一个男人，是“人”这个类的一个实例(instance)。从这个意义上说，类(抽象的)是对象(具体的)的模板，而对象是类的实例。

类具有对象的两个要素：一是描述对象的数据元素，称为类的属性(attribute)；另外一个描述对象的行为和功能，称为类的方法(method)，它用于操作对象或执行相关对象之间的交互。属性和方法一起称为类的成员(member)。

为了完成【项目 1-1】，可以将被普查人员的生日信息看成一个类——BirthDate，而具体某个人的生日信息看成一个具体的对象。在 BirthDate 类中，可以包括用于描述生日的年、月、日信息的 year、month、day 三个属性，以及设置和获取 year、month、day 三个属性的六个方法。程序可以设计如下：

程序 1-1 设计记录生日信息的 BirthDate 类。

目的：理解类、属性和方法的概念。

分析：该程序设计了记录生日信息的 BirthDate 类，该类声明三个属性，分别用于表示生日的年、月、日信息，并提供三个设置属性和三个读取属性的方法。

```

/ ***** /
/ * 程序名: BirthDate.java * /
/ * 功能: 设计记录生日信息的 BirthDate 类 * /
/ * 作者: 池瑞楠 * /
/ * 日期: 2005-04-20 * /
/ ***** /

public class BirthDate {
    / * 声明三个属性 * /
    private int year;

```