

高职高专计算机 任务驱动模式 教材

Java程序设计基础项目化教程

李兴福 主 编

肖仁锋 刘洪海 王艳红 副主编

清华大学出版社





高职高专计算机任务驱动模式教材

Java程序设计基础项目化教程

李兴福 主 编

肖仁锋 刘洪海 王艳红 副主编



清华大学出版社
北 京

内 容 简 介

全书将“学生信息管理系统”分为8个子项目,通过完成一个个的任务,介绍Java语言开发的核心技能。项目1介绍面向对象的基本概念及面向对象的基本思想;项目2介绍Java的运行原理及开发环境的搭建;项目3介绍了Java中的基本面向对象特征;项目4介绍了Java中面向对象的高级特性;项目5介绍了Java中的集合类;项目6介绍了Java的I/O机制;项目7重点讲述了JDBC技术;项目8讲述了Java的GUI编程。8个子项目各有侧重点,阐述面向对象及Java程序设计的某一方面的特性,同时又密切关联,按照软件开发流程,完成了学生信息管理系统的程序设计。本书理论与实践结合,让学生在实践中学学习并验证理论。

本书适合作为应用型本科和高职高专计算机相关专业学生的教材,也可以作为软件开发工程师的参考书。

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 程序设计基础项目化教程/李兴福主编.--北京:清华大学出版社,2014

高职高专计算机任务驱动模式教材

ISBN 978-7-302-37803-7

I. ①J… II. ①李… III. ①JAVA 语言—程序设计—高等职业教育—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2014)第 198037 号

责任编辑:张龙卿

封面设计:徐日强

责任校对:刘 静

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>, 010-62795764

印 刷 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:17.5

字 数:417千字

版 次:2014年11月第1版

印 次:2014年11月第1次印刷

印 数:1~2500

定 价:34.00元

出版说明

我国高职高专教育经过十几年的发展,已经转向深度教学改革阶段。教育部于2006年12月发布了教高[2006]第16号文件《关于全面提高高等职业教育教学质量的若干意见》,大力推行工学结合,突出实践能力培养,全面提高高职高专教学质量。

清华大学出版社作为国内大学出版社的领跑者,为了进一步推动高职高专计算机专业教材的建设工作,适应高职高专院校计算机类人才培养的发展趋势,根据教高[2006]第16号文件的精神,2007年秋季开始了切合新一轮教学改革的教材建设工作。该系列教材一经推出,就得到了很多高职院校的认可和选用,其中部分书籍的销售量都超过了3万册。现重新组织优秀作者对部分图书进行改版,并增加了一些新的图书品种。

目前国内高职高专院校计算机网络与软件专业的教材品种繁多,但符合国家计算机网络与软件技术专业领域技能型紧缺人才培养培训方案,并符合企业的实际需要,能够自成体系的教材还不多。

我们组织国内对计算机网络和软件人才培养模式有研究并且有过一段实践经验的高职高专院校,进行了较长时间的研讨和调研,遴选出一批富有工程实践经验和教学经验的双师型教师,合力编写了这套适用于高职高专计算机网络、软件专业的教材。

本套教材的编写方法是以任务驱动、案例教学为核心,以项目开发为主线。我们研究分析了国内外先进职业教育的培训模式、教学方法和教材特色,消化吸收优秀的经验和成果。以培养技术应用型人才为目标,以企业对人才的需要为依据,把软件工程和项目的思想完全融入教材体系,将基本技能培养和主流技术相结合,课程设置中重点突出、主辅分明、结构合理、衔接紧凑。教材侧重培养学生的实战操作能力,学、思、练相结合,旨在通过项目实践,增强学生的职业能力,使知识从书本中释放并转化为专业技能。

一、教材编写思想

本套教材以案例为中心,以技能培养为目标,围绕开发项目所用到的知识点进行讲解,对某些知识点附上相关的例题,以帮助读者理解,进而将知识转变为技能。

考虑到是以“项目设计”为核心组织教学,所以在每一学期配有相应的实训课程及项目开发手册,要求学生在教师的指导下,能整合本学期所学的知识内容,相互协作,综合应用该学期的知识进行项目开发。同时,在教材中采用了大量的案例,这些案例紧密地结合教材中的各个知识点,循序渐进,由浅入深,在整体上体现了内容主导、实例解析、以点带面的模式,配合课程后期以项目设计贯穿教学内容的教学模式。

软件开发技术具有种类繁多、更新速度快的特点。本套教材在介绍软件开发主流技术的同时,帮助学生建立软件相关技术的横向及纵向的关系,培养学生综合应用所学知识的能力。

二、丛书特色

本系列教材体现目前工学结合的教改思想,充分结合教改现状,突出项目面向教学和任务驱动模式教学改革成果,打造立体化精品教材。

(1) 参照和吸纳国内外优秀计算机网络、软件专业教材的编写思想,采用本土化的实际项目或者任务,以保证其有更强的实用性,并与理论内容有很强的关联性。

(2) 准确把握高职高专软件专业人才的培养目标和特点。

(3) 充分调查研究国内软件企业,确定了基于 Java 和 .NET 的两个主流技术路线,再将其组合成相应的课程链。

(4) 教材通过一个个的教学任务或者教学项目,在做中学,在学中做,以及边学边做,重点突出技能培养。在突出技能培养的同时,还介绍解决思路和方法,培养学生未来在就业岗位上的终身学习能力。

(5) 借鉴或采用项目驱动的教学方法和考核制度,突出计算机网络、软件人才培训的先进性、工具性、实践性和应用性。

(6) 以案例为中心,以能力培养为目标,并以实际工作的例子引入概念,符合学生的认知规律。语言简洁明了、清晰易懂,更具人性化。

(7) 符合国家计算机网络、软件人才的培养目标;采用引入知识点、讲述知识点、强化知识点、应用知识点、综合知识点的模式,由浅入深地展开对技术内容的讲述。

(8) 为了便于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资源。在清华大学出版社网站(www.tup.com.cn)免费提供教材的电子课件、案例库等资源。

高职高专教育正处于新一轮教学深化改革时期,从专业设置、课程体系建设到教材建设,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,并及时反馈给我们。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继续出版优秀的高质量教材。

清华大学出版社

高职高专计算机任务驱动模式教材编审委员会

2014 年 3 月

前言

Java 语言作为面向对象程序设计语言的代表语言,自从诞生之日起就以其面向对象、简单安全、跨平台的特性而迅猛发展,至今 Java 已经是应用最广泛的编程语言之一。本书是 Java 语言的入门级教材,适合于 Java 语言的初学者。

本书使用“学生信息管理系统”一个项目贯穿全书,采用项目化的教学方式,由面向对象入手,放弃面向过程到面向对象的过渡,项目的整体架构不求知识面面俱到,而讲究实用够用,突出实践能力,循序渐进地引导读者在项目开发实践中掌握相关技能。全书将“学生信息管理系统”分为 8 个子项目,通过完成一个个的任务,讲述了 Java 语言开发的核心技能。8 个子项目分别介绍如下。

项目 1——欢迎进入 OOP 世界:本子项目主要完成学生信息管理系统的需求分析,在分析过程中介绍 OOP 的概念、特征、编程思想,还讲解了 OOA、OOD、OOP 的特征、思想与编程过程。

项目 2——开启 Java 之门:本子项目主要完成了学生信息管理系统开发平台的搭建,细致分析了 Java 的运行原理,对 JVM、JDK、JRE 等基本概念进行了介绍,设置了环境变量,完成了通过记事本与 Eclipse 开发 Java 程序的基本过程。

项目 3——类和对象:本子项目主要实现了学生类的抽象,介绍了 Java 基本的数据类型与运算符,详细讲解了对象与类的相关理论,实现了学生类的抽象,使用构造方法创建了学生对象。

项目 4——DAO 模式:本子项目通过 DAO 模式对学生类与班级类进行了业务建模与对象创建,讲解了抽象类、接口的使用方法,介绍了分支结构的实现方法,阐述了工厂模式与 DAO 模式的原理与使用。

项目 5——持有对象:本子项目实现了对学生对象与班级对象的持有,主要讲解了 Java 中 List、Set、Map 等集合的使用方法,介绍了循环与跳转语句的实现方法,阐述了泛型思想及使用泛型的优点。

项目 6——对象持久化——文件:本子项目实现了对象的文件存储,介绍了持久化的概念,讲解了 Java 输入/输出相关理论,并通过对象的序列化进行了对象流的存取。本子项目还介绍了 Java 中异常处理的相关理论。

项目 7——对象持久化——数据库:本子项目实现了对象的数据库存储,介绍了 ORM 的基本概念,讲解了 MySQL 关系数据库的使用方法,对

JDBC 理论及其编程进行了详细的讲解与说明。

项目 8——开启多彩世界：本子项目实现了学生信息管理系统图形界面的设计，主要讲解了 Java GUI 的实现，包括 Swing 的组件、布局管理器与事件处理的相关理论。

本教材具有以下几个特色。

1. 面向对象，深入浅出

本书将软件工程的思想融入其中，采用面向对象的方法进行程序的分析与设计，使用面向对象思想进行需求分析，使用面向对象思想设计程序，适时引入各种恰当的设计模式，用最浅显的语言将面向对象思想与 Java 语言讲述清楚，为读者将来向优秀软件工程师迈进奠定基础。

2. 项目驱动，层层递进

本书采用“学生信息管理系统”贯穿全书，将“学生信息管理系统”分成若干子项目，每个子项目又分成多个任务，在任务的实现过程中完成知识点的讲解，内容层层递进，使知识的讲解不再突兀，易于被学生接受。

3. 图文并茂，重点拓展

本书采用经典的“学生信息管理系统”作为开发案例，实用易懂，并配有详细的图表说明，通过图表更细致、形象、生动地展示知识的内容与操作的过程，图文并茂。对于系统中未涉及的重要知识点，本书采用知识拓展的形式，为学生形成知识体系并为今后的发展奠定基础。

4. 资源丰富，全面共享

本书配有 PPT、Flash 课件、程序源代码、配套视频供教师上课使用，并建有配套的学习网站为大家的学习提供支持。

本书的作者团队由经验丰富的一线骨干教师组成，教学经验丰富，而且参与了大量的 Java 项目的开发，实践经验丰富。在长期的 Java 教学中，将项目开发的经验融入教学中，总结出一套完善的、行之有效的教学方法，并将其融入本书中。济南职业学院的李兴福承担了本书的主要编写任务，参与本书编写的还有肖仁锋、刘洪海、王艳红等一线老师。

由于时间仓促，作者水平有限，书中难免有疏漏之处，敬请广大读者不吝指正。

编 者

2014 年 6 月

目 录

项目 1 欢迎进入 OOP 世界	1
任务 1 理解 OOP 的基本概念	1
1.1.1 任务目标	1
1.1.2 知识学习	1
1.1.3 任务实施	4
1.1.4 任务总结	6
任务 2 了解 OOP 的高级特性	6
1.2.1 任务目标	6
1.2.2 知识学习	6
1.2.3 任务实施	12
1.2.4 任务总结	13
任务 3 学生信息管理系统的需求分析	14
1.3.1 任务目标	14
1.3.2 知识学习	14
1.3.3 任务实施	15
1.3.4 任务总结	21
1.3.5 补充拓展	22
项目 2 开启 Java 之门	24
任务 1 搭建运行环境	24
2.1.1 任务目标	24
2.1.2 知识学习	24
2.1.3 任务实施	27
2.1.4 任务总结	30
任务 2 设计一个简单程序	31
2.2.1 任务目标	31
2.2.2 知识学习	31
2.2.3 任务实施	34
2.2.4 任务总结	35
2.2.5 补充拓展	36
任务 3 Eclipse 的应用	37

2.3.1	任务目标	37
2.3.2	知识学习	37
2.3.3	任务实施	38
2.3.4	任务总结	43
2.3.5	补充拓展	43
项目 3	类和对象	46
任务 1	实现学生类	46
3.1.1	任务目标	46
3.1.2	知识学习	46
3.1.3	任务实施	51
3.1.4	任务总结	54
3.1.5	补充拓展	55
任务 2	创建对象	55
3.2.1	任务目标	55
3.2.2	知识学习	56
3.2.3	任务实施	58
3.2.4	任务总结	60
3.2.5	补充拓展	61
任务 3	类的继承	63
3.3.1	任务目标	63
3.3.2	知识学习	64
3.3.3	任务实施	66
3.3.4	任务总结	68
3.3.5	补充拓展	69
项目 4	DAO 模式	72
任务 1	业务抽象	72
4.1.1	任务目标	72
4.1.2	知识学习	72
4.1.3	任务实施	74
4.1.4	任务总结	77
4.1.5	补充拓展	77
任务 2	业务的简单实现	78
4.2.1	任务目标	78
4.2.2	知识学习	78
4.2.3	任务实施	80
4.2.4	任务总结	83
4.2.5	补充拓展	83

任务 3 工厂实现	85
4.3.1 任务目标	85
4.3.2 知识学习	85
4.3.3 任务实施	87
4.3.4 任务总结	88
4.3.5 补充拓展	89
项目 5 持有对象	91
任务 1 安全持有对象	91
5.1.1 任务目标	91
5.1.2 知识学习	91
5.1.3 任务实施	94
5.1.4 任务总结	96
5.1.5 补充拓展	96
任务 2 集合存取	99
5.2.1 任务目标	99
5.2.2 知识学习	99
5.2.3 任务实施	104
5.2.4 任务总结	107
5.2.5 补充拓展	108
项目 6 对象持久化——文件	112
任务 1 创建文件	112
6.1.1 任务目标	112
6.1.2 知识学习	113
6.1.3 任务实施	117
6.1.4 任务总结	118
任务 2 CRUD——字节流	119
6.2.1 任务目标	119
6.2.2 知识学习	119
6.2.3 任务实施	123
6.2.4 任务总结	128
6.2.5 补充拓展	128
任务 3 CRUD——字符流	130
6.3.1 任务目标	130
6.3.2 知识学习	130
6.3.3 任务实施	133
6.3.4 任务总结	136
6.3.5 补充拓展	137

任务 4	CRUD——对象流	139
6.4.1	任务目标	139
6.4.2	知识学习	139
6.4.3	任务实施	141
6.4.4	任务总结	144
6.4.5	补充拓展	145
项目 7	对象持久化——数据库	148
任务 1	MySQL 关系数据库	148
7.1.1	任务目标	148
7.1.2	知识学习	148
7.1.3	任务实施	152
7.1.4	任务总结	167
任务 2	JDBC	168
7.2.1	任务目标	168
7.2.2	知识学习	168
7.2.3	任务实施	173
7.2.4	任务总结	174
7.2.5	补充拓展	174
任务 3	DML 实现	177
7.3.1	任务目标	177
7.3.2	知识学习	177
7.3.3	任务实施	179
7.3.4	任务总结	181
7.3.5	补充拓展	182
任务 4	DQL 实现	184
7.4.1	任务目标	184
7.4.2	知识学习	184
7.4.3	任务实施	186
7.4.4	任务总结	192
7.4.5	补充拓展	192
项目 8	开启多彩世界	195
任务 1	创建注册登录窗口	195
8.1.1	任务目标	195
8.1.2	知识学习	196
8.1.3	任务实施	206
8.1.4	任务总结	208
8.1.5	补充拓展	208

任务 2 添加事件处理功能	212
8.2.1 任务目标	212
8.2.2 知识学习	212
8.2.3 任务实施	216
8.2.4 任务总结	217
8.2.5 补充拓展	218
任务 3 实现用户权限管理	220
8.3.1 任务目标	220
8.3.2 知识学习	220
8.3.3 任务实施	229
8.3.4 任务总结	232
8.3.5 补充拓展	233
任务 4 实现学生信息管理	239
8.4.1 任务目标	239
8.4.2 知识学习	239
8.4.3 任务实施	244
8.4.4 任务总结	258
8.4.5 补充拓展	259
参考文献	263

项目 1 欢迎进入 OOP 世界

项目名称

欢迎进入 OOP(面向对象程序设计)世界

项目编号

Java_Stu_001

项目目标

能力目标: OOP 需求分析设计的基本能力,具备使用伪代码描述对象的能力。

素质目标: 抽象思维、模块化思维。

重点难点:

- (1) 对抽象的理解。
- (2) 对 OOP 三大特性的理解。
- (3) 伪代码需求分析。

知识提要

OOP 的概念、OOP 的特征、OOP 的基本思想、抽象与伪代码设计

项目分析

面向对象编程是当今流行的编程思想,是软件开发的主流,其思想已超越了简单的程序设计,影响了程序设计、数据库系统、应用结构、分布式系统、网络管理、人工智能等多个领域。

本项目介绍了 OOP 的基本概念与特征,通过伪代码的实现加深对这些概念与特征的理解。参与本项目时,首先应了解 OOP 的基本概念,在实现项目的过程中,锻炼抽象能力,加深对 OOP 思想的理解。

在本项目的最后还列出了学生信息管理系统的需求分析,这一系统将贯穿本书讲解的所有内容,需反复阅读,深刻理解。

任务 1 理解 OOP 的基本概念

1.1.1 任务目标

- 理解 OOP 中基本概念。
- 锻炼软件抽象能力。

1.1.2 知识学习

1. 面向对象的概念

面向对象程序设计(Object-Oriented Programming, OOP),指一种程序设计范型,同时

也是一种程序开发的方法。它将对象作为程序的基本单元,将程序和数据封装其中,以提高软件的重用性、灵活性和扩展性。

2. 面向对象与面向过程的区别

程序设计的概念最简略的描述为:程序设计=数据结构+算法。通俗一点说,程序设计指的是设计、编制、调试程序的方法和过程。

面向过程就是分析出解决问题所需要的步骤,然后用程序把这些步骤一步一步实现。

例如五子棋游戏,面向过程的设计思路就是首先分析问题的步骤:

①开始游戏;②黑子先走;③绘制画面;④判断输赢;⑤白子走;⑥绘制画面;⑦判断输赢;⑧返回步骤②;⑨输出最后结果。把上面每个步骤分别用函数来实现,问题就解决了。

面向对象的设计则是从另外的思路来解决问题,把构成问题的事物分解成若干个对象,建立对象的目的不是为了完成一个步骤,而是为了描述某个事物在整个解决问题的步骤中的行为。面向对象思维将五子棋游戏分解为:

- (1) 玩家对象,黑白双方,这两方的行为是一模一样的。
- (2) 棋盘系统,负责绘制画面。
- (3) 规则系统,负责判定诸如犯规、输赢等。

第一类对象(玩家对象)负责接收用户的输入,并告知第二类对象(棋盘对象)棋子布局的变化,棋盘对象接收到了棋子的变化,就要负责在屏幕上面显示出这种变化,同时利用第三类对象(规则系统)来对棋局进行判定。

3. 对象

面向对象使用对象来解决问题,因此对象就成为面向对象程序设计中的基础与核心,就像上面分析的五子棋游戏主要由三种对象组成的一样,三种对象被设计出来,五子棋的主要功能也就实现了。

从面向对象理论来讲,对象是系统中用来描述客观事物的一个实体,它是构成系统的一个基本单位,它可以指具体的事物,也可以指抽象的事物,如:整数1、2、3,苹果、张三、规则、法律、表单等。一个对象由一组属性和对这组属性进行操作的一组行为组成,如张三具有名字、身高、体重等属性,具有吃饭、睡觉、说话、运动等行为。从更抽象的角度来说,对象是问题域或实现域中某些事物的一个抽象,它反映了该事物在系统中需要保存的信息和发挥的作用;它是一组属性和有权对这些属性进行操作的一组操作的封装体。

图1-1表示了张三这个学生对象的基本属性与行为。

4. 类

类是具有相似属性和行为的对象的集合。类的概念来自于人们认识自然、认识社会的过程。在这一过程中,人们从一个个具体的对象中把共同的特征抽取出来,形成一个一般的概念,这就是“类”;如:昆虫、狮子、爬行动物,因为它们都属于动物的一种类别,所以归类为动物。对于一个具体的类,它有许多具体的个体,这就是属于这个类的“对象”。从面向对象理论来讲,类是对象的模板。即类是对一组有相同属性和相同操作的对象的定义,一个类所包含的方法和数据用于描述一组对象的共同属性和行为。类是在对象之上的抽象;对象则是类的具体化,是类的实例。

图1-2描述了学生类的基本属性与行为。

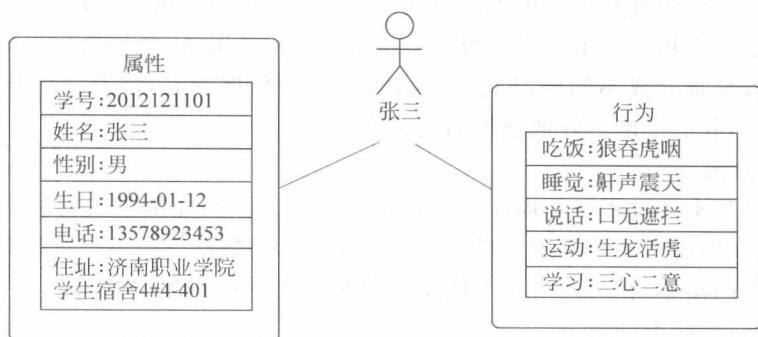


图 1-1 张三的基本属性与行为

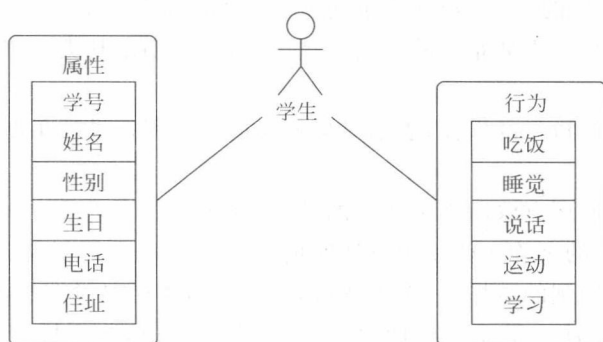


图 1-2 学生类的基本属性与行为

5. 抽象

面向对象的思想在于把软件看成是由各种各样具有特定职责的对象所组成的,不同对象之间的相互作用构成了整个软件。以面向对象的角度去进行程序设计,至少需要以下三步:

- (1) 发现并设计对象;
- (2) 确定对象的职责;
- (3) 确定对象间的相互关系。

按前面所提,类是具有相同行为对象的模板,通过同一个类可以创建不同对象,对象是类的实例。面向对象的设计程序一般从类的设计开始。由类的概念我们可以知道,类是对同一类对象共有属性和行为的抽象,要实现类,就是要用抽象的方法来归纳对象的类型、对象的属性和行为以及对象的协作关系,因此抽象的好坏决定了软件设计的成败。

(1) 抽象使我们更接近于事物的本质。抽象的过程就是提炼存在于事物之间共同拥有的元素,而这些事物之间共同拥有的元素往往是这一事物区别于其他事物关键的东西,这些元素就构成了事物的本质。

(2) 抽象的思维方式使我们能够控制问题域或者系统的复杂度,从而使我们能够找到解决问题的方式。一个问题域或者系统有很多具体的、细节的东西,这些具体的、细节的东西相互交织,使我们很难对问题域或者系统进行正确的分析,也就是我们平常所说的无从下

手。而抽象的强大优势就在于它可以使我们暂时忽略这些具体的、细节的东西,这样呈现在我们面前的就是一个相对简单的问题域或者系统。通常情况下,我们会根据抽象度的不同,将问题域或者系统划分成不同的层次,层次越往上,抽象度越高,涉及细节的东西就越少。通过这样的分析,我们就能更好地把握问题域或者系统。

6. 变量与方法

抽象对象最基本的过程就是描述对象的属性与行为。属性是对象的静态特征,通常是一些基本数据,在面向对象中使用变量来定义并存储这些数据。行为是对象的动态特征,通常是对象所进行的操作,在面向对象中使用方法来定义并运行这些操作。变量和方法是类的基本组成单元,叫作类的成员。

变量用于保存对象中的数据,比如张三这个学生的学号、姓名、性别、生日等信息。从程序设计角度讲,变量由类型、名称、值三个部分组成,例如张三的“学号”这个变量,2012121101 是这个变量的值,学号是这个变量的名称。2012121101 是一串字符串,因此字符串是这个变量的类型。再如张三的“生日”这个变量,日期是其类型,生日是其名称,1994-01-12 是其值。

方法就是对象所能执行的操作,表示对象的功能。从程序设计角度讲,方法包括定义及使用两个过程。

方法定义描述了使用方法所必须具备的信息,比如方法的名称、方法的参数、方法的返回值、方法体等。方法的名称表明使用方法时的方式。方法必须要有方法名。方法的参数表明方法执行时需要用到的数据,一个方法可以有参数,也可以没有参数。方法的返回值表明方法的结果,一个方法一般要有返回值,如果没有返回值,则要标明无返回值(通常写作 void)。方法体体现了操作的具体过程,程序中通常由一系列的语句组成。比如张三的“运动”方法:“运动”是其方法名;像跑步这样的运动可能不需要参数,而像打乒乓球这样的运动则可能需要“乒乓球拍”、“乒乓球”这样的参数;平时的运动可能就是无返回值的(void),而比赛时的运动可能就需要“名次”这样的返回值;像跑步这样的运动,其方法体就描述了跑步的具体过程,可能包括摆臂、迈步等,而像打乒乓球这样的运动,其方法体则可能包括发球、挥拍、移动、扣杀等具体动作过程。

方法的使用就是方法执行的过程,也就是对象实现功能的过程,从程序设计角度来讲,方法的使用叫作方法调用,这一过程体现在对象间的消息传递过程中。对象功能的实现通常由另一对象对其传递消息开始,传递消息一般由三部分组成:接收消息的对象、消息名及实际变元,这三部分其实就是对象、方法名、方法参数,也就是方法的调用。

面向对象中还有一类特别的方法叫作构造方法,用于类的实例化,也就是使用类创建对象。构造方法也是方法,只不过是有些特别之处的方法。

1.1.3 任务实施

1. 绘制“李四”其人

李四与张三一样,是一名在校学生,她是一位温文尔雅的女士,做事专注认真、有条不紊,请根据自己的理解,绘制“李四”其人。

(1) 抽象李四的属性

根据描述我们可以知道李四的姓名与性别,其余属性我们可以根据自己想象描述,李四

的一些可能的属性描述如下：学号——2012121102，生日——1995-02-14，电话——15523456742，住址——济南职业学院学生宿舍 5 号-2-201。

(2) 抽象李四的行为

李四做事专注认真，温文尔雅，我们可以将李四的行为抽象如下：

吃饭——细嚼慢咽；睡觉——渐入梦乡；说话——妙语连珠；运动——举止大方；学习——专心致志。

(3) 绘制“李四”其人

综合李四的属性与行为，我们绘制“李四”其人如图 1-3 所示。

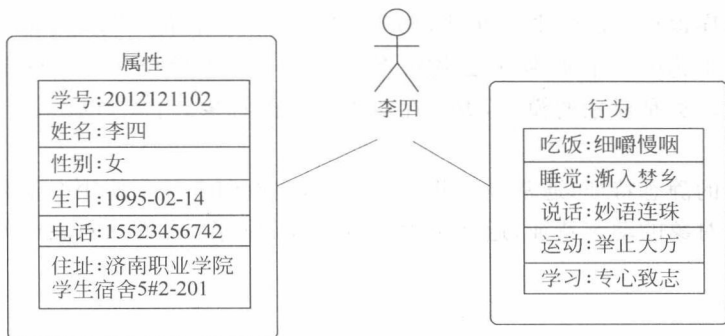


图 1-3 李四的属性与行为

2. 根据“张三”和“李四”两名学生，使用伪代码抽象学生类

伪代码不是某一门语言（如 Java、C 等）的具体代码，而是使用一种非正式的用于描述系统功能的语言，伪代码通常使用类似英语的结构。

(1) 抽象变量

学生类中的数据主要包括学号、姓名、性别、生日、电话、住址，分别使用其英文单词作为变量名，除生日是日期类型外，其余都可抽象为字符串类型，我们用英文单词 Date 表示日期类型，String 表示字符串类型。

(2) 抽象方法

学生类中的方法主要有五个：吃饭、睡觉、说话、运动、学习，分别使用其英文单词代表方法名，为了表述简单，将这些方法都定义为无参数、无返回值、空方法体，我们使用英文单词 void 表示方法无返回值，使用一对空的括号“()”表明方法无参数，使用一对空的花括号“{}”表明方法为空方法体。

(3) 学生类伪代码抽象

我们使用英文单词 class 表明后面定义的是一个类，根据上述分析，可以使用如下的伪代码来描述学生类，其中“//”后的是注释，以方便理解。

```
class Student{                                //学生类
    String number;                            //学号
    String name;                             //姓名
    String sex;                              //性别
    Date birthday;                           //生日
    String phone;                            //电话
```