

高等教育国家级教学成果二等奖

清华大学计算机基础教育课程系列教材

# Java 语言与面向对象 程序设计（第2版）

印 旻 王行言 编著



清华大学出版社

高等教育国家级教学成果二等奖

清华大学计算机基础教育课程系列教材

# Java 语言与面向对象 程序设计（第2版）

---

印 旻 王行言 编著

清华大学出版社  
北京

## 内 容 简 介

本书是《Java 语言与面向对象程序设计》的第 2 版,它系统介绍了 Java 语言与面向对象程序设计的概念、方法与基本技术。

书中首先介绍了面向对象软件开发和 Java 语言的基础知识,然后阐述了面向对象程序设计的基本原则与特点,并借助于 Java 把这些原则与特点融入具体的程序中,帮助读者逐步理解和掌握面向对象程序设计的主要技术与编程思路。本书的后半部分从应用出发,讲述 Java 编程的几个重要专题,其中包括系统类库、常用算法、图形用户界面设计、异常处理及多线程、输入输出、网络编程,以及数据库访问接口等。本书最后一章介绍了 Java 编程环境,并重点介绍了业界广泛使用的集成开发环境 Eclipse。

本书讲解条理清楚,内容深浅适中,并特别注重提高读者运用 Java 语言和面向对象技术解决问题的能力。书中给出了大量经过调试运行的实例,便于初学者入门。本书还有配套的习题解答和上机实验指导。

本书可作为高等学校 Java 程序设计课程的教材,也可作为读者的自学用书。

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 13501256678 13801310933

### 图书在版编目(CIP)数据

Java 语言与面向对象程序设计 / 印旻,王行言编著. —2 版. —北京:清华大学出版社, 2007. 11

(清华大学计算机基础教育课程系列教材)

ISBN 978-7-302-15836-3

I. J… II. ①印… ②王… III. ①JAVA 语言—程序设计—高等学校—教材 ②面向对象语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2007)第 115433 号

责任编辑:焦虹 王冰飞

责任校对:白蕾

责任印制:何芊

出版发行:清华大学出版社

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

[c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

社总机:010-62770175

投稿咨询:010-62772015

地址:北京清华大学学研大厦 A 座

邮编:100084

邮购热线:010-62786544

客户服务:010-62776969

印刷者:北京密云胶印厂

装订者:三河市新茂装订有限公司

经销:全国新华书店

开本:185×260 印张:26

字数:611 千字

版次:2007 年 11 月第 2 版

印次:2007 年 11 月第 1 次印刷

印数:1~4000

定价:34.00 元

---

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:010-62770177 转 3103 产品编号:026802-01

# 序

计算机科学技术的发展不仅极大地促进了整个科学技术的发展,而且明显地加快了经济信息化和社会信息化的进程。因此,计算机教育在各国倍受重视,计算机知识与能力已成为 21 世纪人才素质的基本要素之一。

清华大学自 1990 年开始将计算机教学纳入基础课的范畴,作为校重点课程进行建设和管理,并按照“计算机文化基础”、“计算机技术基础”和“计算机应用基础”三个层次的课程体系组织教学:

第一层次“计算机文化基础”的教学目的是培养学生掌握在未来信息化社会里更好地学习、工作和生活所必须具备的计算机基础知识和基本操作技能,并进行计算机文化道德规范教育。

第二层次“计算机技术基础”是讲授计算机软硬件的基础知识、基本技术与方法,从而为学生进一步学习计算机的后续课程,并利用计算机解决本专业及相关领域中的问题打下必要的基础。

第三层次“计算机应用基础”则是讲解计算机应用中带有基础性、普遍性的知识,讲解计算机应用与开发中的基本技术、工具与环境。

以上述课程体系为依据,设计了计算机基础教育系列课程。随着计算机技术的飞速发展,计算机教学的内容与方法也在不断更新。近几年来,清华大学不断丰富和完善教学内容,在有关课程中先后引入了面向对象技术、多媒体技术、Internet 与互联网技术等。与此同时,在教材与 CAI 课件建设、网络化的教学环境建设等方面也正在大力开展工作,并积极探索适应 21 世纪人才培养的教学模式。

为进一步加强计算机基础教学工作,适应高校正在开展的课程体系与教学内容的改革,及时反映清华大学计算机基础教学的成果,加强与兄弟院校的交流,清华大学在原有工作的基础上,重新规划了“清华大学计算机基础教育课程系列教材”。

该系列教材有如下几个特色:

1. 自成体系: 该系列教材覆盖了计算机基础教学三个层次的教学内容。其中既包括所有大学生都必须掌握的计算机文化基础,也包括适用于各专业的软、硬件基础知识;既包括基本概念、方法与规范,也包括计算机应用开发的工具与环境。

2. 内容先进: 该系列教材注重将计算机技术的最新发展适当地引入教学中来,保持了教学内容的先进性。例如,系列教材中包括了面向对象与可视化编程、多媒体技术与应用、Internet 与互联网技术、大型数据库技术等。

3. 适应面广: 该系列教材照顾了理、工、文等各种类型专业的教学要求。

4. 立体配套: 为适应教学模式、教学方法和手段的改革, 该系列教材中多数都配有习题集和实验指导、多媒体电子教案, 有的还配有 CAI 课件以及相应的网络教学资源。

本系列教材源于清华大学计算机基础教育的教学实践, 凝聚了工作在第一线的任课教师的教学经验与科研成果。我希望本系列教材不断完善, 不断更新, 为我国高校计算机基础教育做出新的贡献。



# 前言

面向对象技术被称为是程序设计方法学的一场革命,它已经逐步替代了面向过程的程序设计技术,成为计算机应用开发领域的主流技术。面向对象技术比较符合人们观察世界和处理问题的思维模式,而将数据与处理数据的操作封装在一起的机制也符合现代大规模软件开发的要求,并易于实现软件的复用。

Java 语言是面向对象技术成功应用的典范。诞生于 1995 年的 Java 语言在短短的几年间便席卷全球,以 20 世纪末网络科技和网络经济所特有的令人瞠目结舌的速度迅速发展。

进入 21 世纪,社会信息化的进程明显加快,电子政务、电子商务等术语对大家来说已不陌生,基于 Web 的应用系统得到广泛应用。就大家所熟悉的大学校园来说,教务管理系统、选课系统、网络教学系统等在学校教学活动中发挥了重要作用。而开发这些信息系统的主流技术就是 Java。

由于 Java 语言所具有的简洁性、纯面向对象等特征,也使得它非常适合于大学中面向对象程序设计的教学。从教材建设的特点出发,本书对内容的编排、剪裁和例题选择都作了严格的控制,确保了全书深度和广度适中,并遵循由浅入深、循序渐进的组织原则。本书可以作为大专院校的公共课教材,也可以作为读者的自学用书。学习本书之前应该对计算机操作有一定的认识,但不必具有编程经验。

下面简要介绍本书的主要内容与教学安排。

第 1 章 面向对象软件开发概述:讲述面向对象技术的基本思想,包括面向对象问题求解的提出、类与对象的概念,以及面向对象软件开发的一般过程,是学习面向对象程序设计的基础知识。

第 2 章 Java 概述:介绍 Java 的基本开发环境,并通过几个简单示例,使读者对 Java 程序有一个感性认识,是 Java 的入门介绍。

第 3 章 Java 语言基础:系统介绍 Java 语言的语法特征,并依次介绍了 Java 的数据类型、表达式和流程控制语句,为学习后面章节提供了语言编程基础。

第 4 章 抽象、封装与类和第 5 章 继承与多态:属本书核心内容。以 Java 的面向对象编程为主线,详细讨论了面向对象技术的四大特征:封装、抽象、继承和多态的概念及实现方法,以及接口、包等重要内容。通过这两章的学习,读者可以对面向对象技术和 Java 的面向对象编程有较为深入的理解和掌握。

要掌握好 Java 语言并具有利用它解决实际问题的能力,仅仅学习语法规则是不够的,还需要掌握 Java 的应用程序编程接口,即 Java 的类库。本书从第 6 章开始介绍 Java 的常用标准类库及一些重要的编程技术。

第 6 章 工具类与算法:首先介绍 Java 语言基础类库及 Applet 类,然后以较大篇幅

介绍了基于面向对象思想与方法的数据结构与算法。本章依次介绍了数组、向量和字符串的使用,查找、排序、递归等常用算法,以及链表、队列、堆栈、树等常见数据结构的实现及应用。

第 7 章 图形用户界面的设计与实现:介绍 Java 图形界面的设计与编程接口,其中包括常用组件的使用与事件处理机制。

第 8 章 Java 高级编程:介绍 Java 编程中的几个重要专题,其中包括异常处理、多线程编程、输入输出以及网络编程等。

第 9 章 Java 数据库编程接口:首先介绍数据库的基础知识及 SQL 语言,然后介绍 Java 数据库编程接口——JDBC,这是编写数据库应用程序的基础。

第 10 章 Java 开发环境与工具:本章介绍了基于命令行方式的 JDK 开发工具和当前最为流行的集成开发环境 Eclipse。后者为 Java 程序员提供了理想的开发平台。读者可结合本章介绍的环境上机练习各章的例题与习题。建议读者先使用 JDK 编程,在 Java 编程有了一定基础后,再尝试使用 Eclipse。

程序设计课程是一门实践性很强的课程。读者只有在学习书本内容的同时辅以相应的实际练习和实验环节,才能真正掌握书中介绍的知识和技能。为此本书中引入了大量的例题,还配有习题解答和上机实验指导书。只要读者能够按照书中的要求边学边练,就一定能很快登堂入室,享受在 Java 语言和面向对象技术所构造的无限畅想空间中遨游的乐趣。

最后感谢读者选择使用本书。由于作者水平所限,书中难免疏漏,欢迎各位同行和广大读者对本书提出修改意见和建议。

作者

2007 年 6 月

# 目 录

<b>第 1 章 面向对象软件开发概述</b> .....	1
1.1 面向对象问题求解的提出 .....	1
1.2 面向对象问题求解概述 .....	2
1.3 对象、类与实体.....	3
1.4 对象的属性与相互关系 .....	4
1.4.1 对象的属性.....	4
1.4.2 对象的关系.....	5
1.5 面向对象的软件开发过程 .....	6
1.5.1 面向对象的分析.....	6
1.5.2 面向对象的设计.....	8
1.5.3 面向对象的实现.....	8
1.6 面向对象程序设计方法的优点 .....	9
1.6.1 可重用性.....	9
1.6.2 可扩展性 .....	10
1.6.3 可管理性 .....	12
1.7 小结.....	12
习题 .....	12
<b>第 2 章 Java 概述</b> .....	14
2.1 Java 开发环境 .....	14
2.2 第一个 Java Application 程序 .....	15
2.2.1 源程序编辑 .....	16
2.2.2 字节码的编译生成 .....	17
2.2.3 字节码的解释与运行 .....	18
2.3 第一个 Java Applet 程序 .....	20
2.3.1 源程序的编辑与编译 .....	20
2.3.2 代码嵌入 .....	22
2.3.3 Applet 的运行 .....	23
2.4 图形界面的输入输出.....	25
2.4.1 Java Applet 图形界面输入输出 .....	25
2.4.2 Java Application 图形界面输入输出 .....	26
2.5 字符界面的输入输出.....	29



---

2.6	Java 语言的特点 .....	31
2.7	小结 .....	32
	习题 .....	32
<b>第 3 章</b>	<b>Java 语言基础 .....</b>	<b>33</b>
3.1	Java 程序的构成 .....	33
3.2	数据类型、变量与常量 .....	34
3.2.1	数据类型 .....	34
3.2.2	标识符 .....	35
3.2.3	常量 .....	36
3.2.4	变量 .....	38
3.3	表达式 .....	41
3.3.1	赋值与强制类型转换 .....	41
3.3.2	字符串连接 .....	42
3.3.3	算术运算 .....	43
3.3.4	关系运算 .....	45
3.3.5	逻辑运算 .....	46
3.3.6	位运算 .....	47
3.3.7	其他运算符 .....	48
3.3.8	运算符的优先级与结合性 .....	49
3.3.9	注释 .....	50
3.4	流程控制语句 .....	50
3.4.1	结构化程序设计的三种基本流程 .....	50
3.4.2	分支语句 .....	51
3.4.3	循环语句 .....	54
3.4.4	跳转语句 .....	58
3.5	小结 .....	59
	习题 .....	59
<b>第 4 章</b>	<b>抽象、封装与类 .....</b>	<b>61</b>
4.1	抽象与封装 .....	61
4.1.1	抽象 .....	61
4.1.2	封装 .....	62
4.2	Java 的类 .....	62
4.2.1	系统定义的类 .....	63
4.2.2	用户程序自定义类 .....	65
4.2.3	创建对象与定义构造函数 .....	67
4.3	类的修饰符 .....	71

4.3.1	抽象类	71
4.3.2	最终类	72
4.4	域	73
4.4.1	域的定义	73
4.4.2	静态域	74
4.4.3	静态初始化器	76
4.4.4	最终域	77
4.5	方法	78
4.5.1	方法的定义	78
4.5.2	抽象方法	79
4.5.3	静态方法	83
4.5.4	其他方法	84
4.6	访问控制符	85
4.6.1	类的访问控制	86
4.6.2	类成员的访问控制	86
4.7	类的设计	90
4.8	小结	95
	习题	95
<b>第5章</b>	<b>继承与多态</b>	97
5.1	继承的基本概念	97
5.2	类的继承	98
5.2.1	派生子类	98
5.2.2	域的继承与隐藏	101
5.2.3	方法的继承与覆盖	105
5.2.4	this 与 super	108
5.3	多态	112
5.3.1	多态概念	112
5.3.2	方法覆盖实现的多态	112
5.3.3	方法重载实现的多态	113
5.3.4	对象引用的多态	114
5.4	方法的重载	121
5.5	构造函数的重载	123
5.5.1	构造函数的重载	123
5.5.2	调用父类的构造函数	125
5.5.3	对象初始化的过程	131
5.6	包及其使用	133
5.6.1	包的基本概念	133

5.6.2	包的创建	134
5.6.3	包的使用	136
5.7	接口	138
5.7.1	接口概述	138
5.7.2	声明接口	139
5.7.3	实现接口	140
5.8	小结	142
	习题	142
<b>第 6 章</b>	<b>工具类与算法</b>	<b>146</b>
6.1	语言基础类库	146
6.1.1	Object 类	146
6.1.2	数据类型类	146
6.1.3	Math 类	148
6.1.4	System 类	148
6.2	Applet 类与 Applet 小程序	149
6.2.1	Applet 的基本工作原理	149
6.2.2	Applet 类	150
6.2.3	HTML 文件参数传递	154
6.3	数组	155
6.4	向量	158
6.5	字符串	160
6.5.1	String 类	161
6.5.2	StringBuffer 类	165
6.5.3	Java Application 命令行参数	166
6.6	递归	168
6.7	排序	170
6.7.1	冒泡排序	170
6.7.2	选择排序	173
6.7.3	插入排序	174
6.7.4	利用系统类实现排序	176
6.8	查找	177
6.8.1	查找算法	177
6.8.2	利用系统类实现查找	181
6.9	链表	182
6.9.1	链表的节点	183
6.9.2	创建链表	184
6.9.3	遍历链表	186

---

6.9.4	链表的插入操作	186
6.9.5	链表的删除操作	187
6.10	队列	188
6.11	堆栈	190
6.12	二叉树	193
6.13	小结	199
	习题	200
<b>第7章</b>	<b>图形用户界面的设计与实现</b>	<b>202</b>
7.1	图形用户界面概述	202
7.2	用户自定义成分	204
7.2.1	绘制图形	204
7.2.2	设置字体——Font类	205
7.2.3	设置颜色——Color类	207
7.2.4	显示图像	209
7.2.5	实现动画效果	210
7.3	Java的标准组件与事件处理	211
7.3.1	Java的事件处理机制	211
7.3.2	GUI标准组件概述	213
7.3.3	事件与监听者接口	215
7.4	标签、按钮与动作事件	218
7.4.1	标签	218
7.4.2	按钮	218
7.4.3	动作事件	220
7.5	文本框、文本区域与文本事件	221
7.5.1	文本框与文本域	221
7.5.2	文本事件	222
7.6	单选按钮、复选框、列表框与选择事件	224
7.6.1	选择事件	224
7.6.2	复选框	224
7.6.3	单选按钮组	227
7.6.4	下拉列表	229
7.6.5	列表框	232
7.7	设计事件处理专用类	235
7.7.1	内部类	235
7.7.2	用内部类实现事件处理	236
7.7.3	焦点事件	238
7.8	滚动条与调整事件	241

---

7.8.1	调整事件	241
7.8.2	滚动条	242
7.9	画布与鼠标、键盘事件	244
7.9.1	鼠标事件	244
7.9.2	键盘事件	247
7.9.3	画布	247
7.10	布局设计	252
7.10.1	布局管理器的概念	252
7.10.2	FlowLayout 布局管理器	253
7.10.3	BorderLayout 布局管理器	253
7.10.4	CardLayout 布局管理器	255
7.10.5	GridLayout 布局管理器	257
7.11	容器组件	258
7.11.1	容器组件类	258
7.11.2	Panel 与容器事件	259
7.11.3	Frame 与窗口事件	261
7.12	菜单的定义与使用	264
7.13	对话框及组件事件	272
7.14	Swing GUI 组件	276
7.14.1	JApplet	276
7.14.2	JButton	277
7.14.3	JSlider	280
7.14.4	JPasswordField	281
7.14.5	JTabbedPane	283
7.15	小结	285
	习题	285
<b>第 8 章</b>	<b>Java 高级编程</b>	<b>288</b>
8.1	异常处理	288
8.1.1	异常与异常类	288
8.1.2	抛出异常	291
8.1.3	异常的处理	292
8.2	Java 多线程机制	298
8.2.1	Java 中的线程	299
8.2.2	Thread 类与 Runnable 接口	300
8.2.3	如何在程序中实现多线程	302
8.3	流式输入输出与文件处理	306
8.3.1	Java 基本输入输出流类	307

8.3.2	流的类型——节点流和过滤流	310
8.3.3	几种具体的输入输出流	311
8.3.4	标准输入输出	315
8.3.5	文件的处理与随机访问	318
8.4	用 Java 实现底层网络通信	326
8.4.1	基于连接的流式套接字	327
8.4.2	无连接的数据报	334
8.5	Java 程序对网上资源的访问	339
8.6	小结	347
	习题	347
<b>第 9 章</b>	<b>Java 数据库编程接口</b>	<b>349</b>
9.1	数据库基础知识	349
9.1.1	数据库技术概述	349
9.1.2	数据库结构	350
9.2	SQL 语言简介	352
9.2.1	SQL 语言基础知识	352
9.2.2	表的创建与数据维护	353
9.2.3	数据查询	355
9.3	Access 数据库实例	360
9.3.1	Access 操作界面简介	360
9.3.2	在 Access 中创建表	361
9.3.3	表中数据的维护与浏览	362
9.3.4	创建指向 Access 数据库的数据源	363
9.4	JDBC 与数据库访问	363
9.4.1	JDBC 概述	363
9.4.2	利用 JDBC 访问数据库的基本方法	364
9.4.3	JDBC 的常用类与接口	365
9.5	Java 数据库应用实例	371
9.6	小结	374
	习题	374
<b>第 10 章</b>	<b>Java 开发环境与工具</b>	<b>375</b>
10.1	JDK 开发工具	375
10.1.1	JDK 基本命令	375
10.1.2	JDK 基本组成	378
10.1.3	JDK 的下载与安装	379
10.2	Eclipse 集成开发环境	380

10.2.1	Eclipse 安装 .....	380
10.2.2	Eclipse 界面组成 .....	381
10.2.3	Eclipse 的项目与工作空间 .....	384
10.2.4	开发一个 Java 项目的基本过程 .....	386
10.2.5	Java 编辑器使用 .....	390
10.2.6	Java 程序调试 .....	393
10.2.7	帮助信息 .....	397

# 第 1 章

## 面向对象软件开发概述

面向对象的软件开发和相应的面向对象的问题求解是当今计算机技术发展的重要成果和趋势之一。本章将集中介绍面向对象软件开发和面向对象程序设计中的基本概念和基本方法,使读者对面向对象软件开发方法的体系、原则、基本思想和特点有一定的了解。

### 1.1 面向对象问题求解的提出

早期计算机中运行的程序大都是为特定的硬件系统专门设计的,称为面向机器的程序。这类程序的运行速度和效率都很高,但是可读性和可移植性很差,随着软件开发规模的扩大,这类面向机器的程序逐渐被以 FORTRAN、C 等为代表的面向过程的程序所取代。

面向过程的程序遵循面向过程的问题求解方法。其中心思想是用计算机能够理解的逻辑来描述和表达待解决的问题及其具体的解决过程。数据结构、算法是面向过程问题求解的核心组成。其中数据结构利用计算机的离散逻辑来量化表达需要解决的问题,而算法则研究如何快捷、高效地组织解决问题的具体过程。面向过程的问题求解可以精确、完备地描述具体的求解过程(这里的过程通常是指操作),但却不足以把一个包含了多个相互关联的过程的复杂系统表述清楚,而面向对象的问题求解则可以胜任这件工作。面向对象问题求解关心的不仅仅是孤立的单个过程,而是孕育所有这些过程的母体系统,它能够使计算机逻辑来模拟描述系统本身,包括系统的组成,系统的各种可能状态,以及系统中可能产生的各种过程与过程引起的系统状态切换。

面向对象技术代表了一种全新的程序设计思路和观察、表述、处理问题的方法,与传统的面向过程的开发方法不同,面向对象的程序设计和问题求解力求符合人们日常自然的思维习惯,降低、分解问题的难度和复杂性,提高整个求解过程的可控制性、可监测性和可维护性,从而达到以较小的代价和较高的效率获得较满意效果的目的。

最早的面向对象的软件是 1966 年推出的 Simula I,它首次提出模拟人类的思维方法,把数据和相关的操作集成在一起的思想。但是由于当时硬件条件的局限和方法本身不够成熟,这种技术没有得到推广和使用。随着软件危机的出现和过程化开发方法固有局限性的暴露,人们把目光重新转回到面向对象的方法上来。1980 年提出的 Smalltalk 80 语言已经是一种比较成熟、有效的面向对象的工具了,利用 Smalltalk 80 也确实实现了一些面向对象的应用,但是这个语言更重要的作用是提出了一种新的思想观念和解决问题的新思路和新方法,它向人们展示了面向对象这个虽然稚嫩,但却充满希望的发展方



向。其后,先后产生了 Lisp、Clascal、Object Pascal 和 C++ 等多种面向对象的语言,这中间最有影响,也是对面向对象技术的普及推动最大的当属 C++。

C++ 语言在兼容原有最流行的 C 语言的基础之上,加入了面向对象的有关内容和规则。由于它的很多语法规则与 C 语言相近,所以很容易为广大的 C 程序员所接受;同时 C++ 所具有的面向对象功能简化了应用软件的开发、设计和维护,为开发大型软件提供了很大的方便。C++ 的广泛推广和成功应用证明了新兴的面向对象技术的实力和前景,C++ 也正在取代 C 而成为主流编程语言。

Java 是 20 世纪 90 年代新出现的面向对象的编程语言。相对于 C++、Java 去除了其中为了兼容 C 语言而保留的非面向对象的内容,使程序更加严谨、可靠、易懂。尤其是 Java 所特有的“一次编写、多次使用”的跨平台优点,使得它特别适合在网络应用开发中使用,成为面向对象开发工具中极具潜力的一员。

面向对象的程序设计方法的出现和广泛应用是计算机软件技术发展中的一个重大变革和飞跃。相对于之前的程序设计方法,面向对象技术能够更好地适应当今软件开发在规模、复杂性、可靠性和质量、效率上的种种需求,因而被越来越多地推广和使用,其方法本身也在这诸多实践的检验和磨炼中日趋成熟、标准化和体系化,逐渐成为目前公认的主流程序设计方法。

## 1.2 面向对象问题求解概述

不同于面向过程的程序设计中以具体的解题过程为研究和实现的主体,面向对象的程序设计是以需解决的问题中所涉及到的各种对象为主要矛盾。

在面向对象的方法学中,“对象”是现实世界的实体或概念在计算机逻辑中的抽象表示。具体地,对象是具有唯一对象名和固定对外接口的一组属性和操作的集合,用来模拟组成或影响现实世界问题的一个或一组因素。其中对象名是区别于其他对象的标志;对外接口是对象在约定好的运行框架和消息传递机制中与外界通信的通道;对象的属性表示了它所处于的状态;而对象的操作则用来改变对象的状态达到特定的功能。对象的最主要特点是以数据为中心,它是一个集成了数据和其上操作的独立、自恰的逻辑单位。

面向对象的问题求解就是力图从实际问题中抽象出这些封装了数据和操作的对象,通过定义属性和操作来表述它们的特征和功能,通过定义接口来描述它们的地位及与其他对象的关系,最终形成一个广泛联系的可理解、可扩充、可维护、更接近于问题本来面目的动态对象模型系统。

面向对象的程序设计将在面向对象的问题求解所形成的对象模型基础之上,选择一种面向对象的高级语言来具体实现这个模型。相对于传统的面向过程的程序设计方法,面向对象的程序设计具有如下的优点:

- (1) 对象的数据封装特性彻底消除了传统结构方法中数据与操作分离所带来的种种问题,提高了程序的可复用性和可维护性,降低了程序员保持数据与操作相容的负担。
- (2) 对象的数据封装特性还可以把对象的私有数据和公共数据分离开,保护了私有数据,减少了可能的模块间干扰,达到降低程序复杂性、提高可控性的目的。