



“十三五”高等教育规划教材

高等院校电气信息类专业“互联网+”创新规划教材

Java程序设计教程

(第2版)

杜晓昕 主 编

金 涛 张剑飞 副主编

Java



扫一扫联系客服



课件答案



教学视频



源代码



北京大学出版社
PEKING UNIVERSITY PRESS

高等教育规划教材

电气信息类专业“互联网+”创新规划教材

Java 程序设计教程(第 2 版)

杜晓昕 主 编

金 涛 张剑飞 副主编

贵州师范学院内部使用



北京大学出版社
PEKING UNIVERSITY PRESS

内 容 简 介

本书共分为 11 章, 主要包括 Java 语言的发展历程与特点、开发环境搭建与上机操作、Java 应用程序的编辑、编译与运行过程、数据类型、运算符与表达式、控制结构、数组、对象与类、接口与内部类、Java 常用类、I/O 流与异常处理、泛型与集合框架、多线程与图形用户界面、网络编程、数据库编程等内容。

本书按照基本概念、语句结构、程序思想、代码实现、案例分析的思路来介绍 Java 语言及涉及的相关技术, 将 Java 语言基础知识和实践应用有机结合起来。本书内容翔实、全面, 案例丰富, 并配有相关教学视频, 有利于学生对照学习, 提高学习效率, 并且所选案例密切联系实际, 力求知识性与实用性相结合。

本书既可作为高等院校计算机科学与技术、软件工程和信息管理等相关专业的教学用书, 也可作为工程技术人员参考用书。

图书在版编目(CIP)数据

Java 程序设计教程 / 杜晓昕主编. —2 版. —北京: 北京大学出版社, 2019.4

高等院校电气信息类专业“互联网+”创新规划教材

ISBN 978-7-301-30420-4

I. ①J… II. ①杜… 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 (2019) 第 055185 号

书 名	Java 程序设计教程(第 2 版) Java CHENGXU SHEJI JIAOCHENG(DI-ER BAN)
著作责任者	杜晓昕 主编
责任编辑	郑 双
数字编辑	刘 蓉
标准书号	ISBN 978-7-301-30420-4
出版发行	北京大学出版社
地 址	北京市海淀区成府路 205 号 100871
网 址	http://www.pup.cn 新浪微博: @北京大学出版社
电子信箱	pup_6@163.com
电 话	邮购部 010-62752015 发行部 010-62750672 编辑部 010-62750667
印 刷 者	大厂回族自治县彩虹印刷有限公司
经 销 者	新华书店
	787 毫米×1092 毫米 16 开本 24.25 印张 572 千字
	2011 年 9 月第 1 版
	2019 年 4 月第 2 版 2019 年 4 月第 1 次印刷
定 价	58.00 元

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有, 侵权必究

举报电话: 010-62752024 电子信箱: fd@pup.pku.edu.cn

图书如有印装质量问题, 请与出版部联系, 电话: 010-62756370

第 2 版前言

Java 语言是在 C 和 C++ 语言的基础上进行简化和改进的一种新型语言, 是随着 Internet 以及信息技术的飞速发展而发展起来的。它具有面向对象、与平台无关、安全、稳定、多线程的特点以及强大的网络编程功能, 已成为编程者的首选工具之一。目前, 国内外的高等教育中, 部分学校将 Java 语言列为本科计算机类及相关专业的第一门编程语言。为了帮助读者更好、更快地学习和掌握 Java, 编者编写了本书。通过本书, 读者能够系统地学习和掌握 Java 基本知识、集合框架、网络编程以及数据库编程等内容。本书也可作为 Java 初学者的首选参考书。

随着 Java 技术的不断更新与发展, 结合多年的教学经验, 编者对本书第 1 版内容作了较大的更新和改动。在第 2 版中删除了第 1 版的第 9~13 章, 增加了 Java 常用类、泛型与集合框架和 Java 数据库编程, 并在每章最后增加了案例分析, 教学内容更加科学合理。同时, 本书第 2 版升级为“互联网+”教材, 编者对各章的重点、难点内容录制了教学视频, 教学视频、程序源代码和习题答案都可通过扫描二维码直接获取, 帮助读者更好、更快地理解内容。

与同类教材相比, 本书具有以下特色。

(1) 注重基础与应用。本书按照程序设计思维的主要流程来安排各章节, 易于读者理解。每章均按照基本概念、语句结构、程序思想、代码实现、案例分析的思路来介绍 Java 语言, 有利于学习者对照学习, 提高学习效率。本书采用由基础到应用的循序渐进的学习模式, 适合学习者全面掌握 Java 语言。

(2) 案例驱动。每章都有经典案例分析, 通过经典案例将各知识点有机地结合起来, 达到学以致用目的。本书注重提高读者利用面向对象技术和 Java 语言解决实际问题的能力。

(3) 教学便利。本书采用可视化开发工具与代码解读相结合的方法, 既能使学习者直观感受设计开发的高效, 也能使学习者回味相应代码的作用。这符合人们认识事物的心理过程, 也平衡了实践的操作直观性与理论的系统完整性; 同时还能充分调动学习者的学习积极性和主动性, 给教师提供了更大的教学设计空间。

(4) 视频讲解。编者对各章的重点、难点内容录制了 68 个讲解视频, 帮助读者更好地理解相关知识。

(5) 注重碎片化学习。本书为“互联网+”教材, 学生可以随时随地扫码观看重点、难点内容的讲解, 并获取习题答案和程序源代码。

综上, 本书集基础知识、案例驱动、技术实用与教学便利于一体, 充分体现软件工程的理念, 兼顾学习与应用, 是一本适合 Java 程序设计初学者、高等院校教学和“卓越工程师”人才培养的教材。

本书在内容体系上共分为 11 章。第 1 章介绍 Java 语言的发展历程及特点、开发环境的搭建、开发工具的使用以及 Java 应用程序的编辑、编译与运行过程等。第 2 章介绍 Java

程序设计的基础知识,包括数据类型、运算符与表达式、流程控制、数组等。第3章介绍面向对象的概念、类的基本组成、对象的引用、方法的重载与重写、继承以及常用的修饰符等。第4章介绍抽象类的使用场合及定义方法、接口的定义与实现、抽象类和接口的异同、JDK8 接口的新特性、多态必须满足的条件及其优势、内部类的定义及使用方法等。第5章介绍基本数据类型的封装类、Object 类、字符串处理类、Math 类及日期处理类等 Java 常用类。第6章介绍 File 类、常见的输入/输出流、异常的概念、异常的处理机制以及自定义异常等。第7章介绍 Java 集合框架中主要的类和接口及其关系,包括接口 List、Set 集合和 Map 集合的创建、访问和遍历方法以及集合工具类的用法等。第8章介绍线程的概念与使用、线程同步与共享、多线程操作等。第9章介绍 TCP/IP、UDP、Socket 协议,以及如何利用 Java 语言进行网络编程等。第10章介绍图形用户界面的创建流程、布局管理器、常用 Swing 组件的功能和用法以及相应的事件处理机制等。第11章介绍 MySQL 数据库、MySQL 图形化软件管理工具 Navicat Premium、JDBC API 中的常见接口或类以及学生信息管理系统等。

本书已经在腾讯课程上开设免费课程,作者精心准备了 68 个一共长达 322 分钟的高清教学视频,读者可以在腾讯课程上搜索“Java 程序设计教程”关键字查找,也可以直接输入下方网址进行学习: <https://ke.qq.com/course/395759>。

本书由杜晓昕担任主编,金涛、张剑飞担任副主编,其中第 1、3、5、11 章由杜晓昕编写,第 4、7、8、9 章由金涛编写,第 2、6、10 章由张剑飞编写。本书得到齐齐哈尔大学 2010 重点教材项目资助。此外,感谢北京大学出版社为本书提供的二维码扫描技术,感谢郑双等编辑对本书认真细致的审稿。

由于编者水平有限,书中难免存在一些疏漏和不妥之处,敬请专家和读者批评指正。

编 者

2018 年 10 月



【资源索引】

目 录

第 1 章 Java 语言概述	1	2.4.3 循环结构	40
1.1 Java 语言简介	2	2.4.4 跳转结构	44
1.1.1 Java 语言发展简介	2	2.5 数组	46
1.1.2 Java 语言的特点	3	2.5.1 一维数组	46
1.2 开发环境的搭建	4	2.5.2 多维数组	52
1.2.1 下载 JDK	5	2.6 案例分析	55
1.2.2 安装 JDK	5	2.6.1 最大公约数和最小公倍数	55
1.2.3 设置 path 与 classpath	6	2.6.2 百鸡问题	56
1.2.4 测试 Java 开发环境	8	2.6.3 猴子吃桃子问题	57
1.3 用命令行方式描述 Java 应用程序的 开发过程	8	2.6.4 折半查找	58
1.3.1 源程序的编辑	9	2.6.5 杨辉三角	59
1.3.2 源程序的编译	10	小结	60
1.3.3 字节码文件的运行	11	习题	60
1.4 辅助工具简介	12	第 3 章 面向对象基础	64
1.5 Eclipse 集成开发工具简介	14	3.1 面向对象的基本特征	65
小结	17	3.2 类	66
习题	17	3.2.1 类的定义	66
第 2 章 Java 程序设计基础	18	3.2.2 成员变量和局部变量	67
2.1 标识符与关键字	19	3.2.3 成员方法	70
2.1.1 标识符	19	3.3 对象的创建和构造方法	72
2.1.2 关键字	19	3.3.1 对象的声明	72
2.2 数据类型	20	3.3.2 对象的创建	72
2.2.1 基本数据类型	20	3.3.3 对象的使用	73
2.2.2 常量	21	3.3.4 构造方法	73
2.2.3 变量	22	3.4 方法重载	75
2.2.4 基本数据类型之间的转换	23	3.5 类的继承	76
2.3 运算符与表达式	25	3.5.1 继承的定义	76
2.3.1 运算符	25	3.5.2 成员变量的隐藏和方法重写 (覆盖)	78
2.3.2 表达式	30	3.5.3 this 与 super 关键字	81
2.4 控制结构	31	3.5.4 继承中的构造方法	85
2.4.1 顺序结构	31	3.6 包	87
2.4.2 选择结构	31	3.6.1 包的声明	88
		3.6.2 包的导入	88

3.7 权限控制	88	5.4 Math 类	159
3.7.1 公有访问修饰符: public	89	5.5 日期处理类	160
3.7.2 保护访问修饰符: protected	90	5.5.1 Date 类	160
3.7.3 默认访问修饰符: default	90	5.5.2 Calendar 类	162
3.7.4 私有访问修饰符: private	91	5.6 案例分析	165
3.8 关键字 final 与 static	93	5.6.1 进制转换	165
3.8.1 关键字 final	93	5.6.2 校验文件名和邮箱地址	166
3.8.2 关键字 static	96	5.6.3 批量单词替换和统计问题	168
3.9 案例分析	99	5.6.4 万年历	169
3.9.1 图书管理系统	99	小结	170
3.9.2 超市售货管理系统	106	习题	171
小结	113	第 6 章 I/O 流与异常	174
习题	113	6.1 File 类	175
第 4 章 抽象类、接口与内部类	118	6.1.1 File 类的构造方法	175
4.1 抽象类	119	6.1.2 File 类的成员方法	175
4.2 接口	121	6.1.3 使用 File 类	177
4.2.1 接口的定义	121	6.2 流	178
4.2.2 接口的实现	122	6.2.1 流的基本概念	178
4.2.3 抽象类和接口的异同	123	6.2.2 输入/输出流	178
4.2.4 JDK8 接口新特性	124	6.3 字节流	180
4.3 多态	126	6.3.1 InputStream 和 OutputStream	180
4.4 内部类	128	6.3.2 FileInputStream 和 FileOutputStream	181
4.5 案例分析	130	6.4 字符流	182
小结	139	6.4.1 Reader 和 Writer	182
习题	139	6.4.2 InputStreamReader 和 OutputStreamWriter	183
第 5 章 Java 常用类	143	6.4.3 FileReader 和 FileWriter	184
5.1 基本数据类型的封装类	144	6.4.4 BufferedReader 和 BufferedWriter	185
5.1.1 封装类的构造方法	144	6.4.5 PrintStream 和 PrintWriter	187
5.1.2 封装类的常用方法	145	6.5 序列化	188
5.1.3 自动装箱与自动拆箱	145	6.5.1 对象序列化	188
5.2 Object 类	146	6.5.2 对象解序列化	190
5.2.1 toString()方法	146	6.6 异常	193
5.2.2 equals(Object obj)方法	148	6.6.1 异常的概念	193
5.2.3 getClass()方法	149	6.6.2 异常处理	193
5.3 字符串处理类	150	6.6.3 使用 throws 声明异常	197
5.3.1 String 类	150		
5.3.2 StringBuffer 类	156		
5.3.3 StringBuilder 类	158		

6.6.4 使用 throw 抛出异常	198	8.3.2 线程的控制	264
6.6.5 异常的多态	199	8.4 线程的同步	272
6.6.6 自定义异常	199	8.4.1 同步方法	272
6.7 案例分析	201	8.4.2 同步块	276
6.7.1 在文本中对指定字符串进行 查找与替换	201	8.4.3 多线程产生死锁	278
6.7.2 取钱	205	8.5 案例分析	278
小结	207	8.5.1 生产者-消费者案例	278
习题	208	8.5.2 多线程实现排序案例	281
第 7 章 泛型与集合框架	212	小结	285
7.1 泛型	213	习题	286
7.1.1 泛型定义	213	第 9 章 Java 的网络程序设计	288
7.1.2 通配符	215	9.1 基础知识	289
7.1.3 有界类型	217	9.1.1 TCP/IP 分层结构	289
7.1.4 泛型的限制	220	9.1.2 套接字概述	292
7.2 集合框架简介	221	9.2 Java 网络包(java.net)	293
7.3 接口 Collection	223	9.2.1 服务器端 ServerSocket	293
7.4 接口 List	224	9.2.2 客户端 Socket	294
7.4.1 ArrayList 类	226	9.2.3 使用 BufferedReader 从 Socket 上 读取数据	296
7.4.2 LinkedList 类	233	9.2.4 使用 PrintWriter 写数据到 Socket 上	296
7.5 Set 集合	235	9.3 Socket 编程实例	297
7.6 Map 集合	237	9.3.1 单客户端通信	297
7.7 集合工具	244	9.3.2 多客户端聊天程序	299
7.8 案例分析	247	9.4 案例分析	303
7.8.1 用 Collection 实现图书的添加和 查看	247	小结	312
7.8.2 用 TreeSet 实现信息的存储和 查找	250	习题	313
小结	253	第 10 章 图形用户界面	315
习题	253	10.1 图形用户界面概述	316
第 8 章 多线程程序设计	256	10.2 事件处理	316
8.1 线程的概念	257	10.2.1 事件处理模型	316
8.2 线程的创建和启动	257	10.2.2 事件类	318
8.2.1 继承 Thread 类	258	10.2.3 事件监听器	319
8.2.2 实现 Runnable 接口	260	10.2.4 事件及其相应的监听器 接口	320
8.2.3 两种线程创建方式比较	260	10.3 Swing 组件	323
8.3 线程的状态与控制	263	10.3.1 窗体——JFrame 类	324
8.3.1 线程的状态	263	10.3.2 面板——JPanel 类	325
		10.3.3 标签——JLabel 类	327

10.3.4 按钮——JButton 类	328	11.2 JDBC 简介	359
10.3.5 文本框——JTextField 类与 JPasswordField 类	329	11.3 JDBC 的 API 接口	360
10.3.6 文本区——JTextArea 类	331	11.3.1 DriverManager 类	360
10.3.7 列表组件——JComboBox 类和 JList 类	331	11.3.2 Driver 接口	361
10.3.8 复选框和单选按钮——JCheckBox 类和 JRadioButton 类	335	11.3.3 Connection 接口	361
10.4 布局管理器	338	11.3.4 Statement 接口	362
10.5 案例分析	342	11.3.5 PreparedStatement 接口	363
小结	348	11.3.6 ResultSet 接口	364
习题	348	11.4 案例分析	365
第 11 章 Java 数据库编程	352	11.4.1 下载并加载 MySQL 数据库 驱动	365
11.1 MySQL 数据库	353	11.4.2 连接数据库	366
11.1.1 下载与安装 MySQL 数据库	353	11.4.3 数据库的插入	368
11.1.2 安装 MySQL 图形化管理工具 Navicat Premium	356	11.4.4 数据库的查询	370
		11.4.5 数据库的更新	372
		11.4.6 数据库的删除	373
		小结	375
		习题	375
		参考文献	379

Java 语言概述

学习目标

内 容	要 求
Java 语言的产生	了解
Java 语言的特点	熟悉
JDK 的下载与安装	掌握
环境变量的设置	掌握
Java 应用程序的编辑、编译、运行	掌握
EditPlus 的安装与使用	掌握
Eclipse 的安装与使用	掌握

Java 是一种新型的面向对象的编程语言。它是随着 Internet 及信息技术的飞速发展而发展起来的，是目前最常用的一种功能强大的跨平台的计算机编程语言，是主要的网络开发语言之一，也是发展迅速的嵌入式操作系统的绝佳组合。由于 Java 语言开源、提供功能丰富的类库，而且具有面向对象、分布式、多线程、可移植性、安全性高和稳定性强等特点，目前重量级的公司都广泛采用 Java 语言进行项目的开发。Java 语言已经在众多的高级语言中脱颖而出。



【第 1 章 代码下载】

1.1 Java 语言简介

在学习 Java 语言之前, 首先了解一下 Java 语言的发展历程及其特点。

1.1.1 Java 语言发展简介

1991 年, 美国 Sun 公司为了能够在消费电子产品上开发应用程序, 成立了“绿色项目组”(Green Project), 该小组主要由 James Gosling 负责, 成员主要包括 Patrick Naughton、Chris Warth、Ed Frank 和 Mike Sheridan 等。这个小组最初的目标是能够在诸如电冰箱、电视机、PDA 等数字控制的电子消费产品上开发应用程序, 然而消费电子产品种类繁多, 即使是同一类消费电子产品, 其采用的处理芯片和操作系统也不相同, 也存在着跨平台的问题。当时最流行的编程语言是 C 和 C++ 语言, 该小组的研究人员就考虑是否可以采用 C++ 语言来编写消费电子产品的应用程序。但是研究表明, 对于消费电子产品而言, C++ 语言过于复杂和庞大, 并不适用, 安全性也并不令人满意。于是该小组就以 C++ 为基石, 融合 C 和 C++ 等传统语言的优点, 开发了一种独立于硬件平台的、面向对象的程序设计语言, 并命名为 Oak(取名自 Gosling 办公室外的一棵橡树)。当时, Oak 语言并没有引起人们的注意。

直到 1994 年, 随着互联网和 WWW 的飞速发展, James Gosling 认为市场需要一种不依赖实际硬件和软件环境、安全可靠、可交互的浏览器, Sun 公司发现 Oak 语言所具有的跨平台、面向对象、安全性高等特点非常符合互联网的需要。于是, “绿色项目组”将他们的开发目标转向了 Internet, 用 Oak 语言编写了一系列网络应用程序, 例如, 网络浏览器 WebRunner 等。

1995 年, 由于商标冲突, Oak 语言被改名为 Java 语言。同年, WebRunner 正式改名为 HotJava。HotJava 浏览器得到了 Sun 公司首席执行官 Scott McNealy 的支持, 并得以研发和发展。这个完全用 Java 语言设计的浏览器不仅充分显示了 Java 语言环境的威力, 而且为在更复杂、离散、异构的 Internet 网上进行分布式 Java 编程提供了一个理想的平台。后来, Sun 公司又决定让程序开发者免费使用 Java, 这才真正地將 Java 推向了全世界。

其实 Java 名字的由来还流传着一个故事, 一天, Java 小组成员正在喝咖啡时, 议论给新语言改个什么名字的问题, 有人提议用 Java[Java(爪哇)是印度尼西亚盛产咖啡的一座岛屿], 这个提议得到了其他成员的赞同, 于是就采用 Java 来命名此新语言。

Sun 公司虽然推出了 Java, 但这仅仅是一门编程语言, 如果想开发比较大的项目则必须要有一个强大的开发类库, 于是 Sun 公司在 1996 年推出了 JDK1.0。该版本包括两个方面: JRE(Java RunTime Environment, Java 运行环境)和 JDK(Java Development Kit, Java 软件开发工具包)。在 JRE 中包括 API(核心 API、用户界面 API、集成 API)、发布技术、JVM(Java Virtual Machine, Java 虚拟机); JDK 包括编译 Java 程序的编译器(javac 命令)、解释器(java 命令)等。Sun 公司在 1997 年推出 JDK1.1, 新增了 JIT(Just In Time Compiler, 即时编译器)。它与传统编译器的区别在于, 传统编译器只能编译一条语句, 运行完后扔掉, 再编译下一条语句; 而 JIT 则是将经常用到的指令保存在内存中, 当下次调用时不需要再编译, 大大

提高了 JDK 的效率。

一直以来,Java 主要应用在网页的 Applet 上以及一些移动设备中。但是,到了 1996 年年底,Flash 的面世动摇了 Java 在网页 Applet 上的应用地位。

虽然从 1995 年 Java 诞生到 1998 年,Java 依然是互联网上使用最广的语言,但是 Java 并没有找到它自己准确的位置。直到 1998 年年底,Sun 公司推出了 JDK1.2,这是 Java 发展史上最重要的版本之一,其中加入了许多新的设计。

2000 年 5 月,JDK1.3 发布,对 JDK 1.2 版本进行了改进,扩展了标准类库。

2002 年 2 月,Sun 公司发布了 Java 发展史上最为成熟的版本 JDK1.4,自此 Java 的计算能力有了大幅提升。

2004 年 9 月,J2SE1.5 发布,成为 Java 语言发展史上的一座里程碑。为了表示该版本的重要性,J2SE 1.5 更名为 Java SE 5.0。

2005 年 6 月,JavaOne 大会召开,Sun 公司发布了 Java SE 6。此时,Java 的各种版本已经更名,已取消其中的数字“2”:J2EE 更名为 Java EE(Java Enterprise Edition),J2SE 更名为 Java SE(Java Standard Edition),J2ME 更名为 Java ME(Java Micro Edition)。

2006 年 11 月,Sun 公司宣布 Java 技术作为免费软件对外发布。

2011 年 7 月,甲骨文(Oracle)公司发布 Java SE 7。这也是 Oracle 公司收购 Sun 公司后发布的一个重要版本。

2014 年 3 月 18 日,Oracle 公司发布 Java SE 8,这次版本升级为 Java 带来了全新的 Lambda 表达式。除此之外,Java SE 8 还增加了大量新特性,这些新特性使得 Java 变得更加强大。

2017 年 9 月,Oracle 公司发布 Java SE 9.0,主要是引入了一种新的 Java 编程组件,也就是模块。

2018 年 4 月,Oracle 公司发布 Java SE 10.0,主要是少部分 API 更新及 bug 修复。

由于 Java 提供了强大的图形、图像、音频、视频、多线程和网络交互能力,它已经成为当今推广最快的、最流行的网络编程语言。Java 的出现引起了软件开发的重大变革,成为推动 IT 业蓬勃发展的最新动力。它的出现对整个计算机软件业的发展产生了重大而深远的影响。

目前,Java 技术通常分为 3 大部分:Java SE、Java ME 和 Java EE。

Java SE 主要用于桌面应用程序的编程,为台式机和 workstation 提供一个开发和运行的平台。它是最基础的 Java 技术,定义了一般的 Java 语言规范,如程序界面、I/O、多线程和网络编程等。本书在学习 Java 的过程中,主要是采用 Java SE 来进行开发。

Java ME 是一种高度优化的 Java 运行环境,主要是面向消费类电子设备(如手机、机顶盒、PDA 等)提供的一个 Java 运行平台。

Java EE 主要是为实现分布式企业开发提供的一个应用服务器的运行和开发平台。

1.1.2 Java 语言的特点

Java 语言是一门重要的网络编程语言,具有的特点如下。

1. 简单性

Java 语言是在 C 和 C++语言的基础上进行简化和改进的一种新型语言,它的语法与 C

和 C++ 语言的语法类似, 简单且容易掌握。同时 Java 语言摒弃了 C 和 C++ 语言的复杂、不安全特性, 如摒弃了 C 语言的全局变量、宏定义、全局函数, 以及结构、联合和指针数据类型、指针的操作和内存的管理等。此外, Java 语言提供了种类丰富、功能强大的类库, 并且通过垃圾自动回收机制简化了程序内存管理, 使 Java 程序变得简单容易编写。Java 程序的简单性是其得以迅速普及的重要原因之一。

2. 完全面向对象

在现实世界中, 任何实体都可以看作是一个对象。面向对象模型是一种模拟人类社会和人解决实际问题的模型, 它更符合人们的思维习惯。Java 语言是一种完全面向对象的编程语言, 它将数据封装于类中, 利用类的优点, 实现了程序的简洁性和便于维护性。面向对象也是 Java 语言最重要的特性。

3. 平台无关性

平台无关性有两种: 源代码级和目标代码级。C 和 C++ 具有一定程度的源代码级平台无关性, 用 C 和 C++ 语言编写的应用程序不用修改, 只需重新编译就可以在不同平台上运行。Java 的平台无关性是目标代码级的, 是指 Java 语言编写的应用程序的目标文件直接可以在不同的软、硬件平台上运行, 这也是 Java 语言具有“一次编译, 到处运行”外号的原因。Java 语言的平台无关性主要是由 JVM 实现的。

4. 安全性

现今的 Java 语言主要用于网络应用程序的开发, 因此对安全性有很高的要求。Java 语言去除了 C 和 C++ 语言中易造成错误的指针, 增加了自动内存管理等措施; 同时, Java 语言提供了异常处理机制, 有效地避免了因程序编写错误而导致的死机现象, 保证了 Java 程序运行的安全稳定。

5. 多线程

多线程机制类似于多进程机制, 多线程机制使一个进程能够被划分为若干线程并并发执行。多线程机制能够带来更好的交互性能和实时控制性能。C 和 C++ 语言采用单线程体系结构, 而 Java 语言支持多线程技术。

1.2 开发环境的搭建

要编写一个 Java 程序, 必须先安装开发环境, 开发环境包括开发 Java 程序必需的 JDK 工具和一个编辑软件。

JDK 是 Java 软件开发工具箱, 提供了编译和运行 Java 程序的所有工具和常用的类库。

编辑软件可以使用计算机上的任何一个文本编辑器, 如记事本、UltraEdit、EditPlus、TextPad 等。另外, 对于大型项目开发来说, 为了用户更方便地进行程序的编写及调试, 可以使用功能强大的集成开发环境(Integrated Developing Environment, IDE), 如 JCreator、Eclipse、JBuilder 等, 这些 IDE 都提供了拼写检查、代码自动完成、关键字特殊显示、第三方插件等功能。



【教学视频】

本小节主要对 JDK 的下载、安装、配置和测试进行详细的讲解。

1.2.1 下载 JDK

Oracle 公司于 2010 年 1 月完成对 Sun 公司的收购, 所以, JDK 可以从 Oracle 公司的官方网站“<http://www.oracle.com>”免费下载。JDK 在本书编写时的最新版本为 jdk-10.0.1, 读者可以根据不同的操作系统平台来下载相应的 JDK, 本书以 64 位的 Windows 10 系统为例, 介绍其下载的具体过程。

(1) 在浏览器的地址栏中输入“<http://www.oracle.com/technetwork/java/javase/downloads/index.html>”, 打开如图 1.1 所示的页面(因网页更新等情况, 读者打开的网页可能与书中介绍的略有不同)。

(2) 在页面里单击“Downloads”按钮, 进入如图 1.2 所示的 Java SE 下载页面。



图 1.1 进入 Java 的官方网站



图 1.2 Java SE 下载页面

(3) 进入 Java SE 下载页面后, 根据机器的操作系统, 选择相应的 JDK 版本, 这里选择“Windows”, 并选中“Accept License Agreement”单选按钮, 如图 1.3 所示。

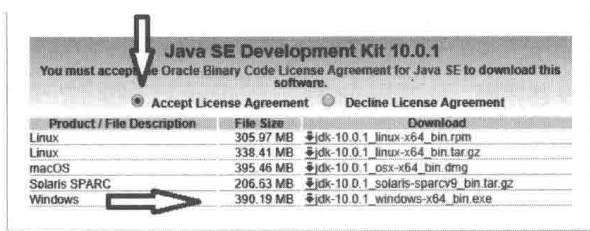


图 1.3 JDK 文件下载页面

(4) 单击“jdk-10.0.1_windows-x64_bin.exe”链接进行下载。

1.2.2 安装 JDK

下载完成后, 即可进行安装, 下面以 Windows 10 操作系统为例介绍 JDK 的安装步骤。双击已下载的安装程序 jdk-10.0.1_windows-x64_bin.exe, 运行 Java SE 的安装程序, 如图 1.4 所示。

(1) 选择需要安装的功能组件, 单击“更改”按钮, 可更改 JDK 的安装目录, 如图 1.5 所示。完成设置后单击“下一步”按钮, 继续安装。



【教学视频】

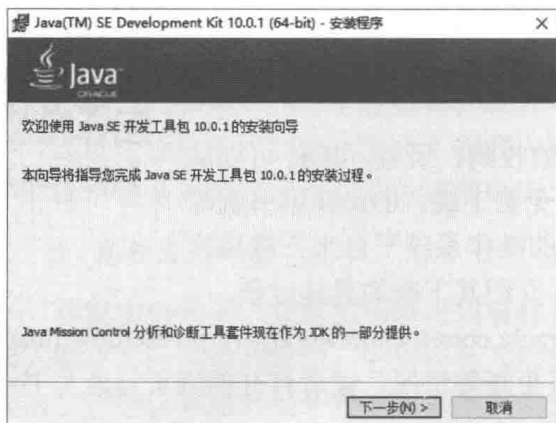


图 1.4 安装 JDK 的初始界面



图 1.5 JDK 安装界面

(2) JDK 安装完毕后,如果是第一次安装 JDK,会自动跳转到 JRE 安装界面,如图 1.6 所示。单击“更改”按钮,可更改安装 JRE 的目录。这里采用默认设置,完成设置后单击“下一步”按钮,继续安装。

(3) JRE 安装完成后,会弹出如图 1.7 所示的界面,单击“关闭”按钮,完成安装。



图 1.6 JRE 安装界面

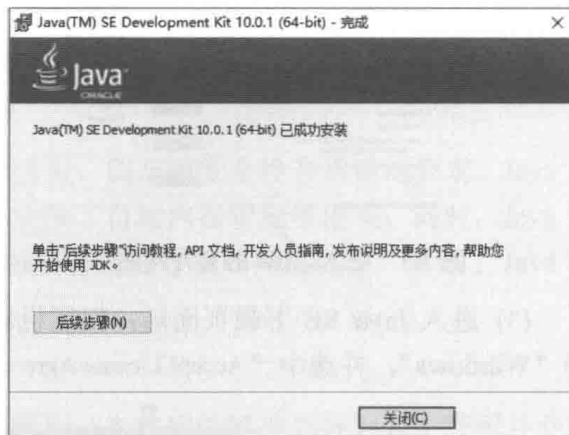


图 1.7 安装完成界面

1.2.3 设置 path 与 classpath

安装完 JDK 后,为了使系统能自动找到命令所在的目录,需设置环境变量。设置的环境变量主要包括 path 和 classpath 这两个环境变量,下面给出在 Windows 操作系统中设置环境变量的主要步骤。



【教学视频】

(1) 右击“计算机”图标,在弹出的快捷菜单中选择“属性”命令,弹出“系统属性”对话框,如图 1.8 所示。

(2) 在“系统属性”对话框中,选择“高级”选项卡,如图 1.9 所示。

(3) 单击“环境变量”按钮,弹出“环境变量”设置对话框,如图 1.10 所示。

(4) 在“系统变量”列表框中找到变量名“Path”,并选中,双击进入“编辑系统变量”对话框,如图 1.11 所示。

查看有关计算机的基本信息



图 1.8 “系统属性”对话框



图 1.9 “高级”选项卡



图 1.10 “环境变量”对话框

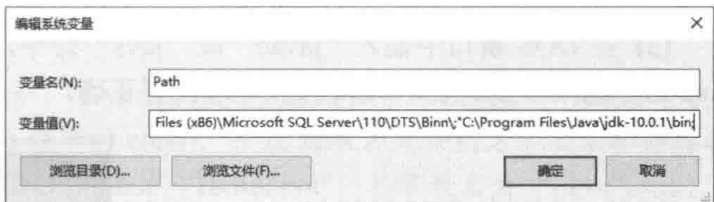


图 1.11 “编辑系统变量”对话框

(5) 在“变量值”文本框的最后面输入“;C:\Program Files\Java\jdk-10.0.1\bin”，注意有分号，主要是为了与前面原来的内容分隔开。完成编辑后，单击“确定”按钮。

(6) 在“环境变量”对话框中单击“新建”按钮，弹出“新建系统变量”对话框，如图 1.12 所示。在“变量名”文本框中输入“classpath”，在“变量值”文本框中输入“.;C:\Program Files\Java\jdk-10.0.1\lib\tools.jar;C:\Program Files\Java\jdk-10.0.1\lib\dt.jar”，如图 1.13 所示。完成编辑后，单击“确定”按钮，完成环境变量的配置。

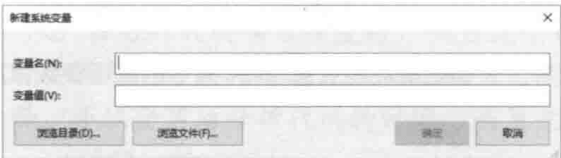


图 1.12 “新建系统变量”对话框

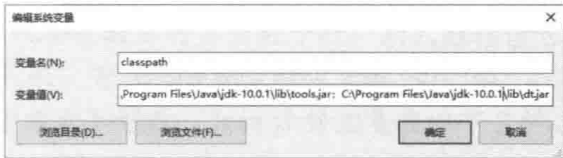


图 1.13 classpath 系统变量设置

说明：①设置 path 环境变量的目的是指向 JDK 的 bin 目录，在 bin 目录下放置了各种编译执行命令，通过该环境变量的设置，不管源文件在任何路径上，都可以通过该环境变

量直接找到相应的命令对源文件进行编译执行，否则，必须将源程序复制到 bin 目录下，方可进行编译执行；②设置 classpath 环境变量的目的是当需要导入已经定义好的类时，可以直接从 classpath 类路径中查找，“.”代表的是当前目录。

1.2.4 测试 Java 开发环境

配置完成后，需要测试配置是否正确，其具体步骤如下。

(1) 单击“开始”按钮，在弹出的“开始”菜单中再选择“运行”命令，弹出如图 1.14 所示的对话框。在“打开”文本框中输入“cmd”，单击“确定”按钮，弹出如图 1.15 所示的窗口。

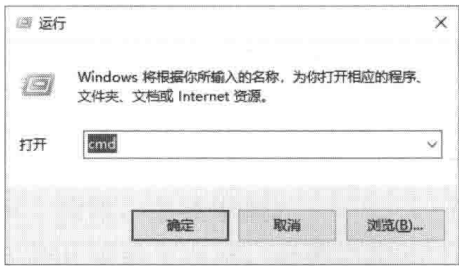


图 1.14 “运行”对话框

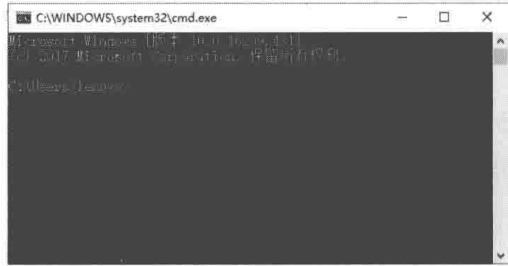


图 1.15 DOS 窗口

(2) 在 DOS 窗口中输入“java -version”命令，此命令用于显示 Java 的版本信息，若安装成功将会出现如图 1.16 所示的界面，否则就应该返回去检查安装过程是否有问题。

(3) 在 DOS 窗口中输入“javac”或“java”命令，如果出现如图 1.17 所示的 javac 或 java 命令选项参数，表示 Java 开发环境配置正确。



图 1.16 显示 Java 的版本信息

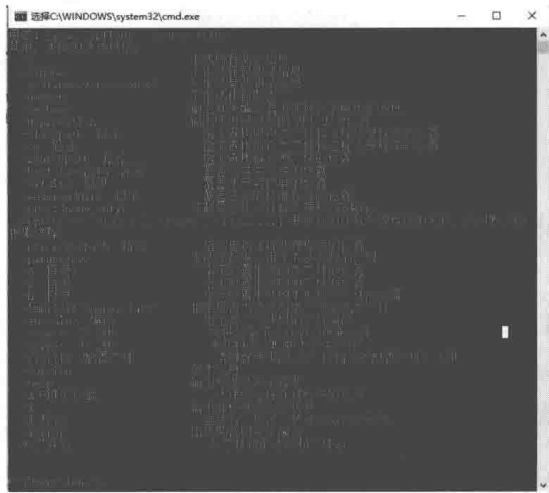


图 1.17 正确配置环境变量 javac 命令的选项参数信息

1.3 用命令行方式描述 Java 应用程序的开发过程

Java 的开发环境搭建好后，就可以编写 Java 程序了。根据结构组成和运行环境的差异，Java 程序共分为两类：Java 应用程序(Java Application)和 Java 小应用程序(Java Applet)。Java