



孙宇霞 编著
王 健

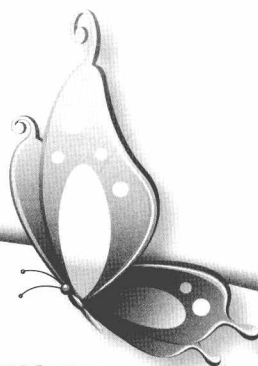
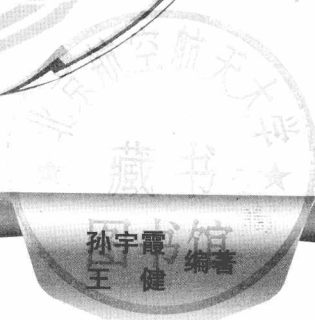
Java编程 从基础到应用

清华大学出版社



014036643

TP312JA
1573



Java编程 从基础到应用



北航

C1723452

清华大学出版社
北京

TP 312JA

1573

P

内 容 简 介

Java 是当今最流行的程序设计语言之一。本书以 Java 最新版本的 Java SE 7.0 为基础, 全面介绍 Java 语言的功能和特点, 主要包括 Java 语言基础知识、面向对象编程技术、异常处理、输入/输出流、集合框架、基于 Swing 的图形用户界面、数据库编程等内容。最后通过一个综合案例来介绍 Java 编程在实际开发中的应用。

本书内容丰富、语言简练易懂, 并辅以大量的案例, 即使初学者也可轻松地掌握通过 Java 进行面向对象编程的思维。本书可作为高等院校计算机或相关专业的 Java 语言教材, 也可作为 Java 编程爱好者的参考书。

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 编程从基础到应用 / 孙宇霞等编著. —北京: 清华大学出版社, 2014

从基础到应用

ISBN 978-7-302-31105-8

I. ①J… II. ①孙… III. ①JAVA 语言-程序设计-教材 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2012) 第 308683 号

责任编辑: 夏兆彦

封面设计: 胡文航

责任校对: 徐俊伟

责任印制: 宋 林

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

印 刷 者: 清华大学印刷厂

装 订 者: 三河市新茂装订有限公司

经 销: 全国新华书店

开 本: 185mm×260mm 印 张: 28.75 字 数: 721 千字

(附光盘一张)

版 次: 2014 年 3 月第 1 版

印 次: 2014 年 3 月第 1 次印刷

印 数: 1~4000

定 价: 59.00 元

FOREWORD

前言

Java 是当前最流行的程序设计语言之一，它的出现大大地促进了软件产业和互联网的发展。Java 之所以如此流行，是因为它是一种简单易学易用的、纯面向对象的、可移植的、安全的、高效的、健壮的、分布式的、多线程的、结构中立的、可解释执行的和动态的语言。本书以 Java SE 7.0 为基础，全面讲解 Java SE 的主要内容。

本书共 15 章，从计算机基础开始，系统地介绍 Java 编程的相关知识，重点在流程控制和数组的使用、面向对象技术，以及数据库编程技术和 GUI 编程技术的讲解，使读者可以从最基础的知识开始系统全面地学习 Java 编程，从一个 Java 初学者逐渐达到可以熟练使用 Java 程序语言进行程序设计的程序。同时，本书可以帮助读者建立逻辑思维的概念，培养解决问题的能力。

本书内容

第 1 章：Java 概述。主要讲解 Java 语言的发展历史、Java 语言的特点、Java 运行机制以及 Java 环境配置和 Java 程序的开发步骤。

第 2 章：简单数据类型及运算。主要讲解 Java 语言中的基本数据类型、变量、常量，以及运算符与表达式之间的整合使用。

第 3 章：流程控制。主要讲解 Java 语言中的条件语句、循环语句和跳转语句的应用。

第 4 章：数组类型。主要讲解一维数组的创建和使用，以及多维数组的创建和访问，并扩展了对数组元素的搜索、排序等操作。

第 5 章：字符串。主要讲解字符串的创建方式、常用方法以及对字符串的常用操作，并介绍 StringBuffer 类的创建和使用。

第 6 章：类与对象。主要介绍类与对象的创建和使用，并比较它们之间的异同点。

第 7 章：深入面向对象编程。主要介绍面向对象的封装、继承、多态三大特征的应用，并介绍抽象类与接口的区别及使用。

第 8 章：内部类与匿名类。主要介绍 Java 中内部类的分类及各个类型内部类的特点及应用，并简单介绍匿名类的应用。

第 9 章：异常处理。主要介绍异常类型，以及 Java 中捕获异常的多种方式，并详细介绍在 Java 中如何声明异常、抛出异常和自定义异常。

第 10 章：Java 常用工具类。主要介绍 Java 语言中常用的工具类，包括 Object 类、系统类 (System)、数学类 (Math)、包装类 (Integer、

Float、Double 和 Character 类)、日期操作类 (Date 类和 Calendar 类)、格式化日期类 (DateFormat 类和 SimpleDateFormat 类) 和 Random 类。

第 11 章: Java 的输入/输出流。主要介绍常用字节流对文件的读取/写入操作以及常用字符流对文件的操作, 并介绍使用 File 类中的方法来获取文件的相关信息, 比如: 文件名、文件大小、文件内容长度等。

第 12 章: 集合框架。主要介绍 Java 语言中的集合框架常用接口, 包括 Set 接口、List 接口和 Map 接口, 同时也介绍实现这 3 个接口的常用集合类, 包括 HashSet、TreeSet、ArrayList、LinkedList 和 HashMap。在本章的最后, 还将详细介绍集合辅助类 Collections, 使用该类中的方法可以对集合元素进行排序、替换和复制。

第 13 章: Swing 程序开发。主要介绍 Swing 中的窗口和面板容器以及基本组件的创建及使用, 并简单介绍 JTabbedPane 页签面板容器、JTable 表格组件、JTree 树和 JFileChooser 文件选择器的应用以及它们常用的事件处理监听器。在本章的最后, 将重点介绍 Swing 程序中的 6 种布局管理器和常用对话框的应用。

第 14 章: JDBC 数据库编程。主要介绍 JDBC 连接数据库的相关知识, 包括 JDBC 的工作原理、Java 程序连接数据库的基本步骤和 JDBC 的应用。

第 15 章: 进销存管理系统。本章使用 Java 语言中的 Swing 程序开发一个桌面程序——进销存管理系统, 该系统使用 JDBC 连接方式连接 MySQL 5.5 数据库, 从而能动态地管理库存信息、进货信息、销售信息、商品信息、供应商信息、客户信息等数据。本章通过一个综合案例介绍 Java 编程在实际开发中的应用。

本书特色

本书中大量内容来自真实的 Java 程序, 力求通过展示读者实际操作时的问题的方法使读者更容易地掌握 Java 应用开发。本书还引用了大量来自一线论坛的问题并进行讲解, 力求通过解决读者提出的疑难问题给出正确的答案。本书难度适中、内容由浅入深、实用性强、覆盖面广、条理清晰。

- ❑ **结构独特** 通过“概念、语法描述、示例描述、示例应用、示例分析、运行结果”的模式将每个知识与实际应用中的问题相结合。
- ❑ **形式新颖** 用准确的语言总结概念、用直观的图示演示过程、用详细的注释解释代码、用形象的比方帮助记忆。
- ❑ **内容丰富** 涵盖了实际开发中使用 Java 语言对文件进行操作、连接数据库、Swing 技术等方面的热点问题。本书每章介绍完基本知识后, 都会给出 1~2 个综合实例以综合应用本章知识, 这些综合实例均采用一线实战技术, 为读者以后的实际创作提供思路; 每章在章尾给出一些综合问题的处理方法, 从而将本章的主要知识点在实际应用中容易出现的一些常见问题给予解决, 能够帮助读者快速提高。
- ❑ **随书光盘** 本书为实例配备了视频教学文件, 读者可以通过视频文件更加直观地学习 Java 的使用知识。
- ❑ **网站技术支持** 读者在学习或者工作的过程中, 如果遇到实际问题, 可以直接登

录 www.itzcn.com 与我们联系，作者会在第一时间给予帮助。

- 贴心的提示 为了便于读者阅读，全书还穿插着一些技巧、提示等小贴士，体例约定如下。

提示：通常是一些贴心的提醒，让读者加深印象或提供建议，或者是解决问题的方法。

注意：提出学习过程中需要特别注意的一些知识点和内容，或者相关信息。

技巧：通过简短的文字，指出知识点在应用时的一些小窍门。

读者对象

本书具有知识全面、实例精彩、指导性强的特点，力求以全面的知识性及丰富的实例来指导读者透彻地学习 Java 基础知识。本书可以作为 Java 的入门书籍，也可以帮助中级读者提高技能。

本书适合以下人员阅读学习。

- 没有编程基础的 Java 入门人员。
- 有一些编程基础，并且希望全面学习 Java 编程的读者。
- 各大中专院校的在校学生和相关授课老师。
- 相关社会培训班的学员。

除了封面署名人员之外，参与本书编写的人员还有马海军、李海庆、陶丽、王咏梅、康显丽、郝军启、朱俊成、宋强、孙洪叶、袁江涛、张东平、吴鹏、王新伟、刘青凤、汤莉、冀明、王超英、王丹花、闫琰、张丽莉、李卫平、王慧、牛红惠、丁国庆、黄锦刚、李旎、王中行、李志国等。在编写过程中难免会有漏洞，欢迎读者通过我们的网站 www.itzcn.com 与我们联系，帮助我们改正提高。

CONTENTS

目录

第 1 章 Java 概述	1
1.1 Java 语言简介	1
1.1.1 Java 的发展历史	1
1.1.2 Java 语言的特点	2
1.1.3 Java 运行机制	3
1.2 Java 程序运行环境	4
1.2.1 JDK 介绍	4
1.2.2 下载和安装 JDK	5
1.2.3 配置 JDK	6
1.3 Java 的开发工具	7
1.3.1 MyEclipse 概述	7
1.3.2 MyEclipse 的配置	8
1.3.3 使用 MyEclipse 开发应用程序	9
1.4 Java Application 和 Java Applet 介绍	11
1.4.1 Application 介绍	11
1.4.2 Applet 介绍	13
1.5 包	18
1.5.1 创建包	18
1.5.2 使用包	19
1.6 Java 程序的基本要素	19
1.7 编码规范	22
1.8 Java 程序调试技巧	24
1.9 项目案例：创建 HelloWorld 程序	26
1.10 习题	28
1.11 实践疑难解答	29
1.11.1 在类中找不到主方法错误	29
1.11.2 找不到或无法加载主类	30
第 2 章 简单数据类型及运算	32
2.1 Java 基本数据类型	32
2.1.1 整数类型	33
2.1.2 浮点型	34
2.1.3 字符型	34
2.1.4 布尔类型	35
2.1.5 数据类型间的转换	35

2.2 变量与常量	36	3.6 实践疑难解答	80
2.2.1 变量	37	3.6.1 关于 for 循环语句问题	80
2.2.2 变量的作用域	37	3.6.2 关于 Java 的 switch 语句的一个问题	81
2.2.3 常量	37		
2.3 运算符与表达式	38	第 4 章 数组类型	83
2.3.1 算术运算符及表达式	38	4.1 一维数组	83
2.3.2 赋值运算符及表达式	39	4.1.1 创建一维数组	83
2.3.3 自增和自减运算符	40	4.1.2 初始化一维数组	84
2.3.4 关系运算符及表达式	41	4.1.3 一维数组的应用	84
2.3.5 逻辑运算符及表达式	41	4.2 多维数组	85
2.3.6 位运算符及表达式	42	4.2.1 创建多维数组	86
2.3.7 条件运算符及表达式	44	4.2.2 初始化多维数组	86
2.3.8 运算符的优先级	45	4.2.3 访问多维数组中的元素	87
2.4 项目案例：打印超市购物清单	46	4.3 复制数组	89
2.5 习题	48	4.3.1 使用 copyOf()方法和 copyOfRange()方法	89
2.6 实践疑难解答	50	4.3.2 使用 arraycopy()方法	91
2.6.1 Java 数据类型的问题	50	4.3.3 使用 clone()方法	92
2.6.2 Java 数据类型的出错问题	51	4.4 数组排序	92
第 3 章 流程控制	52	4.4.1 选择排序	93
3.1 条件语句	52	4.4.2 冒泡排序	94
3.1.1 简单的 if 条件语句	52	4.4.3 快速排序	95
3.1.2 if-else 条件语句	54	4.5 搜索数组	95
3.1.3 if-else if-else 条件语句	55	4.6 项目案例：实现学生成绩查询功能	97
3.1.4 嵌套 if 语句	57	4.7 习题	99
3.1.5 switch 多分支语句	59	4.8 实践疑难解答	101
3.1.6 包含 break 的 switch 语句	62	4.8.1 能不能向数组中添加更多的元素	101
3.2 循环语句	64	4.8.2 遍历数组时提示 ArrayIndexOutOfBoundsException	101
3.2.1 for 循环语句	64	4.8.3 获取二维数组的行数与列数	102
3.2.2 while 循环语句	68	4.8.4 对二维数组进行排序	102
3.2.3 do-while 循环语句	69		
3.2.4 while 循环与 do-while 循环的异同	71	第 5 章 字符串	104
3.3 跳转语句	71	5.1 类 String 字符串	104
3.3.1 break 语句	71	5.1.1 直接使用字符串常量创建字符串	104
3.3.2 continue 语句	73	5.1.2 使用 new 关键字创建字符串	105
3.3.3 return 语句	75	5.2 字符串的常用方法	106
3.4 项目案例：打印出所有三位数的“水仙花数”	77		
3.5 习题	78		

5.3 字符串的基本操作.....	106	6.4 常用修饰符.....	151
5.3.1 获取字符串的长度.....	107	6.4.1 static 静态修饰符.....	151
5.3.2 转换字符串的大小写.....	107	6.4.2 final 修饰符.....	154
5.3.3 去掉字符串的首尾空格.....	107	6.5 项目案例：完成消息的发送功能.....	157
5.3.4 比较字符串的大小.....	107	6.6 习题.....	160
5.3.5 字符串查找.....	108	6.7 实践疑难解答.....	162
5.3.6 字符串分割.....	111	6.7.1 Java 类和对对象的问题.....	162
5.3.7 字符串截取.....	112	6.7.2 类与对象的区别.....	163
5.3.8 字符串替换.....	112		
5.3.9 字符串连接.....	113		
5.4 类 StringBuffer 字符串.....	115	第 7 章 深入面向对象编程.....	165
5.4.1 StringBuffer 类的构造方法.....	115	7.1 类的封装.....	165
5.4.2 StringBuffer 类的常用方法.....	115	7.2 类的继承.....	167
5.5 正则表达式.....	119	7.2.1 简单继承.....	167
5.6 项目案例：根据用户输入数值的个数		7.2.2 单继承.....	170
打印不同的信息.....	121	7.2.3 使用 super 关键字.....	171
5.7 习题.....	123	7.2.4 对象类型转换.....	173
5.8 实践疑难解答.....	126	7.2.5 强制对象类型转换.....	174
5.8.1 如何处理特殊的分割符.....	126	7.3 方法重载.....	175
5.8.2 可不可以分割字符串时使用多个		7.4 方法重写.....	176
分割符.....	127	7.5 抽象类与接口.....	178
5.8.3 使用 equals()方法比较两个相同的		7.5.1 抽象类.....	178
StringBuffer 对象.....	128	7.5.2 接口.....	180
		7.6 项目案例：实现购物清单的打印.....	185
		7.7 习题.....	187
		7.8 实践疑难解答.....	189
		7.8.1 Java 中的继承问题.....	189
		7.8.2 Java 继承的运行顺序问题.....	190
		7.8.3 Java 多态问题.....	191
第 6 章 类与对象.....	129		
6.1 面向对象概述.....	129	第 8 章 内部类与匿名类.....	193
6.1.1 对象概述.....	129	8.1 内部类概述.....	193
6.1.2 面向对象的核心特性.....	130	8.2 内部类.....	194
6.2 类结构.....	133	8.2.1 实例内部类.....	195
6.2.1 类的定义.....	133	8.2.2 静态内部类.....	197
6.2.2 成员变量的定义.....	135	8.2.3 局部内部类.....	198
6.2.3 成员方法的定义.....	136	8.3 匿名类.....	199
6.2.4 构造方法.....	139	8.4 习题.....	201
6.2.5 main()方法.....	142	8.5 实践疑难解答.....	202
6.2.6 方法的可变参数.....	144	8.5.1 外部类可以称为内部类的父	
6.2.7 this 关键字.....	145	类吗.....	202
6.3 对象.....	146		
6.3.1 对象的创建.....	146		
6.3.2 访问对象的属性和行为.....	148		
6.3.3 对象的销毁.....	150		

8.5.2 局部内部类所在的方法返回类 型问题·····	203	10.4.1 基本数据类型对应的包装类···	240
8.5.3 局部内部类只能访问所在方法中 的 final 参数与变量·····	203	10.4.2 Integer 类·····	241
8.5.4 Outer\$1Inner.class 这种文件名怎 么理解·····	204	10.4.3 Float 类·····	243
第 9 章 异常处理 ·····	205	10.4.4 Double 类·····	244
9.1 异常的基本概述·····	205	10.4.5 Character 类·····	245
9.2 异常类型·····	206	10.5 日期操作类·····	249
9.3 异常处理·····	208	10.5.1 java.util.Date 类·····	249
9.3.1 使用 try-catch 语句捕获异常···	208	10.5.2 java.util.Calendar 类·····	251
9.3.2 使用多重 catch 语句捕获异常··	210	10.6 格式化日期类·····	255
9.3.3 使用 try-catch-finally 语句捕获 异常·····	211	10.6.1 DateFormat 类·····	255
9.4 声明异常和抛出异常·····	214	10.6.2 SimpleDateFormat 类·····	257
9.4.1 throws 声明异常·····	214	10.7 Random 类·····	258
9.4.2 throw 抛出异常·····	215	10.8 项目案例：计算上完课程后的日期··	260
9.5 自定义异常·····	217	10.9 习题·····	262
9.6 项目案例：模拟 ATM 取款业务办理··	219	10.10 实践疑难解答·····	264
9.7 习题·····	221	10.10.1 继承 Object 类的问题·····	264
9.8 实践疑难解答·····	226	10.10.2 Java 中 Random 问题·····	265
9.8.1 try-catch 能嵌套使用吗·····	226	10.10.3 Integer.parseInt()和 Integer.valueOf()区别·····	266
9.8.2 捕获异常时编译不通过·····	226	第 11 章 Java 的输入/输出流 ·····	268
9.8.3 try-finally 执行结果·····	227	11.1 输入/输出流概述·····	268
9.8.4 try-finally 语句中包含 return 语句·····	228	11.1.1 字节流·····	268
第 10 章 Java 常用工具类 ·····	230	11.1.2 字符流·····	269
10.1 Object 类·····	230	11.1.3 预定义流·····	270
10.2 系统类 System·····	233	11.2 字节流·····	272
10.2.1 System 类中的成员变量·····	233	11.2.1 字节输入流 InputStream·····	272
10.2.2 System 类中的成员方法·····	234	11.2.2 字节输出流 OutputStream·····	273
10.3 数学类 Math·····	237	11.2.3 字节数组输入流 ByteArray- InputStream·····	274
10.3.1 三角函数方法·····	237	11.2.4 字节数组输出流 ByteArray- OutputStream·····	275
10.3.2 指数函数方法·····	238	11.2.5 文件字节输入流 FileInputStream·····	276
10.3.3 最大值、最小值和绝对值函数 方法·····	239	11.2.6 文件输出流 FileOutputStream···	278
10.3.4 取整函数方法·····	239	11.3 字符流·····	280
10.4 包装类·····	240	11.3.1 字符输入流 Reader·····	280
		11.3.2 字符输出流 Writer·····	281
		11.3.3 字符文件输入流 FileReader·····	282
		11.3.4 字符文件输出流 FileWriter·····	283

11.3.5 字符缓冲区输入流 BufferedReader	285	13.2 Swing 容器和组件的层次	324
11.3.6 字符缓冲区输出流 BufferedWriter	286	13.2.1 JComponent 类层次结构图	324
11.4 File 类	287	13.2.2 Swing 包	325
11.5 RandomAccessFile 类	289	13.2.3 组件的分类	325
11.6 项目案例：存储商品信息并读取	291	13.3 窗口和面板	327
11.7 习题	294	13.3.1 JFrame 窗口	327
11.8 实践疑难解答	295	13.3.2 JPanel 面板	329
11.8.1 读取文件内容时出现中文乱 码问题	295	13.4 Swing 基本组件	330
11.8.2 使用 System.in 接收汉字时，拼音 也会接收	296	13.4.1 按钮组件——JButton	330
第 12 章 集合框架	297	13.4.2 标签组件——JLabel	331
12.1 集合简介	297	13.4.3 单行文本框组件—— JTextField	333
12.1.1 集合概述	297	13.4.4 多行文本域组件—— JTextArea	334
12.1.2 Collection 接口	299	13.4.5 复选框组件——JCheckBox	335
12.2 无序列表——Set 集合	299	13.4.6 单选按钮组件—— JRadioButton	337
12.2.1 HashSet 类	299	13.4.7 下拉列表组件—— JComboBox	338
12.2.2 TreeSet 类	301	13.5 JTabbedPane 页签面板	342
12.3 有序列表——List 集合	304	13.6 JTable 表格	344
12.3.1 ArrayList 类	304	13.7 JTree 树	346
12.3.2 LinkedList 类	305	13.8 JFileChooser 文件选择器	350
12.4 映射——Map	307	13.9 布局管理器	352
12.5 集合辅助类 Collections	309	13.9.1 BorderLayout 管理器	352
12.5.1 集合排序	309	13.9.2 FlowLayout 管理器	354
12.5.2 集合反序	310	13.9.3 CardLayout 管理器	355
12.5.3 替换所有的元素	311	13.9.4 GridLayout 管理器	356
12.5.4 复制集合	312	13.9.5 GridBagLayout 管理器	358
12.6 集合的高级应用——泛型	314	13.9.6 BoxLayout 管理器和 Box 容器	360
12.7 项目案例：部门管理系统	315	13.10 对话框	362
12.8 习题	319	13.10.1 消息对话框	362
12.9 实践疑难解答	320	13.10.2 确认对话框	363
12.9.1 List 集合的 remove()方法使用 问题	320	13.10.3 输入对话框	364
12.9.2 Set 集合里面怎么可以添加重 复元素	321	13.10.4 选项对话框	365
第 13 章 Swing 程序开发	323	13.11 项目案例：制作计算器	366
13.1 图形对象概述	323	13.12 习题	369

13.13 实践疑难解答	371
13.13.1 JButton 按钮如何响应回车事件	371
13.13.2 单击对话框中的【否】按钮实现退出程序	371

第 14 章 JDBC 数据库编程 372

14.1 JDBC 简介	372
14.1.1 JDBC 的工作机制	372
14.1.2 JDBC API 结构	373
14.1.3 JDBC 驱动	373
14.2 JDBC 连接数据库的基本步骤	376
14.3 JDBC 主要的接口和类	378
14.3.1 Connection 接口	378
14.3.2 DriverManager 类	378
14.3.3 Statement 接口	379
14.3.4 PreparedStatement 接口	382
14.3.5 ResultSet 接口	384
14.4 项目案例：学生管理系统	387
14.5 习题	394
14.6 实践疑难解答	395
14.6.1 中文乱码问题	395
14.6.2 如何从 ResultSet 对象中获取对应属性	395

第 15 章 进销存管理系统 397

15.1 系统分析与设计	397
15.1.1 系统功能结构	397
15.1.2 系统流程图	397
15.2 数据库设计	399
15.3 主窗体设计	402
15.3.1 创建主窗体	402
15.3.2 创建导航面板	404
15.4 公共模块设计	407
15.4.1 编写 Item 类	407
15.4.2 编写 Dao 公共类	408
15.5 基础信息模块设计	413
15.5.1 添加客户	413
15.5.2 修改与删除客户	417
15.6 进货管理模块设计	422
15.6.1 进货单	422
15.6.2 进货退货	428
15.7 客户查询模块设计	432
15.8 库存管理——价格调整	436
15.9 系统管理模块设计	440
15.9.1 更新密码	440
15.9.2 权限管理	445

第1章

Java 概述

Java 语言是一种简单的、面向对象的开发语言，它具有分布式、安全性，又可以实现多线程，更主要的是它与平台无关，解决了困扰软件界多年的软件移植问题。同时，Java 也是一种跨平台的语言，在当前的语言开发中是非常流行的，其广泛应用于信息技术、科学研究、军事工业、航空航天等各个领域。

本章将对 Java 语言的发展、特点、运行环境以及开发工具进行简单的介绍。

本章学习要点：

- 了解 Java 语言的特点
- 了解 Java 语言的发展史
- 掌握 Java 语言的工作原理
- 熟练掌握 JDK 安装和配置
- 熟练掌握开发 Java Application
- 掌握使用记事本和 MyEclipse 开发 Java 程序

1.1 Java 语言简介

Java 是一种编程语言，设计用于互联网的分布式环境。Java 语言的语法与 C 语言和 C++ 语言很接近，使得大多数程序员很容易学习和使用 Java。另一方面，Java 丢弃了 C++ 中很少使用的、很难理解的、令人迷惑的那些特性。最重要的是，Java 语言不使用指针，并提供了自动的垃圾回收机制。

1.1.1 Java 的发展历史

1991 年，Sun 公司为了进军家用电子消费市场，成立了一个代号为 Green 的项目组。Oak（橡树）系统出现。Oak 以 C++ 语言为基本，吸收了 C++ 中符合面向对象程序设计要求的部分，同时加入了一些满足网络设计要求的部分。

1994 年，Green 项目组成员认真分析计算机网络应用的特点，认为 Oak 满足网络应用所要求的平台独立性、系统可靠性和安全性等，并用 Oak 设计了一个称为 WebRunner（后来称为 HotJava）的 WWW 浏览器。

1995 年 5 月 23 日，Sun 公司正式发布了 Java 和 HotJava 两项产品。Java 语言一经推出，就受到了业界的关注。Netscape 公司第一个认可 Java 语言，并于 1995 年 8 月将 Java

解释器集成到它的主打产品 Navigator 浏览器中。接着, Microsoft 公司在 Internet Explorer 浏览器中认可了 Java 语言。Java 语言开始了自己的发展历程。

目前使用的 Java 版本是 Java 2, 它包括 J2SE、J2EE、J2ME 这 3 个版本, 分别用于不同的领域。

- J2SE (Java Standard Edition) ——用于工作站、PC, 为桌面开发和低端商务应用提供了 Java 标准平台。
- J2EE (Java Enterprise Edition) ——用于服务器, 构建可扩展的企业级 Java 平台。
- J2ME (Java Micro Edition) ——嵌入式 Java 消费电子平台, 适用于消费性电子产品和嵌入式设备。

1.1.2 Java 语言的特点

Java 是一种跨平台、适合于分布式计算环境的面向对象编程语言。具体来说, 它具有如下特性: 简单性、面向对象、分布式、解释型、可靠、安全、平台无关、可移植、高性能、多线程、动态性等。以下简单介绍其重要特性。

□ 结构简单

Java 语言的程序构成与 C 语言和 C++ 语言类似, 但是 Java 语言摒弃了 C 语言和 C++ 语言的复杂、不安全特性。例如: 指针的操作和内存的管理。此外, Java 语言提供了种类丰富、功能强大的类库, 提高了编程效率。

□ 面向对象

在现实世界中, 任何实体都可以看作是一个对象, 对象具有状态和行为两大特征。在 Java 语言中, 没有采用传统的、以过程为中心的编程方法, 而是采用以对象为中心, 通过对象之间的调用来解决问题的编程方法。

□ 平台无关

使用 Java 语言编写的应用程序不需要进行任何修改, 就可以在不同的软、硬件平台上运行。这主要是通过 Java 虚拟机 (JVM) 来实现的。

□ 可靠性

因为 Java 最初的设计目的是应用于电子类家庭消费产品, 所以要求较高的可靠性。例如: 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。

□ 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。如果没有安全保证, 用户运行从网络下载的 Java 语言应用程序是十分危险的。Java 语言通过使用编译器和注译器, 在很大程度上避免了病毒程序的产生和网络程序对本地系统的破坏。

□ 多线程

多线程是指在一个程序中可以同时执行多个简单任务。线程也被称为轻量进程, 是一个传统大进程里分出来的独立的可并发执行的单位。C 语言和 C++ 语言采用单线程体系结构, 而 Java 语言支持多线程技术。

Java 是一种比 C/C++ 更具动态特性的语言。在设计上强调为运行中的运算环境提供支持。Java 是在运行时为模块与模块之间建立连接, 并能够更直接地运用面向对象设计体系。

程序库可以自由地增加新方法和实例变量，而不会对它们的客户产生任何影响。

1.1.3 Java 运行机制

Java 程序的运行必须经过编写、编译、运行 3 个步骤：

- ❑ 编写是指在 Java 开发环境中进行程序代码的输入，最终形成后缀名为.java 的 Java 源文件。
- ❑ 编译是指使用 Java 编译器对源文件进行错误排查的过程，编译后将生成后缀名为.class 的字节码文件，这不像 C 语言那样最终生成可执行文件。
- ❑ 运行是指使用 Java 解释器将字节码文件翻译成机器代码，执行并显示结果。

Java 程序运行流程如图 1-1 所示。



图 1-1 Java 程序运行流程

字节码文件是一种和任何具体机器环境及操作系统环境无关的中间代码，它是一种二进制文件，是 Java 源文件由 Java 编译器编译后生成的目标代码文件。编程人员和计算机都无法直接读懂字节码文件，它必须由专用的 Java 解释器来解释执行，因此 Java 是一种在编译基础上进行解释运行的语言。

Java 解释器负责将字节码文件翻译成具体硬件环境和操作系统平台下的机器代码，以便执行。因此 Java 程序不能直接运行在现有的操作系统平台上，它必须运行在被称为 Java 虚拟机的软件平台之上。

Java 虚拟机 (JVM) 是运行 Java 程序的软件环境，Java 解释器就是 Java 虚拟机的一部分。在运行 Java 程序时，首先会启动 JVM，然后由它来负责解释执行 Java 的字节码，并且 Java 字节码只能运行于 JVM 之上。这样利用 JVM 就可以把 Java 字节码程序和具体的硬件平台以及操作系统环境分隔开来，只要在不同的计算机上安装了针对于特定具体平台的 JVM，Java 程序就可以运行，而不用考虑当前具体的硬件平台及操作系统环境，也不用考虑字节码文件是在何种平台上生成的。JVM 把这种不同软硬件平台的具体差别隐藏起来，从而实现了真正的二进制代码级的跨平台移植。JVM 是 Java 平台无关的基础，Java 的跨平台特性正是通过在 JVM 中运行 Java 程序实现的。Java 的这种运行机制可以通过图 1-2 来说明。

Java 语言这种“一次编写，到处运行 (write once, run anywhere)”的方式，有效地解决了目前大多数高级程序设计语言需要针对不同系统来编译产生不同机器代码的问题，即硬件环境和操作平台的异构问题，大大降低了程序开发、维护和管理开销。

注

Java 程序通过 JVM 可以达到跨平台特性，但 JVM 是不跨平台的。也就是说，不同操作系统之上的 JVM 是不同的，Windows 平台之上的 JVM 不能用在 Linux 上面，反之亦然。

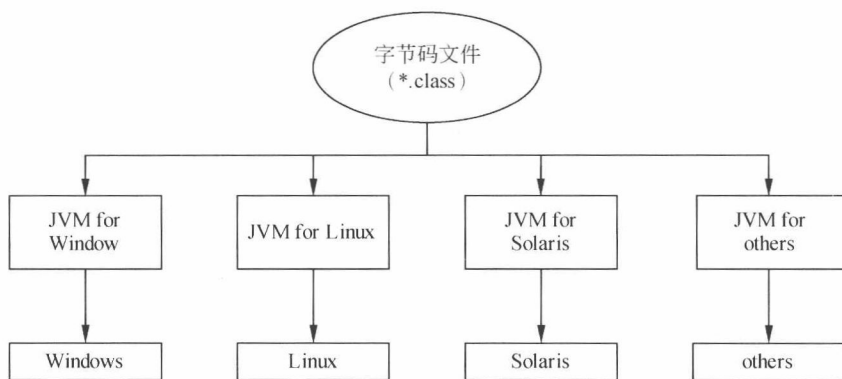


图 1-2 JVM 工作方式

1.2 Java 程序运行环境

要运行 Java 程序，只需要下载一个 JDK 开发工具包就可以，因为 JDK 开发工具包中包含了 Java 运行环境（Java Runtime Environment）、Java 工具和 Java 基础类库。JDK 开发工具包是整个 Java 的核心，无论哪种 Java 应用程序，实际上都是需要 JDK 开发工具包的。本节将介绍 JDK 是如何工作的，以及它的下载、安装和配置。

1.2.1 JDK 介绍

JDK（Java Development Kit）是一种用于构建在 Java 平台上发布的应用程序、Applet 和组件的开发环境，即编写 Java 程序必须使用 JDK，它提供了编译 Java 和运行 Java 程序的环境。

JDK 是一切 Java 应用程序的基础，所有的 Java 应用程序是构建在 JDK 之上的。JDK 中还包括完整的 JRE（Java Runtime Environment，Java 运行环境），包括了用于产品环境的各种库类，以及给开发者使用的补充库，如国际化的库、IDL 库。JDK 中还包括各种例子程序，用以展示 Java API（在后面章节将会讲解）中的各部分。

从初学者角度来看，采用 JDK 开发 Java 程序能够快速理解程序中各部分代码之间的关系，从而有利于理解 Java 面向对象的设计思想。JDK 的另一个显著特点是随着 Java（J2EE、J2SE 以及 J2ME）版本的升级而升级；但它的缺点也非常明显，从事大规模企业级 Java 应用开发非常困难，不能进行复杂的 Java 软件开发，也不利于团体协同开发。

Java 开发工具包（JDK）是 Java 程序员最初使用的开发环境。JDK 由一个标准类库和一组测试及建立文档的 Java 实用程序组成。其核心 Java API 是一些预定义类库，开发人员需要用这些类来访问 Java 语言的功能。Java API 包括一些重要的语言结构以及基本图形，网络和文件 I/O。一般来说，Java API 的非 I/O 部分对于运行 Java 的所有平台是相同的，而 I/O 部分则仅在通用 Java 环境中实现。作为 JDK 实用程序，工具库中有 7 种主要的可执

行程序，如下所示。

- ❑ **javac** Java 编译器，将 Java 源代码转换成字节码。
- ❑ **java** Java 解释器，直接从类文件执行 Java 应用程序字节代码。
- ❑ **appletviewer** 小程序浏览器，一种执行 HTML 文件上的 Java 小程序的 Java 浏览器。
- ❑ **javadoc** 根据 Java 源码及说明语句生成 HTML 文档。
- ❑ **jdb** Java 调试器，可以逐行执行程序，设置断点和检查变量。
- ❑ **javah** 产生可以调用 Java 过程的 C 过程，或建立能被 Java 程序调用的 C 过程的头文件。
- ❑ **javap** Java 反汇编器，显示编译类文件中的可访问功能和数据，同时显示字节代码含义。

1.2.2 下载和安装 JDK

JDK 开发工具包是整个 Java 的核心，无论 Java 应用服务器是哪种，实际上都是内置了某个版本的 JDK，因此 Java 程序的运行时离不开 JDK。本节将介绍 JDK 的下载和安装。

1. 获取 JDK 开发工具包

首先从 JDK 7 的官方网站 <http://jdk7.java.net/> 上下载最新版本 JDK，下载页面如图 1-3 所示。

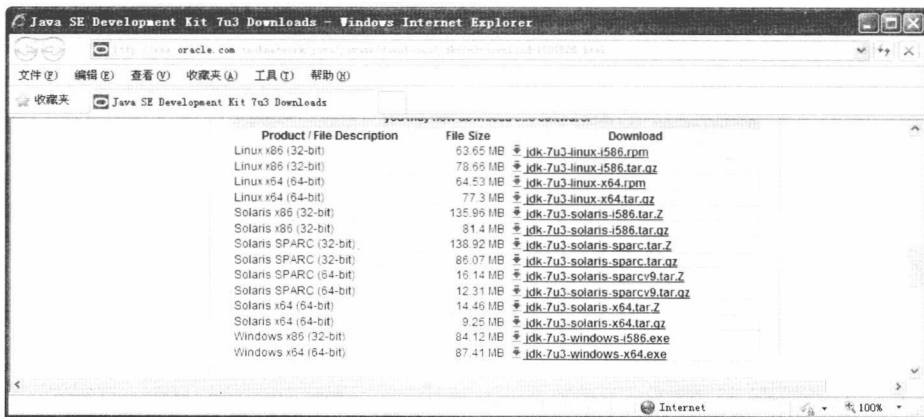


图 1-3 JDK 下载页面

从图 1-3 的表格中，可以看到不同平台需要安装的 JDK 是不一样的，这里需要下载的是 Windows x86(32-bit)平台下的 JDK，即点击 [jdk-7u3-windows-i586.exe](#)（该内容是一个超链接）进行下载。

2. 安装 JDK

当下载完成后，在电脑磁盘中会发现一个名称为 `jdk-7u3-windows-i586.exe` 的可执行文