

高等学校计算机类专业  
“十三五”规划教材

# Java 程序设计教程

Courses of Programming in Java

向守超 刘 军 张永志 编著

高等学校计算机类专业“十三五”规划教材

# Java 程序设计教程

向守超 刘 军 张永志 编著

西安电子科技大学出版社

## 内 容 简 介

本书深入浅出讲述了 Java 面向对象程序设计的基础知识, 并对 Java 的高级应用进行深入分析和讲解。内容涵盖 Java 概述, 数据类型与运算符, 流程控制结构, 数组, 类和对象, Java 常用类, 继承与多态, 抽象类、接口和枚举, 异常, 泛型与集合, 输入/输出流, 多线程, Swing UI 设计, JDBC 与 MySQL 编程, 网络编程。本书所有代码都是基于 Java 8 环境调试运行的。

本书注重可读性和实用性, 并且重点突出, 强调动手操作能力。书中配备了大量的例题和习题, 这些例题和习题既能帮助理解知识, 又具有启发性。本书通俗易懂, 便于自学, 针对较难理解的问题, 都是从简单到复杂, 逐步深入地引入例子, 便于读者掌握 Java 面向对象编程的思想。

本书适用面广, 既可作为高校、培训机构的 Java 教材, 也可作为计算机科学与技术、物联网工程、软件工程、电子商务等专业的程序设计课程的教材, 也可供自学者及软件开发人员参考。

## 图书在版编目(CIP)数据

Java 程序设计教程 / 向守超, 刘军, 张永志编著. —西安: 西安电子科技大学出版社, 2019.7  
ISBN 978-7-5606-5326-6

I. ① J… II. ① 向… ② 刘… ③ 张… III. ① JAVA 语言—程序设计—教材 IV. ① TP312.8

中国版本图书馆 CIP 数据核字(2019)第 086001 号

策划编辑 刘玉芳

责任编辑 万晶晶

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)88242885 88201467 邮 编 710071

网 址 www.xduph.com 电子邮箱 xdupfb001@163.com

经 销 新华书店

印刷单位 陕西天意印务有限责任公司

版 次 2019 年 7 月第 1 版 2019 年 7 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印 张 24.25

字 数 578 千字

印 数 1~3000 册

定 价 56.00 元

ISBN 978-7-5606-5326-6 / TP

**XDUP 5628001-1**

\*\*\*如有印装问题可调换\*\*\*

# 前 言

本书基于 Java 8 新特性而编写，具有很强的可读性和实用性，并根据知识点的难易程度，精心选择了许多例题和习题，帮助读者理解知识，启发学习。

全书共分为 15 章，分别为 Java 概述，数据类型与运算符，流程控制结构，数组，类和对象，Java 常用类，继承与多态，抽象类、接口和枚举，异常，泛型与集合，输入/输出流，多线程，Swing UI 设计，JDBC 与 MySQL 编程，网络编程。

第 1 章介绍 Java 语言的来历、发展史、特点和 Java 体系架构，详细讲解了 Java 集成开发环境的安装和配置使用，以及 Java 应用程序的基本结构和基本编写规范。第 2 章讲解了 Java 的基本数据类型、各种运算符和表达式。第 3 章详细介绍了 Java 流程控制结构以及转移语句。第 4 章介绍了数组的创建、初始化、冒泡排序、foreach 遍历、二维数组以及 Arrays 类的常用方法。第 5 章作为面向对象的开始，重点讲述了类的声明、对象的创建和使用、成员方法和构造方法的基本结构特点和应用规则、方法的重载、包的应用以及访问权限修饰符等关键字的使用原则。第 6 章讲述了常用的实用类，包括 Object 类、字符串类、Scanner 类、Math 类、Date 类、Calendar 类和一些格式化处理规则。第 7 章讲述了类的继承、多态、内部类以及类之间的其他各种关系。第 8 章讲述了抽象类、接口和枚举的应用。第 9 章讲述了 Java 的异常处理机制，包括捕获异常、抛出异常和自定义异常。第 10 章讲述了泛型和集合，包括泛型的定义、通配符、泛型的限制、集合框架、迭代器接口以及几种常用集合类的使用方法。第 11 章重点讲述了输入/输出流，包括文件类、字节流、字符流、过滤流、转换流和对象流，以及 NIO 的应用。第 12 章讲述了多线程的创建、启动、实现、优先级、生命周期以及实现多线程的同步和网络通信等内容。第 13 章介绍了 Java 的 Swing UI 组件的设计，包括基本组件、高级组件、容器和布局，并把事件处理作为难点讲解。第 14 章讲述了 JDBC 与 MySQL 数据库的编程，重点是数据库的访问和操作数据库。第 15 章讲述了关于 Java 的网络编程的基础知识，针对套接字，用通俗易懂的语言给予了详细的讲解，使读者能够轻松接受网络编程知识，并通过聊天室案例实现网络编程，有利于读者举一反三编写相应的网络程序。

本书由重庆机电职业技术大学向守超老师负责第 1 章到第 7 章的编写，重庆电信职业学院刘军老师负责第 8 章到第 12 章的编写，重庆机电职业技术大学张永志老师负责第 13 章到第 15 章的编写。由于编者水平有限且时间仓促，书中难免存在疏漏和不足，希望同行专家和读者给予批评和指正。

编 者  
2019 年 1 月

# 目 录

第 1 章 Java 概述.....	1	2.3.3 数据类型转换.....	29
1.1 Java 简介.....	1	2.4 运算符.....	30
1.1.1 Java 起源.....	1	2.4.1 自增、自减运算符.....	31
1.1.2 Java 发展史.....	2	2.4.2 算术运算符.....	32
1.1.3 Java 的特点.....	2	2.4.3 关系运算符.....	34
1.2 Java 体系.....	4	2.4.4 逻辑运算符.....	35
1.2.1 Java 应用平台.....	4	2.4.5 位运算符.....	37
1.2.2 Java 专有名词.....	5	2.4.6 赋值运算符.....	38
1.2.3 Java 跨平台原理.....	5	2.4.7 条件运算符.....	40
1.3 JDK 工具.....	7	2.4.8 运算符优先级.....	40
1.3.1 JDK 简介.....	7	练习题.....	41
1.3.2 JDK 安装配置.....	7	第 3 章 流程控制结构.....	44
1.3.3 JDK 目录.....	9	3.1 语句概述.....	44
1.4 集成开发环境.....	10	3.2 分支结构.....	45
1.4.1 Eclipse 简介.....	10	3.2.1 if 条件语句.....	45
1.4.2 Eclipse 下载及安装.....	10	3.2.2 switch 开关语句.....	49
1.4.3 Eclipse 基本操作.....	12	3.3 循环结构.....	51
1.5 Java 应用程序.....	16	3.4 转移语句.....	55
1.5.1 Java 语言编写规范.....	16	练习题.....	59
1.5.2 Java 注释.....	17	第 4 章 数组.....	62
1.5.3 Java 打印输出.....	18	4.1 创建数组.....	62
1.5.4 Hello World 程序.....	19	4.1.1 数组的声明.....	62
练习题.....	19	4.1.2 数组的初始化.....	63
第 2 章 数据类型与运算符.....	21	4.2 访问数组.....	64
2.1 字符.....	21	4.3 冒泡排序算法.....	65
2.1.1 字符集.....	21	4.4 foreach 遍历数组.....	66
2.1.2 分隔符.....	22	4.5 二维数组.....	67
2.1.3 标识符.....	22	4.6 Arrays 类.....	69
2.1.4 关键字.....	23	练习题.....	72
2.2 变量和常量.....	23	第 5 章 类和对象.....	76
2.2.1 变量.....	23	5.1 面向对象思想.....	76
2.2.2 常量.....	24	5.1.1 面向对象简介.....	76
2.3 数据类型.....	24	5.1.2 面向对象的基本名称.....	77
2.3.1 基本数据类型.....	24	5.2 类和对象.....	79
2.3.2 引用数据类型.....	28	5.3 方法.....	85

5.3.1	方法的声明	85	7.3.1	上转型对象	150
5.3.2	方法的参数传递机制	86	7.3.2	引用变量的强制类型转换	152
5.3.3	构造方法	88	7.3.3	instanceof 运算符	153
5.3.4	方法的重载	90	7.4	内部类	154
5.3.5	static 关键字	91	7.4.1	非静态内部类	154
5.3.6	this 关键字	92	7.4.2	局部内部类	156
5.3.7	可变参数	94	7.4.3	静态内部类	158
5.4	包	95	7.4.4	匿名内部类	159
5.5	访问权限修饰符	98	7.5	类之间的其他关系	161
5.6	单例类	102	7.5.1	依赖关系	162
练习题		103	7.5.2	关联关系	162
<b>第 6 章</b>	<b>Java 常用类</b>	106	7.5.3	聚合关系	163
6.1	基本类型的封装类	106	7.5.4	组成关系	164
6.2	Object 类	108	练习题		166
6.2.1	equals()方法	109	<b>第 8 章</b>	<b>抽象类、接口和枚举</b>	168
6.2.2	toString()方法	110	8.1	抽象类	168
6.3	字符串类	112	8.1.1	抽象类的定义	168
6.3.1	String 类	113	8.1.2	抽象类的使用	169
6.3.2	StringBuffer 类	115	8.1.3	抽象类的作用	171
6.3.3	StringBuilder 类	117	8.2	接口	172
6.4	Scanner 类	119	8.2.1	接口的定义	172
6.5	Math 类	120	8.2.2	接口的实现	173
6.6	Date 类与 Calendar 类	123	8.2.3	接口的继承	175
6.6.1	Date 类	123	8.3	枚举	176
6.6.2	Calendar 类	125	8.3.1	枚举类的定义	177
6.7	格式化处理	126	8.3.2	包含属性和方法的枚举类	179
6.7.1	数字格式化	127	8.3.3	Enum 类	181
6.7.2	货币格式化	128	练习题		184
6.7.3	日期格式化	129	<b>第 9 章</b>	<b>异常</b>	186
6.7.4	消息格式化	132	9.1	异常概述	186
练习题		135	9.1.1	异常类	186
<b>第 7 章</b>	<b>继承与多态</b>	138	9.1.2	异常处理机制	187
7.1	类之间关系概述	138	9.2	捕获异常	188
7.2	继承	139	9.2.1	try...catch 语句	189
7.2.1	继承的特点	139	9.2.2	try...catch...finally 语句	192
7.2.2	方法的重写	143	9.2.3	嵌套的 try...catch 语句	195
7.2.3	super 关键字	144	9.2.4	多异常捕获	197
7.2.4	final 关键字	148	9.3	抛出异常	198
7.3	多态	150	9.3.1	throw 抛出异常对象	199

9.3.2 throws 声明抛出异常序列 .....	199	11.7.3 Channel.....	250
9.4 自定义异常.....	201	11.7.4 字符集和 Charset.....	252
练习题.....	202	11.7.5 文件锁 .....	254
<b>第 10 章 泛型与集合</b> .....	<b>205</b>	11.7.6 NIO.2.....	255
10.1 泛型.....	205	练习题.....	257
10.1.1 泛型定义.....	205	<b>第 12 章 多线程</b> .....	<b>260</b>
10.1.2 通配符 .....	207	12.1 线程概述.....	260
10.1.3 有界类型 .....	208	12.1.1 线程和进程.....	260
10.1.4 泛型的限制.....	212	12.1.2 Java 线程模型.....	262
10.2 集合概述.....	212	12.1.3 主线程.....	263
10.2.1 集合框架.....	212	12.2 线程的创建和启动.....	264
10.2.2 迭代器接口.....	214	12.3 线程的生命周期.....	269
10.3 集合类.....	215	12.3.1 新建和就绪状态.....	270
10.3.1 Collection 接口 .....	215	12.3.2 运行和阻塞状态 .....	271
10.3.2 List 接口及其实现类.....	217	12.3.3 死亡状态.....	272
10.3.3 Set 接口及其实现类.....	219	12.4 线程的优先级.....	274
10.3.4 Queue 接口及其实现类.....	221	12.5 线程的同步.....	276
10.3.5 Map 接口及其实现类.....	224	12.5.1 同步代码块.....	276
10.4 集合转换.....	226	12.5.2 同步方法.....	279
练习题.....	228	12.5.3 同步锁.....	282
<b>第 11 章 输入/输出流</b> .....	<b>230</b>	12.6 线程通信.....	285
11.1 输入/输出流概述.....	230	12.7 Timer 定时器.....	288
11.2 File 类.....	231	练习题.....	289
11.3 字节流.....	234	<b>第 13 章 Swing UI 设计</b> .....	<b>292</b>
11.3.1 InputStream.....	234	13.1 WindowBuilder 插件.....	292
11.3.2 OutputStream.....	236	13.1.1 WindowBuilder 插件安装 .....	292
11.4 字符流.....	238	13.1.2 WindowBuilder 插件的	
11.4.1 Reader.....	238	使用过程.....	295
11.4.2 Writer.....	240	13.2 GUI 概述.....	298
11.5 过滤流和转换流.....	241	13.2.1 AWT 和 Swing.....	298
11.5.1 过滤流.....	241	13.2.2 Swing 组件层次.....	299
11.5.2 转换流.....	243	13.3 容器与布局.....	299
11.6 对象流.....	244	13.3.1 JFrame 顶级容器 .....	300
11.6.1 对象序列化与反序列化 .....	245	13.3.2 JPanel 中间容器.....	300
11.6.2 ObjectInputStream 和		13.3.3 BorderLayout 边界布局.....	301
ObjectOutputStream .....	245	13.3.4 FlowLayout 流布局.....	301
11.7 NIO.....	247	13.3.5 GridLayout 网格布局 .....	302
11.7.1 NIO 概述 .....	248	13.3.6 CardLayout 卡片布局.....	302
11.7.2 Buffer.....	248	13.3.7 NULL 空布局 .....	303

13.4 基本组件.....	304	14.1.3 JDBC API.....	340
13.4.1 Icon 图标.....	305	14.2 数据库环境搭建.....	343
13.4.2 JButton 按钮.....	306	14.2.1 创建数据库表.....	343
13.4.3 JLabel 标签.....	306	14.2.2 设置 MySQL 驱动类.....	344
13.4.4 文本组件.....	307	14.3 数据库访问.....	345
13.4.5 JComboBox 组合框.....	308	14.3.1 加载数据库驱动.....	345
13.4.6 JList 列表框.....	308	14.3.2 建立数据库连接.....	346
13.4.7 JRadioButton 单选按钮.....	309	14.3.3 创建 Statement 对象.....	346
13.4.8 JCheckBox 复选框.....	310	14.3.4 执行 SQL 语句.....	347
13.4.9 用户注册界面.....	310	14.3.5 访问结果集.....	347
13.5 事件处理.....	316	14.4 操作数据库.....	349
13.5.1 Java 事件处理机制.....	316	14.4.1 execute()方法.....	349
13.5.2 事件和事件监听器.....	318	14.4.2 executeUpdate()方法.....	351
13.6 标准对话框.....	319	14.4.3 PreparedStatement 接口.....	353
13.6.1 消息对话框.....	320	14.5 事务处理.....	355
13.6.2 输入对话框.....	321	练习题.....	357
13.6.3 确认对话框.....	321	<b>第 15 章 网络编程</b> .....	359
13.6.4 选项对话框.....	322	15.1 Java 网络 API.....	359
13.7 菜单.....	322	15.1.1 InetAddress 类.....	359
13.7.1 下拉式菜单.....	322	15.1.2 URL 类.....	362
13.7.2 弹出式菜单.....	326	15.1.3 URLConnection 类.....	364
13.8 表格与树.....	327	15.1.4 URLDecoder 类和 URLEncoder 类.....	365
13.8.1 表格.....	328	15.2 基于 TCP 的网络编程.....	366
13.8.2 树.....	331	15.2.1 Socket 类.....	367
练习题.....	336	15.2.2 ServerSocket 类.....	369
<b>第 14 章 JDBC 与 MySQL 编程</b> .....	338	15.2.3 聊天室.....	372
14.1 JDBC 基础.....	338	练习题.....	378
14.1.1 JDBC 简介.....	338	<b>参考文献</b> .....	380
14.1.2 JDBC 驱动.....	339		



# 第 1 章 Java 概述



## 本章学习目标:

- 熟悉 Java 语言的产生、特点以及编写规范
- 掌握 Java 开发环境和开发工具的使用
- 掌握编写简单的 Java 程序
- 掌握创建、编译和运行 Java 程序的基本步骤

随着计算机技术的不断发展和计算机网络的完善与普及,面向对象程序设计的 Java 语言越来越受到编程爱好者的青睐。本章首先简要介绍了 Java 语言的发展、特点和编写规范,其次详细介绍 Java 语言的开发环境和开发工具的使用,最后详细介绍了 Java 程序的创建、编译和运行的整个过程。

## 1.1 Java 简介

Java 是一种可用于编写跨平台应用程序的面向对象程序设计语言,也是 Java SE(标准版)、Java EE(企业版)和 Java ME(微型版)三种平台的总称。由于 Java 具有“一次编写,多处应用(Write Once, Run Anywhere)”的特点,因此被广泛应用于 PC、数据中心、游戏控制台、科学超级计算机、移动电话和互联网等不同的媒介。Java 具有卓越的通用性、高效性、平台移植性和安全性,为其赢得了大量的爱好者和专业社群组织。

### 1.1.1 Java 起源

Java 自 1995 年诞生,至今已有 20 多年的历史。Java 的名字来源于印度尼西亚爪哇岛。该地因盛产咖啡而闻名,因此 Java 的图标是一杯正冒着热气的咖啡,如图 1.1 所示。Java 来自于 Sun 公司的一个“绿色项目(Green Project)”,其原先的目的是为家用消费电子产品开发一个分布式代码系统,目标是把 E-mail 发给电冰箱、电视机等家用电器,对这些电器进行控制以及信息交流。詹姆斯·高斯林(James Gosling)(见图 1.2)加入到该项目小组。开始,项目小组准备采



图 1.1 Java 图标

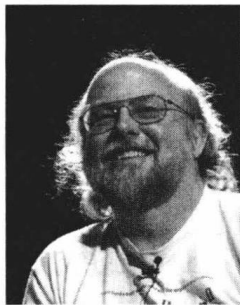


图 1.2 詹姆斯·高斯林



用 C++，但 C++ 太复杂，安全性差，最后高斯林用 C++ 开发了一种新的语言 Oak(橡树)，这就是 Java 的前身，1994 年 Oak 被正式更名为 Java。詹姆斯·高斯林(James Gosling)也被人们亲切地称为 Java 之父。

## 1.1.2 Java 发展史

自 1995 年,Java 先后经历了 8 个版本的变更,版权的所有者也一度由 Sun 变为 Oracle。表 1-1 所示为 Java 发展过程中几个重要的里程碑。

表 1-1 Java 发展史

日期	版本号	说明
1995 年 5 月 23 日	无	Java 语言诞生
1996 年 1 月	JDK 1.0	第一个 JDK 1.0 诞生,还不能进行真正的应用开发
1998 年 12 月 8 日	JDK 1.2	企业平台 J2EE 发布,里程碑式的产品,性能提高,完整的 API
1999 年 6 月	Java 三个版本	标准版(J2SE)、企业版(J2EE)、微型版(J2ME)
2000 年 5 月 8 日	JDK 1.3	JDK1.3 发布,对 1.2 版进行改进,扩展标准类库
2000 年 5 月 29 日	JDK 1.4	JDK 1.4 正式发布,提高系统性能,修正一些 Bug
2001 年 9 月 24 日	J2SE 1.3	J2SE 1.3 正式发布
2002 年 2 月 26 日	J2SE 1.4	计算能力有了大幅提升
2004 年 9 月 30 日	J2SE 5.0	Java 语言发展史上的重要里程碑,从该版本开始,增加了泛型类、foreach 循环、可变元参数,自动打包、枚举、静态导入和元数据等技术,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	Java SE 6.0	发布 Java SE 6.0,此时 Java 的各种版本已更名,取消数字“2”,分别更名为:Java EE、Java SE、Java ME
2006 年 12 月	JRE 6.0	SUN 公司发布 JRE 6.0
2009 年 04 月 20 日	收购	甲骨文 74 亿美元收购 Sun,获得 Java 版权
2011 年 7 月 28 日	Java SE 7.0	甲骨文发布 Java SE 7.0 正式版
2014 年 3 月	Java SE 8.0	又一里程碑,甲骨文发布 Java SE 8.0,增加 Lambda、Default Method 等新特性
2014 年 11 月	Java SE 9.0	增加的新特性比较重要的有:统一的 JVM 日志,支持 HTTP 2.0、Unicode 7.0、安全数据包传输(DTLS)、Linux/AArch64

## 1.1.3 Java 的特点

Java 语言之所以受到广大编程爱好者的青睐,是因为其有着以下几方面的语言优势。

### 1. 资源开源性

Java 技术虽然最初由 Sun 公司开发,但是 Java Community Process(JCP,一个由全世界的 Java 开发人员和获得许可的人员组成的开放性组织)可以对 Java 技术规范、参考实现和



技术兼容包进行开发和修订。虚拟机和类库的源代码都可以免费获取，但只能查阅，不能修改和再发布。

## 2. 跨平台性

Java 是一种与平台无关的语言，它可以跨越各种操作系统、硬件平台以及可移动和嵌入式部件，其源代码被编译成一种结构中立的中间文件(.class, 字节码)，在 Java 虚拟机上运行。

## 3. 健壮性和安全性

Java 是一种强类型的语言，其强类型机制、异常处理、垃圾的自动回收等都是 Java 程序健壮性的重要保证，对指针的丢弃是 Java 的明智选择，Java 的安全检查机制使得 Java 更具健壮性。

在安全性方面，Java 提供了一个安全机制以防止恶意代码的攻击，对通过网络下载的一类具有一个安全防范机制，如分配不同的名字空间以防替代本地的同名类、字节代码检查，并提供安全管理机制(类 SecurityManager)，让 Java 应用设置安全哨兵。

## 4. 高性能性

Java 是一种解释执行的语言，其速度和使用与其他基于编译器的语言(C 和 C++)相当。为了提高执行速度，Java 引入 JIT(Just In Time, 即时)编译技术，可以对执行过程进行优化，如监控经常执行代码并进行优化、保存翻译过的机器码、消除函数调用(内嵌)等，以加快程序的执行速度。事实上，与其他解释型的高级脚本语言相比，Java 的运行速度随着 JIT 编译器技术的发展越来越接近于 C++。

## 5. 简单性

Java 语言简单易学、使用方便，语法结构更加简洁统一，编程风格类似于 C++，而且摒弃了 C++ 中容易引发程序错误的一些特性，如指针、结构体等。在内存管理方法中，Java 提供垃圾内存自动回收机制，有效地避免了 C++ 中内存泄漏的问题。Java 还提供了丰富的类库，使开发人员不需要懂得底层的工作原理就可以实现应用开发。

## 6. 面向对象

Java 是一种完全面向对象的语言，基于对象的编程更符合人的思维模式和编写习惯，已经成为主流的程序设计方式。Java 语言支持继承、重载、多态等面向对象的特性。

## 7. 动态性

Java 的动态特性是其面向对象设计方法的扩展。Java 允许程序动态地装入运行过程中所需要的类，这是采用 C++ 语言进行面向对象程序设计所无法实现的。

## 8. 多线程

Java 内置了对多线程的支持，提供了用于同步多个线程的解决方案。相对于 C/C++，使用 Java 编写多线程使应用程序变得更加简单，这种对线程的内置支持使交互式应用程序能在 Internet 上顺利运行。

## 9. 支持分布式网络应用

Java 除了支持基本的语言功能，还包括一个支持 HTTP 和 FTP 等基于 TCP/IP 的类库。Java 应用程序可通过 URL 打开，并访问网络上的对象，其访问方式与访问本地文件系统一



样方便。由于在网上传输的只是 Java 程序的字节码，而且 Java 程序中的每一个类都单独编译成一个字节码文件，所以传输量小、传输速度快，因此，Java 是适合在网上运行的编程语言。

## 1.2 Java 体系

### 1.2.1 Java 应用平台

1999 年，在美国旧金山的 Java One 大会上，Sun 公司公布了 Java 体系架构，该架构根据不同级别的应用开发划分了三个版本，如图 1.3 所示。Java 三个平台应用于不同方向。

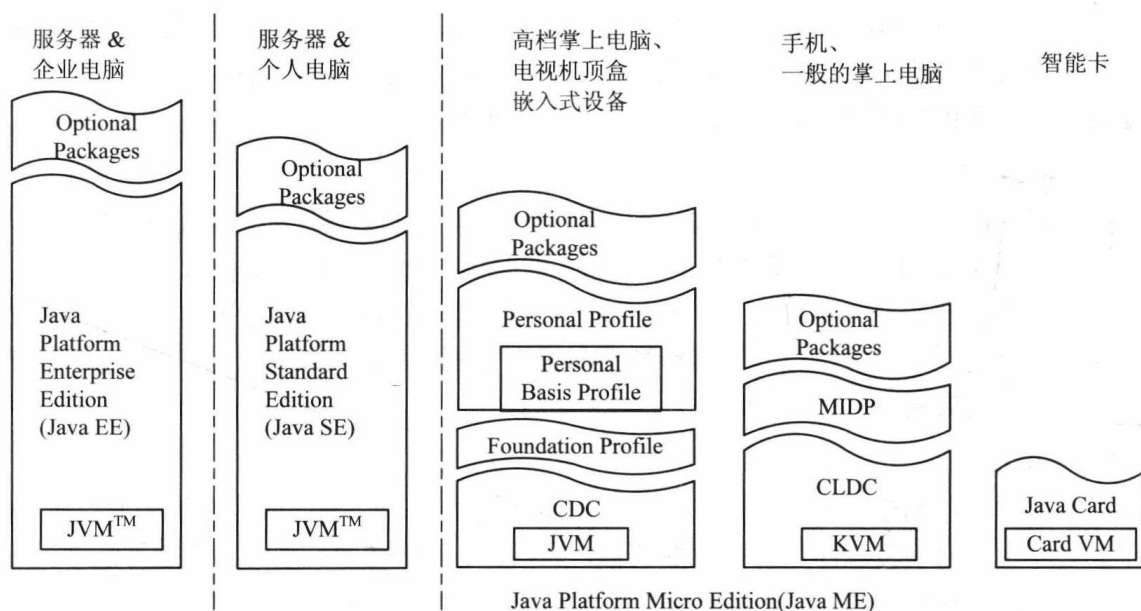


图 1.3 Java 应用平台

#### 1. Java SE(Java Standard Edition, Java 标准版)

Java SE 是 Java 技术的基础,适用于桌面系统应用程序(Application)、网页小程序(Applet)以及服务器程序的开发。Java SE 主要包括 Java 语言核心技术和应用,如数据库访问、I/O、网络编程、多线程等。

#### 2. Java EE(Java Enterprise Edition, Java 企业版)

Java EE 是企业级解决方案,支持开发、部署和管理等相关复杂问题的体系结构,主要用于分布式系统的开发、构建企业级的服务器应用,例如,电子商务网站、ERP 系统等。Java EE 在 Java SE 基础上定义了一系列的服务、API 和协议等,如 Servlet、JSP、RMI、EJB、Java Mail、JTA 等。

#### 3. Java ME(Java Micro Edition, Java 微型版)

Java ME 是各版本中最小的,在 SE 基础上进行了裁剪和高度优化,目的是在小型的受限设备上开发和部署应用程序,例如手机、PDA、智能卡、机顶盒、汽车导航或家电系统等。Java ME 遵循微型开发规范和技术,如 MIDLet、CLDC、Personal Profile 等。



## 1.2.2 Java 专有名词

### 1. JDK

JDK(Java Development Kit, Java 开发工具包)是 Sun 公司提供的一套用于开发 Java 程序的开发工具包。JDK 提供编译、运行 Java 程序所需要的各种工具及资源,包括 Java 开发工具、Java 运行时的环境以及 Java 的基础类库。

### 2. JRE

JRE(Java Runtime Environment, Java 运行时环境)是运行 Java 程序所依赖的环境的集合,包括类加载器、字节码校验器、Java 虚拟机、Java API。JRE 已包含在 JDK 中,但是如果仅仅是为了运行 Java 程序,而不是从事 Java 开发,可以直接下载安装 JRE。

### 3. JVM

JVM(Java Virtual Machine, Java 虚拟机)是一个虚构出来的计算机,是通过在实际的计算机上仿真模拟各种计算机功能来实现的。Java 虚拟机有自己完善的硬件架构,如处理器、堆栈、寄存器等,还具有相应的指令系统。Java 虚拟机屏蔽了与具体操作系统平台相关的信息,只需将 Java 语言程序编译成在 Java 虚拟机上运行的目标代码(.class, 字节码),就可以在多种平台上不加修改地运行。Java 虚拟机在执行字节码时,实际上最终还是把字节码解释成具体平台上的机器指令来执行。

### 4. SDK

SDK(Software Development Kit, 软件开发工具包)在版本 1.2 到 1.4 时,被称为 Java SDK。

JVM、JRE 和 JDK 三者有非常紧密的关系,从范围上来讲是从小到大的关系,如图 1.4 所示。因此,在计算机上安装 JDK 时,会同时将 JRE 和 JVM 安装到计算机中。

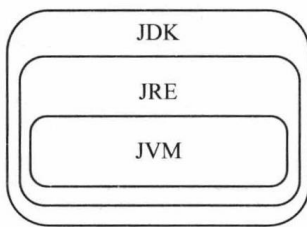


图 1.4 JVM、JRE 和 JDK 的关系

## 1.2.3 Java 跨平台原理

JVM 在具体的操作系统之上运行,其本身具有一套虚拟指令,这套虚拟指令通常是在软件上而不是在硬件上实现。JVM 形成一个抽象层,将底层硬件平台、操作系统与编译过的代码联系起来。Java 字节码的格式通用,具有跨平台特性,但这种跨平台性是建立在 JVM 虚拟机的基础之上的,只有通过 JVM 处理后才可以将字节码转换为特定机器上的机器码,然后在特定的机器上运行。JVM 跨平台原理如图 1.5 所示。

JVM 体现了 Java 程序具有“一次编译,多处应用”的特性,如图 1.6 所示。首先,Java 编译器将 Java 源程序编译成 Java 字节码;然后,字节码在本地或通过网络传达给 JVM 虚拟机;接着,JVM 对字节码进行即时编译或解释执行后,形成二进制的机器码;最后,生



成的机器码可以在硬件设备上直接运行。

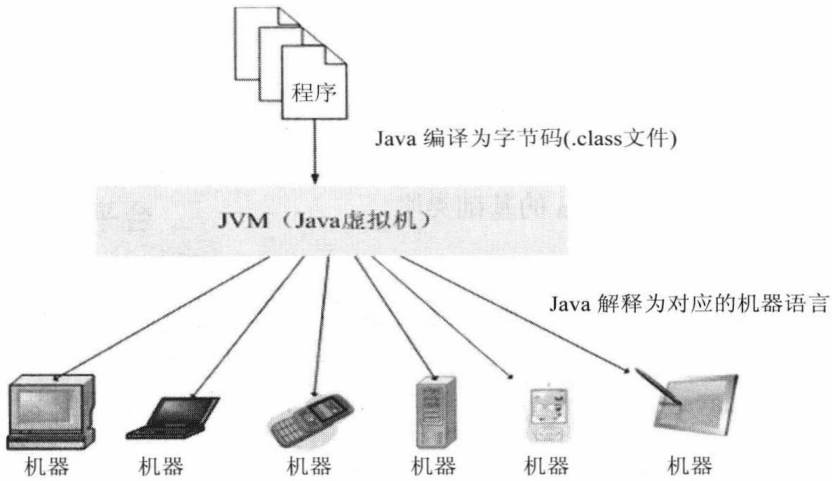


图 1.5 JVM 跨平台示意图

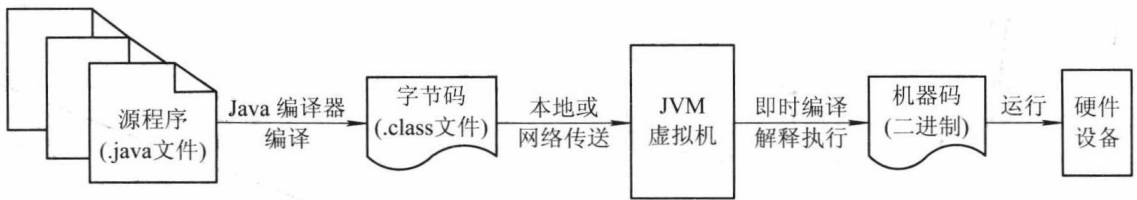


图 1.6 Java 程序运行机制

JVM 执行时将在其内部创建一个运行时环境，每次读取并执行一条 Java 语句会经过三个过程：装载、校验和执行，如图 1.7 所示。

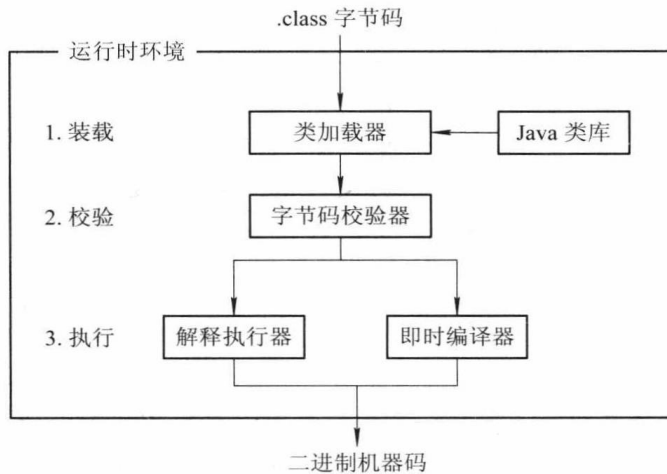


图 1.7 JVM 运行过程图

Java 字节码有两种执行方式：

(1) 解释执行方式。JVM 通过解释器将字节码逐条读入，逐条解释翻译成对应的机器指令。很显然，这种执行方式虽灵活但执行速度会比较慢。为了提高执行速度，引入了 JIT(Just-In-Time Compiler)技术。

(2) 即时编译方式(JIT 编译)。当 JIT 编译启用时(默认是启用的)，JVM 将解释后的字



字节码文件发给 JIT 编译器, JIT 编译器将字节码编译成机器代码, 并把编译过的机器码保存起来, 以备下次使用。为了加快执行速度, JIT 目前只对经常使用的热代码进行编译。

通常解释执行方式采用较多。由于 JVM 规格描述具有足够的灵活性, 因此使得将字节码翻译为机器代码的工作具有较高的效率。对于那些对运行速度要求较高的应用程序, 解释器可将 Java 字节码即时编译为机器码, 从而很好地保证了 Java 代码的可移植性和高性能。

## 1.3 JDK 工具

“工欲善其事, 必先利其器。”在开发的第一步, 必须搭建起开发环境。本书以 Java SE Development Kit 8 在 Windows 操作系统的下载、安装作为范例, 讲解整个 Java 开发环境的安装及配置过程。

### 1.3.1 JDK 简介

JDK(Java Development Kit)是 Sun Microsystems 公司针对 Java 开发人员发布的免费软件开发工具包。JDK 是整个 Java 的核心, 是学好 Java 的第一步。如果没有 JDK, 则无法安装或者运行 Eclipse。普通用户并不需要安装 JDK 来运行 Java 程序, 而只需要安装 JRE(Java Runtime Environment), 而程序开发者必须安装 JDK 来编译、调试程序。

从 Sun 公司的 JDK 5.0 开始, JDK 提供了泛型等非常实用的功能, 其版本也不断更新, 运行效率得到了非常大的提高, 其环境变量也可以不需要手动配置。

JDK 包含一批用于 Java 开发的组件, 其中包括以下几部分:

- (1) Java 开发工具。Java 开发工具都是可执行程序, 主要包括: javac.exe(编译工具)、java.exe(运行工具)、javadoc.exe(生成 JavaDoc 文档的工具)和 jar.exe(打包工具)等。
- (2) Java 运行环境。Java 虚拟机可以运行在各种操作系统平台上, 负责解析和执行 Java 程序。
- (3) Java 继承类库(rt.jar)。Java 继承类库提供了最基础的 Java 类以及各种实用类, 如 java.lang、java.io、java.util、java.awt、java.swing 和 java.sql 包中的类都位于 JDK 类库中。

### 1.3.2 JDK 安装配置

JDK 下载与安装包括下载 JDK 和安装 JDK 两个内容, 下面详细介绍下载 JDK 和安装 JDK 的具体过程。

#### 1. 下载 JDK

进入 Oracle 官方网站可以下载 JDK 的最新版本。

Oracle 官方网站: <http://www.oracle.com>

JDK 8.0 的下载地址: <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

JDK 8.0 的下载页面如图 1.8 所示。下载 JDK 8.0 的 Windows x64 版本, 即 jdk-8u171-windows-x64.exe。由于不同版本的下载地址会经常发生变化, 因此最有效的方法是访问官方网站, 通过导航找到下载页面。如果是 32 位操作系统, 则下载对应的“x86”版本。



Java SE Development Kit 8u171		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
<input type="radio"/> Accept License Agreement <input type="radio"/> Decline License Agreement		
Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	77.97 MB	jdk-8u171-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	74.89 MB	jdk-8u171-linux-arm64-vfp-hflt.tar.gz
Linux x86	170.05 MB	jdk-8u171-linux-i586.rpm
Linux x86	184.88 MB	jdk-8u171-linux-i586.tar.gz
Linux x64	167.14 MB	jdk-8u171-linux-x64.rpm
Linux x64	182.05 MB	jdk-8u171-linux-x64.tar.gz
Mac OS X x64	247.84 MB	jdk-8u171-macosx-x64.dmg
Solaris SPARC 64-bit (SVR4 package)	139.83 MB	jdk-8u171-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	99.19 MB	jdk-8u171-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 package)	140.6 MB	jdk-8u171-solaris-x64.tar.Z
Solaris x64	97.05 MB	jdk-8u171-solaris-x64.tar.gz
Windows x86	199.1 MB	jdk-8u171-windows-i586.exe
Windows x64	207.27 MB	jdk-8u171-windows-x64.exe

图 1.8 下载 Windows x64 版本

## 2. 安装 JDK

安装 JDK 可按以下步骤操作：

(1) 运行 JDK 的安装文件，进入 JDK 的安装程序向导界面，如图 1.9 所示。

(2) 单击“下一步”按钮，进入定制安装界面，如图 1.10 所示。可以单击右下方的“更改”按钮，设置 JDK 的安装路径，否则进入默认安装路径。

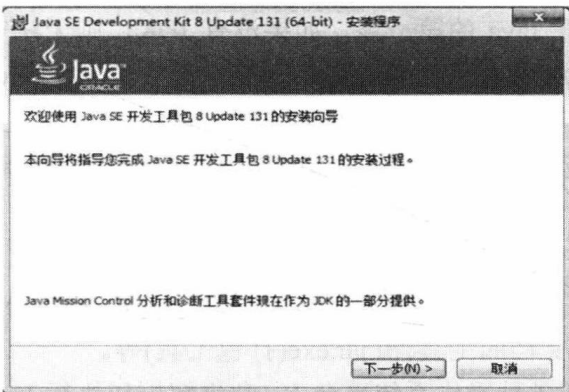


图 1.9 安装程序向导界面

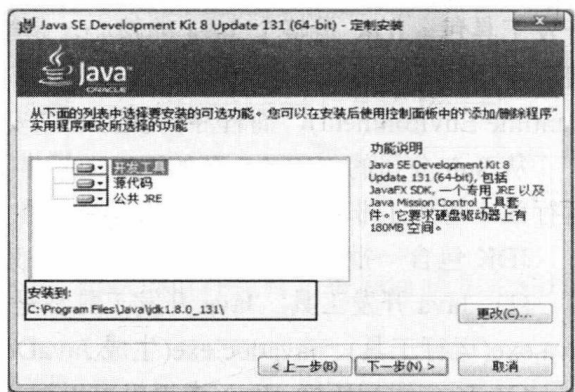


图 1.10 定制安装界面

(3) 单击“下一步”按钮，进入安装进度界面，如图 1.11 所示。

(4) JDK 安装进度完成，进入“目标文件夹”对话框，如图 1.12 所示。可以单击“更改”按钮，选择 JRE 的安装路径。一般要求 JDK 和 JRE 安装在同一个文件夹内。

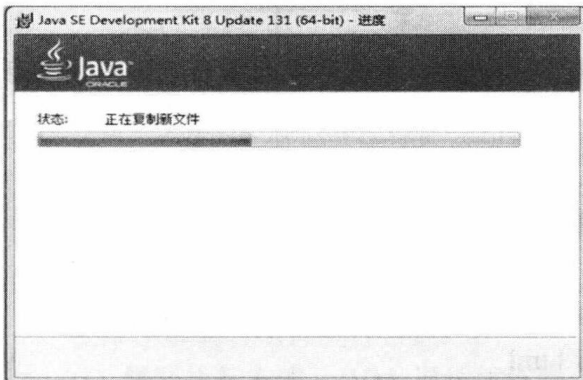


图 1.11 安装进度界面

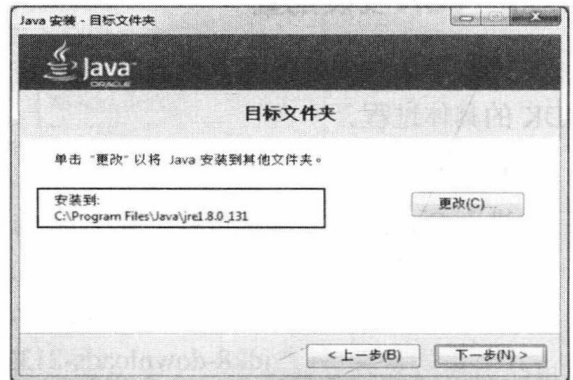


图 1.12 “目标文件夹”对话框

(5) 单击“下一步”按钮，进入 JRE 安装进度界面，如图 1.13 所示。当安装完成以后，便进入 JDK 安装成功界面，如图 1.14 所示。单击“关闭”按钮，完成 JDK 整个安装过程。

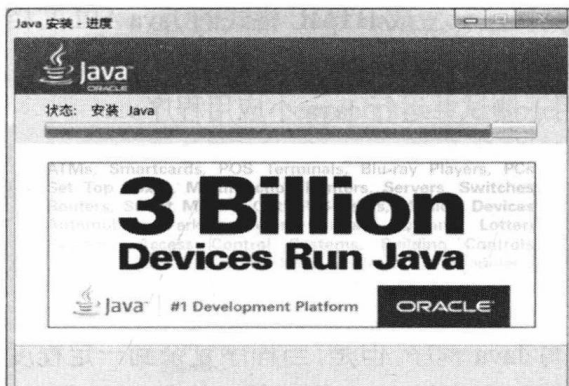


图 1.13 JRE 安装进度界面

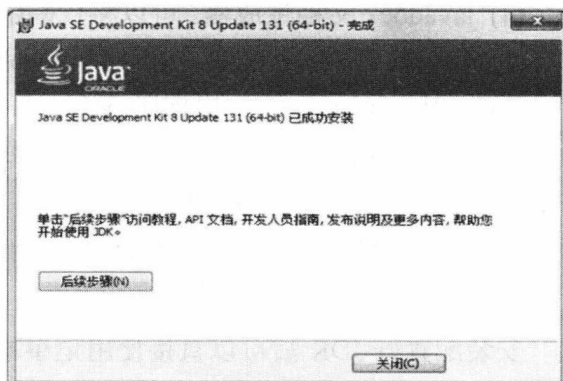


图 1.14 JDK 安装成功界面

### 1.3.3 JDK 目录

JDK 安装完成后，在安装的位置中可以找到如图 1.15 所示的目录。

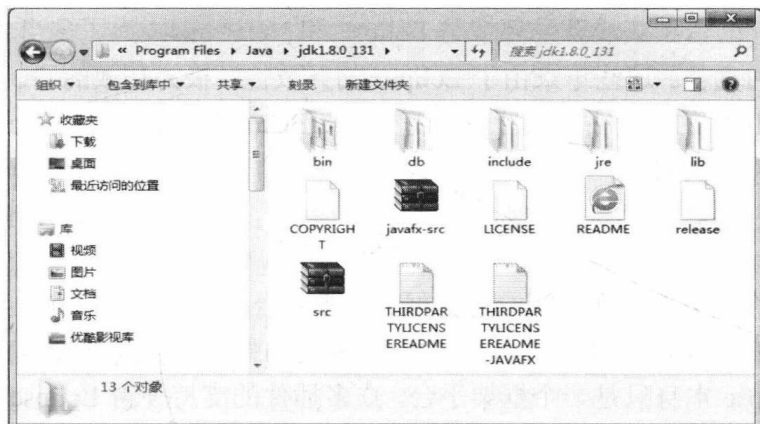


图 1.15 JDK 8.0 安装完成目录

JDK 主要目录如下：

(1) bin: JDK 包中命令及工具所在目录。这是 JDK 中非常重要的目录，它包含大量开发过程中的常用工具程序，如编译器、解释器、打包工具、代码转换器和相关调试工具等。

(2) jre: 运行环境目录。这是 JDK 自己附带的 Java 运行环境。

(3) lib: 类库所在目录。包含了开发所需要的类库(即 Java API)和支持文件。

(4) db: 附带数据库目录。在 JDK 6.0 以上的版本中附带 Apache Derby 数据库，这是一个 Java 编写的数据库，支持 JDBC 4.0。

(5) include: 包含本地代码的 C 头文件的目录。用于支持 java 本地接口和 Java 虚拟机调试程序接口的本地代码编译。

(6) src: 源代码压缩文件。Java 提供的 API 都可以通过此文件查看其源代码是如何实现的。在 JDK 的 bin 目录下，提供了大量的开发工具程序，以下是几个常用的工具：

(1) javac: Java 语言编译器。可以将 Java 源文件编译成与平台无关的字节码文件(.class 文件)。

(2) java: Java 字节码解释器。将字节码文件在不同的平台中解释执行。

(3) javap: Java 字节码分解程序。可以查看 Java 程序的变量以及方法等信息。