

轻松学
编程

轻松学

Java编程

赵振方 陈景霞 王慧 等编著



DVD-ROM

1600分钟多媒体教学视频

每个知识点按照“概念→语法描述→实例描述→实例应用→运行结果→示例分析”的模式进行讲解

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，用形象的比喻帮助记忆

重要知识点覆盖全面，实例丰富多彩

在随书附赠的光盘中配备了完整的视频教学讲座和本书所使用到的全部项目文件与代码

读者可直接登录www.itzcn.com与作者取得联系，会在第一时间内得到帮助



化学工业出版社

轻松学
编程

轻松学

Java编程

赵振方 陈景霞 王慧 等编著



DVD-ROM

1600分钟多媒体教学视频

每个知识点按照“概念→语法描述→实例描述→实例应用→运行结果→示例分析”的模式进行讲解

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，用形象的比喻帮助记忆

重要知识点覆盖全面，实例丰富多彩

在随书附赠的光盘中配备了完整的视频教学讲座和本书所使用到的全部项目文件与代码

读者可直接登录www.itzcn.com与作者取得联系，会在第一时间内得到帮助



化学工业出版社

本书全面介绍 Java 编程知识, 首先介绍了 Java 的工作原理、基本语法、对数组的操作、对字符串的操作, 并深入地讲解了 Java 语言中的面向对象编程, 包括类与对象的基本知识及面向对象的封装、继承、多态; 然后介绍了 Java 的开发应用, 包括 Java 中的常用工具类、异常处理、I/O 流、集合框架、图形用户界面和数据库编程等相关知识; 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。

本书可以作为 Java 的基础入门学习书籍, 也可以帮助中级程序员提高编程技能, 掌握面向实践的应用技能。

图书在版编目 (CIP) 数据

轻松学 Java 编程 / 赵振方, 陈景霞, 王慧等编著. —

北京: 化学工业出版社, 2012. 3

(轻松学编程)

ISBN 978-7-122-13420-2

ISBN 978-7-89472-585-1 (光盘)

I. 轻… II. ①赵… ②陈… ③王… III. Java
语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2012) 第 019122 号

责任编辑: 瞿 微

装帧设计: 王晓宇

出版发行: 化学工业出版社 (北京市东城区青年湖南街 13 号 邮政编码 100011)

印 装: 化学工业出版社印刷厂

787mm×1092mm 1/16 印张 22¹/₂ 字数 576 千字 2012 年 4 月北京第 1 版第 1 次印刷

购书咨询: 010-64518888 (传真: 010-64519686)

售后服务: 010-64518899

网 址: <http://www.cip.com.cn>

凡购买本书, 如有缺损质量问题, 本社销售中心负责调换。

定 价: 45.00 元 (含 1DVD-ROM)

版权所有 违者必究

Java 是当今较流行的一种简单的、面向对象的、分布式的、健壮安全、结构中立、可移植的、性能优异的多线程动态语言。该语言由 Sun 公司于 1995 年推出，发展到今天，已不仅仅表示一门程序语言，更是一种软件开发平台，目前已经演化出了 J2SE、J2EE 和 J2ME 三个版本。凭借其易学易用和功能强大的特点，可以进行桌面应用、Web 应用、分布式系统及嵌入式系统等应用程序开发，并且在信息技术、科学研究、军事工业、航空航天等各个领域得到了非常广泛的应用。

● 本书内容

第 1 章：Java 语言概述。介绍了 Java 语言的特点和发展史，Java 程序的工作原理及环境配置，以及 Java 程序的开发步骤。

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

第 3 章：流程控制语句。主要介绍了 Java 语言中的选择语句（if、if-else、if-else if-else、嵌套的 if 语句和 switch 语句）、循环语句（while、do-while、for 和 foreach 语句）和跳转语句（break、continue 和 return 语句）。

第 4 章：数组。主要介绍了一维数组的创建和使用，多维数组的创建、使用和访问，以及对数组元素的搜索、排序等操作。

第 5 章：字符串处理。主要介绍了字符串的创建方式，字符串的常用方法及对字符串的常用操作，以及 StringBuffer 类的使用。

第 6 章：类与对象。主要介绍了面向对象的核心特征，类与对象的区别，在 Java 语言中常用的修饰符，包括 private、friendly、protected、public、static 和 final。

第 7 章：深入面向对象编程。主要介绍了面向对象的封装、继承、多态 3 大特征的应用，抽象类与接口的区别及使用，以及包的创建及命名规则。

第 8 章：异常处理。主要介绍了异常类型的种类，异常处理机制，以及如何抛出异常、声明异常及如何自定义异常。

第 9 章：Java 常用工具类。主要介绍了 Java 语言中常用的工具类，包括 Object 类、包装类（Integer 类和 Character 类）、日期类、格式化日期类和 Random 类。

第 10 章：Java 的输入/输出流。主要介绍了如何使用字节流来写入/读取文件，如何使用过滤流来对文件中的内容进行读、写操作和字符流对文件的应用，以及如何使用 File 类中的方法来获取文件的相关信息，如文件名称、文件大小、文件内容长度等。

第 11 章：集合框架。主要介绍了 Java 语言中的集合框架常用接口（Set 接口、List 接口和 Map 接口），以及实现这 3 个接口的常用集合类（ArrayList、LinkedList 和 HashMap）。

第 12 章：图形用户界面。首先介绍了 AWT 中容器的使用及基本组件的创建；然后介绍 5 种布局管理器的应用；最后介绍了组件中的事件处理机制。

第 13 章：数据库编程。主要介绍了 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 14 章：学生信息管理系统。使用 Java 语言的 AWT 技术开发了一个桌面程序——学生管理系统，该系统使用 JDBC 连接方式连接 MySQL 5.5 数据库，能动态地管理学生信息数据。

● 本书特色

本书中大量内容来自真实的 Java 程序，力求通过实际操作使读者更容易地掌握 Java 应用开发。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、运行结果、示例分析”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，用形象的比喻帮助记忆。

● 技术文档

将一些非常简单的知识点或者理论性的内容安排在这里，通常这些文档没有具体的实际问题，但又是读者必须要了解的，如一些概念和术语。

● 内容丰富

涵盖了实际开发中 Java 技术所遇到的 I/O 流、AWT 等方面的热点问题。

● 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进行学习。

● 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 与我们取得联系，编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。

● 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编程技能。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以及相关社会培训班的学员。

● 本书编者

本书主要由赵振方、陈景霞、王慧编写，其他参与编写、资料整理、程序开发的人员还有邱森、周衡坤、张浩华、王丹花、胡海静、李焕娜、李振等。其中陕西科技大学教师陈景霞负责第 4~6 章的编写。

由于编者水平有限，书中难免存在不足和疏漏之处，恳请读者批评指正。

编者

2012 年 2 月

第 1 章 Java 语言概述	1
1.1 Java 语言简介	2
1.1.1 Java 语言的特点	2
1.1.2 Java 语言的发展	2
1.2 Java 语言的工作原理	3
1.2.1 Java 程序的运行流程	3
1.2.2 JVM 介绍	4
1.3 Java 程序的运行环境	5
1.3.1 JDK 介绍	5
1.3.2 下载和安装 JDK	6
1.3.3 配置 JDK	8
1.3.4 获得 Java API 文档	9
1.4 Java Application 和 Java Applet 介绍	10
1.4.1 Application 介绍	11
1.4.2 Applet 介绍	13
1.5 Java 程序的基本要素	14
1.6 Java 的开发工具	18
1.6.1 Eclipse 概述	18
1.6.2 Eclipse 的下载和安装	19
1.6.3 Eclipse 开发程序	20
1.7 Java 程序调试技巧	22
1.8 实践案例：创建 HelloWorld 程序	24
1.9 动手练一练	25
1.9.1 练习题	25
1.9.2 上机实践	26
第 2 章 简单数据类型及运算	27
2.1 Java 数据类型	28
2.1.1 整数类型	29
2.1.2 浮点型	30
2.1.3 字符型	30
2.1.4 布尔类型	31
2.1.5 数据类型间的转换	31
2.1.6 转义序列	33

2.2	变量与常量	34
2.2.1	变量	34
2.2.2	变量的作用域	35
2.2.3	常量	36
2.3	运算符与表达式	37
2.3.1	算术运算符及表达式	37
2.3.2	赋值运算符及表达式	38
2.3.3	自增和自减运算符	39
2.3.4	关系运算符及表达式	42
2.3.5	逻辑运算符及表达式	43
2.3.6	位运算符及表达式	45
2.3.7	条件运算符及表达式	47
2.3.8	运算符的优先级	48
2.4	实践案例：打印超市售货清单	49
2.5	动手练一练	50
2.5.1	练习题	50
2.5.2	上机实践	52
第3章	流程控制语句	53
3.1	选择语句	54
3.1.1	简单的 if 条件语句	54
3.1.2	if-else 条件语句	56
3.1.3	if-else if-else 条件语句	57
3.1.4	嵌套 if 语句	59
3.1.5	switch 多分支语句	61
3.2	循环语句	63
3.2.1	while 循环语句	63
3.2.2	do - while 循环语句	64
3.2.3	for 循环语句	66
3.2.4	foreach 循环语句	68
3.3	跳转语句	69
3.3.1	break 语句	69
3.3.2	continue 语句	71
3.3.3	return 语句	73
3.4	实践案例：制作万年历	74
3.5	动手练一练	77
3.5.1	练习题	77
3.5.2	上机实践	79

第4章 数组	80
4.1 一维数组	81
4.1.1 声明数组	81
4.1.2 分配空间	81
4.1.3 一维数组的初始化	82
4.1.4 一维数组的应用	84
4.2 多维数组	85
4.2.1 声明多维数组	85
4.2.2 给多维数组分配空间	85
4.2.3 初始化多维数组	86
4.2.4 访问多维数组中的元素	87
4.3 搜索数组	88
4.4 数组排序	89
4.4.1 Arrays.sort()排序法	90
4.4.2 直接选择排序法	91
4.4.3 冒泡排序法	92
4.5 实践案例：购物结账程序	94
4.6 动手练一练	96
4.6.1 练习题	96
4.6.2 上机实践	97
第5章 字符串	98
5.1 创建字符串	99
5.1.1 直接使用字符串常量创建字符串变量	99
5.1.2 使用 new 关键字创建字符串变量	99
5.2 字符串的常用方法	100
5.3 字符串的常用操作	101
5.3.1 获取字符串的长度	101
5.3.2 转换字符串的大小写	102
5.3.3 去掉字符串的首尾空格	102
5.3.4 字符串比较	102
5.3.5 字符串搜索	103
5.3.6 字符串提取	106
5.3.7 字符串连接	107
5.3.8 字符串替换	108
5.3.9 字符串分割	109
5.4 StringBuffer 类	110
5.4.1 构造 StringBuffer 类	110

5.4.2	StringBuffer 类的常用方法	111
5.5	实践案例：完善购物结账系统	114
5.6	动手练一练	117
5.6.1	练习题	117
5.6.2	上机实践	118
第 6 章	类与对象	120
6.1	面向对象概述	121
6.1.1	对象的概述	121
6.1.2	面向对象的核心特性	121
6.2	类结构	124
6.2.1	类概述	124
6.2.2	类和对象的区别	125
6.2.3	类是对象的类型	125
6.3	Java 是面向对象的语言	126
6.3.1	类的定义	126
6.3.2	定义类的成员变量	127
6.3.3	定义类的成员方法	128
6.3.4	创建类的构造方法	131
6.3.5	创建对象	134
6.3.6	访问对象的属性和方法	137
6.3.7	类的主方法——main()方法	138
6.3.8	对象的销毁	139
6.3.9	this 关键字	140
6.4	常用修饰符	141
6.4.1	访问控制修饰符	141
6.4.2	static 静态修饰符	143
6.4.3	final 修饰符	147
6.5	实践案例：实现发送短信功能	148
6.6	动手练一练	151
6.6.1	练习题	151
6.6.2	上机实践	153
第 7 章	深入面向对象编程	154
7.1	类的封装	155
7.2	类的继承	157
7.2.1	在 Java 语言中实现继承	157
7.2.2	单继承	162
7.2.3	使用 super 关键字	163

7.3	方法重载和方法重写	165
7.3.1	方法重载	165
7.3.2	方法重写和变量隐藏	166
7.4	抽象类与接口	168
7.4.1	抽象类	168
7.4.2	接口	170
7.5	包	173
7.5.1	定义包	173
7.5.2	使用包	175
7.6	实践案例：使用多级继承实现计算长方体体积、重量和价值	176
7.7	动手练一练	179
7.7.1	练习题	179
7.7.2	上机实践	180
第 8 章	异常处理	181
8.1	异常和异常类型	182
8.1.1	异常的基本概念	182
8.1.2	异常类型	183
8.2	Java 异常处理机制	185
8.2.1	使用 try-catch 语句捕获异常	185
8.2.2	使用多重 catch 语句捕获异常	187
8.2.3	使用 try-catch-finally 语句捕获异常	188
8.3	抛出异常和声明异常	191
8.3.1	抛出异常	191
8.3.2	声明异常	193
8.4	自定义异常	194
8.5	实践案例：完成添加用户的功能	196
8.6	动手练一练	198
8.6.1	练习题	198
8.6.2	上机实践	199
第 9 章	Java 常用工具类	200
9.1	Object 类	201
9.2	包装类	201
9.2.1	基本数据类型所对应的包装类	201
9.2.2	Integer 类	202
9.2.3	Character 类	204
9.3	日期类	207
9.3.1	java.util.Date 类	207

9.3.2	java.util.Calendar 类	209
9.4	格式化日期类	211
9.4.1	DateFormat 类	211
9.4.2	SimpleDateFormat 类	213
9.5	Random 类	214
9.6	实践案例：使用时间类完成日期查询功能	217
9.7	动手练一练	221
9.7.1	练习题	221
9.7.2	上机实践	222
第 10 章	Java 的输入/输出流	223
10.1	输入/输出流概述	224
10.1.1	字节流	224
10.1.2	字符流	225
10.1.3	预定义流	226
10.2	字节流	227
10.2.1	字节输入/输出流	228
10.2.2	文件字节输入/输出流	229
10.3	字符流	233
10.3.1	字符输入/输出流	234
10.3.2	字符文件输入/输出流	235
10.3.3	字符缓冲区输入/输出流	238
10.4	File 类	240
10.4.1	File 类的构造函数	240
10.4.2	File 类的常用方法	240
10.5	RandomAccessFile 类随机访问文件	243
10.6	实践案例：将新录学生信息添加到文件中并读取	245
10.7	动手练一练	248
10.7.1	练习题	248
10.7.2	上机实践	249
第 11 章	集合框架	250
11.1	集合框架概述	251
11.2	Collection 接口	252
11.3	无序列表——Set 集合	253
11.3.1	HashSet 类	253
11.3.2	TreeSet 类	254
11.4	有序列表——List 集合	257
11.4.1	ArrayList 类	258

11.4.2	LinkedList 类	259
11.5	映射——Map	261
11.6	泛型集合	263
11.7	实践案例：使用泛型集合模拟 BBS 论坛	264
11.8	动手练一练	267
11.8.1	练习题	267
11.8.2	上机实践	268
第 12 章	图形用户界面	269
12.1	图形对象概述	270
12.2	容器	271
12.2.1	Window 类	271
12.2.2	Panel 类	274
12.3	AWT 基本组件	275
12.3.1	按钮组件——Button	275
12.3.2	标签——Label	276
12.3.3	单行文本框——TextField	277
12.3.4	多行文本域——TextArea	279
12.3.5	复选框——Checkbox	280
12.3.6	复选框组——CheckboxGroup	281
12.4	布局管理器	282
12.4.1	BorderLayout 管理器	282
12.4.2	FlowLayout 管理器	284
12.4.3	CardLayout 管理器	284
12.4.4	GridLayout 管理器	285
12.4.5	GridBagLayout 管理器	287
12.5	事件处理	289
12.5.1	事件处理概述	289
12.5.2	事件类	290
12.5.3	事件监听器	291
12.6	实践案例：制作简易计算器	295
12.7	动手练一练	301
12.7.1	练习题	301
12.7.2	上机实践	302
第 13 章	数据库编程	303
13.1	JDBC 简介	304
13.1.1	JDBC 程序的工作原理	304
13.1.2	JDBC API 结构	305

13.1.3	JDBC 驱动	305
13.2	连接数据库的基本步骤	308
13.3	JDBC 应用	309
13.3.1	DriverManager 类	309
13.3.2	Statement 接口	310
13.3.3	PreparedStatement 接口	312
13.3.4	ResultSet 接口	314
13.4	实践案例：开发一套用户管理系统	317
13.5	动手练一练	324
13.5.1	练习题	324
13.5.2	上机实践	325
第 14 章	学生信息管理系统	326
14.1	系统介绍	327
14.1.1	需求分析	327
14.1.2	系统架构	327
14.2	数据库设计和实现	328
14.3	数据访问类	328
14.4	业务处理类	330
14.5	学生信息管理界面	333
14.5.1	学生信息添加界面	333
14.5.2	学生信息修改界面	335
14.5.3	主界面	338
14.6	总结	345
	练习题参考答案	346

第 1 章

Java 语言概述

Java 语言是一种简单的、跨平台的、面向对象的开发语言，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。Java 语言发展到今天，互联网的快速发展起了很大的推动作用。本章将对 Java 语言的发展、特点、运行环境等方面进行简单介绍。

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 的安装和配置
- 熟练掌握 Java API 的使用
- 熟练掌握如何开发 Java Application
- 熟练掌握如何开发 Java Applet
- 掌握使用 Eclipse 开发 Java 程序

1.1 Java 语言简介

作为一种跨平台的、面向对象、基于网络及安全性的网络编程语言，Java 越来越多地被用来编写各种各样的程序，如桌面应用程序、Web 程序和嵌入式程序。本节将向大家讲解 Java 语言的特点、历史和概念。

1.1.1 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的编程语言。它具有结构简单、面向对象、分布式、解释型、可靠性、安全性、平台无关、可移植、高性能、多线程、动态性等特性。以下对其重要特性进行简单介绍。

- 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，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。
- 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。
- 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。
- 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 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。

1.1.2 Java 语言的发展

Java 是 Sun 公司于 1991 年开发的新一代编程语言，最初目的是为家用消费类电子产品开发分布式代码系统，后来为了使整个系统与平台无关，该项目小组的领导人 James Gosling 决

定开发一种新语言，被称为 Oak，这就是 Java 语言的前身，后来改名为 Java。随着 Internet 的迅速发展，Web 的应用日益广泛，Java 语言也得到了迅速发展。1994 年，Gosling 用 Java 开发了一个实时性较高、可靠安全、有交互功能的新型 Web 浏览器，它不依赖于任何硬件平台和软件平台。这种新的浏览器被称为 HotJava。该浏览器 1995 年在业界发表，引起了巨大轰动，Java 的地位也随之得到肯定。

Java 语言发展非常迅猛。1995 年 3 月 Sun 公司发布了 Java 语言的 Alpha1.0a2 版本，1996 年 1 月发布了 Java 语言的第一个开发包 JDK v1.0；1997 年 2 月发布了 Java 语言的开发包 JDK v1.1，从而奠定了 Java 语言在计算机语言中的地位；1998 年 12 月 Sun 公司发布 Java 2 开发平台 Jdk v1.2，Java 2 平台是 Java 发展史上的里程碑；1999 年 6 月，Sun 公司重新组织 Java 平台的集成方法，并将企业级应用平台作为 Java 语言发展方向，包括了以下 3 个成员。

- J2ME (Java 2 Micro Edition): 用于嵌入式应用的 Java 2 平台。
- J2SE (Java 2 Standard Edition): 用于工作站、PC 机的 Java 2 标准平台。
- J2EE (Java 2 Enterprise Edition): 可扩展的企业级应用的 Java 2 平台。

Java 是面向对象的计算机语言。虽然许多面向对象的语言一开始就是严格的过程化语言，Java 却从一开始就被设计为面向对象的语言。

1.2 Java 语言的工作原理

Java 是一种跨平台的语言，同时也是既编译又解释的语言。Java 可以运行在不同平台的计算机上，但运行的计算机上必须装有相应的 Java 解释器，而这种解释器被称为 JVM。本节将详细介绍 Java 程序的运行流程，并简单地介绍 JVM 的应用。

1.2.1 Java 程序的运行流程

Java 可以编写各种各样的程序，如单机应用程序、Web 程序等。但不管哪种 Java 程序，其运行的流程大致如图 1-1 所示。

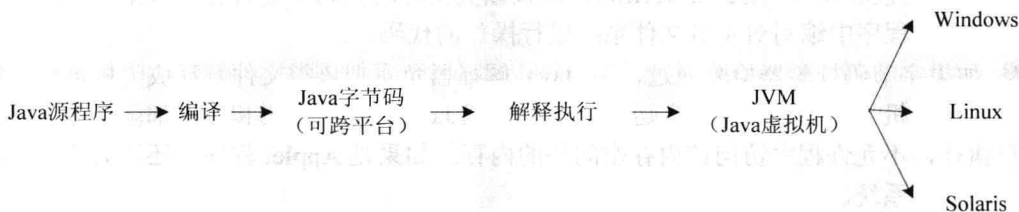


图 1-1 Java 程序的运行流程

从图 1-1 中可以观察到，首先应该创建源程序，Java 源程序一般是以 java 作为扩展名的文本文件，该文件用 Java 语言写成，可以用任何文本编辑器创建与编辑。源程序完成后，就可以进行编译，需要使用 Java 编译器 javac，读取 Java 源程序并翻译成 Java 虚拟机能够明白

的指令集合，且以字节码的形式保存在文件中。

通常，字节码文件以 `class` 作为扩展名，Java 解释器读取字节码，取出指令并翻译成计算机能执行的代码，完成运行过程。该字节码运行的平台是 Java 虚拟机，只要计算机上装有 Java 虚拟机，无论采用哪种操作系统，硬件配置情况如何，运行的结果都一样。

可以看出，Java 源程序首先被编译器编译成字节码文件，然后被 JVM 加载，解释执行。在此处 Java 虚拟机屏蔽了与具体操作系统平台相关的信息，使得 Java 语言编译程序只需生成在 Java 虚拟机上运行的目标代码（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执行。

1.2.2 JVM 介绍

Java 语言的一个非常重要的特点就是与平台的无关性。而使用 Java 虚拟机是实现这一特点的关键。Java Virtual Machine (Java 虚拟机)，是一个虚构出来的计算机，通过在实际的计算机上仿真模拟各种计算机功能来实现，可运行 Java 字节码的虚拟计算机系统。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执行，即在机器和编译程序之间加入了一层抽象的虚拟的机器。这台虚拟的机器在任何平台上都提供给编译程序一个共同的接口。编译程序只需要面向虚拟机，生成虚拟机能够理解的代码，然后由解释器来将虚拟机代码转换为特定系统的机器码执行。

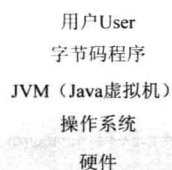


图 1-2 JVM 在计算机中的位置

中需要检验该类文件是否符合类文件规范。

Step 02 字节码校验器 (bytecode verifier) 检查该类文件的代码中是否存在某些非法操作，例如在 applet 程序中编写对本机文件系统进行操作的代码。

Step 03 如果字节码校验器检验通过，由 Java 解释器负责把该类文件解释成为机器码进行执行。Java 虚拟机采用的是“沙箱”运行模式，即把 Java 程序的代码和数据都限制在一定内存空间里执行，不允许程序访问该内存空间外的内存。如果是 Applet 程序，还不允许访问客户端机器的文件系统。

JVM 虚拟机解释执行的过程如图 1-3 所示。

Java 虚拟机的执行引擎有以下三种。

- 常用的解释器。
- JIT (Just-In-Time) 是一个可将 Java 字节码转换为本地机器码的代码生成器。由 JIT 调用的 Java 程序，其运行速度通常要比由解释程序执行字节码时的速度高得多。第一次执行的字节码会编译成本地的机器代码，被缓存在系统中，以后可以随时调用。