



全国高等职业教育“十三五”规划教材

Java程序设计案例教程

主编 许 敏 史荧中



电子课件、源代码下载网址 www.cmpedu.com

 **机械工业出版社**
CHINA MACHINE PRESS

全国高等职业教育“十三五”规划教材

Java 程序设计案例教程

主 编 许 敏 史 荧 中

参 编 李 萍 程 成

主 审 刘培林



机械工业出版社

本书共 12 章,第 1 章是 Java 语言概述;第 2~4 章讲述了 Java 语言基础,包括数据类型、常量和变量、运算符和表达式、简单的输入与输出、Java 程序的控制结构和数组;第 5、6 章讲述了面向对象程序设计,包括类的封装、继承和多态;第 7 章讲述了常用实用类,包括常用工具类和集合容器类等;第 8 章讲述了 Java 异常处理;第 9 章讲述了 Java 输入流和输出流;第 10、11 章讲述了图形用户界面设计和数据库编程技术;第 12 章讲述了多线程技术。

全书贯彻“理实一体化”的教学理念,以职工工资管理系统为载体,将项目开发分解为若干相对独立的工作任务。工作任务与相关理论知识交互配合,既是对理论知识的延伸与拓展,又是对理论知识掌握程度的检验。

本书可以作为高职高专院校计算机、电子等相关专业的教材,也可作为 Java 程序设计的入门语言教程来学习,还可作为从事计算机应用工作的工程技术人员培训和自学的参考书。

本书配有授课电子课件及源代码,需要的教师可登录 www.cmpedu.com 免费注册、审核通过后下载,或联系编辑索取(QQ: 1239258369, 电话: 010-88379739)。

图书在版编目(CIP)数据

Java 程序设计案例教程 / 许敏, 史荧中主编. —北京: 机械工业出版社, 2018.7

全国高等职业教育“十三五”规划教材

ISBN 978-7-111-60245-3

I. ①J… II. ①许… ②史… III. ①Java 语言—程序设计—
—高等职业教育—教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字(2018)第 133278 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

策划编辑: 鹿 征 责任编辑: 鹿 征

责任校对: 张艳霞 责任印制: 张 博

三河市宏达印刷有限公司印刷

2018 年 7 月第 1 版·第 1 次印刷

184mm×260mm·16.75 印张·409 千字

0001—3000 册

标准书号: ISBN 978-7-111-60245-3

定价: 49.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

网络服务

服务咨询热线: (010) 88379833

机工官网: www.cmpbook.com

读者购书热线: (010) 88379649

机工官博: weibo.com/cmp1952

教育服务网: www.cmpedu.com

封面无防伪标均为盗版

金书网: www.golden-book.com

全国高等职业教育“十三五”规划教材

计算机专业编委会成员名单

名誉主任 周智文

主 任 睦碧霞

副 主 任 林 东 王协瑞 张福强 陶书中
龚小勇 王 泰 李宏达 赵佩华
刘瑞新

委 员 (按姓氏笔画顺序)

万 钢	万雅静	卫振林	马 伟
王亚盛	尹敬齐	史宝会	宁 蒙
乔芑喆	刘本军	刘剑昀	齐 虹
江 南	安 进	孙修东	李华忠
李 萍	李 强	何万里	余永佳
张 欣	张洪斌	张瑞英	陈志峰
范美英	郎登何	赵国玲	赵增敏
胡国胜	钮文良	贺 平	顾正刚
徐义晗	徐立新	唐乾林	黄能耿
黄崇本	曹 毅	傅亚莉	裴有柱

秘 书 长 胡毓坚

出版说明

《国务院关于加强发展现代职业教育的决定》指出：到 2020 年，形成适应发展需求、产教深度融合、中职高职衔接、职业教育与普通教育相互沟通，体现终身教育理念，具有中国特色、世界水平的现代职业教育体系，推进人才培养模式创新，坚持校企合作、工学结合，强化教学、学习、实训相融合的教育教学活动，推行项目教学、案例教学、工作过程导向教学等教学模式，引导社会力量参与教学过程，共同开发课程和教材等教育资源。机械工业出版社组织全国 60 余所职业院校（其中大部分是示范性院校和骨干院校）的骨干教师共同策划、编写并出版的“全国高等职业教育规划教材”系列丛书，已历经十余年的积淀和发展，今后将更加紧密结合国家职业教育文件精神，致力于建设符合现代职业教育教学需求的教材体系，打造充分适应现代职业教育教学模式的、体现工学结合特点的新型精品化教材。

“全国高等职业教育规划教材”历经“十五”到“十三五”，涵盖计算机、电子和机电三个专业，目前在销教材 300 余种，其中“十五”“十一五”“十二五”累计获奖教材 60 余种，更有 4 种获得国家级精品教材。该系列教材依托于高职高专计算机、电子、机电三个专业编委会，充分体现职业院校教学改革和课程改革的需要，其内容和质量颇受授课教师的认可。

在系列教材策划和编写的过程中，主编院校通过编委会平台充分调研相关院校的专业课程体系，认真讨论课程教学大纲，积极听取相关专家意见，并融合教学中的实践经验，吸收职业教育改革成果，寻求企业合作，针对不同的课程性质采取差异化的编写策略。其中，核心基础课程的教材在保持扎实的理论基础的同时，增加实训和习题以及相关的多媒体配套资源；实践性较强的课程则强调理论与实训紧密结合，采用理实一体的编写模式；涉及实用技术的课程则在教材中引入了最新的知识、技术、工艺和方法，同时重视企业参与，吸纳来自企业的真实案例。此外，根据实际教学的需要对部分课程进行了整合和优化。

归纳起来，本系列教材具有以下特点：

- 1) 围绕培养学生的职业技能这条主线来设计教材的结构、内容和形式。
- 2) 合理安排基础知识和实践知识的比例。基础知识以“必需、够用”为度，强调专业技术应用能力的训练，适当增加实训环节。
- 3) 符合高职学生的学习特点和认知规律。对基本理论和方法的论述容易理解、清晰简洁，多用图表来表达信息；增加相关技术在生产中的应用实例，引导学生主动学习。
- 4) 教材内容紧随技术和经济的发展而更新，及时将新知识、新技术、新工艺和新案例等引入教材。同时注重吸收最新的教学理念，并积极支持新专业的教材建设。
- 5) 注重立体化教材建设。通过主教材、电子教案、配套素材光盘、实训指导和习题及解答等教学资源的有机结合，提高教学服务水平，为高素质技能型人才的培养创造良好的条件。

由于我国高等职业教育改革和发展的速度很快，加之我们的水平和经验有限，因此在教材的编写和出版过程中难免出现问题和疏漏。我们恳请使用这套教材的师生及时向我们反馈质量信息，以利于我们今后不断提高教材的出版质量，为广大师生提供更多、更适用的教材。

机械工业出版社

前 言

自 Java 语言诞生以来,一直受到业界的追捧,并逐渐成为主流的程序设计语言之一,特别是在企业级大型应用的开发方面占据了主导地位。通过对 Java 语言的学习,读者可以掌握 Java 程序设计语言的基本概念、基本语法和编程规范,理解面向对象的编程思想,掌握 Java API、异常处理、I/O 处理以及图形用户界面编程,了解多线程的概念,理解面向对象的程序设计的思路和方法,具有使用 Eclipse 进行 Java 程序的编程能力。本书具有以下特点:

(1) 本书知识介绍采用传统模式,确保逻辑性和易读性;最后通过工作任务实现知识、技能、方法三者的有机融合。

(2) 本书的编写,不仅仅是为了学习某种特定的语言,还融入了软件设计的思想,为后续 Java Web 开发、Java 框架编程等课程的学习奠定基础。

(3) 以遵循企业软件项目开发规范的完整案例贯穿全书内容,确保实践内容有完整策划。

① 从形式上看,工作任务是知识与技能的结合,每个工作任务都包含任务描述、相关知识、任务设计、任务实施、运行结果和任务小结 6 个完整的部分。

② 从内容及编排来看,工作任务源于真实项目的简化,与相应理论知识互为补充,难度上循序渐进,适于学习。

(4) 本书采用“任务驱动、案例教学”方法,突出实例与理论的紧密结合,通过典型案例由浅入深地介绍 Java 基础语法和面向对象程序设计,将知识讲解、技能训练和职业素质培养有机结合,融“教、学、做”三者于一体,适合采用“项目驱动、案例教学、理论实践一体化”等教学模式,以此进一步强化学生技能的培养。

(5) 每章后都有小结,并配有一定量的习题,便于教师教学和学生自学。各章内容充实,安排合理,衔接自然。

本书共 12 章,其中第 3、4、5、6、10、11 章由无锡职业技术学院许敏副教授编写,第 1、2、7、8 章由无锡职业技术学院史荧中副教授编写,第 9 章由无锡职业技术学院李萍副教授编写,第 12 章由中国船舶科学研究中心程成高级工程师编写。全书由许敏副教授统稿,无锡职业技术学院刘培林教授主审。在本书的编写过程中,参考了目前国内外优秀的有关 Java 程序设计方面的书籍资料,在此谨向有关作者表示感谢。

由于编者水平有限,书中难免会有疏漏和错误之处,请读者批评指正。

编 者

目 录

出版说明	
前言	
第 1 章 Java 语言概述	1
【引例描述】	1
【知识储备】	1
1.1 Java 语言的发展和特点	1
1.1.1 Java 语言的发展	1
1.1.2 Java 语言的特点	2
1.1.3 Java 与 C++ 的比较	3
1.1.4 Java 程序的工作机制	4
1.2 Java 开发环境的安装与配置	5
1.2.1 Java 开发环境的安装	5
1.2.2 Java 开发环境的配置	6
1.3 Java 程序设计举例	8
1.4 Eclipse 开发工具的安装及使用	9
1.4.1 Eclipse 的安装及内部架构	9
1.4.2 使用 Eclipse 开发应用程序	11
【任务实现】	12
工作任务 1 安装 JDK 并熟悉 Eclipse 开发环境	12
【本章小结】	13
【习题 1】	13
第 2 章 Java 语言基础	15
【引例描述】	15
【知识储备】	15
2.1 标识符、关键字和分隔符	15
2.1.1 Java 标识符	15
2.1.2 关键字	16
2.1.3 分隔符	16
2.1.4 代码注释	16
2.2 数据类型	17
2.2.1 数据类型的划分	17
2.2.2 基本数据类型	17
2.3 常量和变量	20
2.3.1 常量	20

2.3.2	变量	20
2.4	运算符和表达式	21
2.4.1	算术运算符	21
2.4.2	关系运算符	22
2.4.3	布尔运算符	23
2.4.4	位运算符	24
2.4.5	赋值运算符	24
2.4.6	条件运算符	25
2.4.7	运算符优先级	25
2.4.8	表达式与语句	26
2.5	简单的输入与输出	27
2.5.1	输出	27
2.5.2	输入	28
	【任务实现】	29
	工作任务 2 工资计算器界面设计	29
	【本章小结】	31
	【习题 2】	31
第 3 章	Java 程序的控制结构	33
	【引例描述】	33
	【知识储备】	33
3.1	结构化程序设计的 3 种基本结构	33
3.2	顺序语句	34
3.3	分支语句	34
3.3.1	if 语句	34
3.3.2	switch 语句	38
3.4	循环语句	41
3.4.1	while 语句	41
3.4.2	do...while 语句	42
3.4.3	for 语句	43
3.5	跳转语句	46
3.5.1	break 语句	46
3.5.2	continue 语句	46
	【任务实现】	47
	工作任务 3 职工工资管理系统用户类型选择	47
	工作任务 4 职工工资计算器	49
	【本章小结】	50
	【习题 3】	50
第 4 章	数组	53
	【引例描述】	53

【知识储备】	53
4.1 一维数组的定义与使用	53
4.1.1 一维数组的定义	53
4.1.2 一维数组的初始化及内存分配	54
4.1.3 一维数组的引用	55
4.2 多维数组的定义与使用	57
4.2.1 二维数组的定义	57
4.2.2 二维数组的初始化及内存分配	58
4.2.3 二维数组的引用	59
4.3 数组的常用方法	62
【任务实现】	63
工作任务 5 多职工工资计算器	63
【本章小结】	65
【习题 4】	65
第 5 章 类与对象	68
【引例描述】	68
【知识储备】	68
5.1 面向对象的基本概念	68
5.1.1 面向对象的术语	68
5.1.2 面向对象的基本特征	69
5.2 类的封装	69
5.3 成员访问权限	76
5.3.1 访问控制修饰符	76
5.3.2 static 修饰符	78
5.3.3 final 和 abstract 修饰符	81
5.4 内部类	81
5.4.1 成员类	81
5.4.2 静态成员类	82
5.4.3 局部类	82
5.4.4 匿名类	82
【任务实现】	83
工作任务 6 职工类设计	83
【本章小结】	87
【习题 5】	87
第 6 章 继承和多态	89
【引例描述】	89
【知识储备】	89
6.1 类的继承性	89
6.1.1 继承的概念	89

6.1.2 继承的实现	89
6.2 继承的规则	92
6.2.1 成员变量的继承	92
6.2.2 成员方法的继承	93
6.2.3 this 和 super 关键字	95
6.3 抽象类和最终类	95
6.3.1 抽象类和抽象方法	96
6.3.2 最终类和最终方法	98
6.4 类对象之间的类型转换	98
6.5 接口	100
6.5.1 接口的声明	100
6.5.2 接口的实现	100
6.5.3 接口的继承	101
6.5.4 接口的特点	103
6.6 类的多态	103
【任务实现】	105
工作任务 7 职工工资类设计	105
工作任务 8 添加职工信息程序设计	107
【本章小结】	110
【习题 6】	110
第 7 章 常用实用类	114
【引例描述】	114
【知识储备】	114
7.1 常用工具类	114
7.1.1 Math 类	114
7.1.2 String 类	116
7.1.3 StringBuffer 类	118
7.1.4 Date 类	120
7.1.5 Calendar 类	120
7.2 Java 集合容器	121
7.3 Collection 接口	122
7.3.1 Iterator 接口	123
7.3.2 foreach 循环	123
7.4 List 接口	124
7.5 Set 接口	125
7.6 Map 接口	127
【任务实现】	128
工作任务 9 使用常用工具类设计添加职工工资信息	128
工作任务 10 使用集合类添加职工工资信息 (Collection)	133

工作任务 11 修改职工工资信息 (List)	136
工作任务 12 删除职工工资信息 (Set)	139
工作任务 13 查找职工工资信息 (Map)	141
【本章小结】	146
【习题 7】	146
第 8 章 异常处理	148
【引例描述】	148
【知识储备】	148
8.1 Java 异常简介	148
8.1.1 Java 异常	148
8.1.2 几种常见的异常	148
8.1.3 常见 Java 异常类列表	149
8.2 异常机制	150
8.2.1 捕获异常	151
8.2.2 使用 throw 抛出异常	153
8.2.3 使用 throws 声明抛出异常	153
8.3 自定义异常	154
【任务实现】	156
工作任务 14 添加职工工资信息数据校验并提示校验结果	156
【本章小结】	161
【习题 8】	161
第 9 章 输入与输出	163
【引例描述】	163
【知识储备】	163
9.1 数据流的基本概念	163
9.1.1 输入/输出流	163
9.1.2 输入/输出类	164
9.1.3 标准输入/输出	165
9.2 字节输入/输出流	166
9.2.1 文件输入/输出字节流	166
9.2.2 过滤流	169
9.3 字符输入/输出流	173
9.3.1 输入/输出字符流	173
9.3.2 文件输入/输出字符流	174
9.3.3 缓冲字符流	176
9.4 文件处理	177
9.4.1 文件类的使用	177
9.4.2 文件的顺序访问	180
9.4.3 文件的随机访问	180

9.5 对象的串行化	181
9.5.1 串行化概述	181
9.5.2 对象串行化的实现	182
9.5.3 串行化的注意事项	184
【任务实现】	184
工作任务 15 职工工资管理数据持久化	184
【本章小结】	189
【习题 9】	189
第 10 章 图形用户界面设计	191
【引例描述】	191
【知识储备】	191
10.1 GUI 介绍	191
10.2 容器	192
10.2.1 框架 (JFrame)	193
10.2.2 面板 (JPanel)	193
10.3 布局管理	194
10.3.1 FlowLayout 布局管理器	194
10.3.2 BorderLayout 布局管理器	195
10.3.3 GridLayout 布局管理器	196
10.4 组件	197
10.4.1 标签 (JLabel)	197
10.4.2 按钮 (JButton)	197
10.4.3 文本框 (JText)	198
10.4.4 复选框 (JCheckBox)	200
10.4.5 单选按钮 (JRadioButton)	201
10.4.6 列表框 (JList)	202
10.4.7 组合框 (JComboBox)	202
10.5 事件处理	204
10.5.1 事件处理介绍	204
10.5.2 使用内部类和匿名类作为监听器	205
10.5.3 事件处理模型	206
10.5.4 事件、监听器和适配器	207
【任务实现】	211
工作任务 16 用户登录界面设计	211
工作任务 17 职工工资录入界面	212
工作任务 18 职工工资管理实现	216
【本章小结】	221
【习题 10】	221
第 11 章 数据库编程	223

【引例描述】	223
【知识储备】	223
11.1 数据库编程概述	223
11.1.1 JDBC 概述	223
11.1.2 MySQL 介绍	223
11.2 数据库访问流程	226
11.2.1 访问数据库步骤	226
11.2.2 连接数据库	226
11.2.3 执行 SQL 语句	229
11.2.4 获得查询结果	231
11.2.5 关闭连接	233
【任务实现】	234
工作任务 19 职工工资管理实现（数据库编程）	234
【本章小结】	241
【习题 11】	241
第 12 章 多线程	243
【引例描述】	243
【知识储备】	243
12.1 线程概述	243
12.1.1 进程	243
12.1.2 线程	243
12.2 多线程的实现方法	244
12.2.1 继承 Thread 类	244
12.2.2 实现 Runnable 接口	245
12.2.3 两种实现方法的比较	246
12.3 线程的状态控制	246
12.4 线程的同步	248
12.5 线程的通信	250
【任务实现】	252
工作任务 20 时钟	252
【本章小结】	254
【习题 12】	254
参考文献	256

第1章 Java 语言概述

【引例描述】

➤ 问题提出

读者选择 Java 语言作为开发工具，开发一个“职工工资管理”程序。开发 Java 应用程序，首先必须提供 Java 的开发环境，即安装 JDK。如何安装 Java 开发工具包，配置系统的环境变量？如何使用 Eclipse 开发环境？

➤ 解决方案

本章介绍 Java 语言的发展历史、主要特点及工作机制，并将 Java 语言与 C++ 进行了比较；讲授 Java 语言开发环境的安装和使用，并学习 Java 应用程序的编写、Eclipse 集成开发工具的使用等入门知识。

通过本章学习，读者可掌握 Java 语言的特点与工作机制，能安装使用 Java 开发平台并编写第一个 Java 应用程序。

【知识储备】

1.1 Java 语言的发展和特点

1.1.1 Java 语言的发展

Java 语言诞生于 1991 年，它是 Sun 公司（后被 Oracle 公司收购）为一些消费性电子产品所设计的，目的是开发一个新的语言，可以对电冰箱、电视机、电子游戏机等家用电器进行编程控制，和设备进行信息交流。鉴于这些电子产品有一个共同的特点：计算处理能力和内存都非常有限，因此要求：①该语言必须非常小且能生成非常紧凑的代码，这样才能在上述环境中执行。②由于不同的厂商选择不同的 CPU，所以要求该语言不能和特定的体系结构绑在一起，要求语言本身是中立的，也就是跨平台的。因此，James Gosling 领导的 Green 项目小组创建了新的程序设计语言——Oak 语言（Java 语言的前身），保留了大部分与 C++ 相似的语法，改进了 C++ 中过于复杂或具有危险性的特性。Oak 是一种可移植性语言，也就是一种与平台不相关的语言，能够在各种芯片上运行，这样各家厂商就可以降低研发成本，直接把应用程序应用在自家的产品上。

1994 年，Oak 的技术日趋成熟，Internet 正在蓬勃发展。用户迫切希望能够在网络上创建一类无须考虑软、硬件平台就可以执行的应用程序，并且这些程序还要有极大的安全保障，正是由于这种需求给 Oak 带来了前所未有的施展舞台。1995 年 5 月 23 日，Oak 语言改

名为 Java (Oak 商标已被注册), 并且在 Sun World 大会上正式发布了 Java 和 HotJava 浏览器。由于 Java 只是一种编程语言, 如果想要开发复杂的应用程序, 需要有一个强大的开发库支持, 因此, Sun 公司在 1996 年 1 月 23 日发布了 JDK1.0 版本。Java 语言第一次提出了“Write Once, Run Anywhere”的口号, JDK 被发布后立即引起极大的下载量, Java 成为网络编程的主流语言之一。Java 语言的发展历史如下:

- 1996 年 1 月, Sun 发布了 JDK 1.0;
- 1997 年 2 月, Sun 发布了 JDK 1.1;
- 1998 年 12 月, Sun 发布了 JDK 1.2 (Java 2 平台);
- 1999 年 6 月, Sun 重新定义 Java 技术架构, 并将 Java 2 平台分为 3 个版本: 标准版 (J2SE)、企业版 (J2EE) 和微缩版 (J2ME);
- 2000 年 5 月, Sun 分别发布了 JDK 1.3 和 1.4;
- 2004 年 9 月, JDK 1.5 发布, 成为 Java 语言发展史上又一里程碑。为了表示该版本的重要性, JDK 1.5 更名为 JDK 5; 此时, Java 的各种版本已经更名, 取消其中的数字“2”: J2EE 更名为 Java EE, J2SE 更名为 Java SE, J2ME 更名为 Java ME;
- 2006 年 12 月, Sun 公司发布 JDK 6.0;
- 2010 年 9 月, 发布 JDK 7.0, 增加了简单闭包功能;
- 2014 年 3 月, 发布 JDK 8.0, 增加了 Lambda 表达式等新特性;
- 2017 年 9 月, 发布 JDK 9.0, 增加了模块化源代码等新特性。

1.1.2 Java 语言的特点

Java 语言是一种高级的、通用的、面向对象的程序设计语言。其语法与 C 或 C++类似, 但在组织结构上截然不同。它是一种完全面向对象的程序语言, 程序的基本处理单位是类。由于当初在设计 Java 的时候, 倾向于把它设计成一种具有生产力的语言, 而不仅仅是研究性的语言, 因此在学习运用 Java 语言的时候, 能很快感受到它的便利与强大功能。Java 语言有如下特点。

1. 简单的特性

Java 语言的语法与 C 和 C++很接近, 使得大多数程序员很容易学习和使用。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, 如操作符重载、多继承、自动的强制类型转换等。Java 语言不使用指针, 并提供了自动的垃圾收集, 使得程序员不必为内存管理而担忧。

2. 面向对象的特性

面向对象是 Java 语言最重要的特性。Java 语言提供类、接口和继承等原语, 为了简单起见, 只支持类之间的单继承, 但支持接口之间的多继承, 并支持类与接口之间的实现机制。Java 全面支持动态绑定, 而 C++只对虚函数使用动态绑定。总之, Java 是一个纯粹的面向对象程序设计语言。

3. 分布式处理的特性

Java 支持 Internet 应用的开发, 在基本的 Java 应用编程接口中有一个网络应用编程接口, 它提供了用于网络应用编程的类库, 包括 URL、URLConnection、Socket、ServerSocket 等。Java 应用程序可通过一个特定的 URL 来打开并访问对象, 就像访问本地文件系统一样简单方便。

4. 健壮特性

Java 的强类型机制、异常处理、垃圾的自动收集等是 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。

5. 结构中立的特性

Java 程序（后缀为 java 的文件）在 Java 平台上被编译为体系结构中立的字节码格式（后缀为 class 的文件），只要安装了 Java 运行时系统，Java 程序就可以在任意的处理器上运行。

6. 安全特性

Java 的安全性可从 4 个方面得到保证。

1) Java 语言自身提供的安全。在 Java 语言里，指针、释放内存等 C++ 中的功能被删除，避免了非法内存操作。

2) 编译器提供的安全。当 Java 用来创建浏览器时，语言功能和一些浏览器本身提供的功能结合起来，使它更安全。Java 代码在机器上执行前，要经过很多次的测试。它需要通过代码校验、检测指针操作、检测是否正改变一个对象的类型等测试。

3) 字节码校验。如果字节代码通过代码检验没有返回错误，可以确定代码没有堆栈上溢出和下溢出，所有操作代码参数类型都是正确的，并且没有发生非法数据转换。

4) 类装载。类装载通过本机类与网络资源类的名称分开，来保持安全性，通过分配不同的命名空间以防替代本地的同名类，以避免特洛伊木马病毒的出现。

7. 可移植的特性

这种可移植性来源于体系结构中立性，另外，Java 还严格规定了各个基本数据类型的长度。Java 系统本身 also 具有很强的可移植性，Java 编译器是用 Java 实现的，解释器是用标准 C 实现的。

8. 解释的特性

Java 编译器将 Java 源文件生成类文件，扩展名为.class，类文件可通过 java 命令加载、解释、执行，将 Java 代码转换为机器可执行代码。Java 解释器能直接运行目标代码指令。

9. 高性能的特性

与那些解释型的高级脚本语言相比，Java 是高性能的。事实上，Java 的运行速度随着 JIT（Just-In-Time）编译器技术的发展越来越接近于 C++。

10. 多线程的特性

Java 语言内置支持多线程功能，使得在一个程序里可同时执行多个小任务。多线程带来的更大的好处是更好的交互性能和实时控制性能。

11. 动态的特性

Java 语言的设计目标之一是适应于动态变化的环境。Java 程序需要的类能够动态地被载入到运行环境，也可以通过网络来载入所需要的类。

1.1.3 Java 与 C++ 的比较

Java 提供了一个功能强大语言的所有功能，并且几乎没有一点含混特征。C++ 安全性不高，但 C 和 C++ 有着大量的用户，因此 Java 设计成与 C++ 相似的形式，让大家容易学习。Java 去掉了 C++ 语言的许多功能，让 Java 语言很精炼，并增加了一些很有用的功能，因此 Java 和 C++ 之间存在着一些显著的差异。事实上，这些差异正是技术进步的表现，因此读者需要特别关注这些差异。Java 和 C++ 之间的差异可见表 1-1。

表 1-1 Java 语言与 C++语言的比较

项 目	Java 语言	C++语言
全局变量	不存在全局变量	存在
数据类型	整数类型无符号的区别	整数类型有正负之分
类型转换	转换时进行兼容性检查，安全性好	通过指针可进行任意的类型转换
范围运算符	无	有作用域运算符“::”
数组	通过对象操作，有 length 属性	通过指针操作，无 length 属性
内存管理	内存使用后不必回收，有自动垃圾回收机制	内存使用后必须回收，否则造成内存崩溃
提前声明	不必提前声明	需先声明后使用
预处理	无	有预处理机制
头文件	import 包	include 头文件，维护相对困难
指针	无	有，功能强，但带来安全问题
goto 关键字	作为保留的关键字，无作用	有，可以使用
virtual 关键字	无	有
结构、联合和枚举	不支持	有
注释文档	提供对注释文档的支持	无
多重继承	不允许，但允许一个类实现多个接口	允许
操作符重载	不支持	支持
字符串	支持字符串变量	不支持字符串变量

1.1.4 Java 程序的工作机制

Java 语言的核心设计理念是跨平台 and 安全性，为此，Java 发展了 Java 虚拟机、Java 字节码和垃圾回收机制三大核心技术。

1. Java 虚拟机

前文已经提到过，平台无关性是 Java 最重要的特性，而实现这一特性的基础就是 Java 虚拟机（Java Virtual Machine, JVM）。从底层看，Java 虚拟机是以 Java 字节码为指令组的软 CPU。图 1-1 显示了 Java 程序运行过程。从图中可以看出，在服务器端，首先由开发人员编写 Java 源程序并存储为.java 文件；其次，Java 编译器将.java 文件编译成字节码并保存为.class 文件；最后将.class 文件存放在 Web 服务器上。在客户端，用户访问服务器端的主页，下载 Java 程序，再依赖本地 Java 虚拟机对.class 文件解释执行。

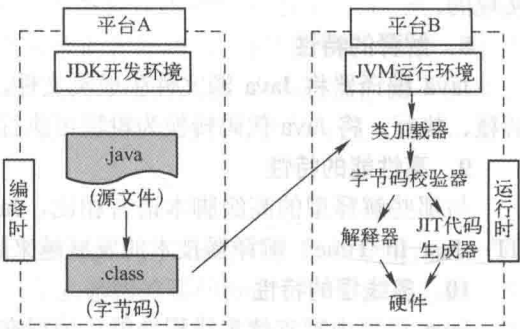


图 1-1 Java 程序运行过程

Java 虚拟机包含类加载器、字节码校验器和 JIT 实时编译器。类加载器用来取得从网络获取的或存于本地机器上的类文件字节码。下一步由字节码校验器检查这些类文件是否有正确的格式，以确认在运行时不会有破坏内存的行为。Java 解释器将字节码解释翻译成机器码，而 JIT 编译器也可将字节码转为本地机器码，但它可使原本是解释执行方式的虚拟机提高到编译式的运行效率。

2. Java 字节码

Java 源程序通过 Java 编译器编译后的产品是字节码文件（.class），与机器码不同，它不