

从基础  
到实践

基础内容完整覆盖  
实践应用循序渐进

从必需  
到拓展

必需知识全面够用  
拓展材料精心筛选

从入门  
到进阶

内容编排由浅入深  
知识面广词汇量大

重点  
推荐

# Java程序设计

## 案例教程

■ 邢海燕 陈静 卜令瑞◎主编



北京理工大学出版社  
BEIJING INSTITUTE OF TECHNOLOGY PRESS

# Java 程序设计案例教程

主 编 邢海燕 陈 静 卜令瑞  
副主编 靳晓娟 牟艳霞 李 朋  
孙艳华 李 兵

 **北京理工大学出版社**  
BEIJING INSTITUTE OF TECHNOLOGY PRESS

## 内 容 简 介

Java 是计算机类相关专业的编程入门语言，本书从 Java 语言的安装环境、基础语法、控制结构到面向对象编程、图形用户界面、输入/输出、数据库编程和网络编程，一步步深入，由简单到复杂。本书的每一部分内容都以学习任务为导向，涵盖了各个知识点。通过任务实施，学员可以巩固前面所学的知识和技术，积累项目开发经验。

本书通俗易懂，对图示、代码几乎都加了注释，帮助读者快速理解；提供大量的案例，以增强读者的动手能力，激发学习兴趣。本书提供电子教案、课件、源代码、习题及答案，为教师授课和学生学习提供便利。

本书既可以用作计算机相关专业的程序设计课程教材，也可以用作 Java 技术基础的培训教材，并且是一本适合广大计算机编程初学者学习的入门级读物。

版权专有 侵权必究

---

### 图书在版编目 (CIP) 数据

Java 程序设计案例教程/邢海燕，陈静，卜令瑞主编. —北京：北京理工大学出版社，2021.4

ISBN 978 - 7 - 5682 - 9460 - 7

I. ①J… II. ①邢…②陈…③卜… III. ①JAVA 语言 - 程序设计 - 教材  
IV. ①TP312.8

中国版本图书馆 CIP 数据核字 (2021) 第 005049 号

---

出版发行 / 北京理工大学出版社有限责任公司

社 址 / 北京市海淀区中关村南大街 5 号

邮 编 / 100081

电 话 / (010) 68914775 (总编室)

(010) 82562903 (教材售后服务热线)

(010) 68948351 (其他图书服务热线)

网 址 / <http://www.bitpress.com.cn>

经 销 / 全国各地新华书店

印 刷 / 三河市天利华印刷装订有限公司

开 本 / 787 毫米 × 1092 毫米 1/16

印 张 / 21

字 数 / 462 千字

版 次 / 2021 年 4 月第 1 版 2021 年 4 月第 1 次印刷

定 价 / 89.00 元

责任编辑 / 王玲玲

文案编辑 / 王玲玲

责任校对 / 周瑞红

责任印制 / 施胜娟

---

图书出现印装质量问题，请拨打售后服务热线，本社负责调换

Java 语言于 20 世纪 90 年代初期诞生，伴随着计算机平台的多样化及互联网的迅猛发展而发展，逐渐成为重要的网络编程语言。Java 语言是一种面向对象的程序设计语言，它的风格和 C++ 语言十分接近，换句话说，它继承了 C++ 语言面向对象技术的核心，但舍弃了 C++ 语言中容易产生错误的部分。Java 语言具有面向对象、分布式、解释性、健壮性、安全性、跨平台、可移植性、高性能、多线程等重要技术特性。

本书共分 9 个模块，介绍了 Java 程序开发环境的搭建及 Java 程序的开发流程、Java 语言的基本语法规则、面向对象的编程基础和进阶、Java 图形用户界面编程、Java 多线程和异常处理机制、Java 的输入/输出流、Java 数据库编程和 Java 网络编程等知识。

每一个模块的内容都由案例引入，根据案例需求进行知识讲解，并配有相应的实训，逐步增加难度和复杂度。内容的选取以理论知识必需、够用为度，突出具有针对性和实用性的职业能力训练项目，侧重于培养读者使用 Java 语言进行面向对象程序设计的基本技能，而不是对 Java 技术进行百科全书式的介绍。

本书主要特色如下：

(1) 每一个模块中的内容都由案例引入。每个模块除了引入的案例外，另有若干针对性和实用性强的职业能力训练项目。

(2) 本书包含丰富的案例，图文并茂，突出实践，以达到活学活用的目的。本书采用的案例及训练项目都经过严格筛选，并经由企业人士指导。

本书提供电子教案、课件、源代码、习题及答案，为教师授课和学生学习提供便利。本书可以为后续的 Java 高级开发及其他编程语言的学习打下良好的基础。本书既可以用作计算机相关专业的程序设计课程教材，也可以用作 Java 技术基础的培训教材，并且是一本适合广大计算机编程初学者学习的入门级读物。

本书由山东劳动职业技术学院、德州职业技术学院、山东师范大学和联想集团合作完成。本书由邢海燕、陈静、卜令瑞主编，靳晓娟、牟艳霞（德州职业技术学院）、李朋（山东师范大学）、孙艳华、李兵（联想集团）副主编。其中，模块 1 由陈静编写，模块 2 和模块 9 由靳晓娟编写，模块 3 和模块 5 由李朋编写，模块 4 由邢海燕编写，模块 6 由卜令瑞编写，模块 7 由孙艳华编写，模块 8 由牟艳霞和李兵编写。邢海燕负责全书的统稿工作。

由于编者水平有限，书中难免存在疏漏之处，欢迎广大读者批评指正。

# Contents

# 目录

模块 1 Java 开发环境搭建	1
任务 1 Java 概述	1
任务 2 搭建开发环境	5
任务 3 第一个 Java 程序	15
习题	25
模块 2 Java 语言基础	26
任务 1 求圆的面积和周长案例	26
任务 2 判断大小写字母案例	42
任务 3 数字排序案例	54
习题	69
模块 3 面向对象编程基础	72
任务 1 定义名为 Student 的学生类	72
任务 2 计算长方形的面积	78
任务 3 Teacher 教师类	91
习题	100
模块 4 面向对象编程进阶	103
任务 1 动物类的继承	103
任务 2 形状类和矩形类	125
任务 3 计算器	137
习题	149
模块 5 Java 图形用户界面开发	152
任务 1 HelloWorld 窗体和对话框	152
任务 2 用户注册界面设计	157
任务 3 简单计算器	192
习题	203
模块 6 Java 多线程与异常处理	204
任务 1 移动文字与改变颜色案例	204
任务 2 银行存取款案例	214

任务 3 数组越界和除数为零异常案例 .....	224
习题 .....	230
模块 7 输入/输出流 .....	232
任务 1 文件管理操作 .....	232
任务 2 文件编辑器 .....	239
任务 3 文件复制 .....	251
习题 .....	258
模块 8 数据库编程 .....	260
任务 学生信息管理系统 .....	260
习题 .....	305
模块 9 Java 网络编程 .....	307
任务 1 通过 URL 类访问网络资源案例 .....	307
任务 2 基于 TCP 协议的网络通信案例 .....	314
任务 3 基于 UDP 协议的网络通信案例 .....	321
习题 .....	326

# 模块 1

## Java 开发环境搭建

### 【模块教学目标】

- 掌握 Java 的开发环境的搭建、Java 平台的构成
- 掌握 Java 程序的两种开发方法、开发过程及各自特点
- 了解 Java 语言的特点

## 任务 1 Java 概述

### 导入任务

了解 Java 平台的架构；了解 Java 语言的特点；了解 Java 的应用。

### 知识准备

#### 一、Java 简介

Java 是一种高级的面向对象的程序设计语言。Java 既安全、可移植，又可跨平台，并且人们发现它能够解决 Internet 上的大型应用问题，从 PC 机到手机上都有 Java 开发的程序和游戏。1991 年，Sun 公司的 James Gosling 等开始开发名称为 Oak 的语言，目标定位在家用电器等小型系统的程序语言，主要应用于电视机、电话、闹钟、烤面包机等家用电器的控制和通信。由于这些智能化家电的市场需求没有预期的高，Sun 公司放弃了该项计划。随着互联网的发展，Sun 公司看见 Oak 在互联网上的应用前景，于是改造了 Oak，于 1995 年 5 月以 Java 的名称正式发布。Java 伴随着互联网的迅猛发展而发展，逐渐成为重要的网络编程语言。

Java 编程语言的风格十分接近 C++ 语言。继承了 C++ 语言面向对象技术的核心，Java 舍弃了 C++ 语言中容易引起错误的指针，改为引用替换，同时移除原 C++ 与原来运算符重

载特性，也移除多重继承特性，改用接口替换，增加垃圾回收器功能。在 Java SE 1.5 版本中引入了泛型编程、类型安全的枚举、不定长参数和自动装/拆箱特性。Sun 公司对 Java 语言的解释是：“Java 编程语言是个简单、面向对象、分布式、解释性、健壮、安全与系统无关、可移植、高性能、多线程和动态的语言。”

Java 不同于一般的编译语言或直译语言。它首先将源代码编译成字节码，然后依赖各种不同平台上的虚拟机来解释执行字节码，从而实现了“一次编写，到处运行”的跨平台特性。在早期 JVM 中，这在一定程度上降低了 Java 程序的运行效率。但在 J2SE1.4.2 发布后，Java 的运行速度有了大幅提升。与传统形态不同，Sun 公司在推出 Java 时就将其作为开放的技术。全球数以万计的 Java 开发公司被要求所设计的 Java 软件必须相互兼容。“Java 语言靠群体的力量而非公司的力量”是 Sun 公司的口号之一，并获得了广大软件开发商的认同。这与微软公司所倡导的注重精英和封闭式的模式完全不同，此外，微软公司后来推出了与之竞争的 .NET 平台及模仿 Java 的 C# 语言。后来 Sun 公司被甲骨文公司并购，Java 也随之成为甲骨文公司的产品。

## 二、Java 语言的特点

Java 语言的特点主要表现在：简单、面向对象、自动的内存管理、分布计算、健壮性、安全性、解释执行、跨平台、多线程及异常处理、动态性等方面。

### 1. 简单

由于 Java 的结构类似于 C 和 C++，所以熟悉 C 与 C++ 语言的编程人员稍加学习就很容易掌握 Java 的编程技术了。出于安全、稳定性的考虑，Java 语言去除了 C++ 中一些不容易理解又容易出错的部分，如指针。Java 所具有的自动内存管理机制也大大简化了 Java 程序设计开发。

### 2. 面向对象

Java 语言是一种新的面向对象的程序设计语言，它除了几种基本的数据类型外，大都是类似 C++ 中的对象和方法，程序代码大多体现了类机制，以类的形式组织，由类来定义对象的各种行为。

Java 提供了简单的类机制和动态的构架模型，对象中封装了它的状态变量和方法（函数、过程），实现了模块化和信息隐藏；而类则提供了一类对象的原型，通过继承和多态机制，子类可以使用或者重新定义父类或者超类所提供的过程，从而实现代码的复用。

### 3. 自动内存管理

Java 的自动垃圾回收（auto garbage collection）实现了内存的自动管理，因此简化了 Java 程序开发的工作，早期的垃圾回收（garbage collection, GC）对系统资源抢占太多，从而影响整个系统的运行，Java2 对 GC 进行的改良使 Java 的效率有了很大提高。GC 的工作机制是周期性地自动回收无用存储单元。Java 的自动内存回收机制简化程序开发的同时，提高了程序的稳定性和可靠性。

### 4. 分布计算

Java 为程序开发提供了 java.net 包，该包提供了一组使程序开发者可以轻易实现基于 TCP/IP 的分布式应用系统。此外，Java 还提供了专门针对互联网应用的类库，如 URL、

Java mail 等。

### 5. 健壮性

Java 语言的设计目标之一就是编写高可靠性的软件。Java 语言提供了编译时检查和运行时检查，用户可以满怀信心地编写 Java 代码，在开发过程中系统将会发现很多错误，不至于将错误推迟到产品发布时才发现。

### 6. 安全性

Java 的设计目的是提供一个用于网络/分布式的计算环境。因此，Java 强调安全性，例如确保无病毒、小应用程序运行安全控制等。Java 的验证技术是以公钥（public-key）加密算法为基础，并且从环境变量、类加载器、文件系统、网络资源和名字空间等方面实施安全策略。

### 7. 解释执行

Java 解释器（interpreter）可以直接在任何已移植的解释器的机器上解释、执行 Java 字节码，不需重新编译。当然，其版本向上兼容，因此，如果是高版本环境下编译的 Java 字节码，到低版本环境下运行也许会有部分问题。

### 8. 平台无关性

Java 是网络空间的“世界语”，编译后的 Java 字节码可以在所有提供 Java 虚拟机（JVM）的多种不同主机、不同处理器上运行。“一次编写，到处运行”也许是 Java 最诱人的特点。用 Java 开发而成的系统，其移植工作几乎为零，一般情况下只需对配置文件、批处理文件做相应修改即可实现平滑移植。

### 9. 多线程

Java 的多线程（multithreading）机制使程序可以并行运行。同步机制保证了对共享数据的正确操作。多线程使程序设计者可以用不同的线程分别实现各种不同的行为，例如，用户在 WWW 浏览器中浏览网页还可以听音乐，后台浏览器又可以同时下载图像，因此，使用 Java 语言可以非常轻松地实现网络上的实时交互行为。

### 10. 异常处理

C 语言程序员大都使用 goto 语句来做条件跳转，Java 编程中不支持 goto 语句。Java 采用异常模型使程序的主流逻辑变得更加清晰明了，并且能够简化错误处理工作。

### 11. 动态性

Java 语言的设计目标之一是适应动态变化的环境。Java 在类库中可以自由地加入新方法和实例变量，而不影响用户程序的执行。Java 通过接口来支持多重继承，使其具有更灵活的方式和扩展性。

## 三、Java 语言的应用

Java 介于编译型语言和解释型语言之间。编译型语言如 C、C++，代码是直接编译成机器码执行，但是不同的平台（x86、ARM 等）CPU 的指令集不同，因此，需要编译出每一种平台的对应机器码。解释型语言如 Python、Ruby 没有这个问题，可以由解释器直接加载源码然后运行，代价是运行效率太低。而 Java 是将代码编译成一种“字节码”，它类似于抽象的 CPU 指令，然后针对不同平台编写虚拟机，不同平台的虚拟机负责加载字节码并执行，

这样就实现了“一次编写，到处运行”的效果。当然，这是针对 Java 开发者而言的，对于虚拟机，需要为每个平台分别开发。为了保证不同平台、不同公司开发的虚拟机都能正确执行 Java 字节码，Sun 公司制定了一系列的 Java 虚拟机规范。从实践的角度看，JVM 的兼容性做得非常好，低版本的 Java 字节码完全可以正常运行在高版本的 JVM 上。

随着 Java 的广泛应用，Java 分为三个体系 Java SE (Java 2 Platform Standard Edition, Java 平台标准版)，Java EE (Java 2 Platform Enterprise Edition, Java 平台企业版)，Java ME (Java 2 Platform Micro Edition, Java 平台微型版)。

**Java SE:** Java SE 以前称为 J2SE。它是允许开发和部署在桌面、服务器、嵌入式环境和实时环境中使用的 Java 应用程序。Java SE 包含了支持 Java Web 服务开发的类，并为 Java EE 提供基础。

**Java EE:** 这个版本以前称为 J2EE。企业版本帮助开发和部署可移植、健壮、可伸缩且安全的服务器端 Java 应用程序。Java EE 是在 Java SE 的基础上构建的，它提供 Web 服务、组件模型、管理和通信 API，可以用来实现企业级的面向服务体系结构 (service-oriented architecture, SOA) 和 Web 2.0 应用程序。

**Java ME:** 这个版本以前称为 J2ME。Java ME 为在移动设备和嵌入式设备 (比如手机、PDA、电视机顶盒和打印机) 上运行的应用程序提供一个健壮且灵活的环境。Java ME 包括灵活的用户界面、健壮的安全模型、许多内置的网络协议及对可以动态下载的连接和离线应用程序的丰富支持。基于 Java ME 规范的应用程序只需编写一次，就可以用于许多设备，并且可以利用每个设备的本机功能。

这三者之间的关系如图 1-1 所示。

简单来说，Java SE 就是标准版，包含标准的 JVM 和标准库，而 Java EE 是企业版，它只是在 Java SE 的基础上加上了大量的 API 和库，以方便开发 Web 应用、数据库、消息服务等。Java EE 使用的虚拟机和 Java SE 完全相同。Java ME 和 Java SE 不同，它是一个针对嵌入式设备的“瘦身版”，Java SE 的标准库无法在 Java ME 上使用，Java ME 的虚拟机也是“瘦身版”。

毫无疑问，Java SE 是整个 Java 平台的核心，而 Java EE 是进一步学习 Web 应用所必需的。人们熟悉的 Spring 等框架都是 Java EE 开源生态系统的一部分。不幸的是，Java ME 在早期的智能手机中广泛应用，现在 Android 开发成为移动平台的标准之一，因此，没有特殊需求，不建议学习 Java ME。

根据目前需要的岗位核心能力，推荐的 Java 学习途径如下：

- ①首先要学习 Java SE，掌握 Java 语言、Java 核心开发技术及 Java 标准库的使用。
- ②若要从从事 Java EE 工程师、Web 开发给工程师等岗位，需要继续学习 Java EE，学习的后续内容是 Spring 框架、数据库开发、分布式架构。
- ③若要从从事大数据开发工程师等岗位，那么 Hadoop、Spark、Flink 这些大数据平台就是需要学习的，它们都基于 Java 或 Scala 开发的。

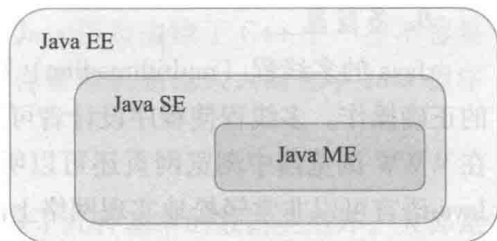


图 1-1 Java EE、Java SE、Java ME 三者之间的关系

④若要从事移动开发工程师岗位，那么就深入 Android 平台，掌握 Android App 的开发。无论怎么选择，Java SE 的核心技术是基础。

#### 四、Java 未来前景

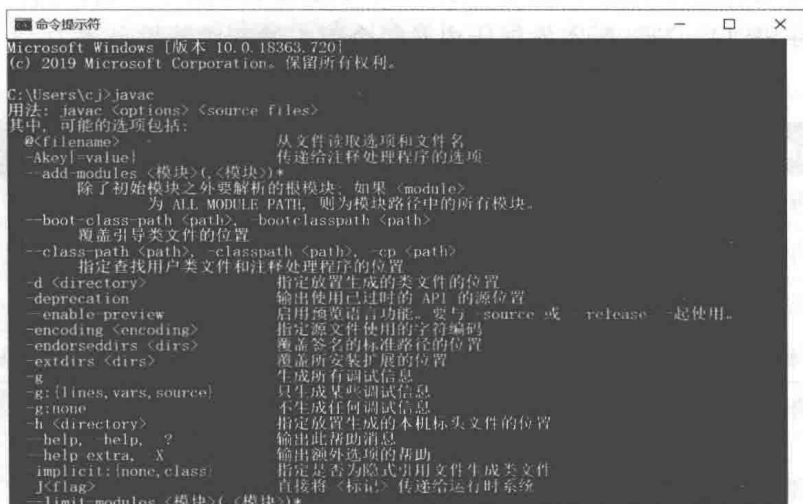
由于 Java 语言具有上述优秀特性，所以其应用前景必然美好，未来发展肯定会与互联网的发展需求绑定：

- ①所有面向对象的应用开发。
- ②软件工程中需求分析、系统设计、开发实现和维护。
- ③中小型多媒体系统设计与实现。
- ④消息传输系统。
- ⑤分布计算交易管理应用（JTS/RMI/CORBA/JDBC 等技术应用）。
- ⑥Internet 的系统管理功能模块的设计，包括 Web 页面的动态设计、网站信息提供管理和交互操作设计等。
- ⑦Intranet（企业内部网）上完全基于 Java 和 Web 技术的应用开发。
- ⑧Web 服务器后端与各类数据库连接管理器（队列、缓冲池）。
- ⑨安全扫描系统（包括网络安全扫描、数据库安全扫描、用户安全扫描等）。
- ⑩网络/应用管理系统。
- ⑪其他应用类型的程序。

## 任务2 搭建开发环境

### 导入任务

安装 Java 开发工具包 JDK，进行环境变量的配置，完成 JDK 开发环境的搭建。安装配置成功后，在命令提示符窗口输入“javac”命令，得到如图 1-2 所示效果。



```

命令提示符
Microsoft Windows [版本 10.0.18363.720]
(c) 2019 Microsoft Corporation. 保留所有权利。

C:\Users\c.j>javac
用法: javac <options> <source files>
其中, 可能的选项包括:
#<filename>          从文件读取选项和文件名
-Akey[=value]        传递给注释处理程序的选项
-add-modules <模块>(<模块>)*
                    除了初始模块之外要解析的根模块, 如果 <module>
                    为 ALL_MODULE_PATH, 则为模块路径中的所有模块。
-boot-class-path <path> -bootclasspath <path>
                    覆盖引导类文件的位置
-class-path <path> -classpath <path> -cp <path>
                    指定查找用户类文件和注释处理程序的位置
-d <directory>      指定放置生成的类文件的位置
-deprecation        输出使用已过时的 API 的源位置
-enable-preview     启用预览语言功能。要与 -source 或 -release 一起使用。
-encoding <encoding>
                    指定源文件使用的字符编码
-endorsestdirs <dirs>
                    覆盖所有安装扩展的位置
-extdirs <dirs>     生成所有测试信息
-g                 只生成某些调试信息
-g:none            不生成任何调试信息
-h <directory>     指定放置生成的本机标头文件的位置
-help             ?
                    输出此帮助消息
-help extra, X    输出额外选项的帮助
-implicit:[none, class]
                    指定是否为隐式调用文件生成类文件
-J<flag>          直接将 <标志> 传递给运行时系统
-limit-modules <模块>(<模块>)*
  
```

图 1-2 运行效果图

## 知识准备

### 一、Java 开发环境

Java 运行环境，即 Java Runtime Environment，简称为 JRE，是在任何平台上运行 Java 编写的程序都需要用到的软件。终端用户可以以软件或者插件方式得到和使用 JRE。Sun 公司还发布了一个 JRE 的更复杂的版本，叫作 JDK (Java Development Kit)，即 Java 开发者工具包。那么 JRE 和 JDK 之间是什么关系呢？简单地说，JRE 就是运行 Java 字节码的虚拟机。但是，如果只有 Java 源码（源程序），要编译成 Java 字节码，就需要 JDK，因为 JDK 除了包含 JRE，还提供了编译器、调试器等开发工具，因此，搭建 Java 的开发环境就变成了安装配置 JDK。JDK 和 JRE 二者之间的关系如图 1-3 所示。

JDK 现在是一个开源、免费的工具。JDK 是其他 Java 开发工具的基础，也就是说，在安装其他开发工具以前，必须首先安装 JDK。对于初学者来说，使用该开发工具进行学习，可以在学习的初期把精力放在 Java 语言语法的学习上，体会更多底层的知识，对于以后的程序开发很有帮助。

但是 JDK 未提供 Java 源代码的编写环境，这个是 Sun 公司提供的很多基础开发工具的通病，所以实际的代码编写还需要在其他的文本编辑器中进行。比如 Java 可以在记事本中进行代码编写，其实大部分程序设计语言的源代码都是一个文本文件，只是存储成了不同的后缀名罢了。

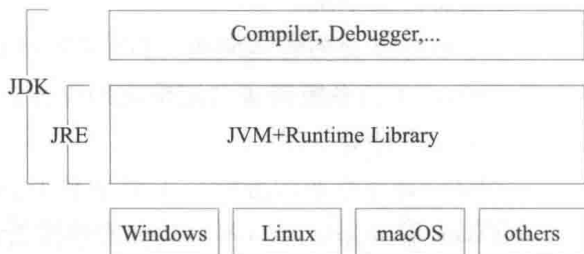


图 1-3 JDK 和 JRE 二者关系

### 二、JDK 版本介绍

1995 年 5 月 23 日，Java 语言诞生。从 1996 年发布 1.0 版本开始，到目前为止，最新的 Java 版本是 Java SE 14。Java 版本发展历史及每个版本的新增特性见表 1-1。

表 1-1 Java 版本发展历史

版本号	发布时间	新增新特性
Java 1.0	1996 年 1 月	Java 虚拟机、Applet、AWT
Java 1.1	1997 年 2 月	JAR 文件格式、JDBC、Java Beans、RMI
Java 1.2	1998 年 12 月	EJB、Java Plug-in、Java IDL、Swing
Java 1.3	2000 年 5 月	Math、Timer API、JNDI、RMI-IIOP、Java 2D API、Java Sound
Java 1.4	2004 年 2 月	Regular Expressions、异常链、NIO、日志类、XML 解析器、XSLT 解析器
Java 1.5	2004 年 9 月	自动装箱与拆箱、泛型、动态注解、枚举、可变量参数、遍历循环、静态导入

续表

版本号	发布时间	新增新特性
Java 6	2006 年 12 月	提供动态语言支持、Desktop 类和 SystemTray 类、使用 JAXB2 来实现对象与 XML 之间的映射、理解 STAX、使用 Compiler API、轻量级 Http Server API、插入式注解处理 API、使用 Console 开发控制台程序、Common Annotations、Java GUI 界面的显示、嵌入式数据库 Derby、Web 服务元数据、Jtable 的排序和过滤、更简单和更强大的 JAX - WS
Java 7	2011 年 7 月	switch 语句中可以使用字符串、泛型实例化类型自动推断、自定义自动关闭类、新增一些读取环境信息的工具方法、Boolean 类型反转、空指针安全、参与位运算、两个 char 之间的 equals、更加安全的加减乘除、对 Java 集合 ( Collections ) 的增强支持、数值可加下划线、支持二进制数字
Java 8	2014 年 3 月	接口的默认方法、Lambda 表达式、函数式接口、方法与构造函数引用、扩展了集合类、新的 Date API、Annotation 多重注解、Streams API、Parallel Streams、Map 数据结构改进
Java 9	2017 年 9 月	Jigsaw 模块化项目、简化进程 API、轻量级 JSON API、钱和货币的 API、改善锁竞争机制、代码分段缓存、智能 Java 编译、HTTP2.0、客户端、Kulla 计划
Java 10	2018 年 3 月	局部变量的类型推断、GC 改进和内存管理、线程本地握手、备用内存设备上的堆分配、其他 Unicode 语言 - 标记扩展、基于 Java 的实验性 JIT 编译器、开源根证书、根证书颁发认证 ( CA )、将 JDK 生态整合单个存储库、删除工具 javah、Java REPL ( JShell )
Java 11	2018 年 9 月	本地变量类型推断、字符串增强、集合增强、Stream 增强、Optional 增强、InputStream 增强、标准化 HTTP Client API、单个命令编译运行源代码
Java 12	2019 年 3 月	微基准测试套件、Switch 表达式、Shenandoah 垃圾回收、JVM Constants API、Default CDS Archives
Java 13	2019 年 9 月	动态应用程序类 - 数据共享、增强 ZGC 释放未使用内存、Socket API 重构、Switch 表达式扩展 ( 预览功能 )、文本块 ( 预览功能 )
Java 14	2020 年 3 月	instanceof 的模式匹配 ( 预览 )、改进了 NullPointerException 的可读性、Record ( 预览特性 )、改进的 switch 表达式实现完全支持

### 三、JDK 的下载、安装、配置和测试

#### (一) JDK 的下载

如果需要获得 JDK 最新版本，可以到 Oracle 公司的官方网站上进行下载，下载地址为 <https://www.oracle.com/java/technologies/javase-downloads.html>，下载最新版本的“Java SE 14”，选择对应的操作系统即可。若 PC 机是 Windows 操作系统，可做如图 1-4 所示选择。

其实如果不需要安装 JDK 最新版本的话，也可以在国内主流的下载站点下载 JDK 的安装程序，只是这些程序的版本可能稍微老一些，这些对于初学者来说影响不大。

JDK 14.0.1 checksum

**Java SE Development Kit 14**  
This software is licensed under the Oracle Technology Network License Agreement for Oracle Java SE








Product / File Description	File Size	Download
Linux Debian Package	157.92 MB	 jdk-14.0.1_linux-x64_bin.deb
Linux RPM Package	165.04 MB	 jdk-14.0.1_linux-x64_bin.rpm
Linux Compressed Archive	182.04 MB	 jdk-14.0.1_linux-x64_bin.tar.gz
macOS Installer	175.77 MB	 jdk-14.0.1_osx-x64_bin.dmg
macOS Compressed Archive	176.19 MB	 jdk-14.0.1_osx-x64_bin.tar.gz
Windows x64 Installer	162.07 MB	 jdk-14.0.1_windows-x64_bin.exe
Windows x64 Compressed Archive	181.53 MB	 jdk-14.0.1_windows-x64_bin.zip

图 1-4 JDK 下载界面

## (二) JDK 的安装

Windows 操作系统上的 JDK 安装程序是一个 exe 可执行程序，直接安装即可，在安装过程中可以选择安装路径及安装的组件等，如果没有特殊要求，选择默认设置即可。程序默认的安装路径在 C:\Program Files\Java 目录下，也可以更改安装路径，如图 1-5 所示。按照软件提示一步步安装即可，得到如图 1-6 所示结果，表明安装成功。

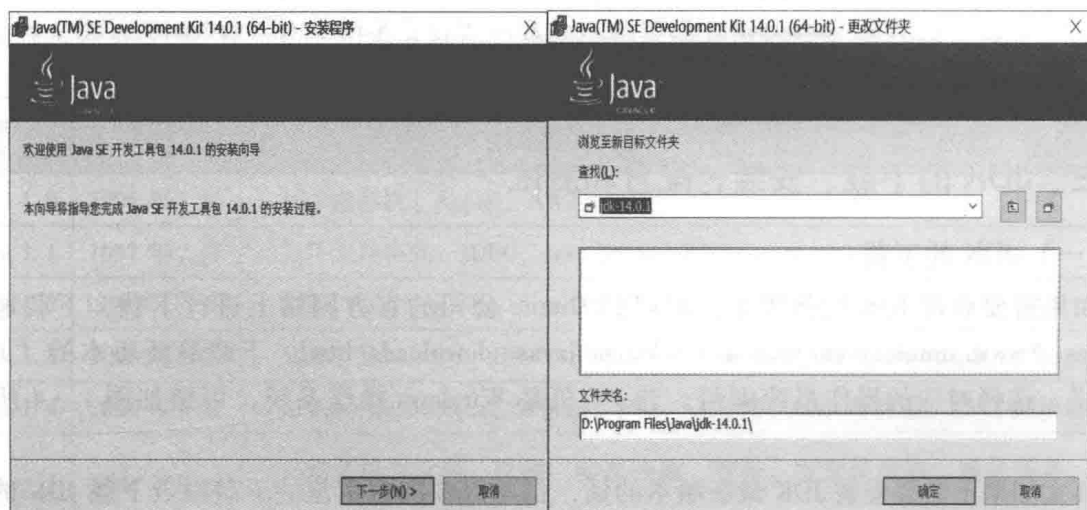


图 1-5 JDK 更改安装路径选择

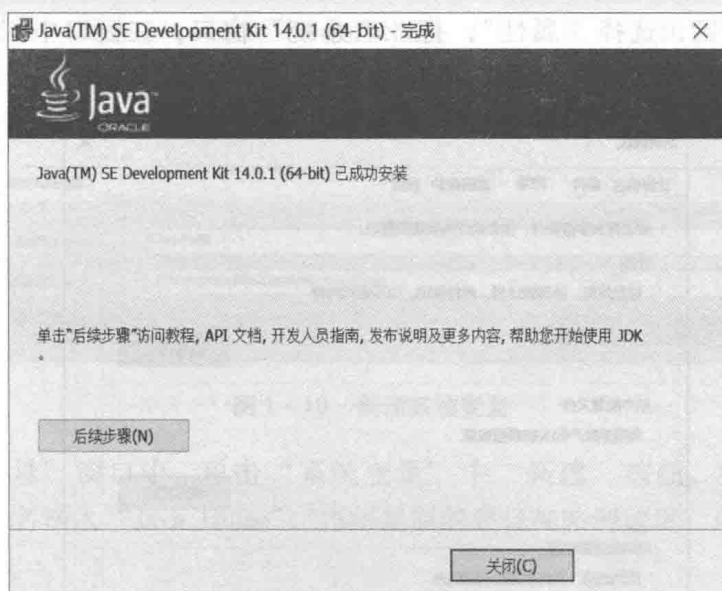


图 1-6 安装成功界面

### (三) JDK 的配置

JDK 安装完成以后, 可以不用设置就进行使用, 但是为了使用方便, 一般需要进行简单的配置。由于 JDK 提供的编译和运行工具都是基于命令行的, 所以需要进行一下 DOS 下面的一个设定, 把 JDK 安装目录下 bin 目录中的可执行文件都添加到 DOS 的外部命令中, 这样就可以在任意路径下直接使用 bin 目录下的 exe 程序了。配置的参数为操作系统中的 path 环境变量, 该变量的用途是系统查找可执行程序所在的路径。配置步骤如下。

①找到 JDK 的安装路径, 打开文件夹 D:\Program Files\Java\jdk-14.0.1\bin, 选中地址栏的地址路径, 右击, 选择“复制”, 如图 1-7 所示。

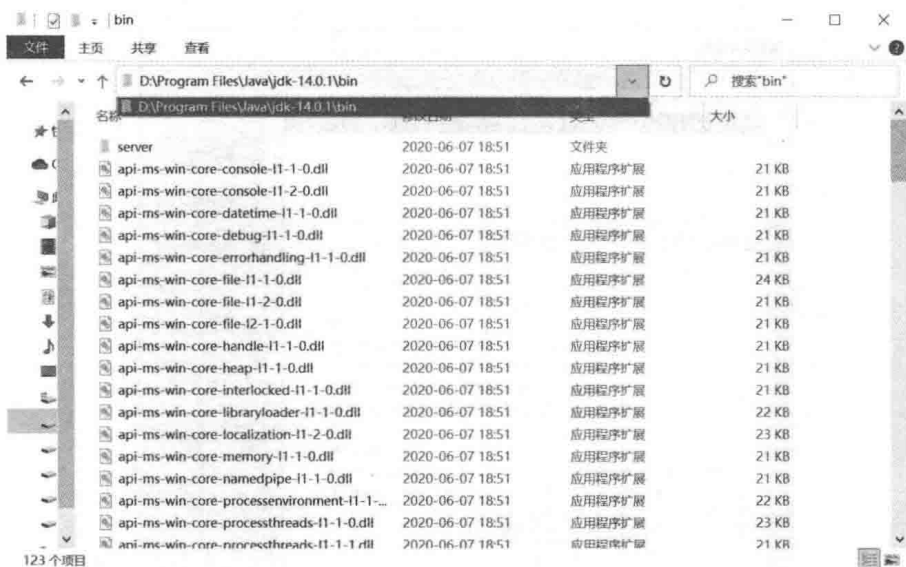


图 1-7 bin 目录中的可执行文件的路径

②单击“开始”→“设置”→“控制面板”→“系统”。也可以选择桌面上的“我的

电脑”，单击鼠标右键，选择“属性”，打开“系统”窗口，在窗口中选择“高级系统设置”，打开“系统属性”窗口，如图 1-8 所示。

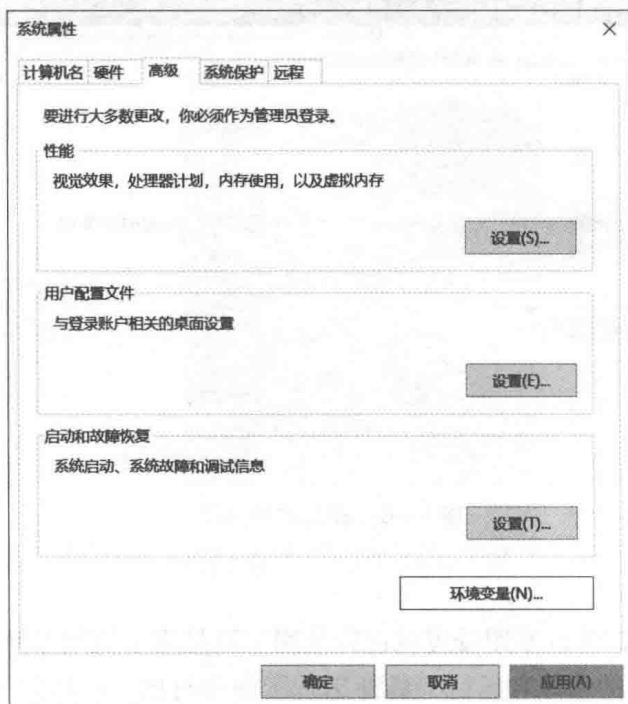


图 1-8 “系统属性”窗口

③在“系统属性”窗口中，单击“高级”属性页中的“环境变量”按钮。选中用户变量中的 Path 变量这一行，单击“编辑”按钮，打开“编辑环境变量”对话框，选择“编辑”按钮，在第二行空白处单击右键，选择“粘贴”，将之前复制的 bin 目录中的可执行文件的路径粘贴到这里，如图 1-9 所示。

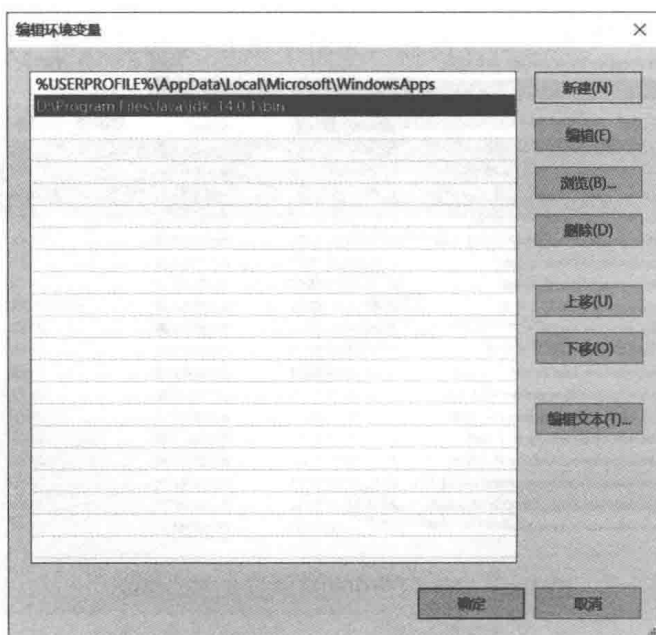


图 1-9 编辑用户变量

④在“环境变量”窗口中，单击“系统变量”中“新建”按钮，打开“新建系统变量”对话框，变量名输入“ClassPath”，变量值输入“.”；，再将复制的路径粘贴到这里，单击“确定”按钮，如图1-10所示。

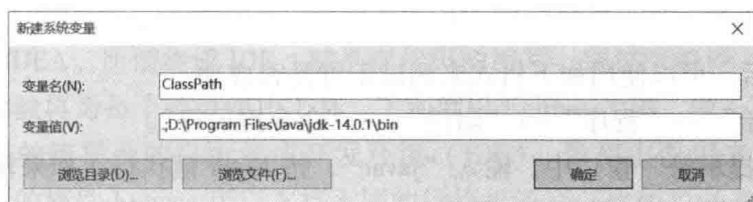


图1-10 新建系统变量

⑤在“环境变量”窗口中，单击“系统变量”中“新建”按钮，打开“新建系统变量”对话框，变量名输入“Java\_Home”，再将复制的路径粘贴到这里，单击“确定”按钮，如图1-11所示。

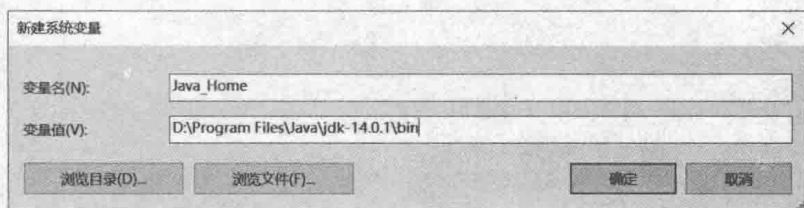


图1-11 新建系统变量 Java\_Home

⑥新建完成两个系统变量后，效果如图1-12所示，单击“确定”→“确定”按钮，保存设置。

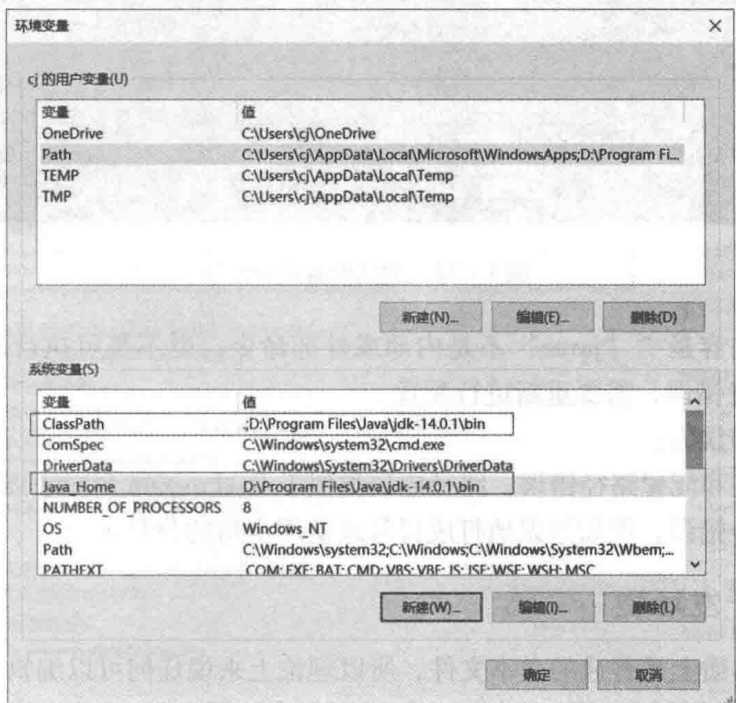


图1-12 系统变量设置成功