



“十四五”职业教育国家规划教材

“十四五”职业教育辽宁省规划教材



扫描二维码
共享立体资源

Java语言程序设计

主 编 康 伟

第二版

北京出版集团
北京出版社

Java语言程序设计
(第二版)

主
编
康
伟

北京出版集团
北京出版社

图书在版编目 (CIP) 数据

Java 语言程序设计 / 康伟主编. —2 版. —北京:
北京出版社, 2021.6 (2023 重印)
十二五规划教材: 2014 版
ISBN 978-7-200-16291-2

I. ① J… II. ① 康… III. ① JAVA 语言—程序设计—
高等学校—教材 IV. ① TP312.8

中国版本图书馆 CIP 数据核字 (2021) 第 009489 号

Java 语言程序设计 (第二版)

Java YUYAN CHENGXU SHEJI (DI-ER BAN)

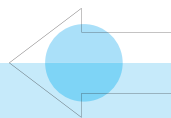
主 编: 康 伟
出 版: 北京出版集团
北 京 出 版 社
地 址: 北京北三环中路 6 号
邮 编: 100120
网 址: www.bph.com.cn
总 发 行: 北京出版集团
经 销: 新华书店
印 刷: 定州启航印刷有限公司
版 印 次: 2021 年 6 月第 2 版 2023 年 7 月修订 2023 年 7 月第 3 次印刷
成品尺寸: 185 毫米 × 260 毫米
印 张: 15.5
字 数: 331 千字
书 号: ISBN 978-7-200-16291-2
定 价: 46.50 元

教材意见建议接收方式: 010-58572162 邮箱: jiaocai@bphg.com.cn

如有印装质量问题, 由本社负责调换

质量监督电话: 010-82685218 010-58572162 010-58572393

单元一 认识 Java 语言	1
任务 1 Java 语言的发展与特点	2
任务 2 第一个 Java 程序	10
单元二 Java 基本语法	15
任务 1 常量和变量	16
任务 2 算术表达式与赋值表达式	24
任务 3 条件判断	29
任务 4 数据类型的转换	35
任务 5 流程控制语句	37
任务 6 数组	49
任务 7 字符串	54
单元三 Java 面向对象特性	64
任务 1 类的定义	65
任务 2 创建和使用对象	69
任务 3 static 修饰符	73
任务 4 类的封装	75
任务 5 继承	79
任务 6 抽象类	83
任务 7 多态	86
任务 8 接口	91
任务 9 包	94
单元四 Java 的异常处理	107
任务 1 异常的处理	108
任务 2 自定义异常	114



单元五 Java 的多线程 119

任务 龟兔赛跑 120

单元六 Java Applet 133

任务 1 在网页中运行 Java 程序 134

任务 2 在网页中绘图 137

单元七 Java 的图形用户界面 144

任务 1 Java 图形用户界面程序 145

任务 2 界面布局 159

任务 3 事件监听 166

任务 4 Applet 中的事件处理 173

单元八 输入 / 输出与文件处理 179

任务 1 使用输入 / 输出流 180

任务 2 创建文件和文件夹 188

任务 3 文件输入 / 输出操作 190

单元九 Java 数据库编程 200

任务 1 连接数据库 201

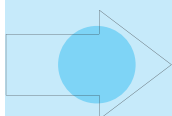
任务 2 访问记录 206

单元十 Java 的网络应用 214

任务 1 一个简单的 JSP 页面 215

任务 2 简单 Bean 及其使用 229

任务 3 基本 Servlet 程序 234



认识 Java 语言

语言是人类进行沟通交流的表达方式。两个人之间进行沟通与信息交换，必须彼此能听懂对方的语言，Java 语言也是一门语言，只不过它是一门计算机编程语言（计算机语言），那么什么是计算机语言呢？

- 人与计算机之间进行传递信息的语言，计算机能“读懂”，专业人士也能“读懂”。人们可以通过使用计算机语言指挥计算机完成某种特定操作。
- 计算机语言中也有字符、语法规则、句子等。
- 常见的计算机语言如 C、C++、C#、Java 等。

电脑操作系统、QQ 聊天软件、电脑游戏、WORD、WPS、微信、支付宝、高考的网上报名系统、智能手机的各种 App 软件、宇宙飞船的飞行控制等，这些都是计算机语言的杰作，它们都是由专业人士通过各种计算机语言编程实现的程序。

时至今日，我们无论在工作或生活中几乎无时无刻不在使用计算机程序，试想一下，离开计算机程序我们会怎么样？

学习计算机语言是我们让计算机为我们所用的必由之路。本单元主要介绍了 Java 语言的发展历史、主要特点、Java 平台家族以及 Java 程序开发基础环境（即 JDK 和集成开发环境）。通过本单元学习，读者将对 Java 语言的基本原理、语法结构与编程方法有一定的了解，能够简单地编辑、调试和运行 Java 程序代码。

学习目标

- 了解 Java 语言的产生发展。
- 了解 Java 语言的语法机制，掌握 Java 程序的结构、编程规范。
- 掌握 Java 程序运行的基本原理与 Java 虚拟机（JVM）的作用。
- 了解 Java 程序的开发工具，并掌握安装 Java 开发环境的方法。

任务 1

Java 语言的发展与特点

任务描述

了解 Java 程序的开发工具，选择 JDK+eclipse 来搭建 Java 开发环境，编写简单 Java 程序调试程序并测试开发环境。

任务目标

1. 能够说出 Java 语言的发展历程和工作原理。
2. 能够掌握 Java 语言的语法机制。
3. 能够掌握 Java 运行系统基本原理与 Java 虚拟机 (JVM) 的作用。
4. 能够掌握 Java 开发环境的安装与配置，了解开发工具。

5. Java 的发展不是一帆风顺的，包含了许多曲折和失败，然而在科学探索的道路上要有信心和毅力。科教兴国战略和人才兴国战略需要我们。

预备知识

一、Java 的起源与发展

(一) Java 的起源

1991 年，美国 Sun (Sun Microsystems) 公司一个称为 Green 的项目小组为了能够在消费电子产品上开发应用程序，积极寻找合适的编程语言。消费电子产品种类繁多，包括 PDA、机顶盒、手机等，即使是同一类消费电子产品所采用的处理芯片和操作系统也不相同，存在着跨平台的问题。当时最流行的编程语言是 C 和 C++ 语言，Sun 公司的研究人员就考虑是否可以采用 C++ 语言来编写消费电子产品的应用程序，但是研究表明，对于消费电子产品而言，C++ 语言过于复杂和庞大，并不适用，安全性也并不令人满意。于是，Bill Joy 先生领导的研究小组就着手设计和开发出一种语言，称之为 Oak。该语言采用了许多 C 语言的语法，提高了安全性，并且是面向对象的语言，但是 Oak 语言在商业上并未获得成功。1995 年，互联网在世界上蓬勃发展，Sun 公司发现 Oak 语言所具有的跨平台、面向对象、安全性高等特点非常符合互联网的需要，于是改进了该语言的设计，达到如下几个目标：

- (1) 创建一种面向对象的程序设计语言，而不是面向过程的语言；
- (2) 提供一个解释执行的程序运行环境，是程序代码独立于平台；
- (3) 吸收 C 和 C++ 的优点，使程序员容易掌握；
- (4) 去掉 C 和 C++ 中影响程序健壮性的部分，使程序更安全，例如指针、内存申

请和释放；

- (5) 实现多线程，使得程序能够同时执行多个任务；
- (6) 提供动态下载程序代码的机制；
- (7) 提供代码校验机制以保证安全性。

最终，Sun 公司给该语言取名为 Java 语言，造就了一代成功的编程语言。

(二) Java 的发展

1995 年，美国 Sun 公司正式向 IT 业界推出了 Java 语言，该语言具有安全、跨平台、面向对象、简单、适用于网络等显著特点，当时以 Web 为主要形式的互联网正在迅猛发展，Java 语言的出现迅速引起所有程序员和软件公司的极大关注，程序员们纷纷尝试用 Java 语言编写网络应用程序，并利用网络把程序发布到世界各地进行运行，包括 IBM、Oracle、微软、Netscape、Apple、SGI 等大公司纷纷与 Sun 公司签订合同，获得授权使用 Java 平台技术。微软公司总裁比尔·盖茨先生在经过研究后认为：“Java 语言是长时间以来最卓越的程序设计语言。”目前，Java 语言已经成为流行的网络编程语言，许多大学纷纷开设 Java 课程，Java 成为世界上程序员最多的编程语言。

在经历了以大型机为代表的集中计算模式和以 PC 为代表的分散计算模式之后，互联网的出现使得计算模式进入了网络计算时代。网络计算模式的一个特点是计算机是异构的，即计算机的类型和操作系统是不一样的，例如 SUN 工作站的硬件是 SPARC 体系，软件是 UNIX 中的 Solaris 操作系统，而 PC 的硬件是 INTEL 体系，操作系统是 Windows 或者是 Linux，因此相应的编程语言基本上只是适用于单机系统，例如 COBOL、FORTRAN、C、C++ 等；网络计算模式的另一个特点是代码可以通过网络在各种计算机上进行迁移，这就迫切需要一种跨平台的编程语言，使得用它编写的程序能够在网络中的各种计算机上正常运行，Java 就是在这种需求下应运而生的。正是因为 Java 语言符合了互联网时代的发展要求，才使它获得了巨大的成功。

二、什么是 Java 技术

Java 技术是一种独立于操作系统平台的编程语言与开发平台。Java 编程语言的句法与 C++ 的句法相似，语义则与 Small Talk TM 的语义相似。Java 编程语言可被用来创建任何常规编程语言所能创建的应用程序。

设计 Java 编程语言的主要目标是：提供一种易于编程的语言；消除其他语言在诸如指针运算和存储器管理方面影响健壮性的缺陷；利用面向对象的概念使程序真正地成为可视化程序；为使代码尽可能清晰合理、简明流畅提供一种方法；为获得以下两点益处提供一种解释环境：提高开发速度——消除编译→链接→装载→测试周期、代码可移植性——使操作系统能为运行环境做系统级调用；为运行不止一个活动线程的程序提供了一种方式；通过允许下载代码模块，从而当程序运行时也能动态支持程序改变；为那些保证安全性而装载的代码模块提供了一种检查方法。

Java 不仅是编程语言，还是一个开发平台。Java 技术给程序员提供了许多工具：编译器、解释器、文档生成器和文件打包工具等。同时 Java 还是一个程序发布平台，有两

种主要的“发布环境”。首先 Java 运行时环境 (Java Runtime Environment, JRE) 包含了完整的类文件包, 其次许多主要的浏览器都提供了 Java 解释器和运行时环境。目前 Sun 公司把 Java 平台划分为: 标准版 (J2SE)、企业版 (J2EE)、微缩版 (J2ME) 三个版本, 针对不同的市场目标和设备进行定位。

J2SE 即 Java 2 Standard Edition, 主要目的是为台式机和 workstation 提供一个开发和运行的平台。我们在学习 Java 的过程中, 主要是采用 J2SE 来进行开发。

J2EE 即 Java 2 Enterprise Edition, 主要目的是为企业计算提供一个应用服务器的运行和开发平台。J2EE 本身是一个开放的标准, 任何软件厂商都可以推出自己的符合 J2EE 标准的产品, 使用户可以有多种选择。IBM、Oracle、BEA、HP 等 29 家厂商已经推出了自己的产品, 其中尤以 BEA 公司的 weblogic 产品和 IBM 公司的 websphere 最为著名。J2EE 逐步发展成为可以与微软的 .NET 战略相对抗的网络计算平台。

J2ME 即 Java 2 Micro Edition, 主要是面向消费电子产品, 为消费电子产品提供一个 Java 的运行平台, 使得 Java 程序能够在手机、机顶盒、PDA 等产品上运行。

三、Java 的语法机制概述

(一) Java 源程序的组成部分

```
package 语句;
// 该部分至多只有一句, 必须放在源程序的第一句
import 语句;
// 该部分可以有若干 import 语句或者没有, 必须放在所有的类定义之前
public classDefinition;
// 公共类定义部分, 至多只有一个公共类的定义
// Java 语言规定该 Java 源程序的文件名必须与该公共类名完全一致
classDefinition;
// 类定义部分, 可以有 0 个或者多个类定义
interfaceDefinition;
// 接口定义部分, 可以有 0 个或者多个接口定义
```

例如, 一个 Java 源程序可以是如下结构, 该源程序命名为 HelloWorldApp.Java:

```
package Javawork.newcentury;
// 把编译生成的所有 .class 文件放到包 Javawork.helloworld 中 import java.
awt.*; // 告诉编译器本程序中用到系统的 AWT 包
import javawork.newcentury;
// 告诉编译器本程序中用到用户自定义的包 javawork.newcentury
public class HelloWorldApp{.....}
// 公共类 HelloWorldApp 的定义, 名字与文件名相同
class TheFirstClass{.....} // 第一个普通类 TheFirstClass 的定义
class TheSecondClass{.....} // 第二个普通类 TheSecondClass 的定义
.....
// 其他普通类的定义
interface TheFirstInterface{.....}
// 第一个接口 TheFirstInterface 的定义
.....
// 其他接口定义
```

（二）源文件的命名规则

如果在源程序中包含有公共类的定义，则该源文件名必须与该公共类的名字完全一致，字母的大小写都必须一样。这是 Java 语言的一个严格的规定，如果不遵守，在编译时就会出错。因此，在一个 Java 源程序中至多只能有一个公共类的定义。如果源程序中不包含公共类的定义，则该文件名可以任意取名。如果在一个源程序中有多个类定义，则在编译时将为每个类生成一个 .class 文件。

（三）注释

为了方便他人理解程序中的某个变量、方法或者类对象的定义，提高程序的可读性，程序中往往需要编写注释。系统编译时，编译器会跳过注释。注释有以下三种形式：

（1）单行注释，以“//”开始，该行后面的内容都会成为注释，可以放在语句开头或者注释掉该行或者放在语句结尾对该句进行注解。

（2）多行注释，以“/*”开始，“*/”结束。

（3）文档注释，以“/**”开始，“*/”结束。通常用于对方法或类的注解，说明代码含义、参数及其功能等，这种注释可以由 javadoc 程序处理成 HTML 文档，其有规定的格式。javadoc 能够识别的标识如表 1-1 所示。

表 1-1 javadoc 中常用注释标记

标识	含义
@author	指明类或方法的作者
@deprecated	指明反对使用这个类或者成员
@exception	指明一个方法可能引发的异常
@param	指明方法的主要参数
@return	指明方法的返回值
@see	指明对另一个对象的链接
@throws	确定一个方法出现的异常
@version	指明版本信息
@docRoot	指明当前文档的根目录

四、Java 运行系统与 JVM

Java 的运行系统是一个完整、有机的程序设计系统，其功能主要由 JVM、JDK 以及开发平台组成。

（一）Java 虚拟机——JVM

JVM (Java Virtual Machine)，Java 虚拟机，是软件模拟的计算机，可以在任何处理器上（无论是在计算机中还是在其他电子设备中）安全并且兼容地执行保存在 .class 文件中的字节码。Java 虚拟机的“机器码”保存在 .class 文件中，也可以称之为字节码文件。Java 程序的跨平台主要是指字节码文件可以在任何具有 Java 虚拟机的计算机或者电子

设备上运行，Java 虚拟机中的 Java 解释器负责将字节码文件解释成为特定的机器码进行运行。Java 源程序需要通过编译器编译成为 .class 文件，Java 程序的编译和执行过程如图 1-1 所示。

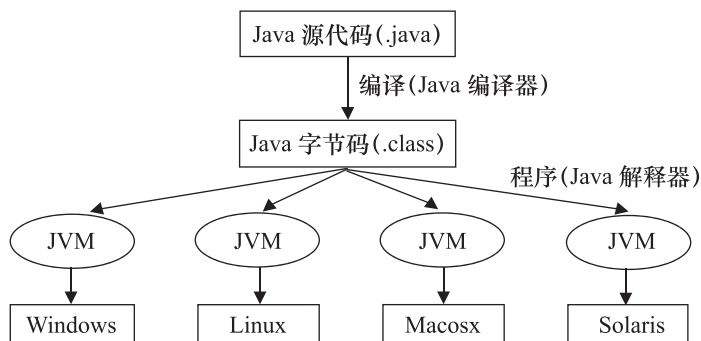


图 1-1 Java 程序的编译和执行过程

但是，Java 虚拟机的建立需要针对不同的软硬件平台做专门的实现，既要考虑处理器的型号，也要考虑操作系统的种类。目前在 SPARC 结构、X86 结构、MIPS 和 PPC 等嵌入式处理芯片上，在 UNIX、Linux、Windows 和部分实时操作系统上都有 Java 虚拟机的实现，Java 程序通过 Java 虚拟机可以实现跨平台特性。但 Java 虚拟机并不是跨平台的，也就是说，不同操作系统上的 Java 虚拟机是不同的，即 Windows 平台上的 Java 虚拟机不能用在 Linux 平台上，反之亦然。

(二) JVM 的组成

JVM 为下列各项做出了定义：

- 指令集（相当于中央处理器 CPU）
- 注册集
- 类文件格式
- 栈
- 垃圾收集堆
- 存储区

(三) 垃圾自动回收机制

在程序的执行过程中，部分内存使用过后就处于废弃状态，如果不及时进行无用内存的回收，就会造成内存泄漏，进而导致系统崩溃。在 C++ 语言中是由程序员进行内存回收的，程序员需要在编写程序的时候把不再使用的对象内存释放掉；但是这种人为的管理内存释放的方法却往往由于程序员的疏忽而致使内存无法回收，同时也增加了程序员的工作量。而在 Java 运行环境中，始终存在着一个系统级的线程，专门跟踪内存的使用情况，定期检测出不再使用的内存，并进行自动回收，避免了内存的泄漏，也减轻了程序员的工作量。

(四) 代码安全性检查机制

字节码的执行需要经过三个步骤：首先，由类装载器(Class Loader)负责把类文件

(.class 文件)加载到 Java 虚拟机中,在此过程需要检验该类文件是否符合类文件规范;其次,字节码校验器(Bytecode Verifier)检查该类文件的代码中是否存在着某些非法操作,例如:Applet 程序中写本机文件系统的操作;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。

五、Java 程序开发

(一)JDK 的安装和配置

1. 下载并安装 JDK

Java 开发工具包(Java Development Kits,简称 JDK)是 Java 应用程序开发的基础工具包,开发 Java 应用程序首先必须下载 JDK(JDK 可以从 Oracle 公司的官方网站上下载)。下载完成后,双击图标进行安装,安装过程中可以自定义安装目录等信息。例如:可以选择安装目录为 C:\Program Files(x86)\Java\jdk1.8.0_144。

2. 认识 JDK

在安装完成后,可以发现在安装目录下具有 bin、include、jre、lib、sample、demo 等文件夹,其作用如表 1-2 所示。

表 1-2 JDK 包的目录及其作用

bin	bin 目录提供了工具程序,包括 javac、java、javadoc 等
jre	jre 目录下的文件是 JDK 附带的 JRE 资源包
lib	lib 目录下提供了 Java 工具所需要的资源文件
demo	demo 目录下提供了 Java 编写的范例程序
src.zip	src.zip 目录下提供了 API 类的源代码压缩文件

3. 配置 JDK 环境变量

(1)右击“此电脑”,如图 1-2 所示,选择“属性”命令。

(2)切换到“系统”窗口,如图 1-3 所示,单击“高级系统设置”。

(3)切换到“系统属性”窗口,单击“高级”选项卡,如图 1-4 所示,单击“环境变量”按钮。

(4)切换到“环境变量”对话框,如图 1-5 所示,在“系统变量”中选择变量“Path”,然后单击“编辑”按钮。

(5)切换到“编辑环境变量”窗口,如



图 1-2 配置 JDK 系统路径 1

图 1-6 所示，点击“编辑”或“新建”按钮，添加路径 C:\Program Files (x86)\Java\jdk1.8.0_144\bin，这样使得系统可以在任何路径下识别 Java 命令。



图 1-3 配置 JDK 系统路径 2



图 1-4 配置 JDK 系统路径 3

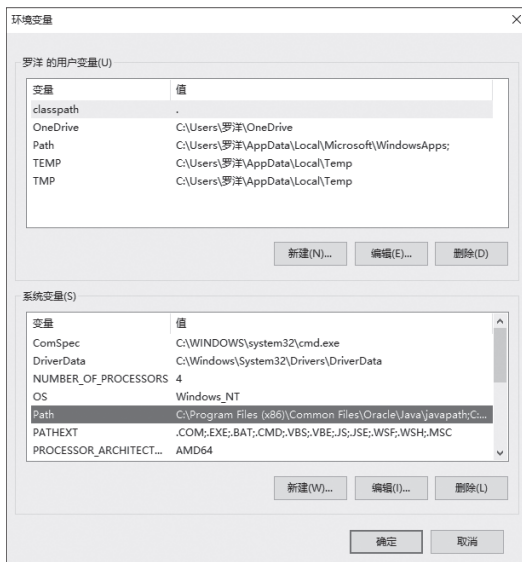


图 1-5 配置 JDK 系统路径 4

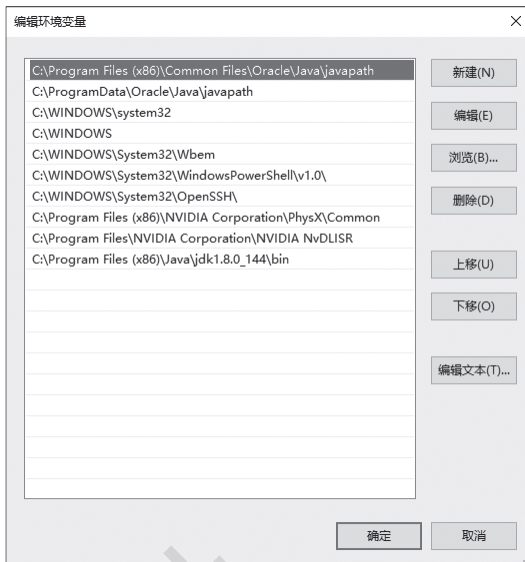


图 1-6 配置 JDK 系统路径 5

(6) 在步骤(4)中，在“用户变量”中单击“新建”，如图 1-7 所示，在变量名中输入“classpath”，在变量值中输入“.”，classpath 为 Java 加载类(class 或 lib)路径，只有类在 classpath 中，Java 命令才能识别，设为“.”表示当前路径，然后单击“确定”按钮。



图 1-7 配置 JDK 系统路径 6

(二) Eclipse 简介

1. Eclipse 简介

为了提高程序的开发效率，大部分软件开发人员都是使用集成开发环境(Integrated Development Environment, IDE)来进行 Java 程序开发的。常用的 Java 的 IDE 主要有: JCreator, JBuilder, Eclipse 等。其中 Eclipse 是 IBM 花费巨资开发的一款功能完整且成熟的 IDE 集成开发环境。它是一个开源的，基于 Java 的可扩展开发平台，是目前最流行的 Java 语言开发工具。Eclipse 具有强大的代码编排功能。可以帮助程序开发人员完成语法修正、代码修正、补全文字信息提示等编码工作，大大提高了程序开发的效率。

尽管 Eclipse 是使用 Java 语言开发的，但它的用途并不限于 Java 语言。例如，支持诸如 C/C++、COBOL 和 Eiffel 等编程语言的插件已经可用，或预计会推出。Eclipse 框架还可用来作为与软件开发无关的其他应用程序类型的基础，比如内容管理系统。

基于 Eclipse 的应用程序的突出例子是 IBM 的 WebSphere Studio Workbench，它构成了 IBM Java 开发工具系列的基础。例如，WebSphere Studio Application Developer 添加了对 JSP、Servlet、EJB、XML、Web 服务和数据库访问的支持。

2. 下载并安装 Eclipse

最新版本的 Eclipse IDE 可以从它的官方网站上下载，如图 1-8 所示。下载后安装 Eclipse 之前必须保证已安装恰当的 JDK。

Eclipse 是一个绿色软件，无须安装即可执行。如需中文版的集成开发环境，可在 Eclipse 官方网站下载相应的中文语言包。解压后，分别将其 features、plugins 目录下的文件复制到 Eclipse 安装目录下的 features、plugins 目录中，重新启动 Eclipse 即可。

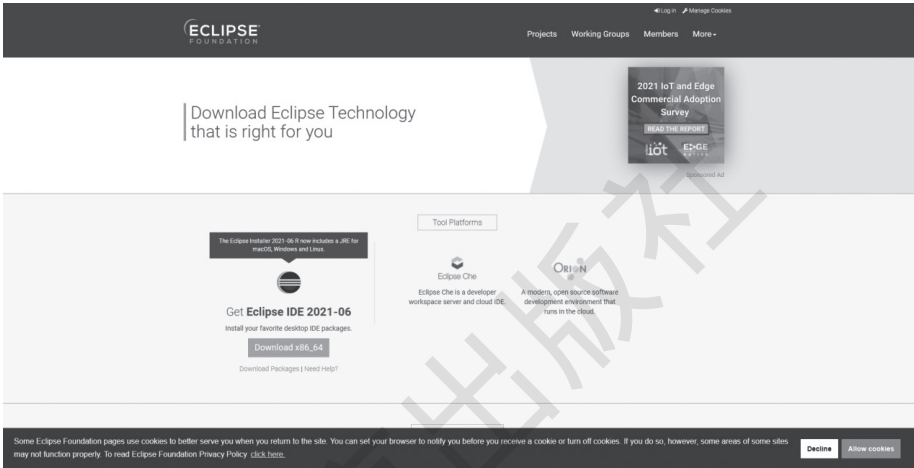


图 1-8 Eclipse 下载页面

任务 2

第一个 Java 程序

任务描述

编写一个 Java 应用程序，它能够利用来自 Java 标准库的 System 对象的多种方法，打印出与当前运行的系统有关的资料。其中“//”代表一种注释方式，表示从这个符号开始到这行结束的所有内容都是注释。在每个程序文件的开头，如果这个文件的代码中用到了系统所提供的额外的类，就必须放置一个 import 语句。说它是“额外的”是因为一个特殊的类库“Java.lang.*”会自动导入到每个 Java 文件。

任务目标

- 1. 能够了解 Java 程序分为两类。
- 2. 能够掌握 Java 标准库中 System 对象的多种方法。

3. 能够编写最简单的“字符串输出”程序，掌握编写、编译、运行 Java 程序的基本方法。

4. 养成良好编程习惯，促进对编程的理解，具备初步的编程思维，好的习惯会让自己终生受益。

预备知识

Java 程序分为 Java Application (Java 应用程序) 和 Java Applet (Java 小应用程序) 两种。怎样在开发平台上开发、编译以及运行 Java 应用程序呢？下面介绍 Java 的基本语法。

(1) Java 中的程序代码可分为结构定义语句和功能执行语句。其中，结构定义语句用于声明一个类或者方法；功能执行语句用于实现具体的功能，每条功能执行语句的最后都必须用分号(;)结束。

(2) 虽然 Java 没有严格要求用什么样的格式来编排程序代码，但是，出于可读性的考虑，应该让自己编写的程序代码整齐美观，层次清晰。以下两种方法都可以，但是建议使用后一种。

方式一：

```
public class HelloWorld{public static void
main(String[]
Args){System.out.println("我的第一个 java 程序");}}
```

方式二：

```
public class HelloWorld
{
    public static void main(String[] Args)
    {
        System.out.println("我的第一个 java 程序");
    }
}
```

(3) Java 程序中，一句连续的字符串儿不能分开在两行中书写。例如，下面这条语句在编译时将会出错。

```
System.out.println("我的
第一个 java 程序");
```

任务实施

```
/* 源程序文件名: Ex0101.Java*/
import java.util.Date;
import java.util.Properties;
```

```

public class Ex0101    // 创建公共类，名为 Ex0101
{
    public static void main(String args[])
    {
        System.out.println(new Date());
        Properties p = System.getProperties();
        p.list(System.out);
        System.out.println("--- Memory Usage:");
        Runtime rt = Runtime.getRuntime();
        System.out.println("Total Memory = "+rt.totalMemory()+" Free Memory = "+rt.freeMemory());    // 打印总内存大小，打印空闲内存大小
    }
}

```

程序运行结果如图 1-9 所示。

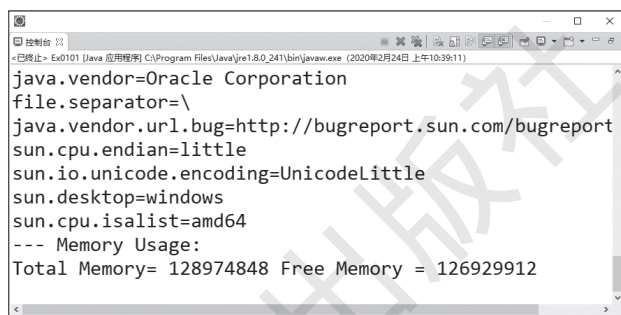


图 1-9 Ex0101.Java 的运行结果

该任务的主要目的是使大家对 Java 的编程结构有一个总体认识，在 Java 中所有的东西都是对象，程序就是由一大堆对象组合而成的，同时每个对象又属于某个类，所有的代码都是在某一个类中完成的。



单元小结

通过本单元的学习，大家已经可以阅读一些简单的 Java 程序了。

Java 语言的产生源于 Sun 公司为消费电子产品上应用程序的开发寻找一门编程语言的过程中，而随着互联网时代的到来，原有的 Oak 软件就顺理成章地改造成 Java 语言推向了市场，其跨平台、面向对象、安全等特点使其得到广泛的应用。通过在不同的硬件上实现的 Java 虚拟机，Java 的字节码文件就可以跨平台地进行运行，无用内存自动回收器也给程序员带来了极大的方便。

Java 是一门纯粹的面向对象的编程语言。面向对象编程的思路认为程序都是对象的组合，因此要克服面向过程编程的思路，直接按照对象和类的思想去组织程序，面向对象所具有的封装性、继承性、多态性等特点使其具有强大的生命力。

Sun 公司为全世界 Java 开发人员提供了一套免费的软件开发包 JDK，它不仅是 Java

的开发平台，还是 Java 的运行平台。

Java 源程序存放在 .Java 文件中，可以使用任意一种文本编辑器编辑。源程序经过“Javac”命令编译过后，就生成了源程序中各个类的相应的 .class 文件。而用“Java”命令则可以运行包含 main() 的 .class 文件。

作为面向对象编程人员来说，还必须掌握一些常用的 IDE 开发环境的使用。现阶段最受程序员喜爱的 IDE 环境是 Eclipse。

思考与练习

一、简答题

- (1) 什么是 Java 编程语言？简述 J2SE、J2EE 和 J2ME 的特点及使用方向。
- (2) Java 源文件的扩展名是什么？Java 字节码文件的扩展名是什么？一个 Java 源文件编译后只有一个字节码文件吗？
- (3) 作为 Java 程序员，可以选择哪些 IDE 集成开发环境来开发程序？
- (4) 为什么说 Java 语言是与平台无关的语言？这种机制是由什么技术来实现的？
- (5) 下载并安装 JDK，配置 Java 命令路径和类路径。
- (6) 下载并安装 Eclipse，使用 Eclipse 编辑、调试并运行任务：“第一个 Java 程序”。

二、选择题

- (1) 下列说法正确的是 ()。
A. Java 程序中的 main 方法必须写在类里面
B. Java 程序中可以有多个 main 方法
C. Java 程序中类名必须与文件名一样
D. Java 程序的 main 方法中如果只有一条语句，可以不用 {} (大括号) 括起来
- (2) 下列不属于 Java 语言的特点是 ()。
A. 面向过程
B. 简单性
C. 平台独立
D. 一次编写，到处运行
- (3) Java 命令中对应编译文件的是 ()。
A. Javac
B. Java
C. Applet
D. Bin
- (4) 下列有关 Java 语言的说法中，错误的是 ()。
A. Java 是高级语言
B. Java 是编译型语言
C. Java 是结构化设计语言
D. Java 是移植性强的语言
- (5) Java 安装目录中，放置 java.exe 和 javac.exe 文件的目录是 ()。
A. \bin
B. \lib
C. \src
D. \demo

三、填空题

- (1) Java 语言是一种完全的 _____ 的程序设计语言。
- (2) Java 的 _____ 大大减轻了程序员的负担，不用再编写专门的内存回收程序解决内存分配问题，不仅提高了编程效率，而且进一步增强了 Java 程序的稳固性。
- (3) Java 源文件中可以有多个类，但最多只能有一个类被 _____ 修饰。
- (4) Java 语言是 _____ 大小写的。



实训任务

- (1) 参照本单元任务，创建一个名为 HelloWorldApp 的 Java Application，在屏幕上简单地显示一句话“Hello World!”。
- (2) 编写一个名为 PrintArguments 的程序，打印从命令行中获取的四个参数。例如：在命令行中输入“Java PrintArguments 10 20 30 40”时，将打印出“10 20 30 40”。



命令行运行程序



拓展阅读——
哈工大高会军教授