

应用型本科信息大类专业“十三五”规划教材

# Java 程序设计与实战

主编 雷 鸿 孙海南 吴 亮

主审 金弘林

应用型本科信息大类专业“十三五”规划教材

# Java 程序设计与实战

主 编 雷 鸿 孙海南 吴 亮

副主编 曾 辉 钱 程 黄金水

主 审 金弘林

西安电子科技大学出版社

## 内 容 简 介

本书分理论篇和实战篇。理论篇的主要内容包括 Java 的开发环境配置、基本语法、基本结构、数组和字符串、对象和类、继承和封装、常用类和集合框架等；实战篇以 4 个实用、有趣的游戏或项目作为训练题材，训练内容涉及本书所介绍的基本知识和技术要点，进一步强化读者对 Java 的基本语法、基本结构、数组和字符串、类和对象等面向对象设计思想及相关技巧的理解，进而全面提高实践动手能力。本书使用的开发环境是 JDK1.8+IDE(Eclipse)，全书内容由浅入深、结构合理、重点难点突出、注重应用。

本书的编写结合了企业软件开发的经验，为学校教学量身定做，针对每个章节都设置了建议的基本教学课时；书中列出了所有实例的代码以及开发过程中用到的软件，供读者学习和参考。本书可作为高校计算机科学与技术、软件工程、网络工程、物联网、计算机软件、计算机信息管理、电子信息技术和经济管理等相关专业的程序设计教材，也可以作为科研、程序设计等人员的参考书籍。

### 图书在版编目(CIP)数据

Java 程序设计与实战 / 雷鸿, 孙海南, 吴亮主编. —西安: 西安电子科技大学出版社, 2019.1

ISBN 978-7-5606-5171-2

I. ① J… II. ① 雷… ② 孙… ③ 吴… III. ① JAVA 语言—程序设计 IV. ① TP312.8

中国版本图书馆 CIP 数据核字(2018)第 286824 号

策划编辑 杨丕勇

责任编辑 张 玮

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 陕西日报社

版 次 2019 年 1 月第 1 版 2019 年 1 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印 张 14

字 数 329 千字

印 数 1~3000 册

定 价 35.00 元

ISBN 978-7-5606-5171-2 / TP

**XDUP 5473001-1**

\*\*\*如有印装问题可调换\*\*\*

# 前 言

面向对象的程序设计技术已经成为当今计算机应用开发领域的主流技术，而 Java 语言自 1991 年诞生以来，至今经历了二十多年的磨砺，已经成为目前世界上最流行、使用最广泛的面向对象程序设计语言之一。Java 语言既是当今大多数软件开发人员的首选，也是高校开设程序设计类课程的首选语言。

本书的编者既有长期从事教学工作的一线教师，也有具备极其丰富的企业软件开发经验的项目经理、设计者和开发人员。在编写过程中，编者以“实战教学法”贯穿本书，全书具有以下特点：

(1) 实战性：所有内容都由案例引入，通俗易懂，并且附有项目实战，进一步加强学生的实践动手能力。

(2) 流行性：书中讲解的都是软件开发过程中 Java 程序设计最流行的方法、技巧和设计模式，并加强对面向对象设计思想的培养。

(3) 适合教学：书中每一个章节都安排适当的知识点和相关案例，并且给出建议课时，教师可以根据实际情况选用，也可以进行适当增减。

本书分为理论篇和实践篇两部分。理论篇共 8 章，各章内容如下：

第 1 章为开发简单 Java 应用程序，主要介绍 Java 的起源历史、工作原理、常用的开发工具、JDK 和 Eclipse 的安装配置。建议 2 学时。

第 2 章为 Java 的基本语法(上)，主要介绍 Java 语言基础，包括标识符、变量和常量、数据类型、类型转换等。建议 4 学时。

第 3 章为 Java 的基本语法(下)，主要介绍 Java 语言的表达式以及流程控制语句。建议 4 学时。

第 4 章为 Java 的数组，主要介绍 Java 数组相关的知识。建议 4 学时。

第 5 章为 Java 的类和对象(上)，主要介绍面向对象中类和对象的基本概念、成员变量和方法以及包的导入等内容。建议 4 学时。

第 6 章为 Java 的类和对象(下)，主要介绍面向对象的类的继承基本概念和特性、Java 中类和接口的概念以及方法的重载和覆盖。建议 4 学时。

第 7 章为 Java 的常用类，主要介绍 Java 的常用类，如 String、StringBuffer、StringBuilder 等。建议 4 学时。

第 8 章为 Java 的集合框架，主要介绍 Java 的集合框架。建议 6 学时。

实战篇的内容如下：

实战 S1 为 Java 开发环境平台搭建，介绍 Java 开发环境的构建方法和软件开发的基本技巧。建议 4 学时。

实战 S2 为利用 Java 循环和分支结构开发万年历，介绍 Java 程序设计基本结构(顺序、分支和循环结构)的使用方法和相关技巧。建议 8 学时。

实战 S3 为利用 Java 的类和对象开发猜拳游戏，介绍定义类、描述类的属性和方法，

创建和使用对象，使用包组织以及开发 Java 工程的基本方法和流程。建议 10 学时。

实战 S4 为利用 Java 综合知识开发 MINI 音乐管理系统，介绍项目开发的业务逻辑分析、程序设计的基本流程和基本方法，以及运行和测试的基本方法。建议 10 学时。

本书可作为高校计算机相关专业 Java 程序设计课程的相关教材，也可作为 Java 技术培训教材以及缺乏 Java 项目实战经验的程序员的快速入门教材。

本书提供所有实例的源代码，供读者学习参考使用，所有程序和项目均已经过作者精心调试。

编 者

2018 年 10 月

# 目 录

## 理 论 篇

第 1 章 开发简单 Java 应用程序 .....	2	2.5.1 自动类型转换 .....	28
1.1 Java 语言的发展过程 .....	2	2.5.2 强制类型转换 .....	29
1.2 Java 的工作原理及特点 .....	2	2.5.3 几种特殊类型的转换 .....	29
1.2.1 Java 程序的处理过程 .....	3	习题 .....	32
1.2.2 Java 字节码文件 .....	3	第 3 章 Java 的基本语法(下).....	33
1.2.3 Java 虚拟机 .....	3	3.1 if 语句.....	33
1.2.4 垃圾回收 .....	4	3.1.1 简单 if 语句.....	33
1.3 Java SDK 的安装与配置 .....	4	3.1.2 标准 if-else 语句 .....	34
1.3.1 下载并安装 Java SDK		3.1.3 多重 if 语句 .....	36
开发工具 .....	4	3.1.4 嵌套 if 语句 .....	37
1.3.2 JDK 的配置与测试 .....	6	3.2 switch-case 语句 .....	39
1.4 Eclipse 的安装与使用 .....	7	3.3 for 语句 .....	43
1.4.1 Eclipse 的安装与启动.....	7	3.3.1 break 语句.....	44
1.4.2 Eclipse 的使用 .....	8	3.3.2 continue 语句.....	48
1.5 Java 程序开发 .....	12	3.3.3 多重 for 循环语句.....	50
1.5.1 Java Application.....	12	3.3.4 多重 for 循环语句中的 break.....	51
1.5.2 Java Applet.....	14	3.4 while 语句 .....	52
习题 .....	16	习题 .....	54
第 2 章 Java 的基本语法(上).....	17	第 4 章 Java 的数组 .....	55
2.1 代码编写规则 .....	17	4.1 数组 .....	55
2.1.1 代码编写规则 .....	17	4.1.1 数组的声明 .....	55
2.1.2 注释 .....	18	4.1.2 数组的创建 .....	55
2.2 标识符的作用 .....	20	4.1.3 数组的使用注意事项 .....	56
2.2.1 标识符的命名 .....	20	4.2 基本数据类型数组 .....	57
2.2.2 保留字/关键字 .....	20	4.2.1 基本数据类型数组的初始化 .....	57
2.3 变量和常量 .....	21	4.2.2 基本数据类型数组的应用 .....	58
2.3.1 变量 .....	21	4.3 引用数据类型数组 .....	59
2.3.2 常量 .....	23	4.3.1 引用数据类型数组的创建 .....	59
2.4 数据类型 .....	24	4.3.2 引用数据类型数组的初始化 .....	60
2.4.1 简单类型 .....	24	4.3.3 引用数据类型数组的应用 .....	61
2.4.2 引用类型 .....	27	4.4 二维数组 .....	62
2.5 类型转换 .....	28		

4.4.1 二维数组的定义 .....	62	6.4 继承中的访问控制符 .....	96
4.4.2 二维数组的初始化 .....	62	6.5 覆盖 .....	98
4.4.3 二维数组的应用 .....	63	6.6 多态 .....	99
4.5 三维数组及多维数组 .....	63	6.7 引用的范围 .....	101
4.5.1 三维数组 .....	63	6.7.1 引用变量的类型转换 .....	102
4.5.2 多维数组 .....	64	6.7.2 Object 类与 Object 型引用变量 ..	103
习题 .....	65	6.8 “==” 与 equals() .....	105
<b>第 5 章 Java 的类和对象(上)</b> .....	66	6.9 类的层级图 .....	107
5.1 类和对象的概念 .....	66	6.10 抽象类 .....	108
5.1.1 面向对象程序设计概述 .....	66	6.11 接口 .....	110
5.1.2 类的基础概述 .....	67	6.12 final 关键字 .....	112
5.1.3 类是对象的蓝本 .....	70	6.13 接口与回调 .....	113
5.1.4 类的声明 .....	71	6.14 Cloneable 接口与	
5.1.5 类与类之间的关系 .....	73	Enumeration 接口 .....	121
5.2 面向对象 .....	73	习题 .....	124
5.2.1 对象的特征 .....	73	<b>第 7 章 Java 的常用类</b> .....	127
5.2.2 现实对象与软件对象 .....	74	7.1 String 类 .....	127
5.2.3 对象的作用 .....	74	7.1.1 String 类的构造函数 .....	127
5.2.4 对象的创建 .....	75	7.1.2 String 类的方法 .....	128
5.2.5 对象的封装 .....	77	7.1.3 字符串池 .....	130
5.3 HelloWorld 实例分析 .....	78	7.2 StringBuffer 类 .....	131
5.4 成员方法 .....	78	7.2.1 StringBuffer 类的构造函数 .....	131
5.4.1 void 返回方法 .....	79	7.2.2 StringBuffer 类的方法 .....	131
5.4.2 无参方法 .....	80	7.3 StringBuilder 类 .....	132
5.4.3 多参方法 .....	80	7.3.1 StringBuilder 类的构造函数 .....	132
5.5 局部变量和成员变量 .....	81	7.3.2 StringBuilder 类的方法 .....	133
5.6 静态变量与静态方法 .....	82	7.3.3 String 类、StringBuffer 类和	
5.7 包的定义与导入 .....	84	StringBuilder 类的效率比较 .....	133
5.8 访问控制符 .....	85	7.4 Math 类 .....	134
5.9 重载 .....	86	7.5 BigInteger 类 .....	136
5.10 类的实例化 .....	87	7.6 Arrays 类 .....	137
5.11 静态块和实例块 .....	88	7.7 Date 类 .....	139
习题 .....	90	7.8 Locale 类 .....	140
<b>第 6 章 Java 的类和对象(下)</b> .....	91	7.9 Random 类 .....	142
6.1 继承的概念 .....	91	7.10 Calendar 类 .....	143
6.2 子类对象的创建 .....	93	7.11 Java 执行其他的程序 .....	144
6.3 this 与 super .....	94	习题 .....	145

第 8 章 Java 的集合框架.....	146
8.1 Collection 接口 .....	147
8.1.1 常用方法 .....	147
8.1.2 迭代器 .....	150
8.2 List 接口.....	152
8.2.1 常用方法 .....	152
8.2.2 实现原理 .....	156
8.3 Map 接口.....	158

8.3.1 概述 .....	158
8.3.2 常用方法 .....	158
8.3.3 Comparable 接口 .....	163
8.3.4 实现原理 .....	163
8.4 Set 接口.....	165
8.5 集合框架中常用类的区别 .....	168
习题 .....	168

## 实 战 篇

实战 S1 Java 开发环境平台搭建.....	172
S1.1 实战指导 .....	172
S1.2 知识分析 .....	172
S1.3 拓展应用 .....	172

S3.1 实战任务的引入.....	185
S3.2 知识背景.....	185
S3.3 实战任务的实现.....	186
S3.4 拓展应用 .....	191

实战 S2 利用 Java 循环和分支 结构开发万年历 .....	174
S2.1 实战任务的引入.....	174
S2.2 知识背景.....	174
S2.3 实战任务的实现.....	175
S2.4 拓展应用 .....	182

实战 S4 利用 Java 综合知识开发 MINI 音乐管理系统.....	199
S4.1 实战任务的引入.....	199
S4.2 知识背景.....	199
S4.3 实战任务的实现.....	201
S4.4 拓展应用 .....	208

实战 S3 利用 Java 的类和 对象开发猜拳游戏 .....	185
-------------------------------------	-----

致谢 .....	216
----------	-----

# 理论篇



# 第 1 章 开发简单 Java 应用程序

## 本章要点

- ✓ Java 语言的发展过程
- ✓ Java 的工作原理及特点
- ✓ Java SDK 的安装与配置
- ✓ Eclipse 的安装与使用
- ✓ Java 程序开发

## 1.1 Java 语言的发展过程

Java 语言最早诞生于 1991 年,刚开始它只是 Sun 公司为一些消费性电子产品所设计的通用环境。因为当时 Java 的应用对象只限于 PDA、电子游戏机、电视机顶盒之类的消费性电子产品,所以并未被众多的编程技术人员所接受。

在 Java 出现以前,Internet 上的信息内容都是一些静态的 HTML 文档。正是因为 Web 中看不到交互式的内容,所以人们很不满意当时的 Web 浏览器,他们迫切希望能够在 Web 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 Java 带来了前所未有的施展舞台。

Sun 的工程师从 1994 年起将 Java 技术应用于 Web 上,并且开发出了 HotJava 的第一个版本。于是,Java 的名字逐渐变得广为人知。

Java 在 Sun World95 中正式发布。“一次编写,处处运行”的特点使得用 Java 技术开发的软件不用修改或重新编译就可以直接应用于任何计算机上,并使 Java 得到了广泛的关注。

从此以后,随着网络的快速发展,Java 成为应用最广泛的程序语言。一时间使用 Java 技术进行软件开发成为广大技术人员的一种时尚。到 2018 年 6 月为止,Java 已经发布了一系列的版本,目前最新版本为 JDK10.0.1(1.7.0)预览版。

## 1.2 Java 的工作原理及特点

Java 是一种高级的、通用的、面向对象的、适用于网络环境的程序设计语言,同时它



又是一种计算平台，为程序的运行提供了一个统一通用的环境，并屏蔽底层的操作系统及硬件环境的差异性。

### 1.2.1 Java 程序的处理过程

一个 Java 程序的运行必须经过编写、编译、运行三个步骤。

编写是指在某个 Java 语言开发环境中进行程序代码的输入与编辑，最终形成后缀名为“.java”的 Java 源文件。

编译是指使用 Java 编译器(javac 命令)把源文件翻译成二进制代码的过程。这期间也进行语法级别错误和引用错误的排查，编译后将生成后缀名为“.class”的字节码文件。该字节码文件并不是一个可以直接运行的文件。

运行是指使用 Java 解释器(java 命令)将字节码文件翻译成机器代码，执行并得到运行结果。

Java 程序流程如图 1-1 所示。

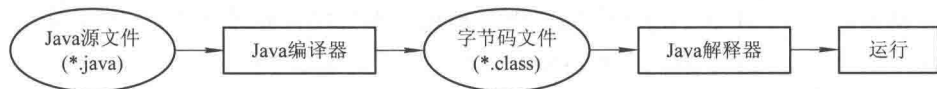


图 1-1 Java 程序运行流程

### 1.2.2 Java 字节码文件

字节码文件是一种和任何具体机器环境及操作系统环境无关的中间代码。Java 字节码文件是一种二进制文件，是 Java 源文件由 Java 编译器编译后生成的目标代码文件。编程人员和计算机都无法直接读懂字节码文件，它必须由专用的 Java 解释器来解释执行，因此 Java 是一种在编译基础上进行解释运行的语言。

### 1.2.3 Java 虚拟机

Java 程序不能直接运行在现有的操作系统平台上，它必须运行在被称为 Java 虚拟机 (Java Virtual Machine, JVM) 的软件平台之上。

JVM 是驻留于计算机内存的虚拟计算机或逻辑计算机，实际上是一段负责解释并执行 Java 字节码的程序。Java 解释器作为 JVM 的一部分，负责将字节码文件翻译成对应于具体硬件环境和操作系统平台的机器代码，以便执行。

在运行 Java 程序时，首先会启动 JVM，然后由它来负责解释执行 Java 的字节码文件，并且 Java 字节码文件只能运行于 JVM 之上。这样利用 JVM 就可以把 Java 字节码程序和具体的硬件平台以及操作系统环境分隔开来，只要在不同的计算机上安装了针对于特定具体平台的 JVM，Java 程序就可以运行，而不用考虑当前具体的硬件平台及操作系统环境，也不用考虑字节码文件是在何种平台上生成的。JVM 把这种不同软硬件平台的具体差别隐藏起来，从而实现了真正的二进制代码级的跨平台移植。JVM 是 Java 平台无关的基础，Java 的跨平台特性正是通过在 JVM 中运行 Java 程序体现的。Java 的这种运行机制如图 1-2 所示。

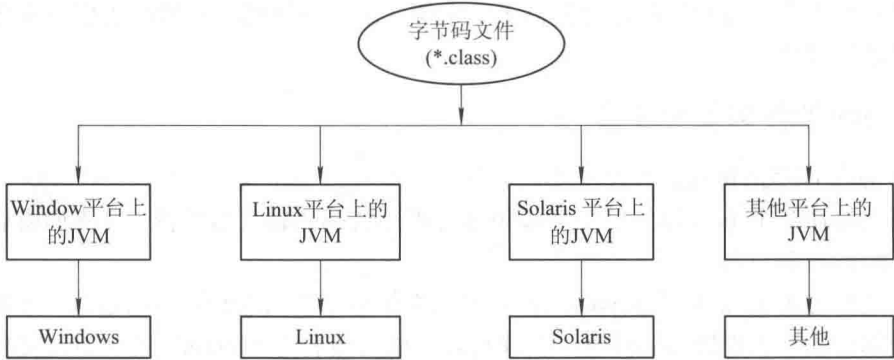


图 1-2 Java 运行机制

Java 语言这种“一次编写，处处运行”的方式，有效地解决了目前大多数高级程序设计语言需要针对不同系统编译产生不同机器代码的问题，即硬件环境和操作平台的异构问题，大大降低了程序开发、维护和管理开销。需要注意的是，Java 程序通过 JVM 可以达到跨平台特性，但 JVM 是不跨平台的。也就是说，不同操作系统之上的 JVM 是不同的，Windows 平台上的 JVM 不能用在 Linux 平台上，反之亦然。

### 1.2.4 垃圾回收

Java 虚拟机使用两个独立的堆内存，分别用于静态内存分配和动态内存分配。其中一个是非垃圾回收堆内存，用于存储所有类的定义、常量池和方法表。另一个堆内存再分为两个可以根据要求往不同方向扩展的小块。垃圾回收的算法适用于存放在动态堆内存中的对象。垃圾回收器将在回收对象实例之前调用 finalize() 方法。即使显式调用垃圾回收方法 (System.gc())，也不能保证立即运行，这是因为垃圾回收线程的运行优先级很低，经常会被中断。

## 1.3 Java SDK 的安装与配置

若要编写 Java 程序，就需要相应的开发工具。现在可用于开发 Java 程序的工具有很多，Java SDK 是 Sun 公司(现被 Oracle 收购)提供的免费开发工具集。

Java SDK(Java Software Development Kit)即 Java 软件开发工具包，也称为 JDK。截至 2018 年 6 月，提供下载的 Java SDK 标准版软件最新版本为 10.0.1，不同的操作系统有不同的版本。下面介绍在 Windows XP 操作系统下安装和配置 Java SDK 的过程。

### 1.3.1 下载并安装 Java SDK 开发工具

Java SDK 目前有以下几个版本：

Java SE(Java Platform, Standard Edition): Java 平台标准版，提供基础 Java 开发工具、执行环境与 API(Application Program Interface)。

Java ME(Java Platform, Micro Edition): Java 平台微型版，适用于消费性电子产品，提供嵌入式系统所使用的 Java 开发工具、执行环境与 API。



Java EE(Java Platform, Enterprise Edition): Java 平台企业版, 它是由 Sun 公司所提出的一组技术规格, 规划企业用户以 Java 技术开发、分发、管理多层式应用结构。

学习 Java 语言和进行一般的应用开发, 使用 Java SE 版本就足够了。为了运行的稳定性, 本书采用的是 Java SDK 1.6 版本。可以从 Sun 公司的网站上下载该版本, 下载的网址是: <http://java.sun.com/javase/downloads/index.jsp>(图 1-3 所示为下载页面)。

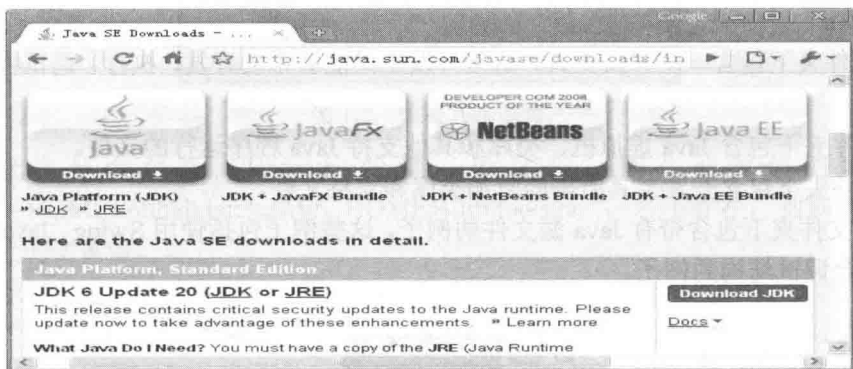


图 1-3 JDK 下载页面

进入下载页面后, 按网页的提示进行操作, 下载后的文件名称类似于 `jdk-6u10-rc2-bin-b32-windows-i586-p-12_sep_2008.exe`, 不同更新版本的文件名可能有差别。具体安装步骤如下:

(1) 双击运行下载的安装文件。在弹出的关于许可证协议的对话框中, 单击“接受”按钮, 接受许可证协议, 否则不能安装。

(2) 在弹出的“自定义安装”对话框中, 选择 JDK 的安装路径, 如图 1-4 所示。单击“更改”按钮可更改安装路径。为加快安装速度, Java DB、公共 JRE、源代码等选项可不安装, 但开发工具选项是必需的。

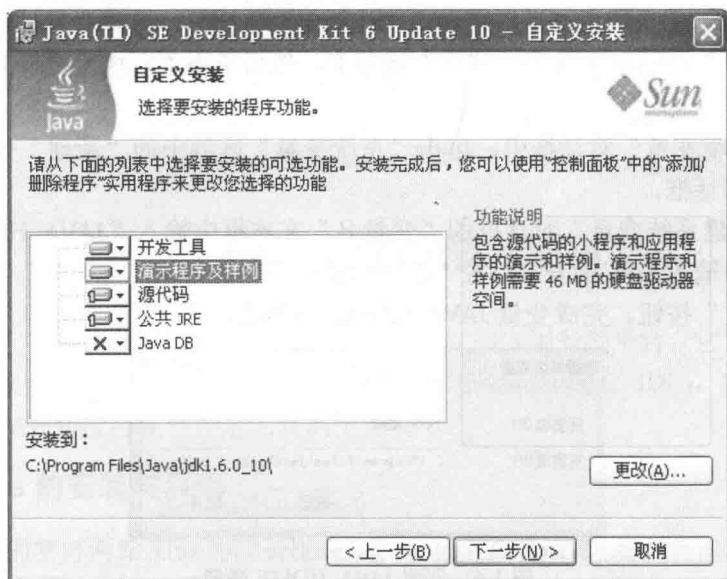


图 1-4 选择 JDK 安装路径



(3) 单击图 1-4 中的“下一步”按钮,开始安装。如果在图 1-4 中选择了安装“公共 JRE”,则在安装的过程中还会弹出另一个“自定义安装”对话框,提示用户选择 Java 运行时环境的安装路径,其操作过程与图 1-4 所示的界面类似。

(4) 单击“完成”按钮完成 JDK 的安装。

在默认情况下,安装后会在 C 盘“Program Files”文件夹下产生如图 1-5 所示的文件夹结构,其中:

bin 文件夹下包含一些开发工具,这些开发工具能够帮助开发、执行、调试以及文档化 Java 程序。

jre 文件夹下包含 Java 虚拟机、类库和其他支持 Java 程序运行的文件。

lib 文件夹下包含开发工具所需的附加类库和支持文件。

demo 文件夹下包含带有 Java 源文件的例子,这些例子包括使用 Swing、Java 的基础类和 Java 平台调试结构的例子。



图 1-5 Java SDK 安装文件夹结构

### 1.3.2 JDK 的配置与测试

JDK 安装完成后,需要设置环境变量及测试 JDK 配置是否成功,具体步骤如下:

(1) 在 Windows 系统桌面上右键单击“我的电脑”图标,选择“属性”菜单项。在弹出的“系统属性”对话框中选择“高级”选项卡,然后单击“环境变量”按钮,弹出“环境变量”对话框。

(2) 在“环境变量”对话框中,单击“系统变量”区域中的“新建”按钮,弹出“新建系统变量”对话框。

(3) 在“新建系统变量”对话框的“变量名”文本框中输入“JAVA\_HOME”,在“变量值”文本框中输入 JDK 的安装路径“C:\Program Files\Java\jdk1.6.0\_10”,如图 1-6 所示。最后单击“确定”按钮,完成变量 JAVA\_HOME 的创建。

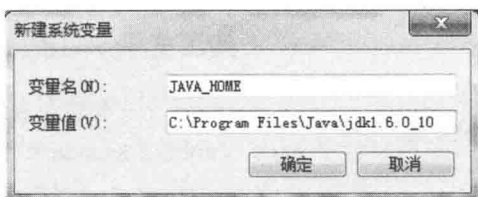


图 1-6 创建 JAVA\_HOME 变量

(4) 查看是否存在 Path 变量。若存在,则编辑该变量,在变量值后加入“;%JAVA\_



HOME%\bin”，如图 1-7 所示；若不存在，则新建该变量，并设置变量值为“%JAVA\_HOME%\bin;”。

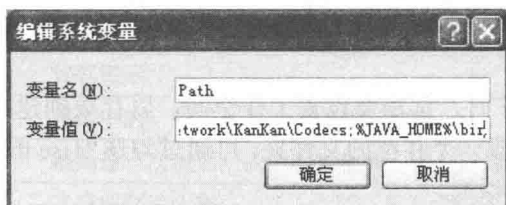


图 1-7 编辑 Path 变量

(5) 查看是否存在 Classpath 变量。若存在，则在变量值的后面加入“.;%JAVA\_HOME%\lib\dt.jar;%JAVA\_HOME%\lib\tools.jar”；若不存在，则创建该变量，并设置同上的变量值。

(6) 测试 JDK 的安装和配置是否成功。依次单击“开始”按钮，单击“运行”菜单项，在弹出的“运行”对话框中输入“cmd”命令，进入命令提示符窗口。进入任意目录下输入“javac”命令，按“Enter”键执行该命令，系统会输入 javac 命令的使用帮助信息，如图 1-8 所示。这说明 JDK 安装配置成功，否则需要检查上面各步骤的操作是否正确。

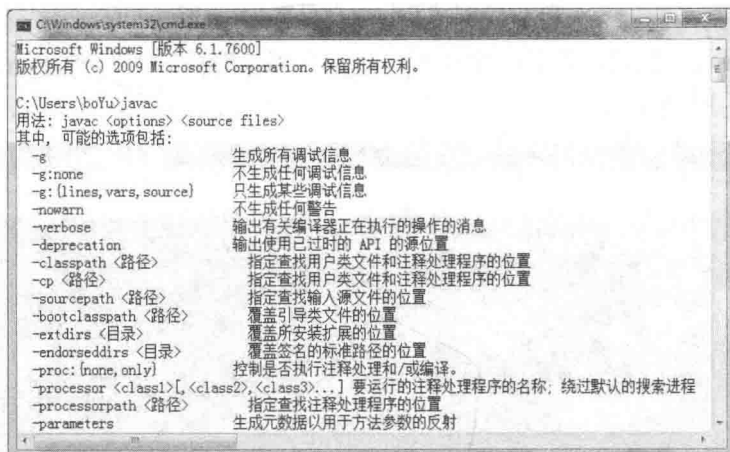


图 1-8 显示 javac 命令的使用帮助

## 1.4 Eclipse 的安装与使用

Eclipse 是一个基于 Java 的、开放源码的、可扩展的应用开发平台，它为编程人员提供了一流的 Java 集成开发环境(Integrated Development Environment, IDE)。它也是一个可以用于构建集成 Web 和应用程序开发工具的平台。

### 1.4.1 Eclipse 的安装与启动

在 Eclipse 的官方网站 <http://www.eclipse.org> 可下载 Eclipse 的最新版本，下载后的文件一般为一个 zip 格式的压缩文件。



- (1) 将下载后的压缩文件解压后，双击 eclipse.exe 文件就可启动 Eclipse。
- (2) 解压完成后，启动的 Eclipse 是英文版的，可以通过安装 Eclipse 的多国语言包来实现 Eclipse 的本地化。多国语言包可从 Eclipse 官方网站下载。具体的 Eclipse 汉化步骤请看相关文献。
- (3) 每次启动 Eclipse 时，都需要设置工作空间，以存放创建的项目。如图 1-9 所示，单击“Browse”按钮可选择一个存在的文件夹，可通过勾选“Use this as the default and do not ask again”选项屏蔽该对话框。

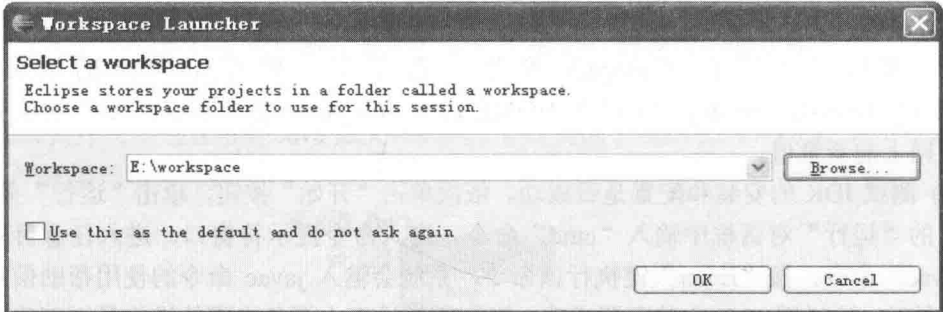


图 1-9 启动 Eclipse 时设置工作空间

- (4) 单击“OK”按钮，若是初次进入，则在第(3)步选择的工作空间会出现 Eclipse 欢迎界面，如图 1-10 所示。

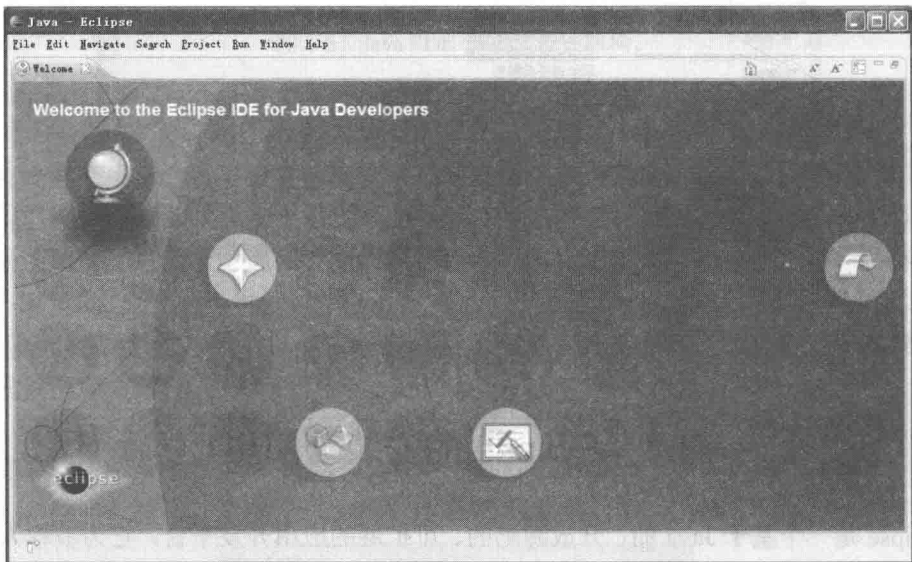


图 1-10 Eclipse 的欢迎界面

## 1.4.2 Eclipse 的使用

### 1. Eclipse3.2 开发工具的常用快捷键

如表 1-1 所示，熟悉 Eclipse3.2 开发工具的常用快捷键可大幅提高效率。



表 1-1 Eclipse3.2 开发工具的常用快捷键

快捷键	功能
F3	跳转到类或变量的声明处
Alt + /	代码提示
Alt + 上下方向键	将选中的一行或多行向上或向下移动
Alt + 左右方向键	跳到前一次或后一次的编辑位置，在代码跟踪时用得比较多
Ctrl + /	注释或取消注释
Ctrl + D	删除光标所在行的代码
Ctrl + K	将光标停留在变量上，按该快捷键可查找下一个相同变量
Ctrl + Q	回到最后编辑的位置
Ctrl + Shift + K	和 Ctrl+K 键查找的方向相反
Ctrl + Shift + X	将所选字符转为大写
Ctrl + Shift + Y	将所选字符转为小写
Ctrl + Shift + /	注释代码块
Ctrl + Shift + \	取消注释代码块
Ctrl + Shift + M	导入未引用的包
Ctrl + Shift + D	在 debug 模式中显示变量值
Ctrl + Shift + T	查找工程中的类
Ctrl + Shift + Down	复制光标所在行至其下一行
鼠标双击括号(小括号、中括号、大括号)	将选择括号内的所有内容

## 2. 使用 Eclipse 开发简单的 Java 应用程序

下面给出使用 Eclipse 开发一个简单的 Java 应用程序的例子，步骤如下：

(1) 启动 Eclipse，弹出如图 1-9 所示的对话框，通过该对话框选择一个工作空间，然后单击“OK”按钮进入 Eclipse 开发界面，如图 1-11 所示。

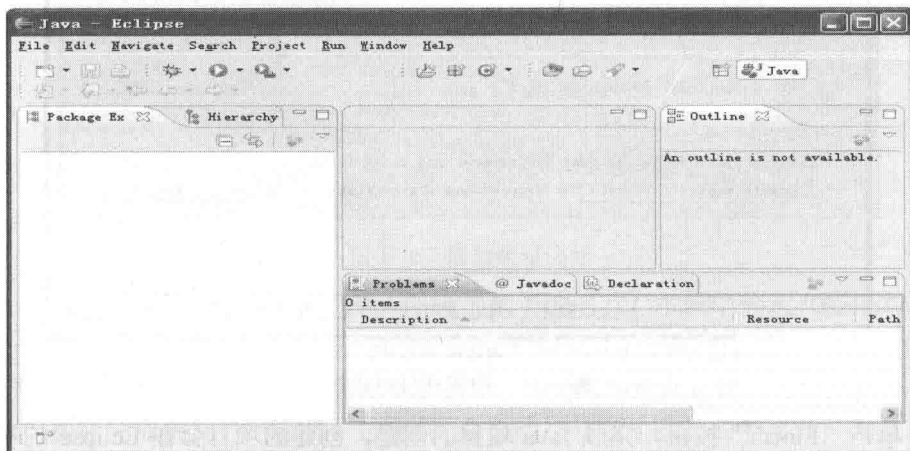


图 1-11 Eclipse 开发界面