

实例讲解
实训强化
培养技能
面向就业

全国高等职业教育计算机类规划教材·实例与实训教程系列

Java基础案例教程

◎马力 编著



- ◆ 编者具有多年软件开发项目经验，并在教学一线有丰富经验和积累
- ◆ 全书共20个典型案例，每章内容围绕几个典型案例展开
- ▶ 案例讲述通过“任务描述”、“任务分析”、“任务实现”和“相关知识”完成
- ▶ 每小节通过穿插大量针对性强的“练一练”来强化“知识点”的学习和掌握



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

全国高等职业教育计算机类规划教材·实例与实训教程系列

Java 基础案例教程

马 力 编著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书根据高职高专教育特点,以着重体现“理论性、实用性、技术性”三者相结合的编写特色,引导读者快速、高效地进入 Java 编程世界。

全书共分 10 章,采用循序渐进的方式,介绍面向对象的基本概念和相关技术,并对 Java 语言进行概述,围绕抽象性、封装性、继承性、多态性以及 I/O、异常等内容由浅入深地进行讲解,并介绍运用 Java 语言及其类库在 J2SE 平台下进行图形用户界面编程、网络编程、数据库编程等内容。全书内容丰富,通俗易懂,通过大量精选的实例,使读者逐步掌握 Java 应用程序开发方法和技巧。每章内容均围绕几个典型案例展开,案例讲述包括“任务描述”、“任务分析”、“任务实现”和“相关知识”,并通过大量针对性强的“练一练”来强化“知识点”的学习和掌握。书后附有“复习与训练”供读者练习和提高。

本书可作为高职高专院校相关专业 Java 课程的教材,也可作为 Java 编程爱好者的自学指导书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

Java 基础案例教程 / 马力编著. —北京: 电子工业出版社, 2010.6

全国高等职业教育计算机类规划教材. 实例与实训教程系列

ISBN 978-7-121-10987-4

I. ①J… II. ①马… III. ①JAVA 语言—程序设计—高等学校: 技术学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2010) 第 099824 号

责任编辑: 左 雅 文字编辑: 裴 杰

印 刷: 北京市海淀区四季青印刷厂

装 订: 涿州市桃园装订有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1092 1/16 印张: 16.75 字数: 429 千字

印 次: 2010 年 6 月第 1 次印刷

印 数: 4 000 册 定价: 28.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zllts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

前 言

Java 语言是近年来最流行的计算机程序设计语言，“Java 语言程序设计”课程在高校的程序设计课程中占重要的地位。编写一本适合高职高专层次的学生快速、高效地进入 Java 编程世界，是本书的编写目标。

本书系电子工业出版社高职高专计算机类信息技术系列教材之一，主要面向计算机及相关专业的学生，也适合非学历教育和计算机爱好者自学。作为一本针对初学者特点而编写的入门教材，在内容安排上力求循序渐进、由浅入深。本书编者是具有多年软件开发项目经验，并且在教学一线的教师，编写时打破传统教材的编写套路，着重体现“理论性、实用性、技术性”三者相结合的编写特色，注意抓住关键、突出重点、分解难点。目的不仅在于讲授知识，更注重培养学习者独立发现问题、分析问题、解决问题的能力 and 进行知识迁移的能力。全书共有 20 个典型案例，每章内容均围绕几个典型案例展开，案例讲述包括“任务描述”、“任务分析”、“任务实现”和“相关知识”四部分内容。其中，“相关知识”部分以小节的形式组织案例所涉及的理论知识，每小节内容中又通过穿插大量针对性强的“练一练”来强化“知识点”的学习和掌握，并根据内容需要，适时加入“案例与知识拓展”。每章后附有“复习与训练”供读者练习和提高。所有实例经精心安排，既有单独成立的，又有前后关联的，在按序介绍内容的同时，逐步形成一个包含基本功能的应用系统。

本书是辽宁省精品课《Java 语言程序设计》历经三年建设，并作为国家示范建设重点专业——软件技术专业课程体系下的核心专业基础课的配套教材，已经配备了非常丰富的立体化网络教学资源库，包括各类教学实训平台、资源库（教学标准、主要教学内容、电子课件，以及案例等）、素材库等可在华信教育资源网（www.hxedu.com.cn）免费下载。精品课程的网址为：[HTTP://218.25.74.218/jpk/08/java/](http://218.25.74.218/jpk/08/java/)。

本书由马力副教授编写。王永江、于波、张俊、王征等老师对本书的编写给予很多帮助，在此一并表示感谢。

由于时间仓促，编者水平有限，书中难免存在错误和不妥之处，敬请广大读者不吝赐教。谢谢！

编 者

序

20世纪90年代以来,以计算机和通信技术为推动力的信息产业在我国获得前所未有的发展,全国各企事业单位对信息技术人才求贤若渴,高等教育计算机及相关专业毕业生供不应求。随后几年,我国各高等院校、众多培训机构相继开设计算机及相关专业,积极扩大招生规模,不久即出现了计算机及相关专业毕业生供大于求的局面。纵观近十年的就业市场变化,计算机专业毕业生经历了“一夜成名、求之不得”的宠幸,也遭遇了“千呼百应、尽失风流”的冷落。

这个时代深深地镌刻着信息的烙印,这个时代是信息技术人才尽情展示才能的舞台。目前我国的劳动力市场,求职人数过剩,但满足企业要求的专业人才又很稀缺。这种结构性的人才市场供求矛盾是我国高等教育亟待解决的问题,更是“以人为本,面向人人”为目标的职业教育不可推卸的责任。

电子工业出版社,作为我国出版职业教育教材最早的出版社之一,是计算机及相关专业高等职业教材重要的出版基地。多年来,我们一直在教材领域为战斗在职业教育第一线的广大职业院校教育工作者贡献着我们的力量,积累了丰富的职业教育教材出版经验。目前,计算机专业高等教育正处于发展中的关键时期,我们有义务、有能力协同全国各高等职业院校,共同探寻适合社会发展需要的人才培养模式,建设满足高等职业教育需求的教学资源——这是我们出版“全国高等职业教育计算机类规划教材·实例与实训教程系列”的初衷。

关于本系列教材的出版,我们力求做到以下几点:

(1) 面向社会人才市场需求,以培养学生技能为目标。工学结合、校企合作是职业教育发展的客观要求,面向就业是职业教育的根本落脚点。本系列教材内容体系的制定是广大高职教育专家、一线高职教师共同智慧的结晶。我们力求教材内容丰富而不臃肿、精简而不残缺,实用为主、够用为度。

(2) 面向高职学校教师,以方便教学为宗旨。针对每个课程的教学特点和授课方法,我们为其配备相应的实训指导、习题解答、电子教案、教学素材、阅读资料、程序源代码、电子课件、网站支持等一系列教学资源,广大教师均可从华信教育资源网(www.huaxin.edu.cn)免费获得。

(3) 面向高职学校学生,以易学、乐学为标准。以实例讲述理论、以项目驱动教学是本系列教材的显著特色。这符合现阶段我国高职学生的认知规律,能够提高他们的学习兴趣,增强他们的学习效果。

这是一个崭新的开始,但永远没有尽头。高等职业教育教材的建设离不开广大职业教育工作者的支持,尤其离不开众多高等职业院校教师的支持。我们诚挚欢迎致力于职业教育事业发展的有识之士、致力于高等职业教材建设的有才之士加入到我们的队伍中来,多批评,勤点拨,广结友,共繁荣,为我国高等职业教育的发展贡献我们最大的力量!

电子工业出版社 职业教育分社

目 录

第 1 章 认识 Java 语言	(1)
1.1 Java 简介	(1)
1.1.1 Java 是什么	(1)
1.1.2 Java 语言的应用前景	(2)
1.1.3 Java 语言的特点	(2)
1.1.4 Java 虚拟机	(3)
1.1.5 Java 的程序类型	(4)
1.1.6 Java 的开发运行环境	(4)
1.2 搭建 Java 开发环境	(4)
1.2.1 JDK 下载	(4)
1.2.2 JDK 安装	(5)
1.2.3 配置环境变量	(6)
1.3 编写第一个 Java 程序	(7)
复习与训练	(10)
第 2 章 Java 基础知识	(11)
案例 1 员工个人信息	(11)
2.1 数据类型	(12)
2.2 变量及赋值	(15)
2.3 基本语言符号	(19)
2.3.1 标识符	(19)
2.3.2 关键字和保留字	(20)
2.4 类型转换	(22)
案例 2 球体的体积	(25)
2.5 常量	(26)
案例 3 利用海伦公式计算三角形面积	(26)
2.6 运算符和表达式	(27)
2.6.1 算术运算符	(28)
2.6.2 关系运算符	(29)
2.6.3 逻辑运算符	(31)
2.6.4 赋值运算符	(32)
2.6.5 三元运算符	(32)
2.6.6 表达式数据类型	(33)
复习与训练	(34)
第 3 章 控制流语句	(36)
案例 4 闰年的判断	(36)
3.1 顺序与分支结构	(37)

案例 5 水仙花数	(41)
3.2 循环结构	(43)
3.2.1 循环语句	(43)
3.2.2 跳转语句	(45)
复习与训练	(46)
第 4 章 数组	(48)
案例 6 评委打分	(48)
4.1 一维数组	(50)
4.1.1 数组的定义与创建	(50)
4.1.2 数组的内存模型	(51)
4.1.3 数组的遍历	(51)
4.1.4 数组的复制	(52)
4.1.5 数组排序	(52)
案例 7 矩阵的乘法	(53)
4.2 多维数组	(55)
4.2.1 二维数组	(55)
4.2.2 多维数组	(57)
案例 8 字符串的使用	(58)
4.3 字符串与 String 类	(60)
4.4 数组工具类	(62)
复习与训练	(64)
第 5 章 Java 面向对象基础	(65)
案例 9 学生成绩榜	(65)
5.1 类的定义	(69)
5.2 实例变量	(74)
5.2.1 构造器	(74)
5.2.2 实例变量	(75)
5.3 方法的定义	(76)
5.3.1 方法的使用	(76)
5.3.2 返回值	(78)
5.4 对象的创建与使用	(78)
5.4.1 new 操作符	(78)
5.4.2 this 关键字	(81)
5.5 方法重载	(82)
案例 10 单子类(单例模式)	(84)
5.6 静态变量与静态方法	(85)
复习与训练	(88)
第 6 章 面向对象高级特性	(91)
案例 11 大象放冰箱(模板模式)	(91)
6.1 继承	(94)

6.1.1	继承的特点	(94)
6.1.2	子类实例化过程	(96)
6.1.3	this 与 super	(97)
6.2	方法覆盖	(98)
6.3	final	(98)
6.4	包	(99)
6.4.1	定义包	(100)
6.4.2	导入包	(100)
6.4.3	访问控制符	(101)
6.5	封装	(102)
6.6	多态	(103)
6.6.1	多态表现形式	(103)
6.6.2	对象的归属	(105)
6.7	抽象类	(105)
案例 12	USB 设备的执行过程	(108)
6.8	接口	(110)
6.8.1	接口的基本语法	(110)
6.8.2	接口的继承	(111)
6.9	内部类	(116)
6.10	泛型	(119)
6.10.1	规则和限制	(120)
6.10.2	逐渐深入泛型	(122)
	复习与训练	(124)
第 7 章	Java 常用类库	(129)
案例 13	企业员工管理	(129)
7.1	lang 包	(136)
7.1.1	包装类	(136)
7.1.2	StringBuffer 类	(138)
7.1.3	Math 类	(139)
7.1.4	Class 类	(140)
7.1.5	Object 类	(141)
7.2	util 包	(148)
7.2.1	Date 类	(148)
7.2.2	Calendar 类	(150)
7.2.3	Random 类	(151)
7.3	单列集合 Collection 接口、List 接口、Set 接口、集合工具类 Collections	(152)
7.3.1	集合框架概述	(152)
7.3.2	Collection 接口	(153)
7.3.3	List 接口	(153)
7.3.4	Set 接口	(157)

7.3.5 Collections	(159)
7.4 键值双列集合 Map	(161)
7.5 泛型的高级应用	(164)
案例 14 用 Java 程序实现文件复制	(165)
7.6 File 类	(169)
7.7 字节流 InputStream 抽象类、OutputStream 抽象类	(171)
7.8 字符流 Reader、Writer	(173)
复习与训练	(177)
第 8 章 异常处理	(178)
案例 15 论坛用户注册	(178)
8.1 异常简介	(181)
8.2 异常的分类	(182)
8.3 异常的处理	(184)
复习与训练	(186)
第 9 章 GUI 编程	(187)
案例 16 计算器	(187)
9.1 AWT (Abstract Window Toolkit)	(193)
9.1.1 容器 (Container)	(194)
9.1.2 布局管理器	(194)
9.1.3 Component	(196)
案例 17 图片浏览器	(204)
9.2 AWT 事件处理机制	(210)
案例 18 实体类代码生成器	(215)
9.3 Swing	(224)
9.3.1 JFrame	(225)
9.3.2 面板 JPanel	(226)
9.3.3 Swing 常用组件	(226)
复习与训练	(230)
第 10 章 多线程与网络编程	(231)
案例 19 英译汉考试系统	(232)
10.1 多线程	(240)
10.1.1 多线程的概念	(240)
10.1.2 线程的创建	(241)
10.1.3 线程的状态与生命周期	(242)
10.1.4 线程休眠和线程中断	(243)
10.1.5 多线程同步	(243)
案例 20 C/S 结构聊天室	(248)
10.2 网络编程	(252)
10.2.1 URL 通信	(252)
10.2.2 Socket 通信机制	(252)
复习与训练	(254)
参考文献	(255)

第1章 认识 Java 语言

Java 作为一种优秀的计算机程序设计语言，由于其简单性以及平台无关性等优秀特性，已经成为软件开发的首选语言。通过本章的学习，使读者对 Java 有一个初步的认识，熟悉 Java 的开发环境。

本章主要内容：

- Java 简介
- Java 语言的特性
- Java 开发环境的搭建方法

1.1 Java 简介

1.1.1 Java 是什么

Java 是 Sun Microsystems 公司（简称 Sun 公司）推出的一门编程语言。Sun 公司对 Java 的定义是：一种简单、面向对象、分布式、跨平台、半编译半解释、健壮、安全、高性能、多线程和动态的语言。

它最初来自于 Sun 公司的一个叫 Green 的项目，其最初目的是为家用消费电子产品开发一个分布式代码系统，以便可以对电冰箱、电视机等家用电器进行控制，于是开发了一种新的语言 Oak。Oak 是一种用于网络的精巧而安全的语言，随着进军 Internet 而发展成了著名的 Java。几位 Java 成员组的会员在咖啡馆喝着 Java（爪哇）咖啡讨论给这个新的语言取什么名字，于是就叫 Java 了。

Java 自从 1995 年正式问世以来，它的快速发展让整个 Web 界发生了翻天覆地的变化。JSP 技术的推出，更让 Java 成为基于 Web 应用程序开发的首选工具。

以下是 Java 大事记。

1995 年 5 月 23 日，Java 语言诞生；

1996 年 1 月，第一个 JDK——JDK1.0 诞生；

1996 年 4 月，10 个最主要的操作系统供应商申明将在其产品中嵌入 Java 技术；

1996 年 9 月，约 8.3 万个网页应用了 Java 技术来制作；

1997 年 2 月 18 日，JDK1.1 发布；

1997 年 4 月 2 日，JavaOne 会议召开，参与者近一万人，创当时全球同类会议规模之最；

1997 年 9 月，JavaDeveloperConnection 社区成员超过十万；

1998 年 2 月，JDK1.1 被下载超过 2000000 次；

1998 年 12 月 8 日，Java 2 企业平台 J2EE 发布；

1999 年 6 月，Sun 公司发布 Java 的三个版本：标准版（J2SE）、企业版（J2EE）和微型版（J2ME）；

2000年5月8日, JDK1.3发布;

2000年5月29日, JDK1.4发布;

2001年6月5日, NOKIA宣布, 到2003年将出售1亿部支持Java的手机;

2001年9月24日, J2EE1.3发布;

2002年2月26日, J2SE1.4发布, 自此Java的计算能力有了大幅提升;

2004年9月30日 18:00PM, J2SE1.5发布, 成为Java语言发展史上的又一里程碑。为了表示该版本的重要性, J2SE1.5更名为Java SE 5.0;

2005年6月, JavaOne大会召开, Sun公司公开Java SE 6。此时, Java的各种版本已经更名, 以取消其中的数字“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JRE6.0。

根据应用范围的不同, Java分为三个版本: 即J2SE、J2EE、J2ME。

● J2SE (Java 2 Platform Standard Edition)

Java的标准版, 是桌面开发和低端商务应用的解决方案。包含了标准的JDK、开发工具、运行时环境和类库。它适用于桌面系统, 提供CORBA标准的ORB技术, 结合Java的RMI支持分布式操作系统。

● J2EE (Java 2 Platform Enterprise Edition)

Java的企业版, 是以企业为环境而开发应用程序的解决方案, 目前已经成为开发商创建电子商务应用的实施标准。

● J2ME (Java 2 Platform Micro Edition)

Java的小型版, 是致力于消费产品和嵌入式设备的最佳解决方案, 包含高度优化精简的Java运行时环境。它提供了HTTP高级Internet协议, 使移动电话能以Client/Server方式直接访问Internet的全部信息, 不同的Client访问不同的文件, 此外还能访问本地存储区, 提供最高效率的无线交流。

1.1.2 Java语言的应用前景

Java语言有着广泛的应用前景, 大体上可以从以下几个方面来考虑其应用:

- (1) 所有面向对象的应用开发, 包括面向对象的事件描述、处理、综合等;
- (2) 计算过程的可视化、可操作化软件的开发;
- (3) 动态画面的设计, 包括图形图像的调用;
- (4) 交互操作的设计(选择交互、定向交互、控制流程等);
- (5) Internet的系统管理功能模块的设计, 包括Web页面的动态设计、管理和交互操作设计等;
- (6) Intranet(企业内部网)上的软件开发(直接面向企业内部用户的软件);
- (7) 与各类数据库连接查询的SQL语句实现;
- (8) 其他应用类型的程序。

1.1.3 Java语言的特点

1. 简单

Java简单明了, 它的简单性主要体现在三个方面:

- (1) Java 的风格类似于 C++，因而 C++ 程序员可以很快就掌握 Java 编程技术；
- (2) Java 摒弃了 C++ 中容易引发错误的地方，如指针；
- (3) Java 提供了丰富的类库，给程序的开发带来了极大的方便。

2. 平台无关性

Internet 由数以万计的各种各样的计算机系统组成，这些系统上的软、硬件千差万别，应用软件要在网络上任何一种计算机系统中都能正常运行，就必须具有平台无关性。软件的平台无关性是软件发展的需要，也是编程人员追求的目标。Java 就是一种平台无关性的编程语言。在源程序级保证了其基本数据类型与平台无关；编译后产生的二进制代码是与具体机器指令集无关的指令集合，通过 Java 虚拟机 (Java Virtual Machine, JVM)，可以在不同平台上运行，即 JVM 隔离了不同的环境与平台，使安装了 JVM 的计算机可以运行 Java 程序，即一次编译，到处运行。

3. 面向对象

Java 是一种面向对象的程序设计语言。面向对象的编程技术使用一种被称为类 (class) 的软件对象，可以重用和扩展代码，使应用程序的开发变得容易。

4. 安全

Java 系统仔细检测对内存的访问，确认合法性，因此 Java 程序不可能造成计算机崩溃。所有程序都可能错误，但 Java 即使出现错误程序也不会崩溃，而是把该例外抛弃，程序会发现这类例外，并加以处理。

传统的程序可以访问计算机的全部内存。内存中的值可能被有意识地或者无意识地修改，这就会造成安全问题。而 Java 程序只能访问内存中允许它们访问的那些部分，因而在这方面是安全的。

除上述特点之外，Java 还具有体系结构中立、分布式、健壮、多线程、动态等特点，读者可以在以后的学习中逐渐体会。

1.1.4 Java 虚拟机

Java 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 虚拟机 (JVM) 是 Java Virtual Machine 的缩写，它是一个虚构出来的计算机，是通过在实际的计算机上仿真模拟各种计算机功能模拟来实现的。Java 虚拟机有自己完善的硬件架构，如处理器、堆栈、寄存器，还具有相应的指令系统。一般的高级语言如果要在不同的平台上运行，至少需要编译成不同的目标代码。而引入 Java 语言虚拟机后，Java 语言在不同平台上运行时不需要重新编译。Java 使用模式 Java 虚拟机屏蔽了与具体平台相关的信息，使得编译程序只需生成在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，把字节码解释成具体平台上的机器指令执行。

Java 虚拟机由五个部分组成：一组指令集、一组寄存器、一个栈、一个无用单元收集堆 (Garbage-collected-heap)、一个方法区域。这五部分是 Java 虚拟机的逻辑成分，不依赖任何实现技术或组织方式，但他们的功能必须在真实机器上以某种方式实现。

1.1.5 Java 的程序类型

Java 程序有两种类型：Java Application 和 Java Applet。这两类程序在组成结构和执行机制上都有一定的差异。

Java Application 即应用程序，是完整的程序，可完成任何计算任务，需要独立的 Java 解释器来解释运行。Java Applet 是可在 Web 网页上运行的，称为小应用程序，不能单独运行，必须嵌入到 HTML 语言编写的 Web 页面中，通过与 Java 兼容的浏览器来控制运行。考虑到网络环境、连接速度等原因，Applet 一般都比较小，适合客户端下载。由于 Java 语言的安全机制，用户一旦载入 Applet，就可以放心地来生成多媒体的用户界面或完成复杂的计算而不必担心病毒的入侵。虽然 Applet 可以和图像、声音、动画等一样从网络上下载，但它不同于这些多媒体的文件格式，它可以接收用户的输入，动态地进行改变，而不仅仅是动画的显示和声音的播放。

1.1.6 Java 的开发运行环境

Java 开发运行环境是指 Java 的开发工具和相应的软、硬件环境。目前有许多用于编写 Java 程序的开发工具，如 Sun 公司的 JDK 和 Java Workshop，Borland 公司的 JBuilder，IBM 公司的 Visual Age for Java，Microsoft 公司的 VJ++，Symantec 公司的 Visual Café 以及 Xinox Software 公司的 JCreator Pro 等。

1.2 搭建 Java 开发环境

Sun 公司提供了一套自己的 Java 开发环境，通常称之为 JDK (Java Development Kit)。Sun 公司提供了用于多种操作系统下的 JDK，随着时间的推移和技术的不断进步，JDK 的版本也在不断升级，如 JDK1.2，JDK1.3，JDK1.4，直至现在的 JDK1.6。Java 用户可以根据自己的使用环境到 Sun 公司的网站上下载相应的 JDK 版本。

JDK 这个 Java 开发工具包是 Java 开发者必须安装的软件环境，它包含了 JRE 和开发 Java 程序所需要的工具，如编译器、调试器、反编译器和文档生成器等。

JDK 这个 Java 开发工具包是 Java 开发者必须安装的软件环境，它包含了 JRE 和开发 Java 程序所需要的工具，如编译器、调试器、反编译器和文档生成器等。

JRE (Java Runtime Environment) 是 Java 运行时环境，包含了类库和 JVM (Java 虚拟机)，是 Java 程序运行的必要环境。如果只安装 JRE 而不安装 JDK，Java 程序也可以运行。

javac.exe 位于 JDK 下的 bin 目录下，是 Java 源文件的编译工具。Java 源文件的扩展名为 .java，如 HelloWorld.java，Java 源文件被编译后生成的字节码文件的扩展名为 .class，如 HelloWorld.class。

java.exe 也位于 JDK 下的 bin 目录下，负责解释执行 Java 字节码文件，就是一个 JVM。

1.2.1 JDK 下载

用户可以到 Sun 公司的网站上下载需要的 JDK 版本，网址是 <http://java.sun.com/>。

要注意的是，根据自己的环境选择相应的 JDK 版本。如图 1-1 所示，最上面的就是 Windows 环境下的 JDK，又分为离线安装包 (Windows Offline Installation, Multi-language)

和在线安装包（Windows Online Installation, Multi-language）。一般选择离线安装包。

Windows Platform - Java SE Development Kit 6 Update 5		
<input checked="" type="checkbox"/>		
<input checked="" type="checkbox"/>	↓ Windows Offline Installation, Multi-language	jdk-6u5-windows-i586-p.exe
<input checked="" type="checkbox"/>	↓ Windows Online Installation, Multi-language	jdk-6u5-windows-i586-p-iftw.exe
Linux Platform - Java SE Development Kit 6 Update 5		
<input checked="" type="checkbox"/>		
<input type="checkbox"/>	↓ Linux RPM in self-extracting file	jdk-6u5-linux-i586-rpm.bin
<input type="checkbox"/>	↓ Linux self-extracting file	jdk-6u5-linux-i586.bin
Solaris SPARC Platform - Java SE Development Kit 6 Update 5		
<input checked="" type="checkbox"/>		
<input type="checkbox"/>	↓ Solaris SPARC 32-bit self-extracting file	jdk-6u5-solaris-sparc.sh
<input type="checkbox"/>	↓ Solaris SPARC 32-bit packages - tar.Z	jdk-6u5-solaris-sparc.tar.Z
<input type="checkbox"/>	↓ Solaris SPARC 64-bit self-extracting file	jdk-6u5-solaris-sparcv9.sh
<input type="checkbox"/>	↓ Solaris SPARC 64-bit packages - tar.Z	jdk-6u5-solaris-sparcv9.tar.Z
Solaris x86 Platform - Java SE Development Kit 6 Update 5		

图 1-1 JDK 下载窗口

1.2.2 JDK 安装

将 JDK1.6 安装包下载并保存到硬盘上，文件名为 jdk-6u5-windows-i586-p.exe。执行该文件，按照向导安装即可。如果选择默认安装，会将 JDK 安装到如图 1-2 所示的路径下。如果想改变安装路径，单击“更改”按钮，在弹出的窗口中给出相应路径即可。

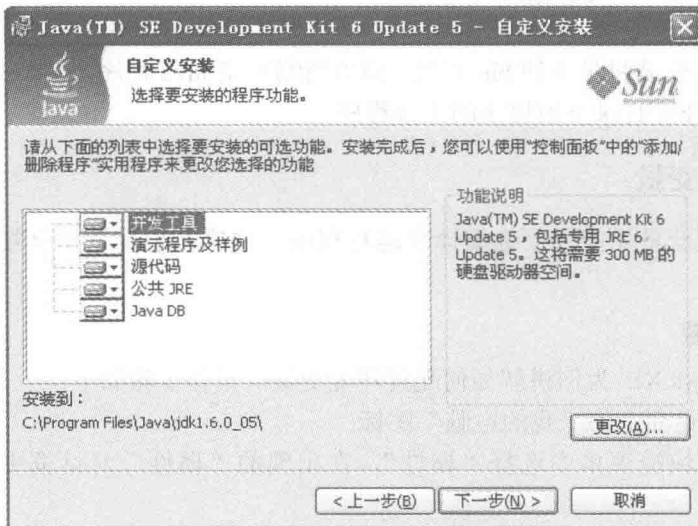


图 1-2 JDK 安装过程

如图 1-3 所示为 JDK 安装目录的内容。



图 1-3 JDK 安装目录的内容

- bin 目录

提供的是 JDK 的工具程序，包括 javac、java、javadoc、appletviewer 等程序。

- demo 目录

一些使用 Java 编写好的范例程序。

- jre 目录

JDK 自己附带的 JRE。

- lib 目录

工具程序实际上会使用的 Java 工具类（例如，javac 工具程序实际上会去使用 tools.jar 中的 com/sun/tools/javac/Main 类）。

- src.zip

Java 提供的 API 类的源代码压缩文件。如果将来需要查看 API 的某些功能是如何实现的，可以查看这个文件中的源代码内容。

在大致了解 JDK 与 JRE 安装目录下的内容之后，这里做个总结，我们到底要记得哪些内容？答案是 JDK 安装目录下的 bin 目录。因为当编写完 Java 程序之后，无论是编译或执行程序，都会使用到 bin 目录下所提供的工具程序。

1.2.3 配置环境变量

正确设置环境变量才能方便地编译和运行程序。通常需要配置的环境变量有两个，即 path 和 classpath。

1. path 的设置

下面以 Windows XP 为例讲解如何配置环境变量，具体步骤如下：

- (1) 右键单击桌面上的“我的电脑”图标；
- (2) 从弹出的快捷菜单中选择“属性”，在出现的“属性”对话框中单击“高级”选项卡；
- (3) 单击“环境变量”按钮，打开“环境变量”对话框；
- (4) 在“系统变量”选项组中，选中“Path”变量进行编辑，以本书中的操作为例，“Path”变量的值为 C:\Program Files\Java\jdk1.6.0_05\bin，如图 1-4 所示。



图 1-4 设置环境变量 Path

2. classpath 的设置

设置 classpath 的步骤与前面介绍的 Path 设置基本相同：

- (1) 右键单击桌面上的“我的电脑”图标；
- (2) 从弹出的快捷菜单中选择“属性”，在出现的“属性”对话框中单击“高级”选项卡；
- (3) 单击“环境变量”，打开“环境变量”对话框；
- (4) 在“系统变量”选项卡中，单击“新建”按钮，在弹出的“新建系统变量”对话框中输入变量名“classpath”，在变量值栏中输入 Java 源程序所在的文件路径，比如：D:\myjava，如图 1-5 所示。

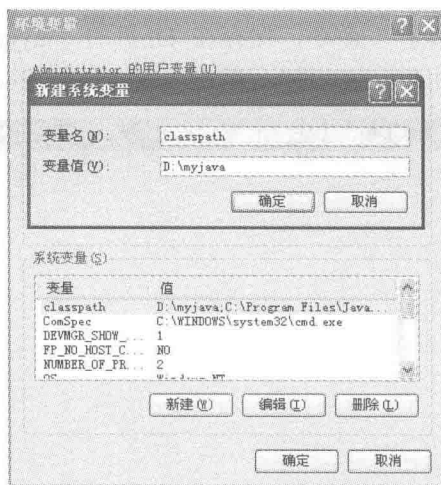


图 1-5 设置环境变量 classpath

1.3 编写第一个 Java 程序

任务描述

编写一个 Java 程序，命名为 Welcome.java。程序运行后，在“命令提示符”窗口中显示

“欢迎来到精彩的 Java 世界!”和“这是我的第一个 Java 程序!”,并用星号将两行文字围在中间,如图 1-6 所示。



图 1-6 “Welcome.java”运行结果

任务分析

该程序有 4 行输出,关键是需要 4 个屏幕输出语句即可以完成。

实现步骤

(1) 执行“开始”→“程序”→“附件”→“记事本”菜单命令;

(2) 在“记事本”中输入下列代码;

```
/*程序 Welcome.java: Java 中 main() 方法的用法*/
public class Welcome
{
    public static void main(String[] args)
    {
        System.out.println("*****");
        System.out.println("* 欢迎来到精彩的 Java 世界! *");
        System.out.println("* 这是我的第一个 Java 程序! *");
        System.out.println("*****");
    }
}
```

(3) 执行“文件”→“保存”命令,保存在 D 盘的“myjava”文件夹中,文件名输入“Welcome.java”,保存类型为“所有文件”。单击“保存”按钮即可,如图 1-7 所示。



图 1-7 文件保存界面

(4) 执行“开始”→“程序”→“附件”→“命令提示符”菜单命令,出现提示符界