

应用型本科院校计算机类专业校企合作实训系列教材

JAVA

程序设计

实验指导教程

主编 郑豪 杨种学 副主编 李青



南京大学出版社



013060456

TP312JA-33

10

应用型本科院校计算机类专业校企合作实训系列教材

JAVA程序设计实验指导教程

主 编 郑 豪 杨种学
副主编 李 青



北航

C1667772



南京大学出版社

TP312JA-33

10

图书在版编目(CIP)数据

JAVA 程序设计实验指导教程 / 郑豪, 杨种学主编. — 南京: 南京大学出版社, 2013. 3

应用型本科院校计算机类专业校企合作实训系列教材

ISBN 978 - 7 - 305 - 11406 - 9

I. ①J… II. ①郑…②杨… III. ①

JAVA 语言—程序设计—高等学校—教学参考资料 IV.

①TP312

中国版本图书馆 CIP 数据核字(2013)第 087408 号

出版发行 南京大学出版社

社 址 南京市汉口路 22 号 邮 编 210093

网 址 <http://www.NjupCo.com>

出版人 左 健

丛 书 名 应用型本科院校计算机类专业校企合作实训系列教材

书 名 JAVA 程序设计实验指导教程

主 编 郑 豪 杨种学

责任编辑 姚 萍 单 宁

责任校对 邓海琴

照 排 南京紫藤制版印务中心

印 刷 南京人文印务有限公司

开 本 787×1092 1/16 印张 10.25 字数 263 千

版 次 2013 年 3 月第 1 版 2013 年 3 月第 1 次印刷

ISBN 978 - 7 - 305 - 11406 - 9

定 价 20.50 元

发行热线 025 - 83594756

电子邮箱 Press@NjupCo.com

Sales@NjupCo.com(市场部)

* 版权所有, 侵权必究

* 凡购买南大版图书, 如有印装质量问题, 请与所购图书销售部门联系调换



北航

C1667772

013060428

林德民系即案非合全研业专类属其甘册编本通用立

应用型本科院校计算机类专业校企合作 实训系列教材编委会

主任委员:刘维周

副主任委员:张相学 徐 琪 杨种学(常务)

委 员(以姓氏笔画为序):

王小正 王江平 王 燕 田丰春 曲 波

李 朔 李 滢 闵宇峰 杨 宁 杨立林

杨蔚鸣 郑 豪 徐家喜 谢 静 潘 雷

甘肃出版集团

序 言

在当前的信息时代和知识经济时代,计算机科学与信息技术的应用已经渗透到国民生活的方方面面,成为推动社会进步和经济发展的重要引擎。

随着产业进步、学科发展和社会分工的进一步精细化,计算机学科新知识、新领域层出不穷,多学科交叉与融合的计算机学科新形态正逐渐形成。2012年,国家教育部公布的《普通高等学校本科专业目录(2012年)》中将计算机类专业分为计算机科学与技术、软件工程、网络工程、物联网工程、信息安全、数字媒体技术等专业。

随着国家信息化步伐的加快和我国高等教育逐步走向大众化,计算机类专业人才培养不仅在数量的增加,也在质量的提高上对目前的计算机类专业教育提出更为迫切的要求。社会需要计算机类专业的教学内容的更新周期越来越短,相应的,我国计算机类专业教育也将改革的目标与重点聚焦于如何培养能够适应社会经济发展需要的高素质工程应用型人才。

作为应用型地方本科高校,南京晓庄学院计算机类专业在多年实践中,逐步形成了陶行知“教学做合一”思想与国际工程教育理念相融合的独具晓庄特色的工程教育新理念。学生在社会生产实践的“做”中产生专业学习需求和形成专业认同,在“做”中增强实践能力和创新能力,在“做”中生成和创造新知识,在“做”中涵养基本人格和公民意识;同时学生应遵循工程教育理念,标准地“做”,系统地“做”,科学地“做”,创造地“做”。

实训实践环节是应用型本科院校人才培养的重要手段之一,是应用型人才培养目标得以实现的重要保证。当前市场上一些实训实践教材导向性不明显,可操作性不强,系统性不够,与社会生产实际联系不紧密。总体上来说没有形成系列,同一专业的不同实训实践教材重复较多,且教材之间的衔接不够。

《教育部关于“十二五”普通高等教育本科教材建设的若干意见(高教[2011]05号)》要求重视和发挥行业协会和知名企业在教材建设中的作用,鼓励行业协会和企业利用其具有的行业资源和人才优势,开发贴近经济社会实际的教材和高质量的实践教材。南京晓庄学院计算机类专业积极开展校企联合实训实践教材建设工作,与国内多家知名企业共同规划建设“应用型本科院校计算机类专业校企合作实训系列教材”。

本系列教材是在计算机学科和计算机类专业课程体系建设基本成熟的基础上,参考《中国计算机科学与技术学科教程 2002》(China Computing Curricula 2002,简称CCC2002)并借鉴ACM和IEEE CC2005课程体系,经过认真的市场调研,由我校优秀

教学科研骨干和行业企业专家通力合作而成的,力求充分体现科学性、先进性、工程性。

本系列教材在规划编写过程中体现了如下一些基本组织原则和特点:

1. 贯彻了“大课程观”、“大教学观”和“大工程观”的教学理念。教材内容的组织和案例的甄选充分考虑了复杂工程背景和宏大工程视野下的工程项目组织、实施和管理,注重强化具有团队协作意识、创新精神等优秀人格素养的卓越工程师的培养。

2. 体现了计算机学科发展趋势和技术进步。教材内容适应社会对现代计算机工程人才培养的需求,反映了基本理论和原理的综合应用,反映了教学体系的调整和教学内容的及时更新,注重将有关技术进步的新成果、新应用纳入教材内容,妥善处理了传统知识的继承与现代工程方法的引进。

3. 反映了计算机类专业改革和人才培养需要。教材规划以 2012 年教育部公布的新专业目录为依据,正确把握了计算机类专业教学内容和课程体系的改革方向。在教材内容和编写体系方面注重了学思结合、知行合一和因材施教,强化了以适应社会需要为目标的教学内容改革,由知识本位转向能力本位,体现了知识、能力、素质协调发展的要求。

4. 整合了行业企业的优质技术资源和项目资源。教材采用校企联合开发和建设的模式,充分利用行业专家、企业工程师和项目经理的项目组织、管理、实施经验的优势,将企业实际实施的工程项目分解为若干可独立执行的案例,注重了问题探究、案例讨论、项目参与式教育教学方式方法的运用。

5. 突出了应用型本科院校基本特点。教材内容以适应社会需要为目标,突出“应用型”的基本特色,围绕培养目标,以工程应用为背景,通过理论与实践相结合,重视学生工程应用能力的培养,增强学生技能的应用。

相信通过这套“应用型本科院校计算机类专业校企合作实训系列教材”的规划出版,能够在形式上和内容上显著提高我国应用型本科院校计算机类专业实践教材的整体水平,继而提高计算机类专业人才的培养质量,培养出符合经济社会发展需要和产业需求的高素质工程应用型人才。

李洪天

南京晓庄学院 党委书记 教授

前 言

Java 程序语言是目前计算机行业使用最为广泛的编程语言之一,相关就业岗位和市场需求都十分庞大。由于 Java 是实践性很强的课程,所以在 Java 的教学中需要实践性的实验指导书。但是很多 Java 实验指导书都是习题集式的教材,没有真正把 Java 教学实战化,和企业需求脱节。为深化教育教学改革,进一步推动专业综合改革,依据国家十二五规划,瞄准 Java 发展前沿,面向经济社会发展需求,借鉴国内外教学改革成果,充分利用信息技术,由高校一线教师和知名企业工程师一起编写了这本 Java 程序设计实验指导书。本书的特色如下:

一、项目驱动式实验模式

始终以项目实战为主线,深入浅出,把 Java 的基础知识融入到各个实验的项目中,各个实验先以小项目训练读者基本的技能,最后再以综合项目实战使用户深入理解。使读者真正地得到了项目实训,掌握了 Java 的程序开发,理解了 Java 的内在机制,也积累了项目开发经验。

二、企业工程师式实战模式

由于本书是校企合作的结晶,有一些知名企业的工程师参与编写,把企业中的需求和项目融进教材中,所以读者在使用过程中能真切地感受到工程化的思想,更加贴近实战,满足企业工作的需要,是 Java 教学中不可多得的实验指导书。

三、由简入繁渐进式教学模式

本书把 Java 程序设计分为 6 大模块,由此实验也针对性的分为 6 个大的实验,每一个实验章节针对一个 Java 的知识模块。从简单的 Java 基础知识开始,到类的继承、多态、封装,最后到一个大的综合性项目,使读者能够循序进展,渐进式的掌握 Java 的知识体系。

本书分为 7 个实验,实验 1 是初识 Java 程序设计,指导读者掌握 Java 的开发环境、基础语法等,由朱杰老师编写;实验 2 是面向对象之类的继承与封装,指导读者掌握类的继承性和封装性,由王洁老师编写;实验 3 是面向对象之类的多态性,指导读者掌握类的多态性,由王小正老师编写;实验 4 是面向对象之接口,指导读者掌握接口,并进一步深入了解多态性,由郑豪老师编写;实验 5 是图形用户界面设计,指导读者能够用 swing 组件编写图形用户界面,由侯青老师编写;实验 6 是多线程设计,指导读者编写多线程的程序,由李青老师编写;实验 7 是综合性实验,指导读者开发一个图书馆信息管理系统,由郑豪老师编写。

由于编者经验有限,书中难免有不足之处,敬请读者和同行们予以批评和指正。

目 录

实验 1 初识 Java 程序设计	1
1.1 知识点回顾	1
1.2 实验练习	3
1.2.1 实验任务 1	3
1.2.2 实验任务 2	7
1.2.3 实验任务 3	10
1.3 项目实战	12
1.3.1 项目描述	12
1.3.2 项目分析	12
1.3.3 项目编写	13
1.4 实验习题	15
实验 2 面向对象之类的封装和继承	17
2.1 知识点回顾	17
2.2 实验练习	23
2.2.1 实验任务 1	23
2.2.2 实验任务 2	26
2.3 项目实战	28
2.3.1 项目描述	28
2.3.2 项目分析	29
2.3.3 项目编写	29
2.4 实验习题	31
实验 3 面向对象之类的多态性	33
3.1 知识点回顾	33
3.2 实验练习	36
3.2.1 实验任务 1	36
3.2.2 实验任务 2	37
3.3 项目实战 1	40
3.3.1 项目描述	40
3.3.2 项目分析	40
3.3.3 项目编写	40
3.4 实验习题	43

实验 4 面向对象之类的接口	48
4.1 知识点回顾	48
4.2 实验练习	51
4.2.1 实验任务 1	51
4.2.2 实验任务 2	54
4.3 项目实战	57
4.3.1 项目描述	57
4.3.2 项目分析	57
4.3.3 项目编写	58
4.4 实验习题	61
实验 5 图形用户界面设计	64
5.1 知识点回顾	64
5.2 实验练习	76
5.2.1 实验任务 1	76
5.2.2 实验任务 2	77
5.3 项目实战	84
5.3.1 项目描述	84
5.3.2 项目分析	84
5.3.3 项目编写	84
5.4 实验习题	88
实验 6 多线程	90
6.1 知识点回顾	90
6.2 实验练习	96
6.2.1 实验任务 1	96
6.2.2 实验任务 2	99
6.3 项目实战	102
6.3.1 项目描述	102
6.3.2 项目分析	103
6.3.3 项目编写	103
6.4 实验习题	107
实验 7 综合项目—图书馆信息管理系统	109
7.1 项目概述	109
7.2 项目需求分析	109
7.2.1 数据需求	109
7.2.2 事务需求	109
7.3 项目数据库设计	110
7.3.1 读者信息表(tb_reader)	110

7.3.2	图书类别信息表(tb_bookType)	111
7.3.3	图书信息表(tb_bookInfo)	111
7.3.4	图书借阅信息表(tb_borrow)	111
7.3.5	图书订购表(tb_order)	112
7.3.6	管理者信息表(tb_operator)	112
7.4	项目总体设计	113
7.4.1	系统登陆界面和主界面设计	113
7.4.2	数据库操作类	114
7.4.3	实体类的设计	114
7.4.4	图书馆信息管理模块类的设计	114
7.5	主要代码分析	115
7.5.1	登录界面的实现	115
7.5.2	主类的实现	118
7.5.3	实体类的实现	123
7.5.4	数据库操作类的实现	130
7.5.5	读者信息管理模块的实现	134
7.5.6	图书类别管理模块的实现	143
7.5.7	图书信息管理模块的实现	145
7.5.8	新书管理模块的实现	148
7.5.9	借阅管理模块的实现	150
7.5.10	系统维护模块的实现	151

实验 1 初识 Java 程序设计

1.1 知识点回顾

1. Java 简介

Java 是一种可以撰写跨平台应用程序的面向对象的程序设计语言,是由美国 Sun Microsystems 公司于 1995 年 5 月开发的 Java 程序设计语言和 Java 平台的总称。Java 语言是从 C++ 程序语言发展而来的,但比 C++ 语言简单,它是当前网络编程中的首选语言。Java 语言的出现是源于对独立于平台的语言的需要,即这种语言编写的程序不会因为芯片的变化而发生无法运行或出现运行错误的情况。

Sun 公司对 Java 编程语言的解释是:Java 编程语言是个简单、面向对象、分布式、解释型、健壮、安全、与系统无关、可移植、高效、多线程和动态的语言。主要特性如下:

(1) Java 语言是简单易用的。一方面,Java 语言来源于 C++ 语言,其语法与 C 语言和 C++ 语言接近,使得大多数程序员很容易学习和使用 Java;另一方面,Java 语言摒弃了 C++ 语言中很少使用的、难以理解且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。

(2) Java 语言是一种面向对象的程序设计语言。Java 语言只支持单继承,但它支持多接口。

(3) Java 语言是分布式的。

(4) Java 语言是健壮的。Java 语言的强制类型机制、异常处理、内存垃圾的自动收集等是程序健壮性的重要保证;对指针的摒弃是 Java 语言的明智选择。

(5) Java 语言是安全的。

(6) Java 语言是与平台无关的,平台无关性是 Java 的一大特点。所谓平台无关是指编译后的 Java 程序可直接在不同的平台上运行而不用重新编译,这一特性使得 Java 随着 Web 应用的普及而迅速普及起来。因此只要在操作系统中配有 JVM,就可以运行编译后的 Java 程序,也就是“一次编写,随处运行”,因而轻松实现跨平台的特点。

(7) Java 语言是可移植的。

(8) Java 语言是解释型的。在运行时,Java 平台中的 Java 解释器对这些字节码进行解释执行,执行过程中需要的类在连接阶段被载入运行环境中。

(9) Java 语言是高性能的。

(10) Java 语言是多线程的。在 Java 语言中,线程是一种特殊的对象,它必须由 Thread 类或其子(孙)类来创建。Java 语言支持多个线程的同时执行,并提供多线程之间的同步机制

来保证对共享数据的正确操作。

(11) Java 语言是动态的。Java 语言需要的类能够动态地被载入运行环境,通过网络来载入。

2. Application 与 Applet

Application(应用程序)和 Applet(小应用程序)是 Java 提供的两种不同类型的程序,Java Application 是一种独立完整的程序,与其他应用程序类似,是可以在计算机操作系统中运行的程序;Java Applet 不是一种独立完整的程序,需要在 WWW 浏览器这种特定环境下运行。

Java Application 程序的结构特点是:程序是由一个或多个文件组成的,每个文件又是由一个或多个类组成的,每个类是由若干个方法和变量组成的。

Java 小应用程序是内嵌于 HTML 文档中,并使用<APPLET>标记的可执行 Java 字节码,是具有动态、安全、跨平台特性的网络应用程序,通过主页发布到 Internet。Java Applet 可以在 Internet 中传输,通过因特网下载并且能在所有支持 Java 的浏览器中运行,它的最大特点是能对用户作出反应,并进行相应的变化。Applet 不能独立在 JVM 中运行,而是由浏览器或 Applet 阅读器(appletviewer)执行。

3. Java 开发运行环境介绍

Java 开发工具分为两大类:一类是基本开发工具,即 Sun Microsystems 公司免费提供的 Java 2 SDK;另一类是专业开发工具,当今最流行的企业级开发工具有 Eclipse、JCreator、JBuilder2008、JDeveloper 等,这些工具功能强大,可以满足企业级开发的需要。

JDK(Java Development Kit)是 Sun 公司开发的 Java 开发工具包,它是一个简单的命令行工具,主要包括软件库,编译 Java 源代码的编译器,运行 Java 字节码的解释器,以及测试 Java Applet 的 Applet 阅读器,还有其他一些有用的工具。它主要是通过 DOS 命令行,在 DOS 环境下进行 Java 程序的编译和运行。用户可以到 Sun 公司的 Java 语言官方网站上下载最新版的 JDK 软件。安装完 JDK 后,需要设置 3 个系统环境变量,JAVA_HOME 的路径、JDK 开发工具的路径和 CLASSPATH 的路径。(具体配置见实验任务 1)

JCreator 是由 Xinox 软件公司所开发的一个可视化的 Java 程序集成开发环境,它给用户提供了包括工程管理、工程模板、代码实现、代码调试器、高亮语句编辑以及完全客户化的用户界面等广泛的功能。

JBuilder 是由 Borland 公司开发的一款功能强大的可视化 Java 集成开发环境,可以快速开发包括复杂企业级应用系统的各种 Java 程序,包括独立运行程序、Applet 程序、Servlet、JSP、EJB、Web Service 等。

Eclipse 是由 IBM 公司开发的一款开放源码的通用工具平台,它专注于为高度集成的工具开发提供一个全功能的、具有商业品质的工业平台,由 Eclipse 项目、Eclipse 工具项目、Eclipse 技术项目和 Eclipse Web 工具平台项目组成。最常用的 Java 开发功能实际上是由 Eclipse 一个主要的插件 JDT(Java Development Tools,Java 开发工具)所提供的,它随 Eclipse SDK 一同发行。Eclipse 是开放源代码的项目,用户可以去<http://www.eclipse.org/downloads/>免费下载 Eclipse 的安装程序。图 1.1 所示的界面布局是在 Java 视图下的工作主界面。工作台是一个桌面开发环境,提供了使用 Eclipse 工具必需的用户界面,以 Java 视图为例,用户的工作界面包含窗口、菜单栏、工具栏等。在资源管理窗口中文件以“文件/目录”的形式进行管理,可以在该窗口中查找所需的文件资源并可以右键单击文件名进行相关操作。

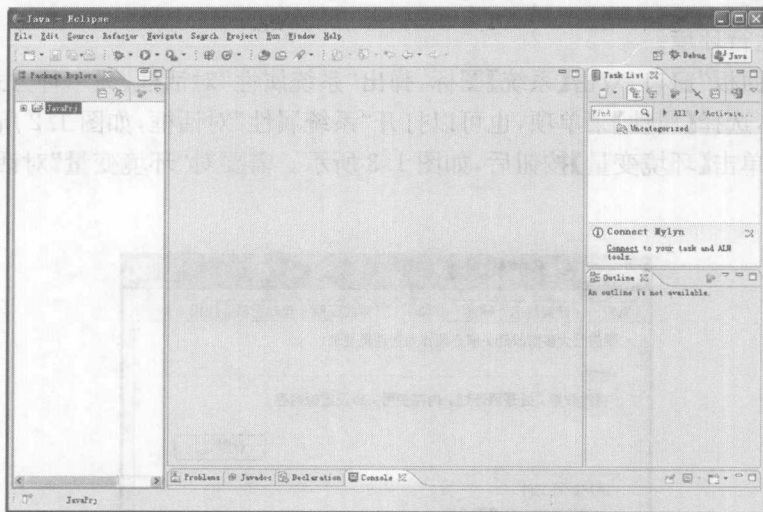


图 1.1 Eclipse 的工作界面

4. Java 核心包

(1) java.lang 包。封装所有编程应用的基本类,如 Object、Class、String、System、Integer、Thread 等。其中 Object 是所有类的根,它所包含的属性和方法被所有类继承。

(2) java.awt 包。封装抽象窗口工具包,提供构建和管理图形用户接口(GUI)设计工具,包含构件、容器和布局管理器。

(3) java.io 包。提供输入/输出文件操作的类。

(4) java.applet 包。为 Applet 提供执行所需的所有类。

(5) java.net 包。提供程序执行网络通信应用及 URL 处理的类。

(6) java.util 包。提供实用程序类和集合类,如系统特性定义和使用、日期方法类、集合 Collection、List、Arrays、Map 等常用工具类。

(7) java.sql 包。提供访问和处理标准数据源数据的类。

(8) java.rmi 包。提供程序远程方法调用所需的类。

(9) javax.naming 包。提供命名服务所需的类和接口。

(10) javax.swing 包。提供构建和管理程序的图形界面的轻量级的构件。

(11) javax.transaction 包。提供事物处理所需的基本类。除此之外,Java 扩展包还提供 Rmi、Sound、Accessibility 等包。

1.2 实验练习

1.2.1 实验任务 1

● 实验任务

掌握开发 Java 应用程序的三个步骤:编写源文件、编译源文件和运行应用程序。

● 实验要点

首先设置运行环境。

打开“控制面板”窗口，双击【系统】图标，弹出“系统属性”对话框；另一种方式是在【我的电脑】图标上右击，选择【属性】菜单项，也可以打开“系统属性”对话框，如图 1.2 所示。然后选择【高级】选项卡，单击【环境变量】按钮后，如图 1.3 所示。需要对“环境变量”对话框中的选项进行设置。

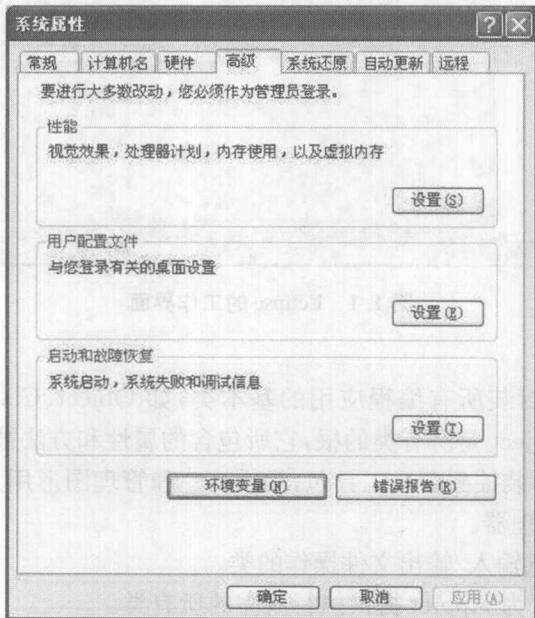


图 1.2 “系统属性”对话框

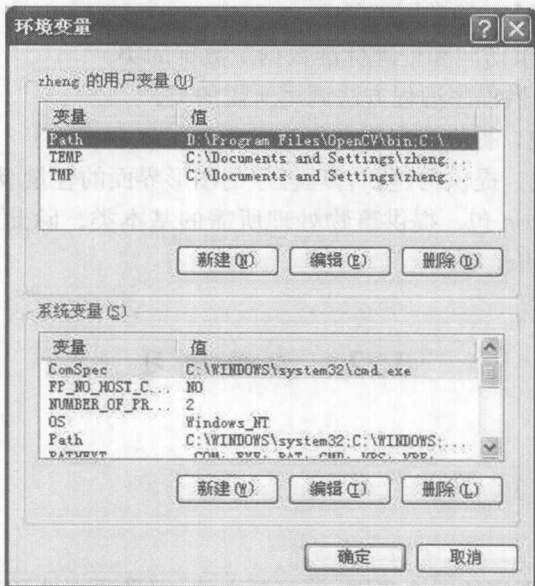


图 1.3 “环境变量”对话框

1. 设置 JAVA_HOME 环境变量

“系统变量”对话框中通常没有 JAVA_HOME 变量,可以单击【新建】按钮,打开如图 1.4 所示的对话框。

在“变量名”文本框中输入 JAVA_HOME,“变量值”文本框中输入安装路径,例如:C:\Program Files\Java\jdk1.6。



图 1.4 新建 JAVA_HOME 环境变量

2. 设置 Path 环境变量

“系统变量”对话框通常已有 Path 变量(没有则新建),单击【编辑】按钮,打开如图 1.5 所示的对话框。

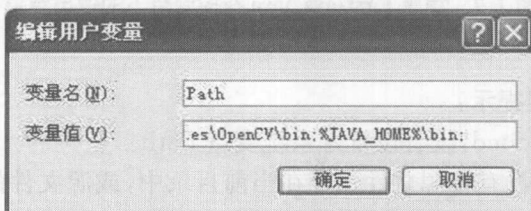


图 1.5 编辑 Path 环境变量

在“变量值”文本框中输入 Java 开发工具的所在路径,即 bin 文件夹的所在路径。可以填写绝对路径,例如 C:\Program Files\Java\jdk1.6\bin;或者输入相对路径路径,由于前面已经定义 JAVA_HOME 变量,所以相对路径是%JAVA_HOME%\bin。

注意:“变量值”文本框中还有一些其他路径,每个路径之间用“;”隔开。

3. 设置 CLASSPATH 环境变量

“系统变量”对话框中通常已有 CLASSPATH 变量(没有则新建),单击【编辑】按钮,打开如图 1.6 所示的对话框。

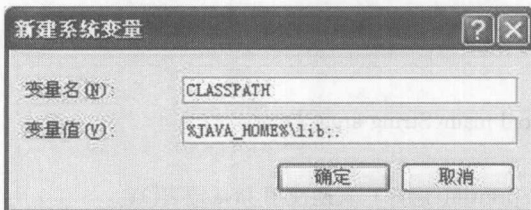


图 1.6 编辑 CLASSPATH 环境变量

Java 源程序一般用“.java”作为扩展名,是一个文本文件,用 Java 语言写成,可以用任何文本编辑器来编辑。可以通过【程序】—>【附件】—>【记事本】来打开文本编辑器。程序编写完

成后,保存文件。注意:文件名保存为 FirstJava.java。

使用 DOS 命令进入 FirstJava.java 文件所存放的目录(如 E:\java)下,使用“javac”命令编译该应用程序,通过编译该文件夹下会生成一个名为 FirstJava.class 的 class 文件,该文件包含程序的字节码,Java 字节码中包含的是 Java 解释程序将要执行的指令码,使用“java”命令执行该 class 文件,可在控制台下看到执行结果,如图 1.7 所示。

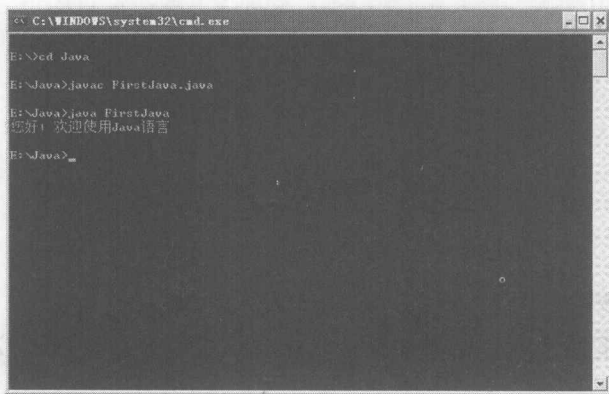


图 1.7 程序 FirstJava.java 在控制台下的输出结果

● 实验分析

可能会遇到下列错误提示:

(1) Command not Found:没有设置好系统变量 Path。

(2) File not Found:没有将源文件保存在当前目录中,或源文件的名字不符合有关规定。

要特别注意:Java 语言的标识符号是区分大小写的。

(3) 出现一些语法错误提示。例如,在汉语输入状态下输入了程序中需要的语句符号等。Java 源程序中语句所涉及的小括号及标点符号都是英文状态下输入的,比如“您好! 欢迎学习 Java 语言”中的引号必须是英文状态下的引号,而字符串里面的符号不受汉语或英语的限制。

(4) Exception in thread “main” java. lang. NoClassFoundError:没有设置好系统变量 ClassPath,或运行的不是主类的名字或程序没有主类。

● 实验主要代码

```
/*
  此类用于在屏幕上显示消息。
 */
public class FirstJava
{
    public static void main(String args[])
    {
        System.out.println("您好! 欢迎使用 Java 语言");
    }
}
```

该程序的功能是在显示器屏幕输出显示字符串:您好! 欢迎使用 Java 语言。

(1) 在 Java 中源程序的文件名并不是任意的,它必须和程序中定义类名相同,扩展名必须

是 java。另外,由于 Java 是区分大小写的,所以也应确保文件名的大小写字母和类名一致。

(2) 所有的 Java 应用程序都通过调用 main() 方法开始执行,因此在 Application 的声明中对 main() 方法的定义是必不可少的。修饰符 static 说明该方法是静态的,它可以在创建对象之前被调用。

(3) public class FirstJava 定义了一个类,是 Java 程序的基本组成部分。在 Java 中,所有功能都是以类的方式实现的。每一个类都由关键字 class 和类名组成。其中类名是自己定义的,且含有 main() 方法的类的类名和源程序文件的文件名必须一致。

(4) 可以添加注释。注释用来解释程序,使程序更好地被理解,它不会被执行。可以用//加单行的注释,用/*...*/加多行的注释。

(5) 使用的变量名、对象名、方法名等标识符要有意义,尽量做到“见名知义”。

1.2.2 实验任务 2

● 实验任务

使用 Eclipse 编写 Java 应用程序 FirstJava.java, 根据命令行参数输出相应信息(如“您好! 欢迎使用 Java 语言”)。

● 实验要点

(1) 打开 Eclipse。双击 eclipse.exe, 启动 Eclipse 平台。

(2) 配置 Eclipse 的 Workspace。初次打开 Eclipse, 会要求用户配置 Eclipse 的 Workspace, 点击【Browse】, 选择合适的工作区保存位置即可。

(3) 创建 Java 项目名 Java。执行【File】—>【New】—>【Java Project】菜单命令, 打开“New Java Project”对话框。在“Project name”文本框中输入项目名称 Java, 然后单击【Finish】按钮, 如图 1.8 所示。

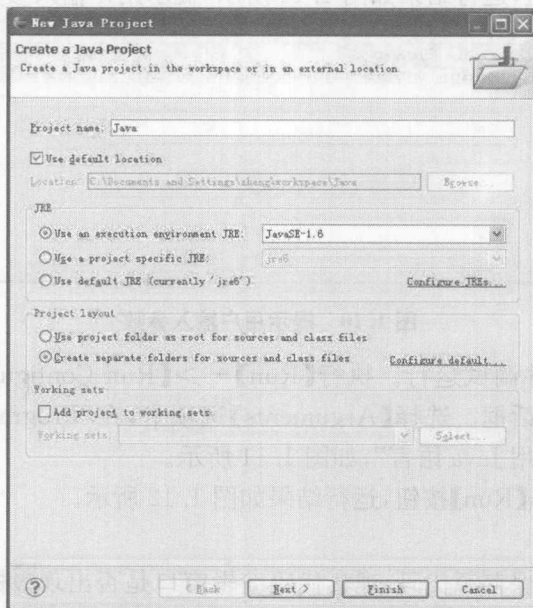


图 1.8 “New Java Project”对话框