

高等院校规划教材

Java 实践教学

主编 陈 超



煤炭工业出版社

高等院校规划教材

Java 实践教学

主 编 陈 超

副主编 李芙玲 白 磊

煤炭工业出版社

· 北 京 ·

图书在版编目 (CIP) 数据

Java 实践教程 / 陈超主编. -- 北京: 煤炭工业出版社,
2015

高等院校规划教材

ISBN 978 - 7 - 5020 - 4830 - 3

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

中国版本图书馆 CIP 数据核字 (2015) 第 060757 号

Java 实践教程 (高等院校规划教材)

主 编 陈 超
责任编辑 李振祥
编 辑 田小琴
责任校对 姜惠萍
封面设计 于春颖

出版发行 煤炭工业出版社 (北京市朝阳区芍药居 35 号 100029)
电 话 010 - 84657898 (总编室)
010 - 64018321 (发行部) 010 - 84657880 (读者服务部)
电子信箱 cciph612@126.com
网 址 www.cciph.com.cn
印 刷 煤炭工业出版社印刷厂
经 销 全国新华书店

开 本 787mm × 1092mm¹/₁₆ 印张 12¹/₄ 字数 281 千字
版 次 2015 年 5 月第 1 版 2015 年 5 月第 1 次印刷
社内编号 7685 定价 27.00 元

版权所有 违者必究

本书如有缺页、倒页、脱页等质量问题,本社负责调换,电话:010 - 84657880

内 容 提 要

本书针对 Java 程序设计的初级及高级操作，精心设计了各章的实验，主要包括：Java 编程环境搭建；数据类型、运算符与表达式；数组、控制语句；类与面向对象编程；类间的关系；Java 基本类库；Java 字符处理；Java 中的线程；输入输出流与文件；基于 Java Swing 的图形用户界面设计；基于 Socket 的网络编程及数据库与 JDBC 编程等。

本书可作为普通本科、大专、高职高专院校计算机及相关专业学生的 Java 语言上机实训指导教材，也可作为高等院校非计算机专业学生自学、各类社会人员培训用书。

前 言

Java 程序设计语言是随着互联网的发展而被推广应用的，它是目前主流的程序设计语言之一。上机实验学习是计算机程序设计语言的重要环节，是对学生进行基本程序设计技能训练，增强学生实践能力的关键环节，是培养学生创新精神和计算思维的重要手段。书中的实验题目和习题紧扣教材各章内容的要点，帮助学生加深理解基本概念和掌握相关知识，同时采用“任务驱动”的描述方式引导学生完成实验，实现全面提高学生实践能力和应用能力的目的。由于编者水平的局限性，所给出的实验程序从结构到代码并不一定是最优的，希望能起到抛砖引玉的作用。

本书本着“通俗易懂，学践并重”的原则进行编写，针对学生的特点，由浅入深、循序渐进地提出一些实践任务，重点培养学生的软件开发能力以及利用 Java 语言解决实际问题的能力，为学生后续的“软件工程”等课程的学习打下坚实的基础。

本书第一章、第二章、第三章、第十二章由陈超编写，第四章、第六章、第七章由白磊编写，第五章由郭慧编写，第八章、第十四章由李芙玲编写，第九章、第十章由杜丽娟编写，第十一章由郭晓欣编写，第十三章由王长利编写。全书由陈超、李芙玲和白磊负责统稿并修改定稿。本书在编写过程中参考了大量的书籍、资料，在此对所有作者表示衷心感谢！

由于时间仓促、水平有限，书中不足之处在所难免，恳请读者批评指正。

编 者

2015 年 1 月

目次

第一章 Java 开发环境配置	1
实验一 JDK 的下载、安装与配置	1
实验二 UltraEdit 的配置与使用	4
实验三 Eclipse 的使用	8
习题	10
第二章 基本数据类型和基本操作	12
实验一 基本数据类型	12
实验二 基本操作	15
习题	19
第三章 控制语句	21
实验一 顺序结构及选择结构	21
实验二 循环结构	23
习题	28
第四章 类和对象	32
实验一 类和对象的创建和使用	32
实验二 类的静态成员与实例成员	33
实验三 方法调用和方法重载	35
实验四 访问权限和包	39
习题	44
第五章 继承与多态	53
实验一 继承	53
实验二 多态	54
实验三 抽象类和接口	57
习题	61
第六章 Java 异常处理	62
实验一 Java 异常处理机制	62
实验二 自定义异常类	68

习题	69
第七章 Java 基本类库	72
实验一 字符串类	72
实验二 日期类	73
实验三 Math 类	74
习题	76
第八章 图形用户界面设计	81
实验一 布局管理和按钮创建	81
实验二 事件处理与文本控件	82
实验三 菜单设计	86
实验四 单选按钮、复选框、组合框和列表框	88
习题	91
第九章 输入/输出流	94
实验一 文件	94
实验二 文件字节流	96
实验三 文件字符流	97
实验四 缓冲字符流	98
实验五 数据流与对象流	100
习题	103
第十章 数组、泛型和集合	115
实验一 一维数组	115
实验二 二维数组	116
实验三 Arrays 类	117
实验四 泛型	118
实验五 集合	119
习题	122
第十一章 多线程	133
实验一 创建多线程	133
实验二 线程同步	134
实验三 线程协同	136
习题	139
第十二章 数据库与 JDBC 编程	141
实验一 ODBC - JDBC 连接数据库	141

实验二 JDBC 连接数据库	149
习题	157
第十三章 网络通信程序设计	158
实验 单客户聊天程序	158
习题	160
第十四章 综合实验	162
实验 学生管理系统	162
参考文献	185

第一章 Java 开发环境配置

实验一 JDK 的下载、安装与配置

一、实验目的

- (1) 熟悉 JDK 的下载及安装过程。
- (2) 掌握 JAVA_HOME、CLASSPATH 及 Path 等环境变量的配置。
- (3) 掌握验证 JDK 的方法。

二、实验指导

下载 JDK 6 并安装在 C:\java\jdk1.6 文件夹中，完成 Java 环境变量的配置，在命令行模式下验证 JDK 是否安装配置正确，其实验步骤如下。

1. 登录 Oracle 官网并下载 JDK 6

打开浏览器，前往 <http://www.oracle.com/technetwork/java/javase/downloads/java-archive-downloads-javase6-419409.html> 根据本机操作系统的类型下载 JDK6（图 1-1），下载时要登录 Oracle 网站（需事先注册 Oracle 网站的免费账号）。

Java SE Development Kit 6u45		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
☒ Accept License Agreement ☐ Decline License Agreement		
Product / File Description	File Size	Download
Linux x86	65.46 MB	jdk-6u45-linux-i586-rpm.bin
Linux x86	68.47 MB	jdk-6u45-linux-i586.bin
Linux x64	65.69 MB	jdk-6u45-linux-x64-rpm.bin
Linux x64	68.75 MB	jdk-6u45-linux-x64.bin
Solaris x86	68.38 MB	jdk-6u45-solaris-i586.sh
Solaris x86 (SVR4 package)	120 MB	jdk-6u45-solaris-i586.tar.Z
Solaris x64	8.5 MB	jdk-6u45-solaris-x64.sh
Solaris x64 (SVR4 package)	12.23 MB	jdk-6u45-solaris-x64.tar.Z
Solaris SPARC	73.41 MB	jdk-6u45-solaris-sparc.sh
Solaris SPARC (SVR4 package)	124.74 MB	jdk-6u45-solaris-sparc.tar.Z
Solaris SPARC 64-bit	12.19 MB	jdk-6u45-solaris-sparcv9.sh
Solaris SPARC 64-bit (SVR4 package)	15.49 MB	jdk-6u45-solaris-sparcv9.tar.Z
Windows x86	69.85 MB	jdk-6u45-windows-i586.exe
Windows x64	59.96 MB	jdk-6u45-windows-x64.exe
Linux Intel Itanium	53.89 MB	jdk-6u45-linux-ia64-rpm.bin
Linux Intel Itanium	56 MB	jdk-6u45-linux-ia64.bin
Windows Intel Itanium	51.72 MB	jdk-6u45-windows-ia64.exe

图 1-1 下载 JDK 6

2. 安装 JDK 6

执行下载好的“jdk-6u45-windows-i586.exe”文件，将安装路径自定义到“C:\java\jdk1.6”，如图 1-2 所示。

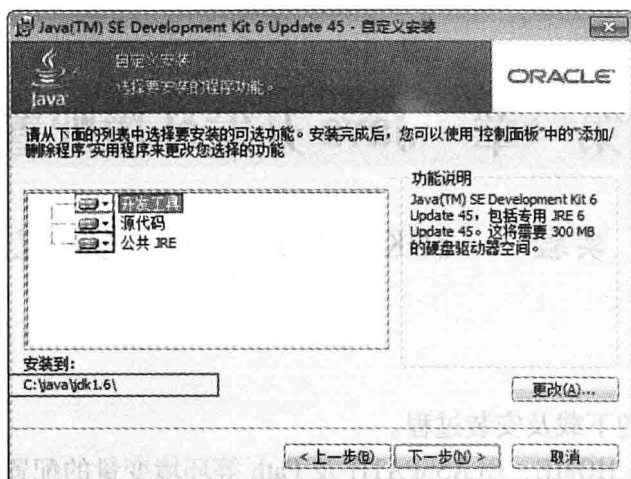


图 1-2 安装 JDK 6

3. 配置 Java 环境变量

(1) 进入 Windows 控制面板的【系统】配置项（可通过在资源管理器地址栏中键入【控制面板\所有控制面板项\系统】打开），通过点击【高级系统设置】弹出“系统属性”对话框并切换到【高级】选项卡，如图 1-3 所示。

(2) 在【高级】选项卡中点击【环境变量】按钮打开“环境变量”对话框，如图 1-4 所示。

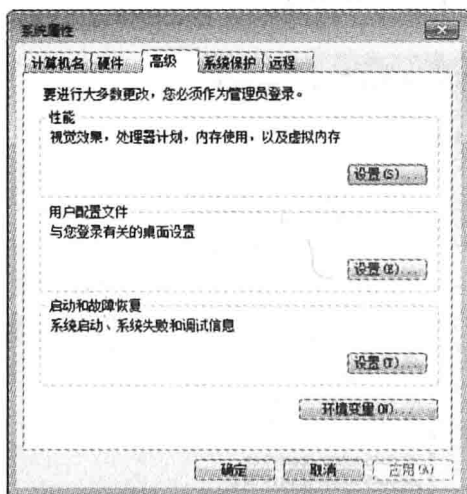


图 1-3 系统属性对话框



图 1-4 环境变量对话框

(3) 在“环境变量”对话框的“系统变量”选项区域中单击【新建】按钮，在“变量名”处输入“JAVA_HOME”，在“变量值”处输入“C:\java\jdk1.6”，点击【确定】后建立 JAVA_HOME 系统变量（图 1-5）；使用同样的方法在“变量名”处输入“CLASSPATH”，在“变量值”处输入“.;% JAVA_HOME% \lib\dt.jar;% JAVA_HOME% \lib\tools.jar”，建立 CLASSPATH 系统变量（图 1-6）。

(4) 在“环境变量”对话框的“系统变量”选项区域中选择变量“Path”，单击

【编辑】按钮，在“变量值”开始处加入“% JAVA_HOME% \bin;”，如图 1-7 所示。

(5) 连续点击【确定】，关闭“环境变量”对话框、“系统属性”对话框。

4. 验证 JDK 是否安装配置正确

(1) 在【开始】菜单中搜索“cmd”，执行搜索到的“cmd.exe”打开命令行窗口，如图 1-8 所示。

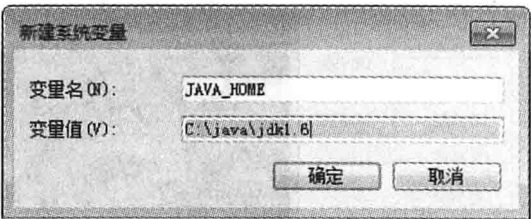


图 1-5 新建 JAVA_HOME 系统变量

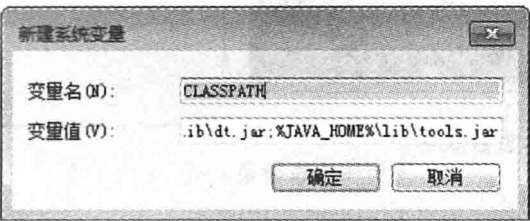


图 1-6 新建 CLASSPATH 系统变量

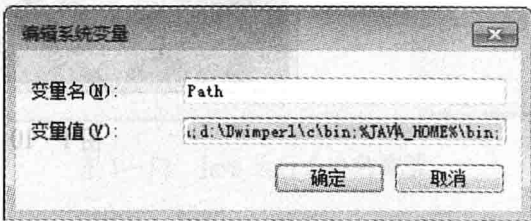


图 1-7 编辑 Path 环境变量

(2) 在命令行窗口中输入“javac”，如果显示“javac 不是内部或外部命令，也不是可运行的程序或批处理文件”，说明 Java 环境变量配置有误，请重新配置；如果配置成功会显示 javac 命令相关的参数说明，如图 1-9 所示。



图 1-8 搜索 cmd.exe



图 1-9 Java 环境变量配置正确时 javac 的显示

也可以在命令行窗口中输入“java -version”，如果配置成功会显示 Java 当前的版本，如图 1-10 所示。

注意：

- (1) 配置中的路径符号“;”“%”必须是英文符号。
- (2) 实际配置路径要与安装路径一致。



图 1-10 Java 当前版本

实验二 UltraEdit 的配置与使用

一、实验目的

- (1) 掌握 UltraEdit 的配置方法。
- (2) 掌握在 UltraEdit 中编写、编译及运行 Java 程序的方法。

二、实验指导

(一) UltraEdit 的配置

在 UltraEdit 中配置 Java 编译命令、Java 运行命令及 Java Applet 运行命令，其实验步骤如下。

1. Java 编译命令的配置

在 UltraEdit 中选择菜单【高级 | 工具配置】，打开“工具配置”对话框；在“命令行”处输入“javac %n%e”，在“工作目录”处输入“%p”，在“菜单项名称”处输入“编译 Java 程序”，在“命令行输出 (DOS 命令)”栏中选择【输出到列表窗口】和【捕捉输出】，如图 1-11 所示；点击【高级】按钮，为该命令选择一个图标；设置完成后依次点击【插入】、【确定】按钮。

2. Java 运行命令的配置

在 UltraEdit 中选择菜单【高级 | 工具配置】，打开“工具配置”对话框；在“命令行”处输入“java %n %modify%”，在“工作目录”处输入“%p”，在“菜单项名称”处输入“运行 Java 程序”，在“命令行输出 (DOS 命令)”栏中选择【输出到列表窗口】、【显示 DOS 窗口】和【捕捉输出】，如图 1-12 所示；点击【高级】按钮，为该命令选择一个图标；设置完成后依次点击【插入】、【确定】按钮。

3. Java Applet 运行命令的配置

在 UltraEdit 中选择菜单项【高级 | 工具配置】，打开“工具配置”对话框；在“命

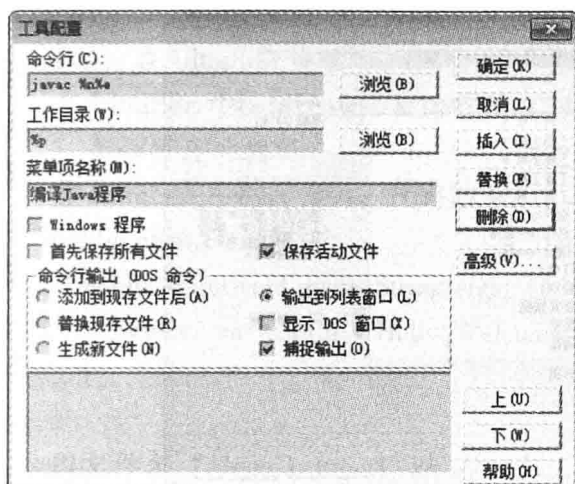


图 1-11 Java 编译命令的配置

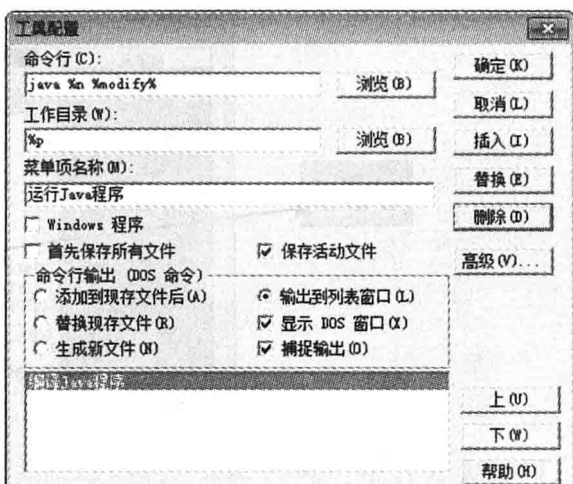


图 1-12 Java 运行命令的配置

令行”处输入“appletviewer %n%e”，在“工作目录”处输入“%p”，在“菜单项名称”处输入“运行 Applet 程序”，在“命令行输出（DOS 命令）”栏中选择【输出到列表窗口】和【捕捉输出】，如图 1-13 所示；点击【高级】按钮，为该命令选择一个图标；设置完成后依次点击【插入】、【确定】按钮。

4. 创建 Java 工具栏

在 UltraEdit 工具栏上单击鼠标右键。选择【自定义】打开“自定义”对话框；点击【新工具栏】按钮并命名“Java 工具栏”；单击对话框左边的【命令】按钮，将“高级”菜单下的“编译 Java 程序”“运行 Java 程序”“运行 Applet 程序”拖放到“Java 工具栏”中，点击【确定】按钮，如图 1-14 所示。

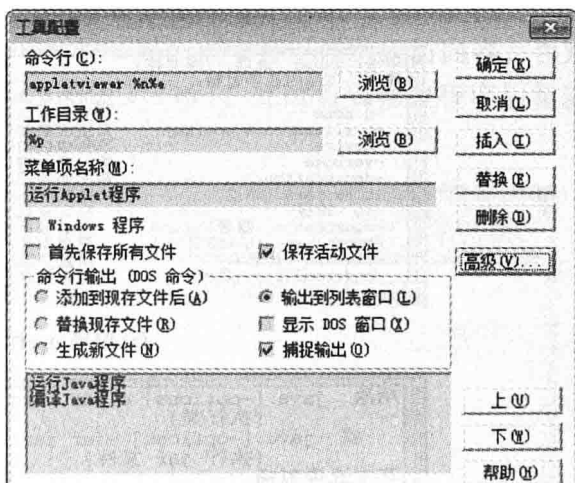


图 1-13 Java Applet 运行命令的配置

5. 验证 UltraEdit 是否配置正确

分别点击 Java 工具栏上的【编译 Java 程序】、【运行 Java 程序】、【运行 Applet 程序】按钮或菜单【高级】下的【编译 Java 程序】、【运行 Java 程序】、【运行 Applet 程序】菜单项，如果配置正确在 UltraEdit 输出框中会显示如图 1-15 所示的信息。

注意：

(1)%n 表示不带后缀的 Java 文件名，例如对于 Java 文件 MyDemo.java,%n 的值为 MyDemo。

(2)%e 表示文件的后缀名。

(3)%p 自动获取当前打开文件的目录，例如 UltraEdit 正在编辑的文件所在的目录为 d:\myjava，那么当前的工作目录%p 的值为 d:\myjava，如果对 Java 源程序进行编译，编译后产生的类文件将自动保存在 d:\myjava 目录下。

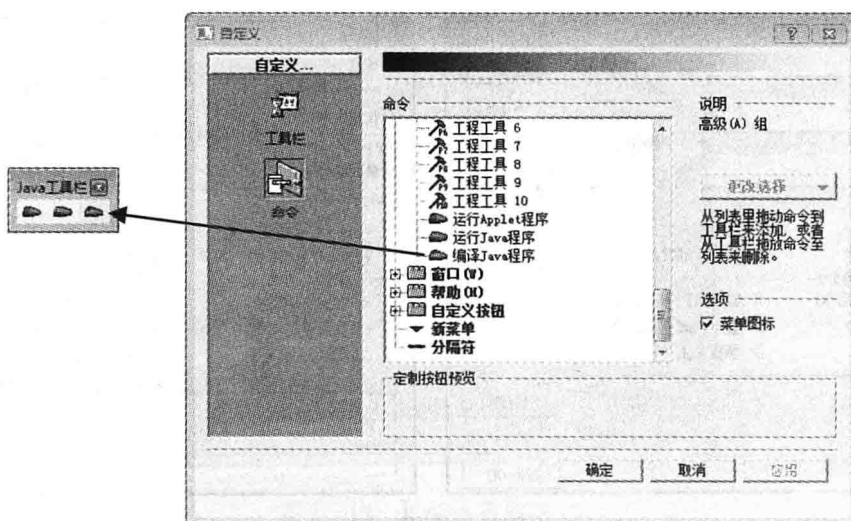
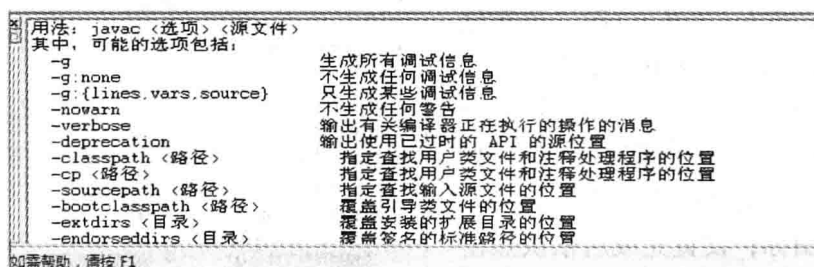
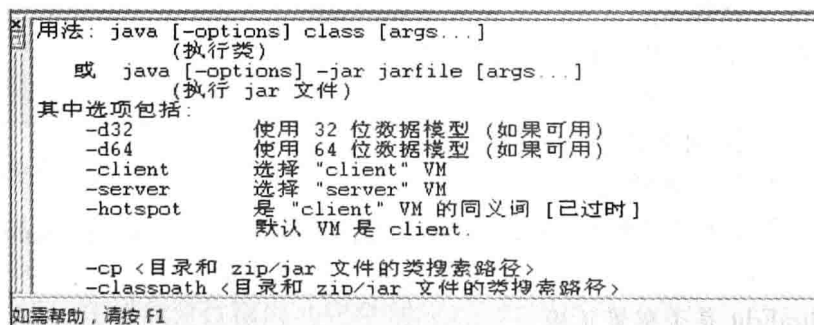


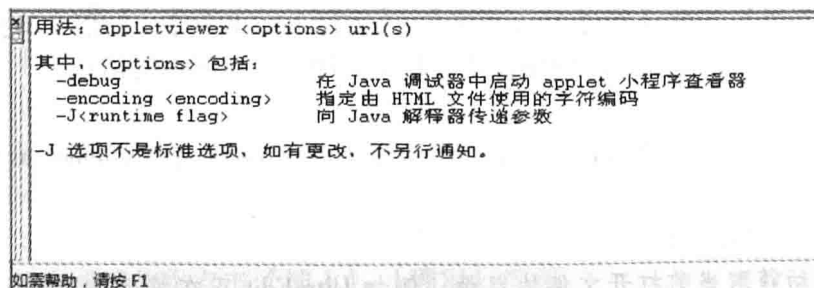
图 1-14 创建 Java 工具栏



(a) 编译Java程序提示信息



(b) 运行Java程序提示信息



(c) 运行Applet程序提示信息

图 1-15 程序提示信息

(4)%modify% 表示 Java 程序运行时从键盘输入的字符串参数。

(二) 在 UltraEdit 中编写 Java 程序

在 UltraEdit 中编辑、编译及运行 Java 程序，其实验步骤如下。

1. Java 程序的编辑

在 UltraEdit 中编写代码，代码如下：

```
public class HelloJava{  
    public static void main( String args[ ] ) {  
        System. out. println( " Hello, Welcome to Java world!" );  
    }  
}
```

以文件名“HelloJava. java”保存文件。值得注意的是：Java 代码严格区分大小写，若需显示行号，可以选择菜单【查看 | 显示行号】。

2. Java 程序的编译

点击 Java 工具栏上的【编译 Java 程序】按钮或菜单【高级】下的【编译 Java 程序】菜单项进行程序的编译。如果程序没有错误，UltraEdit 输出框中将不显示任何内容，否则输出框将提示程序发生错误的行号和错误的类型（图 1-16），此时需要修改程序后重新编译，直到无任何错误提示。

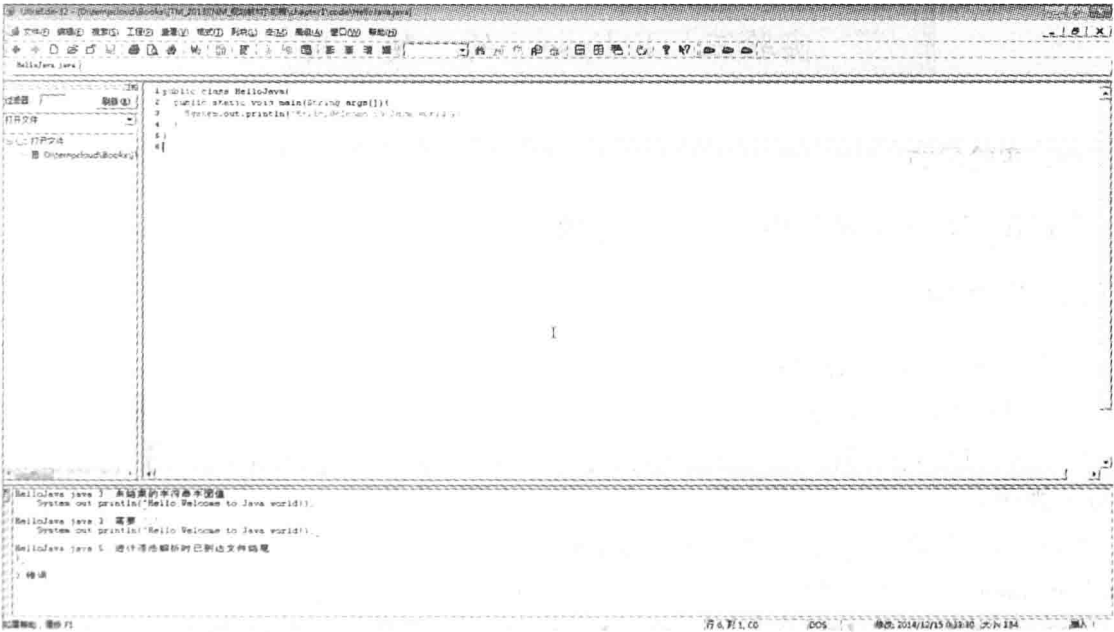


图 1-16 Java 程序的编译

3. Java 程序的运行

点击 Java 工具栏上的【运行 Java 程序】按钮或菜单【高级】下的【运行 Java 程序】菜单项将运行编译成功的 Java 程序，运行时将首先弹出“工具命令”对话框，如果所编写的 Java 程序需要提供命令行参数（图 1-17），应在“工具命令”编辑框的末尾处加上具体的参数，否则可直接点击【确定】按钮（如本例），运行结果将显示在 UltraEdit 的输

出框中，如图 1-18 所示。

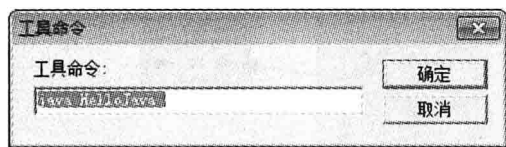


图 1-17 运行时提供命令行参数

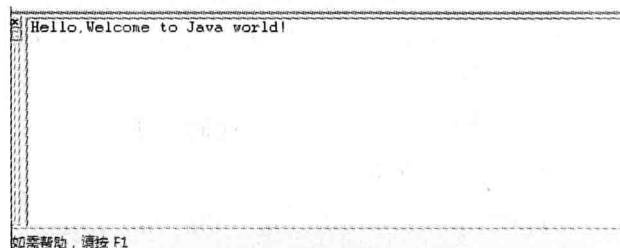


图 1-18 Java 程序的运行

注意：

- (1) 保存文件名必须与公共类类名相同，否则 Java 程序将无法正确编译及运行。
- (2) 当 Java 程序包含多个源文件时，可以将应用程序所在源文件（包含主类的源文件）和其他源文件放在同一目录中，在编译时可以只编译应用程序源文件，Java 会自动先编译应用程序所需要的其他源文件。
- (3) 如果需要编译某个目录下的全部 Java 程序，可以在命令行模式下进入该目录，然后执行“javac *.java”。

实验三 Eclipse 的 使 用

一、实验目的

掌握在 Eclipse 中编辑和运行 Java 程序的方法。

二、实验指导

在 Eclipse 中编辑及运行 Java 程序，其实验步骤如下。

1. 在 Eclipse 中编辑 Java 程序

- (1) 启动 Eclipse 后，先选择菜单项【Window | Open Perspective | Java】将视图切换到 Java 视图。
- (2) 选择菜单项【File | New | Java Project】，打开“New Java Project”对话框，在“Project Name”处输入“MyProject01”，点击【Finish】关闭对话框，如图 1-19 所示。
- (3) 选择菜单项【File | New | Class】，打开“New Java Class”对话框，在“Name”处输入类名称“HelloJava”，如果是应用程序所在类则勾选【public static void main (string [] args)】，点击【Finish】关闭对话框，如图 1-20 所示。
- (4) 在自动建出的“HelloJava.java”文件中加入下方的粗斜体代码并保存。

```
public class HelloJava {  
    /**  
     * @param args  
     */  
}
```

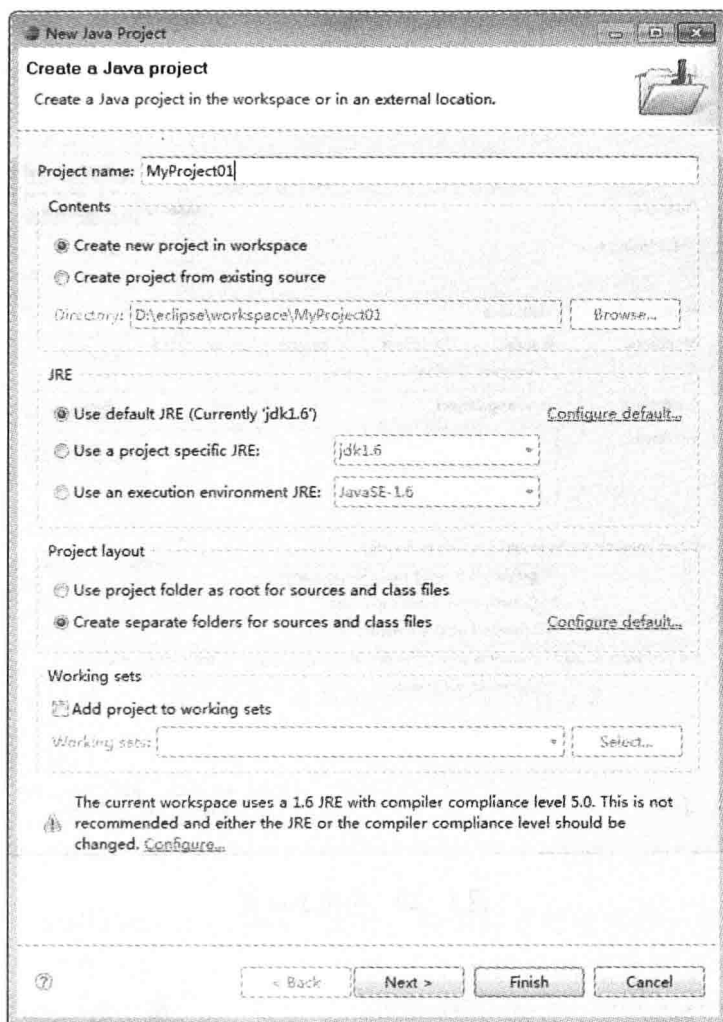


图 1-19 新建 Java 工程

```
public static void main(String[] args) {
    // TODO Auto-generated method stub
    System.out.println("Hello,Welcome to Java world!");
}
```

2. 在 Eclipse 中运行 Java 程序

(1) 选择菜单项【Run | Run】，运行结果将在“Console”窗格中显示，如图 1-21 所示。

(2) 当应用程序运行时，若需要提供命令行参数则选择菜单项【Run | Open Run Dialog...】打开“Run”对话框进行参数设置，在左边窗格中选择“Java Application”→“HelloJava”，在右边窗格中切换到【Arguments】选项卡，然后在“Program arguments”处输入命令行参数，如图 1-22 所示。