



全国高职高专计算机系列精品教材

Java实例应用 教程

JAVA SHILI YINGYONG JIAOCHENG

主 编/王建虹

 中国人民大学出版社

全国高职高专计算机系列精品教材

Java 实例应用教程

主 编 王建虹

副主编 蔡吸礼 张海玉 李 琳

参 编 蔺建霞 韩 慧 贾润亮

中国人民大学出版社

• 北京 •

图书在版编目 (CIP) 数据

Java 实例应用教程/王建虹主编
北京:中国人民大学出版社,2010
(全国高职高专计算机系列精品教材)
ISBN 978-7-300-12432-2

I. ①J…

II. ①王…

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 133444 号

全国高职高专计算机系列精品教材

Java 实例应用教程

主 编 王建虹

副主编 蔡吸礼 张海玉 李 琳

出版发行 中国人民大学出版社

社 址 北京中关村大街 31 号

邮政编码 100080

电 话 010-62511242 (总编室)

010-62511398 (质管部)

010-82501766 (邮购部)

010-62514148 (门市部)

010-62515195 (发行公司)

010-62515275 (盗版举报)

网 址 <http://www.crup.com.cn>

<http://www.ttrnet.com>(人大教研网)

经 销 新华书店

印 刷 北京东方圣雅印刷有限公司

规 格 185mm×260mm 16 开本

版 次 2010 年 8 月第 1 版

印 张 14

印 次 2010 年 8 月第 1 次印刷

字 数 329 000

定 价 26.00 元

版权所有 侵权必究 印装差错 负责调换

前 言

Java 是目前全球最有影响力的软件开发工具之一，是应用非常广泛的一种面向对象的程序设计语言。Java 的出现，给整个软件业带来了巨大的冲击，Java 将不可避免地影响一代程序员。

初学编程的人很苦恼的一件事是不知道如何上手。传统的思路是按部就班，从基本语法到各种应用程序逐步深入。通过近几年的基于工作工程的教学改革，我们将基本概念、基本语法用比较典型的实例贯穿起来，通过学习这些实例，按照完成这些任务的过程来掌握编程知识。因为学习编程需要大量的实践，而这种方法正好符合学生的学习特点，所以取得了较好的教学效果。按照基于工作过程，学中做、做中学的设计思路。从整体上将本书内容分为两个阶段（两部分）：

第一阶段，引入一个完整项目，直接带领学生做。我们用“学生管理系统”这样一个项目，简单直观地为学生展示一个完整的 Java 项目案例，引导学生快速入门，同时培养学生浓厚的学习兴趣。

第二阶段，深入学习 Java 编程。

在第一阶段的学习完成之后，学生对 Java 能做什么，怎么做已经有了大致的了解，这时学生自己就很盼望学习细节的内容；同时我们遵循学校教育系统化的规律特点，我们在第二部分安排系统化的内容，进一步深入学习。

但是内容组织上也打破了传统方法。设计思路是这样的：首先给出本章要完成的任务（一般是一个完整的小案例），然后按照工作过程分析任务，给出实现过程，最后总结所遇到的知识点，并且扩充完善系统知识；理论内容以够用为原则，不涉及太深奥的理论知识，但是有一定的系统性，学生入门以后有能力进一步学习。“千里之行，始于足下”。我们希望通过对本书的学习能使学生避免走很多弯路，以最快最短的时间获得最大的收益。

如果对这种教学方式不熟悉，或者数据库知识还没有学到的话，也可以先学习第二部分的内容，再学习第一部分的内容。

本书的最大特点是：

- 理论与实践一体的开发理念：以完成案例任务为目标，整个内容围绕任务的解决展开，以案例为载体，将知识点有机地融入其中，引导学生自主思考创新，全面开发学生的潜能。

- 合理的学习结构：符合读者循序渐进、由浅入深的学习习惯，内容起点低，操作上手快，学习效果好。

- 简洁流畅的语言：不讲深奥的原理，不涉及不常用的知识，只介绍程序设计最常用的内容。但技术和工具起点高，采用最新的工具 MyEclipse，高调进入 Java 行业。



● 紧扣内容的习题和实训：在讲解完每一个任务之后，每章附有课堂练习，使学生“学中做、做中学”；每章附有紧扣所讲内容的典型习题及实训，以帮助读者巩固所学知识。

本书由王建虹任主编，蔡吸礼、张海玉、李琳任副主编，蔺建霞、韩慧、贾润亮参编。其中项目 1、8、9 由王建虹编写，项目 2、3 由张海玉编写，项目 4、5 由蔡吸礼和贾润亮编写，项目 6 由韩慧编写，项目 7、10 李琳编写，项目 11、12 由蔡吸礼和蔺建霞编写。

在本书的编写过程中，参考了大量的相关技术资料，吸取了许多同仁的宝贵经验，在此深表谢意，同时还要对那些关心和支持本书编写工作的领导、老师和同学们表示感谢。

虽然我们尽心尽力，但错误和缺点难免会有，敬请读者批评指正。

编者

2010 年 6 月

目 录

第 1 篇 使用 Java 开发数据库应用

程序 1

项目 1 初识 Java 程序 3

任务 1 准备知识 3

1.1.1 为什么学习 Java 3

1.1.2 Java 是什么 3

1.1.3 Java 技术平台 4

1.1.4 Java 的开发工具 4

任务 2 Sun JDK 软件包的安装 5

任务 3 工作过程(代码及分析) 7

任务 4 Java 的包结构 9

1.4.1 包的含义 9

1.4.2 包的分类与调用 9

1.4.3 包与 Java 文件路径的关系 ... 11

• 课后练习题 1 • 11

项目 2 学习使用 MyEclipse 12

任务 1 MyEclipse 的下载、安装与

运行 12

2.1.1 MyEclipse 的下载 12

2.1.2 MyEclipse 的安装 12

2.1.3 MyEclipse 的运行 13

任务 2 MyEclipse 的界面布局 14

任务 3 MyEclipse 的基本操作 16

2.3.1 设置工作区 16

2.3.2 创建、打开、关闭 Java

项目 17

2.3.3 导入、导出 Java 项目 17

2.3.4 添加、修改和删除 JRE 18

2.3.5 设置 Java Build Path 信息 ... 18

2.3.6 设置编辑器字体、颜色和显示

行号 19

2.3.7 注释与取消注释 21

2.3.8 生成 getters 和 setters

方法 21

2.3.9 格式化源代码 21

2.3.10 断点和调试器 21

任务 4 安装 VE (Visual Editor) 插件 ... 22

2.4.1 下载 VE 插件 22

2.4.2 安装 VE 插件 22

任务 5 使用 MyEclipse 编写、编译并运行

Java 程序 23

2.5.1 创建 Java 项目 23

2.5.2 创建 Java 类, 编写类的

代码 23

2.5.3 编译运行 Java 程序 25

• 课后练习题 2 • 25

项目 3 学生管理系统的界面设计 26

任务 1 创建学生管理系统的主窗体 26

3.1.1 知识准备: GUI 设计、JFrame

组件 26

3.1.2 工作过程 27

任务 2 创建学生管理系统的主菜单 30

3.2.1 知识准备: 菜单组件、事件

处理 30

3.2.2 工作过程 31

任务 3 创建主窗体的工具栏 34

3.3.1 知识准备: JToolBar 组件、

JButton 组件 34

3.3.2 工作过程 35

任务 4 设计学生信息编辑窗体 37

3.4.1 知识准备: JLabel 组件、JLabel

的常用方法、JTextField 组件、

JComboBox 组件 37

3.4.2 工作过程 39

任务 5 登录窗体的实现 42



3.5.1 知识准备: JPasswordField 组件、JFrame 背景图	42
3.5.2 工作过程	43
任务6 窗体间的跳转	43
3.6.1 知识准备: 窗体间的跳转、跳转条件、消息框	43
3.6.2 工作过程	44
任务7 帮助菜单的实现	45
3.7.1 知识准备: 帮助文件、HTML Help Workshop 的用法介绍	45
3.7.2 工作过程	48
• 课后练习题3 •	49

项目4 学生管理系统的数据库

操作	50
任务1 建立数据库及表	50
4.1.1 知识准备: 数据库概念、SQL 语言、数据库表的设计	50
4.1.2 工作过程	52
任务2 利用 JDBC 访问数据库	53
4.2.1 知识准备: JDBC 简介、JDBC 基本编程模式、异常处理	53
4.2.2 工作过程	55
任务3 完善登录功能	58
4.3.1 知识准备: 登录验证、select 语句	58
4.3.2 工作过程	59
任务4 向数据库添加学生信息	61
4.4.1 知识准备: insert 语句、List 接口和 ArrayList 类	61
4.4.2 工作过程	61
任务5 从数据库中删除学生信息	66
4.5.1 知识准备: delete 语句、确认框	66
4.5.2 工作过程	67
任务6 修改数据库中的学生信息	68
4.6.1 知识准备: update 语句、用 Prepared Statement 代替 Statement	68

4.6.2 工作过程	69
• 课后练习题4 •	70

项目5 完善学生管理系统的数据库

展示	72
任务1 用表格组件 JTable 展示全部学生信息	72
5.1.1 知识准备: JTable 组件、建立 JTable 表格	72
5.1.2 工作过程	73
任务2 用树形结构组件 JTree 显示数据	75
5.2.1 知识准备: JTree 组件及示例	75
5.2.2 工作过程	77
任务3 打包发布学生管理系统	80
5.3.1 知识准备: JAR 文件、在 My Eclipse 中打包 JAR	80
5.3.2 工作过程	80
• 课后练习题5 •	82

第二篇 深入学习 Java 程序

设计

项目6 学习 Java 基本语法

任务1 输出员工信息(数据类型)	85
任务2 计算学生成绩(运算符与表达式)	88
任务3 制作电子万年历(条件语句) ...	90
6.3.1 if 条件语句	91
6.3.2 switch 语句	92
任务4 完善电子万年历(循环语句) ...	94
任务5 判定学生的优秀级别(break、continue 语句)	98
• 课后练习题6 •	100

项目7 使用数组与字符串

任务1 斐波那契数列的输出(一维数组)	102
任务2 矩阵的乘法(二维数组)	104

任务3 学生成绩排序(数组排序)	107
7.3.1 起泡排序	107
7.3.2 Arrays 类的 sort 方法	108
任务4 判断回文字符串(字符串处理 String 类)	109
任务5 字符串的追加(字符串处理 StringBuffer 类)	113
• 课后练习题 7 •	116
项目 8 面向对象的 Java 编程	118
任务1 面向对象概述	118
8.1.1 面向对象的思想(四大发明之活字印刷)	118
8.1.2 面向对象中的基本概念	119
8.1.3 面向对象和基于对象的区别	120
8.1.4 面向对象编程	120
任务2 设计“人”类小程序(创建、封装类)	121
8.2.1 类和方法	121
8.2.2 创建和使用对象	123
8.2.3 为类的对象赋初值(构造方法)	124
8.2.4 完善 Person 类——带 get 和 set 方法(封装)	125
8.2.5 自动统计人数(类成员)	127
任务3 设计亚洲人的类(继承)	129
8.3.1 创建子类	129
8.3.2 子类对父类方法的重写(方法的重写)	131
8.3.3 instanceof 对象运算符	135
任务4 用多态的方法设计“中国人”类	139
8.4.1 构造方法的重载	139
8.4.2 普通方法的重载和重写	141
• 课后练习题 8 •	143
项目 9 理解抽象类、接口和内 部类	145
任务1 吃饭实例(抽象类)	145

任务2 在主板的接口上安装声卡、网卡(接口)	149
任务3 报警门的设计(接口和抽象类的应用)	151
任务4 使用内部类	154
• 课后练习题 9 •	157
项目 10 集合的应用	160
任务1 学生信息存储(集合简介)	160
任务2 学生信息处理(ArrayList 应用)	163
10.2.1 在 ArrayList 中添加和删除对象	163
10.2.2 在 ArrayList 中查询和修改对象	164
10.2.3 ArrayList 的遍历(迭代)	166
10.2.4 ArrayList 的排序	167
任务3 顾客排队(LinkedList 应用)	169
任务4 商品价格处理(HashMap 应用)	171
• 课后练习题 10 •	174
项目 11 深入理解 AWT 和 Swing	176
任务1 创建输入用户信息界面的主窗体(JFrame)	176
任务2 对输入用户信息界面进行布局设计(布局管理器)	179
任务3 向输入用户信息界面添加组件	185
任务4 实现输入用户信息界面上的功能(事件处理)	189
• 课后练习题 11 •	193
项目 12 使用 I/O 流	195
任务1 实现记事本的文件打开和保存功能(文件流、缓冲流)	195
12.1.1 使用字节方式读写文件	195
12.1.2 使用字符方式读写文件	200
12.1.3 使用缓冲流高效率读写	

文件	201	任务 5 记录用户的登录信息到文件	
任务 2 使用 File 类来管理文件和		(对象流)	210
目录	203	• 课后练习题 12 •	212
任务 3 实现数据库中数据的导入、导出		参考文献	214
(数据输入、输出流)	206		
任务 4 随机读取文件中的数据 (随机存取			
文件类)	208		

第 1 篇 使用 Java 开发数据库应用程序



项目 1 初识 Java 程序

我们的任务是开发一个简单的 Java 应用程序：在屏幕上输出“你确实喜欢学习 Java 吗？”同时输出答案 Yes 或者 No。完成本任务后，学生：

- 会用 Java 术语对程序进行初步的分析和评价；
- 会运用 Java 编写命令行程序；
- 会使用 JDK 和 Eclipse 开发环境，编译、运行、调试、维护 Java 程序；
- 掌握简单调试与排错技术。

任务 1 准备知识

1.1.1 为什么学习 Java

近年来，Java 语言一直是世界上应用最广泛的编程语言之一，因此现在有越来越多的人正努力或将要努力进入 Java 领域。如今，Java 已经不再简单地是一门编程语言，它更像一个完整的体系，一个系统的开发平台甚至被延伸成一种开源精神。Java 公用规范（PAS）已被国际标准化组织（ISO）认定，Java 技术已被列为当今世界信息技术三大要点之一。

Java 已经渐渐地渗透到各领域，你可以写出 Java Servlet，将其挂在 Apache 或其他网页服务器上；也可以写出 Java Applet，在网页浏览器上执行；甚至可以用 Java 写出数据库的 Stored Procedure，然后安装到 Oracle 8i 上。

Java 的最大特点是跨平台，如果想发布一个程序到多个平台，又不想改写大部分的程序，那么 Java 是绝佳的选择。Java 2 现在已经可以在 Linux、UNIX 和 Windows 操作系统上执行了。

在国外，特别是在美国，不论是 Sun 公司，还是其他大型企业，对于认证课程都很重视。Java 程序员是美国 Sun 公司国际认证的程序员，是目前全球最受重视、最受欢迎的程序员资格认证之一，具备这一认证就可以获得极好的工作机会和丰厚待遇。

Java 的信息文件都可从网站上免费取得。如果你能上网而且有时间，可以直接通过网络学习对象导向、Java 语言以及 Java API。Sun 公司的网站上有很多非常优秀的在线教材。

1.1.2 Java 是什么

Java 是 Sun Microsystems 于 1995 年推出的高级编程语言。

Java 领域的 JavaSE、JavaEE 技术已发展成为同 C# 和 .NET 平分天下的应用软件开发

平台和技术。

当人们提到“Java”，它们通常指的是：

□ Java 程序语言

一个类似 C++ 或 Smalltalk 的编程语言。学习 Java 程序语言类似学习人类自然语言，也有一套规则和文法。

□ Java 虚拟机 (JVM)

用来执行 Java 程序。JVM 有许多平台的版本，如 Linux 和 Windows 版。有了 JVM，Java 程序就可以在不同的平台上执行，即所谓的“编写一次，到处运行”。

□ Java APIs (指的是函式库的程序设计接口)

是一组预先定义好的类别，可以在程序中直接使用。这些都是免费的，包含了档案存取、网络读写、图形接口等功能。

当人们提到“学习 Java”，通常指的是 Java 语言和 API。大家可能对 JVM 所知不多，只要对 JVM 有基本的认识，对 Java 的学习也是很有帮助的。

学习 Java 的理由中，有些是技术性的，比如垃圾处理和异常处理，有些则是非技术性的。在程序设计的领域，Java 几乎是必备的技能。

垃圾处理机制：

Java 系统不仅要为对象分配所用的内存资源，还需要跟踪资源的使用情况，定期检测出不再使用的内存，由系统自动回收并做再次分配，称为垃圾回收机制 (Garbage Collection)。因此，Java 程序中将不用考虑对象的释放问题，从而减轻程序员的负担，提高程序的安全性，避免因资源耗尽而导致系统瘫痪的隐患。

异常处理机制：

Java 虚拟机提供了可靠的异常处理。Java 强迫你在一遇到有可能出问题的地方就要准备好应对之道。Java 的方法可以抛出异常，通知呼叫者程序出现情况了，这是相当好的机制。

1.1.3 Java 技术平台

自从 Sun 公司推出 Java 以来，就力图使之无所不包，Java 发展到现在，按应用来分主要分为三大块：J2SE、J2ME 和 J2EE。

□ J2SE: Java Platform, Standard Edition

J2SE 是 Java2 的标准版，是 Java 技术的核心，提供基础 Java 开发工具、执行环境与应用程序接口 (API)，可以用于桌面应用程序的编程，是 Java 软件开发方向的基础。

□ J2ME: Java Platform, Micro Edition

J2ME 主要应用于嵌入式系统开发，如手机和 PDA 的编程。

□ J2EE: Java Platform, Enterprise Edition

J2EE 是 Java2 的企业版，主要用于分布式的网络程序的开发，如电子商务网站和 ERP 系统。

1.1.4 Java 的开发工具

任何编程语言都离不开相应的开发工具和程序库，Sun 公司在 1996 年发布了包括运行环境和开发工具在内的 JDK1.0，之后几年又陆续发布了新版本 JDK 1.1、J2SE 1.2、J2SE 1.3、J2SE 1.4、J2SE 5.0。

在 Java 1.0 或 Java 1.1 中, API 库称为 JDK (Java Development Kit), 但在 Java 1.2 版后改名为 Java2 SDK (Software Development Kit), 不过很多人还是习惯称之为 JDK。当我们使用某种计算机语言开发应用程序时, 除了会用到该语言所提供的 API 之外, 还会用到编辑、编译、运行、调试等工具, 而其整合叫做 SDK。除此之外, 还有许多集成开发工具, 归纳如下:

- Sun 公司的 JDK 软件包;
- 微软公司的 Visual J++;
- Borland 公司的 JBuilder;
- IBM 公司的 Visual Age for Java、WebSphere Studio;
- Oracle 公司的 JDeveloper;
- WebGain 公司的 Visual Café;
- TogetherSoft 公司的 Together;
- 开放源代码的 Eclipse、NetBeans 等。

任务 2 Sun JDK 软件包的安装

安装文件可以在 <http://java.sun.com/> 的 download 中下载。

双击相应图标进行安装, 安装过程中可以自定义安装目录等信息。例如, 选择安装目录为 D:\jdk1.5。系统默认的安装路径为 C:\Program Files\Java\jdk1.5.0_07, 默认的组件选择是全部安装, 单击【完成】按钮即完成安装。

为了能正确使用 JDK, 需要手工配置一些环境变量。

例如, Windows 98 环境配置:

在桌面【我的电脑】图标上单击右键, 选择“属性”, 出现“系统属性”对话框, 在【高级】选项卡中单击【环境变量】按钮, 弹出“环境变量”对话框, 如图 1.1 所示。

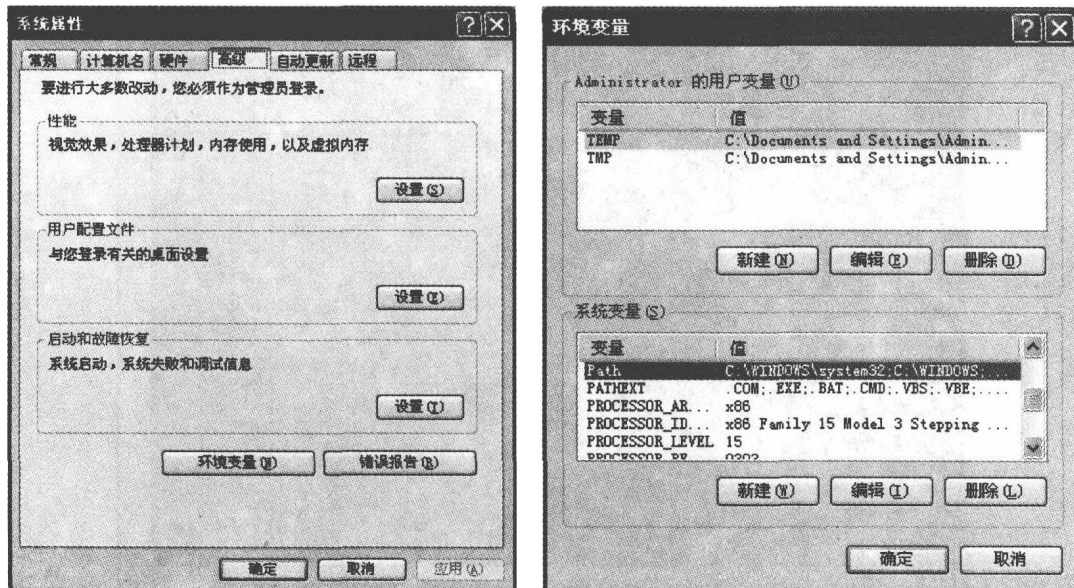


图 1.1 系统特性环境变量

在【系统变量】组框中找到 Path，单击【编辑】按钮，将 C:\Program Files\Java\jdk1.5.0_07\bin 加入到【变量值】文本框中，如图 1.2 所示，单击【确定】按钮结束编辑变量。

若没有找到 Path，则选择【新建】按钮，设置变量名为 Path，变量值为 C:\Program Files\Java\jdk1.5.0_07\bin。

用同样方法设置环境变量 classpath，其值为 .;C:\Program Files\Java\jdk1.5.0_07\lib。

设置完成后，在 DOS 窗口下测试配置是否成功。执行【开始】→【运行】，输入 cmd。如图 1.3 所示。单击【确定】按钮后，打开一个 DOS 窗口。

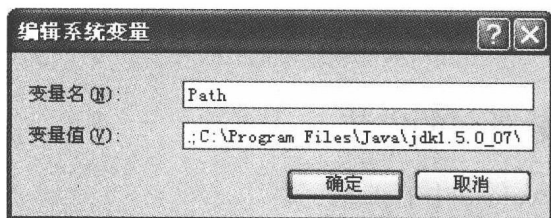


图 1.2 编辑系统变量

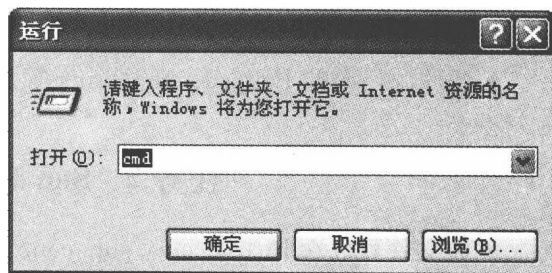


图 1.3 启动 DOS 窗口

在 DOS 窗口下输入 javac 或 java 并回车后，如果出现用法参数提示信息，则说明环境变量配置成功，显示界面如图 1.4 所示。



图 1.4 测试环境配置

任务 3 工作过程（代码及分析）

通常我们用两种方法完成任务：一种是使用 JDK 工具开发程序；另一种是使用 Eclipse 开发环境完成任务（详见项目 2）。

下面介绍使用 JDK 开发第一个 Java 程序。

步骤 1：使用记事本或 UltraEdit 等编辑器编写源代码。在记事本中输入下列代码，然后保存为 Message.java 文件。

为了便于说明程序结构，这里在每行前面加上了行号，具体程序如下所示：

```
1.  /**
2.     这里用的是文档注释
3.     此类用于在屏幕上显示消息
4.     @ version1.0,2009 年 1 月 6 日
5.     @author wjh
6.  */
7.  public class Message {
8.      /* 这里用的是多行注释
9.      这是一个 main 方法
10.     */
11.     public static void main(String [] args) {
12.         //输出此消息,这里用的是单行注释
13.         System.out.print ("你确实喜欢学习 Java 吗?<Yes/No>\n");
14.         System.out.println("我的答案是:" + args[0]);
15.     }
16. }
```

程序说明：

- 第 1~6 行是程序说明节，说明该程序的注释信息，这是一个文档注释的形式。
- 第 8~10 行是多行注释形式。
- 第 12 行是单行注释形式。
- 第 7 行说明所定义的类，约定类名称要首字母大写。
- 第 11 行是用于在命令行方式下运行的 main()方法。其中，public 说明 main()方法可以被外部引用；static 说明 main()方法可以直接从磁盘存储中调入内存执行，不需要经过对象的实例化过程；void 说明 main()方法没有返回值；main 说明方法名称；String[]args 说明 main()方法的输入参数列表，类型为 String。
- 第 13 和 14 行调用 Java 基础类 System.out 的 print()和 println()方法，把字符串输出。

- 要养成程序注释的好习惯。

知识点：

- (1) Java 应用程序框架。



```
public class Message {                                //声明类
    public static void main(String[] args) { //Java 入口程序框架
        ... 这里填写代码! ...
    }
}
```

Java 语言的源程序代码由一个或多个编译单元 (compilation unit) 组成, 每个编译单元只能包含下列内容 (空格和注释除外):

- 一个程序包语句 (package statement);
- 导入语句 (import statements);
- 类的声明 (class declarations);
- 接口声明 (interface declarations)。

每个 Java 语言的编译单元可包含多个类或接口, 但是每个编译单元最多只能有一个类或者接口是公共的。

程序的每行以分号结束, 字母区分大小写。

(2) main() 方法。

一个 Java 应用程序的源程序可以有 main() 方法, 也可以没有 main() 方法。如果该源程序所生成类文件想在命令行方式下直接运行, 则必须有 main() 方法。由于 main() 是一种特殊的成员方法, 是 Java 类程序在命令行方式下执行的入口方法, 所以它的定义是固定的, 形式必须和程序中第 11 行一样。

程序在保存时要特别注意文件名要与源代码中的类名 (用 public 修饰的) 完全相同, 而且有字母大小写的区分, 因此这个程序的文件名必须是 Message.java。我们将这个文件保存到 D:\Myjava 中。

步骤 2: 编译源代码。

Java 程序的编译在 MS-DOS 窗口中进行。打开 MS-DOS 窗口, 进入 D:\Myjava, 输入命令:

```
D:\Myjava > javac Hello.java
```

由编译命令 javac.exe 对当前文件夹中的 Hello.java 文件进行编译。如果编译正确, 将在文件夹 D:\Myjava 产生字节码文件 Message.class。如果程序中有编译错误, 系统将终止编译并显示出错信息, 按行指出错误, 改正错误后, 重复上面编译命令, 直至编译成功。

如果系统未找到 javac.exe 命令, 则说明 path 环境变量设置不正确。

步骤 3: 运行。

Java 程序的运行也在 MS-DOS 窗口中进行。输入如下命令:

```
D:\Myjava > java Message yes
```

由运行命令 java.exe 可运行文件 Message.class, 不用加上后缀名 .class, 如图 1.5 所示。

注意: 该程序需要向应用程序传递参数, 这里只输入 D:\Myjava>java Message 会报错。