



21st CENTURY
案例型规划教材

21世纪全国高职高专计算机案例型规划教材

Java 程序设计案例教程

主 编 徐翠霞
副主编 曲秀清
崔玲玲
刘 倩



北京大学出版社
PEKING UNIVERSITY PRESS

21 世纪全国高职高专计算机案例型规划教材

Java 程序设计案例教程

主 编 徐翠霞

副主编 曲秀清 崔玲玲 刘 倩

参 编 于振梅 刘 平



北京大学出版社
PEKING UNIVERSITY PRESS

内 容 简 介

本书以培养学生的 Java 语言应用能力为目标,具有很强的可读性和实用性,并配备了大量的案例,每个案例都经过精心的考虑,既能帮助读者理解知识,又具有启发性。本书通俗易懂,便于自学,针对较难理解的问题,设计的案例都是从简单到复杂,内容逐步深入,便于读者掌握 Java 编程的技巧。

本书共分 9 章,内容包括 Java 语言基础,类、对象和接口,基于 Swing 的图形用户界面设计,异常处理和多线程,输入输出流,Applet 与图形、图像和多媒体,Java 数据库编程,Java 网络编程基础和综合实训。

本书可作为高等学校 Java 程序设计课程的教材,也可作为 Java 语言自学者的参考书。

图书在版编目(CIP)数据

Java 程序设计案例教程/徐翠霞主编. —北京:北京大学出版社, 2008.7

(21 世纪全国高职高专计算机案例型规划教材)

ISBN 978-7-301-13570-9

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

中国版本图书馆 CIP 数据核字(2008)第 046332 号

书 名: Java 程序设计案例教程

著作责任者: 徐翠霞 主编

策 划 编 辑: 李彦红 王显超

责 任 编 辑: 李彦红

标 准 书 号: ISBN 978-7-301-13570-9/TP · 0946

出 版 者: 北京大学出版社

地 址: 北京市海淀区成府路 205 号 100871

网 址: <http://www.pup.cn> <http://www.pup6.com>

电 话: 邮购部 62752015 发行部 62750672 编辑部 62750667 出版部 62754962

电 子 邮 箱: pup_6@163.com

印 刷 者: 北京飞达印刷有限责任公司

发 行 者: 北京大学出版社

经 销 者: 新华书店

787mm×1092mm 16 开本 22.25 印张 511 千字

2008 年 7 月第 1 版 2008 年 7 月第 1 次印刷

定 价: 33.00 元

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

举报电话: 010-62752024

电子邮箱: fd@pup.pku.edu.cn

21 世纪全国高职高专计算机案例型规划教材 专家编写指导委员会

主 任 刘瑞挺 南开大学
副主任 安志远 北华航天工业学院
 丁桂芝 天津职业大学

委 员 (按拼音顺序排名)

陈 平 马鞍山师范高等专科学校
褚建立 邢台职业技术学院
付忠勇 北京政法职业技术学院
高爱国 淄博职业学院
黄金波 辽宁工程技术大学职业技术学院
李 纓 中华女子学院山东分院
李文华 湖北仙桃职业技术学院
李英兰 西北大学软件职业技术学院
田启明 温州职业技术学院
王成端 潍坊学院
王凤华 唐山工业职业技术学院
薛铁鹰 北京农业职业技术学院
张怀中 湖北职业技术学院
张秀玉 福建信息职业技术学院
赵俊生 甘肃省合作民族师范高等专科学校
顾 问 马 力 微软(中国)公司 Office 软件资深教师
 王立军 教育部教育管理信息中心

信息技术的案例型教材建设

(代丛书序)

刘瑞挺/文

北京大学出版社第六事业部在 2005 年组织编写了两套计算机教材，一套是《21 世纪全国高职高专计算机系列实用规划教材》，截至 2008 年 6 月已经出版了 80 多种；另一套是《21 世纪全国应用型本科计算机系列实用规划教材》，至今已出版了 50 多种。这些教材出版后，在全国高校引起热烈反响，可谓初战告捷。这使北京大学出版社的计算机教材市场规模迅速扩大，编辑队伍茁壮成长，经济效益明显增强，与各类高校师生的关系更加密切。

2007 年 10 月北京大学出版社第六事业部在北京召开了“21 世纪全国高职高专计算机案例型教材建设和教学研讨会”，2008 年 1 月又在北京召开了“21 世纪全国应用型本科计算机案例型教材建设和教学研讨会”。这两次会议为编写案例型教材做了深入的探讨和具体的部署，制定了详细的编写目的、丛书特色、内容要求和风格规范。在内容上强调面向应用、能力驱动、精选案例、严把质量；在风格上力求文字精练、脉络清晰、图表明快、版式新颖。这两次会议吹响了提高教材质量第二战役的进军号。

案例型教材真能提高教学的质量吗？

是的。著名法国哲学家、数学家勒内·笛卡儿(Rene Descartes, 1596—1650)说得好：“由一个例子的考察，我们可以抽出一条规律。(From the consideration of an example we can form a rule.)”事实上，他发明的直角坐标系，正是通过生活实例而得到的灵感。据说是在 1619 年夏天，笛卡儿因病住进医院。中午他躺在病床上，苦苦思索一个数学问题时，忽然看到天花板上有一只苍蝇飞来飞去。当时天花板是用木条做成正方形的格子。笛卡儿发现，要说出这只苍蝇在天花板上的位置，只需说出苍蝇在天花板上的第几行和第几列。当苍蝇落在第四行、第五列的那个正方形时，可以用(4, 5)来表示这个位置……由此他联想到可用类似的办法来描述一个点在平面上的位置。他高兴地跳下床，喊着“我找到了，找到了”，然而不小心把国际象棋撒了一地。当他的目光落到棋盘上时，又兴奋地一拍大腿：“对，对，就是这个图”。笛卡儿锲而不舍的毅力，苦思冥想的钻研，使他开创了解析几何的新纪元。千百年来，代数与几何，井水不犯河水。17 世纪后，数学突飞猛进的发展，在很大程度上归功于笛卡儿坐标系和解析几何学的创立。

这个故事，听起来与阿基米德在浴池洗澡而发现浮力原理，牛顿在苹果树下遇到苹果落到头上而发现万有引力定律，确有异曲同工之妙。这就证明，一个好的例子往往能激发灵感，由特殊到一般，联想出普遍的规律，即所谓的“一叶知秋”、“见微知著”的意思。

回顾计算机发明的历史，每一台机器、每一颗芯片、每一种操作系统、每一类编程语言、每一个算法、每一套软件、每一款外部设备，无不像闪光的珍珠串在一起。每个案例都闪烁着智慧的火花，是创新思想不竭的源泉。在计算机科学技术领域，这样的案例就像大海岸边的贝壳，俯拾皆是。

事实上，案例研究(Case Study)是现代科学广泛使用的一种方法。Case 包含的意义很广：包括 Example 例子，Instance 事例、示例，Actual State 实际状况，Circumstance 情况、事件、境遇，甚至 Project 项目、工程等。

我们知道在计算机的科学术语中,很多是直接来自日常生活的。例如 Computer 一词早在 1646 年就出现于古代英文字典中,但当时它的意义不是“计算机”而是“计算工人”,即专门从事简单计算的工人。同理,Printer 当时也是“印刷工人”而不是“打印机”。正是由于这些“计算工人”和“印刷工人”常出现计算错误和印刷错误,才激发查尔斯·巴贝奇(Charles Babbage, 1791—1871)设计了差分机和分析机,这是最早的专用计算机和通用计算机。这位英国剑桥大学数学教授、机械设计专家、经济学家和哲学家是国际公认的“计算机之父”。

20 世纪 40 年代,人们还用 Calculator 表示计算机器。到电子计算机出现后,才用 Computer 表示计算机。此外,硬件(Hardware)和软件(Software)来自销售人员。总线(Bus)就是公共汽车或大巴,故障和排除故障源自格瑞斯·霍普(Grace Hopper, 1906—1992)发现的“飞蛾子”(Bug)和“抓蛾子”或“抓虫子”(Debug)。其他如鼠标、菜单……不胜枚举。至于哲学家进餐问题,理发师睡觉问题更是操作系统文化中脍炙人口的经典。

以计算机为核心的信息技术,从一开始就与应用紧密结合。例如,ENIAC 用于弹道曲线的计算,ARPANET 用于资源共享以及核战争时的可靠通信。即使是非常抽象的图灵机模型,也受到二战时图灵博士破译纳粹密码工作的影响。

在信息技术中,既有许多成功的案例,也有不少失败的案例;既有先成功而后失败的案例,也有先失败而后成功的案例。好好研究它们的成功经验和失败教训,对于编写案例型教材有重要的意义。

我国正在实现中华民族的伟大复兴,教育是民族振兴的基石。改革开放 30 年来,我国高等教育在数量上、规模上已有相当的发展。当前的重要任务是提高培养人才的质量,必须从学科知识的灌输转变为素质与能力的培养。应当指出,大学课堂在高新技术的武装下,利用 PPT 进行的“高速灌输”、“翻页宣科”有愈演愈烈的趋势,我们不能容忍用“技术”绑架教学,而是让教学工作乘信息技术的东风自由地飞翔。

本系列教材的编写,以学生就业所需的专业知识和操作技能为着眼点,在适度的基础知识与理论体系覆盖下,突出应用型、技能型教学的实用性和可操作性,强化案例教学。本套教材将会有机融入大量最新的示例、实例以及操作性较强的案例,力求提高教材的趣味性和实用性,打破传统教材自身知识框架的封闭性,强化实际操作训练,使本系列教材做到“教师易教,学生乐学,技能实用”。有了广阔的应用背景,再造计算机案例型教材就有了基础。

我相信北京大学出版社在全国各地高校教师的积极支持下,精心设计,严格把关,一定能够建设出一批符合计算机应用型人才培养模式的、以案例型为创新点和兴奋点的精品教材,并且通过一体化设计、实现多种媒体有机结合的立体化教材,为各门计算机课程配齐电子教案、学习指导、习题解答、课程设计等辅导资料。让我们用锲而不舍的毅力,勤奋好学的钻研,向着共同的目标努力吧!

刘瑞挺教授 本系列教材编写指导委员会主任、全国高等院校计算机基础教育研究会副会长、中国计算机学会普及工作委员会顾问、教育部考试中心全国计算机应用技术证书考试委员会副主任、全国计算机等级考试顾问。曾任教育部理科计算机科学教学指导委员会委员、中国计算机学会教育培训委员会副主任。PC Magazine《个人电脑》总编辑、CHIP《新电脑》总顾问、清华大学《计算机教育》总策划。

前 言

Java 语言具有面向对象、与平台无关、安全、稳定和多线程等特点，不仅可以用来开发大型的应用程序，而且特别适合于开发网络应用程序。目前，无论是高校的计算机专业还是 IT 培训学校，都将 Java 作为主要的教学内容之一，这对于培养学生的计算机应用能力具有重要的意义。实践表明，这门课的教学存在一定的问题，主要表现在：学生理解抽象的程序设计语言较困难；学生的实践不充分，缺乏有效的指导，知识学习与应用能力的培养脱节。

案例教学是计算机语言教学最有效的方法之一，好的案例对学生理解知识、掌握如何应用知识都十分重要。本书围绕教学内容组织案例，对学生的知识和能力训练具有很强的针对性，主要特色如下。

- (1) 以知识线索设计案例，分解知识点，有明确的目的和要求，针对性强。
- (2) 选择有代表性的案例，突出重点知识的掌握和应用。
- (3) 将技术指导、代码与分析、应用提高和相关知识有机结合起来。
- (4) 注意新方法、新技术的应用。
- (5) 处理好具体实例与思想方法的关系、局部知识应用与综合应用的关系。
- (6) 强调实用性，培养应用能力。

本书中案例的结构模式为“案例说明→案例目的→技术要点→相关知识及注意事项→代码及分析→应用扩展”。每一章可包含多个案例，并配有相应的习题。通过强化案例和实训教学，加深学生对理论知识的理解。

课程教学目标

1. 知识教学目标

- (1) 掌握 Java 语言的特点，实现机制和体系结构；
- (2) 掌握 Java 语言中面向对象的特性；
- (3) 掌握 Java 语言提供的数据类型和结构；
- (4) 掌握 Java 语言编程的基本技术；
- (5) 学会编写 Java 用户界面程序；
- (6) 学会编写 Java 简单应用程序；
- (7) 学会编写 Java 小应用程序(Applet)；
- (8) 掌握 Java 中数据库应用程序的设计方法；
- (9) 学会利用 java.net 包进行网络编程的基本方法。

2. 能力培养目标

- (1) 具备良好的阅读程序和分析程序能力；
- (2) 具有灵活运用面向对象的思路和方法编制程序的能力。

学时分配

建议课程安排 66 课时，其中理论教学为 40 课时，实训教学为 26 课时。

任课教师教学过程中应注意的事项

建议采用启发式案例教学，应注重培养学生的创新思维能力。

本课程与其他课程的关系

先导课程有《C 语言程序设计》，后继课程有《数据结构》和《计算机网络》。

本书由潍坊学院的徐翠霞担任主编，潍坊学院的曲秀清、崔玲玲、中华女子学院山东分院的刘倩担任副主编。其中，第 2 章和第 6 章由徐翠霞编写；第 7 章和第 8 章由曲秀清编写；第 3 章和第 5 章由崔玲玲编写；第 1 章由刘倩编写；第 4 章由中华女子学院山东分院的于振梅编写；第 9 章由中华女子学院山东分院的刘平编写。全书由徐翠霞负责统稿。

由于作者水平有限，书中难免有错误和疏漏之处，恳请广大读者批评和指正，以使本书得以改进和完善。

编 者

2008 年 4 月

目 录

第 1 章 Java 语言基础.....1

1.1 “一个简单的 Java 应用程序”案例2

1. Java 语言的主要特点3

2. Java 程序的开发过程4

3. Java 程序的分类4

4. Java 运行环境6

1.2 “简易日历表”案例9

1. 标识符和关键字11

2. 常量和变量11

3. Java 数据类型13

4. 运算符和表达式14

5. 控制语句15

1.3 “Fibonacci 数列和杨辉三角形求值”

案例21

1. 声明数组和创建数组22

2. 数组的初始化23

3. 数组元素的使用24

4. 嵌套循环语句24

1.4 “凯撒密码”案例27

1. 创建字符串27

2. 访问字符串28

3. 比较字符串29

4. 字符串“+”操作30

本章小结31

习题 131

第 2 章 类、对象和接口37

2.1 “地址簿管理”案例38

1. 类的创建40

2. 方法的重载42

3. 构造方法43

4. 对象44

5. static 关键字45

6. this 关键字47

7. 内部类48

2.2 “教师学生类”案例53

1. 继承54

2. super 关键字57

3. 抽象类58

4. 接口60

5. 修饰符62

2.3 “矩形圆形类”案例65

1. 包66

2. 常用的 Java 标准包67

3. Java 程序的基本结构68

本章小结70

习题 270

第 3 章 基于 Swing 的图形用户 界面设计77

3.1 “奥运宣传图”案例78

1. 框架窗口79

2. 向 Swing 窗体中 添加组件80

3. 按钮81

4. 标签82

5. 布局管理82

6. 面板94

7. 事件处理95

8. 颜色98

3.2 “创建用户注册界面”案例101

1. 单行文本框101

2. 口令框103

3. 多行文本框104

4. 滚动面板105

5. 列表框107

6. 组合框	109	1. 文件选择对话框	180
7. 复选按钮	110	2. 基本输入输出流	183
8. 单选按钮	110	3. 文件字节流	185
9. 标准对话框	113	4. 格式字节数据流	187
3.3 “Windows 应用程序窗口” 案例	122	5. 缓冲字节流	189
1. 菜单	123	6. 文件字符流	191
2. 工具栏	128	7. 缓冲字符流	192
3. 子窗口	130	8. File 类	195
4. 桌面面板	131	5.2 “通讯录系统” 案例	200
5. 颜色选择对话框	132	1. 串行化的概念	201
6. 滑动块	135	2. 对象输入输出流	201
本章小结	144	3. RandomAccessFile 类	204
习题 3	144	5.3 “文件的压缩和解压缩” 案例	213
第 4 章 异常处理和多线程	148	1. ZipEntry 类	214
4.1 “简单的算术运算” 案例	149	2. ZipFile 类	214
1. 异常的概念	150	3. ZipInputStream 类	214
2. 异常的类层次	150	4. ZipOutputStream 类	216
3. 捕获及处理异常	151	本章小结	221
4. 抛出异常	152	习题 5	221
5. 自定义异常类	153	第 6 章 Applet 与图形、图像 和多媒体	224
6. 基本的输入输出	154	6.1 “HTML 向 Applet 传值” 案例	225
4.2 “模拟自由落体和平抛运动” 案例	160	1. Applet 简介	225
1. 多线程的概念	161	2. Applet 的类层次	225
2. 线程的状态和生命周期	161	3. Applet 的生命周期 及相关方法	226
3. 线程的优先级和调度	162	4. Applet 程序的基本结构	227
4. 线程的创建	162	5. Applet 程序与 Application 程序之间的区别	228
5. Thread 类的构造方法	164	6. HTML 文件中与 Applet 相关的标记	228
6. 多线程的基本控制	165	7. 在 Applet 中输出文字的 基本方法	229
4.3 “生产者-消费者” 案例	169	6.2 “同页 Applet 间的通信” 案例	231
1. 多线程的互斥	169	1. 同页 Applet 间的通信	231
2. 多线程的同步	170	2. Applet 与浏览器间的 通信	232
本章小结	172		
习题 4	173		
第 5 章 输入输出流	178		
5.1 “记事本” 案例	179		

6.3 “小车模型”案例	234
1. 用 Graphics 类绘图	234
2. Graphics 类的基本绘图方法	235
3. Graphics 类的绘图模式	235
4. 用 Graphics2D 绘图	236
6.4 “电子相册”案例	238
1. 在 Applet 中绘制图像	239
2. 图像跟踪技术	240
6.5 “用鼠标绘制图像”案例	243
1. 在 Java Application 中 绘制图像	244
2. 鼠标事件	244
6.6 “文字渐显”案例	247
1. 在 Applet 中实现动画的 基本原理	248
2. 如何避免动画的闪烁	248
3. Applet 中图形绘制的 基本方法与比较	249
4. 字体、颜色的设置方法	249
6.7 “音频播放器”案例	253
1. 在 Applet 中播放音频	253
2. 在 Application 中播放 音频	254
本章小结	257
习题 6	257

第 7 章 Java 数据库编程

7.1 “JDBC-ODBC 桥连接数据库” 案例	263
1. JDBC 的基本功能	263
2. JDBC 连接数据库的方式	264
3. JDBC API 的类和接口	264
7.2 “学生信息库更新”案例	269
1. 创建数据库及基本 数据表	270
2. 创建与数据库的连接	270

3. 实现数据库操作的相关 对象及其方法	270
7.3 “数据库的元数据查询”案例	277
1. 获得数据库的基本信息	278
2. 对信息结果集进行处理	278
7.4 “使用表格浏览修改数据”案例	280
1. TableModel 接口	281
2. DefaultTableModel 类的 构造方法	282
3. DefaultTableModel 类的 主要方法	282
本章小结	286
习题 7	286

第 8 章 Java 网络编程基础

8.1 “IP 地址的获取”案例	290
1. java.net 包及其类	290
2. InetAddress 类及其方法	291
8.2 “使用 URL 类访问网络资源” 案例	292
1. 统一资源定位器 URL	293
2. URL 类及其方法	294
3. JEditorPane 类及其方法	295
8.3 “基于 TCP 协议的 Socket 通信” 案例	297
1. 网络编程的基本知识	298
2. 基于 TCP 的 Socket 编程的相关类	300
8.4 “基于 UDP 协议的 Socket 编程” 实例	304
1. UDP 协议	305
2. 基于 UDP 的 Socket 编程	305
3. DatagramPacket 类	306
4. DatagramSocket 类	306
本章小结	309
习题 8	309

第9章 综合实训 311

实训1 画笔程序 311

实训2 学生信息管理系统 322

实训3 “速算24”扑克游戏 330

习题9 339

附录A Java 运算符的优先级

和结合方向 340

附录B Java 语言关键字 341

参考文献 342

第 1 章

Java 语言基础

教学目标与要求：本章从一个最简单的 Java 程序入手，介绍了两种 Java 程序的基本结构及编译运行方法，进而对程序中所涉及的 Java 的基本语法做了简单介绍。通过本章的学习，读者应该掌握以下内容：

- Java 语言的特点和 Java 程序的运行机制
- Java 开发环境的安装配置
- Java 两种程序的开发方法、过程和各自的特点
- 标识符和关键字
- 常量和变量
- Java 的基本数据类型
- 运算符和表达式
- Java 的语句
- 数组
- String 类字符串

教学重点与难点：两种 Java 程序的开发方法、过程和各自的特点；Java 的基本语法知识。

1.1 “一个简单的 Java 应用程序”案例

【案例说明】

本案例要求编写一个 Java 应用程序，在屏幕上显示“Hello Java!!”。程序编译运行界面如图 1.1 所示。

【案例目的】

- (1) 理解 Java 语言的特点。
- (2) 掌握 Java 程序的运行机制。
- (3) 掌握两种 Java 程序的开发方法、开发过程和各自的特点。

【技术要点】

创建一个名为“HelloWorld.java”的 Java 应用程序，需要使用 JDK 中的 Java 编译器“javac.exe”对其进行编译，也可利用 JCreator 简化编译和运行的过程。为了使读者更清楚地了解程序的执行过程，在 DOS 界面下进行编译和运行。具体步骤如下。

- 1) 在 Edit 编辑器中编辑 Java 源程序

(1) 在 DOS 提示符下输入 `C:\>edit` (下划线部分为用户输入的命令，“↵”表示按 Enter 键)。

(2) 在 Edit 文本编辑窗口中输入程序代码，如图 1.2 所示，并将其放于 `D:\xcx\CH1` 文件夹下，以“HelloWorld.java”文件名保存，如图 1.3 所示。

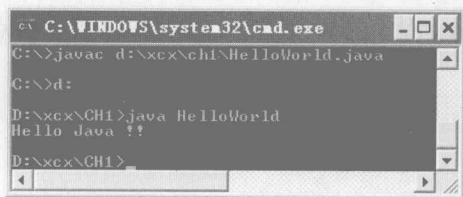


图 1.1 程序编译运行界面

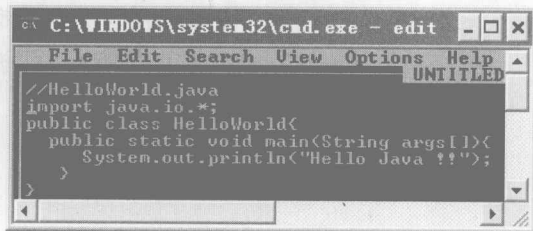


图 1.2 在 Edit 编辑器中新建文件

在图 1.3 中，选择【File】菜单下的【Save】菜单项，在弹出的窗口中，首先在 Directories 列表中选择文件保存的位置 `D:\xcx\CH1` 文件夹，然后在 File Name 处输入文件名 HelloWorld.java，最后单击【OK】按钮完成保存。保存后的界面如图 1.4 所示。

- (3) 选择【File】菜单下的【Exit】菜单项，退出 Edit 编辑器。

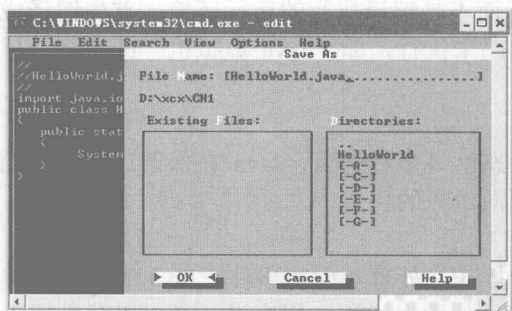


图 1.3 Edit 编辑器文件保存窗口

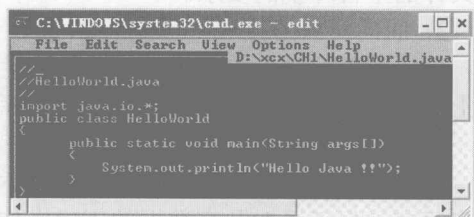


图 1.4 保存后的 Java 程序界面

2) 在 DOS 环境下编译、运行 Java 程序

(1) 为了使得在 DOS 环境下编译、解释 Java 程序更加方便,需要设置路径。例如, JDK 安装在 C:\Program Files\Java\jdk1.6.0 文件夹下,而 javac 编译器在其 bin 子文件夹下。这时应在 DOS 提示符下输入如下命令:

```
path C:\Program Files\Java\jdk1.6.0\bin;%path%✓
```

(2) 编译 HelloWorld.java 程序,执行如下命令:

```
javac D:\xcx\chl\HelloWorld.java✓
```

编译完成后,Java 程序如果有错误,则显示错误提示。例如,若将程序代码中的“String”输入为“string”,则错误提示如图 1.5 所示。如果没有任何错误,则表示编译成功。这时,在 Java 程序所在的同一文件夹下会生成同名但扩展名为“.class”的新文件。

(3) 编译成功后,在用户工作目录下执行下面的命令,解释运行 Java 程序,运行界面如图 1.1 所示。

```
D:\xcx\chl> java HelloWorld✓
```

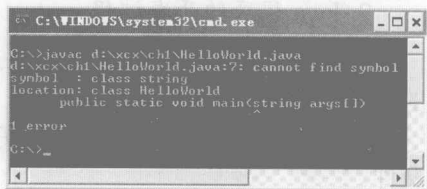


图 1.5 javac 命令出错提示

【相关知识及注意事项】

1. Java 语言的主要特点

Java 语言是一种适用于网络编程的语言,它的基本结构与 C++极为相似。Java 语言产生于 C++语言之后,是完全面向对象的编程语言。它的主要特点如下。

1) 简单性

与 C++相比,Java 不再支持运算符重载、多级继承及广泛的自动强制等易混淆和极少使用的特性,而增加了内存空间自动垃圾收集的功能,复杂性的省略和实用功能的增加使得程序开发变得简单而又可靠。

2) 平台无关性

平台无关性是 Java 最吸引人的地方。Java 是一种网络语言,而网络上充满了各种不同

类型的机器和操作系统。Java 采用了解释执行而不是编译执行的运行环境，首先编译成字节码，然后装载与校验，再解释成不同的机器码来执行，即“Java 虚拟机”的思想，屏蔽了具体的平台环境的要求。

3) 面向对象的技术

面向对象的技术具有继承性、封装性和多态性等多种优点，Java 在保留这些优点的基础上，又具有动态编程的特性，更能发挥出面向对象的优势。

4) 支持多线程

多线程机制使应用程序能同时进行不同的操作，处理不同的事件。Java 有一套成熟的同步语言，保证了对共享数据的正确操作。通过使用多线程，程序设计者可以分别用不同的线程来完成特定的行为。

5) 动态性

Java 在类库中可以自由地加入新方法和实例变量，而不影响用户程序的执行；同时，Java 通过接口来支持多重继承，使其具有更灵活的方式和扩展性。

6) 安全性

作为网络语言，Java 有建立在公共密钥技术基础上的确认技术，提供了足够的安全保障。Java 在运行应用程序时严格检查其访问数据的权限，如不允许网络上的应用程序修改本地的数据。同时，Java 程序运行稳定，轻易不会出现死机现象。

Java 语言除了具有上述主要特点外，还具有高性能、分布性、强大性、解释性和可移植性等特点。

2. Java 程序的开发过程

Java 语言包括 3 种核心机制：Java 虚拟机、垃圾收集机制和代码安全检测。Java 程序的开发过程大致分为 3 个阶段：

(1) 编写 Java 源文件。将编辑好的源程序以扩展名“.java”保存起来，即保存成“*.java”文件。

(2) 编译 Java 源程序。使用 Java 编译器编译“*.java”源程序，从而得到字节码文件“*.class”。

(3) 运行 Java 程序。

3. Java 程序的分类

根据程序结构和运行环境的不同，Java 程序可以分为两类：Java 应用程序(Java Application)和 Java 小应用程序(Java Applet)。应用程序以 main()方法作为程序入口，由 Java 解释器加载执行。Java 应用程序是完整的程序，能够独立运行，而 Java Applet 小应用程序不使用 main()方法作为程序入口，需要嵌入到 HTML 网页中运行，由 appletviewer 或其他支持 Java 的浏览器加载执行，不能独立运行。无论哪种 Java 源程序，都用扩展名为“.java”的文件保存。

1) Java Application 程序

(1) 类定义。

一个 Java 源程序是由若干个类组成的，本案例中的 Java 应用程序只有一个类。class 是 Java 的关键字，用来定义类。public 也是关键字，用来声明一个类是公共类。

源文件的命名规则如下：如果一个 Java 源程序中有多类，那么只能有一个类是 public

类；如果有一个类是 `public` 类，那么 Java 源程序的名字必须与这个类的名字完全相同，扩展名是 `“.java”`；如果源文件中没有 `public` 类，那么源文件的名字只要和某个类的名字相同，而且扩展名是 `“.java”` 就可以了。

(2) `main()` 方法。

应用程序的入口是 `main()` 方法，它有固定的书写方式为：

```
public static void main(String args[]){
    ...
}
```

`main()` 方法之后的两个大括号及括号内的内容叫做方法体。一个 Java 应用程序必须有且仅有一个类含有 `main()` 方法，这个类称为应用程序的主类。`public`、`static` 和 `void` 用于对 `main()` 方法进行声明。在一个 Java 应用程序中，`main()` 方法必须被声明为 `public`、`static` 和 `void`，`public` 声明 `main()` 是公有的方法，`static` 声明 `main()` 是一个类方法，可以通过类名直接调用，而 `void` 则表示 `main()` 方法没有返回值。

在定义 `main()` 方法时，`String args[]` 用来声明一个字符串类型的数组 `args`，它是 `main()` 方法的参数，用来接收程序运行时所需要的参数。

(3) `import` 关键字。

`import` 关键字用于引入类库或类包。包是 Java 用来组织类的文件夹，一组相关的类放在同一个包中，便于编程时引入和使用，同时可以避免类的命名冲突。

(4) 注释。

`“//”` 用于单行注释。注释从 `“//”` 开始，终止于行尾。

`“/****/”` 用于多行注释。注释从 `“/*”` 开始，到 `“*/”` 结束。例如：

```
import java.io.*;           //引入包
public class HelloWorld{    //定义类
    public static void main(String args[]){ //main()方法
        System.out.println("Hello Java !!"); //输出数据
    }
}
```

2) Java Applet 程序

一个 Java Applet 也是由若干个类组成的，一个 Java Applet 不再需要 `main()` 方法，但必须有且仅有一个类扩展了 `Applet` 类，即它是 `Applet` 类的子类，这个类称为 Java Applet 的主类，Java Applet 的主类必须是 `public` 的，`Applet` 类是系统提供的类。

`Applet` 与 `Application` 的区别在于其执行方式不同。`Application` 是从 `main()` 方法开始运行的，而 `Applet` 是在浏览器中运行的，必须创建一个 HTML 文件，通过编写 HTML 代码告诉浏览器载入何种 `Applet` 以及如何运行。

开发 Java Applet 程序的步骤如下。

(1) 编写 Applet 源程序，将其保存为 `“.java”` 文件。

(2) 编译源程序，生成字节码文件 `“.class”`。如果源文件包含了多个类，则会生成多个扩展名为 `“.class”` 的文件，都和源文件存放在相同的目录下。如果对源文件进行了修改，