

SAMS
**Teach
Yourself**

- 全球销量逾百万册的系列图书
- 连续十余年打造的经典品牌
- 直观、循序渐进的学习教程
- 掌握关键知识的最佳起点
- 秉承Read Less, Do More (精读多练)的教学理念
- 以示例引导读者完成最常见的任务

每章内容针对初学者精心设计, **1**小时轻松阅读学习,
24小时彻底掌握关键知识

每章**案例与练习题**助你轻松完成常见任务,
通过**实践**提高应用技能, 巩固所学知识

涵盖
Java 7和
Android
知识

Java

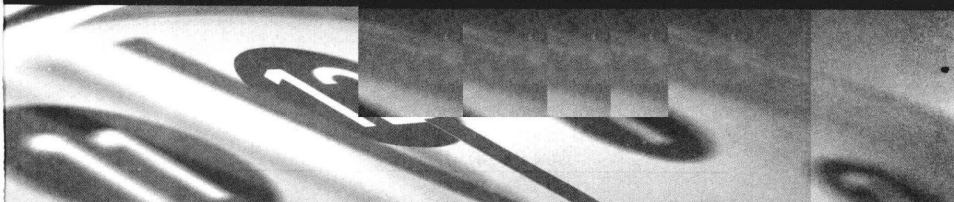
入门经典 (第6版)

[美] Rogers Cadenhead 著
梅兴文 郝记生 译

Java

入门经典 (第6版)

[美] Rogers Cadenhead 著
梅兴文 郝记生 译



人民邮电出版社
北京

图书在版编目(CIP)数据

Java入门经典：第6版 / (美)卡登海德
(Cadenhead, R.) 著；梅兴文，郝记生译。—北京：人
民邮电出版社，2012.3
ISBN 978-7-115-27181-5

I. ①J… II. ①卡… ②梅… ③郝… III. ①
JAVA语言—程序设计 IV. ①TP312

中国版本图书馆CIP数据核字(2011)第258605号

版 权 声 明

Rogers Cadenhead: Sams Teach Yourself Java in 24 Hours (Six Edition)

ISBN: 0672335751

Copyright © 2012 by Sams Publishing.

Authorized translation from the English languages edition published by Sams.

All rights reserved.

本书中文简体字版由美国 Sams 出版公司授权人民邮电出版社出版。未经出版者书面许可，对本书任何部分不得以任何方式复制或抄袭。

版权所有，侵权必究。

Java 入门经典 (第 6 版)

-
- ◆ 著 [美] Rogers Cadenhead
译 梅兴文 郝记生
责任编辑 傅道坤
 - ◆ 人民邮电出版社出版发行 · 北京市崇文区夕照寺街 14 号
邮编 100061 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京铭成印刷有限公司印刷
 - ◆ 开本：787×1092 1/16
印张：19
字数：470 千字 2012 年 3 月第 1 版
印数：1-3 500 册 2012 年 3 月北京第 1 次印刷

著作权合同登记号 图字：01-2011-4670 号

ISBN 978-7-115-27181-5

定价：39.00 元

读者服务热线：(010)67132705 印装质量热线：(010)67129223

反盗版热线：(010)67171154

广告经营许可证：京崇工商广字第 0021 号

内容提要

本书通过大量示例程序循序渐进地引导读者快速掌握使用 Java 开发程序的基本技能。

本书总共 24 章，先讲解了 Java 程序的编写流程、工作原理等内容；然后介绍了有关 Java 编程的基本知识，包括变量、条件语句、循环语句、数组和对象等内容；随后介绍了创建图形用户界面、编写交互式 Web 程序、读写文件，以及使用字体、颜色和图形等相关的知识。本书还介绍了如何使用 Java 来开发 Android app。本书每章都提供了示例程序清单，并辅以示例输出和代码分析，以阐述该章介绍的主题。为加深读者对所学内容的理解，每章末尾都提供了常见问题及其答案以及练习和测验。

本书可作为初学者学习 Java 编程技术的教程，也可供其他语言的程序员学习 Java 时参考。

关于作者

Rogers Cadenhead 是一名作家、计算机程序员、Web 开发人员，他已经编写了 20 多本与 Internet 相关的图书，其中包括《Sams Teach Yourself Java in 21 Days》。他维护着 Drudge Retort 和其他站点，这些站点的年访问量有 2000 万。本书的官方网站是 www.java24hours.com。

献 词

这一次我要打破传统，不再将本书献给我的家人和朋友，而是献给 James Gosling、Mike Sheridan、Kim Polese、Bill Joy 等人，是你们在 1995 年发明了令人惊叹的 Java 编程语言。当我第一次看到 Java 语言在 Web 页面中运行时，就惊叹不已，现在使用该语言编写的 app 更是运行在全世界数百万台 Android 手机上。Java 语言的出现证明了你们在 Sun 公司卓越工作的价值。

致 谢

感谢 Sams 出版社的员工，尤其是 Mark Tabe、Songlin Qiu、Tonya Simpson、Charlotte Kughen 和 Boris Minkin。没有作者可以凭借一己之力来写出一本书。他们出色的工作给了我很多帮助。

谢谢我的妻子 Mary 和儿子 Max、Eli、Sam。尽管他们没有帮我实现当高空飞人杂技演员的梦想，但是在我们这样一个恐高症家庭中，我依然是世界上最自豪的丈夫和父亲。

前 言

作为一名计算机图书作者，我花费了大量的时间呆在书店的计算机图书区，在假装阅读最新一期《In Touch Weekly》杂志的同时，观察读者阅读图书的行为。

根据我的观察，读者拿起本书并翻到前言后，在他将书放下前往咖啡厅喝杯咖啡前，给我留下的时间只有大约 12 秒。

因此，这里长话短说：使用 Java 来进行计算机编程比想象的容易。我本不应该这样说，因为数以千计的程序员正是凭借其 Java 技能在软件开发、Web 应用程序编程和移动 app 开发领域获得了高薪职位，对他们来说，最不想让老板知道的是，只要坚持不懈并有一点空闲时间，任何人都能学会当前使用最为广泛的编程语言。通过阅读本书，你可以快速掌握 Java 编程。

任何人都能学会如何编写计算机程序，尽管他们不能对 DVR 进行编程。Java 是最值得学习的编程语言之一，因为这种功能强大的实用技术已被全球大量的程序员采用。

本书是为非程序员、讨厌学习编程的程序员新手，以及经验丰富但想快速掌握 Java 的老程序员编写的。本书使用的是该语言的最新版本——Java 7。

由于 Java 具有“让一切成为可能”的特性，因此成为一种非常流行的编程语言。你可以使用 Java 来创建具有图形用户界面的程序、设计充分利用 Internet 的软件、读取 XML 数据、开发在 Android 手机上运行的游戏，等等。

本书引导读者从零开始学习 Java 编程，它以平实的语言阐述概念，并包含大量要求读者逐步创建的示例程序。读完本书，读者就能编写自己的 Java 程序，对自己使用该语言的能力充满信心，进而更深入地学习它；读者还将获得日益重要的技能，如网络计算、图形用户界面设计和面向对象编程。

就当前而言，这些术语对你来说也许并不重要。事实上，正是它们使得人们对编程充满了恐惧，并认为很难掌握。然而，如果你能使用计算机计算账目的收支平衡，或者能在 Facebook 上创建电子相簿，通过阅读本书就能够编写计算机程序。

现在，如果你还是愿意去喝咖啡而不是学习 Java，请将本书放回书架，并将封面朝向书店中人来人往的通道，以便别人找到它。

目 录

第 1 章 成为程序员	1	2.8 运行 Java 程序	15
1.1 选择编程语言	2	2.9 总结	16
1.2 告诉计算机做什么	3	2.10 问与答	16
1.3 程序的工作原理	5	2.11 测验	17
1.4 为什么程序不能正常工作	5	2.11.1 问题	17
1.5 选择 Java 编程工具	5	2.11.2 答案	17
1.6 安装 Java 开发工具	6	2.12 练习	18
1.7 总结	6	第 3 章 Java 之旅	19
1.8 问与答	7	3.1 第一站: Oracle	19
1.9 测验	7	3.2 去 Java 学校	21
1.9.1 问题	7	3.3 在 JavaWorld 用午餐	22
1.9.2 答案	7	3.4 在 NASA 仰望天穹	24
1.10 练习	8	3.5 回归正题	24
第 2 章 编写第一个程序	9	3.6 到 Java Boutique 去问路	25
2.1 编写程序所需的工具	9	3.7 在手机上运行 Java	26
2.2 创建 Saluton 程序	10	3.8 总结	27
2.3 开始输入程序	10	3.9 问与答	27
2.3.1 class 语句	11	3.10 测验	28
2.3.2 main 语句的作用	12	3.10.1 问题	28
2.3.3 大括号	12	3.10.2 答案	28
2.4 在变量中存储信息	13	3.11 练习	28
2.5 保存编写好的程序	13	第 4 章 理解 Java 程序的工作原理	29
2.6 将程序编译为 class 文件	14	4.1 创建应用程序	29
2.7 修复错误	14	4.2 向应用程序传递参数	30

4.3	创建 applet	32	6.10	测验	56
4.4	总结	34	6.10.1	问题	56
4.5	问与答	34	6.10.2	答案	56
4.6	测验	34	6.11	练习	56
4.6.1	问题	34	第 7 章	使用条件测试进行判断	57
4.6.2	答案	35	7.1	if 语句	58
4.7	练习	35	7.1.1	小子和大于的比较	58
第 5 章	在程序中存储和修改信息	36	7.1.2	相等和不等	58
5.1	语句和表达式	36	7.1.3	使用块语句组织程序	59
5.2	指定变量类型	37	7.2	if-else 语句	60
5.2.1	整数和浮点数	37	7.3	switch 语句	61
5.2.2	字符和字符串	38	7.4	条件运算符	63
5.2.3	其他数值类型的变量	39	7.5	观察时钟	63
5.2.4	布尔型变量	39	7.6	总结	66
5.3	给变量命名	40	7.7	问与答	66
5.4	在变量中存储信息	40	7.8	测验	67
5.5	运算符	41	7.8.1	问题	67
5.5.1	变量的递增与递减	42	7.8.2	答案	67
5.5.2	运算符优先级	43	7.9	练习	68
5.6	使用表达式	44	第 8 章	使用循环重复执行操作	69
5.7	总结	46	8.1	for 循环	69
5.8	问与答	46	8.2	while 循环	71
5.9	测验	47	8.3	do-while 循环	72
5.9.1	问题	47	8.4	退出循环	73
5.9.2	答案	47	8.5	给循环命名	73
5.10	练习	47	8.6	测试计算机的运行速度	75
第 6 章	使用字符串来交流	48	8.7	总结	76
6.1	在字符串中存储文本	48	8.8	问与答	76
6.2	在程序中显示字符串	49	8.9	测验	76
6.3	在字符串中使用特殊字符	50	8.9.1	问题	77
6.4	拼接字符串	50	8.9.2	答案	77
6.5	将其他变量用于字符串中	51	8.10	练习	77
6.6	字符串的高级处理	52	第 9 章	使用数组存储信息	78
6.6.1	比较两个字符串	52	9.1	创建数组	79
6.6.2	确定字符串的长度	52	9.2	使用数组	80
6.6.3	改变字符串的大小写	53	9.3	多维数组	81
6.6.4	查找字符串	53	9.4	对数组进行排序	82
6.7	导演及演员名单	54	9.5	对字符串中的字符计数	83
6.8	总结	55	9.6	总结	85
6.9	问与答	55			

9.7 问与答	85	11.9.2 答案	110
9.8 测验	86	11.10 练习	110
9.8.1 问题	86	第 12 章 充分利用现有对象	111
9.8.2 答案	86	12.1 继承的威力	111
9.9 练习	86	12.1.1 继承行为和属性	112
第 10 章 创建第一个对象	87	12.1.2 覆盖方法	112
10.1 面向对象编程的工作原理	87	12.2 建立继承	113
10.2 对象示例	88	12.3 使用现有的对象	114
10.3 什么是对象	89	12.4 将相同类型的对象存储到 Vector 中	115
10.4 理解继承	90	12.5 创建子类	117
10.5 建立继承层次	90	12.6 总结	119
10.6 转换对象和简单变量	91	12.7 问与答	119
10.6.1 简单变量的类型转换	92	12.8 测验	120
10.6.2 对象类型转换	92	12.8.1 问题	120
10.6.3 在简单变量和对象之间 进行转换	93	12.8.2 答案	120
10.6.4 自动封装和拆封	94	12.9 练习	120
10.7 创建对象	95	第 13 章 创建简单的用户界面	121
10.8 总结	97	13.1 Swing 和抽象窗口工具包	121
10.9 问与答	97	13.2 使用组件	122
10.10 测验	97	13.2.1 窗口和框架	122
10.10.1 问题	97	13.2.2 按钮	125
10.10.2 答案	98	13.2.3 标签和文本框	127
10.11 练习	98	13.2.4 复选框	127
第 11 章 描述对象	99	13.2.5 组合框	128
11.1 创建变量	99	13.2.6 文本区域	129
11.2 创建类变量	101	13.2.7 面板	130
11.3 用方法来创建行为	102	13.3 创建自己的组件	130
11.3.1 声明方法	102	13.4 总结	133
11.3.2 参数不同的类似方法	103	13.5 问与答	133
11.3.3 构造函数	103	13.6 测验	133
11.3.4 类方法	104	13.6.1 问题	133
11.3.5 方法中变量的作用域	105	13.6.2 答案	134
11.4 将一个类放在另一个类中	105	13.7 练习	134
11.5 使用关键字 this	106	第 14 章 用户界面的布局	135
11.6 使用类方法和类变量	107	14.1 使用布局管理器	135
11.7 总结	109	14.1.1 GridLayout 管理器	137
11.8 问与答	109	14.1.2 BorderLayout 管理器	137
11.9 测验	109	14.1.3 BoxLayout 管理器	138
11.9.1 问题	109		

14.1.4	使用 Insets 将组件隔开	139	17.2	将 applet 放到 Web 页面中	172
14.2	应用程序的界面布局	139	17.3	创建 applet	173
14.3	总结	143	17.3.1	在 applet 窗口中绘画	174
14.4	问与答	143	17.3.2	测试 SalutonApplet 程序	174
14.5	测验	143	17.4	从 Web 页面传递参数	175
14.5.1	问题	144	17.5	在 applet 中接收参数	176
14.5.2	答案	144	17.6	在 applet 中处理参数	176
14.6	练习	144	17.7	使用 object 标记	178
第 15 章	响应用户输入	145	17.8	总结	178
15.1	让程序监听	145	17.9	问与答	179
15.2	设置要监听的组件	146	17.10	测验	179
15.3	处理用户事件	146	17.10.1	问题	179
15.3.1	复选框和组合框事件	147	17.10.2	答案	179
15.3.2	键盘事件	148	17.11	练习	180
15.3.3	启用和禁用组件	149	第 18 章	处理程序中的错误	181
15.4	完善图形应用程序	150	18.1	异常	181
15.5	总结	157	18.1.1	在 try-catch 块中捕获异常	182
15.6	问与答	157	18.1.2	捕获多种不同的异常	184
15.7	测验	157	18.1.3	出现异常后进行处理	186
15.7.1	问题	158	18.1.4	抛出异常	186
15.7.2	答案	158	18.1.5	忽略异常	188
15.8	练习	158	18.2	抛出和捕获异常	188
第 16 章	创建复杂的用户界面	159	18.3	总结	190
16.1	滚动窗格	159	18.4	问与答	190
16.2	滑块	161	18.5	测验	191
16.3	变更监听器	162	18.5.1	问题	191
16.4	使用图像图标和工具栏	165	18.5.2	答案	191
16.5	总结	168	18.6	练习	191
16.6	问与答	169	第 19 章	创建线程程序	192
16.7	测验	169	19.1	线程	192
16.7.1	问题	169	19.1.1	降低程序的速度	192
16.7.2	答案	169	19.1.2	创建线程	193
16.8	练习	169	19.2	使用线程	196
第 17 章	创建交互式 Web 程序	170	19.2.1	声明类	196
17.1	标准 applet 方法	170	19.2.2	创建变量	197
17.1.1	在 applet 窗口中绘画	171	19.3	从 init() 开始	197
17.1.2	初始化 applet	172	19.4	在创建 URL 时捕获错误	198
17.1.3	启动和停止 applet	172	19.5	在 paint() 方法中处理屏幕更新	198
17.1.4	销毁 applet	172	19.6	启动线程	199

19.6.1 运行线程	199	22.2 创建服务实现 Bean	230
19.6.2 停止线程	200	22.3 发布 Web 服务	231
19.7 处理鼠标单击	200	22.4 使用 Web 服务描述语言文件	232
19.8 循环显示链接	201	22.5 创建 Web 服务客户端	234
19.9 总结	203	22.6 总结	236
19.10 问与答	203	22.7 问与答	236
19.11 测验	203	22.8 测验	237
19.11.1 问题	203	22.8.1 问题	237
19.11.2 答案	204	22.8.2 答案	237
19.12 练习	204	22.9 练习	237
第 20 章 读写文件	205	第 23 章 创建 Java2D 图形	239
20.1 流	205	23.1 使用 Font 类	239
20.1.1 文件	206	23.2 使用 Color 类	240
20.1.2 从流中读取数据	207	23.3 创建自定义颜色	241
20.1.3 缓冲输入流	209	23.4 绘制直线和形状	241
20.2 将数据写入流中	211	23.4.1 绘制直线	242
20.3 读写配置属性	212	23.4.2 绘制矩形	242
20.4 总结	215	23.4.3 绘制椭圆和圆	243
20.5 问与答	215	23.4.4 绘制弧线	243
20.6 测验	215	23.5 绘制饼图	244
20.6.1 问题	215	23.6 总结	249
20.6.2 答案	216	23.7 问与答	250
20.7 练习	216	23.8 测验	250
第 21 章 读写 XML 数据	217	23.8.1 问题	250
21.1 创建 XML 文件	217	23.8.2 答案	251
21.2 读取 XML 文件	220	23.9 练习	251
21.3 读取 RSS 聚合内容 (Syndication Feeds)	224	第 24 章 编写 Android app	252
21.4 总结	225	24.1 Android 简介	252
21.5 问与答	226	24.2 创建 Android app	254
21.6 测验	226	24.2.1 剖析一个 Android 新项目	255
21.6.1 问题	226	24.2.2 创建 app	256
21.6.2 答案	226	24.2.3 安装 Android 模拟器	257
21.7 练习	227	24.2.4 创建调试配置	259
第 22 章 利用 JAX-WS 开发 Web 服务	228	24.3 运行 app	260
22.1 定义服务端点接口	228	24.4 设计真实的 app	261
使用注解来简化 Java 代码	229	24.4.1 组织资源	262
		24.4.2 配置 app 的 Manifest 文件	264
		24.4.3 设计用户界面	265
		24.4.4 编写 Java 代码	267

24.5	总结	272	B.3.2	Caféau Lait	282
24.6	问与答	272	B.3.3	Workbench	283
24.7	测验	273	B.3.4	Java 7 Developer Blog	283
24.7.1	问题	273	B.3.5	其他 Java 博客	283
24.7.2	答案	273	B.3.6	InformIT	283
24.8	练习	273	B.3.7	Stack Overflow	283
附录 A	使用 NetBeans IDE	275	B.3.8	Java Review Service	283
A.1	安装 NetBeans	275	B.3.9	JavaWorld 杂志	283
A.2	创建新项目	276	B.3.10	Developer.com's Java Directory	284
A.3	创建新的 Java 类	277	附录 C	本书站点	285
A.4	运行应用程序	278	附录 D	设置 Android 开发环境	287
A.5	修复错误	279	D.1	起步	287
附录 B	Java 资源	281	D.2	安装 Eclipse	287
B.1	可以考虑的其他书	281	D.3	安装 Android SDK	288
B.2	Oracle 公司的 Java 官方站点	282	D.4	安装在 Eclipse 中使用的 Android 插件	289
B.3	其他 Java 站点	282	D.5	设置你的手机	291
B.3.1	本书英文版的配套网站	282			

第 1 章

成为程序员

本章介绍如下内容：

- 选择要学习的编程语言；
- 使用程序来让你的计算机运转起来；
- 程序的工作原理；
- 修复程序中的错误；
- 选择 Java 开发工具；
- 为编写程序做好准备。

你可能听到过下面有关计算机编程的说法：计算机编程极其困难，它要求你获得计算机科学专业的学位，需要投入几千美元来购买计算机硬件和软件，需要极强的分析能力，需要有耐心，而且对含咖啡因的饮料有强烈的爱好。

其实，在上述条件中，除喜欢喝含咖啡因的饮料外，其他所有条件都是不正确的。尽管多年以来计算机程序员一直都说编程很难，而且这种说法可以让程序员更容易找到高薪工作，但是，编程工作其实比大多数人想象的要容易。

现在是学习编程的最佳时代。我们可以从 Internet 上免费下载各种编程工具，数以千计的程序员以开源的形式发布他们编写的程序，以方便他人检查并修正程序中的错误，以及对程序进行改进。即使是在经济紧缩时期，许多公司也仍然在招聘程序开发人员。

如今，数以百万计的移动设备采用的都是 Android（其所有应用程序都以 Java 语言编写的一种操作系统）。如果你有一台 Android 手机，每当你在搜索电影、进行导航服务，以及玩“愤怒的小鸟”游戏时，你用到的都是 Java 程序员开发的软件。

本书旨在向两类人群讲授 Java 编程知识：没有任何编程知识的人，有过编程经验但是又讨厌编程（如同伏地魔讨厌英国孤儿院的孩子一样）的人。本书尽可能采用通用语言而不是

行话或者晦涩的首字母缩写，并对新出现的编程术语进行详细解释。

如果这种尝试得以成功，那么读者在读完本书后将获得丰富的编程技能，而这些技能对读者来说曾经是一种挑战。阅读本书后，读者将能够编写程序，更自信地阅读其他编程图书，从而更轻松地学习新的编程语言（请注意，我说的是“编程”语言，而不是西班牙语、法语以及克林贡这种外星语言）。当然，读者也会掌握 Java 这种应用最为广泛的编程语言。

本章将会简单介绍编程的概念，然后讲解计算机的一些设置指令，以为编写 Java 程序做好准备。

1.1 选择编程语言

如果你能够娴熟地使用计算机来准备一个漂亮的简历，或者是计算账目的收支平衡，抑或是将你在假期中拍摄的照片上传到 Facebook 上与他人共享，那么你就能在计算机上编写程序。

学习计算机编程的关键是选择一门合适的语言，而编程语言的选择往往与所要完成的任务有关。每一种编程语言都非尽善尽美。多年以来，人们在学习编程时，使用的都是各种版本的 BASIC 编程语言，原因是 BASIC 针对的就是初学者。

By the Way

注意：

BASIC 语言发明于 20 世纪 60 年代，它的初衷是方便学生和初学者学习，BASIC 中的字母 B 代表初学者（Beginner's）。使用 BASIC 语言的缺点是容易养成马虎的编程习惯。

人们使用 Microsoft Visual Basic 编写了数千个复杂的用于商业目的或个人用途的程序。但是，使用 Visual Basic 的各种版本编写的程序在运行速度上要比其他语言（比如 C# 和 Visual C++）编写的慢。当程序中使用了大量的图形时（比如游戏），这种速度差异更为明显。

本书讲解的 Java 编程语言是由 Oracle 公司开发的（译者注：Java 语言其实是由 Sun 公司开发的，但是 Sun 公司于 2010 年被 Oracle 公司收购，所以作者在这里说“Java 编程语言是由 Oracle 开发的”）。虽然 Java 语言学习起来比 Visual Basic 等语言困难，但基于几个原因，将其作为入门语言仍然是不错的选择。学习 Java 的优点之一是你可以在 Web 和移动手机上使用 Java 语言。你可以使用 Java 语言来为 Android 手机开发应用程序，可以开发网页游戏，还可以将其用在其他热门的软件开发领域。

Java 语言的另外一个重要的优点是：程序要想运行，则程序语句必须要组织有序。如果你不遵守这些规则，则编写的 Java 程序将无法运行。

刚开始编写 Java 程序时，你可能觉得 Java 的这种挑剔行为不是优点。编写完程序后，你可能要修改多处错误才能让程序通过编译，从而对这种程序编写方式产生厌烦心理。

在接下来的几章中，将介绍 Java 的这些规则以及应避开的陷阱。这些额外工作带来的好处是，程序将更可靠、更有用且没有错误。

Java 是由 Sun 公司的 James Gosling 开发设计的，旨在提供一种更好的计算机编程方式。Gosling 在开发一个项目时，不喜欢编程语言 C++ 的工作方式，因此决定创建一种能够更好地

完成这项工作的新语言。虽然在 Java 是否优于其他编程语言这一点上存在分歧，但 Java 在过去 10 年中取得的成就已经证明了它本身的优势。世界上 30 亿台设备运行的是 Java 程序。从 Java 面世到现在，已经出版了 1000 多种与它相关的图书（这是我们的第 16 本 Java 图书）。

不管 Java 是不是最好的语言，但肯定是当今最值得学习的编程语言。在第 2 章，读者将有机会尝试编写 Java 程序。

学习任何一种编程语言后，再学习其他编程语言将会比较容易。很多语言彼此相似，因此在学习新语言时，不用完全从头开始。例如，很多 C++ 或 Smalltalk 程序员发现学习 Java 相当容易，因为 Java 从这些编程语言中借鉴了许多思想。同样，C# 编程语言也从 Java 借鉴了很多思想，所以 Java 程序员也很容易就能学会 C# 编程语言。

注意：

本章多次提到了 C++，也许读者不明白它指的是什么，以及它是如何发音的。C++ 可以读成“C 加加”，它是由贝尔实验室的 Bjarne Stroustrup 开发的一种编程语言。C++ 是 C 语言的增强版，这就是 C++ 中“++”的含义。那为什么不叫 C+ 呢？本书后面将介绍，++ 是一个计算机编程笑话。

By the
Way

1.2 告诉计算机做什么

计算机程序也叫软件，它告诉计算机该做什么。计算机执行的任何操作（从启动到关机）都是由程序控制的。Windows 7 是程序，Call of Duty 游戏是程序，安装的打印机驱动软件是程序，甚至邮件病毒也是程序。

计算机程序由一系列命令组成，程序运行时，计算机按特定顺序处理这些命令。其中的命令称为语句。

如果你的家里雇佣了一名管家，而且你拥有高度紧张的 A 型人格，你需要为管家制定一组详细的命令供其遵循，如下所示。

亲爱的 Jeeves 先生，在我外出向国会请求紧急援助之时，请完成下述工作：

1. 用吸尘器打扫房间；
2. 去商场；
3. 买些酱油、芥末，并尽可能多买些加州寿司卷；
4. 回家。

谢谢！

Bertie Wooster

如果你告诉管家做什么，那么他就会灵活地完成你安排的任务：如果加州寿司卷卖没了，你的管家会给你买波士顿卷。

但是计算机的处理方式不会这么灵活。它会按照你编写的程序严格执行，一次执行一个语句。

下面是一个采用 BASIC 语言编写的计算机程序示例，相当简单。现在我们来看一下这个

程序，请读者不要纠结于该程序每一行所代表的意义。

```
1 PRINT "Shall we play a game?"
2 INPUT AS
```

将该程序翻译为自然语言，就相当于让计算机来执行如下事情：

亲爱的个人计算机，

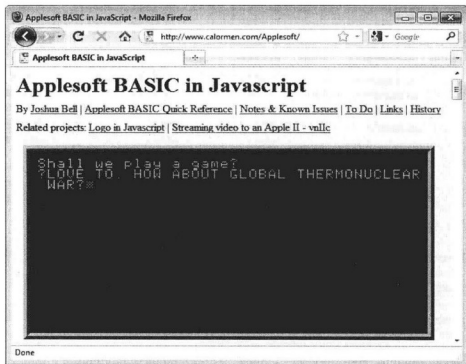
1. 请显示问题 “Shall we play a game?”
2. 给用户一个回答该问题的机会。

爱你的 Snookie Lumps

计算机程序中的每一行都是一条语句。计算机按特定顺序处理程序中的每条语句，就像厨师按菜谱炒菜，或管家 Jeeves 先生按照 Bertie Wooster 的指示行事那样。在 BASIC 语言中，行号用于标识语句的顺序。其他语言（如 Java）不使用行号，而是使用不同的方式告诉计算机如何运行程序。

图 1.1 是一个 BASIC 程序示例，该程序运行的是 Joshua Bell 的 Applesoft BASIC 解释器。该解释器在 Web 浏览器中运行，读者可在 www.calormen.com/Applesoft/ 找到它。

图 1.1
一个 BASIC 程序
示例



鉴于程序的运行方式，如果程序运行时发生错误，不能责怪计算机。毕竟，计算机只是严格按照你的指令去做。因此程序的运行错误应该归咎于程序员。这真是一个坏消息。

当然，好消息是，对编程人员和计算机而言，都不会造成永久伤害。出版本书的过程中没有人受到伤害；你在学习使用 Java 编程时，也不会损坏计算机。

**By the
Way**

注意：

“Shall we play a game (玩游戏怎么样)” 摘自 1983 年的电影 War Games。在该影片中，Matthew Broderick 扮演的年轻计算机程序员在全球几乎爆发热核战争时挽救了人类。