



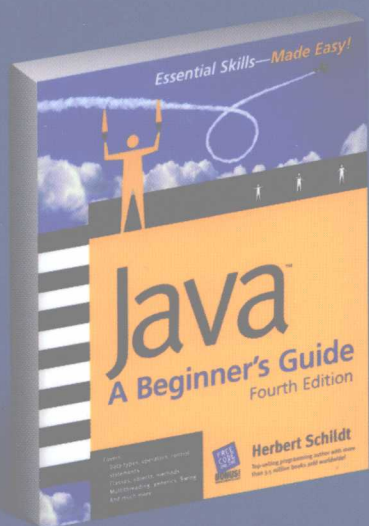
国外经典教材·计算机科学与技术

Mc
Graw
Hill Education

Java实用教程

(第4版)

Java: A Beginner's Guide Fourth Edition



使用最新的Java SE 6版本

包含泛型、applet、Swing等高级议题

全面更新的编程项目和案例

Herbert Schildt 著
杨继萍 张瑜 马海军 等译

Mc
Graw
Hill

清华大学出版社

清华大学出版社

清华大学出版社

Java实用教程

(第4版)



清华大学出版社
地址：北京清华大学学研大厦A座
邮编：100084
电话：(010) 62770175
网址：http://www.tup.tsinghua.edu.cn

清华大学出版社

国外经典教材·计算机科学与技术

Java 实用教程

(第4版)

Herbert Schildt 著

杨继萍 张瑜 马海军 等译

清华大学出版社

北京

Herbert Schildt
Java: A Beginner's Guide, Fourth Edition
EISBN: 978-0-07-226384-8

Copyright © 2007 by The McGraw-Hill Companies, Inc.

Original language published by The McGraw-Hill Companies, Inc. All Rights reserved. No part of this publication may be reproduced or distributed by any means, or stored in a database or retrieval system, without the prior written permission of the publisher.

Simplified Chinese translation edition is published and distributed exclusively by Tsinghua University Press under the authorization by McGraw-Hill Education(Asia) Co., within the territory of the People's Republic of China only, excluding Hong Kong, Macao SAR and Taiwan. Unauthorized export of this edition is a violation of the Copyright Act. Violation of this Law is subject to Civil and Criminal Penalties.

本书中文简体字翻译版由美国麦格劳-希尔教育出版(亚洲)公司授权清华大学出版社在中华人民共和国境内(不包括中国香港、澳门特别行政区和中国台湾)独家出版发行。未经许可之出口,视为违反著作权法,将受法律之制裁。未经出版者预先书面许可,不得以任何方式复制或抄袭本书的任何部分。

北京市版权局著作权合同登记号 图字 01-2008-5248 号

本书封面贴有 McGraw-Hill 公司防伪标签,无标签者不得销售。
版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Java 实用教程(第4版)/(美)希尔特(Schildt, H.)著;杨继萍等译. —北京:清华大学出版社, 2009.1

书名原文: Java: A Beginner's Guide, Fourth Edition

ISBN 978-7-302-18956-5

I. J… II. ①希… ②杨… III. JAVA 语言-程序设计-教材 IV. TP312

中国版本图书馆 CIP 数据核字(2008)第 185926 号

责任编辑:冯志强

责任校对:徐俊伟

责任印制:何 芊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京市世界知识印刷厂

装 订 者:三河市李旗庄少明装订厂

经 销:全国新华书店

开 本:185×260 印 张:32.75 字 数:812 千字

版 次:2009 年 1 月第 1 版 印 次:2009 年 1 月第 1 次印刷

印 数:1~4000

定 价:59.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:028164-01

译者序

本书面向初中级程序员和学生介绍最新的 Java 语言程序设计基础知识，体例完善，语言流畅、实例丰富，是一本在国内外有较高知名度的 Java 编程入门图书。书中采用最新的 Java SE 6 版本，通过逐步的指导帮助读者快速掌握 Java 的编程技术。全书包括 15 章，内容涉及 Java 编程基础，数据类型和运算符，程序控制语句，类、对象和方法，继承，异常处理，I/O 系统，多线程编程，泛型，接口，applet 和 Swing 等。每章都安排有学习检查、专家问答、编程练习，以及思考与练习等，便于教学和自学。

本书前两版的中译本均由清华大学出版社出版，被多所高校选作教材，得到广泛的好评。由于 Java 技术近年来发展迅速，第 4 版在第 3 版的基础上，在内容方面有较大改进，新增了对于 Swing 的介绍，改写了类、继承、泛型和接口等多方面的内容，补充了多个编程练习和专家问答的知识。

本书作者 Herbert Schildt 是世界顶级的编程书的作者，其作品已经在全球销售了 350 多万册，被翻译为各种语言的文字。他是众多畅销书的作者，他编写的 *Beginner's Guide* 和 *Complete Reference* 等编程系列图书在国内外多年来经久不衰，引导一批又一批的读者进入程序设计的殿堂。

由于本书由国外作者编写，带有很多英语的叙述方式和语言风格，虽然我们在翻译时做了适当的变通，但是在不影响阅读的情况下，尽量采用了直译的方法，所以，读者在阅读的过程中，需要稍加一点耐心。

参与本书翻译的除了封面署名人员外，还有王黎、周亚同、陈一婧、钱伟、于会方、王泽波、祁凯、李海庆、王树兴、苏静、朱俊成、王敏、赵元庆、张瑞萍、高孝峰、李乃文、吴俊海、康显丽、邵立新、辛爱军、郝相林、刘万军、王健、张仕禹、傅游、沈学利、何方、任建华、张巍屹、崔群法、王咏梅、阎迎利等人。由于时间仓促，水平有限，疏漏之处在所难免，敬请读者朋友批评指正，可以登录清华大学出版社网站 www.tup.com.cn 与我们联系。

译者

2008.10

前言

Java 是首屈一指的 Internet 语言，也是全球 Web 程序员的通用语言。今天要想成为一名专业的 Web 开发者，就必须掌握 Java。因此，如果你将来要从事基于 Internet 的编程工作，应该选择正确的语言学习——本书正是为了帮助你学习 Java 而编写的。

本书介绍了 Java 编程的基础知识。本书采用分步骤的教学方法，安排了许多示例、自我测试和编程项目。本书不需要读者具备编程经验，而是从最基本的基础知识，从如何编译并运行一个 Java 程序开始讲起。接下来讨论了每一个 Java 的关键字，还介绍了 Java 的一些最重要的高级功能，如多线程编程和泛化。本书还会介绍 Swing。最终读者将会牢固地掌握 Java 编程精髓。

值得说明的是，本书只是你学习 Java 的起点。Java 不仅仅是一些定义语言的元素，它还包括了扩展的库和工具来帮助开发程序。只有成为顶尖的 Java 程序员才能掌握这些领域的奥秘。希望读者在学习完本书之后，继续学习 Java 的其他知识。

0.1 Java 的发展历程

只有少数几种编程语言对程序设计带来过根本性的影响。其中，Java 的影响由于迅速和广泛而格外突出。可以毫不夸张地说，1995 年 Sun 公司发布的 Java 1.0 给计算机程序设计领域带来了一场变革。这场变革迅速地把 Web 带入了一个高度交互的环境，也给计算机语言的设计设置了一个新标准。

多年以来，Java 不断地发展、演化和修订。和其他语言加入新功能的动作迟缓不同，Java 一直站在计算机程序设计语言的前沿，部分原因是其变革的文化，部分原因是它所面对的变化。Java 已经做过或大或小的多次升级。

第一次主要的升级是 Java 1.1 版，这次升级比较大，加入了很多新的库元素，修订了处理事件的方式，重新配置了 1.0 版本的库中的许多功能。

第二个主要的版本是 Java 2，它代表 Java 的第二代，标志着 Java 的“现代化”的到来。Java 2 第一个发布的版本号是 1.2，该号码最初指 Java 库的内部版本号，后来就泛指整个版本号了。Java 2 被 Sun 重新包装为 J2SE (Java 2 Platform Standard Edition)，并且开始把版本号应用于该产品。

Java 的下一升级是 J2SE 1.3，它增强了一些已有的功能，并且紧凑了开发环境。J2SE 1.4 进一步增强了 Java。该版本包括一些重要的新功能，如链式异常、基于通道的 I/O，以及 assert 关键字。

Java 的下一版是 J2SE 5，它是 Java 的第二次变革。以前的几次 Java 升级提供的改进虽然重要，但都是增量式的，而 J2SE 5 却从该语言的作用域、功能和范围等方面提供了根本性的改进。

下面是 J2SE 的一些主要新功能：

- ☐ 泛型
- ☐ 自动装箱/自动拆箱
- ☐ 增强型 for-each 形式的 for 循环
- ☐ 可变长度变元
- ☐ 静态导入
- ☐ 元数据
- ☐ 注释

这些项目都是重要的升级, 每一个项目都代表了 Java 语言的一个重要改进。其中, 泛型、增强型 for 循环和可变长度变元引入了新的语法元素; 自动装箱和自动拆箱修改了语法规则; 元数据增加了一种全新的编程注释方法。

这些新功能的重要性反映在使用的版本号“5”。从版本号的变化方式看, 这一版本的 Java 应该是 1.5。由于新功能和变革如此之多, 常规的版本号升级 (从 1.4 到 1.5) 无法标识实际的变化, 所以 Sun 决定使用版本号 5。因此, 当前的版本叫做 J2SE 5, 开发工具包叫做 JDK 5。但是, 为了维持和以前的一致性, Sun 在内部使用内部版本号 1.5, 即外部版本号为 5, 内部版本号为 1.5。用户在使用编译器显示版本号, 以及搜索在线文档时, 使用的是内部版本号 1.5。

Java 的最新版本是 Java SE 6, 本书使用的就是该版本。随着 Java SE 6 的发布, Sun 公司再次修改了 Java 平台的名称, 把“2”从版本号中删除了。因此, Java 平台现在的名称是 Java SE, 官方名称是 Java Platform Standard Edition 6。和 J2SE 5 一样, Java SE 6 中的“6”是指产品的版本号, 内部的开发者版本号是 1.6。

Java SE 6 建立在 J2SE 5 的基础之上, 做了进一步的增强和改进。Java SE 6 并没有对 Java 语言本身添加较大的功能, 而是增强了 API 库, 添加了多个新包, 改进了运行时环境。总之, Java SE 6 进一步巩固了 J2SE 5 的领先地位。

0.2 本书的组织结构

本书采用教程式的组织结构, 每一章都建立在前面的基础之上。本书共分 15 章, 每一章讨论一个有关 Java 的议题。本书的特色就在于它包含了许多便于读者学习的特色内容。

- ☐ **重要技能** 每一章都包括一些重要技能, 并且在各章中以节标识。
- ☐ **思考与练习** 每一章都有思考与练习, 测试读者学习到的知识。答案在附录中提供。
- ☐ **学习检查** 每一节后面都有一个“学习检查”, 问题的答案在脚注中。
- ☐ **专家问答** 每一章中都有一些“专家问答”, 以一问一答的形式介绍补充知识和要点。
- ☐ **编程练习** 每一章中都包含 1~2 个实习项目, 帮助读者将学习的知识应用到实践中去。

0.3 本书不需要读者具备编程经验

本书假定读者没有任何编程经验。当然, 许多读者都已经或多或少有了一些编程经

验, 对于大多数读者这些编程经验就是 C++。实际上 C++ 和 Java 具有一些相似性。因此, 如果你已经学习过 C++, 学习 Java 将会非常容易。由于许多读者都已经学习过 C++, 因此, 书中会经常指出 C++ 和 Java 之间的相似之处。

0.4 本书需要的软件环境

要想编译和运行本书的程序, 需要获得最新版本的 Sun 的 Java Software Developers Kit (SDK), 本书使用的是用于 Java SE6 的 JDK6。本书在第 1 章介绍了如何获得 Java SDK。

0.5 不要忘记 Web 上的代码

本书所有示例和编程项目的源代码都可以免费从 Web 网址 www.osborne.com 获得。

目 录

第 1 章 Java 基础..... 1	第 2 章 数据类型与运算符.....28
1.1 Java 的起源..... 2	2.1 Java 的原语类型.....29
1.1.1 Java 与 C 和 C++ 的关系..... 3	2.1.1 整数类型.....29
1.1.2 Java 与 C#的关系..... 3	2.1.2 浮点型.....31
1.2 Java 对 Internet 的贡献..... 4	2.1.3 字符型.....31
1.2.1 Java Applet..... 4	2.1.4 布尔类型.....33
1.2.2 安全性..... 4	练习 2-1 闪电有多远?34
1.2.3 可移植性..... 5	2.2 字面值.....35
1.3 Java 的魔法: 字节码..... 5	2.2.1 十六进制与八进制 常量.....35
1.4 Java 的主要术语..... 6	2.2.2 字符转义序列.....35
1.5 面向对象程序设计..... 7	2.2.3 字符串字面值.....36
1.5.1 封装..... 8	2.3 变量详解.....37
1.5.2 多态性..... 8	2.3.1 初始化变量.....37
1.5.3 继承..... 9	2.3.2 动态初始化.....37
1.6 第一个简单的程序..... 10	2.4 作用域和变量的生命期.....38
1.6.1 输入程序..... 10	2.5 算术运算符.....41
1.6.2 编译程序.....11	2.6 关系运算符和逻辑运算符.....43
1.6.3 逐行分析第一个程序.....11	2.7 赋值运算符.....46
1.7 第二个简单程序..... 14	2.8 速记赋值.....46
练习 1-1 将加仑换算为升..... 17	2.9 赋值中的类型转换.....47
1.8 两个控制语句..... 18	2.10 不兼容类型的强制转换.....48
1.8.1 if 语句..... 18	练习 2-2 显示逻辑运算符 的真值表.....51
1.8.2 for 循环语句..... 20	2.11 表达式.....52
1.9 创建代码块..... 21	2.11.1 表达式中的类型 转换.....52
1.10 分号和定位..... 23	2.11.2 间距和圆括号.....54
练习 1-2 改进从加仑到升 的转换程序..... 24	2.12 思考与练习.....54
1.11 Java 的保留关键字..... 25	第 3 章 程序控制语句.....56
1.12 Java 的标识符..... 26	3.1 从键盘输入字符.....57
1.13 思考与练习..... 27	3.2 if 语句.....58

3.2.1 嵌套 if 语句.....	59	4.9 带参数的构造函数.....	114
3.2.2 if-else-if 阶梯状 结构	60	4.10 深入介绍 new 运算符	116
3.3 switch 语句.....	62	4.11 垃圾回收与终止器	117
练习 3-1 建立一个 Java 帮助系统.....	65	练习 4-2 演示垃圾回收	118
3.4 for 循环	68	4.12 this 关键字	120
3.4.1 for 循环的一些变体	69	4.13 思考与练习	122
3.4.2 缺失部分要素的 for 循环	70	第 5 章 其他数据类型与运算符	123
3.4.3 无限循环	71	5.1 数组	124
3.4.4 没有循环体的循环	71	练习 5-1 排序数组	127
3.4.5 在 for 循环内部 声明循环控制变量	72	5.2 多维数组	129
3.4.6 增强型 for 循环	73	5.3 不规则数组	130
3.5 while 循环	73	5.3.1 三维或更多维的 数组	131
3.6 do-while 循环	75	5.3.2 初始化多维数组	132
练习 3-2 改进 Java 帮助 系统	77	5.4 另一种声明数组的语法	133
3.7 使用 break 语句退出循环	79	5.5 数组引用赋值	134
3.8 将 break 语句作为一种 goto 语句使用	81	5.6 使用 length 成员	135
3.9 使用 continue	85	练习 5-2 Queue 类	137
练习 3-3 完成 Java 帮助 系统	87	5.7 for-each 形式的循环	141
3.10 嵌套的循环	90	5.8 字符串	145
3.11 思考与练习	91	5.8.1 构造字符串	146
第 4 章 类、对象和方法	93	5.8.2 操作字符串	146
4.1 类的基础知识	94	5.8.3 字符串数组	148
4.1.1 类的基本形式	94	5.8.4 字符串是不可变的	149
4.1.2 定义类	95	5.9 使用命令行参数	150
4.2 如何创建对象	98	5.10 位运算符	152
4.3 引用变量和赋值	98	5.10.1 位运算符的与、 或、异或和非	152
4.4 方法	99	5.10.2 位移运算符	156
4.5 从方法返回值	102	5.10.3 位运算符赋值 速记符	158
4.6 返回值	102	练习 5-3 ShowBits 类	158
4.7 使用参数	104	5.11 “?” 运算符	161
练习 4-1 创建 Help 类	107	5.12 思考与练习	163
4.8 构造函数	113	第 6 章 方法和类详解	164
		6.1 控制对类成员的访问	165
		练习 6-1 改进 Queue 类	169
		6.2 向方法传递对象	171
		6.3 从方法返回对象	175

6.4	方法重载	177	8.1.1	定义包	249
6.5	重载构造函数	182	8.1.2	寻找包和 CLASSPATH	250
	练习 6-2 重载 Queue 构造 函数	184	8.1.3	一个简短的包的 示例	250
6.6	递归	187	8.2	包和成员访问	252
6.7	理解 static 关键字	189	8.3	理解被保护的成员	254
	练习 6-3 快速排序	193	8.4	导入包	256
6.8	嵌套类和内部类	196	8.5	Java 的类库位于包中	258
6.9	可变长度变元	199	8.6	接口	258
	6.9.1 Varargs 基础	199	8.7	实现接口	259
	6.9.2 重载 Varargs 方法	202	8.8	使用接口引用	263
	6.9.3 Varargs 和歧义	203		练习 8-1 创建队列接口	265
6.10	思考与练习	204	8.9	接口中的变量	271
第 7 章 继承		206	8.10	接口能够扩展	272
7.1	继承的基础知识	207	8.11	思考与练习	273
7.2	构造函数和继承	212	第 9 章 异常处理		274
7.3	使用 super 访问超类成员	218	9.1	异常的层次结构	275
	练习 7-1 扩展 Vehicle 类	219	9.2	异常处理基础	275
7.4	创建多级层次结构	222		9.2.1 使用 try 和 catch	276
7.5	何时调用构造函数	225		9.2.2 一个简单的异常 示例	276
7.6	超类引用和子类对象	226	9.3	未捕获异常的结果	278
7.7	方法重写	231	9.4	使用多个 catch 语句	281
7.8	重写的方法支持多态性	234	9.5	捕获子类异常	282
	7.8.1 为何使用重写的方法	235	9.6	可嵌入的 try 代码块	283
	7.8.2 在 TwoDShape 中 应用方法重写	236	9.7	抛出异常	285
7.9	使用抽象类	240	9.8	Throwable 详解	287
7.10	使用 final	244	9.9	使用 finally	288
	7.10.1 使用 final 防止 重写	244	9.10	使用 throws	290
	7.10.2 使用 final 防止 继承	245	9.11	Java 的内置异常	292
	7.10.3 对数据成员 使用 final	245	9.12	创建异常子类	294
	7.11 Object 类	246		练习 9-1 向排序类添加 异常	295
	7.12 思考与练习	247	9.13	思考与练习	299
第 8 章 包和接口		248	第 10 章 使用 I/O		300
8.1	包	249	10.1	Java 的 I/O 基于流	301
			10.2	字节流和字符流	301
			10.3	字节流类	301

10.4	字符流类	302	11.11	线程的挂起、继续执行和 停止	365
10.5	预定义流	302		练习 11-2 使用主线程	369
10.6	使用字节流	303	11.12	思考与练习	371
	10.6.1 读取控制台输入	304	第 12 章	枚举、自动装箱和静态 导入	372
	10.6.2 写入控制台输出	305	12.1	枚举	373
10.7	使用字节流读写文件	306	12.2	Java 的枚举是类类型	376
	10.7.1 从文件输入	306	12.3	values() 和 valueOf() 方法	376
	10.7.2 写入文件	308	12.4	构造函数、方法、实例 变量和枚举	377
10.8	读写二进制数据	310	12.5	枚举继承 Enum	379
	练习 10-1 文件比较程序	314		练习 12-1 计算机控制的 交通指示灯	380
10.9	随机访问文件	316	12.6	类型包装器	386
10.10	使用 Java 字符流	319	12.7	自动装箱基础	387
	10.10.1 使用字符流的 控制台输入	320	12.8	自动装箱和方法	388
	10.10.2 使用字符流的 控制台输出	322	12.9	发生在表达式中的自动 装箱/自动拆箱	389
10.11	使用字符流的文件 I/O	324	12.10	静态导入	391
	10.11.1 使用 FileWriter	324	12.11	注释(元数据)	394
	10.11.2 使用 FileReader	325	12.12	思考与练习	396
10.12	使用 Java 类型包装器 转换数值字符串	327	第 13 章	泛型	397
	练习 10-2 创建一个基于 磁盘的帮助 系统	329	13.1	泛型基础	398
10.13	思考与练习	335	13.2	一个简单的泛型示例	398
第 11 章	多线程程序设计	336	13.3	约束类型	404
11.1	多线程基本原理	337	13.4	使用通配符变元	407
11.2	Thread 类和 Runnable 接口	337	13.5	约束通配符	410
11.3	创建一个线程	338	13.6	泛型方法	412
	练习 11-1 扩展 Thread	343	13.7	泛型构造函数	414
11.4	创建多重线程	346	13.8	泛型接口	415
11.5	确定线程何时结束	348		练习 13-1 创建一个泛型 队列	417
11.6	线程的优先级	351	13.9	原类型和遗留代码	421
11.7	同步化	354	13.10	擦除特性	424
11.8	使用同步化方法	355	13.11	歧义错误	425
11.9	同步化语句	358	13.12	一些泛型限制	426
11.10	使用 notify()、wait() 和 notifyAll() 的线程通信	360	13.13	思考与练习	428

第 14 章 Applet、事件和其他议题 ··· 429	
14.1 applet 基础····· 430	14.11.3 strictfp·····452
14.2 applet 层次结构····· 433	14.11.4 assert·····452
14.3 一个完整的 applet 框架····· 433	14.11.5 Native 方法·····452
14.4 applet 初始化与终止····· 434	14.12 思考与练习·····454
14.5 请求重绘····· 435	第 15 章 Swing 基础 ·····455
练习 12-1 一个简单的广 告条 applet····· 436	15.1 Swing 的起源和设计原则···456
14.6 使用状态窗口····· 440	15.2 组件和容器·····458
14.7 向 applet 传递参数····· 441	15.2.1 组件·····458
14.8 Applet 类····· 443	15.2.2 容器·····459
14.9 委派事件模型····· 444	15.2.3 顶级容器窗格·····459
14.9.1 事件····· 444	15.3 布局管理器·····459
14.9.2 事件源····· 444	15.4 第一个简单的 Swing 程序·····460
14.9.3 事件侦听器····· 445	15.5 使用 JButton·····465
14.9.4 事件类····· 445	15.6 使用 JTextField·····468
14.9.5 事件侦听器接口···445	15.7 创建 JCheckBox·····472
14.10 使用委派事件模型····· 446	15.8 使用 JList·····475
14.10.1 处理鼠标事件···447	练习 15-1 一个基于 Swing 的文件比较实 用程序·····478
14.10.2 一个简单的鼠标 事件 applet····· 448	15.9 使用匿名内部类处理 事件·····484
14.11 其他 Java 关键字····· 451	15.10 创建一个 Swing applet·····484
14.11.1 transient 和 volatile 修饰符··· 451	15.11 思考与练习·····487
14.11.2 instanceof····· 451	附录 思考与练习答案 ·····489

第 1 章

Java 基础

- 1.1 了解 Java 的历史和基本原理
- 1.2 理解 Java 对 Internet 的贡献
- 1.3 理解字节码的重要性
- 1.4 了解 Java 的术语
- 1.5 理解面向对象程序设计的基本原理
- 1.6 创建、编译和运行一个简单的 Java 程序
- 1.7 使用变量
- 1.8 使用 if 和 for 控制语句
- 1.9 创建代码块
- 1.10 理解如何定位、缩进和终止语句
- 1.11 了解 Java 关键字
- 1.12 理解 Java 标识符的规则

Internet 和 World Wide Web 的兴起从根本上改变了计算机的处理方式。短短数年前, 电脑空间还是由孤立的 PC 机所统治, 而今天, 几乎所有的 PC 机都接入了 Internet。最初, Internet 本身只是用于提供一种共享文件和信息的捷径, 但是今天, 它已经演变成了一个浩瀚的分布式计算空间。这些改变引发了一种新的编程方法的产生, 这就是 Java。

Java 是一种卓越的 Internet 语言, 而且不仅如此, 它还使程序设计产生了革命, 改变了我们考虑程序形式与功能的方式。今天, 要成为职业的程序员就意味着要具备使用 Java 编程的能力, 这一点非常重要。在本书的课程中, 你将学习到掌握 Java 的必要技巧。

本章的目的是向你介绍 Java, 包括它的历史、设计原理和一些最重要的特性。目前, 学习程序设计语言最大的难点是语言的各部分之间不是相互孤立的, 而是相互关联的。这种相互的关联性在 Java 中尤为突出。事实上, 只讨论 Java 的一个方面, 而不涉及其他部分是非常困难的。为了有助于克服这一困难, 本章对 Java 的几个特性进行了简单的概述, 其中包括 Java 程序的基本形式、一些基本的控制结构和运算符。对于这些内容我们并不进行深入讨论, 只是关注一下 Java 程序共有的一些概念。

1.1 Java的起源

驱使计算机语言革新的因素有两个: 即程序设计技术的改进和计算环境的改变。Java 也不例外。在大量继承 C 和 C++ 的基础之上, Java 还增加了反映当前程序设计技术状态的功能与精华。针对在线环境的蓬勃发展, Java 为高度的分布式体系结构提供了流水线程序设计功能。

Java 是 1991 年由 Sun Microsystems 公司的 James Gosling、Patrick Naughton、Chris Warth、Ed Frank 和 Mike Sheridan 共同构想的成果。这个语言最初名为 “Oak”, 于 1995 年更名为 “Java”。多少有些让人吃惊的是, 设计 Java 的最最动力并不是源于 Internet, 而是为了开发一种独立于平台的语言, 使其能够用于创建内嵌于不同家电设备, 如烤箱、微波炉和遥控器的软件。正如你可能猜想到的, 不同类型的 CPU 都可以作为控制器使用。麻烦在于多数的计算机语言都只能用于一个特定的目标。例如 C++ 即是如此。

虽然任何类型的 CPU 或许都能编译 C++ 程序, 然而这需要 CPU 有一个完整的 C++ 编译器。而编译器非常昂贵, 其开发也很耗时。为了找到更好的解决方法, Gosling 和其他人尝试开发一种可移植的交叉平台语言, 使该语言生成的代码可以在不同环境下的不同 CPU 上运行。这一努力最终导致了 Java 的诞生。

大概就是在快要设计出 Java 细节的时候, 另一个在 Java 未来中扮演关键角色的更重要的因素出现了。这个第二动力就是 World Wide Web。在 Web 还未成形的时候, Java 出炉了。对于消费类电子产品的程序设计而言, 它可能是一个有用但却晦涩的语言, 然而随着 Web 的出现, 以及 Web 对可移植语言的需求, Java 被推到了计算机语言设计的前端。

多数程序员在其早期生涯都了解可移植程序是既令人期待, 也让人难以捉摸的。虽然对创建高效可移植 (平台独立) 程序的需要几乎与程序设计学科一样陈旧的同时, 但由于诸多问题, 可移植程序退到了后台。Internet 和 Web 的出现使原有的可移植性问题重新摆上了桌面。因为, Internet 毕竟是一个由许多类型的计算机、操作系统和 CPU 组成的多样化的分布式空间。曾经恼人心绪, 却也无关紧要的问题也就成为了要亟待解决的问题。

到 1993 年, Java 设计团队的成员十分明显地发现, 在创建嵌入式代码时经常遇到的问题同样也出现在创建的 Internet 代码中。了解到这一点以后, Java 的重点从消费类电子产品转移到了 Internet 程序设计。因此, 尽管开发独立于体系结构的程序设计语言的初衷提供了起初的星星之火, 然而却是 Internet 最终促成了 Java 的燎原之势。

1.1.1 Java 与 C 和 C++的关系

Java 与 C 和 C++直接相关。Java 继承了 C 的语法, Java 的对象模型是从 C++改编来的。Java 与 C 和 C++的关系之所以重要, 是出于以下几个原因。

第一, 许多程序员都熟悉 C/C++语法。这样对于他们而言, 学习 Java 就简单了。同样, Java 程序员学习 C/C++也是很简单的。

第二, Java 设计者并没有重开炉灶。相反, 他们进一步对成功的程序设计范式进行了提炼。现代程序设计始于 C, 而后过渡到 C++, 现在则是 Java。通过大量的继承, Java 提供了一个强大的、可以更好利用已有成果的、逻辑一致的程序设计环境, 并且增加了在线环境需求的新功能。然而, 或许是最重要的一点在于, 由于它们的相似性, C、C++和 Java 为专业程序员定义了一个统一的概念架构。程序员从其中一种语言转为另一种语言时, 不会遇到太大的困难。

C 和 C++的核心设计原理之一就是程序员的控制。Java 也继承了这一原理。除了 Internet 环境施加的约束以外, Java 还为程序员提供了完全的控制。如果程序编得好, 就会体现出来, 而如果不好, 也会体现出来。不同的是, Java 并不是一种教学式语言, 它是为专业程序员准备的语言。

Java 还有一个与 C 和 C++共有的属性: 它是由真正的程序员设计、测试和修改的。它与设计者的需求和经验紧密结合。因此, 再没有比这更好的方法来创建如此一流的专业程序设计语言了。

因为 Java 与 C++的相似性, 特别是它们对面向对象程序设计的支持, 使得 Java 被简单地认为是“C++的 Internet 版”。然而, 这种观点是错误的。因为 Java 在实际应用以及基本原理上与 C++有显著的不同。尽管 Java 受到 C++的影响, 但是它绝不是 C++的增强版。例如, Java 不提供对 C++的向上或向下兼容。当然, Java 与 C++的相似是十分明显的, 如果你是一名 C++程序员, 那么在使用 Java 时会有驾轻就熟的感觉。另外, Java 不是为替代 C++而设计的, 而是为了用来解决一系列特定问题而出现的。C++则是用来解决另一个不同系列的问题的。两者将在未来几年中共存。

1.1.2 Java 与 C#的关系

由 Microsoft 创建的用以支持其 .NET 架构的 C#, 与 Java 也密切相关。事实上, C#的许多功能都是直接从 Java 改编来的。Java 和 C#共享相同的 C++语法风格, 都支持分布式程序设计, 使用相同的对象模型。它们之间当然也有不同之处, 但就整体外观而言, 两者极为相似。这就意味着, 如果你已经了解了 C#, 那么学习 Java 就很简单, 反之, 如果你将来要学的是 C#, 那么现在学到的有关 Java 的知识也会对你将来有所帮助。

鉴于 Java 与 C#两者的相似性, 自然有人要问, “C#会替代 Java 吗? ”。Java 和 C#是对两种不同类型计算环境的优化, 正如 C++会和 Java 长期共存一样, C#和 Java 也会长期共存。



1. Java 对 Internet 而言用途很大是因为它可以生成_____程序。^①
2. Java 是由什么语言直接派生出来的?^②

1.2 Java对Internet的贡献

Internet 帮助 Java 走到了程序设计的前台,而 Java 也对 Internet 产生了深远的影响。除了从整体上简化了 Web 编程任务之外,Java 还创造了一种全新的网络程序类型——applet,后者改变了在线世界对于内容的看法。Java 还解决了与 Internet 相关的棘手问题:可移植性和安全性。下面分别进行讨论。

1.2.1 Java Applet

applet 是一种特殊的 Java 程序,用于在 Internet 上传输,由兼容 Java 的 Web 浏览器自动执行。而且,applets 根据需要下载,无需和用户的进一步交互。如果用户单击一个包含 applet 的链接,applet 就会在浏览器自动下载和运行。它们通常用来显示服务器提供的数据,处理用户输入,或者提供在本地而不是服务器执行的简单函数(如贷款计算器)。也就是说,applet 支持把一些功能从服务器转移到客户机。

applet 的创建改变了 Internet 编程的方式,因为它使得对象可以在电脑空间自由地移动。一般而言,有两种主要的对象类别可以在服务器和客户机之间传递:被动信息和动态活动程序。例如,当你读取电子邮件时,就是在查看被动数据。即使你是在下载程序,程序的代码在你执行它之前也是被动的数据。与此不同的是,applet 是动态的、自执行程序。这样的程序是客户机上的活动代理,是由服务器初始化的。

尽管 applet 是人们需要的动态、网络程序,但是它们也在安全性与可移植性领域带来了严重问题。很明显,要在客户机上自动下载并且执行的程序必须保证不会带来危害。它还要能够在各种不同环境和不同操作系统中运行。Java 高效完美地解决了该问题。下面将详细介绍。

1.2.2 安全性

正如你可能意识到的,每次下载一个“普通”程序时都可能会感染病毒、“木马程序”或者其他有害代码。问题的核心是各种各样的代码获得了对系统资源未授权的访问,因此可能会带来危害。例如,病毒程序可能会通过搜索计算机的本地文件系统获取私人信息,如信用卡号、银行账号及密码。Java 为了让 applet 安全地在客户机下载并执行,必须防止 applet 发动类似的攻击。

Java 实现这种保护功能的方法是将 applet 限制于一个 Java 执行环境中,不允许它访

① 可移植。

② C 和 C++。