

新世纪
高等职业教育规划教材

Java 语言程序设计

郭雪妍 主编



机械工业出版社
CHINA MACHINE PRESS

新世纪高等职业教育规划教材

Java 语言程序设计

主编 郭雪妍
参编 孟 敬
主审 赵从军



机械工业出版社

本书通过丰富、实用的例子详细介绍了如何使用 Java 语言进行程序设计。本书包括了 Java 概述、开发工具简介、Java 语言基础、面向对象概念、错误处理、Java GUI、Java 多媒体技术、Java 数据库技术等主要内容。全书内容结构安排合理，示例引用恰当，能够由浅入深、循序渐进地讲述 Java 程序设计的方法。此外，本书的最大特点是全书所引用实例根据具体情况分别采用 JDK 和 JBuilder 两种开发工具调试通过，这在以往的 Java 程序设计教程中是很少见的。本书既可作为高职高专计算机网络及相关专业的教材，也可作为 Java 自学者的参考书。

图书在版编目 (CIP) 数据

Java 语言程序设计/郭雪妍主编. —北京: 机械工业出版社, 2003.6
新世纪高等职业教育规划教材
ISBN 7-111-12113-9

I .J… II .郭… 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 (2003) 第 034668 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

责任编辑: 贡克勤 王小东 版式设计: 霍永明 责任校对: 张晓蓉
封面设计: 张 静 责任印制: 付方敏

北京铭成印刷有限公司印刷·新华书店北京发行所发行

2003 年 6 月第 1 版第 1 次印刷

1000mm × 1400mm B5·8.375 印张·325 千字

定价: 21.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

本社购书热线电话(010)68993821、88379646

封面无防伪标均为盗版

新世纪高等职业教育规划教材编审委员会

主任委员	李维东	广东白云职业技术学院	常务副院长
副主任委员	陈周钦	广东交通职业技术学院	院长
	石令明	广西柳州职业技术学院	院长
	蔡昌荣	广州民航职业技术学院	副院长
	刘国生	番禺职业技术学院	副院长
	覃洪斌	广西职业技术学院	副院长
	姚和芳	湖南铁道职业技术学院	副院长
	韩雪清	机械工业出版社教材编辑室	副主任
委	郑伟光	广东机电职业技术学院	院长
员	张尔利	广西交通职业技术学院	院长
	谈向群	无锡职业技术学院	副院长
	陈大路	温州职业技术学院理工学区	主任
	邹宁	广西机电职业技术学院	副院长
	修德明	济源职业技术学院	副院长
	管平	浙江机电职业技术学院	副院长
	韦荣敏	广西柳州市交通学校	校长
	田玉柯	遵义航天工业学校	校长
	黄秀猛	厦门市工业学校	校长
	韩书平	新乡市高级技工学校	校长
	张毓琴	广东白云职业技术学院	兼委员会秘书

编写说明

20世纪90年代以来,我国高职高专教育为社会主义现代化建设事业培养了大批急需的各类专门人才,提高了劳动者的素质,对于建设社会主义的精神文明,促进社会进步和经济发展起到了重要作用。中共中央、国务院《关于深化教育改革,全面推进素质教育的决定》指出:“要大力发展高等职业教育”,教育部在《教育部关于加强高职高专教育人才培养工作的意见》中明确指出:“高职高专教育是我国高等教育的重要组成部分,培养拥护党的基本路线,适应生产、建设、服务第一线需要的,德、智、体、美等方面全面发展的高等技术应用性专门人才;学生应在具有必备的基础理论知识和专门知识的基础上,重点掌握从事本专业领域实际工作的基本能力和基本技能。”加入WTO以后,我国将面临人才资源的全球竞争,其中包括研究开发型人才的竞争,也包括专业技能型优秀人才的竞争。高等职业教育要适应我国现代化建设的需要,适应世界市场和国际竞争的需要,培养大批符合市场需求的、有熟练技能的高等技术应用性人才。

教材建设工作是整个高职高专教育教学工作中的重要环节,在贯彻国家教育教改精神、保证人才质量方面起着重要作用。改革开放以来,各地已出版了一批高职高专教材,但从整体上看,具有高职高专教育特点的教材极其匮乏,教材建设仍滞后于高职高专教育的发展需要。为此,根据目前高等职业教育发展的要求,机械工业出版社组织全国多所在高等职业教育办学有特色、在社会上影响较大的高职院校成立了“新世纪高等职业教育规划教材编审委员会”,选择教学经验丰富、实践能力强的骨干教师,组织、规划、编写了这套“新世纪高等职业教育规划教材”,教材首批四个系列36本(书目附后)。它凝聚着全体编审人员、编委会委员的大量心血,同时得到了各委员院校的大力支持,在此表示衷心感谢。

本套教材的作者队伍是经编审委员会严格遴选确定的,他们来自高等职业教育的第一线,教学经验丰富、业务上乘、文笔过硬,大多是各校学科和专业的带头人。他们对本专业的课程设置、教学大纲、教学教改都有深刻的认识和独到的见解,对高职教育的特色把握能力强,有较高的编写水平。这些都为编写出具有创新性、适用性强的高职教材打下了良好基础。

本套教材的编写以保证基础、加强应用、体现先进、突出以能力为本位的职教特色为指导思想,在内容上遵循“宽、新、浅、用”的原则。所谓“宽”,即知识面宽,适用面广;所谓“新”,就是要体现新知识、新技术、新工艺、新方

法；所谓“浅”，是指够用为度、通俗易懂；所谓“用”，就是要注重应用、面向实践。

本套教材的出版，促进了高等职业教育的教材建设，将对我国高等职业教育的发展产生积极的影响。同时，我们也希望在今后的使用中不断改进、完善此套教材，更好地为高等职业教育服务，为经济建设服务。

新世纪高等职业教育规划教材编审委员会

前 言

为适应高等职业技术学院和高等专科学校计算机网络专业教学的需求，机械工业出版社“新世纪高等职业教育教材编审委员会”组织了18所高等职业技术学院、高等专科学校和中等专业学校的相关专业的教师和实验技术人员，编写了“新世纪高等职业教育计算机网络技术专业系列教材”，本书是该系列中涉及网络编程方面的教材之一。本课程建议在完成“C语言程序设计”和“C++语言程序设计”课程之后开设。

全书共有九章。第1、2章简要介绍了Java语言及其开发工具；第3章介绍Java语法；第4章介绍了面向对象的知识；第5章介绍错误处理；第6章介绍了Java的GUI；第7章介绍了Java的多媒体技术；第8章介绍Java与数据库；第9章介绍了一个Java的应用实例。

本书的特点是内容覆盖面宽，讨论深入浅出，实例丰富，图文并茂，强调实用，注重实验。本书可供高等职业技术学院、高等专科学校计算机网络专业作为教材使用，也适用于其他学习Java语言编程的读者。

本书第1~4章由广东交通职业技术学院孟敬老师编写；第5~9章由广东白云职业技术学院郭雪妍老师编写。本书由广东白云职业技术学院赵从军老师主审。

由于编者水平有限，加上时间仓促，书中难免有错误和疏漏，敬请读者批评指正。

编 者
2003年3月

目 录

编写说明

前言

第 1 章 Java 语言入门	1
1.1 Java 的发展历史	1
1.1.1 Java 的诞生	1
1.1.2 Java 的现状	1
1.2 Java 语言的特点	2
1.2.1 简单易学	2
1.2.2 平台无关性	3
1.2.3 面向对象	3
1.2.4 安全稳定	3
1.2.5 多线程	4
1.3 Java 平台及其工作机制	4
1.3.1 Java 运行机制	4
1.3.2 Java 平台	5
1.4 Java Application	5
1.5 Java Applet	8
1.6 Java Servlets	12
第 2 章 Java 开发工具简介	14
2.1 J2SDK 简介	14
2.1.1 JDK 工具	16
2.1.2 Java 程序举例	23
2.2 JBuilder 开发工具简介	27
2.2.1 JBuilder 6 的主界面	28
2.2.2 JBuilder 6 的主菜单	31
2.2.3 JBuilder 6 开发举例	31
2.3 其他开发工具简介	41
第 3 章 Java 语言基础	43
3.1 标识符与关键字	43
3.1.1 标识符	43
3.1.2 关键字	43
3.2 数据类型	44

3.2.1 类型	44
3.2.2 常量与变量	45
3.2.3 复合数据类型	53
3.3 运算符与表达式	53
3.3.1 算术运算符	54
3.3.2 关系运算符	59
3.3.3 布尔运算	61
3.3.4 位运算	63
3.3.5 条件运算符	64
3.3.6 其他运算符	64
3.3.7 运算符的优先级与结合性	64
3.4 控制结构	65
3.4.1 分支语句	65
3.4.2 循环语句	71
3.4.3 其他控制结构	72
3.5 数组	77
3.5.1 数组的定义与引用	77
3.5.2 数组应用举例	79
第4章 面向对象的程序设计	83
4.1 面向对象的基本概念	83
4.1.1 对象	83
4.1.2 封装	85
4.1.3 继承	85
4.1.4 多态	85
4.1.5 多重继承	86
4.1.6 接口	86
4.1.7 包	86
4.2 类	86
4.2.1 类定义	87
4.2.2 类对象	88
4.2.3 构造函数	89
4.2.4 类修饰符	90
4.3 方法	91
4.3.1 方法的声明	91
4.3.2 方法的调用	94
4.3.3 方法的参数传递	96
4.3.4 方法的重载	98
4.4 接口	102

4.4.1	接口的定义和实现	102
4.4.2	接口的实现	103
4.4.3	接口的继承和组合	106
4.4.4	接口的多态	108
4.5	Java API	111
4.5.1	创建包	111
4.5.2	使用包	112
4.5.3	Java API 的主要包	114
4.6	JBuilder 6 代码分析	116
4.6.1	创建应用程序	116
4.6.2	代码分析	121
第 5 章	错误处理	126
5.1	语法错误	126
5.2	编译错误	127
5.3	运行时错误	128
5.4	异常处理方式	129
5.4.1	try...catch	129
5.4.2	throw 和 throws	132
第 6 章	Java GUI	134
6.1	GUI 概述	134
6.2	AWT 组件	136
6.2.1	CheckBox 组件	136
6.2.2	List 组件	139
6.2.3	Choice 组件	144
6.2.4	ScrollBar 组件	147
6.3	Swing 组件	151
6.3.1	Swing 组件的分类	151
6.3.2	版面配置管理器	152
6.3.3	Jbutton、Jlabel、JtextArea、JfieldArea、JcheckBox、JRadioButton	167
6.3.4	JFileChooser、JColorChooser、JFontChooser、JMenuBar	178
6.3.5	JOptionPane 组件	195
第 7 章	Java 多媒体技术	204
7.1	绘图	204
7.1.1	直线	204
7.1.2	矩形	206
7.1.3	椭圆	209
7.1.4	弧形	211
7.1.5	多边形	213

7.2 图像和声音	215
7.2.1 图像文件的显示	215
7.2.2 声音文件的播放	218
第 8 章 Java 与数据库	221
8.1 JDBC 概述	221
8.2 JDBC 与 ODBC 和其他 API 的比较	223
8.3 JDBC 在数据库访问的应用	223
8.4 JDBC API	224
8.4.1 JDBC 的使用方法	224
8.4.2 JDBC 接口	226
8.5 JDBC 应用	228
8.5.1 与数据库建立连接	228
8.5.2 执行查询语句	229
8.5.3 检索结果集	231
8.5.4 更新数据库操作	232
8.5.5 参数的输入和输出	234
8.6 动态数据库访问	235
8.7 用 JDBC 访问数据库示例	237
8.7.1 预备知识	237
8.7.2 使用 JDK 实现 Java 与数据库连接	240
8.7.3 使用 JBuilder 实现 Java 与数据库连接	242
第 9 章 Java 应用实例	248
9.1 Client/Server 模式	248
9.2 分布式对象的模式	248
9.3 RMI	249
9.4 CORBA	252
参考文献	258

第 1 章 Java 语言入门

1.1 Java 的发展历史

Java 是 1995 年 6 月由 SUN 公司成功开发的一种面向对象的、跨平台的高级程序设计语言。Java 是一种建立分布式应用程序（即由网络上的多个计算机执行的程序）更简单易行的技术。Java 是网络编程的最优秀工具，使 Internet 作用从作为通信工具扩展到能够运行成熟应用程序的网络。这种突破性技术使企业可以在 Internet 上部署全方位的业务服务和实时交互式信息内容。

1.1.1 Java 的诞生

1991 年，SUN 公司的 Jame Gosling、Bill Joe 等人为了在电视、烤箱等家用消费类电子产品上进行交互式操作而开发了一个名为 Oak 的软件（即一种橡树的名字，也是 Java 的前身）。Oak 是一种用于网络的精巧而安全的语言，但当时并没有引起人们的注意，SUN 公司曾依此投标一个交互式电视项目，结果也失败了。直到 1994 年下半年，Internet 的迅猛发展，环球信息网 WWW 的快速增长，他们用 Java 编制了 Hot Java 浏览器，得到了 SUN 公司首席执行官 Scott McNealy 的支持，触发了 Java 进军 Internet，Java 由此得到迅猛发展。

为了促销和法律的原因，1995 年 Oak 更名为 Java。Java 的取名也有一段趣闻，有一天，几位 Java 成员组的会员正在讨论给这个新的语言取什么名字，当时他们正在咖啡馆喝着 Java 咖啡（Java 是印度尼西亚的重要岛屿，中文名叫爪哇，盛产咖啡），有一个人灵机一动说就叫 Java 怎样，得到了其他人的赞赏，于是，Java 这个名字就这样传开了。逐渐地 Java 成为 Internet 上最受欢迎的开发与编程语言，一些著名的计算机公司纷纷购买了 Java 语言的使用权，Java 语言被美国的著名杂志 PC Magazine 评为 1995 年十大优秀科技产品（计算机类就此一项入选），随之大量出现了用 Java 编写的软件产品，受到工业界的重视与好评，认为“Java 是 20 世纪 80 年代以来计算机界的一件大事”。

1.1.2 Java 的现状

Java 是 SUN 公司推出的新一代的面向对象程序设计语言，它的平台无关性特别适合于 Internet 应用程序开发。

Java 语言有着广泛的应用前景，大体上可以从以下几个方面来考虑其应用：

- 1) 所有面向对象的应用开发，包括面向对象的事件描述、处理、综合等。
- 2) 计算过程的可视化、可操作化的软件的开发。
- 3) 动态画面的设计，包括图形图像的调用。
- 4) 交互操作的设计（选择交互、定向交互、控制流程等）。
- 5) Internet 的系统管理功能模块的设计，包括 Web 页面的动态设计、管理和交互操作设计等。
- 6) Intranet（企业内部网）上的软件开发（直接面向企业内部用户的软件）。
- 7) 与各类数据库连接查询的 SQL 语句实现。
- 8) 其他应用类型的程序。

Java 作为软件开发的一种革命性的技术，其地位已被确立，这表现在以下几个方面：

- 1) 计算机产业的许多大公司购买了 Java 的许可证，包括 IBM、Apple、DEC、Adobe、Silicon Graphics、HP、Oracel、Toshiba 以及最不愿的 Microsoft。这一点说明，Java 已得到了工业界的认可。
- 2) 众多的软件开发商开始开发支持 Java 的软件产品。
- 3) Intranet 正在成为企业信息系统最佳的解决方案，而其中 Java 将发挥不可替代的作用。Intranet 的目的是把 Internet 用于企业内部的信息系统，它的优点表现在便宜以及易于使用和管理。用户不管使用何种类型的机器和操作系统，界面是统一的 Internet 浏览器，而数据库、Web 页面、应用程序（用 Java 编的 Applet）则存在 WWW 服务器上，无论是开发人员，还是管理人员，或是用户都可以受益于该解决案。

1.2 Java 语言的特点

作为一个面世较晚的语言，Java 集中体现和充分利用现代软件技术的新成果、新技术，如面向对象、多线程等。尤其 Java 是定位于网络开发的计算机语言，几乎所有的特性都是围绕这一中心展开的，使得 Java 语言特别适合于开发网络上的应用程序。

1.2.1 简单易学

Java 最初是为对家用电器进行集成控制而设计的一种语言，因此它必须简单明了。而它也是衍生于 C++ 的，它的基础语法部分与 C 语言相似，这样对于掌握 C 语言的学生来说再学 Java 会感到易于入门。出于安全稳定性的考虑，Java 除去了 C++ 中不容易理解和掌握的部分，摒弃了容易引发程序错误的地方，使

Java 更精练，也降低了学习难度。

另外，Java 系统非常小，其基本解释程序和类支持功能只占约 40KB 大小，附加的基本标准库和线程支持功能占 175KB，这样就使 Java 程序能独立工作在非常小的系统上。

1.2.2 平台无关性

Java 应用程序可以在配置了 Java 解释器和运行环境的任何计算机上运行，这种独特的运行机制使得它具有良好的二进制级的可移植性，开发人员利用 Java 编写出与具体平台无关、普遍适用的应用程序，大大降低了应用程序开发、维护和管理开销。

随着 Internet 的发展和以网络为中心的计算机的普及，人们越来越需要一种独立于平台、代码可移植的计算技术，通过将 Java 解释器嵌入各个平台，使得 Java 应用程序不依赖于机器，达到了软件开发平台的统一。

1.2.3 面向对象

Java 是面向对象的程序设计语言。面向对象技术较好的解决了传统面向过程的软件开发中存在的种种弊端。面向对象技术的核心是以更接近于人类思维的方式建立计算机模型，它利用类和对象的机制将数据和加在其上的操作封装在一起，并通过接口与外界交互，使反映现实世界实体的各个类在程序中能够独立、自治、继承；这种方法大大提高了程序的可维护性和重用性，也有利于开发效率的提高。

Java 代码按类组织，每一个类定义构成一个对象行为的一系列方法，一个类可以继承另一个类的一些行为，虽然 Java 中的数字和布尔量等简单类型不是对象，但 Java 为所有的简单类型提供了封装对象，使得 Java 中的简单类型可以作为类来实现。

1.2.4 安全稳定

在网络上使用的应用程序最重要就是安全可靠，同时它还应该是稳定的，不易出现死机等错误。在 Java 语言中，像指针和释放内存等 C++ 功能被去掉，一切对内存的访问都必须通过对象的实例变量来实现，这样大大提高了系统的稳定性，防止了应用程序使用“特洛伊”木马等欺骗手段访问对象的私有成员，同时也避免了指针操作中容易产生的错误，Java 的特有的“沙箱”机制也保证了 Java 程序运行的可靠。

1.2.5 多线程

多线程是当今软件技术的又一项重要的成果，已经在多个领域应用，如操作系统等。

多线程机制使应用程序能够并行执行，而且同步机制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完成特定的行为，而不需要采用全局的事件循环机制，这样就很容易地实现网络上的实时交互操作。

Java 不但内置多线程功能，而且提供语言级的多线程支持，即定义了一些用于建立、管理多线程的类和方法，使得开发具有多线程功能的程序变得容易。

除此之外 Java 还有很多特性，如：分布式、体系中立性、可移植性、高性能、健壮性、半解释性语言和动态性等等。SUN 公司是这样形容自己的 Java 语言的：它是一种简单、面向对象、分布式、解释型、稳定、安全、结构中立、易移植、高性能、多线程的动态语言。这段长长的定语准确地描述了 Java 语言的基本特征，也道出了 Java 为何在网络时代火爆的秘密。

Java 的出现确实给计算机行业吹来了一股清风，它带来了许多新思想和观念，它甚至改变了人们使用计算机的方式。就连环球信息网 WWW 的创始人也说：“计算机行业发展的下一个浪潮就是 Java，并且很快就會发生。”

1.3 Java 平台及其工作机制

1.3.1 Java 运行机制

传统的高级语言按执行模式可以划分为两种：

1) 编译型，如 C、Pascal 等，编辑好的源代码程序通过编译生成的字节码经链接后就成为可以直接执行的可执行代码。

2) 解释型，如 BASIC，其程序不能直接在操作系统级上运行，需要有一个专门的解释器程序来解释执行。

Java 语言程序独特的执行模式是半编译半解释的。采用 Java 编写的程序要由编译器将其翻译成 Java 的字节码，这个字节码也是二进制文件，但它不能直接运行在一般的操作系统平台上，而必须运行在一个称为“Java 虚拟机”的操作系统之外的软件平台上。我们可以理解为：Java 字节码是 Java 虚拟机的机器码。

Java 虚拟机可以把 Java 字节码程序跟具体的软硬件平台分隔开，只要在不同类型的计算机中安装针对其特定系统平台的 Java 虚拟机，就可以屏蔽不同软硬件平台的具体差别，使得 Java 字节码程序在不同的计算机上能够面对相同的 Java

虚拟机，而不必考虑具体的平台差别，从而实现了真正的二进制代码级的跨平台可移植性。比如说，同样的一个 Java 程序，就既可以在 Windows95/NT 下运行，也可以在 Solaris 或 Macintosh 上运行。

1.3.2 Java 平台

平台一般是指程序运行所需要的软件及硬件环境，Java 平台是一种在其他平台之上的纯软件的平台。它由两部分组成：Java 虚拟机 (Java VM)；Java 应用程序接口 (Java API)。Java 的体系结构如图 1-1 所示。

前面简单介绍了 Java 虚拟机，它是 Java 平台的基础。

Java API 是各种组件的集合，这些组件的功能很多，如图形用户接口的组件等。所有用 Java 编写的程序都依赖于这一组预先定义的类的集合，Java 的类是集中在不同的“包”里，根据它们的不同作用将其划分为输入/输出、图形、用户界面、网络、实用工具等各种包，这些包的功能在以后的章节中会详细介绍。

Java 的各种程序，通过 Java API 在虚拟机在运行，这使得 Java 程序具有平台无关性。

Java 语言的开发过程如图 1-2 所示。

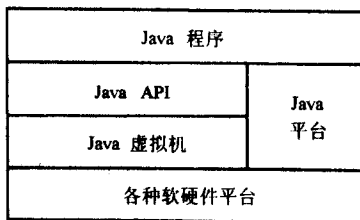


图 1-1 Java 体系结构图

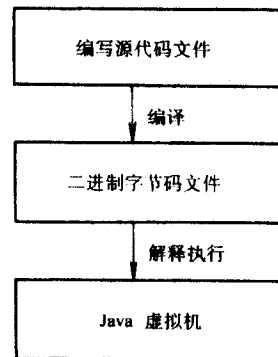


图 1-2 Java 程序开发过程图

1.4 Java Application

Java 是一种优秀的编程语言，它可以用来编写 Java 应用程序 (application)、小应用程序 (applet)、小服务程序 (servlet) 等等。简单的说，Java 小程序就是嵌入 Web 文档的程序，Java 的小服务程序就是在 Web 服务器内运行的特殊程序，而 Java 应用程序是所有其他类型的程序。

例 1-1 Java Applitcation 程序实例。

MyApplitecation _ 1. Java 源代码

```
//第一个 Java 程序
import Java.io. * ;
public class MyApplication _ 1
{
    public static void main ( String args [ ] )
    {
        System.out.println ( "Hello, Java World!" );
    }
}
```

程序的第一行是注释语句，Java 程序共有三种注释方法。

第一种：由“/*”开始，由“*/”结尾，可以多行注释。

第二种：“//”开始，只能有一行。

第三种：由“/** *”开始，由“*/”结尾。（这种可以生成帮助文档）

在上例中，第二行利用 import 语句加载 Java 语言已定义好的类或者包在本程序中使用，这行语句和 C 语言的 # include 语句加载库函数相似。

第三行的关键字 class 是表示一个类定义的开始，类定义由第二行的类头部分和下面一对大括号内的类体部分组成。

在这里要注意 Java 语言是区分大小写的，class 与 Class、CLASS 在 Java 中是代表不同的含义，在类定义中的关键字一定要用小写 class 作为标志。再有在类定义中的类名 MyApplication _ 1 一定要和文件相同。

在类体中通常有两部分组成，一部分是域，包括变量、常量、对象数组等独立的实体；另一部分是方法，是类似于函数的代码单元块，这两部分组成成分通常称为类的成员。在上面的例子中，类体里没有域，只有一个组成部分就是名为 main 的方法。方法又分为方法头和方法体两部分，上例的第五行语句是定义 main 方法的方法头。第七行语句是 main 方法的方法体部分。用来标志方法头的是—对小括号，在小括号前面的并紧靠小括号的是方法名称，如 main、run 等等；小括号内部是该方法使用的形式参数，方法名前面是用来说明这个方法属性的修饰符，其具体的规范将在后面介绍。方法体部分由若于个以分号结尾的语句组成并由一对大括号括起来，在方法体的内部不能再定义方法。

main 的方法是一个特殊的方法，它是所有 Java Application 程序执行的入口点，任何一个 Java Application 程序都要有一个 main 方法且只能有一个。还必须有统一的格式：

```
public static void main ( String [ ] args )
```