

Java

学习宝典

匡松何嘉 主编

魏乐杜小丹李丹周永 副主编

- ✓ **立足于读者：**我们总结多年开发实践经验，结合当前Java技术特点、发展前景和职业需求，为读者设计了一条清晰而有效的学习之路。
- ✓ **涵盖范围广：**本书包括5篇25章，精心筛选基础语法、关键技术、典型开发模块和中型开发项目，涵盖从Java基础学习到高效开发的各个方面。
- ✓ **实用性强：**附赠光盘中包含了一套精彩的Java数据结构和算法教学视频，便于读者在学习时语法、算法齐头并进，达到事半功倍。



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

11.6
小时DVD
多媒体教学视频

Java 学习宝典

匡 松 何 嘉 主编

魏 乐 杜小丹 李 丹 周 永 副主编

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书以一个 Java 程序员学习 Java 技术的成长路线为大纲组织内容, 全书分为 5 篇: 第 1 篇 Java 语言基础, 第 2 篇面向对象程序设计, 第 3 篇 Java 开发中的高级技术, 第 4 篇 Java Web 开发, 第 5 篇项目案例开发实践。希望通过本书能逐步提升读者的开发能力, 完成初学者向高级技术人员的蜕变。

本书以 Java 技术学习权威指南为目标, 全面覆盖 Java 技术的高级主题和项目开发方法, 包括: 多线程、集合框架、网络 API、数据库编程、分布式对象等主题, 同时涉及 JDK 的新内容以及项目开发实践中的具体技术。本书适合 Java 开发新手, 以及高等院校的学生和教师参考。

图书在版编目 (CIP) 数据

Java 学习宝典/匡松, 何嘉主编. —北京: 中国铁道出版社, 2009. 11

ISBN 978-7-113-10757-4

I. J… II. ①匡…②何… III. JAVA 语言—程序设计
IV. TP312

中国版本图书馆 CIP 数据核字 (2009) 第 208541 号

书 名: Java 学习宝典
作 者: 匡 松 何 嘉 主编

策划编辑: 严晓舟 荆 波

责任编辑: 黄园园

封面设计: 付 巍

版式设计: 郑少云

编辑部电话: (010) 63583215

封面制作: 白 雪

责任印制: 李 佳

出版发行: 中国铁道出版社 (北京市宣武区右安门西街 8 号 邮政编码: 100054)

印 刷: 三河市华业印装厂

版 次: 2010 年 1 月第 1 版

2010 年 1 月第 1 次印刷

开 本: 787mm×1 092mm 1/16 印张: 38.5 字数: 895 千

印 数: 3 500 册

书 号: ISBN 978-7-113-10757-4/TP·3636

定 价: 69.00 元 (附赠光盘)

版权所有 侵权必究

凡购买铁道版的图书, 如有缺页、倒页、脱页者, 请与本社计算机图书批销部调换。

编 委 会

主 编：匡 松 何 嘉

副主编：魏 乐 杜小丹 李 丹 周 永

编 委：

于 曦 杨明根 王若愚 侯 婉

肖 斌 刘 畅 徐 静 胡长清

赵华生 吴卫华 张小刚 张 鹏

吴 龙 王 剑 邓 荣 卢 春

Java 是由美国 Sun Microsystems 公司于 1995 年推出的一种程序设计语言。当时正是互联网 Internet 飞速发展之时，可以说是 Internet 造就了 Java 语言。

Java 语言的程序构成与 C 以及 C++ 语言非常类似。Java 吸收了 C 和 C++ 的许多优点，屏蔽了 C 和 C++ 的一些比较复杂、不安全以及并非完全面向对象等不足。经过多年发展，从大型计算机到小小手机，Java 都得到了非常广泛的应用。可以说，Java 无处不在，因而学习 Java 语言已成为当今每一个程序员的紧要任务。

Java 语言学习起来其实很简单，已具备一些 C 以及 C++ 语言基础的读者，更是能很快地过渡到 Java 语言这个神奇的世界。

Java 技术之路

与其他程序设计语言相比，Java 语言被公认为具有简单、面向对象、分布式、健壮、安全、可移植、完全与平台无关等特点。其中，完全平台无关这个特点，几乎成了程序设计界的救命稻草。要同时为大型计算机和微型计算机编写不同的应用程序，只有 Java 的跨平台特点能解决这个问题。再加之其他一些优势，Java 很快风靡全球，成为软件开发的首选语言。同时，在开源领域和一些大公司的推动下，新技术层出不穷，解决各种问题的框架也越来越多，但抽丝剥茧，基本的 Java 语言平台体系结构分为以下三个方面：

(1) Java SE 标准版本 (Java Standard Edition)：主要用于桌面应用程序的编程，是 Java 体系架构的基础平台，用于开发 C/S 架构的桌面应用程序，是学习 Java 的出发基础。

(2) Java EE 企业版本 (Java Enterprise Edition)：由 Sun 公司提出的一个标准，是建立在 Java 平台上的企业级应用的解决方案。它不但具有 Java SE 平台的所有功能，同时还提供了对 EJB、Servlet、JSP、XML 等技术的全面支持，其最终目标是成为一个支持企业级应用开发的体系结构，从而简化企业解决方案的开发、部署和管理等复杂问题。事实上，Java EE 已成为企业级开发的工业标准和首选平台，用于开发 B/S 架构的 Web 应用程序。

(3) Java ME 小型版 (Java Platform Micro Edition)：其实就是 Java 的嵌入式开发版本，应用于大量的消费类电子设备，是手机、掌上电脑、数字机顶盒等电子设备的程序开发平台。它包括虚拟机和一系列标准化的 Java API。

仔细分析一下，达到上述三个基本要求并非很难，你只需要花上 3 个月到半年的时间，找齐学习资料，自己开发一些小程序，再参加一些项目的开发，完成关键基础技术的学习，就可以实现这样的目标。而其他技术，比如 SSH 框架，了解即可。待工作之后，根据实际需要，再不断深入应用和深度掌握。

读者首先要打好基础，学习是一个从渐进到腾飞的过程。开始是入门学习，学一点知识，然后上机练习，再学点新知识，然后再上机练习。坚持下去，不断学习新知识新技能，积累到一定程度就会突然发现，自己学习任何 Java 相关技术的速度都很快了，不像开始的时候，调试一个“Hello World”也需要折腾半天。所以，一定要持之以恒坚持到自己的学习腾飞点的到来。掌握有效的学习方法，拥有比较好的学习资料，有助于提高学习效率。本书的编写，就是总结了大多数程序员的 Java 学习之路。

(1) 首先搞清楚 Java 的基本概念，明白方向。开始学习时，最好要学问结合。比如，网上有个初学者最典型的问题：JavaScript 语言与 Java 语言是什么关系，有何区别？其实 JavaScript 和 Java 之间并没有任何关系，只是它们的取名上颇为相同。一旦掌握了比较复杂的 Java 技术，JavaScript 和 Ajax 等技术的学习也就最多一个月的时间而已。所以，明确核心，抓住主要学习方向是很重要的。现在网络发达，网上热心人很多，读者可以多上网询问。另外，关于学习方向的问题，也可以给我们发邮件：beone2000@126.com。

(2) 打好基础。学习 Java 之初，一定要学好本书第 1 篇的基础内容。对于有些数值类型的基础语法，如果不能建立清晰的概念，特别是还学习过 C++ 等类似的语法，很容易产生混淆。同时，JDK 的安装和基本程序的调试是这个阶段的一个难点，要不断尝试。关于 JDK，有两个问题很容易困扰初学者：一个是 CLASSPATH 的问题，从原理上来说，其实是要搞清楚运行环境中 ClassLoader 是如何加载 Class 的；另一个问题是 package 和 import 问题，即如何寻找类的路径问题。把这两个问题摸索清楚了，就扫除了学习 Java 和使用 JDK 的最大障碍。

(3) 抓住关键的面向对象技术，即本书第 2 篇的内容。学习这部分内容时，一定要不断上机练习 Java 的面向对象语法，同时要不断思考面向对象理论。如果能做到自己看世界，感觉万事万物皆对象，那你就逐步找到 Java 面向对象开发技术的感觉了。同时，建议借阅一些面向对象开发理论的相关图书来结合进行学习。

在这个阶段，最好的学习方法就是多写代码并上机实践，来不断积累经验。当学习一个类以后，你可以写个简单的程序来运行一下，看看有什么结果，然后多调用几个类的方法，再看看运行结果。这样，可以非常直观地理解并掌握类了，而且记忆非常深刻。同时，也不能仅满足于把代码调通，还应该想想如果不这样写，换个方式再试试看行不行。通过这样的方式，你就会彻底地、精通地掌握 Java。

目 录

第 1 篇 Java 语言基础

Java 语言和目前流行的 C 语言以及 C++ 语言语法结构很相似。之所以说 Java 语言是简单的, 是因为 Java 语言虽然继承了 C 语言以及 C++ 语言很多优点, 但是 Java 语言摒弃了 C++ 语言的复杂性, 比如运算符重载、多继承、数据类型自动转换等。

这使得 Java 语言的学习非常简单, 如果读者具备一定的 C 或者 C++ 语言的基础, 在掌握了 JDK 基本使用的基础上, 在本篇的指导下将可以轻松掌握。本篇内容详尽, 即使读者没有任何 Java 基础, 也可以按步骤学习, 假以时日, 掌握 Java 语言毫无问题。

第 1 章 Java 开发基础.....	3
1.1 什么是 Java	3
1.1.1 Java 语言的产生	3
1.1.2 Java 平台	4
1.2 Java 语言的特点	4
1.3 Java 与其他面向对象语言的区别.....	7
1.3.1 Java 与 C++ 的区别	7
1.3.2 Java 与 C# 的区别.....	7
1.4 Java 语言的发展前景	8
1.5 搭建开发环境	9
1.5.1 下载和安装 JDK	9
1.5.2 获取帮助文档	11
1.5.3 设置 Path 环境变量	12
1.5.4 设置 ClassPath 环境变量.....	14
1.5.5 使用命令行工具	14
1.5.6 使用集成开发环境	15
1.6 第一个 Java 应用程序	16
1.6.1 编写源代码	16
1.6.2 使用命令行工具编译	16
1.6.3 使用命令行工具执行	16
1.7 第一个 Java 小应用程序	17
1.7.1 编写源代码	17
1.7.2 使用命令行工具编译	18
1.7.3 编辑 HTML 网页文件	18

1.7.4 运行 HTML 网页文件	18
小结	19
第 2 章 数据类型和程序流程控制	20
2.1 关键字与标识符	20
2.1.1 关键字	20
2.1.2 标识符	21
2.1.3 如何做到规范命名	22
2.2 常用基本数据类型	22
2.2.1 整型	22
2.2.2 浮点型	23
2.2.3 字符型	23
2.2.4 布尔型	23
2.3 常量与变量	24
2.3.1 常量	24
2.3.2 变量	24
2.3.3 常量与变量的区别	25
2.4 运算符	25
2.4.1 算术运算符	25
2.4.2 关系运算符	26
2.4.3 逻辑运算符	27
2.4.4 位运算符	27
2.4.5 赋值运算符	28
2.4.6 其他运算符	29
2.4.7 运算符的优先级别	29
2.5 类型转换	30
2.5.1 自动类型转换	30
2.5.2 强制类型转换	31
2.6 注释符	31
2.7 Java 程序的基本结构	32
2.7.1 类是一个基本单元	32
2.7.2 main 方法	33
2.7.3 自定义方法	34
2.7.4 System.out.println 的使用	34
2.7.5 一个简单而完整的程序	35
2.8 顺序结构	36
2.9 分支结构	37
2.9.1 if...else 分支结构	37
2.9.2 if...else 嵌套	38
2.9.3 switch 语句	39
2.9.4 编程实例	40

2.10 循环结构	42
2.10.1 while 循环	42
2.10.2 do...while 循环	43
2.10.3 for 循环	43
2.10.4 编程实例	45
2.11 跳转结构	48
2.11.1 break	48
2.11.2 continue	48
2.11.3 编程实例	49
小结	50
第 3 章 数组和字符串	51
3.1 Java 中的数组	51
3.1.1 一维数组	51
3.1.2 二维数组	53
3.1.3 多维数组	53
3.2 数组的 length 属性	53
3.3 Arrays 数组工具类的用法	54
3.4 编程实例	55
3.5 String 字符串	62
3.5.1 构造方法	63
3.5.2 String 常用方法	63
3.6 StringBuffer 字符串	68
3.6.1 构造方法	69
3.6.2 StringBuffer 常用方法	69
3.7 String 与 StringBuffer 的区别	70
3.8 equals 与 == 的区别	70
3.9 编程实例	70
小结	76

第 2 篇 面向对象程序设计

面向对象是目前最流行的程序设计方式，具备封装、多态性和继承等机制。在现实世界中，任何实体都可以看成是一个对象，对象具有属性和行为两大特征，Java 语言完全采用这种方式，同样也提供了类、接口和继承等概念。本篇不但让读者细致掌握 Java 面向对象等特性，还介绍了异常处理等 Java 语言所具备的特色技术。

第 4 章 类和对象	79
4.1 传统过程程序设计概述	79
4.2 面向对象程序设计概述	80
4.2.1 抽象	81

4.2.2	封装	81
4.2.3	继承	82
4.2.4	多态	82
4.3	类	83
4.3.1	类的定义	83
4.3.2	类的访问权限修饰符	84
4.3.3	构造方法	84
4.3.4	成员变量的定义	85
4.3.5	成员变量访问权限修饰符	87
4.3.6	成员方法的定义	87
4.3.7	方法的访问权限修饰符	89
4.4	对象	90
4.4.1	对象的概念	90
4.4.2	类与对象的区别	90
4.4.3	new 关键字	91
4.4.4	对象引用成员变量	93
4.4.5	对象引用成员方法	93
4.4.6	对象的销毁	95
4.4.7	JVM 垃圾回收	97
4.5	类中的静态成员	98
4.5.1	静态成员变量	98
4.5.2	静态成员方法	98
4.5.3	静态成员与非静态成员的区别	99
小结		100
第 5 章	继承与多态	101
5.1	继承与多态的概述	101
5.2	类的继承	102
5.2.1	类继承的实现	102
5.2.2	变量与方法的继承	103
5.2.3	子类对父类构造方法的继承和调用	104
5.2.4	关于类的继承的进一步说明	105
5.3	最终类 final	107
5.4	super 关键字	107
5.5	this 关键字	109
5.6	方法的覆盖	111
5.7	成员变量的隐藏	112
5.8	方法的重载	113
5.9	父类与子类的转换	114
小结		115

第 6 章 抽象类与接口	117
6.1 抽象类概述	117
6.2 接口的概述	117
6.3 抽象类与接口的区别	118
6.4 抽象类的定义	118
6.4.1 abstract 关键字	118
6.4.2 抽象方法的定义	119
6.5 抽象类的使用说明	119
6.6 接口的定义	120
6.6.1 关键字 interface	121
6.6.2 接口的使用规则	121
6.6.3 接口多态的应用	123
6.7 接口的使用说明	124
小结	126
第 7 章 内部类与匿名类	127
7.1 内部类概述	127
7.2 内部类的语法及应用	127
7.2.1 实例内部类	127
7.2.2 静态内部类	130
7.2.3 局部内部类	131
7.3 匿名类概述	132
7.4 匿名类的语法及应用	133
7.4.1 与类有关的匿名类	133
7.4.2 与接口有关的匿名类	135
小结	136
第 8 章 Java 常用类库	137
8.1 Object 的讲解	137
8.1.1 Object 概述	137
8.1.2 toString()方法的覆盖	138
8.1.3 equals()方法的覆盖	139
8.1.4 对象的克隆	141
8.2 Math 类和 BigInteger 类	146
8.2.1 Math 概述	146
8.2.2 random()方法	147
8.2.3 数字的格式控制	148
8.2.4 BigInteger 类	148
8.3 数据类型封装类	150
8.3.1 Integer 类	150
8.3.2 Long 类	152

8.3.3	Byte 类	152
8.3.4	Short 类	153
8.3.5	Float 类	153
8.3.6	Double 类	154
8.3.7	Boolean 类	154
8.4	包	154
8.4.1	包的概念	154
8.4.2	自动导入的包	155
8.4.3	包的命名	156
8.4.4	import 关键字	157
8.4.5	package 关键字	158
	小结	160
第 9 章	异常处理	161
9.1	异常的概念	161
9.1.1	什么是异常 (What)	161
9.1.2	为什么要有异常 (Why)	162
9.1.3	从哪里来, 往哪里去 (Where)	162
9.1.4	什么时候 (When)	162
9.1.5	哪种异常 (Which)	166
9.2	异常类	167
9.2.1	异常类的层次结构	167
9.2.2	Exception 类以及子类	168
9.2.3	Error 类以及子类	168
9.3	异常处理机制	169
9.3.1	异常的产生	169
9.3.2	异常处理	170
9.4	抛出异常	173
9.4.1	throw 语句	173
9.4.2	throws 语句	174
9.5	自定义异常	176
9.5.1	定义异常类	176
9.5.2	抛出异常	177
9.5.3	捕获并处理异常	177
	小结	178
第 10 章	集合框架	180
10.1	Java 集合概述	180
10.2	Collection 接口	180
10.2.1	Collection 概述	181
10.2.2	Collection 层次结构	181
10.2.3	常用方法	181

10.3 List 集合	182
10.3.1 ArrayList 类的使用	182
10.3.2 LinkedList 类的使用	184
10.3.3 Vector 类的使用	185
10.3.4 Stack 类的使用	186
10.4 枚举和迭代	187
10.5 Set 集合	189
10.5.1 HashSet 类的使用	190
10.5.2 TreeSet 类的使用	191
10.5.3 HashSet 与 TreeSet 的区别	193
10.6 Map 接口	193
10.6.1 Map 层次结构	194
10.6.2 使用 HashMap 类	194
10.6.3 使用 TreeMap 类	196
10.7 Collections 工具类	198
10.7.1 Collections 的常用方法	198
10.7.2 Collections 应用实例	199
小结	201
第 11 章 输入/输出操作	202
11.1 I/O 流概述	202
11.2 字节流	203
11.2.1 InputStream 类和 OutputStream 类	203
11.2.2 字节流的常用方法	204
11.2.3 文件流 FileInputStream 类	204
11.2.4 文件流 FileOutputStream 类	205
11.2.5 缓存流 BufferedInputStream 类	206
11.2.6 缓存流 BufferedOutputStream 类	208
11.2.7 基本数据流 DataInputStream 类	208
11.2.8 基本数据流 DataOutputStream 类	209
11.3 字符流	210
11.3.1 Reader 类和 Writer 类	210
11.3.2 字符流的常用方法	211
11.3.3 缓存流 BufferedReader 流	212
11.3.4 缓存流 BufferedWriter 流	213
11.3.5 转换流 InputStreamReader 类	214
11.3.6 转换流 OutputStreamWriter 类	214
11.3.7 PrintStream 类/PrintWriter 类	214
11.3.8 编程实例	216
11.4 File 文件类	217
11.4.1 File 概述	217
11.4.2 File 构造方法	217

11.4.3	File 的常用方法	218
11.4.4	编程实例	219
11.5	随机文件类 RandomAccessFile	222
11.5.1	RandomAccessFile 概述	222
11.5.2	RandomAccessFile 常用方法	223
11.6	对象序列化	225
11.6.1	对象流 ObjectInputStream 类	226
11.6.2	对象流 ObjectOutputStream 类	227
11.6.3	Serializable 接口	228
小结		230

第 3 篇 Java 开发中的高级技术

C 语言和 C++ 语言采用的是单线程体系结构，而 Java 语言很好的支持多线程。多线程的好处是可以在程序运行过程中，同时可以完成多个不同任务的同时进行，以达到更好的交互性和实时性。多线程带来的比如多个线程同步、共享变量访问等问题，Java 语言也提供了简单易学的解决办法。在本篇中，重点学习多线程和数据库等一个开发人员必须掌握的高级语言技术，为后面的项目开发打下良好的基础。

第 12 章	多线程	233
12.1	线程的概述	233
12.1.1	进程	233
12.1.2	线程	234
12.2	线程的创建方式	234
12.2.1	继承线程类	234
12.2.2	实现线程接口	236
12.3	线程的生命周期	237
12.4	线程的调度和优先级	239
12.5	线程的控制	240
12.5.1	线程的启动	240
12.5.2	线程的休眠	240
12.5.3	当前线程状态的检查	244
12.5.4	结束线程	245
12.6	线程的同步问题	247
12.6.1	使用 synchronized 同步块	247
12.6.2	使用 synchronized 同步方法	249
12.7	线程的死锁问题	251
12.7.1	造成死锁的原因	251
12.7.2	哲学家进餐的问题	252
12.7.3	解决线程死锁问题	253
12.7.4	实例一模拟解决死锁问题	253

12.8 线程间的通信	258
12.8.1 与线程间通信相关的方法	258
12.8.2 生产者—消费者问题	258
小结	261
第 13 章 反射机制	262
13.1 什么是反射	262
13.2 反射包 reflection 概述	263
13.3 Class 概述	265
13.4 获取 Class 的方式	267
13.4.1 getClass()	267
13.4.2 使用类的名字表达式	268
13.4.3 forName()	269
13.5 使用反射的好处	270
13.5.1 获取某类的成员变量	270
13.5.2 获取某类的成员方法	271
13.5.3 修改成员的值	273
小结	275
第 14 章 泛型	276
14.1 为什么使用泛型	276
14.2 定义泛型类的语法	276
14.3 泛型编程实例	279
14.4 泛型的高级语法	284
14.4.1 通配字符	284
14.4.2 限制泛型	285
小结	286
第 15 章 JDBC 程序设计	288
15.1 JDBC 概述	288
15.2 数据库的基本操作	289
15.3 JDBC 常用接口使用介绍	292
15.3.1 Driver	293
15.3.2 DriverManager	293
15.3.3 Connection	294
15.3.4 Statement	296
15.3.5 ResultSet	296
15.3.6 CallableStatement	297
15.4 Java 连接数据库	298
15.4.1 注册数据库驱动程序	299
15.4.2 获得数据库连接	300

15.4.3	发送和执行 SQL 语句	301
15.4.4	Java 连接数据库编程实例	301
15.5	执行静态 SQL	302
15.6	执行预编译 SQL	304
15.7	结果集 ResultSet	306
15.7.1	ResultSet 的常用方法	306
15.7.2	可滚动和可更新的结果集	306
15.8	存储过程	311
15.8.1	存储过程的建立	311
15.8.2	执行存储过程	312
15.9	事务的处理	314
15.9.1	什么是事务	314
15.9.2	Java 中事务的处理	314
小结		316

第 4 篇 Java Web 开发

未来是 Web 开发的世界, Java Web 开发也是 Java 领域的重点。Java 为 Internet 而生, 在 Web 开发领域, 也有非常好的积累和沉淀。本篇从 JSP、Servlet、JavaBean 和应用的角度向读者展示了 Java Web 开发中各种技术的应用, 循序渐进地引导读者快速掌握 Java Web 开发。

第 16 章	Tomcat 的使用	319
16.1	Tomcat 介绍	319
16.2	Tomcat 的工作模式	319
16.2.1	Servlet 容器	320
16.2.2	Tomcat 的工作模式	320
16.3	Tomcat 安装	321
16.3.1	安装 JDK	321
16.3.2	安装 Tomcat	321
16.3.3	Tomcat 6.0 新特性及变化	322
16.4	Tomcat 的目录结构及相关设置	322
16.4.1	预览目录结构	322
16.4.2	相关设置	323
16.5	测试运行 Tomcat	324
16.6	Tomcat 下部署 Web 程序	324
16.6.1	部署描述符 web.xml	325
16.6.2	实例演示: 创建和发布过程	335
16.7	通过 IDE 部署 Web 程序	339
小结		341

第 17 章	Servlet 技术	342
17.1	Servlet 介绍	342
17.2	Servlet 和 CGI 的比较	343
17.3	Servlet 的生命周期	343
17.3.1	加载和实例化 Servlet	344
17.3.2	初始化	344
17.3.3	处理请求	346
17.3.4	服务结束	347
17.4	Servlet API	347
17.4.1	javax.servlet 包	348
17.4.2	javax.servlet.http 包	362
17.5	会话跟踪	372
17.5.1	会话状态跟踪 API	373
17.5.2	会话状态跟踪 API 示例	375
17.6	Cookie 概述	377
17.6.1	Cookie 操作	377
17.6.2	使用 Cookie 的 Servlet	378
	小结	381
第 18 章	JSP 技术	382
18.1	JSP 技术概述	382
18.1.1	动态网页技术比较	382
18.1.2	JSP 工作原理	383
18.2	JSP 核心语法	384
18.2.1	JSP 页面的基本结构	384
18.2.2	变量的声明	386
18.2.3	表达式	386
18.2.4	JSP 脚本	387
18.2.5	JSP 的注释	388
18.2.6	JSP 的指令标签	389
18.2.7	JSP 的动作标签	393
18.3	JSP 的内置对象	400
18.3.1	request 对象	400
18.3.2	response 对象	402
18.3.3	session 对象	404
18.3.4	aplication 对象	406
18.3.5	out 对象	407
18.3.6	Cookie 对象	408
18.3.7	page 对象	410
18.3.8	exception 对象	410