

Java

面向对象编程



孙卫琴

编著

飞思科技产品研发中心 监制

Helping readers to master OOP(Object-Oriented Programming)
with Java and get deep understanding of how the language works

全书贯穿六条主线:

面向对象编程思想

Java语言的语法

Java虚拟机执行Java程序的原理

在实际项目中的运用

设计模式

性能优化技巧



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

Java 开发专家

Java

面向对象编程

孙卫琴 编著
飞思科技产品研发中心 监制



电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书内容由浅入深,紧密结合实际,利用大量典型实例,详细讲解 Java 面向对象的编程思想、编程语法和设计模式,介绍常见 Java 类库的用法,总结优化 Java 编程的各种宝贵经验,深入阐述 Java 虚拟机执行 Java 程序的原理。书中实例均基于最新的 JDK 1.5 版本。全书内容包括面向对象的编程思想、Java 语言的基础知识、异常处理、类与对象的生命周期、多线程、Java 集合、输入/输出和 GUI 编程等。其最大特色是以六条主线贯穿全书:面向对象编程思想、Java 语言的语法、Java 虚拟机执行 Java 程序的原理、在实际项目中的运用、设计模式和性能优化技巧。同时,本书还贯穿了 Sun 公司的 SCJP (Sun Certified Java Programmer) 认证的考试要点。

书中实例源文件请到 <http://www.fecit.com.cn> 的“下载专区”下载。

本书适用于所有 Java 编程人员,包括 Java 初学者及资深 Java 开发人员;亦可作为高校的 Java 教材,企业 Java 的培训教材,以及 Sun 公司的 SCJP 认证的辅导材料。

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。

图书在版编目 (CIP) 数据

Java 面向对象编程 / 孙卫琴编著. —北京: 电子工业出版社, 2006.7

(Java 开发专家)

ISBN 7-121-02538-8

I. J... II. 孙... III. Java 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2006) 第 040606 号

责任编辑: 郭 晶 孙伟娟

印 刷: 北京市天宇星印刷厂

出版发行: 电子工业出版社

北京海淀区万寿路 173 信箱 邮编: 100036

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 40.25 字数: 1030.4 千字

印 次: 2006 年 7 月第 1 次印刷

印 数: 8 000 册 定价: 65.80 元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系电话: 010-68279077。质量投诉请发邮件至 zlt@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

作者介绍:



孙卫琴 1975 年出生，籍贯江苏常州，1997 年毕业于上海交通大学，随后主要从事基于 Java 的软件开发工作，在此期间还从事过 Java 方面的培训和咨询工作。2002 年开始 Java 技术领域的图书创作。

代表著作有:

《Java 面向对象编程》

2006

《精通 Hibernate:Java 对象持久化技术详解》

2005

5 次印刷

《精通 Struts: 基于 MVC 的 Java Web 设计与开发》

2004

11 次印刷

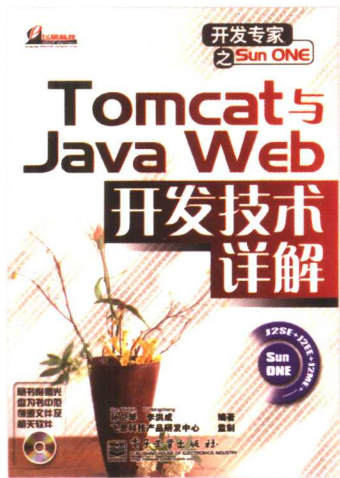
《Tomcat 与 Java Web 开发技术详解》

2004

9 次印刷

个人网站: www.javathinker.org

E-mail: linda_j2ee@yahoo.com.cn



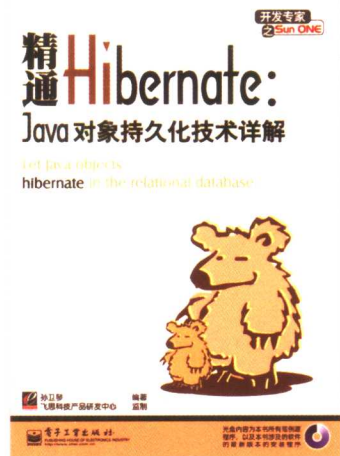
ISBN 7-5053-9392-8

定价：45.00元



ISBN 7-121-00052-0

定价：49.00元



ISBN 7-121-01136-0

定价：59.00元

热烈祝贺“开发专家之 Sun ONE”系列被
评为电子工业出版社“最佳品牌奖”

“开发专家之 Sun ONE”全新提升为“Java 开发专家”系列
——源自精品 成就理想

出版说明

★ 从“开发专家之 Sun ONE”到“Java 开发专家”

“开发专家之 Sun ONE”系列丛书从诞生之日至今，已经四岁了。在这个系列里面，我们一直努力体现着这么一个理念：用一种较为敏锐的视角来跟踪 IT 技术的发展轨迹，并把可能为广大程序员所希望获得的知识，用图书出版的方式奉献给大家。

在这个系列中，我们陆续出版了约 30 种图书，有《Java 与模式》、《JSP 应用开发详解（第二版）》、《精通 EJB（第三版）》、《Tomcat 与 Java Web 应用开发详解》、《精通 Struts：基于 MVC 的 Java Web 设计与开发》、《JBoss 管理与开发核心技术（第三版）》、《精通 Spring》、《精通 Hibernate：Java 对象持久化技术详解》等一大批读者朋友耳熟能详的作品。很多作品都是在国内没有同类图书的情况下出版的。在这几年的出版工作中，我们时刻感受着市场的风险，也时刻收获着无数读者给我们的认可。

在这个系列中，凝聚了大量资深技术专家的心血。有大家都熟知的阎宏、刘晓华、孙卫琴、罗时飞等，还有一些正在不断腾跃的开发高手。这些非常优秀的国内原创者们一直都在支持着“开发专家之 Sun ONE”系列的出版工作，在这里，我们要向他们说声：谢谢。

桃李不言，下自成蹊。由于这些年“开发专家之 Sun ONE”在“两个效益”中的杰出表现，电子工业出版社授予这个系列“最佳品牌奖”。

时代不断前进，技术不断变革。为了顺应 Java 领域的技术发展态势，为了赋予这个经典的图书系列更强的生命力，我们将“开发专家之 Sun ONE”升级为“Java 开发专家”。我们将继承原有的出版理念，紧密跟踪技术热点和发展趋势，会聚更多优秀作者，全力奉献更经典的作品。

★ 规划你的 Java 开发之路

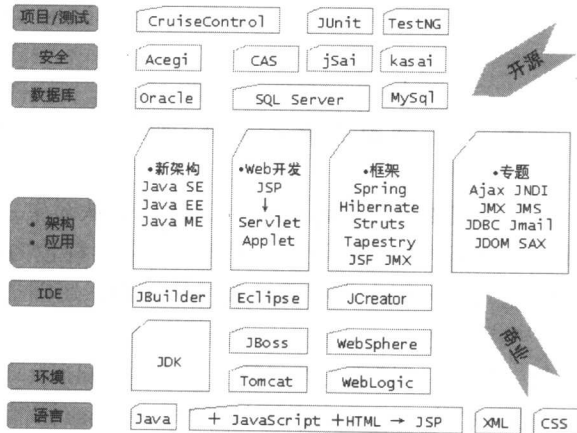
喜马拉雅山脉的最高峰不断地在温室效应中降低，而 Java 世界的颠峰永远都在技术人员的追求中不断升高。每个人都有不同的路，每个人都有不同的行路方式，不过，往往“到了山顶才发现，错误的路和正确的路就差那么几步！”

身处 Java 洪流中的程序员最累（不过大家都说 Java 程序员薪水最高，呵呵），我们简单整理了一下 Java 领域的相关技术、工具、架构，如下图所示。这个框图中的每一个英文单词（或缩写）都可以写成一本书。Java 领域还有一个特点，那就是商业产品和开源产品层出不穷，潮流不断。相比于其他领域，如.NET，Java 开发更是体现了这句谚语：条条大路通罗马。

罗马只有一个，大路却有多条。看上去，似乎到罗马很容易，反正路多嘛。不过，路多却容易迷失方向。当你在 Java 领域中摸爬滚打几年后，发现自己在无数条道路上走了很久，却不知道罗马何日才能到达，甚至连罗马的方向都不知道，这时你肯定会很失落。

很遗憾，在这个简短的出版说明文章里面，我们无法告诉你每一条连贯的、不费周折的通往罗马的道路该如何走。或许，通过“Java 开发专家”系列中的某本书，你可以找到属于你的正确道路。在一般情况下，我们不会就某一项很窄的话题来单独写一本书，我们还是希望通过我们的一些专业和智慧，尽力把一些相关技术整合起来，用较为简明的方式表达出来，最后由你来选择。

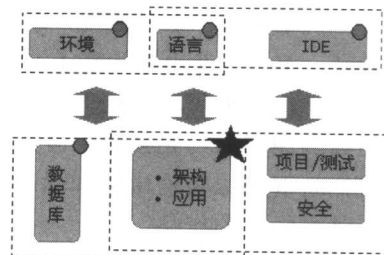
这里有句话与大家共勉：少走弯路，就是捷径！



★ “Java 开发专家”的奉献

犹如在上面那个框图中展现的那样，我们希望在各个层面、各个方向上都能给读者奉献出优秀的图书作品，全面体现技术与应用的结合。从宏观上看，我们会从语言、IDE、环境、数据库、架构与应用、安全、项目与测试等方面进行选择，选出一些读者迫切需要的技术来先行规划。

“Java 开发专家”虽然新蓓初绽，但因其源自盛放的“开发专家之 Sun ONE”系列而根基稳健。两个系列会有一段很长的并行时间，我们会用一种优化的方式来保证读者的顺利选择。无论哪一个系列，必定都有大家喜欢的图书。



在技术上，有着持久化的方法，在学习上，也需要有持久化的精神。

从“开发专家之 Sun ONE”到“Java 开发专家”，希望可以带给你持久化的动力。

联系方式

咨询电话：(010) 68134545 88254160

电子邮件：support@fecit.com.cn

服务网址：<http://www.fecit.com.cn> <http://www.fecit.net>

通用网址：计算机图书、飞思、飞思教育、飞思科技、FECIT

Java 自 1996 年正式发布以来，经历了出生、成长和壮大的阶段，现在已经成为 IT 领域里的主流编程语言。Java 起源于 Sun 公司的一个叫“Green”的项目，目的是开发嵌入家用电器的分布式软件系统，使电器更加智能化。图 P-1 所示为参与 Green 项目的开发人员。Green 项目一开始准备采用 C++ 语言，但是考虑到 C++ 语言太复杂，而且安全性差，于是决定基于 C++ 语言开发一种新的 Oak 语言（即 Java 的前身）。



Java之父: James Gosling

图 P-1 参与 Green 项目的开发人员

Oak 是一种适用于网络编程的精巧而安全的语言，它保留了许多 C++ 语言的语法，但去除了明确的资源引用、指针算法与操作符重载等潜在的危险特性。并且 Oak 语言具有与硬件无关的特性，制造商只需要更改芯片，就可以将烤面包机上的程序代码移植到微波炉上或其他电器上，而不必改变软件，这就大大降低了开发成本。

当 Oak 语言成熟时，全球 Internet 也在迅速发展。Sun 公司的开发小组认识到 Oak 非常适合于 Internet 编程。1994 年，他们完成了一个用 Oak 语言编写的早期的 Web 浏览器，称为 WebRunner，后改名为 HotJava，展示了 Oak 作为 Internet 开发工具的能力。

1995 年，Oak 语言更名为 Java 语言（以下简称为 Java）。Java 的取名有一个趣闻。据说，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（爪哇）咖啡。有一个人灵机一动说，就叫 Java 怎样，这得到了其他人的赞赏。于是，Java 这个名字就这样传开了。

1996 年，Sun 公司发布 JDK 1.0，计算机产业的各大公司（包括 IBM、Apple、DEC、Adobe、Silicon Graphics、HP、Oracle、Toshiba 和 Microsoft 等）相继从 Sun 公司购买了 Java 技术许可证，开发相应的产品。

1998 年，Sun 公司发布了 JDK 1.2（从这个版本开始的 Java 技术都称为 Java 2）。Java 2 不仅兼容于智能卡和小型消费类设备，还兼容于大型服务器系统，它使软件开发商、服务提供商和设备制造商更加容易抢占市场机遇。这一开发工具极大地简化了编程人员编制企业级 Web 应用的工作，把一次编程到处使用的诺言应用到服务器领域。

1999 年，Sun 公司把 Java 2 技术分成 J2SE、J2EE 和 J2ME。其中 J2SE 就是指从 1.2 版本开始的 JDK，它为创建和运行 Java 程序提供了最基本的环境。J2EE 和 J2ME 建立在 J2SE 的基础上，J2EE 为分布式的企业应用提供开发和运行环境，J2ME 为嵌入式应用（比如运行在手机里的 Java 程序）提供开发和运行环境。

在进入 21 世纪以来，随着 Web 技术成为展示和操作数据的事实标准，企业利用 J2EE 平台对原来分散的子系统进行整合。尽管应用整合可以通过多种手段来实现，但 J2EE 在出现后，因其天生具备良好的开放性和可扩展性，使之在应用整合和开发的过程中发挥了愈来愈显著的优势。J2EE 逐渐成为开发商创建电子商务应用的事实标准。

Java 的公用规范（Publicly Available Specification, PAS）在 1997 年被国际标准化组织（ISO）认定，这是 ISO 第一次破例接受一个具有商业色彩的公司作为公用规范 PAS 的提交者。

总之，面向对象的 Java 语言具备一次编程、任何地方均可运行的能力，这使其成为服务提供商和系统集成商用以支持多种操作系统和硬件平台的首选解决方案。Java 作为软件开发的一种革命性的技术，其地位已被确定。如今，Java 技术已被列为当今世界信息技术的主流之一。表 P-1 对 Java 的发展历史做了总结。

表 P-1 Java 的发展历史

年 份	Java 的发展历史
1995	Java 语言诞生
1996	JDK 1.0 发布，10 个最主要的操作系统供应商申明将在其产品中支持 Java 技术
1997	JDK 1.1 发布
1998	JDK 1.1 下载量超过 200 万次，JDK 1.2（称 Java 2）发布，JFC/Swing 技术发布，JFC/Swing 被下载了 50 多万次
1999	Java 被分成 J2SE、J2EE 和 J2ME，JSP/Servlet 技术诞生
2000	JDK 1.3 发布，JDK 1.4 发布
2001	Nokia 公司宣布到 2003 年将出售 1 亿部支持 Java 的手机，J2EE 1.3 发布
2002	J2SE 1.4 发布，至此 Java 的计算能力有了大幅度提升。J2EE SDK 的下载量达到 200 万次
2003	5.5 亿台桌面上运行 Java 程序，75% 的开发人员将 Java 作为首要开发工具
2004	J2SE 1.5 发布，这是 Java 语言在发展史上的又一里程碑事件。为了表示这个版本的重要性，J2SE 1.5 更名为 J2SE 5.0
2005	JavaOne 大会召开，Sun 公司公开 Java SE6。此时，Java 的各种版本被更名，取消其中的数字“2”：J2EE 更名为 Java EE，J2SE 更名为 Java SE，J2ME 更名为 Java ME

Java 语言的特点

Java 应用如此广泛是因为 Java 具有多方面的优势。其特点如下：

(1) 面向对象。Java 自诞生之时就设计成面向对象的语言，而 C++ 语言是一种强制面向对象的语言。面向对象可以说是 Java 最重要的特性，它不支持类似 C 语言那样的面向过程的程序设计技术。Java 支持静态和动态风格的代码重用。

(2) 跨平台。对于 Java 程序，不管是 Windows 平台还是 UNIX 平台，或者其他平台，它都适用。Java 编辑器把 Java 源程序编译成与体系结构无关的字节码指令，只要安装了 Java 运行系统，Java 程序就可在任意的处理器上运行。这些字节码指令由 Java 虚拟机来执行，Java 虚拟机的解释器在得到字节码后，对它进行转换，使之能够在不同的平台运行。

(3) 直接支持分布式的网络应用。除了支持基本的语言功能以外, Java 核心类库还包括一个支持 HTTP、SMTP 和 FTP 等基于 TCP/IP 协议的类库。因此, Java 应用程序可凭借 URL 打开并访问网络上的对象, 其访问方式与访问本地文件系统几乎完全相同。在 Java 出现以前, 为分布式环境尤其是 Internet 提供动态的内容无疑是一项非常宏伟、难以想像的任务, 但 Java 的语言特性却使我们很容易地达到这个目标。

(4) 安全性和健壮性。Java 致力于检查程序在编译和运行时的错误, 类型检查能够帮助检查出许多开发早期出现的错误。Java 支持自动内存管理, 这不但让程序员减轻了许多负担, 也让程序员减少了犯错的机会。Java 自己操纵内存减小了内存出错的可能性。Java 还能够检测数组边界, 避免了覆盖数据的可能。在 Java 语言里, 指针和释放内存等功能均被抛弃, 从而避免了非法内存操作的危险。

以上特点, 是 C++ 语言及其他语言无可比拟的 (C++ 语言尽管也是面向对象的, 但并不是严格意义上的面向对象语言)。单从面向对象的特性来看, Java 类似于 SmallTalk, 但其他特性, 尤其是适用于分布式计算环境的特性远远超越了 SmallTalk。Java 发展到现在, 已经不仅仅是一种语言, 可以说是一种技术, 这个技术涉及到网络和编程等领域。另外, Java 是非常简单、高效的, 有调查数据发现: 用 C++ 和 Java 来做一个相同功能的项目, 用 Java 写的程序要比用 C++ 写的程序节省 60% 的代码和 66% 的时间。可以说, 用 Java 语言编程时间短、功能强, 编程人员接手起来更容易、更简便。

本书的组织结构和主要内容

本书以六条主线贯穿全书: 面向对象编程思想、Java 语言的语法、Java 虚拟机执行 Java 程序的原理、在实际项目中的运用、设计模式和性能优化技巧。书中的每一章都会围绕若干条主线来展开内容, 并且根据全书的布局, 合理安排每一章内容的深度。本书主要内容包括: 面向对象的开发方法概述、数据类型和变量、操作符、流程控制、继承、Java 语言中的修饰符、接口、异常处理、类的生命周期、对象的生命周期、内部类、多线程与并发、数组、Java 集合、输入输出、图形用户界面、Swing 组件和 Java 常用类。

这本书是否适合你

在如今的 Java 领域, 各种新技术、新工具层出不穷, 一方面, 每一种技术都会不停地升级换代; 另一方面, 还会不断涌现出新的技术和工具。Java 世界就像小时候玩的万花筒, 尽管实质上只是由几个普通的玻璃碎片组成, 但只要轻轻一摇, 就会变换出千万种缤纷的图案。Java 世界如此变化多端, 很容易让初学 Java 的人有无从下手的感觉。常常会有读者问我这样的问题:

我学了 Java 已经一年多了, 现在就只能用 JSP 写点东西, 其他的東西实在太多了, 我整天学都学不完, 很迷惘, 不知道该如何有针对性地去学, 去找到一份 Java 工作, 现在是困死在 Java 里了。

撰写本书, 目的是为了帮助读者看清 Java 万花筒的本质, 从复杂的表象中寻找普遍的规律, 深刻理解 Java 的核心思想。只有掌握了普遍的规律与核心思想, 才能以不变应万

变，轻轻松松地把握 Java 技术发展的新趋势，迅速地领略并且能熟练运用一门新的技术，而不成为被动的追随者，知其然而不知其所以然。

阅读本书，读者对 Java 的领悟将逐步达到以下境界：

- 熟悉 Java 语法，熟练地编译和调试程序。
- 按照面向对象的思想来快速理解 JDK 类库及其他第三方提供的类库，通过阅读 JavaDoc 和相关文档，知道如何正确地使用这些类库。
- 按照面向对象的思想来分析问题领域，设计对象模型。
- 在开发过程中会运用现有的一些优秀设计模式，提高开发效率。
- 当一个方法有多种实现方式时，能够从可维护、可重用及性能优化的角度选择最佳的实现方式。
- 理解 Java 虚拟机执行 Java 程序的原理，从而更深入地理解 Java 语言的各种特性和语法规则。

致谢

本书在编写过程中得到了 Sun 公司在技术上的大力支持，飞思科技产品研发中心负责监制工作，复旦软件学院的 MSE 师生们为本书的编写提供了有益的帮助，JavaThinker 网站的许多热心网友及作者的亲友，包括李洪成、李红军、李建伟、杨源、曹文伟、gogolong、lwy0329 和 fjchenq 等为本书提供了宝贵的建议，在此表示衷心的感谢！

尽管我们尽了最大努力，但本书难免会有不妥之处，欢迎各界专家和读者朋友批评指正。



飞思图书推荐

精通 Hibernate: Java 对象持久化技术详解 (含光盘)

特色:

- ★ 由浅入深详尽阐述 Java 开源技术
- ★ 知名 IT 作家孙卫琴最新作品, 上市 3 个月销量突破 10000 册。
- ★ 2005 年 IT 图书市场亮点, 大受读者好评

Hibernate 是非常流行的对象-关系映射工具。本书详细介绍了运用目前最成熟的 Hibernate 2.1 版本进行 Java 对象持久化的技术。Hibernate 是连接 Java 对象模型和关系数据模型的桥梁, 通过本书, 读者不仅能掌握用 Hibernate 工具对这两种模型进行映射的技术, 还能获得设计与开发 Java 对象模型和关系数据模型的先进经验。书中内容注重理论与实践相结合, 列举了大量具有典型性和实用价值的 Hibernate 应用实例, 并提供了详细的开发和部署步骤。随书附赠光盘内容为本书所有范例源程序, 以及本书涉及的软件的最新版本的安装程序。



书号: TP011360
定价: 59.00



书号: TP000520
定价: 49.00

精通 Struts: 基于 MVC 的 Java Web 设计与开发 (含光盘)

最畅销 Struts 图书 销量已过 35000 册

特色:

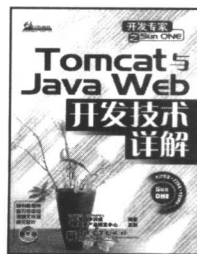
- ★ 面向 Java Web 开发者, 透彻分析 Struts 框架
- ★ 基于当红开源软件 Struts 最新版本
- ★ 结合大量典型实用的 Java Web 实例
- ★ 孙卫琴又一力作, 《Tomcat 与 Java Web 开发技术详解》姐妹篇

Struts 是目前非常流行的基于 MVC 的 Java Web 框架。详细介绍了在最新的 Struts1.1 及 Struts1.2 版本上设计和开发 Java Web 应用的各种技术。内容注重理论与实践相结合, 列举了大量典型 Web 应用实例, 提供了详细的开发和部署步骤。光盘内容包含所有范例源程序, 及涉及软件最新版本的安装程序。

Tomcat 与 Java Web 开发技术详解 (含光盘)

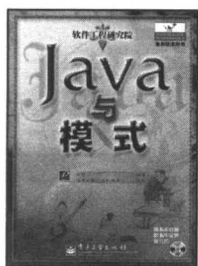
特色: 本书为孙卫琴开山之作, 上市之初就获好评如潮, 现销量已超过 25000 册。

本书详细介绍了在最新 Tomcat 5 版本上开发 Java Web 应用的各种技术。主要内容包括: Tomcat 和 Java Web 开发的基础知识, Java Web 开发的高级技术, Tomcat 与当前其他通用软件的集成, 以及 Tomcat 的各种高级功能。是精通 Tomcat 技术和开发 Java Web 应用的必备的实用手册。随书附赠光盘内容为本书所有范例源程序, 以及本书涉及到的所有软件的最新版本的安装程序。



书号: TP93920
定价: 45.00

Java 与模式 (含光盘)



书号: TP80000
定价: 88.00

一本被誉为深入研究软件工程的必读圣经

Java 大师阎宏倾情打造

讲解设计原则及设计模式的实用宝典, 目的是要给工作繁忙的 Java 系统设计师一个快速而准确的设计原则和设计模式的辅导。每个章节讲解一个编程模式, 并说明其用意、结构、适用情况等, 附带丰富实例、练习题, 方便研习, 帮助读者更好地理解。

JSP 应用开发详解 (第二版) (含光盘)

最畅销 JSP 图书, 销量突破 35000 册



书号: TP94190
定价: 55.00

J2EE 技术无疑成为当今电子商务的最佳解决方案, 在最近两年, JSP 技术又有了巨大的发展和变化。在最新发布的 J2EE 1.4 框架中, 对 JSP 和 Servlet 技术进行了升级。在新的 JSP 2.0 规范中, 为 JSP 的易用性进行了重新的设计, 使得不懂 Java 的页面设计人员也能够开发 JSP 应用。在讨论标签语言、表达式语言和 Struts 技术方面, 本书具有独到之处。



书号: TP01051
定价: 39.00

精通 Spring

本书深入剖析了当前流行的轻量级开发框架 Spring 技术。全书共分成 3 部分。第一部分, 重点阐述 Spring 的架构。这部分内容采用循序渐进的方式, 带领开发者进入 Spring 中, 主要阐述了 Spring IoC 和 Spring AOP。第二部分, 重点阐述 Spring 的使用。这部分内容从简化 Java/J2EE 的角度出发, 从 J2EE 平台各个技术层面分析并给出大量的研究实例, 对 Spring 提供的 API 进行阐述。主要阐述了 Spring 对 J2EE API 提供的服务抽象。第三部分, 重点阐述 Spring 高级专题。这部分内容重点对视图技术进行了研究, 因为对于开发 Web 应用而言, 前端界面的开发往往工作量很大。因此, 使用合理的视图技术开发 Web 应用对于项目的成功与否很关键。另外, Web 应用的安全性往往也是企业应用中最为重要的需求之一, 而用于 Spring 的 Acegi 安全框架很好地解决了这个问题, 这也是第三部分重点研究的内容之一。本书适合所有的 Java、J2EE 开发者阅读。

第 1 章 面向对象开发方法概述	1
1.1 结构化的软件开发	
方法简介	3
1.2 面向对象的软件开发	
方法简介	5
1.2.1 对象模型	6
1.2.2 UML: 可视化	
建模语言	6
1.2.3 Rational Rose:	
可视化建模工具	7
1.3 面向对象开发中的核心	
思想和概念	7
1.3.1 问题领域、对象、属性、状	
态、行为、方法、实现	7
1.3.2 类、类型	9
1.3.3 消息、服务	10
1.3.4 接口	11
1.3.5 封装、透明	12
1.3.6 抽象	16
1.3.7 继承、扩展、覆盖	17
1.3.8 组合	19
1.3.9 多态、动态绑定	22
1.4 UML 语言简介	24
1.4.1 用例图	24
1.4.2 类框图	25
1.4.3 时序图	26
1.4.4 协作图	27
1.4.5 状态转换图	27
1.4.6 组件图	28
1.4.7 部署图	29
1.5 类之间的关系	29
1.5.1 关联 (Association)	29
1.5.2 依赖 (Dependency)	31
1.5.3 聚集 (Aggregation)	31
1.5.4 泛化 (Generalization)	32
1.5.5 实现 (Realization)	32
1.5.6 区分依赖、关联和聚集关系	33
1.6 实现 Panel 系统	35

1.6.1 扩展 Panel 系统	37
1.6.2 用配置文件进一步提高 Panel	
系统的可维护性	39
1.6.3 运行 Panel 系统	40
1.7 小结	41
1.8 思考题	42
第 2 章 第一个 Java 应用	43
2.1 创建 Java 源文件	43
2.1.1 Java 源文件结构	44
2.1.2 包声明语句	45
2.1.3 包引入语句	47
2.1.4 方法的声明	49
2.1.5 程序入口 main()方法的声明	50
2.1.6 给 main()方法传递参数	51
2.1.7 注释语句	51
2.1.8 关键字	51
2.1.9 标识符	52
2.1.10 编程规范	52
2.2 用 JDK 管理 Java 应用	53
2.2.1 JDK 简介	54
2.2.2 编译 Java 源文件	55
2.2.3 运行 Java 程序	56
2.2.4 给 Java 应用打包	59
2.3 JavaDoc 文档	60
2.3.1 JavaDoc 标记	62
2.3.2 javadoc 命令的用法	67
2.4 Java 虚拟机的运行时数据区	69
2.5 小结	71
2.6 思考题	71
第 3 章 数据类型和变量	73
3.1 基本数据类型	74
3.1.1 boolean 类型	74
3.1.2 byte、short、int 和	
long 类型	75
3.1.3 char 类型与字符编码	76
3.1.4 float 和 double 类型	79
3.2 引用类型	83

3.2.1	基本类型与引用类型的区别	84	4.10	小结	131
3.2.2	用 new 关键字创建对象	85	4.11	思考题	134
3.3	变量的作用域	87	第 5 章	流程控制	137
3.3.1	静态变量和实例变量的生命周期	88	5.1	分支语句	138
3.3.2	局部变量的生命周期	91	5.1.1	if...else 语句	138
3.3.3	成员变量和局部变量同名	92	5.1.2	switch 语句	141
3.3.4	将局部变量的作用域最小化	93	5.2	循环语句	144
3.4	对象的默认引用: this	94	5.2.1	while 语句	145
3.5	参数传递	96	5.2.2	do...while 语句	146
3.6	变量的初始化及默认值	98	5.2.3	for 语句	147
3.6.1	成员变量的初始化	98	5.2.4	多重循环	150
3.6.2	局部变量的初始化	99	5.3	流程跳转语句	151
3.7	直接数	100	5.4	综合例子: 八皇后问题	153
3.7.1	直接数的类型	100	5.5	小结	156
3.7.2	直接数的赋值	101	5.6	思考题	157
3.8	小结	103	第 6 章	继承	161
3.9	思考题	104	6.1	继承的基本语法	161
第 4 章	操作符	107	6.2	方法重载 (Overload)	163
4.1	操作符简介	107	6.3	方法覆盖 (Override)	165
4.2	整型操作符	108	6.4	方法覆盖与方法重载的异同	170
4.2.1	一元整型操作符	109	6.5	super 关键字	171
4.2.2	二元整型操作符	110	6.6	多态	172
4.3	浮点型操作符	115	6.7	继承的利弊和使用原则	177
4.4	比较操作符和逻辑操作符	116	6.7.1	继承树的层次不可太多	177
4.5	三元操作符 “?:”	119	6.7.2	继承树的上层为抽象层	177
4.6	字符串连接操作符 “+”	119	6.7.3	继承关系最大的弱点: 打破封装	178
4.7	操作符 “==” 与对象的 equals() 方法	121	6.7.4	精心设计专门用于被继承的类	180
4.7.1	操作符 “==”	121	6.7.5	区分对象的属性与继承	181
4.7.2	对象的 equals() 方法	122	6.8	比较组合与继承	183
4.8	instanceof 操作符	125	6.8.1	组合关系的分解过程对应继承关系的抽象过程	184
4.9	变量的赋值和类型转换	127	6.8.2	组合关系的组合过程对应继承关系的扩展过程	186
4.9.1	基本数据类型转换	128	6.9	小结	189
4.9.2	引用类型的类型转换	131			

6.10 思考题	190	9.2.3 throws 子句: 声明可能会出现的异常	256
第 7 章 Java 语言中的修饰符	195	9.2.4 throw 语句: 抛出异常	256
7.1 访问控制修饰符	196	9.2.5 异常处理语句的语法规则	257
7.2 abstract 修饰符	198	9.2.6 异常流程的运行过程	260
7.3 final 修饰符	200	9.3 Java 异常类	264
7.3.1 final 类	201	9.3.1 运行时异常	266
7.3.2 final 方法	201	9.3.2 受检查异常	266
7.3.3 final 变量	202	9.3.3 区分运行时异常和受检查异常	267
7.4 static 修饰符	206	9.3.4 区分运行时异常和错误	268
7.4.1 static 变量	206	9.4 用户定义异常	268
7.4.2 static 方法	209	9.4.1 异常转译和异常链	269
7.4.3 static 代码块	212	9.4.2 处理多样化异常	271
7.5 小结	214	9.5 异常处理原则	273
7.6 思考题	215	9.5.1 异常只能用于非正常情况	273
第 8 章 接口	219	9.5.2 为异常提供说明文档	274
8.1 接口的概念和基本特征	220	9.5.3 尽可能地避免异常	275
8.2 比较抽象类与接口	222	9.5.4 保持异常的原子性	276
8.3 与接口相关的设计模式	227	9.5.5 避免过于庞大的 try 代码块	277
8.3.1 定制服务模式	227	9.5.6 在 catch 子句中指定具体的异常类型	277
8.3.2 适配器模式	231	9.5.7 不要在 catch 代码块中忽略被捕获的异常	278
8.3.3 默认适配器模式	235	9.6 小结	278
8.3.4 代理模式	236	9.7 思考题	279
8.3.5 标识类型模式	241	第 10 章 类的生命周期	283
8.3.6 常量接口模式	242	10.1 Java 虚拟机及程序的生命周期	283
8.4 小结	243	10.2 类的加载、连接和初始化	283
8.5 思考题	244	10.2.1 类的加载	284
第 9 章 异常处理	247	10.2.2 类的验证	285
9.1 Java 异常处理机制概述	248	10.2.3 类的准备	285
9.1.1 Java 异常处理机制的优点	248	10.2.4 类的解析	286
9.1.2 Java 虚拟机的方法调用栈	250	10.2.5 类的初始化	286
9.1.3 异常处理对性能的影响	253	10.2.6 类的初始化的时机	288
9.2 运用 Java 异常处理机制	253	10.3 类加载器	291
9.2.1 try...catch 语句: 捕获异常	253		
9.2.2 finally 语句: 任何情况下都必须执行的代码	254		

10.3.1	类加载的父亲委托 (Parent Delegation) 机制	293
10.3.2	创建用户自定义的类加载器	294
10.3.3	URLClassLoader 类	300
10.4	类的卸载	301
10.5	小结	302
10.6	思考题	303
第 11 章	对象的生命周期	305
11.1	创建对象的方式	305
11.2	构造方法	308
11.2.1	重载构造方法	309
11.2.2	默认构造方法	310
11.2.3	子类调用父类的构造方法	311
11.2.4	构造方法的作用域	314
11.2.5	构造方法的访问级别	315
11.3	静态工厂方法	316
11.3.1	单例 (singleton) 类	317
11.3.2	枚举类	319
11.3.3	不可变 (immutable) 类与可变类	321
11.3.4	具有实例缓存的不可变类	325
11.3.5	松耦合的系统接口	327
11.4	垃圾回收	328
11.4.1	对象的可触及性	329
11.4.2	垃圾回收的时间	330
11.4.3	对象的 finalize() 方法简介	331
11.4.4	对象的 finalize() 方法的特点	331
11.4.5	比较 finalize() 方法和 finally 代码块	334
11.5	清除过期的对象引用	334
11.6	对象的强、软、弱和虚引用	336
11.7	小结	342
11.8	思考题	342

第 12 章	内部类	345
12.1	内部类的基本语法	345
12.1.1	实例内部类	346
12.1.2	静态内部类	349
12.1.3	局部内部类	351
12.2	内部类的继承	352
12.3	子类与父类中的内部类同名	353
12.4	匿名类	354
12.5	内部接口及接口中的内部类	357
12.6	内部类的用途	358
12.6.1	封装类型	358
12.6.2	直接访问外部类的成员	358
12.6.3	回调 (CallBack)	359
12.7	内部类的类文件	361
12.8	小结	361
12.9	思考题	362
第 13 章	多线程	365
13.1	Java 线程的运行机制	365
13.2	线程的创建和启动	367
13.2.1	扩展 java.lang.Thread 类	367
13.2.2	实现 Runnable 接口	372
13.3	线程的状态转换	373
13.3.1	新建状态 (New)	373
13.3.2	就绪状态 (Runnable)	374
13.3.3	运行状态 (Running)	374
13.3.4	阻塞状态 (Blocked)	374
13.3.5	死亡状态 (Dead)	375
13.4	线程调度	376
13.4.1	调整各个线程的优先级	377
13.4.2	线程睡眠: Thread.sleep() 方法	379
13.4.3	线程让步: Thread.yield() 方法	380
13.4.4	等待其他线程结束: join()	381