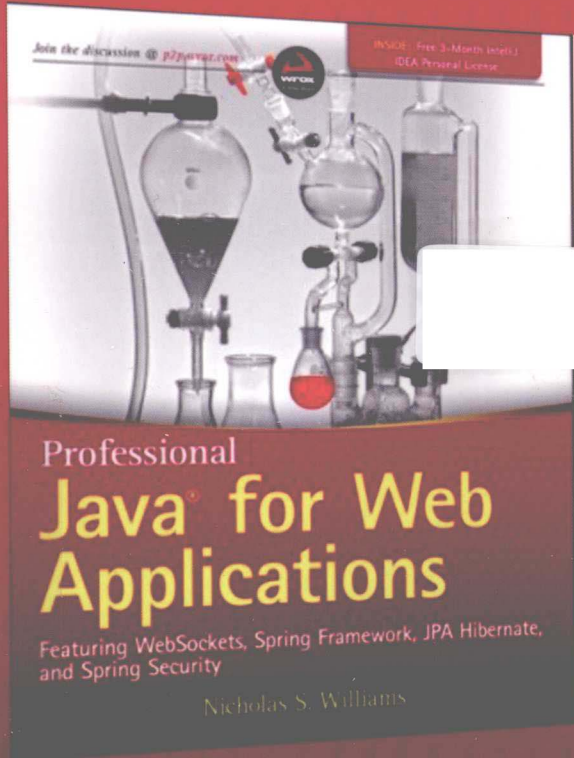




Java Web高级编程

——涵盖WebSockets、Spring Framework、JPA Hibernate和Spring Security



[美] Nicholas S. Williams
王肖峰

著
译

Java Web 高级编程

——涵盖 WebSockets、Spring Framework、
JPA Hibernate 和 Spring Security

[美] Nicholas S. Williams

王肖峰



清华大学出版社

北 京

Nicholas S. Williams

Professional Java for Web Applications

EISBN: 978-1-118-65646-4

Copyright © 2014 by John Wiley & Sons, Inc., Indianapolis, Indiana

All Rights Reserved. This translation published under license.

Trademarks: Wiley, Wrox, the Wrox logo, Programmer to Programmer, and related trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries, and may not be used without written permission. Java is a registered trademark of Oracle America, Inc. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc., is not associated with any product or vendor mentioned in this book.

本书中文简体字版由 Wiley Publishing, Inc. 授权清华大学出版社出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

北京市版权局著作权合同登记号 图字：01-2014-3653

Copies of this book sold without a Wiley sticker on the cover are unauthorized and illegal.

本书封面贴有 Wiley 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 Web 高级编程：涵盖 WebSockets、Spring Framework、JPA Hibernate 和 Spring Security / (美) 威廉斯 (Williams, N.S.) 著；王肖峰 译。—北京：清华大学出版社，2015

书名原文：Professional Java for Web Applications

ISBN 978-7-302-40095-0

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

中国版本图书馆 CIP 数据核字(2015)第 089657 号

责任编辑：王 军 于 平

装帧设计：牛静敏

责任校对：成凤进

责任印制：刘海龙

出版发行：清华大学出版社

网 址：<http://www.tup.com.cn>, <http://www.wqbook.com>

地 址：北京清华大学学研大厦 A 座 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：清华大学印刷厂

装 订 者：三河市新茂装订有限公司

经 销：全国新华书店

开 本：185mm×260mm 印 张：51.75 字 数：1389 千字

版 次：2015 年 6 月第 1 版 印 次：2015 年 6 月第 1 次印刷

印 数：1~3000

定 价：99.80 元

产品编号：060402-01

译者序

毫无疑问，Java 是这些年来最流行的编程语言之一。它无处不在——计算机、手机、网站以及各种嵌入式设备中都存在着大量的 Java 应用程序，而其中应用最为广泛的应该就是 Java EE Web 应用程序(以及安卓应用程序，不过本书的主题是 Java EE 开发)。通过使用 Java EE 平台中的各种组件，我们可以轻松构建出稳定而功能丰富的企业级 Web 应用程序。

随着 Spring 的出现，Java 开发更是变得简洁和轻松。Spring 是一个一站式的开发框架，它通过自身实现和第三方集成两种方式提供了 Java 企业应用程序表现层、业务层、持久层等相关技术。而它的几个特性——依赖注入(DI)、反转控制(LoC)和面向切面编程(AOP)，相信大家更是应该耳熟能详了。本书将对 Spring 框架的这些特性进行深入的讲解。另外，本书还将讲解如何使用 Spring Security 保护自己的应用程序。

在开发过程中的另一个重要技术就是如何存储数据了，在这方面对象关系映射(O/RM)得到了充分的发展，涌现出了一大批优秀的框架(Hibernate、iBatis、Toplink 等)，而 Hibernate 更是其中的佼佼者。随后又出现了统一的规范 JPA，又再次促进了对象关系映射的发展。而对于开发者来说，这也简化了大家的学习曲线，我们只需要掌握 JPA 规范就可以轻松地在各种不同实现之间切换。

无论是 Java EE 开发，还是 Spring 和 JPA，它们都拥有丰富的内容，任意一项其实都可以拿出来单独通过一本书来讲解，而在这里本书对这些内容进行了巧妙的组合，既对开发中经常使用的内容进行了详细的讲解，也保证了内容的清晰。每章除了自己独有的样例之外，本书还使用了一个贯穿全书的样例，通过不断对它进行改进来演示真实的开发过程。除此之外，本书还对 Java SE 7 和 8 中新增的特性进行了讲解。

本书主要面向已经具有丰富的 Java 语言和 Java SE 平台知识的软件开发者和软件工程师。开发者通过阅读本书可以扩展自己 Java 方面的知识，提升自己的技能。架构师通过阅读本书可以在团队项目中应用本书详细讨论的一些 Web 软件开发概念和模式。团队管理者通过阅读本书也可以扩展自己的知识库，使他们可以更好地与开发者进行沟通，并提出解决某些问题的建议。总而言之，本书是一本学习各种 Java EE 开发技术的佳作。

另外，感谢清华大学出版社的编辑们为本书付出的心血。同样感谢妻子对我翻译工作的支持和鼓励。没有你们的支持，本书就不可能顺利出版。

对于这本经典之作，译者对本书进行了严格审校，对其中一些具有争议的地方也进行了反复的考证，但个人精力有限，难免有疏漏之处，敬请各位读者见谅。如有任何意见或建议，请不吝指正。本书全部章节由王肖峰翻译，参与翻译工作的还有白广州、杜欣、高国一、韩丽威、胡银、孙其淳、孙绍辰、王小红、王耀光、徐保科、尤大鹏、张立红、邓伟、李宾、马宁宁、余建伟、王鹤、王蕊、袁强强、张宏。

最后，祝愿各位读者通过阅读本书可以熟练掌握 Java EE 开发中涉及的各种技术，从而开发出更加强大的企业级应用程序。

作者简介



Nick Williams 就职于 UL Workplace Health and Safety(位于田纳西州的富兰克林), 是一位软件工程师。从贝尔蒙特大学获得计算机科学硕士学位之后, 他从事商业和开源软件项目超过了 9 年时间。他同时也是 DNSCrawler.com 的创建者, 该网站用于提供免费的 DNS 和 IP 故障排除工具, 同时还提供了 NWTS Java 代码。另外, 它还是一个专注于编写满足商业需求的 Java 库的开源社区。在 2010 年, 纳什维尔技术委员会(Nashville Technology Council)授予他中部田纳西州最杰出软件工程师称号。Nick 是 Apache Logging(包括 log4j)

和 Jackson Data Processor JSR 310 Data Types 的代码提交者。他还为 Apache Tomcat 8.0、Spring Framework 4.0、Spring Security 3.2、Spring Data Commons 1.6、Spring Data JPA 1.4 和 JBoss Logging 3.2 添加了一些新的特性; 他也是包括 OpenJDK 在内的其他几个项目的贡献者; 并且他还是 Java Community Process(JCP)的成员。

技术编辑简介

Jake Radakovich 在 2009 年加入了 UL Workplace Health and Safety, 现在是 Occupational Health Manager 产品的一位软件开发者。在此之前, 他曾经是中部田纳西州立大学的一位研究助理, 主要参与开发 AlgoTutor, 这是一个基于 Web 的算法开发系统。他在中部田纳西州立大学获得计算机科学和数学的学士学位。

Manuel Jordan Elera 是一位自主的开发者和研究者, 他喜欢通过自己的实验学习新的技术, 也喜欢创造新的集成技术。他曾获得 2010 年的 Springy Award, 而且还是 2013 年的 Community Champion 和 Spring Champion。在休闲时间内, 他喜欢阅读圣经和创作乐曲。Manuel 是 Spring 社区论坛的一位高级成员, 他的 ID 为 dr_pompeii。你可以通过他的博客了解和联系他, 也可以通过他的 Twitter 账户 @dr_pompeii 与他沟通。

致 谢

感谢我的妻子 Allison，在这充满了压力的一年中，她一直支持我的工作，还不断地提醒我本书的截稿日期，帮助我顺利完成了本书的编写。

感谢我的父母和兄弟姐妹，他们确信我一定能够完成任何想做的事情。

感谢 Joyce Blair Crowell 和 William Hooper 博士，他们的帮助和教导使我受益匪浅。

感谢 Sarah Ann Stewart 博士，感谢他在我最沮丧的时候给予了我极大的信任。

感谢 Mrs. Lockhart，感谢他鼓励我进行本书的创作。

感谢 Jay，感谢他将我介绍给了 Mary。感谢 Mary 和 Maureen，他们使本书的出版成为现实。

感谢 Jake。噢，感谢他愿意成为我的技术编辑。

前 言

尽管许多人并没有意识到，但实际上大多数人每天都在使用 Java。它无处不在——在 TV 中，在蓝光播放器中，在计算机中。一些流行的智能手机也运行在基于 Java 的操作系统上；并且它为你每天使用的许多网站都提供了技术支持。当你想到 Java 时，可能自然会想象到浏览器 applet 或者与操作系统中其他应用程序风格不匹配的桌面应用程序。你可能甚至会想到一直通知你升级 Java 的、让人讨厌的系统托盘提醒。

但其实 Java 的应用远比你能想到的(每天可以见到的)要广泛。Java 是一门强大的语言，但它的强大之处更多地体现在它的平台中。尽管 Java SE 平台已经提供了创建控制台、桌面和浏览器应用程序所必不可少的工具，但 Java EE 平台仍然对它进行了巨大的扩充，Java EE 平台可以帮助创建内容丰富的强大 Web 应用程序。本书将对这些工具进行讲解，并向你展示如何创建现代的、有用的企业级 Java Web 应用程序。

0.1 本书面向的读者

本书主要面向已经具有丰富的 Java 语言和 Java SE 平台知识的软件开发者和软件工程师。本书可以通过自学完成，现有的 Java 开发者通过学习本书可以扩展自己 Java 方面的知识，提升自己的技能——从 applet、控制台或桌面应用开发发展到企业级 Web 应用开发。你可以从头到尾按顺序阅读本书的所有章节，也可以挑选其中一些感兴趣的章节进行阅读，将本书作为参考书使用。尽管某些章节可能会引用到之前章节中的例子，但本书已经努力使各章内容变得更加独立。所有的样例代码都可以从 wrox.com 网站和 <http://www.tupwk.com.cn/downpage> 获得，当某个例子依赖于之前章节中的另一个例子时，这些样例代码可以帮助你轻松地查看示例的完整内容，而无须寻找另一章节的代码。

本书对于已经具有 Java EE 平台经验的开发者也是非常有用的，它可以帮助他们提升自己的技能或者学习一些最新 Java EE 版本中的新特性。本书对于软件架构师来说也是非常有用的，因为除了具体的工具和平台组件之外，本书还对几种不同的 Web 软件开发概念和模式进行了详细讲解。本书可以帮助架构师在团队的项目中应用这些新的想法。

如果你是一个软件开发团队的管理者，那么本书对于你来说也是非常有用的。毫无疑问，你每天都在努力地与团队中的开发者和工程师沟通。通过阅读本书，你可以扩展自己的知识面，理解开发者使用的工具，从而更好地进行沟通，并可以为你的团队提供解决某些问题的建议。在阅读本书之后，你也可以为自己的团队购买几本该书，帮助团队中的开发者提高技能，并在项目中应用这些概念。

最后，老师和学生们也可以将该书用于课堂中。作为教科书，它可以是宝贵的 300 和 400

级课程，它为学生讲解了实际工作中将会用到的技能，这些内容有助于他们毕业后在职场上获得成功。

0.2 本书不面向的读者

本书不适合那些没有 Java 经验并且从未编写或编译过 Java 应用程序的读者。如果你之前没有任何 Java 经验，那么可能会发现本书中的内容和示例都难于理解。这是因为本书未涉及 Java 语言语法或者 Java SE 平台的规范。本书假定读者都已经能够熟练编写、编译和调试 Java 代码，并且熟悉标准平台(Java SE)。只有一些在 Java SE 8 中添加的新特性和工具才会出现在本书中。

另外，读者最好了解以下技术和概念。尽管其中一些概念可能看起来很简单，但需要注意的是，如果你不熟悉其中的某些概念，在阅读本书的某些章节时，可能会感到很困难。

- Internet、TCP、HTTP 协议
- 超文本标记语言(HTML)，包括 HTML5
- 可扩展标记语言(XML)
- JavaScript 或 ECMAScript，包括 jQuery 和浏览器调试工具
- 层叠样式表(CSS)
- 结构化查询语言(SQL)和关系数据库，尤其是 MySQL(如果你熟悉其他关系数据库的话，也可以轻松地使用 MySQL)
- 事务和事务概念，例如 ACID(原子性、一致性、隔离性、持久性)
- 集成开发环境(IDE)的使用
- 简单命令行任务的执行(不需要精通命令行)

0.3 本书会涉及的内容

本书将对 Java EE 平台版本 7 和其中的许多技术进行详细讲解。本书首先将介绍什么是 Java EE 平台以及它的发展过程，接着介绍应用服务器和 Servlet 容器以及它们的工作原理。然后讲解 Spring Framework、发布-订阅、高级消息队列协议(AMQP)、对象关系映射(O/RM)、Hibernate ORM、Spring Data、全文搜索、Apache Lucene、Hibernate Search、Spring Security 和 OAuth。本书还将对下列 Java EE 7 组件进行讲解：

- Servlets 3.1 – JSR 340
- JavaServer Pages (JSP) 2.3 – JSR 245
- Java Unified Expression Language (JUEL 或仅 EL) 3.0 – JSR 341
- Java API for WebSockets – JSR 356
- Bean Validation (BV) 1.1 – JSR 349
- Java Message Service (JMS) 2.0 – JSR 343
- Java Persistence API (JPA) 2.1 – JSR 338
- Java Transaction API (JTA) 1.2 — JSR 907

本书还将广泛地使用 lambda 表达式和新的 JSR 310 Java 8 Date and Time API，它们都被添加到 Java SE 8 中。

第 I 部分：创建企业级应用程序

本部分将对 Servlet、过滤器、监听器和 JavaServer Pages(JSP)进行讲解。本部分首先将讲解 Servlet 如何响应 HTTP 请求，以及过滤器如何协助它完成对请求的处理。还将讲解如何使用 JSP 轻松创建出强大的用户界面，以及如何通过结合使用 JSP 标记和全新的 Expression Language 3.0，创建出不含 Java 代码的视图，这些视图可以由不具有 Java 知识的 UI 开发者进行维护。本部分还将讲解 HTTP 会话，以及如何使用它们创建出丰富的用户体验(可以跨越应用程序中的多个页面)。另外还会对一门全新的技术 WebSockets 进行讲解，通过它我们可以创建出更加丰富、更具有交互性的用户界面，因为它将在应用程序和客户端(例如浏览器)之间提供全双工的双向通信。最后，本部分将讲解应用程序日志的最佳实践和技术，当你创建了一个包含大量代码的复杂应用程序时，日志的使用是非常重要的。

第 II 部分：添加 Spring Framework

从第 II 部分开始，我们将开始使用 Spring Framework 和 Spring MVC。该部分包含的内容有：依赖注入(DI)、反转控制(IoC)和面向切面编程(AOP)。我们将使用 XML 和基于注解的配置搭建高级 Spring Framework 项目，还将使用 Spring 工具实现 bean 验证和国际化。我们将使用 Spring MVC 控制器和 Spring Web Services 创建出 RESful 和 SOAP Web 服务，还将学习如何使用 Spring Framework 内建的消息传送系统。最终我们将学习高级消息队列协议(AMQP)，并学习如何配置和使用 RabbitMQ。

第 III 部分：使用 JPA 和 Hibernate ORM 持久化数据

第 III 部分将专注于数据持久化和使用不同的方式将对象存储在数据库中。在介绍了使用原生 JDBC 持久化实体的一些基本问题之后，该部分将开始讲解对象关系映射(O/RM)和 Hibernate ORM 及其 API。接下来将讲解 Java Persistence API，该 API 抽象出了一些公共 API，不管底层使用的是哪种 O/RM 实现，我们都可以编写相同的代码。然后讲解了 Spring Data，以及它如何帮助在不用编写任何持久化代码的情况下，创建持久化应用程序。最后讲解了几种搜索持久化数据的不同方法，以及如何结合使用 Hibernate Search 和 Apache Lucene 作为潜在的全文搜索工具。

第 IV 部分：使用 Spring Security 保护应用程序

本书的最后一部分介绍了认证和授权的概念，并展示了同时可用于这两种目的的几种技术。然后讲解了如何在 Spring Framework 应用程序中集成 Spring Security。最后讲解了如何使用 OAuth 1.0a 和 OAuth 2.0 保护 Web 服务，以及如何创建自定义的访问令牌类型，对 OAuth 2.0 实现进行增强。

0.4 本书不会涉及的内容

本书不会讲解基本的 Java 语法或 Java SE 平台，尽管其中会有部分内容涉及 Java SE 7 和 Java SE 8 中新增的特性。本书也不会讲解如何编写基于 Java 的控制台应用程序、桌面应用程序或 applet。如果你需要这方面的书籍，Wrox 有许多书籍可供选择。

更重要的是，本书不会讲解如何管理 Java EE 应用服务器环境。现在有众多的应用服务器和 Web 容器可以使用，没有哪两种服务器管理方式是一样的。使用哪种应用服务器完全取决于应用程序的特性、商业需求、商业实践和服务器环境。所以讲解如何管理一些最常见的应用服务器也是不合实际的。学习如何部署和管理 Java EE 应用服务器或 Web 容器的最好方法是查询它的文档，在某些情况下，最好的方式是进行实验(因为 Web 容器的使用是完成本书示例的必需内容，所以第 2 章将对一些基本的任务进行讲解，包括安装、启动、停止以及部署应用程序到 Apache Tomcat)。

参考 0.2 节——本书不包含其中列出的技术和概念方面的知识。它也不包含下面的 Java EE 7 组件，这些组件在大多数简单的 Web 容器并未得到支持，在使用 Spring Framework 和与它相关的项目时也不需要这些组件。

- Java API for RESTful Web Services (JAX-RS) 2.0 – JSR 339
- JavaServer Faces (JSF) 2.2 – JSR 344
- Enterprise JavaBeans (EJB) 3.2 – JSR 345
- Contexts and Dependency Injection (CDI) 1.1 – JSR 346
- JCache – JSR 107
- State Management – JSR 350
- Batch Applications for the Java Platform – JSR 352
- Concurrency Utilities for Java EE – JSR 236
- Java API for JSON Processing – JSR 353

0.5 需要使用的工具

完成和运行本书中的示例需要使用几种不同的工具。在开始之前，请确保你已经在计算机中安装或启用以下工具：

- Apache Maven 版本 3.1.1 或更新的版本
- 执行特定任务的命令行以及提供了命令行读取的操作系统(换句话说，不能在智能手机或平板电脑上编译和运行示例)
- 一个强大的文本编辑器，用于完成某些任务，例如编辑配置文件。不要使用 Windows Notepad 或 Apple TextEdit 作为文本编辑器。如果需要新的文本编辑器，那么可以考虑：
 - **Windows** — Notepad++或 Sublime Text 2
 - **Mac OS X** — TextWrangler、Sublime Text 2 或 Vim
 - **Linux** — Sublime Text 2 或 Vim

0.5.1 支持 Java SE 8 的 Java 开发工具包

必须在计算机中安装支持 Java SE 8 的 Java 开发工具包(JDK)。Java SE 8 已在 2014 年 3 月发布。可以从 Oracle 标准 Java SE 下载网站(<http://www.oracle.com/technetwork/java/javase/downloads/index.html>)获得该 JDK。总是使用最新版的 JDK, 为自己的计算机下载合适的版本和架构。如果你的计算机使用的是 64 位处理器和 64 位操作系统, 那么应该下载 64 位的 Java 安装包。

0.5.2 集成开发环境

你需要一个集成开发环境或 IDE, 用于编译和执行样例代码以及正常的实验。IDE 通常也被称为交互式开发环境, 是一种包含了编码、编译、部署和调试功能的软件应用程序, 软件开发者使用它创建软件。现在有许多不同的 Java IDE 可供选择, 其中的某些 IDE 特别优秀。许多人觉得一种 IDE 比另一种好仅仅是因为个人的观点和经验——对于一位开发者来说非常优秀的 IDE 可能并不适用于另一位开发者。不过, 通常包含了智能代码建议、代码补全、代码生成、语法检查、拼写检查和框架集成(Spring Framework、JPA、Hibernate ORM 等)等功能的 IDE 都是非常有用的, 相对不提供这些功能的 IDE 来说, 这些功能会为开发者提供具有更高生产力的工作环境。

你可能已经拥有了自己经常使用的 IDE, 或者你只是使用文本编辑器和命令行。如果你已经有了 IDE, 那么它也可能无法运行本书中的示例。在选择 IDE(或者评估当前使用的 IDE 能否满足需要)时, 应该选择一个包含了智能代码补全和建议、语法检查和集成 Java EE、Spring Framework、Spring Security、Spring Data、JPA 和 Hibernate ORM 等功能的 IDE。这意味着它需要能够检查 Java EE、Spring、JPA 和 Hibernate 配置, 并告诉你这些配置中是否包含错误或问题。下面将介绍三种支持多种语言的 IDE, 并针对本书做出建议。

1. NetBeans IDE 8.0

NetBeans——免费的 IDE——是一个由 Oracle 赞助的标准 Java IDE, 类似于 Microsoft Visual Studio 是 .NET 开发的标准 IDE。不过它不是最流行的 Java IDE。只有 NetBeans IDE 8.0 才支持 Java SE 8 和 Java EE 7, 之前的版本不支持。NetBeans 提供了强大的特性集, 并为所有 Java EE 特性提供了内建支持。它还支持 C、C++ 和 PHP 开发。你也可以通过插件扩展 NetBeans 的功能, 现在有支持 Spring Framework 和 Hibernate ORM 的插件。不过, NetBeans 特性集不如其他 IDE 丰富, 所以在学习本书的过程中, 不推荐使用它。本书中的样例代码并未涉及 NetBeans 的下载格式, 但如果你选择使用 NetBeans, 那么可以通过导入 Maven 项目的方式导入代码。下载 NetBeans 的网址为 <https://netbeans.org/>。

2. 支持 Java EE 开发的 Eclipse Luna IDE 4.4

Eclipse 是另一个免费的 IDE, 并且是世界上应用最广泛的 Java IDE。其中的一个强大之处在于它的可扩展性, 与它支持插件的特性相比该特性要强大得多。使用 Eclipse 平台时, 可以为特定的任务或工作流完全自定义 IDE 的布局。它已经包含了支持 Spring Framework、Spring Data、Spring Security、Hibernate ORM 等的插件和扩展。Spring 社区也提供了一个

Eclipse 的自定义版本——称为 Spring Tool Suite——非常适用于基于 Spring 项目的开发。不过，以作者的观点来看，Eclipse 是一种很难高效使用的 IDE。完成很简单的任务却需要大量的工作。从历史上看，兼容的 Eclipse 版本通常都在 Java SE 和 EE 发布之后发布。在编写本书时，Eclipse 社区尚未发布兼容 Java SE 8 和 Java EE 7 的版本。因此，不推荐使用 Eclipse 运行本书中的示例。如果选择使用——或者继续使用——Eclipse 的话，那么需要获得支持 Java EE 开发的 Eclipse Luna IDE 4.4，它已在 2014 年 7 月发布。下载 Eclipse IDE 的网址为 <http://www.eclipse.org/downloads/>。

由于 Eclipse IDE 的广泛应用，本书中的样例代码可以作为 Eclipse 项目下载，只要 Eclipse Luna 4.4 能够支持它们的运行。

3. IntelliJ IDEA 13 终极版

JetBrains 的 IntelliJ IDEA 不论是社区版(免费的)还是终极版(付费的)都是具有丰富功能的 Java IDE。以作者的观点来看，它是最易用也是最强大的 Java IDE。它的代码建议和补全功能以及框架支持都是其他任何 IDE 所无法比拟的。另外，从历史上看，它对 Java SE 和 Java EE 发布之前的实验版本提供了较好的支持。例如，IntelliJ IDEA 12 在 2012 年 12 月已经提供了对 Java SE 8 的支持——比 Java SE 8 的发布早了整整 15 个月，比 Eclipse IDE 对 Java SE 8 的支持早了 18 个月。如果你喜欢在 Java SE 和 Java EE 发布之前测试它们的新版本，并在发布之后立即使用它们，那么 IntelliJ IDEA 是你必然的选择。

不过它的强大是有代价的。社区版本可用于许多不同类型的 Java SE 项目，但不可用于 Java EE 项目。如果需要完整地支持 Java EE、Spring 项目和 Hibernate ORM，那么就要购买终极版。终极版的定价是非常合理的，对于公司、个人和学生来说都是有竞争力的，它只是 Microsoft Visual Studio 同等版本花费的一小部分。教育机构可以获得免费许可用于正式的教学使用，开源组织也可以获得免费许可用于项目开发。在这里，你可以从 <http://www.jetbrains.com/idea/download/> 下载 IntelliJ IDEA 13 终极版的 30 天免费版，之后你可以购买一份许可(或者如果你符合条件的话，可以申请免费许可)，可用在以后任何时间下载的版本中。对于本书中的所有样例代码，推荐使用 IntelliJ IDEA 终极版。直到 Eclipse Luna 4.4 能够支持本书示例的运行之前，所有的下载代码都可被用于 IntelliJ IDEA 项目中。

请确保下载 IntelliJ IDEA 的最新版本。尽管版本 13.0.x 是本书出版之前的最新版本，但版本 13.1.x 已在 2014 年 4 月发布，它将包含几个对 Spring Framework 和 Java EE 7 支持的改进，版本 14.0.x 已在 2014 年 11 月发布。

0.5.3 Java EE 7 Web 容器

在阅读本书时最后一个需要的工具是 Java EE Web 容器(必须是实现了 Servlet、JSP、JUEL 和 Java EE 7 中 WebSocket 规范的 Web 容器)。第 2 章将对该内容进行详细的讲解，并介绍大多数流行的 Web 容器和应用服务器，以及如何下载、安装和使用 Apache Tomcat 8.0。

0.6 本书约定

在本书中，有几种约定可用于帮助你注意某些内容或者显示代码中的某些问题。本节列出了这些约定的示例。



注意：注意表示注意、提示、技巧、提醒或者与当前讨论内容有关的其他有趣信息。要注意这些方框。



警告：警告包含了一些与周围文本直接相关，并且一定要牢记的重要信息。警告通常表示陷阱、危险和潜在的损失或者损坏的数据。要关注这些方框。

0.7 代码示例

作为一本软件开发书籍，该书广泛地应用了代码示例，通过它们证明我们正在讨论的主题。大多数情况下，这些示例都是完整的 IDE 项目，你可以在 IDE 中打开、编译和执行。所有的示例都可以从 wrox.com 代码下载网站和 <http://www.tupwk.com.cn/downpage> 获得。请访问网址 <http://www.wrox.com/go/projavaforwebapps> 并单击 Download Code 选项卡。你可以将所有代码示例下载为单个 ZIP 文件或者将每章的示例单独下载为一个 ZIP 文件。在每章的示例中，可以找到每个示例的两个版本：一个 IntelliJ IDEA 项目和一个 Eclipse 项目。使用自己选择的 IDE 对应的版本即可。如果你使用的不是这两种 IDE 中的某一个，那么你的 IDE 应该能够将 IntelliJ IDEA 项目作为简单的 Maven 项目导入。



注意：请记住，Eclipse 版本的代码示例在 Eclipse Luna 4.4 发布之后才能获得。如果在此之前阅读该书，请下载 IntelliJ IDEA 示例项目。

在本书的初始部分，你可以直接在 IDE 中创建这些示例，而无须从代码网站下载(如果你愿意的话)。不过当示例变得越来越复杂时，这种方式就不可取了。最重要的代码在书中已经打印了出来，但打印出每一行代码是不实际的。另外，大部分省略的代码都是重复的。例如，从第 II 部分到第 IV 部分的大多数示例项目中的 Spring Framework 配置基本是一致的。在这种情况下，只显示出配置中与之之前章节的不同部分才是合理的，而不是重新打印出完整的配置。出于这个原因，如果你希望执行并测试这些示例的话，就需要从 wrox.com 代码下载网站下载大多数样例代码。

在每章的首页，你将会看到一块标题为“本章需要从 wrox.com 下载的代码”的区域。

该部分列出了本章将使用的所有样例代码的名字，并提醒你下载这些代码示例的链接。其中一些章节未包含代码示例下载，但大多数都包含。

0.8 Maven 依赖

本书中的样例代码将广泛运用第三方依赖，例如 Spring Framework、Hibernate ORM 和 Spring Security。在下载网站的代码中包含这些依赖 JAR，将使下载文件变大，你可能需要下载几百兆的文件。为了解决这个问题，样例代码将使用 Apache Maven 及其依赖管理功能。所有的示例项目都是 Maven 项目。在 IDE 中打开每个项目时，IDE 将自动在本地 Maven 库中查找这些依赖，或者如果需要的话，IDE 将下载它们到本地 Maven 库中。

在每章的首页，你将会看到一个标题为“本章新增的 Maven 依赖”的区域。该部分列出了本章中新增的 Maven 依赖。你也可以查看 pom.xml 文件来检查每个示例项目的依赖。某些章节未引入新的 Maven 依赖，但大多数章节都引入了。

每个 Maven 依赖都会有作用域，用于定义依赖在哪个类路径上可用。最常见的作用域——“compile”作用域——表示该依赖在项目的编译类路径上、单元测试编译和执行类路径上以及运行应用程序时的最终运行时类路径上可用。在 Java EE Web 应用程序中，这意味着依赖将被复制到所部署的应用程序中。“runtime”作用域表示依赖将在单元测试执行和运行时执行类路径上可用，但不像“compile”作用域一样在编译应用程序或者执行单元测试时也可用。一个运行时依赖将被复制到所部署的应用程序中。最后，“provided”作用域表示运行应用程序的容器将会提供该依赖。在 Java EE 应用程序中，这意味着依赖已经在 Servlet 容器或应用服务器的类路径上了，因此不需要再复制到部署的应用程序中。Maven 和 IDE 将保证在编译应用程序和执行单元测试时，“provided”作用域的依赖是可用的。Maven 还有其他类型的作用域，但在本书中将只会用到这些。

在文本和示例项目中，有一些 Maven 依赖将会包含排除(exclusions)，用于忽略某些特定的依赖——它们被称为临时依赖。很多时候这些排除都是多余的，列出它们只是为了使代码更清晰。当某个依赖依赖于另一个旧版本的依赖，而不是现在正在使用的依赖时，通过排除可以更清楚地表示这里出现了矛盾，并且避免了由 Maven 的最近算法所引起的问题。不过，某些排除的存在是由于 Java SE 或 Java EE 的新版本已经提供了依赖，或者因为依赖 ID 发生了变化。当出现这种情况时，文本中会进行提示。

0.9 将安全相关的内容安排在最后的原因

坦白地说，应用程序安全会影响其他内容的学习。在产品中添加认证和授权所使用的技术将会干扰你的代码，并使学习过程变得困难。安全是第一位的，而且一直记得安全问题永远也不是错误。不过，只要使用了正确的工具，在项目完成(或基本上完成)之后为它添加验证和授权也是相当简单的一件事。本书首先将关注于如何使用工业标准工具创建出高质量、特性丰富的 Web 应用程序。在学会了创建强大应用程序所需的技能之后，本书的第IV部分将向你展示如何为现有的应用程序添加认证和授权，保护它不受未授权的和恶意访问的攻击。

0.10 勘误表

我们力图使本书尽可能地全面和准确，但没有人是完美的，所以书中仍然会有错误存在。本书偶尔可能会包含一些需要修正的错误。如果你发现了内容错误、拼写错误或错误代码，请告诉我们！通过您提供的反馈，可以帮助节省其他读者解决这些问题的时间和努力，同时也可以帮助改进本书未来的版本。

如果你希望阅读本书已经发现的勘误表，请访问 Wrox 的网站 <http://www.wrox.com/>，并使用搜索框寻找该书名。搜索 ISBN 是最快的方式。在本书的首页，单击勘误表的链接。在这里你可以看到所有由读者提交并由 Wrox 编辑验证过的错误。如果你无法解决自己发现的错误，请访问 Wrox 的技术支持页面并填写报告问题的表单。在我们验证该错误并修正之后，我们将把它发布到本书的勘误页面，并在未来的版本中解决该问题。

目 录

第 I 部分 创建企业级应用程序	
第 1 章 介绍 Java EE 平台3	
1.1 Java 平台时间线.....3	
1.1.1 起始.....3	
1.1.2 企业级 Java 的诞生.....4	
1.1.3 Java SE 和 Java EE 共同发展.....5	
1.1.4 了解最新的平台特性.....7	
1.1.5 持续发展.....11	
1.2 了解基本的 Web 应用程序结构.....11	
1.2.1 Servlet、过滤器、监听器和 JSP.....11	
1.2.2 目录结构和 WAR 文件.....12	
1.2.3 部署描述符.....13	
1.2.4 类加载器架构.....14	
1.2.5 企业级应用程序归档文件.....14	
1.3 小结.....15	
第 2 章 使用 Web 容器17	
2.1 选择 Web 容器.....17	
2.1.1 Apache Tomcat.....18	
2.1.2 GlassFish.....19	
2.1.3 JBoss 和 WildFly.....19	
2.1.4 其他容器和应用服务器.....20	
2.1.5 本书使用 Tomcat 的原因.....20	
2.2 在个人计算机中安装 Tomcat.....21	
2.2.1 将 Tomcat 安装为 Windows 服务.....21	
2.2.2 将 Tomcat 安装为命令行应用程序.....21	
2.2.3 配置自定义的 JSP 编译器.....23	
2.3 在 Tomcat 中部署和卸载应用程序.....24	
2.3.1 手动部署和卸载.....24	
2.3.2 使用 Tomcat 管理器.....25	
2.4 通过 IDE 调试 Tomcat.....26	
2.4.1 使用 IntelliJ IDEA.....27	
2.4.2 使用 Eclipse.....30	
2.5 小结.....34	
第 3 章 创建第一个 Servlet35	
3.1 创建 Servlet 类.....36	
3.1.1 选择要继承的 Servlet 类.....36	
3.1.2 使用初始化方法和销毁方法.....38	
3.2 配置可部署的 Servlet.....39	
3.2.1 向描述符中添加 Servlet.....39	
3.2.2 将 Servlet 映射到 URL.....40	
3.2.3 运行和调试 Servlet.....42	
3.3 了解 doGet、doPost 和其他方法.....44	
3.3.1 在 service 方法执行的过程中.....44	
3.3.2 使用 HttpServletRequest.....44	
3.3.3 使用 HttpServletResponse.....47	
3.4 使用参数和接受表单提交.....49	
3.5 使用初始化参数配置应用程序.....53	
3.5.1 使用上下文初始化参数.....53	
3.5.2 使用 Servlet 初始化参数.....54	
3.6 通过表单上传文件.....56	