

MANNING

全球有超过 100 000 的开发者使用本书来学习 Spring
畅销经典 Spring 技术图书，针对 Spring 4 全新升级

Spring

实战

(第4版)

Spring IN ACTION

FOURTH EDITION

[美] Craig Walls 著
张卫滨 译



中国工信出版集团

人民邮电出版社
POSTS & TELECOM PRESS

 MANNING

Spring 实战

(第4版)



[美] Craig Walls 著
张卫滨 译

人民邮电出版社
北京

图书在版编目 (C I P) 数据

Spring实战 : 第4版 / (美) 沃尔斯 (Walls, C.)
著 ; 张卫滨译. — 北京 : 人民邮电出版社, 2016. 4
ISBN 978-7-115-41730-5

I. ①S… II. ①沃… ②张… III. ①JAVA语言—程序设计 IV. ①TP312

中国版本图书馆CIP数据核字(2016)第033868号

版 权 声 明

Original English language edition, entitled Spring in Action, 4th Edition by Craig Walls Bibeault published by Manning Publications Co., 209 Bruce Park Avenue, Greenwich, CT 06830. Copyright ©2015 by Manning Publications Co.

Simplified Chinese-language edition copyright ©2016 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由 **Manning Publications Co.** 授权人民邮电出版社独家出版。未经出版者书面许可, 不得以任何方式复制本书内容。

版权所有, 侵权必究。

-
- ◆ 著 [美] Craig Walls
 - 译 张卫滨
 - 责任编辑 陈冀康
 - 责任印制 张佳莹 焦志炜
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
 - 邮编 100164 电子邮件 315@ptpress.com.cn
 - 网址 <http://www.ptpress.com.cn>
 - 北京艺辉印刷有限公司印刷
 - ◆ 开本: 800×1000 1/16
 - 印张: 37.5
 - 字数: 811千字 2016年4月第1版
 - 印数: 1-4000册 2016年4月北京第1次印刷
 - 著作权合同登记号 图字: 01-2014-7524号

定价: 89.00元

读者服务热线: (010) 81055410 印装质量热线: (010) 81055316

反盗版热线: (010) 81055315

内容提要

本书是经典的、畅销的 Spring 学习和实践指南。

第 4 版针对 Spring 4 进行了全面更新。全书分为 4 部分。第 1 部分介绍 Spring 框架的核心知识。第 2 部分在此基础上介绍了如何使用 Spring 构建 Web 应用程序。第 3 部分告别前端，介绍了如何在应用程序的后端使用 Spring。第 4 部分描述了如何使用 Spring 与其他的应用和服务进行集成。

本书适用于已具有一定 Java 编程基础的读者，以及在 Java 平台下进行各类软件开发的开发人员、测试人员，尤其适用于企业级 Java 开发人员。本书既可以被刚开始学习 Spring 的读者当作学习指南，也可以被那些想深入了解 Spring 某方面功能的资深用户作为参考用书。

关于本书

Spring 框架是以简化 Java EE 应用程序的开发为目标而创建的。同样，本书是为了帮助读者更容易地使用 Spring 而编写的。我的目标不是为读者详细地列出 Spring API，而是希望通过现实中的实际示例代码来为 Java EE 开发人员展现 Spring 框架。因为 Spring 是一个模块化的框架，所以这本书也是按照这种方式编写的。我们知道并不是所有的开发人员都有相同的需求，有些人想从头学习 Spring，而有的可能只想排出几个主题，然后按照自己的节奏来学习。所以，本书既可以被刚开始学习 Spring 的读者当作学习指南，也可以被那些想深入了解某方面功能的读者作为参考。

本书适用于所有的 Java 开发人员，企业级 Java 开发人员将会发现更有帮助。我会循序渐进地指导读者浏览本书中每章复杂的示例代码，但 Spring 的真正强大之处在于它能够使企业级应用程序的开发更简单。因此，企业级应用程序的开发人员会更加欣赏本书的示例代码。因为 Spring 的绝大部分内容都是提供企业级服务的，所以这里包含了许多 Spring 和 EJB 的比较。

路线图

本书分为 4 部分。第 1 部分介绍 Spring 框架的核心知识。第 2 部分在此基础上介绍如何使用 Spring 构建 Web 应用程序。第 3 部分告别前端，介绍如何在应用程序的后端使用 Spring。第 4 部分描述如何使用 Spring 与其他的应用和服务进行集成。

在第 1 部分中，读者将会学习到 Spring 容器、依赖注入（dependency injection, DI）和面向切面编程（aspect-oriented programming, AOP），也就是 Spring 框架的核心。这能让读者很好地理解 Spring 的基础原理，而这些原理将会在本书各个章节都会用到。

- 第 1 章将会概要地介绍 Spring，包括 DI 和 AOP 的一些基本样例。同时，读者还会了解到更大的 Spring 生态系统的整体情况。
- 第 2 章更为详细地介绍 DI，展现应用程序中的各个组件（bean）如何装配在一起。这包括基于 XML 装配、基于 Java 装配以及自动装配。

- 在掌握了基本的 bean 装配后，第 3 章会介绍几种高级装配技术，读者可能并不会经常用到这些技术，但是如果用到的话，本章的内容将会告诉读者如何发挥 Spring 容器最强大的威力。
- 第 4 章介绍如何使用 Spring 的 AOP 来为对象解耦那些对其提供服务的横切性关注点。这一章也为后面各章提供基础，在后面读者将会使用 AOP 来提供声明式服务，如事务、安全和缓存。

在第 2 部分中，读者将会看到如何使用 Spring 来构建 Web 应用程序。

- 第 5 章介绍使用 Spring MVC 的基础知识，这是 Spring 中的基础 Web 框架。读者将会看到如何编写控制器来处理请求，并使用模型数据产生响应。
- 当控制器的工作完成后，模型数据必须要使用一个视图来进行渲染。第 6 章将会探讨在 Spring 中可以使用的各种视图技术，包括 JSP、Apache Tiles 以及 Thymeleaf。
- 第 7 章的内容不再是 Spring MVC 的基础知识了，在本章中，读者将会学习到如何自定义 Spring MVC 配置、处理 multipart 类型的文件上传、处理在控制器中可能会出现的异常并且会通过 flash 属性在请求之间传递数据。
- 第 8 章将会介绍 Spring Web Flow，这是 Spring MVC 的一个扩展，能够开发会话式的 Web 应用程序。在本章中，读者将会学习到如何构建引导用户完成特定流程的 Web 应用程序。
- 第 9 章读者将会学到如何使用 Spring Security 为自己的应用程序 Web 层实现安全性。

第 3 部分所关注的内容不再是应用程序的前端了，而是关注于如何处理和持久化数据。

- 第 10 章首先会介绍如何使用 Spring 对 JDBC 的抽象实现关系型数据库中的数据持久化。
- 第 11 章从另外一个角度介绍数据持久化，也就是使用 Java 持久化 API (JPA) 存储关系型数据库中的数据。
- 第 12 章将会介绍如何将 Spring 与非关系型数据库结合使用，如 MongoDB 和 Neo4j。
- 不管数据存储在哪里，缓存都有助于性能的提升，这是通过只有在必要的时候才去查询数据库实现的。第 13 章将会为读者介绍 Spring 对声明式缓存的支持。
- 第 14 章重新回到 Spring Security，将会介绍如何通过 AOP 将安全性应用到方法级别。

本书的最后一部分会介绍如何将 Spring 应用程序与其他系统进行集成。

- 第 15 章将会学习如何创建与使用远程服务，包括 RMI、Hessian、Burlap 以及基于 SOAP 的服务。

- 第 16 章将会再次回到 Spring MVC，我们将会看到如何创建 RESTful 服务，在这个过程中所使用的编程模型与之前在第 5 章中所描述的是一致的。
- 第 17 章将会探讨 Spring 对异步消息的支持，本章将会包括 Java 消息服务 (Java Message Service, JMS) 以及高级消息队列协议 (Advanced Message Queuing Protocol, AMQP)。
- 在第 18 章中，异步消息有了新的花样，在这一章中读者会看到如何将 Spring 与 WebSocket 和 STOMP 结合起来，实现服务端与客户端之间的异步通信。
- 第 19 章将会介绍如何使用 Spring 发送 E-mail。
- 第 20 章会关注于 Spring 对 Java 管理扩展 (Java Management Extensions, JMX) 功能的支持，借助这项功能可以对 Spring 应用程序进行监控和修改运行时配置。
- 最后，在第 21 章，读者将会看到一个全新并且会改变游戏规则的 Spring 使用方式，名为 Spring Boot。我们将会看到 Spring Boot 如何将 Spring 应用中样板式的配置移除掉，这样就能让读者更加专注于业务功能。

代码规范与下载

本书中有大量的示例代码。这些代码将会使用固定宽度的代码字体。本书正文中的类名、方法名或 XML 片段也都使用代码字体。

很多 Spring 类和包的名字很长（不过会有较强的表达性）。鉴于此，我们有时候会用到换行符（↵）。

本书中的示例代码并不都是完整的。为了关注某个主题，我有时候只会展示类的一个或两个方法。本书所构建的应用程序完整代码可以在出版社站点上下载，地址是 www.manning.com/SpringinActionFourthEdition。

作者在线

购买了本书，读者就可以免费访问 Manning 出版社提供的在线论坛，在这里读者可以给本书写评论，问一些技术问题并可以得到作者和其他用户的帮助。要进入这个论坛或订阅它，读者可以在浏览器中访问 www.manning.com/SpringinActionFourthEdition。这个页面会告诉读者注册后怎样进入论坛，能够得到什么帮助以及论坛的规则。

Manning 对读者的许诺是为读者提供一个交流平台，在这里读者之间以及读者和作者之间可以进行有意义的交流。对于作者来说，对论坛进行多少次的访问不是强制的，他们对本书论坛的贡献是自愿和免费的。我们建议读者尽量向作者问一些有挑战性的问题，以保持他们的兴趣！

只要本书还在发售，读者就可以访问作者在线论坛以及以前讨论的归档信息。

作者简介

Craig Walls 是 Pivotal 的高级工程师，是 Spring Social 和 Spring Sync 的项目领导者，同时也是 Manning 出版社各版 *Spring In Action* 的作者，目前这本书已经更新到了第 4 版。他非常热心于 Spring 框架的推广，经常在当地的用户组和会议上演讲并在博客上撰写 Spring 相关的内容。在不琢磨代码的时候，Craig Walls 会尽可能多地陪伴他的妻子和两个女儿。

封面插图简介

《Spring 实战》第 4 版的封面人物是“Le Caraco”，也就是约旦西南部卡拉克(Karak)省的居民。该省的首府是 Al-Karak，那里的山顶有座古城堡，对死海和周边的平原有着极佳的视野。这幅图出自 1796 年出版的法国旅游图书，*Encyclopédies Voyages*，该书由 J. G. St. Sauveur 编写。在那时，为了娱乐而去旅游还是相对新鲜的做法，而像这样的旅游指南是很流行的，它能够让旅行家和足不出户的人们了解法国其他地区 and 国外的居民。

Encyclopédies Voyages 中多种多样的图画生动描绘了 200 年前世界上各个城镇和地区的独特魅力。在那时，相隔数十千米的两个地区着装就不相同，可以通过着装判断人们究竟属于哪个地区。这本旅行指南展现了那个时代和其他历史时代的隔离感和距离感，这与我们这个运动过度的时代是截然不同的。

从那以后，服装风格发生了改变，富有地方特色的多样性开始淡化。现在，有时很难说一个洲的居民和其他洲的居民有什么不同。从积极的方面来看，我们或许是用原来文化和视觉上的多样性换来了个人风格的多变性，或者可以说是更为多样化和有趣的知识科技生活。

这本旅行指南中的图片反映了两个世纪前各个地区生活的多样性，我们现在用图书封面的方式对其进行了再现。Manning 出版社的员工都认为这是计算机行业中一个很有意思的创意。

前言

百尺竿头更进一步。十几年前，Spring 刚刚进入 Java 开发领域，其目标是简化企业级 Java 开发。它使用更为简单和轻量级的模型，该模型基于简单老式的 Java 对象，以此挑战了当时重量级的开发模型。

现在，已经过去了很多年，Spring 也发布了众多的版本，我们可以看到 Spring 在企业级应用开发领域已经有了巨大的影响力。对于无数的 Java 项目来说，它就是事实上的标准，并且对于一些规范和它本来想取代的框架，Spring 也对其演进产生了影响。毫无疑问，如果 Spring 不挑战之前版本的企业级 JavaBean (EJB) 规范的话，现在的 EJB 规范肯定是完全不同的一个样子。

但是，Spring 本身也在持续地演化和提升，它一直致力于将困难的开发任务进行简化，不断地为 Java 开发人员带来创新性的特性。在 Spring 最初所挑战的领域，Spring 已经突飞猛进，涉及的范围扩展到 Java 应用开发的各个方面。

因此，为了介绍 Spring 的现状，我们需要对这本书升级了。在本书上一版出版到现在的几年间，发生了太多的事情，想在这一版中将所有的变化都涵盖进来是不可能的。不过，在第 4 版的《Spring 实战》中，我依然会使其包含尽可能多的内容。下面列出了在这一版中新增的一些令人兴奋的新内容：

- 强调基于 Java 的 Spring 配置，基于 Java 的配置方案几乎可以用在所有 Spring 开发领域之中；
- 条件化的配置以及 profile 特性能够让 Spring 在运行时确定该使用或忽略哪些 Spring 配置；
- Spring MVC 的多项增强和改善，尤其是与创建 REST 服务相关的；
- 在 Spring 应用中使用 Thymeleaf 替代 JSP；
- 使用基于 Java 的配置启用 Spring Security；
- 使用 Spring Data，在运行时自动为 JPA、MongoDB 和 Neo4j 生成 Repository 实现；

- Spring 新提供的声明式缓存支持；
- 借助 WebSocket 和 STOMP，实现异步的 Web 消息；
- Spring Boot，改变使用 Spring 游戏规则的新方法。

如果在 Spring 方面读者已经有相当多经验的话，那么将会发现这些新元素对于自己的 Spring 工具箱来说是非常有价值的补充。如果读者是要学习 Spring 的新手，那么就赶上了学习 Spring 的一个好时代，这本书会帮助读者起步。

对于 Spring 的使用来说，这的确是一个令人兴奋的时代。在过去的 12 年里，在使用 Spring 进行开发以及编写与之相关的文章方面形成了一股浪潮。我迫不及待地想看到 Spring 接下来会做些什么！

译者序

3年前，有幸和耿渊同学合作翻译了《Spring 实战（第3版）》。3年的时光过去了，技术在不断发展，这本书也推出了最新的第4版，顺利将这本书翻译完成后，顿时感觉轻松了许多。译书是一件比较辛苦的工作，但是在这3年的时间内，每当看到有朋友选择本书来学习 Spring，自己觉得还是蛮有成就感的。所以，看到本书的第4版时，我迫不及待地联系编辑约定了本书的翻译事宜。

本书的作者 Craig Walls 先生，从10年前编写本书的第1版开始，持续把一件事情做好，紧跟技术的发展，不断地升级和更新这本书的内容，世界范围内无数的 Java 开发者通过这本书学习和掌握了 Spring 技术。

本书的主题是 Spring 框架，从十多年前问世以来，它一直致力于简化 JEE 应用的开发。从最初的挑战者，到现在诸多标准的制定者；从传统的 JEE 应用，到大数据、NoSQL、企业应用集成、批处理、移动开发等领域，Spring 都在参与和发挥影响力。新版本的 Spring 提供了更加丰富的功能，但更重要的是 Spring 在想尽办法简化开发人员的使用，包括自动配置、基于 Java 的配置，还有现在越来越受欢迎的 Spring Boot。Spring Boot 是对 Spring 本身的一种颠覆和革命，但是唯有这种颠覆，才会换来开发人员更多的喜爱和框架本身的发展。

这本书从第1版到第4版之所以长盛不衰，是因为它紧跟了技术的发展；Spring 十多年来一直受到 Java 开发者的青睐，是因为它不断地进步和改善，并且坚持最初的目标：简化企业级 Java 的开发。处于一个不断革新的领域，我们技术人员何尝不需要如此呢，只有不断地汲取新的知识，学习新的技术，才能保证不被时代所淘汰。

本书涵盖了 Spring 框架的许多领域，既有核心框架，也有各种功能扩展，不少的同学曾经对我言及，感觉书中所讲述的内容深度不够，但是我个人认为，对于开源框架的学习，我们会有不同的掌握深度，从最初的使用、配置，到设计原理，再到源码分析，一本书很难面面俱到深入介绍所有的内容，但是它却能够提供一个方向，让我们按图索骥深入学习更多的知识。

译书占用了大量的业余时间，因此感谢我的爱人，帮我承担了许多家务和带孩子的工作，还要感谢我的儿子，每天看到他的成长和进步，都让我感觉如果懈怠的话，该被小朋友嘲笑了。

尽管在翻译的过程中，我力争达到准确和通畅，并与作者进行了很多的沟通和交流，但限于水平和时间，肯定还有许多的不足或纰漏之处，热忱期待您提出意见，希望本书能够对您有用！您可以通过 levinzhang1981@126.com 联系我。

张卫滨

2015年11月于大连

致谢

在本书付印之前，在本书捆扎之前，在本书装箱之前，在本书交付运输之前，在本书到达你手里之前，在整个过程中，有很多双手都曾经接触过它。即便你阅读电子版，省去了上面所述的流程，在你所下载的位和字节上依然凝结着很多双手的辛勤劳动——编辑、审阅、录入以及校对。如果没有这么多人的付出，这本书也就不会存在了。

首先，我要感谢 Manning 辛苦工作的每个人，当这本书的进展速度没有达到预期时，他们给予了足够的耐心，并促使我完成这本书：Marjan Bace、Michael Stephens、Cynthia Kane、Andy Carroll、Benjamin Berg、Alyson Brener、Dottie Marisco、Mary Piergies、Janet Vail 以及幕后的其他很多人。

写书的时候，尽早和频繁的反馈是相当重要的，这一点与开发软件是一样的。当这本书还非常粗糙的时候，有些人审阅了初稿并提供反馈，帮助本书最终成型。要感谢下面的人：Bob Casazza、ChaoHo Hsieh、Christophe Martini、Gregor Zurowski、James Wright、Jeelani Basha、Jens Richter、Jonathan Thoms、Josh Hart、Karen Christenson、Mario Arias、Michael Roberts、Paul Balogh、Ricardo da Silva Lima。尤其要感谢 John Ryan，在本书交付前，他对书稿进行了全面的技术审校。

当然，我要感谢美丽的妻子，感谢她容忍我开始了这个新的写作工程，感谢她整个过程中所给予我的鼓励。我深深地爱着你。

Maisy 和 Madi，世界上最可爱的小姑娘，感谢你们的拥抱、欢笑以及对本书内容别出心裁的见解。

对于 Spring 团队的同事，怎么说呢？你们太酷了！能够作为推动 Spring 前进的团队中的一员，我感到非常荣幸和感激。你们层出不穷的新创意总是让我感到惊叹。

感谢我在用户组和 No Fluff/Just Stuff 会议上演讲时所遇到的每个人。

最后，感谢 Phoenicians，你们（以及 Epcot）太棒了！¹

¹ Phoenicians 指的是远古时代的腓尼基人，他们被认为是字母系统的创建者，基于字母的所有现代语言都由此衍生而来。在迪斯尼世界的 Epcot，有名为 Spaceship Earth 的时光穿梭体验，我们可以了解到人类交流的历史，甚至能够回到腓尼基人的时代，在这段旅程的旁白中这样说道：如果你觉得学习字母语言很容易的话，那感谢腓尼基人吧，是他们发明了它。这是作者的一种幽默说法。——译者注

第 1 部分 Spring 的核心

第 1 章 Spring 之旅 3

- 1.1 简化 Java 开发 4
 - 1.1.1 激发 POJO 的潜能 5
 - 1.1.2 依赖注入 5
 - 1.1.3 应用切面 11
 - 1.1.4 使用模板消除样板式代码 16
- 1.2 容纳你的 Bean 18
 - 1.2.1 使用应用上下文 19
 - 1.2.2 bean 的生命周期 20
- 1.3 俯瞰 Spring 风景线 21
 - 1.3.1 Spring 模块 22
 - 1.3.2 Spring Portfolio 24
- 1.4 Spring 的新功能 27
 - 1.4.1 Spring 3.1 新特性 27
 - 1.4.2 Spring 3.2 新特性 28
 - 1.4.3 Spring 4.0 新特性 30
- 1.5 小结 30

第 2 章 装配 Bean 33

- 2.1 Spring 配置的可选方案 34
- 2.2 自动化装配 bean 35
 - 2.2.1 创建可被发现的 bean 35
 - 2.2.2 为组件扫描的 bean 命名 38
 - 2.2.3 设置组件扫描的基础包 39
 - 2.2.4 通过为 bean 添加注解实现自动装配 40

- 2.2.5 验证自动装配 42
- 2.3 通过 Java 代码装配 bean 44
 - 2.3.1 创建配置类 44
 - 2.3.2 声明简单的 bean 45
 - 2.3.3 借助 JavaConfig 实现注入 46
- 2.4 通过 XML 装配 bean 48
 - 2.4.1 创建 XML 配置规范 48
 - 2.4.2 声明一个简单的 <bean> 49
 - 2.4.3 借助构造器注入初始化 bean 50
 - 2.4.4 设置属性 56
- 2.5 导入和混合配置 61
 - 2.5.1 在 JavaConfig 中引用 XML 配置 61
 - 2.5.2 在 XML 配置中引用 JavaConfig 63
- 2.6 小结 65

第 3 章 高级装配 67

- 3.1 环境与 profile 67
 - 3.1.1 配置 profile bean 69
 - 3.1.2 激活 profile 73
- 3.2 条件化的 bean 75
- 3.3 处理自动装配的歧义性 78
 - 3.3.1 标示首选的 bean 79
 - 3.3.2 限定自动装配的 bean 80
- 3.4 bean 的作用域 84

- 3.4.1 使用会话和请求作用域 86
- 3.4.2 在 XML 中声明作用域代理 87
- 3.5 运行时值注入 88
 - 3.5.1 注入外部的值 89
 - 3.5.2 使用 Spring 表达式语言进行装配 93
- 3.6 小结 99

第 4 章 面向切面的 Spring 101

- 4.1 什么是面向切面编程 102
 - 4.1.1 定义 AOP 术语 103
 - 4.1.2 Spring 对 AOP 的支持 105
- 4.2 通过切点来选择连

- 接点 107
 - 4.2.1 编写切点 108
 - 4.2.2 在切点中选择 bean 109
- 4.3 使用注解创建切面 109
 - 4.3.1 定义切面 110
 - 4.3.2 创建环绕通知 114
 - 4.3.3 处理通知中的参数 115
 - 4.3.4 通过注解引入新功能 118
- 4.4 在 XML 中声明切面 120
 - 4.4.1 声明前置和后置通知 122
 - 4.4.2 声明环绕通知 124
 - 4.4.3 为通知传递参数 125
 - 4.4.4 通过切面引入新的功能 127
- 4.5 注入 AspectJ 切面 128
- 4.6 小结 131

第 2 部分 Web 中的 Spring

第 5 章 构建 Spring Web 应用程序 135

- 5.1 Spring MVC 起步 136
 - 5.1.1 跟踪 Spring MVC 的请求 136
 - 5.1.2 搭建 Spring MVC 138
 - 5.1.3 Spitttr 应用简介 142
- 5.2 编写基本的控制器 143
 - 5.2.1 测试控制器 145
 - 5.2.2 定义类级别请求处理 146
 - 5.2.3 传递模型数据到视图中 147
- 5.3 接受请求的输入 153
 - 5.3.1 处理查询参数 153
 - 5.3.2 通过路径参数接受输入 155
- 5.4 处理表单 158
 - 5.4.1 编写处理表单的控制器 160
 - 5.4.2 校验表单 163
- 5.5 小结 166

第 6 章 渲染 Web 视图 167

- 6.1 理解视图解析 167
- 6.2 创建 JSP 视图 170

- 6.2.1 配置适用于 JSP 的视图解析器 170
- 6.2.2 使用 Spring 的 JSP 库 172
- 6.3 使用 Apache Tiles 视图定义布局 185
 - 6.3.1 配置 Tiles 视图解析器 185
- 6.4 使用 Thymeleaf 190
 - 6.4.1 配置 Thymeleaf 视图解析器 191
 - 6.4.2 定义 Thymeleaf 模板 192
- 6.5 小结 196

第 7 章 Spring MVC 的高级技术 199

- 7.1 Spring MVC 配置的替代方案 200
 - 7.1.1 自定义 DispatcherServlet 配置 200
 - 7.1.2 添加其他的 Servlet 和 Filter 201
 - 7.1.3 在 web.xml 中声明 DispatcherServlet 203
- 7.2 处理 multipart 形式的
数据 206
 - 7.2.1 配置 multipart 解析器 207
 - 7.2.2 处理 multipart 请求 210

- 7.3 处理异常 214
 - 7.3.1 将异常映射为 HTTP 状态码 215
 - 7.3.2 编写异常处理的方法 216
- 7.4 为控制器添加通知 218
- 7.5 跨重定向请求传递数据 219
 - 7.5.1 通过 URL 模板进行重定向 220
 - 7.5.2 使用 flash 属性 221
- 7.6 小结 223

8 第 8 章 使用 Spring Web Flow 225

- 8.1 在 Spring 中配置 Web Flow 226
 - 8.1.1 装配流程执行器 226
 - 8.1.2 配置流程注册表 226
 - 8.1.3 处理流程请求 227
- 8.2 流程的组件 228
 - 8.2.1 状态 228
 - 8.2.2 转移 232
 - 8.2.3 流程数据 233
- 8.3 组合起来：披萨流程 234
 - 8.3.1 定义基本流程 235
 - 8.3.2 收集顾客信息 238
 - 8.3.3 构建订单 244
 - 8.3.4 支付 246
- 8.4 保护 Web 流程 248
- 8.5 小结 248

9 第 9 章 保护 Web 应用 251

第 3 部分 后端中的 Spring

10 第 10 章 通过 Spring 和 JDBC 征服数据库 287

- 10.1 Spring 的数据访问哲学 288
 - 10.1.1 了解 Spring 的数据访问异常体系 289
 - 10.1.2 数据访问模板化 291
- 10.2 配置数据源 293
 - 10.2.1 使用 JNDI 数据源 294
 - 10.2.2 使用数据源连接池 294

- 9.1 Spring Security 简介 252
 - 9.1.1 理解 Spring Security 的模块 252
 - 9.1.2 过滤 Web 请求 253
 - 9.1.3 编写简单的安全性配置 254
- 9.2 选择查询用户详细信息的服务 257
 - 9.2.1 使用基于内存的用户存储 257
 - 9.2.2 基于数据库表进行认证 259
 - 9.2.3 基于 LDAP 进行认证 261
 - 9.2.4 配置自定义的用户服务 265
- 9.3 拦截请求 267
 - 9.3.1 使用 Spring 表达式进行安全保护 269
 - 9.3.2 强制通道的安全性 271
 - 9.3.3 防止跨站请求伪造 272
- 9.4 认证用户 273
 - 9.4.1 添加自定义的登录页 274
 - 9.4.2 启用 HTTP Basic 认证 276
 - 9.4.3 启用 Remember-me 功能 276
 - 9.4.4 退出 277
- 9.5 保护视图 278
 - 9.5.1 使用 Spring Security 的 JSP 标签库 278
 - 9.5.2 使用 Thymeleaf 的 Spring Security 方言 282
- 9.6 小结 283

- 10.2.3 基于 JDBC 驱动的数据源 296
- 10.2.4 使用嵌入式的数据源 297
- 10.2.5 使用 profile 选择数据源 298

- 10.3 在 Spring 中使用 JDBC 300
 - 10.3.1 应对失控的 JDBC 代码 301
 - 10.3.2 使用 JDBC 模板 304
- 10.4 小结 309

11 第 11 章 使用对象-关系映射持久化数据 311

- 11.1 在 Spring 中集成 Hibernate 312
 - 11.1.1 声明 Hibernate 的 Session 工厂 313
 - 11.1.2 构建不依赖于 Spring 的 Hibernate 代码 315
- 11.2 Spring 与 Java 持久化 API 317
 - 11.2.1 配置实体管理器工厂 317
 - 11.2.2 编写基于 JPA 的 Repository 322
- 11.3 借助 Spring Data 实现自动化的 JPA Repository 324
 - 11.3.1 定义查询方法 327
 - 11.3.2 声明自定义查询 330
 - 11.3.3 混合自定义的功能 331
- 11.4 小结 332

12 第 12 章 使用 NoSQL 数据库 335

- 12.1 使用 MongoDB 持久化文档数据 336
 - 12.1.1 启用 MongoDB 337
 - 12.1.2 为模型添加注解, 实现 MongoDB 持久化 340
 - 12.1.3 使用 MongoTemplate 访问 MongoDB 343
 - 12.1.4 编写 MongoDB Repository 344
- 12.2 使用 Neo4j 操作图数据 349
 - 12.2.1 配置 Spring Data Neo4j 349
 - 12.2.2 使用注解标注图

实体 352

- 12.2.3 使用 Neo4jTemplate 355
- 12.2.4 创建自动化的 Neo4j Repository 356

12.3 使用 Redis 操作 key-value 数据 361

- 12.3.1 连接到 Redis 361
- 12.3.2 使用 RedisTemplate 362
- 12.3.3 使用 key 和 value 的序列化器 366

12.4 小结 367

13 第 13 章 缓存数据 369

- 13.1 启用对缓存的支持 370
 - 13.1.1 配置缓存管理器 371
- 13.2 为方法添加注解以支持缓存 375
 - 13.2.1 填充缓存 376
 - 13.2.2 移除缓存条目 380
- 13.3 使用 XML 声明缓存 381
- 13.4 小结 385

14 第 14 章 保护方法应用 387

- 14.1 使用注解保护方法 388
 - 14.1.1 使用 @Secured 注解限制方法调用 388
 - 14.1.2 在 Spring Security 中使用 JSR-250 的 @RolesAllowed 注解 389
- 14.2 使用表达式实现方法级别的安全性 390
 - 14.2.1 表述方法访问规则 391
 - 14.2.2 过滤方法的输入和输出 393
- 14.3 小结 397

第 4 部分 Spring 集成

15 第 15 章 使用远程服务 401

- 15.1 Spring 远程调用概览 402
- 15.2 使用 RMI 404

15.2.1 导出 RMI 服务 405

15.2.2 装配 RMI 服务 407

- 15.3 使用 Hessian 和 Burlap 发布远程服务 409