

Spring Boot 2 + Thymeleaf

企业应用实战

杨恩雄 编著

疯狂源自梦想
技术成就辉煌

Spring Boot 2 + Thymeleaf 企业应用实战

杨恩雄 编著



电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

Java 经过多年的发展，市面上出现的开发框架多不胜数，技术发展日新月异。每一次技术的更替都牵动着广大 Java 程序员的心。

在本书的前面部分，讲解目前市面上流行的 Spring Boot 框架，使用该框架我们可以轻松构建出各种项目。在此基础上，介绍 Spring MVC 的使用。学习完这些知识后，读者可以掌握 Spring Boot、Spring MVC 的核心内容。在本书的第二部分，会深入讲解 Thymeleaf 框架，这是一个较为新颖的模板引擎。使用 Thymeleaf，可以完全抛弃 JSP，进入一种更为简便的页面开发模式。第 11 章讲解目前市面上常见的几个 JavaScript 框架，例如 jQuery、Bootstrap 及 Vue.js。学习完这些知识后，读者可以使用这些框架开发大部分的页面功能。第 12 章介绍如何使用 Spring Data 框架操作各个主流数据库，例如 MySQL、Redis 和 MongoDB 等。第 13 章介绍如何使用 Spring Cloud 中的主要框架，学完本章知识后，读者可以开发最基本的微服务项目。本书的最后一章，以一个案例对所学知识进行运用。

本书定位为了一本 Java Web 开发的入门级书籍，从表现层到持久层，力求为大家提供一整套 Java Web 开发的技术教程。

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

Spring Boot 2+Thymeleaf 企业应用实战 / 杨恩雄编著. —北京：电子工业出版社，2018.9
ISBN 978-7-121-34924-9

I. ①S… II. ①杨… III. ①JAVA 语言—程序设计 IV. ①TP312.8

中国版本图书馆 CIP 数据核字 (2018) 第 196828 号

策划编辑：张月萍

责任编辑：牛 勇

印 刷：三河市华成印务有限公司

装 订：三河市华成印务有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱

邮编：100036

开 本：787×1092 1/16 印张：19.75

字数：530 千字

版 次：2018 年 9 月第 1 版

印 次：2018 年 9 月第 1 次印刷

定 价：69.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：(010) 88254888, 88258888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式：010-51260888-819, faq@phei.com.cn。



前 言

Java 经过多年的发展，市面上出现的开发框架多不胜数，技术发展日新月异。程序员群体是一个求知若渴的群体，也是一个容易接受新知识的群体，为了学习新技术，多少人食不知味、夜不能寐。笔者有幸成为一名平凡的程序员，从业十余年，面对各种新技术，时常怀着一颗谦卑的心，不停前进，不断学习。

笔者从 2006 年开始进行 Java Web 开发，使用过大部分主流的 Java 技术，例如 EJB、Servlet、MVC 框架等，在此过程中开发过多个大型项目，经历了这些项目从无到有的过程，深知系统开发之不易。一直以来，期望能使用最简单的方式开发出最完美的系统。近年来随着 Spring Boot 的兴起，Java 的 Web 开发趋向简单，笔者希望能将 Spring Boot 等技术进行推广，从而降低 Java Web 开发的门槛，因此编纂本书。笔者学识浅薄，加之仓促付梓，书中错漏定然难免，望读者指正。

本书能得以出版，得益于多方襄助，对他们的感激之情，难以言表。感谢传道并解惑的恩师，感谢聪颖而好学的读者，感谢善良和亲爱的家人，笔者会谢意永存、铭感不忘。

本书特点

笔者长期工作于企业的 IT 部门，有着丰富的企业应用开发经验，因此本书有以下特点。

1. 涵盖多个技术框架

笔者期望本书能给读者提供一整套的企业应用开发方案，因此本书介绍了目前最为流行的 Spring Boot、Thymeleaf、Spring Data 以及多个 JavaScript 框架。学习完本书知识后，读者掌握的并不是一两个技术框架，而是一整套当前非常流行的开发技术。

2. 内容深入

本书以 Spring Boot 与 Thymeleaf 为核心，其中 Thymeleaf 是目前非常流行的模板框架，本书对该框架进行了系统讲解，内容由浅入深，务求使读者充分掌握这个模板框架。

3. 案例贴近实际，视频带领入门

笔者长期供职于 IT 企业，奋斗于研发一线，因此本书的各个知识点对应的案例更加贴近实际。看得懂、学得会、做得出是笔者一贯的风格，对于初次接触这些知识的读者，本书提供了配套的视频，助力你更快地掌握知识。

4. 注释详细

简单易懂是笔者一直以来坚持的目标，本书中的几乎每行 Java 代码均有注释，读者可以很容易地了解代码的意思，轻松掌握知识。

本书写给谁看

笔者一直以来希望能将复杂的技术简单化，因此将本书定位成一本入门级的 Java Web 开发书籍，只要你有一定的 Java 语言基础，就可以在本书的基础上，学习使用 Spring Boot+Thymeleaf

进行 Java Web 应用开发。

衷心感谢

首先非常感谢李刚老师，一直以来，他既是我的老师，也是我的技术后盾，非常幸运人生能有这样一位良师益友。

其次感谢出版社编辑，为本书提出了许多宝贵的意见。

最后感谢我的家人，你们是我前进的动力。

下载本书代码与联系作者

本书的所有代码、软件，均可以到以下网盘获取：<http://pan.baidu.com/s/1sl5FWK1>。读者也可以扫描以下的二维码，关注“疯狂图书”公众号获取本书资料。公众号内除了提供本书代码外，还提供了相关视频供读者下载。



如在学习过程中有技术相关的问题，可以发邮件到笔者的邮箱：yangenxiong@163.com。

2018年7月22日于广州

目 录 CONTENTS

第 1 章 概述	1	2.4 使用 Feign 调用服务	18
1.1 Java EE 开发技术	2	2.5 本章小结	19
1.1.1 Java EE 的三层架构	2	第 3 章 Spring Boot 配置	20
1.1.2 MVC 框架	2	3.1 配置文件	21
1.1.3 数据访问层框架	3	3.1.1 默认配置文件	21
1.1.4 视图技术	3	3.1.2 指定配置文件	22
1.2 JDK 的下载与安装	3	3.1.3 yml 文件	22
1.3 安装与配置 Maven	4	3.1.4 使用 profile 指定配置	23
1.3.1 关于 Maven	4	3.2 Web 配置	24
1.3.2 下载与安装 Maven	4	3.2.1 Servlet 配置	24
1.3.3 配置远程仓库	5	3.2.2 Listener 配置	25
1.4 安装 Eclipse	5	3.2.3 Filter 配置	26
1.4.1 Eclipse 版本	5	3.2.4 构建可部署的 war 包	26
1.4.2 在 Eclipse 中配置 Maven	5	3.2.5 JSP 配置	28
1.4.3 获取本书代码	6	3.3 其他配置	30
1.5 本章小结	6	3.3.1 服务器常用配置	30
第 2 章 初试 Spring Boot	7	3.3.2 响应压缩	30
2.1 Spring Boot 介绍	8	3.3.3 SSL 配置	32
2.1.1 Spring Boot 简介	8	3.3.4 使用其他服务器	32
2.1.2 starter 模块	8	3.3.5 服务器访问日志	33
2.2 构建第一个 Spring Boot 程序	9	3.3.6 banner 配置	34
2.2.1 新建 Maven 项目	9	3.4 自定义配置	35
2.2.2 编写启动类	10	3.4.1 @Value 注解	35
2.2.3 编写控制器	11	3.4.2 定义属性映射对象	36
2.2.4 开发环境的热部署	12	3.4.3 使用配置	37
2.3 运行单元测试	12	3.4.4 属性自动识别	38
2.3.1 测试 Web 服务	12	3.5 本章小结	39
2.3.2 模拟 Web 测试	13	第 4 章 Spring Boot 的注解	40
2.3.3 测试业务组件	13	4.1 Spring 的常用注解	41
2.3.4 模拟业务组件	14	4.1.1 bean 定义	41
2.4 发布与调用 REST 服务	15	4.1.2 依赖注入	42
2.4.1 REST	15	4.1.3 使用 Primary 注解	44
2.4.2 发布 REST 服务	15	4.1.4 Scope 注解	45
2.4.3 使用 RestTemplate 调用服务	16		

4.1.5	方法注入	46	5.3.3	在 Servlet 中使用 Thymeleaf	72
4.1.6	AOP 注解	47	5.3.4	编写模板文件	73
4.1.7	ComponentScan 注解	49	5.4	在 Struts2 中整合 Thymeleaf	74
4.2	高级 Spring 注解	50	5.4.1	建立与配置 Struts2 项目	74
4.2.1	限定注解	50	5.4.2	配置 Struts Action	75
4.2.2	自定义限定注解	51	5.4.3	整合 Thymeleaf	76
4.2.3	自定义 bean 的生命周期	52	5.5	在 Spring Boot 中整合 Thymeleaf	77
4.3	Spring MVC 的常用注解	54	5.5.1	建立与配置项目	77
4.3.1	Controller 注解	54	5.5.2	编写控制器与模板	78
4.3.2	RequestMapping 注解	55	5.5.3	静态资源	79
4.3.3	PathVariable 注解	56	5.6	本章小结	79
4.3.4	MatrixVariable 注解	56			
4.3.5	RequestParam 注解	57	第 6 章	Thymeleaf 对象的使用	80
4.3.6	文件上传	58	6.1	文本处理	81
4.3.7	RestController 注解	58	6.1.1	外部文本输出	81
4.4	Spring Boot 的条件注解	59	6.1.2	文本国际化	82
4.4.1	类条件注解	59	6.1.3	处理转义文本	83
4.4.2	项目依赖问题	60	6.1.4	读取变量	83
4.4.3	其他条件注解	61	6.1.5	在文本信息中使用变量	84
4.4.4	自定义条件注解	61	6.1.6	其他读取变量的方式	84
4.4.5	Spring Boot 的自动配置	62	6.1.7	链接表达式	85
4.5	本章小结	63	6.2	基本对象	86
第 5 章	初试 Thymeleaf	64	6.2.1	基本对象概述	86
5.1	Thymeleaf	65	6.2.2	#ctx 对象	87
5.1.1	什么是 Thymeleaf	65	6.2.3	Web 对象	87
5.1.2	支持的模板类型	65	6.2.4	内置变量	88
5.1.3	HTML 模板	65	6.2.5	使用基本对象的注意事项	88
5.1.4	XML 模板	66	6.3	数字对象	89
5.2	初试 Thymeleaf API	66	6.3.1	整数格式化	89
5.2.1	处理 HTML	66	6.3.2	小数格式化	90
5.2.2	更换模板解析器	67	6.3.3	货币格式化	91
5.2.3	处理资源文件	67	6.3.4	百分比格式化	92
5.2.4	变量处理	68	6.3.5	使用 sequence 方法	92
5.2.5	遍历集合	69	6.4	字符串对象	92
5.2.6	设置前缀与后缀	69	6.4.1	toString 与 length 方法	93
5.3	在 Servlet 应用中整合 Thymeleaf	70	6.4.2	非空判断与默认值处理	93
5.3.1	建立项目	70	6.4.3	包含判断	94
5.3.2	配置模板引擎	71	6.4.4	截取与替换	94
			6.4.5	追加与拼接	95

6.4.6	分割与连接	96	7.4.2	switch case 语法	117
6.4.7	大小写转换	96	7.4.3	使用 th:each 属性	117
6.4.8	字符串编码与解码	97	7.4.4	常见迭代类型	118
6.4.9	其他处理	98	7.4.5	迭代状态对象	118
6.5	日期对象	98	7.4.6	数据延迟加载	120
6.5.1	格式化日期	98	7.4.7	在 Spring Boot 中使用延迟加载	121
6.5.2	获取日期字段	99	7.4.8	星号表达式	121
6.5.3	创建日期	100	7.5	本章小结	122
6.6	数组与集合对象	101	第 8 章 深入 Thymeleaf 模板	123	
6.6.1	数组	101	8.1	模板片断	124
6.6.2	List 与 Set	101	8.1.1	定义与引用模板片断	124
6.6.3	Map	101	8.1.2	片断引用语法	125
6.7	其他内置对象	102	8.1.3	选择器的基础语法	126
6.7.1	messages 对象	102	8.1.4	选择器的其他语法	128
6.7.2	uris 对象	102	8.1.5	insert、replace 与 include 的区别	128
6.7.3	aggregates 对象	103	8.1.6	含有变量的片断引用	129
6.8	本章小结	103	8.1.7	片断块引用	130
第 7 章 Thymeleaf 常用语法	104		8.1.8	无操作符的使用	131
7.1	表达式语法之运算符	105	8.1.9	删除模板	132
7.1.1	表达式常量	105	8.1.10	基本表达式总结	134
7.1.2	字符串拼接	106	8.2	模板与逻辑分离	134
7.1.3	算术运算符	106	8.2.1	逻辑分离例子	134
7.1.4	关系运算符	107	8.2.2	逻辑分离配置	136
7.1.5	条件运算符	108	8.2.3	关关节点的选择	137
7.1.6	无操作符	108	8.3	模板注释	137
7.2	表达式语法进阶	108	8.3.1	解析层注释	138
7.2.1	数据转换	108	8.3.2	原型注释	139
7.2.2	自定义数据转换类	109	8.4	内联语法	139
7.2.3	Spring Boot 数据转换	110	8.4.1	使用内联语法显示文本	139
7.2.4	表达式预处理	112	8.4.2	在 JavaScript 中使用内联语法	140
7.2.5	表达式调用工具方法	113	8.4.3	在 CSS 中使用内联语法	140
7.3	属性设置	114	8.4.4	内联语法注释	141
7.3.1	使用 th:attr	114	8.4.5	内联语法序列化	142
7.3.2	设置 HTML 节点属性	114	8.5	模板缓存	143
7.3.3	属性值拼接	115	8.5.1	开启与配置缓存	143
7.3.4	HTML5 属性支持	115	8.5.2	缓存有效时间	145
7.4	条件判断与数据迭代	116	8.6	本章小结	146
7.4.1	条件判断	116			

第 9 章 Thymeleaf 原理与扩展	147	10.2.3 HTML 节点的 id 生成	190
9.1 概念与接口介绍	148	10.2.4 实现 label 节点	192
9.1.1 方言	148	10.2.5 在模板中处理 checkbox 与 radio ...	193
9.1.2 处理器	149	10.2.6 表单提交处理 checkbox 与 radio ...	195
9.1.3 模板处理者	150	10.3 表单验证	195
9.1.4 模板解析事件	150	10.3.1 表单验证	195
9.1.5 节点与标签	151	10.3.2 修改信息文件	197
9.2 深入 Thymeleaf 模板处理	151	10.3.3 输出错误信息	197
9.2.1 策略模式	151	10.4 片断的使用	198
9.2.2 责任链模式	153	10.4.1 引用片断	198
9.2.3 合并使用策略模式与责任链模式 ...	156	10.4.2 使用 bean 选择片断	199
9.2.4 模板处理过程	159	10.4.3 在控制器中引用片断	201
9.2.5 模板事件	161	10.5 自定义标签	201
9.3 处理器	161	10.5.1 方言与处理器	201
9.3.1 预处理器	161	10.5.2 配置方言	203
9.3.2 后处理器	163	10.6 本章小结	204
9.3.3 一般处理器	164		
9.3.4 处理器优先级	165	第 11 章 使用 JavaScript 库	205
9.3.5 标签处理器	167	11.1 使用 jQuery	206
9.3.6 模型处理器	168	11.1.1 概述	206
9.3.7 边界处理器	170	11.1.2 选择器	206
9.3.8 其他处理器	170	11.1.3 事件	207
9.4 Thymeleaf 扩展	171	11.1.4 数据列表	208
9.4.1 自定义标签	171	11.1.5 AJAX 调用 GET 方法	209
9.4.2 自定义模板属性	173	11.1.6 AJAX 调用 POST 方法	210
9.4.3 自定义内置对象	175	11.1.7 表单验证	211
9.4.4 自定义执行属性	176	11.2 Bootstrap	213
9.5 本章小结	177	11.2.1 概述	213
第 10 章 Spring Boot 与 Thymeleaf 整合	178	11.2.2 数据列表	214
10.1 在 Spring Boot 中配置 Thymeleaf	179	11.2.3 分页	215
10.1.1 模板引擎配置	179	11.2.4 表单	218
10.1.2 Spring MVC 视图解析器	181	11.2.5 警告框的使用	219
10.1.3 Thymeleaf 视图解析器	184	11.3 Vue.js	221
10.1.4 数据转换配置	186	11.3.1 概述	221
10.1.5 模板国际化	186	11.3.2 组件	222
10.2 表单处理	188	11.3.3 指令	224
10.2.1 表单提交	188	11.3.4 循环指令	225
10.2.2 使用 th:field 属性	189	11.3.5 表单验证	226
		11.4 本章小结	227

第 12 章 数据库实战.....	228	13.2.4 服务调用者	255
12.1 概述	229	13.2.5 程序结构	257
12.1.1 关于 Spring Data	229	13.3 Ribbon 负载均衡	257
12.1.2 Spring Data 的功能	229	13.3.1 Ribbon 介绍	258
12.1.3 Spring Data 模块	230	13.3.2 第一个 Ribbon 程序	258
12.2 Spring Data 与 JPA	230	13.3.3 在 Spring Cloud 中测试使用 Ribbon	259
12.2.1 构建项目	230	13.4 客户端 Feign	260
12.2.2 数据访问层与业务层	231	13.4.1 Feign 介绍	260
12.2.3 自定义数据存储逻辑	233	13.4.2 第一个 Feign 程序	260
12.2.4 方法名查询	233	13.4.3 使用 Feign 调用微服务	262
12.2.5 使用 @Query 注解	234	13.5 容错框架 Hystrix	263
12.3 Spring Data 与 MongoDB	235	13.5.1 Hystrix 介绍	263
12.3.1 安装 MongoDB	235	13.5.2 第一个 Hystrix 程序	265
12.3.2 配置权限	236	13.5.3 在 Spring Cloud 中整合 Hystrix	266
12.3.3 MongoDB 的相关概念	236	13.6 本章小结	268
12.3.4 构建项目	236	第 14 章 实战案例	269
12.3.5 数据访问层与业务层	237	14.1 系统功能及界面	270
12.3.6 自定义数据存储逻辑	238	14.1.1 登录与首页	270
12.3.7 方法名查询	239	14.1.2 图书管理模块	271
12.3.8 使用 @Query 注解	240	14.1.3 入库模块	272
12.4 Spring Data 与 Redis	241	14.1.4 销售模块	273
12.4.1 Redis 的安装与配置	241	14.1.5 导航组件与分页组件	274
12.4.2 Redis 的数据类型	242	14.1.6 运行本章案例	275
12.4.3 使用 Jedis	242	14.2 数据库设计	275
12.4.4 构建 Spring Data 项目	243	14.2.1 用户表	275
12.4.5 数据访问层与业务层	244	14.2.2 图书表与库存表	275
12.4.6 自定义数据存储逻辑	246	14.2.3 入库单数据表	276
12.4.7 方法名查询	247	14.2.4 销售单数据表	276
12.5 本章小结	247	14.3 搭建项目框架	277
第 13 章 开发微服务	248	14.3.1 项目依赖与项目结构	277
13.1 微服务与 Spring Cloud	249	14.3.2 项目配置	278
13.1.1 微服务简介	249	14.3.3 实体设计	279
13.1.2 Netflix 与 Spring Cloud	249	14.3.4 用户登录	281
13.1.3 Spring Cloud 模块介绍	250	14.3.5 片断定义	282
13.2 Eureka 服务治理	250	14.4 图书管理	283
13.2.1 Eureka 架构	250	14.4.1 数据列表	283
13.2.2 构建服务器	251	14.4.2 通用的分页组件	285
13.2.3 服务提供者	253		

14.4.3	图片上传	286	14.5.4	新建销售单	297
14.4.4	新建图书	288	14.5.5	操作库存	299
14.4.5	查看与修改图书	289	14.6	首页数据查询与排序	300
14.4.6	删除图书	290	14.6.1	库存查询	300
14.5	销售单模块	291	14.6.2	图书查询与轮播	301
14.5.1	列表与分页功能	291	14.6.3	数据排序	303
14.5.2	选择图书	293	14.7	本章小结	304
14.5.3	删除已选图书与修改数量	295			

第 1 章

概 述

本章要点

- ✎ Java EE 开发技术介绍
- ✎ Eclipse
- ✎ 构建工具 Maven

Java 经过多年的发展,市面上的开发框架多不胜数,技术发展日新月异。本书将会以 Spring Boot、Thymeleaf 与 Spring Data 整合开发为主线,讲述一套全新的 Java EE 开发技术。

本章作为第一章,会首先对 Java EE 的开发技术做一个简单的概述。为了避免读者在学习知识的过程中出现障碍,本章会搭建开发环境,介绍本书所使用的开发工具,如果读者已经掌握了这些工具,则可跳过本章,直接学习后面的知识。

1.1 Java EE 开发技术

首先来了解目前 Java EE 领域一些常用的开发框架。

1.1.1 Java EE 的三层架构

经典的 Java EE 应用是三层架构,分别为表现层、业务逻辑层和数据访问层。这种应用的划分方式,可以让项目的参与人员发挥各自的专长,更加专注于特定的开发工作。经典的 Java EE 三层架构如图 1-1 所示。

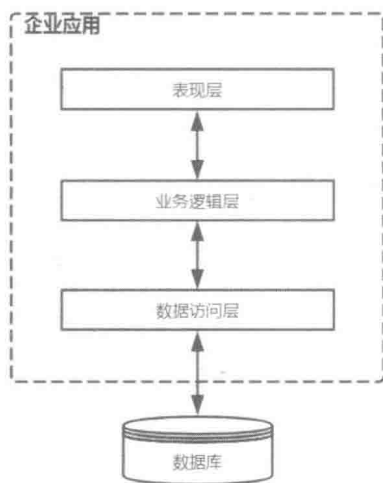


图 1-1 Java EE 应用三层架构

在过去的十年间,这个经典的架构被应用于各种企业级项目中,深得开发者的青睐,精通三层开发技术几乎成为 Java 程序员的标配。这么多年来,基于这个架构,涌现出了不少优秀的开发框架,下面将介绍目前市面上较为流行的开发框架。

1.1.2 MVC 框架

说起 Servlet 框架,不得不说最经典的 Struts。它是一个基于 Servlet/JSP 的 MVC 框架,采用了经典的 MVC 模式。它的诞生,大大减轻了 Web 开发的工作量,一度成为 Java Web 领域中最流行的框架。但是由于 Struts 的部分缺陷(例如过于依赖 Servlet),造成很多开发人员转去使用其他的 MVC 框架(例如 WebWork、Tapestry)。Struts 在发展的过程中,结合了 WebWork 框架,从而推出了全新的 Struts2。实际上,Struts2 已经抛弃了 Struts1 的核心,完全采用 WebWork 的核心设计。目前 Struts2 仍然是主流的 MVC 框架之一。

除了 Struts 之外,Java EE 领域另一个佼佼者 Spring,也推出了自己的 MVC 框架:Spring Web MVC。这是一个基于 Servlet 构建的 Web 框架,刚推出时虽然并没有得到广泛的应用,但

随着它不断的发展,现在俨然成为 MVC 框架的首选。Spring 在刚开始,只负责提供 IoC 容器,用于整合、管理企业应用中的各个组件。之前可能有朋友将 Spring 划分到业务逻辑层中,认为它是一个业务逻辑层的框架,但随着 Spring 涉猎的领域越来越广,其不仅推出了 Spring MVC,还推出了数据访问层框架 Spring Data。Spring Data 的作用更像是一个数据访问层的适配器。这样来看, Spring 大有“一统江湖”之势。本书后面的章节,也会涉及 Spring Data 的相关知识。

以上简单介绍了 Struts 和 Spring MVC,除了它们之外,还有诸如 Tapestry、JSF 等 MVC 框架,在此就不展开讨论了。由于本书以 Spring Boot 为核心,因此本书会使用 Spring MVC 框架。

1.1.3 数据访问层框架

数据访问层主要对数据进行增、删、改、查。最初人们使用 JDBC 的 API 进行数据库操作,后来,著名程序员 Gavin King 创建了 Hibernate 框架,该框架对 JDBC 进行了封装,让 Java 对象与数据库表建立映射关系,操作对象等于操作数据,这使得 Hibernate 迅速成为数据访问层的热门框架。除了 Hibernate 之外,另一个开源框架 MyBatis 也是业界的宠儿,一旦涉及关系型数据库的项目,大家首先想到的就是这两个开源框架。随着 NoSQL 数据库的兴起,目前在数据访问层,我们拥有更多的技术选项。Spring 针对数据访问层,提供了 Spring Data 框架,这等于在各个数据访问框架之上,增加了一个适配层,其目的是使用同样的模型、代码来实现不同数据库操作。本书将主要讲解 Spring Data 框架。

1.1.4 视图技术

在传统的 Java Web 开发中,多使用 JSP 视图,但也有人使用 Freemarker、Velocity 等模板技术。Spring MVC 支持多种模板技术,例如 Freemarker、Thymeleaf、Groovy Markup 等。Freemarker 作为一个老牌的模板引擎,目前市面上已有非常多的文档可以参考。相对于 Freemarker 来说, Thymeleaf 较为新颖,本书将以 Thymeleaf 为核心,深入讲解 Thymeleaf 技术。

本节主要向大家简单介绍了 Java EE 三层架构及相应的主流技术,再次申明,本书将会以 Spring Boot、Thymeleaf、Spring Data 整合为主线,讲解全新的 Java EE 开发技术。下面开始搭建本书的开发环境,如果读者已经掌握相关的工具,可直接跳过下面的内容。

1.2 JDK 的下载与安装

在本书中,使用的操作系统为 64 位的 Windows 7,使用 64 位的 JDK10。读者可以自行到 Oracle 官方网站下载 JDK,也可以访问 <http://pan.baidu.com/s/1s15FWK1> 下的 soft 目录。JDK 安装文件名称为 jdk-10.0.1_windows-x64_bin.exe。安装 JDK 后,添加 Windows 环境变量,如图 1-2 所示。

修改 Path 变量,在变量值后追加“%JAVA_HOME%\bin”,如图 1-3 所示。



图 1-2 JAVA_HOME

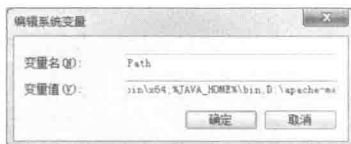


图 1-3 配置 Path

配置完成后，在命令行窗口输入 `java -version`，输出如图 1-4 所示。



图 1-4 查看 Java 版本

1.3 安装与配置 Maven

1.3.1 关于 Maven

Maven 是 Apache 下的一个开源项目，用于项目的构建。使用 Maven 可以对项目的依赖包进行管理，它也支持构建脚本的继承。对于一些模块（子项目）较多的项目来说，Maven 是更好的选择，子项目可以继承父项目的构建脚本，从而减少了构建脚本的冗余。

除此之外，Maven 本身的插件机制也使其更加强大和灵活，使用者可以配置各种 Maven 插件来完成自己的事情。如果感觉官方或者第三方提供的 Maven 插件不够用，还可以编写符合自己要求的 Maven 插件。Maven 为使用者提供了一个统一的依赖仓库，各种开源项目的发布包都可以在上面找到。在一间公司或者一个项目组内部，甚至可以搭建私有的 Maven 仓库，将自己项目的包放到私有仓库中，供其他项目组或者开发者使用。

在 Maven 的众多特性中，最为重要的是它对依赖包的管理，Maven 将项目所使用的依赖包的信息放到 `pom.xml` 的 `dependencies` 节点。例如我们要使用 `spring-core` 模块的 jar 包，只需在 `pom.xml` 中配置该模块的依赖信息，Maven 会自动将 `spring-beans` 等模块引入我们项目的环境变量中。

1.3.2 下载与安装 Maven

本书所使用的 Maven 版本为 3.5，可以到 Maven 官方网站下载：<http://maven.apache.org/>，也可以到本书的 `soft` 目录下，找到文件名为 `apache-maven-3.5.0-bin.zip` 的包。下载并解压后得到 `apache-maven-3.5.0` 目录，将主目录下的 `bin` 目录加入到系统的环境变量中，如图 1-5 所示。

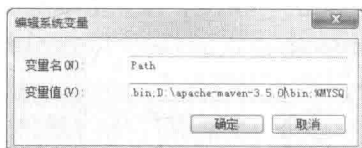


图 1-5 修改环境变量

配置完成后，打开 `cmd` 命令行，输入“`mvn -v`”，稍后可以看到输出的 Maven 版本信息。

Maven 下载的依赖包会被存放在本地仓库中，默认路径为 C:\Users\用户名\.m2\repository。

1.3.3 配置远程仓库

如果不进行仓库配置，在默认情况下，会到 Apache 官方的仓库下载依赖包。由于 Apache 官方的仓库位于国外，所以下载速度较慢，这会降低开发效率，笔者建议使用国内的 Maven 仓库或者搭建自己的仓库。本书的重点不是 Maven，因此直接使用了由阿里云提供的 Maven 仓库。修改 apache-maven-3.5.0/conf 目录下的 setting.xml，在 mirror 节点下加入以下配置：

```
<mirror>
  <id>alimaven</id>
  <name>aliyun maven</name>
  <url>http://maven.aliyun.com/nexus/content/groups/public/</url>
  <mirrorOf>central</mirrorOf>
</mirror>
```

配置完后，以后在使用过程中，Maven 会先到阿里云的仓库下载依赖包。

1.4 安装 Eclipse

1.4.1 Eclipse 版本

本书使用的 Eclipse 版本为 Oxygen (4.7)，大家可以从以下的地址得到该版本的 Eclipse：<http://www.eclipse.org/downloads/packages/eclipse-ide-java-developers/oxygen3a>，也可以在上一节提到的 soft 目录下找到该版本的 Eclipse，文件名为 eclipse-java-oxygen-3a-win32-x86_64.zip。

1.4.2 在 Eclipse 中配置 Maven

Luna 版本的 Eclipse 自带了 Maven 插件，默认使用的是 Maven 3.2，由于我们前面安装的是 Maven 3.5 版本，因此需要在 Eclipse 中指定 Maven 版本以及配置文件。按照如图 1-6 所示指定 Maven 的版本，按照如图 1-7 所示指定配置文件。

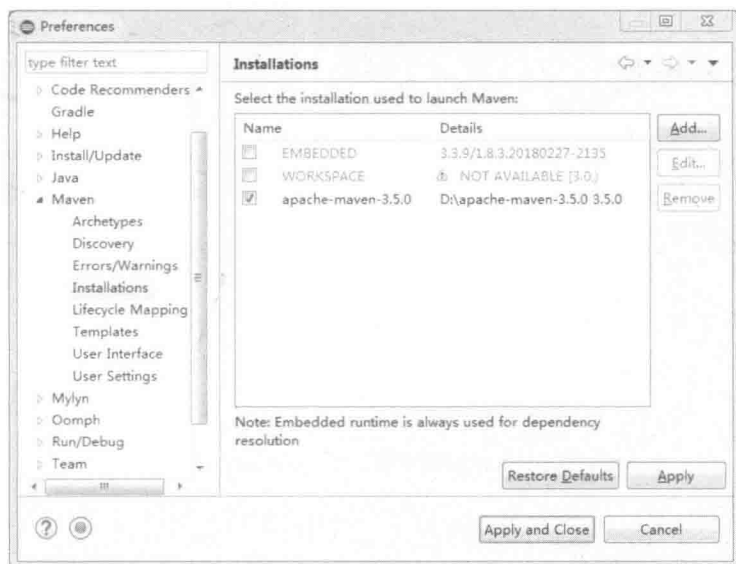


图 1-6 在 Eclipse 中指定 Maven 的版本

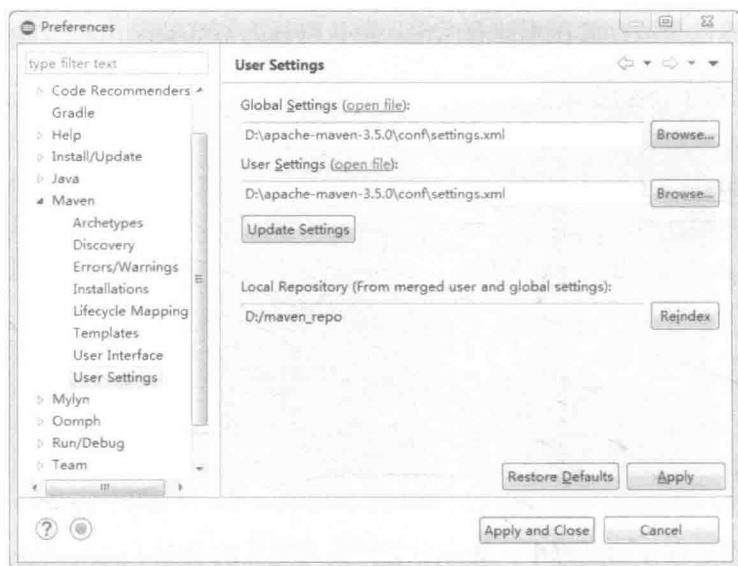


图 1-7 指定 Maven 的配置文件

注意：本书的案例，如无特别说明均以 Maven 项目的形式导入。

1.4.3 获取本书代码

本书的所有代码、软件，均可以到以下网盘获取：<http://pan.baidu.com/s/1sl5FWK1>。读者也可以扫描以下的二维码，关注“疯狂图书”公众号获取，公众号内除了包含本书代码外，还有相关的配套视频供读者下载。



1.5 本章小结

本章主要介绍如何搭建本书使用的开发环境，包括 JDK、Eclipse 与 Maven 的安装。环境的搭建较为简单，熟悉的读者可直接跳过。除了开发环境之外，这一章还介绍了传统 Java EE 的三层架构及对应的主流框架。本书将会以 Spring Boot、Thymeleaf、Spring Data 整合开发为主线，向大家介绍一套全新的 Java EE 开发技术。下一章我们介绍 Spring Boot。