第1章 Spring Boot来临　1

1.1　Spring的历史　1

1.2　注解还是XML　2

1.3　Spring Boot的优点　3

1.4　传统Spring MVC和Spring Boot的对比　4

第　2章 聊聊开发环境搭建和基本开发　10

2.1　搭建Spring Boot开发环境　10

2.1.1　搭建Eclipse开发环境　10

2.1.2　搭建IntelliJ IDEA开发环境　13

2.2　Spring Boot的依赖和自动配置　15

2.3　使用自定义配置　19

2.4　开发自己的Spring Boot项目　21

第3章　全注解下的Spring IoC　23

3.1　IoC容器简介　23

3.2　装配你的Bean　27

3.2.1　通过扫描装配你的Bean　27

3.2.2　自定义第三方Bean　31

3.3　依赖注入　32

3.3.1　注解@Autowired　34

3.3.2　消除歧义性——@Primary和@Quelifier　35

3.3.3　带有参数的构造方法类的装配　36

3.4　生命周期　37

3.5　使用属性文件　42

3.6　条件装配Bean　45

3.7　Bean的作用域　46

3.8　使用@Profile　48

3.9　引入XML配置Bean　50

3.10　使用Spring EL　51

第4章　开始约定编程——Spring AOP　53

4.1　约定编程　53

4.1.1　约定　53

4.1.2　ProxyBean的实现　57

4.1.3　总结　60

4.2　AOP的概念　61

4.2.1　为什么使用AOP　61

4.2.2　AOP 术语和流程　64

4.3　AOP开发详解　65

4.3.1　确定连接点　65

4.3.2　开发切面　66

4.3.3　切点定义　67

4.3.4　测试AOP　68

4.3.5　环绕通知　71

4.3.6　引入　72

4.3.7　通知获取参数　74

4.3.8　织入　75

4.4　多个切面　77

第5章　访问数据库　82

5.1　配置数据源　83

5.1.1　启动默认数据源　83

5.1.2　配置自定义数据源　83

5.2　使用JdbcTemplate操作数据库　86

5.3　使用JPA（Hibernate）操作数据　90

5.3.1　概述　90

5.3.2　开发JPA　90

5.4　整合MyBatis框架　96

5.4.1　MyBatis简介　96

5.4.2　MyBatis的配置　97

5.4.3　Spring Boot整合MyBatis　101

5.4.4　MyBatis的其他配置　104

第6章　聊聊数据库事务处理　107

6.1　JDBC的数据库事务　108

6.2　Spring声明式事务的使用　110

6.2.1　Spring声明式数据库事务约定　110

6.2.2　@Transactional的配置项　111

6.2.3　Spring事务管理器　113

6.2.4　测试数据库事务　114

6.3　隔离级别　118

6.3.1　数据库事务的知识　118

6.3.2　详解隔离级别　120

6.4　传播行为　124

6.4.1　传播行为的定义　125

6.4.2　测试传播行为　126

6.5　@Transactional自调用失效问题　130

第7章　使用性能利器——Redis　134

7.1　spring-data-redis项目简介　135

7.1.1　spring-data-redis项目的设计　135

7.1.2　RedisTemplate　137

7.1.3　Spring对Redis数据类型操作的封装　139

7.1.4　SessionCallback和RedisCallback接口　141

7.2　在Spring Boot中配置和使用Redis　142

7.2.1　在Spring Boot中配置Redis　142

7.2.2　操作Redis数据类型　143

7.3　Redis的一些特殊用法　148

7.3.1　使用Redis事务　148

7.3.2　使用Redis流水线　149

7.3.3　使用Redis发布订阅　150

7.3.4　使用Lua脚本　153

7.4　使用Spring缓存注解操作Redis　156

7.4.1　缓存管理器和缓存的启用　156

7.4.2　开发缓存注解　157

7.4.3　测试缓存注解　163

7.4.4　缓存注解自调用失效问题　165

7.4.5　缓存脏数据说明　165

7.4.6　自定义缓存管理器　166

第8章　文档数据库——MongoDB　168

8.1　配置MongoDB　169

8.2　使用MongoTemplate实例　170

8.2.1　搭建开发环境　170

8.2.2　使用MongoTemplate操作文档　173

8.3　使用JPA　178

8.3.1　基本用法　178

8.3.2　使用自定义查询　180

第9章　初识Spring MVC　183

9.1　Spring MVC框架的设计　183

9.2　Spring MVC流程　184

9.3　定制Spring MVC的初始化　191

9.4　Spring MVC实例　192

9.4.1　开发控制器　193

9.4.2　视图和视图渲染　194

第　10章 深入Spring MVC开发　197

10.1　处理器映射　197

10.2　获取控制器参数　198

10.2.1　在无注解下获取参数　199

10.2.2　使用@RequestParam获取参数　199

10.2.3　传递数组　200

10.2.4　传递JSON　200

10.2.5　通过URL传递参数　203

10.2.6　获取格式化参数　204

10.3　自定义参数转换规则　205

10.3.1　处理器获取参数逻辑　205

10.3.2　一对一转换器（Converter）　208

10.3.3　GenericConverter集合和数组转换　210

10.4　数据验证　211

10.4.1　JSR-303 验证　211

10.4.2　参数验证机制　214

10.5　数据模型　217

10.6　视图和视图解析器　219

10.6.1　视图设计　219

10.6.2　视图实例——导出PDF文件　220

10.7　文件上传　224

10.7.1　Spring MVC对文件上传的支持　224

10.7.2　开发文件上传功能　226

10.8　拦截器　228

10.8.1　拦截器的设计　228

10.8.2　开发拦截器　229

10.8.3　多个拦截器的顺序　231

10.9　国际化　234

10.9.1　国际化消息源　234

10.9.2　国际化解析器　235

10.9.3　国际化实例——SessionLocaleResolver　237

10.10　Spring MVC拾遗　240

10.10.1　@ResponseBody转换为JSON的秘密　240

10.10.2　重定向　241

10.10.3　操作会话对象　243

10.10.4　给控制器增加通知　245

10.10.5　获取请求头参数　247

第　11章 构建REST风格网站　249

11.1　REST简述　249

11.1.1　REST名词解释　249

11.1.2　HTTP的动作　250

11.1.3　REST风格的一些误区　251

11.2　使用Spring MVC开发REST风格端点　251

11.2.1　Spring MVC整合REST　252

11.2.2　使用Spring开发REST风格的端点　252

11.2.3　使用@RestController　260

11.2.4　渲染结果　261

11.2.5　处理HTTP状态码、异常和响应头　262

11.3　客户端请求RestTemplate　266

11.3.1　使用RestTemplate请求后端　267

11.3.2　获取响应头、状态码和资源交换　269

第　12章 安全——Spring Security　271

12.1　概述和简单安全认证　271

12.2　使用WebSecurityConfigurerAdapter自定义　273

12.3　自定义用户服务信息　274

12.3.1　使用内存签名服务　275

12.3.2　使用数据库定义用户认证服务　276

12.3.3　使用自定义用户认证服务　279

12.4　限制请求　281

12.4.1　配置请求路径访问权限　282

12.4.2　使用Spring表达式配置访问权限　283

12.4.3　强制使用HTTPS　285

12.4.4　防止跨站点请求伪造　285

12.5　用户认证功能　287

12.5.1　自定义登录页面　287

12.5.2　启用HTTP Basic认证　288

12.5.3　登出　289

第　13章 学点Spring其他的技术　291

13.1　异步线程池　291

13.1.1　定义线程池和开启异步可用　292

13.1.2　异步实例　292

13.2　异步消息　294

13.2.1　JMS实例——ActiveMQ　295

13.2.2　使用AMQP——RabbitMQ　299

13.3　定时任务　303

13.4　WebSocket 应用　306

13.4.1　开发简易的WebSocket服务　306

13.4.2　使用STOMP　311

第　14章 Spring 5新框架——WebFlux　319

14.1　基础概念　319

14.1.1　响应式编程的宣言　320

14.1.2　Reactor模型　320

14.1.3　Spring WebFlux的概述　322

14.1.4　WebHandler接口和运行流程　323

14.2　通过Spring MVC方式开发WebFlux服务端　325

14.2.1　开发持久层　325

14.2.2　开发服务层　327

14.2.3　开发控制层　328

14.2.4　配置服务　330

14.2.5　客户端开发——WebClient　331

14.3　深入WebFlux服务端开发　335

14.3.1　类型转换器——Converter　335

14.3.2　验证器——Validator　337

14.3.3　访问静态资源　338

14.4　深入客户端开发　339

14.4.1　处理服务端错误和转换　339

14.4.2　设置请求头　341

14.5　使用路由函数方式开发WebFlux　342

14.5.1　开发处理器　342

14.5.2　开发请求路由　346

14.5.3　使用过滤器　347

第　15章 实践一下——抢购商品　349

15.1　设计与开发　349

15.1.1　数据库表设计　349

15.1.2　使用MyBatis开发持久层　350

15.1.3　使用Spring开发业务层和控制层　353

15.1.4　测试和配置　355

15.2　高并发开发　357

15.2.1　超发现象　357

15.2.2　悲观锁　358

15.2.3　乐观锁　359

15.2.4　使用Redis处理高并发　365

第　16章 部署、测试和监控　371

16.1　部署和运行　371

16.1.1　打包　371

16.1.2　运行项目　373

16.1.3　热部署　375

16.2　测试　376

16.2.1　构建测试类　376

16.2.2　使用随机端口和REST风格测试　377

16.2.3　Mock测试　378

16.3　Actuator监控端点　379

16.4　HTTP监控　381

16.4.1　查看敏感信息　382

16.4.2　shutdown端点　383

16.4.3　配置端点　385

16.4.4　自定义端点　387

16.4.5　健康指标项　389

16.5　JMX监控　392

第　17章 分布式开发——Spring Cloud　393

17.1　服务治理和服务发现——Eureka　395

17.1.1　配置服务治理节点　395

17.1.2　服务发现　397

17.1.3　配置多个服务治理中心节点　401

17.2　微服务之间的调用　403

17.2.1　Ribbon客户端负载均衡　403

17.2.2　Feign声明式调用　406

17.3　断路器——Hystrix　409

17.3.1　使用降级服务　410

17.3.2　启用Hystrix仪表盘　412

17.4　路由网关——Zuul　415

17.4.1　构建Zuul网关　415

17.4.2　使用过滤器　418

17.5　使用@SpringCloudApplication　421

附录　Spring Boot知识点补充　423