
Table of Contents

Preface.....	ix
1. Spring Boot in a Nutshell.....	1
Spring Boot's Three Foundational Features	1
Starters for Simplified Dependency Management	1
Executable JARs for Simplified Deployment	3
Autoconfiguration	4
Summary	6
2. Choosing Your Tools and Getting Started.....	7
Maven or Gradle?	7
Apache Maven	7
Gradle	9
Choosing Between Maven and Gradle	10
Java or Kotlin?	11
Java	11
Kotlin	11
Choosing Between Java and Kotlin	12
Choosing a Version of Spring Boot	13
The Spring Initializr	13
Straight Outta Commandline	18
Staying In Integrated Development Environments (IDEs)	20
Cruising Down main()	22
Summary	23
3. Creating Your First Spring Boot REST API.....	25
The Hows and Whys of APIs	25
What Is REST, and Why Does It Matter?	26

Your API, HTTP Verb Style	26
Back to the Initializr	27
Creating a Simple Domain	29
GET-ting	31
@RestController in a Nutshell	31
POST-ing	35
PUT-ting	35
DELETE-ing	36
And More	36
Trust, but Verify	38
Summary	41
4. Adding Database Access to Your Spring Boot App.....	43
Priming Autoconfig for Database Access	43
What Do We Hope to Gain?	44
Adding a Database Dependency	44
Adding Code	45
Saving and Retrieving Data	52
A Bit of Polishing	56
Summary	58
5. Configuring and Inspecting Your Spring Boot App.....	59
Application Configuration	60
@Value	61
@ConfigurationProperties	66
Potential Third-Party Option	70
Autoconfiguration Report	73
Actuator	75
Getting Actuator to Open Up	81
Becoming More Environmentally Aware Using Actuator	82
Turning Up the Volume on Logging with Actuator	84
Summary	85
6. Really Digging into Data.....	87
Defining Entities	88
Template Support	88
Repository Support	89
@Before	89
Creating a Template-Based Service Using Redis	90
Initializing the Project	90
Developing the Redis Service	91
Converting from Template to Repository	99

Creating a Repository-Based Service Using the Java Persistence API (JPA)	102
Initializing the Project	103
Developing the JPA (MySQL) Service	103
Loading Data	107
Creating a Repository-Based Service Using a NoSQL Document Database	111
Initializing the Project	112
Developing the MongoDB Service	114
Creating a Repository-Based Service Using a NoSQL Graph Database	119
Initializing the Project	119
Developing the Neo4j Service	120
Summary	129
7. Creating Applications Using Spring MVC.....	131
Spring MVC: What Does It Mean?	131
End User Interactions Using Template Engines	132
Initializing the Project	133
Developing the Aircraft Positions Application	133
Passing Messages	139
Powering Up PlaneFinder	140
Extending the Aircraft Positions Application	144
Creating Conversations with WebSocket	148
What Is WebSocket?	149
Refactoring the Aircraft Positions Application	149
Summary	156
8. Reactive Programming with Project Reactor and Spring WebFlux.....	159
Introduction to Reactive Programming	159
Project Reactor	163
Tomcat versus Netty	164
Reactive Data Access	165
R2DBC with H2	165
Reactive Thymeleaf	176
RSocket for Fully Reactive Interprocess Communication	177
What Is RSocket?	177
Putting RSocket to Work	178
Summary	182
9. Testing Spring Boot Applications for Increased Production Readiness.....	185
Unit Testing	185
Introducing @SpringBootTest	186
Important Unit Tests for the Aircraft Positions Application	187
Refactoring for Better Testing	192

Testing Slices	199
Summary	204
10. Securing Your Spring Boot Application.....	207
Authentication and Authorization	207
Authentication	208
Authorization	209
Spring Security in a Nutshell	209
The HTTP Firewall	210
Security Filter Chains	210
Request and Response Headers	210
Implementing Forms-Based Authentication and Authorization with Spring Security	211
Adding Spring Security Dependencies	211
Adding Authentication	217
Authorization	223
Implementing OpenID Connect and OAuth2 for Authentication and Authorization	231
Aircraft Positions Client Application	233
PlaneFinder Resource Server	239
Summary	247
11. Deploying Your Spring Boot Application.....	249
Revisiting the Spring Boot Executable JAR	250
Building a “Fully Executable” Spring Boot JAR	251
What Does It Mean?	256
Exploding JARs	256
Deploying Spring Boot Applications to Containers	262
Creating a Container Image from an IDE	264
Creating a Container Image from the Command Line	265
Verifying the Image Exists	266
Running the Containerized Application	268
Utilities for Examining Spring Boot Application Container Images	269
Pack	269
Dive	270
Summary	271
12. Going Deeper with Reactive.....	273
When Reactive?	274
Testing Reactive Applications	274
But First, Refactoring	275
And Now, the Testing	282

Diagnosing and Debugging Reactive Applications	289
Hooks.onOperatorDebug()	290
Checkpoints	299
ReactorDebugAgent.init()	302
Summary	304
Index.....	305

