

Chapter 1: Quick Start with Java

SPRING INITIALIZR bootstrap your application now

Generate a Gradle Project with Java and Spring Boot 2.0.0 M3

Project Metadata

Artifact coordinates

Group

Artifact

Dependencies

Add Spring Boot Starters and dependencies to your application

Search for dependencies

Selected Dependencies

Reactive Web Reactive MongoDB Thymeleaf Lombok

Generate Project ⌘ + ↵

```
[  
- {  
    id: 1,  
    name: "Quick start with Java"  
},  
- {  
    id: 2,  
    name: "Reactive Web with Spring Boot"  
},  
- {  
    id: 3,  
    name: "...and more!"  
}  
]
```

Chapter 2: Reactive Web with Spring Boot

The screenshot shows the Spring Initializr interface on a Mac OS X desktop. The window title is "Spring Initializr". The address bar shows "start.spring.io". The main header reads "SPRING INITIALIZR bootstrap your application now". Below it, there's a form to "Generate a Gradle Project with Spring Boot 2.0.0 (SNAPSHOT)". The "Project Metadata" section includes fields for "Artifact coordinates", "Group" (set to "com.greglturnquist.learningspringboot"), and "Artifact" (set to "learning-spring-boot"). The "Dependencies" section allows adding Spring Boot Starters and dependencies, with a search bar containing "Web, Security, JPA, Actuator, Devtools...". Under "Selected Dependencies", three items are listed: "Reactive Web", "Lombok", and "Thymeleaf". A large green "Generate Project" button is at the bottom.

Spring Initializr

start.spring.io

SPRING INITIALIZR bootstrap your application now

Generate a Gradle Project with Spring Boot 2.0.0 (SNAPSHOT)

Project Metadata

Artifact coordinates

Group

com.greglturnquist.learningspringboot

Artifact

learning-spring-boot

Dependencies

Add Spring Boot Starters and dependencies to your application

Search for dependencies

Web, Security, JPA, Actuator, Devtools...

Selected Dependencies

Reactive Web Lombok Thymeleaf

Generate Project

Chapter 3: Reactive Data Access with Spring Boot

```
org.mongodb.driver.cluster ..... : Cluster created with settings {hosts=[localhost:27017], mode=SINGLE, requiredCluste
org.mongodb.driver.connection ..... : Opened connection [connectionId{localValue:1, serverValue:202}] to localhost:27017
org.mongodb.driver.cluster ..... : Monitor thread successfully connected to server with description ServerDescription{
org.mongodb.driver.cluster ..... : Cluster created with settings {hosts=[localhost:27017], mode=SINGLE, requiredCluste
org.mongodb.driver.connection ..... : Opened connection [connectionId{localValue:2, serverValue:203}] to localhost:27017
org.mongodb.driver.cluster ..... : Monitor thread successfully connected to server with description ServerDescription{
o.s.j.e.a.AnnotationMBeanExporter ..... : Registering beans for JMX exposure on startup
r.ipc.netty.tcp.BlockingNettyContext ..... : Started HttpServer on /0:0:0:0:0:0:8080
o.s.b.web.embedded.netty.NettyWebServer ..... : Netty started on port(s): 8080
org.mongodb.driver.connection ..... : Opened connection [connectionId{localValue:3, serverValue:204}] to localhost:27017
```

```
org.mongodb.driver.cluster ..... : Cluster created with settings {hosts=[localhost:27017], mode=SINGLE, requiredCluste
org.mongodb.driver.connection ..... : Opened connection [connectionId{localValue:1, serverValue:205}] to localhost:27017
org.mongodb.driver.cluster ..... : Monitor thread successfully connected to server with description ServerDescription{
org.mongodb.driver.cluster ..... : Cluster created with settings {hosts=[localhost:27017], mode=SINGLE, requiredCluste
org.mongodb.driver.connection ..... : Opened connection [connectionId{localValue:2, serverValue:206}] to localhost:27017
org.mongodb.driver.cluster ..... : Monitor thread successfully connected to server with description ServerDescription{
.m.c.i.MongoPersistentEntityIndexCreator ..... : Analyzing class class com.greglturnquist.learningspringboot.Image for index inform
.m.c.i.MongoPersistentEntityIndexCreator ..... : Analyzing class class com.greglturnquist.learningspringboot.Image for index inform
o.s.j.e.a.AnnotationMBeanExporter ..... : Registering beans for JMX exposure on startup
r.ipc.netty.tcp.BlockingNettyContext ..... : Started HttpServer on /0:0:0:0:0:0:8080
o.s.b.web.embedded.netty.NettyWebServer ..... : Netty started on port(s): 8080
org.mongodb.driver.connection ..... : Opened connection [connectionId{localValue:3, serverValue:207}] to localhost:27017
o.s.data.mongodb.core.MongoTemplate ..... : Dropped collection [image]
o.s.data.mongodb.core.MongoTemplate ..... : Inserting Document containing fields: [_id, name, _class] in collection: image
o.s.data.mongodb.core.MongoTemplate ..... : Inserting Document containing fields: [_id, name, _class] in collection: image
o.s.data.mongodb.core.MongoTemplate ..... : Inserting Document containing fields: [_id, name, _class] in collection: image
```

```
createImage-files..... : | onSubscribe([Synchronous.Fuseable] FluxArray.ArraySubscription)
createImage-flatMap..... : onSubscribe(FluxFlatMap.FlatMapMain)
createImage-done..... : onSubscribe(MonoIgnoreElements.IgnoreElementsSubscriber)
createImage-done..... : request(unbounded)
createImage-flatMap..... : request(unbounded)
createImage-files..... : | request(256)
createImage-files..... : | onNext(Mock for FilePart, hashCode: 154449199)
createImage-when..... : onSubscribe([Fuseable] MonoWhen.WhenCoordinator)
createImage-when..... : request(32)
o.s.d.m.core.ReactiveMongoTemplate..... : Saving Document containing fields: [_id, name, _class]
createImage-save..... : onSubscribe(FluxOnErrorResume.ResumeSubscriber)
createImage-save..... : request(unbounded)
createImage-copy..... : | onSubscribe([Fuseable] MonoFlatMap.FlatMapMain)
createImage-copy..... : | request(unbounded)
createImage-picktarget..... : | onSubscribe([Synchronous.Fuseable] Operators.ScalarSubscription)
createImage-newfile..... : | onSubscribe([Fuseable] FluxMapFuseable.MapFuseableSubscriber)
createImage-newfile..... : | request(unbounded)
createImage-picktarget..... : | request(unbounded)
createImage-picktarget..... : | onNext(upload-dir/alpha.jpg)
createImage-newfile..... : | onNext(upload-dir/alpha.jpg)
createImage-copy..... : | onComplete()
createImage-picktarget..... : | onComplete()
createImage-newfile..... : | onComplete()
createImage-files..... : | onNext(Mock for FilePart, hashCode: 430329518)
createImage-when..... : onSubscribe([Fuseable] MonoWhen.WhenCoordinator)
createImage-when..... : request(32)
o.s.d.m.core.ReactiveMongoTemplate..... : Saving Document containing fields: [_id, name, _class]
createImage-save..... : onSubscribe(FluxOnErrorResume.ResumeSubscriber)
createImage-save..... : request(unbounded)
createImage-copy..... : | onSubscribe([Fuseable] MonoFlatMap.FlatMapMain)
createImage-copy..... : | request(unbounded)
createImage-picktarget..... : | onSubscribe([Synchronous.Fuseable] Operators.ScalarSubscription)
createImage-newfile..... : | onSubscribe([Fuseable] FluxMapFuseable.MapFuseableSubscriber)
createImage-newfile..... : | request(unbounded)
createImage-picktarget..... : | request(unbounded)
createImage-picktarget..... : | onNext(upload-dir;bravo.jpg)
org.mongodb.driver.connection..... : Opened connection [connectionId{localValue:4, serverValue:220}] to lo
createImage-newfile..... : | onNext(upload-dir;bravo.jpg)
createImage-copy..... : | onComplete()
createImage-picktarget..... : | onComplete()
createImage-newfile..... : | onComplete()
createImage-files..... : | onComplete()
org.mongodb.driver.connection..... : Opened connection [connectionId{localValue:5, serverValue:221}] to lo
createImage-save..... : onNext(Image(id=85c62c00-c315-4a6b-914b-059e1020ee6b, name=bravo.jpg)
createImage-save..... : onNext(Image(id=7f4fb639-8b0d-4791-b88a-366b2cce0b55, name=alpha.jpg)
createImage-when..... : onComplete()
createImage-when..... : onComplete()
createImage-save..... : onComplete()
createImage-flatMap..... : onComplete()
createImage-done..... : onComplete()
```

Learning Spring Boot - 2nd Edition

Id	Name	Image	
1	learning-spring-boot-cover.jpg		<button>Delete</button>
3	bazinga.png		<button>Delete</button>
2	learning-spring-boot-2nd-edition-cover.jpg		<button>Delete</button>

No file chosen

Id	Name	Image	
3	bazinga.png		<button>Delete</button>
2	learning-spring-boot-2nd-edition-cover.jpg		<button>Delete</button>

```
o.s.d.m.r.MongoQueryCreator.....: Created query Query: { "name": "learning-spring-boot-cover.jpg"}, Fields:  
o.s.d.m.core.ReactiveMongoTemplate.....: findOne using query: { "name": "learning-spring-boot-cover.jpg"} fields:  
deleteImage-when.....: onSubscribe([Fuseable] MonoWhen.WhenCoordinator)  
deleteImage-done.....: onSubscribe(MonoIgnoreElements.IgnoreElementsSubscriber)  
deleteImage-done.....: request(unbounded)  
deleteImage-when.....: request(unbounded)  
deleteImage-record.....: | onSubscribe([Fuseable] MonoFlatMap.FlatMapMain)  
deleteImage-record.....: | request(unbounded)  
o.s.d.m.core.ReactiveMongoTemplate.....: findOne using query: { "name": "learning-spring-boot-cover.jpg"} fields:  
deleteImage-find.....: onSubscribe(FluxOnErrorResume.ResumeSubscriber)  
deleteImage-find.....: request(unbounded)  
deleteImage-file.....: onSubscribe([Fuseable] Operators.EmptySubscription)  
deleteImage-file.....: request(unbounded)  
deleteImage-file.....: onComplete()  
deleteImage-find.....: onNext(Image{id=1, name=learning-spring-boot-cover.jpg})  
o.s.d.m.core.ReactiveMongoTemplate.....: Remove using query: { "_id": "1"} in collection: image.  
deleteImage-find.....: onComplete()  
deleteImage-record.....: | onComplete()  
deleteImage-when.....: onComplete()  
deleteImage-done.....: onComplete()
```

Chapter 4: Testing with Spring Boot

The screenshot shows an IDE interface with a code editor and a context menu open over a Java test class named `ImageTests`. The code editor contains the following code:

```
ImageTests
package com.greglturnquist.learningspringboot;

import ...

/**
 * @author Greg
 */
// tag::code[]
public class ImageTests {

    @Test
    public void imagesManagedByLombokShouldWork() {
        Image image = Image.builder()
            .id("id")
            .fileName("file-name.jpg")
            .build();

        assertThat(image.getId()).isEqualTo("id");
        assertThat(image.getFileName()).isEqualTo("file-name.jpg");
    }

}
// end::code[]
```

The context menu is open at the bottom of the code editor, listing various actions such as Run, Debug, and Coverage. The "Run 'ImageTests'" option is highlighted in blue, indicating it is selected.

The screenshot shows the IDE's run tool window with the title "Run ImageTests". It displays the following execution results:

Test	Status	Time
ImageTests (com.greglturnquist.learningspringboot)	OK	49ms
imagesManagedByLombokShouldWork	OK	49ms

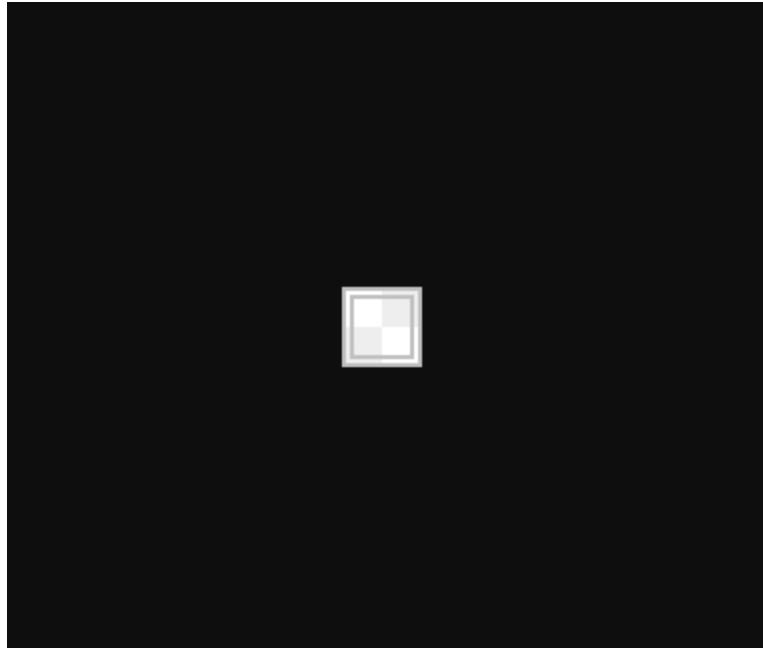
Run LivelImageRepositoryTests and 1 more		
		<default package>
		LivelImageRepositoryTests 401ms
		findAllShouldWork 259ms
		findByNameShouldWork 225ms
		EmbeddedImageRepositoryTests 34ms
		findAllShouldWork 142ms
		findByNameShouldWork 130ms
		12ms

Run HomeControllerTests		
		HomeControllerTests (com.greglturnquist.learningspringboot) 834ms
		baseRouteShouldListAllImages 527ms
		deleteImageShouldWork 161ms
		fetchingNullImageShouldFail 78ms
		fetchingImageShouldWork 68ms

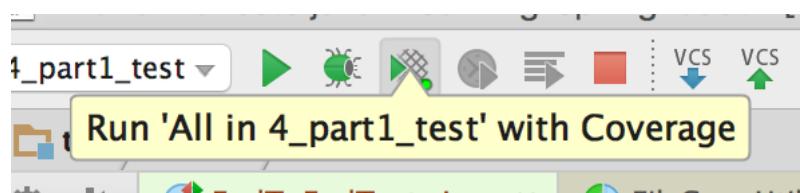
Learning Spring Boot - 2nd Edition

Id	Name	Image	
1	learning-spring-boot-cover.jpg		Delete
2	learning-spring-boot-2nd-edition-cover.jpg		Delete
3	bazinga.png		Delete

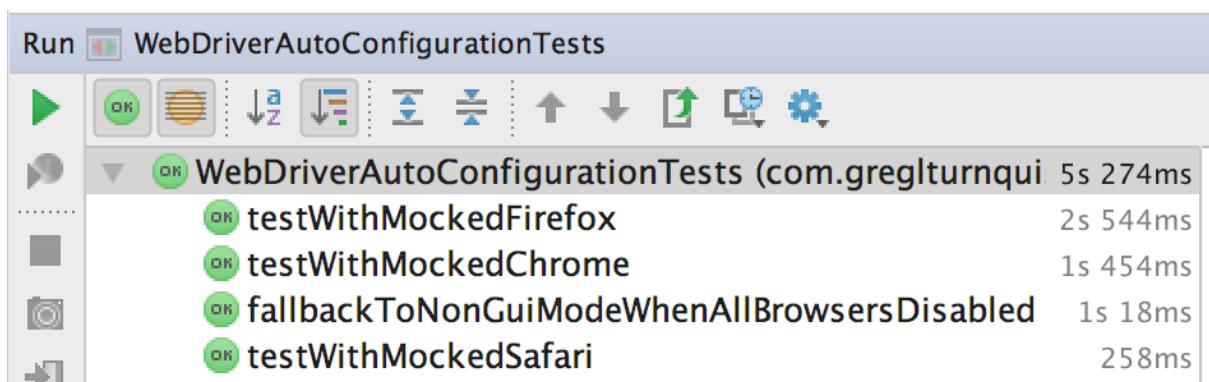
No file chosen



Run All in 4_part1_test		
	<default package>	2s 428ms
▼	EndToEndTests	1s 657ms
homePageShouldWork	1s 657ms	
▼	EmbeddedImageRepositoryTests	433ms
findAllShouldWork	400ms	
findByNameShouldWork	33ms	
▼	HomeControllerTests	251ms
fetchingNullImageShouldFail	84ms	
deleteImageShouldWork	79ms	
fetchingImageShouldWork	45ms	
baseRouteShouldListAllImages	43ms	
▼	LiveImageRepositoryTests	51ms
findAllShouldWork	40ms	
findByNameShouldWork	11ms	
▼	ImageServiceTests	35ms
createImageShouldWork	13ms	
deleteImageShouldWork	8ms	
findOneShouldReturnNotYetFetchedUrl	8ms	
findAllShouldJustReturnTheFlux	6ms	
▼	ImageTests	1ms
imagesManagedByLombokShouldWork	1ms	



```
▼ src
  ▼ main
    ▼ java 85% classes, 88% lines covered
      ▼ com.greglturnquist.learningspringboot 85% classes, 88% lines covered
        CApiController 25% methods, 25% lines covered
        HomeController 83% methods, 90% lines covered
        Image 100% methods, 100% lines covered
        ImageRepository
        ImageService 100% methods, 92% lines covered
        InitDatabase 100% methods, 100% lines covered
        LearningSpringBootApplication 0% methods, 33% lines covered
        ReactiveThymeleafConfig 100% methods, 100% lines covered
```



Chapter 5: Developer Tools for Spring Boot Apps



Learning Spring Boot - 2nd Edition

Id	Name	Image	
57f16c0883a5d2446f27602e	test		Delete
57f16c0883a5d2446f27602f	test2		Delete
57f16c0883a5d2446f276030	test3		Delete

No file chosen



```
<body>
```

```
.... <h1>Learning Spring Boot - 2nd Edition</h1>
```

```
.... <h3>Using the LiveReload plugin in your browser will speed up efforts.</h3>
```

Learning Spring Boot - 2nd Edition

Using the LiveReload plugin in your browser will speed up efforts.

Learning Spring Boot - 2nd Edition

Using the LiveReload plugin in your browser will speed up efforts.

DevTools can also detect code changes too

The screenshot shows the IntelliJ IDEA interface. At the top, there's a 'Run/Debug Configurations' bar with a search field containing 'LearningSpringBootApplication' and a 'Share' checkbox. Below this is a tab bar with 'Configuration' (which is selected), 'Code Coverage', and 'Logs'. The main area contains configuration fields: 'Main class:' set to 'com.greglturnquist.learningspringboot.LearningSpringBootApplication', 'VM options:' empty, 'Program arguments:' containing '--debug', and 'Working directory:' empty. To the right of these fields is a large scrollable window showing DevTools logs. The logs output includes:

```
2016-09-30 14:57:52.106 DEBUG 50763 --- [ restartedMain] o.s.b.a.e.jmx.EndpointConfigurer : JMX endpoint registered for application
2016-09-30 14:57:52.128 DEBUG 50763 --- [ restartedMain] autoConfigurationReportLogger : =====
===== AUTO-CONFIGURATION REPORT =====
=====

Positive matches:
-----
... AuditAutoConfiguration#auditListener matched
..... - @ConditionalOnMissingBean (types: org.springframework.boot.actuate.audit.AuditEventRepositoryConfiguration)
... AuditAutoConfiguration.AuditEventRepositoryConfiguration matched
..... - @ConditionalOnMissingBean (types: org.springframework.boot.actuate.audit.AuditEventRepository)
```

The screenshot shows a web browser window with the address bar displaying "localhost:8080/application". The page content is a JSON object with various fields and their corresponding href values:

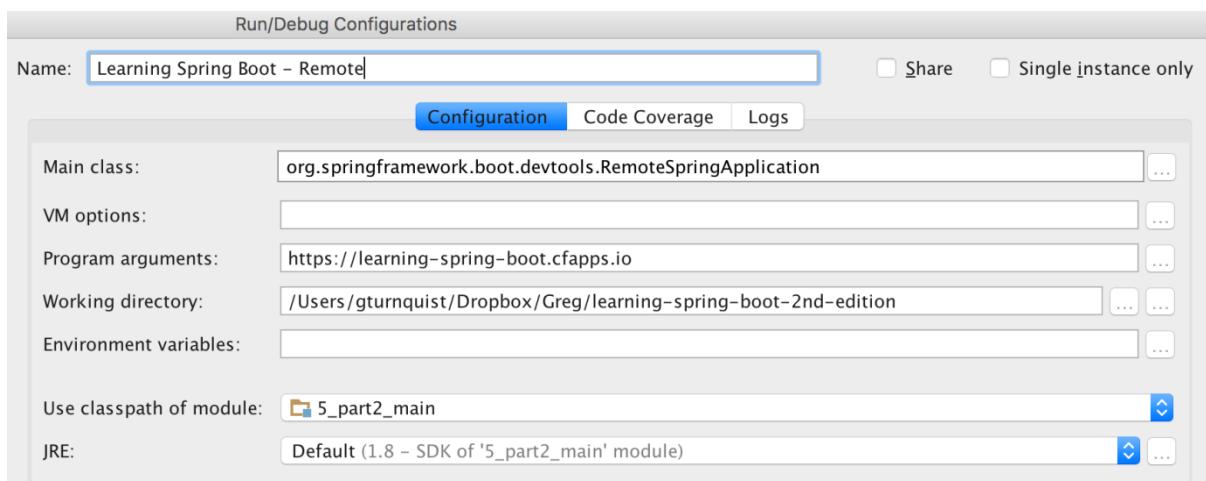
```
// 20170912231024
// http://localhost:8080/application

{
  "_links": {
    "self": {
      "href": "http://localhost:8080/application",
      "templated": false
    },
    "auditevents": {
      "href": "http://localhost:8080/application/auditevents",
      "templated": false
    },
    "beans": {
      "href": "http://localhost:8080/application/beans",
      "templated": false
    },
    "autoconfig": {
      "href": "http://localhost:8080/application/autoconfig",
      "templated": false
    },
    "configprops": {
      "href": "http://localhost:8080/application/configprops",
      "templated": false
    },
    "env": {
      "href": "http://localhost:8080/application/env",
      "templated": false
    },
    "env-toMatch": {
      "href": "http://localhost:8080/application/env/{toMatch}",
      "templated": true
    },
    "health": {
      "href": "http://localhost:8080/application/health",
      "templated": false
    },
    "status": {
      "href": "http://localhost:8080/application/status",
      "templated": false
    }
}
```

The screenshot shows a web browser window with the URL `localhost:8080/application/autoconfig`. The page displays a JSON configuration report. The content includes several sections of code, likely from a `ConfigurableApplicationContext` or similar configuration source, detailing the state of beans and their configurations.

```
// 20170912220059
// http://localhost:8080/application/autoconfig

{
  "positiveMatches": [
    {
      "AuditAutoConfiguration#auditListener": [
        {
          "condition": "OnBeanCondition",
          "message": "@ConditionalOnMissingBean (types: org.springframework.boot.actuate.audit.listener.AbstractAuditListener; SearchStrategy: all) did not find any beans"
        }
      ],
      "AuditAutoConfiguration.AuditEventRepositoryConfiguration": [
        {
          "condition": "OnBeanCondition",
          "message": "@ConditionalOnMissingBean (types: org.springframework.boot.actuate.audit.AuditEventRepository; SearchStrategy: all) did not find any beans"
        }
      ],
      "AuditEventsEndpointAutoConfiguration#auditEventsEndpoint": [
        {
          "condition": "OnBeanCondition",
          "message": "@ConditionalOnBean (types: org.springframework.boot.actuate.audit.AuditEventRepository; SearchStrategy: all) found bean 'auditEventRepository'; @ConditionalOnMissingBean (types: org.springframework.boot.actuate.audit.AuditEventsEndpoint; SearchStrategy: all) did not find any beans"
        }
      ],
    }
  ]
}
```



← → ⌂ ⌄ learning-spring-boot.cfapps.io

Learning Spring Boot - 2nd Edition

It's really handy to make local edits and watch them go out to the cloud automatically.



localhost:8080/application/health

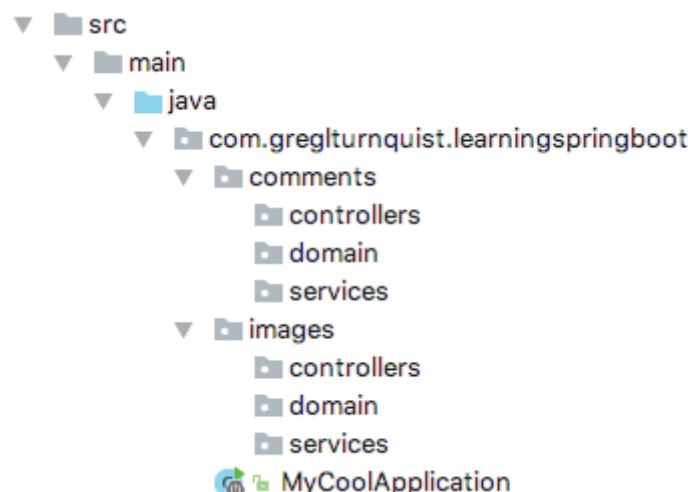
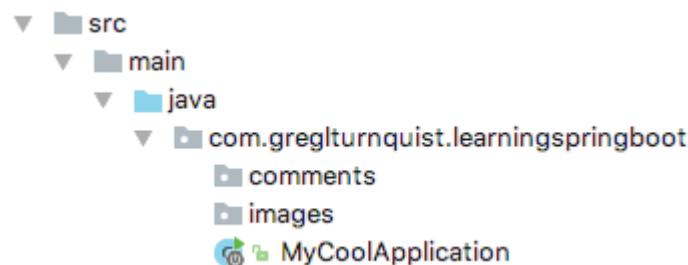
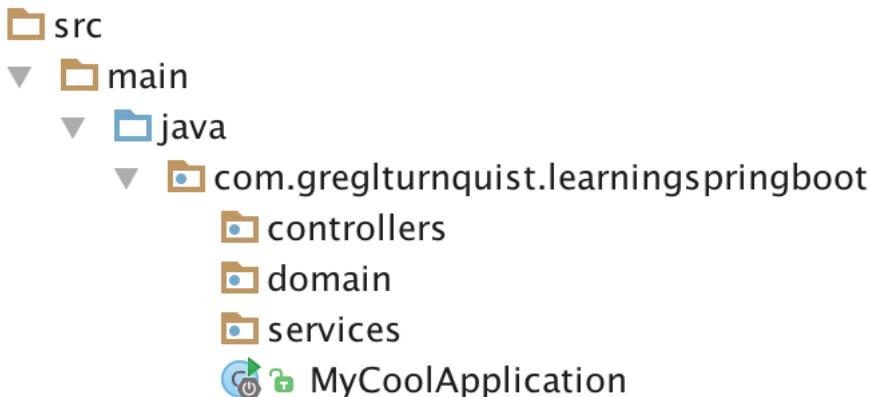
```
// 20170912234318
// http://localhost:8080/application/health
{
  "status": "DOWN",
  "details": {
    "mongo": {
      "status": "UP",
      "details": {
        "version": "3.4.6"
      }
    },
    "diskSpace": {
      "status": "UP",
      "details": {
        "total": 498937626624,
        "free": 43624402944,
        "threshold": 10485760
      }
    },
    "learningSpringBoot": {
      "status": "DOWN",
      "details": {
        "HTTP Status Code": 404
      }
    }
  }
}
```

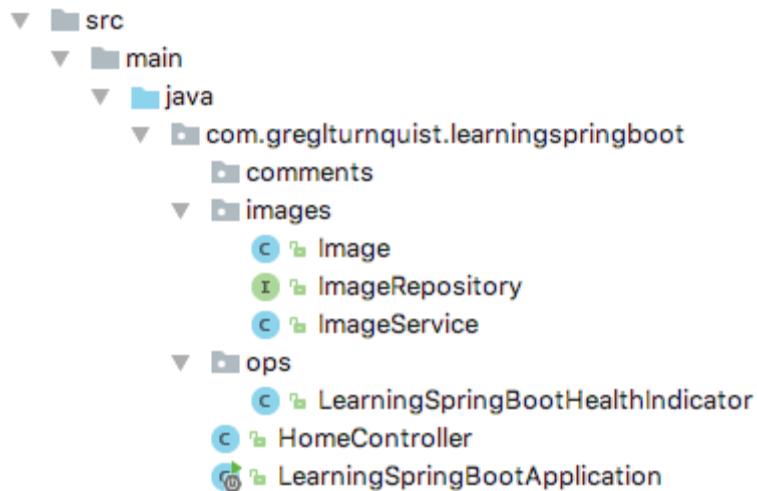
Learning Spring Boot - 2nd Edition

Id	Name	Image	
57f07b6c8dd1b00a1c3cee4a	B05771_MockupCover_Normal.jpg		<button data-bbox="1283 444 1367 473">Delete</button>
57f07b788dd1b00a1c3cee4b	spring-boot-project-logo.png		<button data-bbox="1283 640 1367 669">Delete</button>
57f07ba48dd1b00a1c3cee4d	bazinga.png		<button data-bbox="1283 813 1367 842">Delete</button>

No file chosen

Chapter 6: AMQP Messaging with Spring Boot





```
[o.s.b.d.a.OptionalLiveReloadServer.....: LiveReload server is running on port 35729
[o.s.j.e.a.AnnotationMBeanExporter.....: Registering beans for JMX exposure on startup
[o.s.j.e.a.AnnotationMBeanExporter.....: Bean with name 'rabbitConnectionFactory' has been a
[o.s.j.e.a.AnnotationMBeanExporter.....: Located managed bean 'rabbitConnectionFactory': reg
[o.s.c.support.DefaultLifecycleProcessor: Starting beans in phase -2147482648
[o.s.c.support.DefaultLifecycleProcessor: Starting beans in phase 0
[o.s.c.support.DefaultLifecycleProcessor: Starting beans in phase 2147483647
[o.s.a.r.c.CachingConnectionFactory.....: Created new connection: rabbitConnectionFactory#168
[o.s.amqp.rabbit.core.RabbitAdmin.....: Auto-declaring a non-durable, auto-delete, or exclu
[r.ipc.netty.tcp.BlockingNettyContext....: Started HttpServer on /0:0:0:0:0:0:0:8080
[o.s.b.web.embedded.netty.NettyWebServer: Netty started on port(s): 8080
[org.mongodb.driver.connection.....: Opened connection [connectionId{localValue:3, serve
boot-cover.jpg)
boot-2nd-edition-cover.jpg)

[c.g.l.LearningSpringBootApplication.....: Started LearningSpringBootApplication in 6.655 seco
```

Learning Spring Boot - 2nd Edition

1 of 1

20 item(s) per page

Id	Name	Image	
57f8641e83a5d272b18d4ce1	B05771_MockupCover_Normal.jpg		<input type="button" value="Delete"/> <ul style="list-style-type: none">I love the cover!Can I order a copy yet? <input type="text"/> <input type="button" value="Submit"/>
57f8643b83a5d272b18d4ce3	bazinga.png		<input type="button" value="Delete"/> <input type="text"/> <input type="button" value="Submit"/>
57f8645883a5d272b18d4ce5	spring-boot-project-logo.png		<input type="button" value="Delete"/> <input type="text" value="Spring Boot is awesome!"/> <input type="button" value="Submit"/>

No file chosen

Name:

Main class:

VM options:

Program arguments:

Working directory:

Environment variables:

Use classpath of module: 6_part2_main

JRE: Default (1.8 – SDK of '6_part2_main' module)

Spring Boot Settings

Enable debug output Hide Banner

Active Profiles: simulator

```
integration.channel.output.errorRate.mean: 0,
integration.channel.output.errorRate.max: 0,
integration.channel.output.errorRate.min: 0,
integration.channel.output.errorRate.stdev: 0,
integration.channel.output.errorRate.count: 0,
integration.channel.output.sendCount: 1080,
integration.channel.output.sendRate.mean: 33.2953175275134,
integration.channel.output.sendRate.max: 0.10356745398044587,
integration.channel.output.sendRate.min: 0.00007268595695495606,
integration.channel.output.sendRate.stdev: 127.44592804513556,
integration.channel.output.sendRate.count: 1080,
integration.channel.output.receiveCount: -1,
integration.channel.input.errorRate.mean: 0,
integration.channel.input.errorRate.max: 0,
integration.channel.input.errorRate.min: 0,
integration.channel.input.errorRate.stdev: 0,
integration.channel.input.errorRate.count: 0,
integration.channel.input.sendCount: 1080,
integration.channel.input.sendRate.mean: 30.60592083415993,
integration.channel.input.sendRate.max: 0.10221006900072098,
integration.channel.input.sendRate.min: 0.000672169029712677,
integration.channel.input.sendRate.stdev: 20.733646866079425,
integration.channel.input.sendRate.count: 1080,
integration.channel.input.receiveCount: -1,
```

```

{
    status: "UP",
    - learningSpringBoot: {
        status: "UP"
    },
    - diskSpace: {
        status: "UP",
        total: 498937626624,
        free: 89344151552,
        threshold: 10485760
    },
    - rabbit: {
        status: "UP",
        version: "3.6.4"
    },
    - mongo: {
        status: "UP",
        version: "3.2.6"
    },
    - binders: {
        status: "UP",
        - rabbit: {
            status: "UP",
            - binderHealthIndicator: {
                status: "UP",
                version: "3.6.4"
            }
        }
    }
}

```

```

o.s.a.r.c.CachingConnectionFactory.....: Created new connection: rabbitConnectionFactory#1470a7b3:0/SimpleConnection@31792a;
o.s.integration.channel.DirectChannel...: Channel 'unknown.channel.name' has 1 subscriber(s).
o.s.c.s.binding.BindableProxyFactory...: Binding outputs for :interface org.springframework.cloud.stream.messaging.Processor
o.s.c.s.binding.BindableProxyFactory...: Binding :interface org.springframework.cloud.stream.messaging.Processor:output
o.s.integration.channel.DirectChannel...: Channel 'application.output' has 1 subscriber(s).
o.s.c.support.DefaultLifecycleProcessor: Starting beans in phase 0
o.s.i.endpoint.EventDrivenConsumer....: Adding {logging-channel-adapter:_org.springframework.integration.errorLogger} as a
o.s.i.channel.PublishSubscribeChannel.: Channel 'application.errorChannel' has 1 subscriber(s).
o.s.i.endpoint.EventDrivenConsumer....: started _org.springframework.integration.errorLogger
o.s.c.support.DefaultLifecycleProcessor: Starting beans in phase 2147482647
o.s.c.s.binding.BindableProxyFactory...: Binding inputs for :interface org.springframework.cloud.stream.messaging.Source
o.s.c.s.binding.BindableProxyFactory...: Binding inputs for :interface org.springframework.cloud.stream.messaging.Processor
o.s.c.s.binding.BindableProxyFactory...: Binding :interface org.springframework.cloud.stream.messaging.Processor:input
c.s.b.r.p.RabbitExchangeQueueProvisioner: declaring queue for inbound: learning-spring-boot-comments.learning-spring-boot, bc
c.s.b.r.p.RabbitExchangeQueueProvisioner: autoBindDLQ=false for: learning-spring-boot-comments.learning-spring-boot
o.s.i.a.i.AmqpInboundChannelAdapter...: started inbound.learning-spring-boot
o.s.i.endpoint.EventDrivenConsumer....: Adding {message-handler:inbound.learning-spring-boot-comments.learning-spring-boot}
o.s.i.endpoint.EventDrivenConsumer....: started inbound.learning-spring-boot-comments.learning-spring-boot

```

```
o.s.integration.channel.DirectChannel : preSend on channel 'output', message: GenericMessage [payload=Comment(id=null, image=null)]
  abstractMessageChannelBinder$SendingHandler : org.springframework.cloud.stream.binder.AbstractMessageChannelBinder$SendingHandler
    o.s.i.codec.kryo.CompositeKryoRegistrar : registering [40, java.io.File] with serializer org.springframework.integration.codec.KryoSerializer
      o.s.i.a.outbound.AmqpOutboundEndpoint : org.springframework.integration.amqp.outbound.AmqpOutboundEndpoint@52038bac received
        s.i.m.AbstractHeaderMapper$HeaderMatcher : headerName=[contentType] WILL be mapped, matched pattern=*
          o.s.i.a.outbound.AmqpOutboundEndpoint : handler 'org.springframework.integration.amqp.outbound.AmqpOutboundEndpoint@52038bac'
            o.s.integration.channel.DirectChannel : postSend (sent=true) on channel 'output', message: GenericMessage [payload=Comment(id=null, image=null)]
              s.i.m.AbstractHeaderMapper$HeaderMatcher : headerName=[amqp_receivedDeliveryMode] WILL be mapped, matched pattern=*
                s.i.m.AbstractHeaderMapper$HeaderMatcher : headerName=[amqp_receivedRoutingKey] WILL be mapped, matched pattern=*
                  s.i.m.AbstractHeaderMapper$HeaderMatcher : headerName=[amqp_receivedExchange] WILL be mapped, matched pattern=*
                    s.i.m.AbstractHeaderMapper$HeaderMatcher : headerName=[amqp_deliveryTag] WILL be mapped, matched pattern=*
                      s.i.m.AbstractHeaderMapper$HeaderMatcher : headerName=[amqp_correlationId] WILL be mapped, matched pattern=*
                        s.i.m.AbstractHeaderMapper$HeaderMatcher : headerName=[amqp_redelivered] WILL be mapped, matched pattern=*
                          s.i.m.AbstractHeaderMapper$HeaderMatcher : headerName=[contentType] WILL be mapped, matched pattern=*
                            s.i.m.AbstractHeaderMapper$HeaderMatcher : headerName=[contentType] WILL be mapped, matched pattern=*
                              abstractMessageChannelBinder$ReceivingHandler : org.springframework.cloud.stream.binder.AbstractMessageChannelBinder$ReceivingHandler
                                findAll : | onSubscribe([Fuseable] FluxOnAssembly.OnAssemblySubscriber)
                                  findAll : | request(256)
                                findAll : preSend on channel 'input', message: GenericMessage [payload=Comment(id=null, image=null)]
                                  : | onNext(Image(id=59894898c4d956e34025dacc, name=learning-spring-boot-cover.jpg))
                                  findAll : | request(1)
                                  findAll : | onNext(Image(id=59894898c4d956e34025dacc, name=learning-spring-boot-2nd-edition-cover.jpg))
                                  findAll : | request(1)
                                  findAll : | onNext(Image(id=59894899c4d956e34025dace, name=bazinga.png))
                                  findAll : | onComplete()
                                o.s.integration.channel.DirectChannel : postSend (sent=true) on channel 'input', message: GenericMessage [payload=Comment(id=null, image=null)]
```

Chapter 7: Microservices with Spring Boot

Generate a Gradle Project with Java
and Spring Boot 2.0.0 (SNAPSHOT)

Project Metadata

Artifact coordinates

Group

com.greglturnquist.learningspringboot



Artifact

learning-spring-boot-eureka-server



Dependencies

Add Spring Boot Starters and dependencies to your application

Search for dependencies

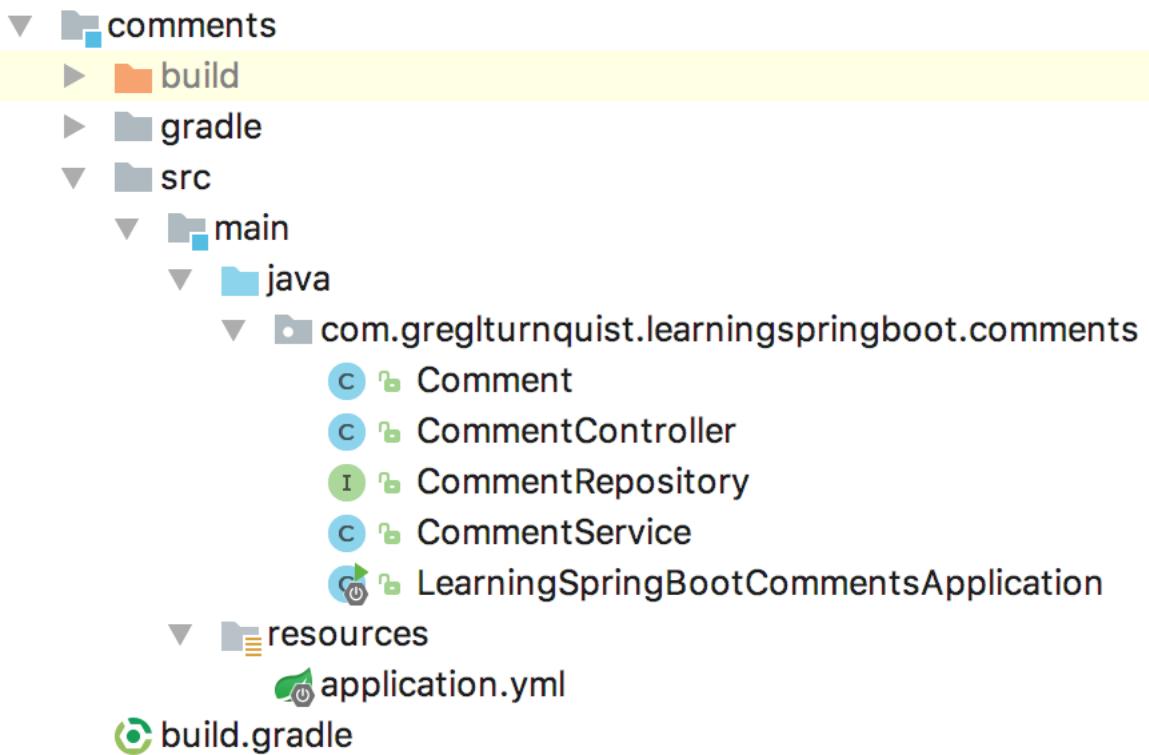
Web, Security, JPA, Actuator, Devtools...

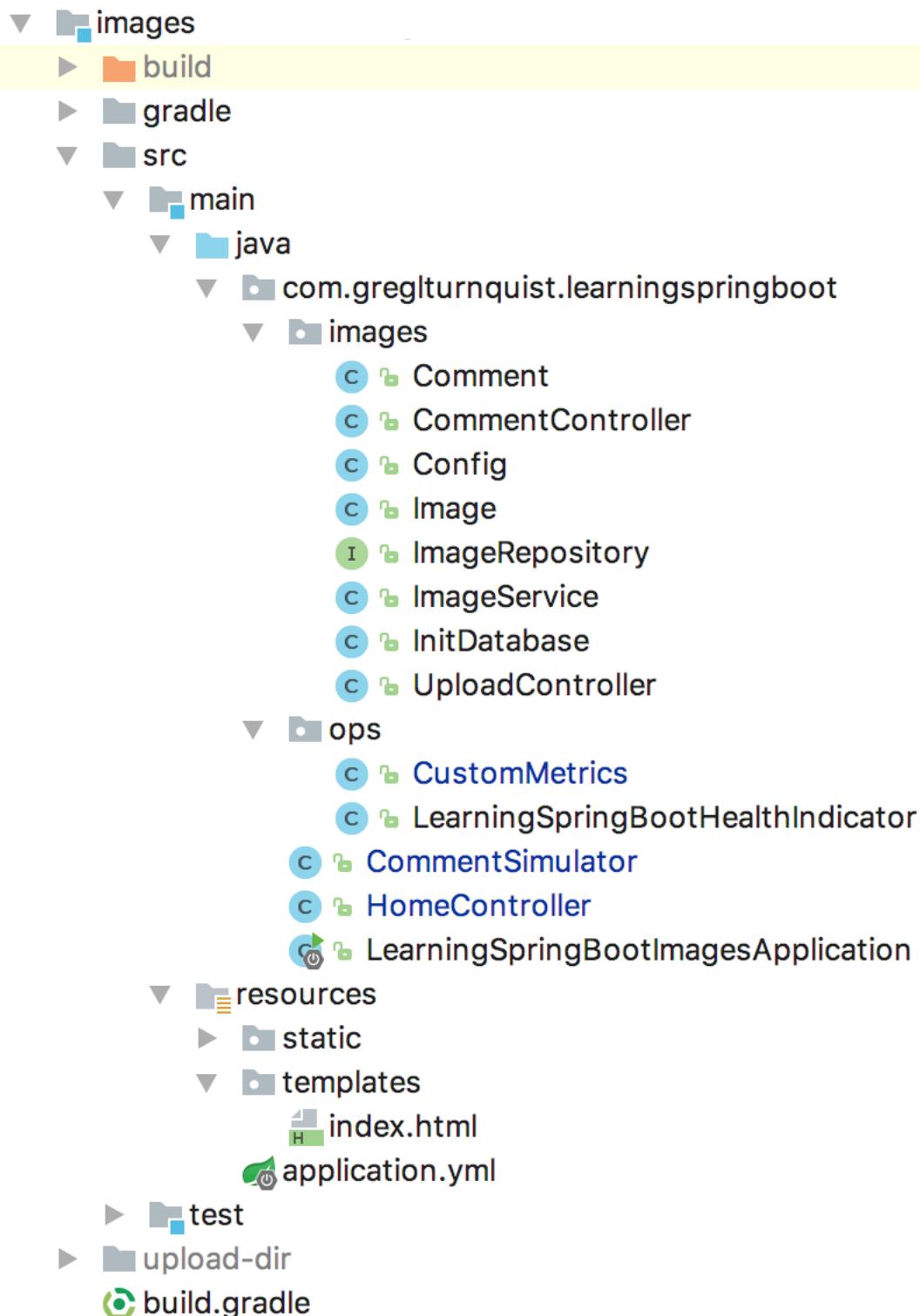
Selected Dependencies

Eureka Server X



Generate Project ⌘ + ↵





The screenshot shows a web browser window titled "Eureka". The address bar displays "localhost:8761". The main content area is titled "spring Eureka" and "System Status". Below this, there is a table with the following data:

Environment	test
Data center	default
Current time	2016-10-20T22:42:53 -0500
Uptime	00:19
Lease expiration enabled	false
Renews threshold	5
Renews (last min)	4

DS Replicas

Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
COMMENTS	n/a (1)	(1)	UP (1) - retina:comments:9000
IMAGES	n/a (1)	(1)	UP (1) - retina:images

The screenshot shows a web browser window titled "retina:9000/info". The address bar displays "localhost:8761" and "retina:9000/info". The main content area shows the following JSON response:

```
{ }
```

Learning Spring Boot - 2nd Edition

Id	Name	Image	
5809914283a5d2ac3e0ea0f6	platform-spring-boot.png		Delete <input type="text"/> Submit
5809916283a5d2ac3e0ea0f7	B05771_MockupCover_Normal.jpg		Delete <ul style="list-style-type: none">• Cool cover!• Can't wait to get my copy <input type="text"/> Submit

Choose File No file chosen

Upload

DS Replicas

Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
COMMENTS	n/a (2)	(2)	UP (2) - retina:comments:9001 , retina:comments:9000
IMAGES	n/a (1)	(1)	UP (1) - retina:images

Hystrix Dashboard

localhost:7979/hystrix



Hystrix Dashboard

<http://localhost:8080/hystrix.stream>

Cluster via Turbine (default cluster): http://turbine-hostname:port/turbine.stream
Cluster via Turbine (custom cluster): http://turbine-hostname:port/turbine.stream?cluster=[clusterName]
Single Hystrix App: http://hystrix-app:port/hystrix.stream

Delay: 2000 ms Title: Images

Monitor Stream

Hystrix Monitor

localhost:7979/hystrix/mo...



HYSTRIX
DEFEND YOUR APP

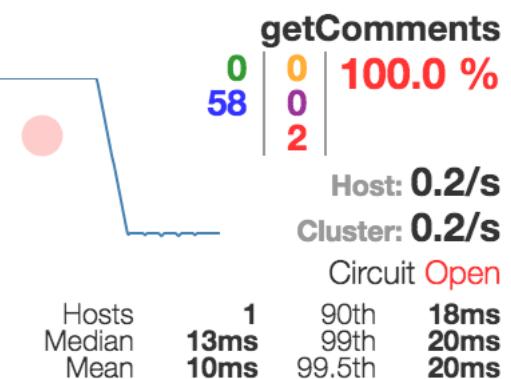
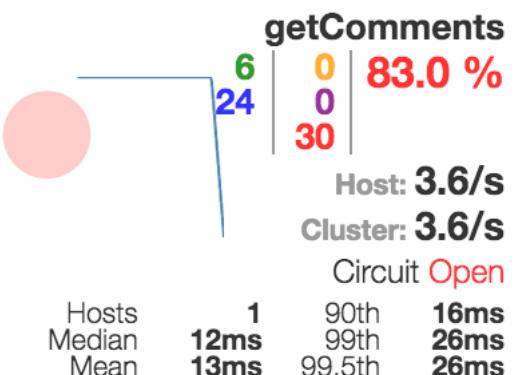
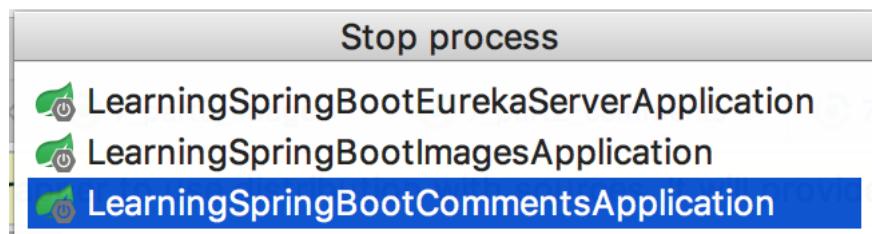
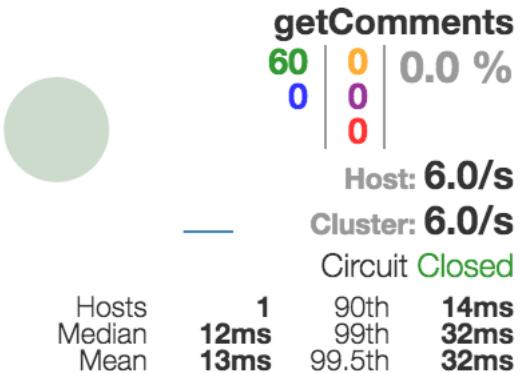
Hystrix Stream: http://localhost:8080/hystrix.stream

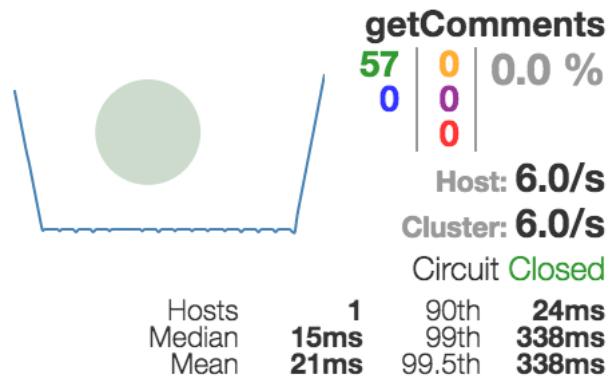
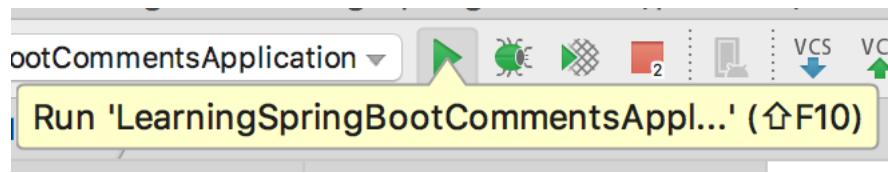
Circuit Sort: [Error then Volume](#) | [Alphabetical](#) | [Volume](#) | [Error](#) | [Mean](#) | [Median](#) | [90](#) | [99](#) | [99.5](#)
[Success](#) | [Short-Circuited](#) | [Timeout](#) | [Rejected](#) | [Failure](#) | [Error %](#)

	getComments
Hosts	0 0 0.0 %
Median	0 0 0
Mean	0 0 0
	Host: 0.0/s
	Cluster: 0.0/s
	Circuit: Closed
Hosts	1
Median	0ms
Mean	0ms
	90th: 0ms
	99th: 0ms
	99.5th: 0ms

Thread Pools Sort: [Alphabetical](#) | [Volume](#) |

	CommentHelper
Active	0
Queued	0
Pool Size	3
	Max Active: 0
	Executions: 0
	Queue Size: 5
	Host: 0.0/s
	Cluster: 0.0/s

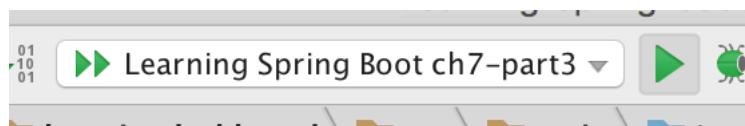
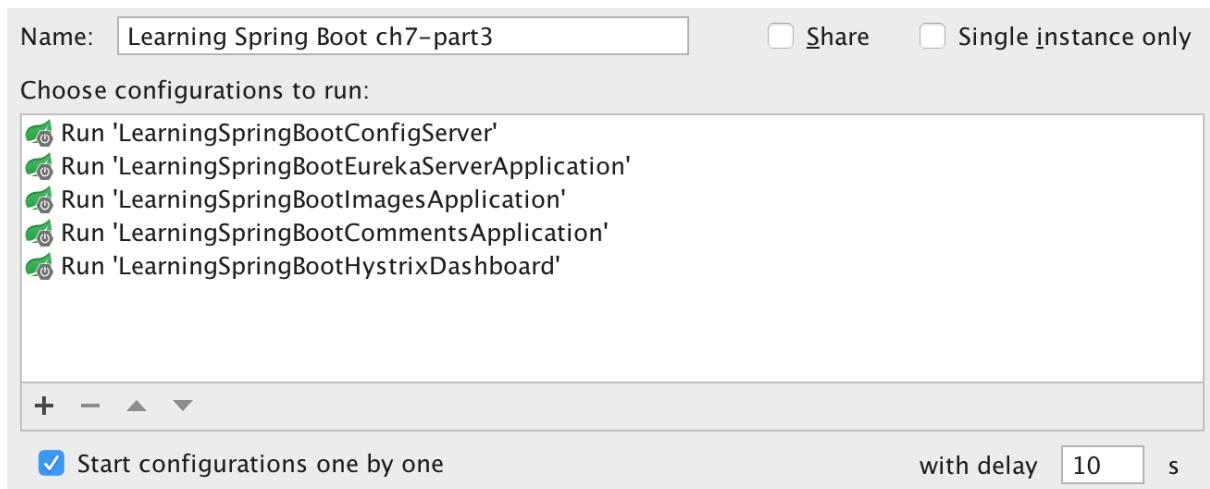




A screenshot of a browser window showing the URL 'localhost:8888/eureka/default'. The page displays a JSON configuration file:

```
// 20170813194344
// http://localhost:8888/eureka/default

{
  "name": "eureka",
  "profiles": [
    "default"
  ],
  "label": null,
  "version": "a69475c00e47bc91949e76c79d2df39720967a21",
  "state": null,
  "propertySources": [
    {
      "name": "https://github.com/gregturn/learning-spring-boot-config-repo/eureka.yml",
      "source": [
        {
          "server.port": 8761,
          "eureka.instance.hostname": "localhost",
          "eureka.client.registerWithEureka": false,
          "eureka.client.fetchRegistry": false,
          "eureka.client.serviceUrl.defaultZone": "http://${eureka.instance.hostname}:${server.port}/eureka/"
        }
      ]
    }
  ]
}
```



```
:or.: Fetching config from server at: http://localhost:8888
:or.: Located environment: name=comments, profiles=[default], label=master, version=1.0.0
:or.: Located property source: CompositePropertySource [name='configService', ...]
:or.: No active profile set, falling back to default profiles: default
```

Chapter 8: WebSockets with Spring Boot



Name	Image		
B05771_MockupCover_Normal_.jpg		<button>Delete</button>	<input type="text" value="Nice cover!"/> <button>Submit</button>
bazanga.png		<button>Delete</button>	<input type="text"/> <button>Submit</button>

Name	Image		
B05771_MockupCover_Normal_.jpg		<button>Delete</button>	<ul style="list-style-type: none">• Nice cover! <input type="text"/> <button>Submit</button>
bazanga.png		<button>Delete</button>	<input type="text"/> <button>Submit</button>

```
Received {"id":"599b90cec4d95697c2bea08b","imageId":"0a84bee9-f441-48bf-9c6a-558dab29c9d2","comment":"Nice cover!"}!
```

No file chosen

Greetings!

2f05fa8e: Do you like the new cover?
298b3bcf: You bet! Wish I could get it in a t-shirt
2f05fa8e: Me too. Did you get e-book, or print?
298b3bcf: Both
298b3bcf: I always get both

Publishing "Do you like the new cover?"

Received 2f05fa8e: Do you like the new cover?

Received 298b3bcf: You bet! Wish I could get it in a t-shirt

Publishing "Me too. Did you get e-book, or print?"

Received 2f05fa8e: Me too. Did you get e-book, or print?

Received 298b3bcf: Both

Received 298b3bcf: I always get both

> |

Received 2f05fa8e: Do you like the new cover?

Publishing "You bet! Wish I could get it in a t-shirt"

Received 298b3bcf: You bet! Wish I could get it in a t-shirt

Received 2f05fa8e: Me too. Did you get e-book, or print?

Publishing "Both"

Received 298b3bcf: Both

Publishing "I always get both"

Received 298b3bcf: I always get both

>

Learning Spring Boot - 2nd Edition



Greetings!

(greg)(all): Do you like the cover?



Greetings!

(greg)(all): Do you like the cover?

(oliver)(all): I sure do



Greetings!

(greg)(all): Do you like the cover?
(oliver)(all): I sure do

@greg Are you going to offer it in a t-shirt?

Send

Greetings!

(greg)(all): Do you like the cover?
(oliver)(all): I sure do
(phil): @greg Are you going to offer it in a t-shirt?

|

Send

Greetings!

(greg)(all): Do you like the cover?
(oliver)(all): I sure do

|

Send

Publishing "Do you like the new cover?"

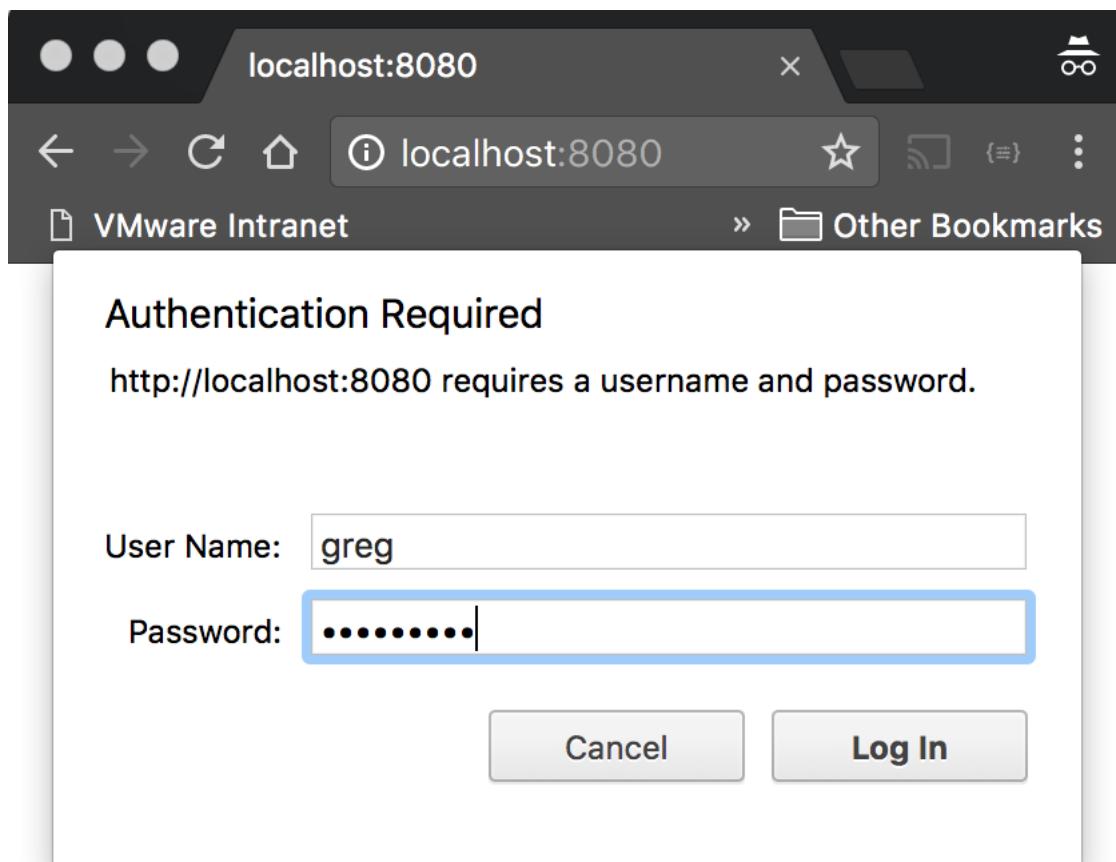
Received (greg)(all): Do you like the new cover?

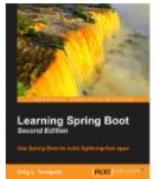
Received (oliver)(all): I sure do.

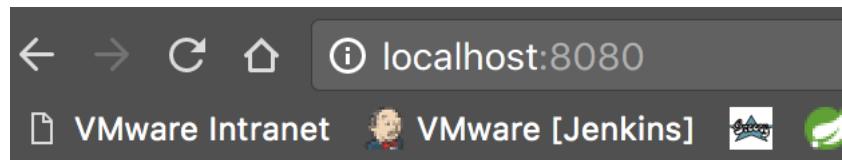
Received (phil): @greg Are you going to offer it in a t-shirt?

> |

Chapter 9: Securing Your App with Spring Boot



Name	Image		Owner	C
bazinga.png		<button>Delete</button>	greg	
B05771_MockupCover_Normal.png		<button>Delete</button>	phil	



greg has [ROLE_ADMIN, ROLE_USER]

Learning Spring Boot

	Image		Owner	(
		<input type="button" value="Delete"/>	greg	
ng		<input type="button" value="Delete"/>	phil	

	Image		Owner	(
			greg	
,		<input type="button" value="Delete"/>	phil	

x Headers Preview Response Cookies Timing

▼ General

Request URL: http://localhost:8080/images/bazinga.png/raw

Request Method: GET

Status Code: 200

Remote Address: [::1]:8080

▼ Response Headers [view source](#)

Content-Type: image/jpeg

Date: Thu, 24 Nov 2016 06:18:44 GMT

Transfer-Encoding: chunked

X-Application-Context: chat:8080

▼ Request Headers [view source](#)

Accept: image/webp,image/*,*/*;q=0.8

Accept-Encoding: gzip, deflate, sdch, br

Accept-Language: en-US,en;q=0.8

Authorization: Basic Z3JlZzp0dXJucXVpc3Q=

Cache-Control: no-cache

Connection: keep-alive

Cookie: SESSION=8ab12c94-dd78-438e-a483-b75e8f37629f

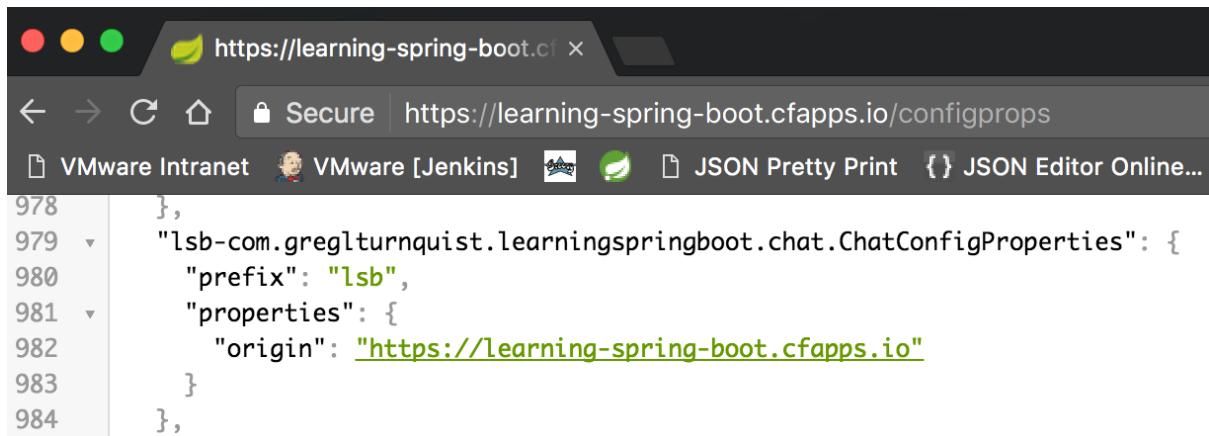
Host: localhost:8080

Pragma: no-cache

Referer: http://localhost:8080/

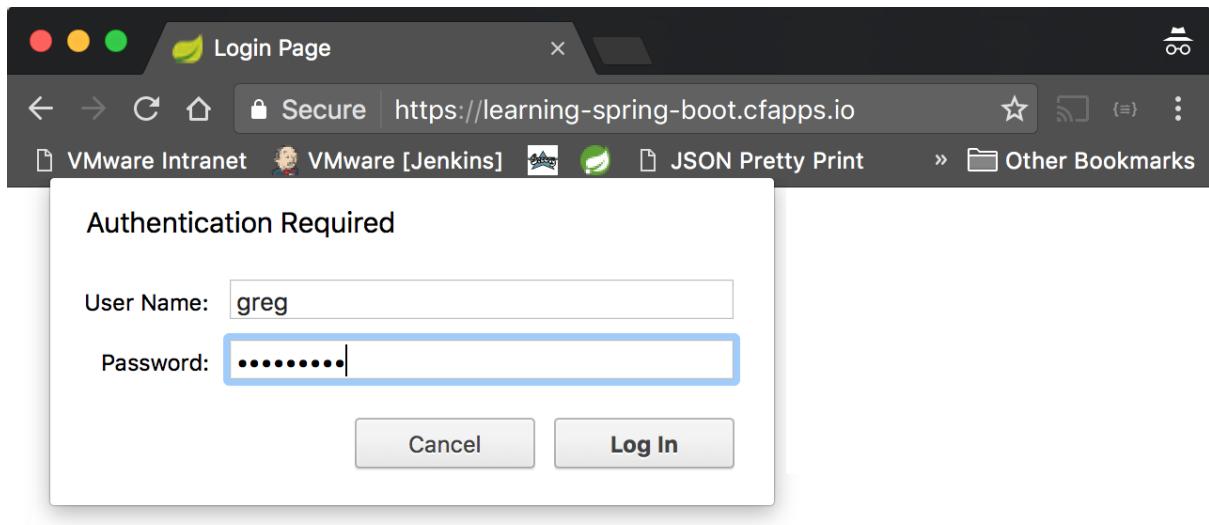
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_5) Ap

Chapter 10: Taking Your App to Production with Spring Boot



A screenshot of a web browser window titled "https://learning-spring-boot.cf". The address bar shows the URL "https://learning-spring-boot.cfapps.io/configprops". The page content is a JSON document with the following code:

```
978 },
979 "lsb-com.greglturnquist.learningspringboot.chat.ChatConfigProperties": {
980     "prefix": "lsb",
981     "properties": {
982         "origin": "https://learning-spring-boot.cfapps.io"
983     }
984 },
```



Secure https://learning-spring-boot.cfapps.io

VMware Intranet VMware [Jenkins] Other Bookmarks

Learning Spring Boot - 2nd Edition

Id	Name	Image	
8d22f555-f830-4d38-aff0-4d11562013eb	B05771_MockupCover_Normal.png		<button data-bbox="1192 698 1298 732">Delete</button>
37f6c64e-7be3-4bbb-9e6d-82271822d4b4	bazinga.png		<button data-bbox="1192 916 1298 950">Delete</button>
d1459c17-e0f6-4a2a-82f9-b7243e47bab4	platform-spring-boot.png		<button data-bbox="1192 1118 1298 1152">Delete</button>

Choose File No file chosen

Upload

Learning Spring Boot: Spring-a ×

← → ⌂ ⌄ Secure https://learning-spring-boot.cfapps.io

VMware Intranet VMware [Jenkins] JSON Pretty Print { } JSON

Greetings!

greg(channel): Wow! Are we really in production?
phil(channel): We sure are.

Cool. Can't wait to work on the next feature

Send