



Desenvolvimento de Software para WEB

Aula 19 - Spring Boot

Professor: Anderson Almada

Introdução

- O Spring Boot é um projeto da Spring que veio para facilitar o processo de configuração e publicação de nossas aplicações.
- A intenção é ter o seu projeto rodando o mais rápido possível e sem complicação.
- Basta que você diga pra ele quais módulos deseja utilizar (WEB, Template, Persistência, Segurança, etc.) que ele vai reconhecer e configurar.

Introdução

- Você escolhe os módulos que deseja através dos starters que inclui no **pom.xml** do seu projeto.
- Eles, basicamente, são dependências que agrupam outras dependências.
- Apesar do Spring Boot, através da convenção, já deixar tudo configurado, nada impede que você crie as suas customizações caso sejam necessárias.
- O maior benefício do Spring Boot é que ele nos deixa mais livres para pensarmos nas regras de negócio da nossa aplicação.

Instalação

- <https://spring.io/tools>

[PROJECTS](#)[GUIDES](#)[BLOG](#)[TRAINING & CERTIFICATION](#)

Spring Tools 4 is the next generation of Spring tooling for your favorite coding environment. Largely rebuilt from scratch, it provides world-class support for developing Spring-based enterprise applications, whether you prefer Eclipse, Visual Studio Code, or Atom IDE.

Spring com Eclipse

- Como opcional, utilize a ferramenta do spring para a criação dos projetos
 - <https://start.spring.io/>

The screenshot shows the Spring Initializr website interface. On the left, there is a sidebar with the Spring Initializr logo and the text "Bootstrap your application". Below this, there are four sections: "Project", "Language", "Spring Boot", and "Project Metadata". The "Project" section has two tabs: "Maven Project" (selected) and "Gradle Project". The "Language" section has three tabs: "Java" (selected), "Kotlin", and "Groovy". The "Spring Boot" section has five tabs: "2.2.0 M3", "2.2.0 (SNAPSHOT)", "2.1.6 (SNAPSHOT)", "2.1.5" (selected), and "1.5.21". The "Project Metadata" section has two input fields: "Group" with the value "com.example" and "Artifact" with the value "demo". At the bottom right of the sidebar, there is a "More options" button. On the right side of the page, there are links for "Github", "Twitter", and "Help". Below these links, there is a blue box with the text "Help us improve the site!" and a link "Take a quick survey".

Spring Initializr
Bootstrap your application

Project

Language

Spring Boot

Project Metadata

Maven Project

Gradle Project

Java

Kotlin

Groovy

2.2.0 M3

2.2.0 (SNAPSHOT)

2.1.6 (SNAPSHOT)

2.1.5

1.5.21

Group

com.example

Artifact

demo

More options

Github | Twitter | Help

Help us improve the site!
[Take a quick survey](#)

Help us improve the site!
[Take a quick survey.](#)

Project

Language

Spring Boot

Project Metadata

© 2013-2019 Pivotal Software
start.spring.io is powered by
[Spring Initializr](#) and [Pivotal Web Services](#)

Maven Project

Gradle Project

Java

Kotlin

Groovy

2.2.0 M3

2.2.0 (SNAPSHOT)

2.1.6 (SNAPSHOT)

2.1.5

1.5.21

Group
com.example

Artifact
demo

Name
demo

Description
Demo project for Spring Boot

Generate Project - alt + ⌘

Package Name
com.example.demo

Packaging

Jar **War**

Java Version

12 11 **8**

Fewer options

Dependencies

[See all](#)

Search dependencies to add

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

Selected dependencies

DevTools [Core]

Spring Boot Development Tools

Web [Web]

Servlet web application with Spring MVC
and Tomcat

Package Name
`com.example.demo`

Packaging
War

Java Version
8

Fewer options

Dependencies
[See all](#)

Search dependencies to add
Web, Security, JPA, Actuator, Devtools...

Selected dependencies

- DevTools** [Core]
Spring Boot Development Tools
- Web** [Web]
Servlet web application with Spring MVC and Tomcat

Generate Project - alt + ↵

Spring com Eclipse

- Extrai o arquivo o qual foi realizado o download
- Importa no Eclipse (maven)
- Espere a realização dos downloads das dependências
- **Execute como um Java Application**



Quick Access

Project Explorer

- caos-api-stand-maven
- caos-controller
- Crypto
- DateServer
- demo
 - Deployment Descriptor: demo
 - JAX-WS Web Services
 - Java Resources
 - src/main/java
 - com.example.demo
 - DemoApplication.java
 - DemoApplication
 - ServletInitializer.java
 - com.example.demo.model
 - Curso.java
 - src/main/resources
 - src/test/java
 - Libraries
 - JavaScript Resources
 - Deployed Resources
 - src
 - target
 - HELP.md

demo/pom.xml DemoApplication ServletInitializer mvnw Tomcat v8.0 Ser HELP.md

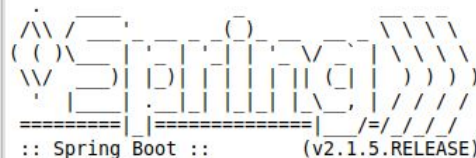
Curso.java

1 package com.example.demo.model;

Markers Properties Servers Data Source Explorer Snippets Problems Console

DemoApplication [Java Application] /usr/lib/jvm/java-8-oracle/bin/java (25 de mai de 2019 10:14:09)

```
2019-05-25 14:38:37.866 INFO 8070 --- [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s):
2019-05-25 14:38:37.867 INFO 8070 --- [ restartedMain] com.example.demo.DemoApplication : Started DemoApplication in
2019-05-25 14:38:37.869 INFO 8070 --- [ restartedMain] .ConditionEvaluationDeltaLoggingListener : Condition evaluation unchar
2019-05-25 14:49:58.444 INFO 8070 --- [ Thread-16] o.s.s.concurrent.ThreadPoolTaskExecutor : Shutting down ExecutorServ
```



```
2019-05-25 14:49:58.599 INFO 8070 --- [ restartedMain] com.example.demo.DemoApplication : Starting DemoApplication or
2019-05-25 14:49:58.600 INFO 8070 --- [ restartedMain] com.example.demo.DemoApplication : No active profile set, fall
2019-05-25 14:49:58.847 INFO 8070 --- [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with po
2019-05-25 14:49:58.849 INFO 8070 --- [ restartedMain] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2019-05-25 14:49:58.849 INFO 8070 --- [ restartedMain] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [A
2019-05-25 14:49:58.855 INFO 8070 --- [ restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedde
2019-05-25 14:49:58.856 INFO 8070 --- [ restartedMain] o.s.web.context.ContextLoader : Root WebApplicationContext:
2019-05-25 14:49:58.949 INFO 8070 --- [ restartedMain] o.s.s.concurrent.ThreadPoolTaskExecutor : Initializing ExecutorServic
2019-05-25 14:49:59.002 INFO 8070 --- [ restartedMain] o.s.b.d.a.OptionalLiveReloadServer : LiveReload server is runnin
2019-05-25 14:49:59.034 INFO 8070 --- [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s):
2019-05-25 14:49:59.035 INFO 8070 --- [ restartedMain] com.example.demo.DemoApplication : Started DemoApplication in
2019-05-25 14:49:59.036 INFO 8070 --- [ restartedMain] .ConditionEvaluationDeltaLoggingListener : Condition evaluation unchar
```

Properties para o upload de imagens

- resources/uploads.properties
 - folder=/var/www/vue/vue-spring/uploads/

Crie a classe model da entidade

- Curso
 - Id
 - Nome
 - Duração
 - <https://pastebin.com/EbRmbb1F>

Crie a classe service da entidade

- `public class CursoService {`

Adicione a anotação para REST

- **@RestController**
- `public class CursoService {`

Adicione a anotação para o mapeamento

- @RestController
- **@RequestMapping(path = "/api/cursos")**
- public class CursoService {

Adicione a anotação para CROS

- @RestController
- @CrossOrigin
- @RequestMapping(path = "/api/cursos")
- public class CursoService {

Persistência em memória

```
private Map<Integer, Curso> cursos;  
int i = 3;  
public CursoService() {  
    cursos = new HashMap<Integer, Curso>();  
  
    Curso c1 = new Curso(1, "Workshop Rest", "24hs");  
    Curso c2 = new Curso(2, "Workshop Spring MVC", "24hs");  
    Curso c3 = new Curso(3, "Desenvolvimento Web com JSF 2", "60hs");  
  
    cursos.put(1, c1);  
    cursos.put(2, c2);  
    cursos.put(3, c3);  
}
```

Métodos para upload de imagens

- <https://pastebin.com/Wkm1APMU>

GET ALL - Spring

```
@RequestMapping(method = RequestMethod.GET)
public ResponseEntity<List<Curso>> getCursos() {
    return new ResponseEntity<List<Curso>>(new ArrayList<Curso>(cursos.values()), HttpStatus.OK);
}
```

GET - Spring

```
@RequestMapping(value = "{id}", method = RequestMethod.GET)
public ResponseEntity<Curso> getCurso(@PathVariable("id") Integer id) {
    Curso curso = cursos.get(id);

    if (curso == null) {
        return new ResponseEntity<>(HttpStatus.NOT_FOUND);
    }

    return new ResponseEntity<Curso>(curso, HttpStatus.OK);
}
```

POST - Spring

```
@RequestMapping(method = RequestMethod.POST)
public ResponseEntity<Curso> addCurso(String nome, String duracao, MultipartFile image) {
    if (nome == null || duracao == null || nome.equals("null") || duracao.equals("null") || image == null) {
        return new ResponseEntity<>(HttpStatus.NOT_FOUND);
    }
    int id = ++i;
    Curso curso = new Curso(id, nome, duracao);
    cursos.put(id, curso);
    try {
        uploadFile(image.getInputStream(), id);
    } catch (IOException e) {
        e.printStackTrace();
    }
    return new ResponseEntity<Curso>(curso, HttpStatus.OK);
}
```

PUT - Spring

```
@RequestMapping(value = "{id}", method = RequestMethod.PUT)
public ResponseEntity<Curso> atualizarCurso(
    @PathVariable("id") int id, String nome, String duracao, MultipartFile image) {
    if (nome == null || duracao == null || nome.equals("null") || duracao.equals("null")) {
        return new ResponseEntity<>(HttpStatus.NOT_FOUND);
    }
    Curso curso = new Curso(id, nome, duracao);  cursos.put(id, curso);
    try {
        if (image != null) {
            uploadFile(image.getInputStream(), id);
        }
    } catch (IOException e) {
        e.printStackTrace();
    }
    return new ResponseEntity<Curso>(curso, HttpStatus.OK);
}
```

DELETE - Spring

```
@RequestMapping(value = "{id}", method = RequestMethod.DELETE)
public ResponseEntity<?> deletarCurso(@PathVariable("id") int id) {
    if(cursos.containsKey(id)) {
        cursos.remove(id);
        return new ResponseEntity<>(HttpStatus.NO_CONTENT);
    } else {
        return new ResponseEntity<>(HttpStatus.NOT_FOUND);
    }
}
```

CursoService

- <https://pastebin.com/vjHQNMXi>

Vue-spring (client)

- https://github.com/andersonalmada/desenvolvimento_web/blob/master/extra/vue-spring.zip

Links importantes

- <https://spring.io/projects>
- <https://start.spring.io/>
- <https://blog.algaworks.com/spring-boot/>
- <https://blog.algaworks.com/como-criar-web-services-restful-com-spring-boot>



Dúvidas??

E-mail: almada@crateus.ufc.br