

Olá, tudo bem?

Na videoaula passada, você conheceu o Spring e seu ecossistema. Além disso, foi feito uso do componente Spring MVC. Agora, dando continuidade aos estudos, será abordado tudo sobre o Spring Boot. Com isso, será possível identificá-lo, realizar a criação de um projeto, a partir do Spring Initializr, e realizar a inclusão de um recurso, ou funcionalidade, no Spring Boot, que será acessado a partir da raiz do servidor web embarcado (Tomcat).

Sabendo disso, o conceito do Spring Boot pode ser explicado da seguinte forma: ele é um componente que fornece uma variedade de ferramentas para a construção rápida e de fácil configuração de aplicações baseadas no Spring e no seu ecossistema. Dessa forma, sua arquitetura é fundamentada no Spring Core, que proporciona mecanismo de injeção de dependências, de consulta e de manipulação de objetos em tempo de execução (Spring Expression Language - SpEL), assim como algumas funcionalidades de programação orientada a aspectos.

Portanto, o Spring Boot é praticamente um template. Por isso, você pode agregar a ele vários outros componentes que fazem parte do ecossistema do Spring. Sabendo que todos eles vêm pré-configurados para que não haja nenhum esforço para configuração e execução. Além disso, o Spring Boot ainda traz, integrado a ele, um servidor web embarcado já configurado e pronto para utilização, ou seja, o Tomcat, que será usado para inicializar e gerenciar as aplicações.

Diante disso, como você deve fazer para construir um projeto baseado no Spring Boot? Bem, fique tranquilo, pois, ao longo desta aula, essa pergunta será respondida. Vamos lá?

Então, para que seja possível construir um projeto baseado no Spring Boot é necessário utilizar o Spring Initializr. Trata-se de uma ferramenta online com o objetivo de criar projetos Maven pré-configurados baseados no Spring Boot. Ademais, consiste em uma forma de facilitar a vida dos desenvolvedores, dando mais flexibilidade para que eles montem seus projetos de acordo com as suas necessidades.

Para isso, a utilização do Spring Initializr se dá a partir do acesso a URL <https://start.spring.io/> (https dois pontos barra barra start ponto spring ponto io barra) no seu navegador.

#Audiodescrição: A imagem mostra a captura da tela inicial do “spring inicializr”. Nela, há as opções de configuração da ferramenta à esquerda e, à direita, ficam as dependências.

Agora, você irá criar um projeto Spring Boot, informando as propriedades e dependências necessárias para o seu desenvolvimento. Elas são: Project, Language, Spring Boot, Project Metadata, Packaging e Java. Posteriormente, você conhecerá as dependências. A seguir, será abordado como configurar cada uma delas. Vamos lá!

Ainda na página inicial do Spring Initializr, você deve selecionar a opção projeto Maven no campo Project. Depois, basta informar qual a linguagem a ser utilizada para desenvolver seu projeto. Para isso, selecione a opção Java do campo Language. Nessa etapa, você indicará qual a versão do Spring Boot será utilizada em seu projeto. Para isso, selecione a opção 2.3.0 (dois ponto três ponto zero) do campo Spring Boot.

O campo Project Metadata tem como objetivo descrever algumas características do seu projeto para o Maven. Portanto, ele é composto pelos seguintes subcampos: group, artifact, name, description, package, packaging e java, que são responsáveis por descreverem como o projeto será estruturado e organizado pelo Maven, assim como o modo de empacotamento e a versão do Java a ser utilizada.

Dessa forma, no subcampo Group, você irá inserir o valor br.com.lead (br ponto com ponto lead). Lembrando que esse subcampo diz respeito ao nome do grupo no qual o seu projeto faz parte. Já para o subcampo Artifact, deve-se inserir o valor springdemo, pois esse subcampo faz referência ao nome do artefato a ser criado.

Agora, no subcampo Name, que é referente ao nome do projeto, você deve inserir o valor springdemo. Porém, para o subcampo Description, insira o valor Demonstrado na criação de um projeto Spring Boot, pois esse subcampo diz respeito a finalidade do projeto.

No subcampo Package name, que é referente ao pacote do projeto, adicione o valor br.com.lead.springdemo (br ponto com ponto lead ponto springdemo). E, no subcampo Packaging, que indica a forma de empacotamento realizada pelo Maven, em seu projeto, selecione o valor Jar.

Por fim, no subcampo Java, que é referente a versão do Java que você adotará para seu projeto, selecione o valor 8.

Até aqui, você realizou a configuração das propriedades do Spring Initialz. Portanto, nesse momento, configure as Dependencies. Ainda na página inicial do Spring Initialz, navegue até a opção ADD DEPENDENCIES e acione essa opção. Em seguida, uma tela de popup será aberta, e, então, vai ser necessário pesquisar pela dependência denominada Spring Web. Na parte superior dessa janela, é possível digitar a expressão Spring Web no campo de busca. Depois, acione a opção desejada. Pronto, ela será adicionada ao seu projeto!

#Audiodescrição: A imagem mostra a tela de popup aberta, com a dependência denominada Spring Web selecionada.

#Audiodescrição: A imagem mostra a captura da tela inicial do “spring inicializ”. Nela, há as opções de configuração da ferramenta à esquerda, e à direita, o campo dependências, agora com a dependência “Spring Web” sendo exibida.

Por fim, acione o botão Generate e, automaticamente, o Spring Initializr irá criar o projeto e realizar o download.

O próximo passo é verificar se o seu projeto foi configurado corretamente. Para isso, importe o projeto para o Eclipse e depois execute-o. Assim, navegue no Eclipse até o menu File, escolha a alternativa Import e realize a importação do projeto, a partir da opção Existing Maven Projects. Feito isso, automaticamente, o Maven irá montar a estrutura necessária e incluir as dependências.

Com o projeto pronto para desenvolvimento e execução, agora, você irá incluir o recurso, ou a funcionalidade, na classe SpringdemoApplication, que se encontra no pacote `br.com.lead.springdemo` (`br ponto com ponto lead ponto springdemo`). O objetivo disso é apresentar uma mensagem de “Bem-vindo ao Spring Boot” a ser acessada via URL <http://localhost:8080> (`http dois pontos barra barra localhost dois pontos 8080`). Para isso, a classe SpringdemoApplication deve ser anotada com `@Controller` (`arroba controller`), que vai indicar que se trata de um controlador de requisições do Spring. Além disso, declare um método chamado `bemVindoAoSpring`, que retorna um String, ou seja, a mensagem Spring boot em execução. Esse método não deve possuir parâmetros. Em seguida, com o método criado, anote-o com `@RequestMapping` (`arroba request mapping`) e defina o seu valor, ou seja, o caminho / (`barra`), em que a barra indica a raiz do servidor. Portanto, existirá um

endpoint, que responderá às requisições vindas desse caminho. Assim, o código a seguir apresenta a classe SpringdemoApplication.java com o método responsável por mostrar a mensagem de bem-vindo ao acessar a URL <http://localhost:8080> (http dois pontos barra barra localhost dois pontos 8080).

#código

```
package br.com.lead.springboottest;

import org.springframework.boot.SpringApplication;

@SpringBootApplication
@Controller

public class SpringBootTestApplication {

    @RequestMapping("/")

    @ResponseBody

    public String bemVindoAoSpring(){

        return "Bem-vindo ao Spring Boot.";

    }

    public static void main(String[] args){

        SpringApplication.run(SpringBootTestApplication.class, args);

    }

}
```

Agora que você já sobrescreveu a classe SpringdemoApplication, execute o seu projeto. Para isso, faça uso de uma aplicação Java.

Dessa maneira, qualquer projeto, que tenha como base o Spring Boot, possui uma classe de inicialização que contém o famoso método main, sendo ele o encarregado por executar as aplicações.

No projeto criado por você, a partir do Spring Initializr, a classe, onde se encontra o método main, é a SpringdemoApplication. Portanto, navegue até a opção Run As e escolha a opção Java Application. Pronto! Sua aplicação será inicializada a partir do Spring Boot.

Para concluir essa etapa, acesse a URL <http://localhost:8080> (http dois pontos barra barra localhost dois pontos 8080), no seu navegador, para que seja exibida a mensagem “Bem-vindo ao Spring Boot”. Na prática, a ideia é que você crie um recurso, ou uma funcionalidade, e o disponibilize para ser acessado, a partir da raiz do servidor web embarcado no Spring Boot, que, no seu caso, é o Tomcat.

#A imagem mostra a captura de tela da barra de endereço preenchida pela URL “localhost:8080”.

Etapa concluída! Nesta videoaula, você aprendeu como criar um projeto baseado no Spring Boot, a partir do Spring Initializr. Além de criar um recurso chamado bemVindoAoSpring, e pode-lo acessá-lo pela URL <http://localhost:8080> (http dois pontos barra barra localhost dois pontos 8080). Na próxima videoaula, você irá conhecer os métodos HTTP e como criar uma API REST no Spring.

Até a próxima aula e bons estudos!

Referências

Apache Tomcat. Disponível em: <https://pt.wikipedia.org/wiki/Apache_Tomcat>. Acesso em: 10 jun 2020.

Container (tipo de dado abstrato). Disponível: <[https://pt.wikipedia.org/wiki/Container_\(tipo_de_dado_abstrato\)](https://pt.wikipedia.org/wiki/Container_(tipo_de_dado_abstrato))>. Acesso em: 10 jun 2020.

Disponível: <<https://medium.com/java-e-arquitetura-de-forma-mastigada/iniciando-com-spring-boot-f82e2a456204>>. Acesso em: 10 jun 2020.

PERRY, J. Steven. **Informações básicas sobre o Spring Boot**. Disponível em: <<https://www.ibm.com/developerworks/br/library/j-spring-boot-basics-perry/index.html>>.

<https://blog.geekhunter.com.br/tudo-o-que-voce-precisa-saber-sobre-o-spring-boot/>>. Acesso em: 10 jun 2020.

Spring Boot. Disponível em: <<https://spring.io/projects/spring-boot>>. Acesso em: 10 jun 2020.

Spring Boot: Tudo que você precisa saber!. Disponível em:
<<https://blog.geekhunter.com.br/tudo-o-que-voce-precisa-saber-sobre-o-spring-boot/>>.

Acesso em: 10 jun 2020.

Spring Expression Language (SpEL). Disponível em:
<<https://docs.spring.io/spring/docs/4.3.10.RELEASE/spring-framework-reference/html/expressions.html>>. Acesso em: 10 jun 2020.

Spring Initializr. Disponível em: <<https://start.spring.io/>>. Acesso em: 10 jun 2020.

VIEIRA, Marcelo de Souza. **Iniciando com Spring Boot**. Disponível em:
<<https://medium.com/java-e-arquitetura-de-forma-mastigada/iniciando-com-spring-boot-f82e2a456204>>. Acesso em: 10 jun 2020.