

Tutorial SpringBoot com JPA, Hibernate, H2, Derby e Rest

Parte I

Glauco Todesco

Download:

<https://github.com/glaucotodesco/CursoSpringBootComAngular/tree/Tutorial-ParteI>

1. Gere um projeto com as características abaixo:

Acesse: <https://start.spring.io/>

The screenshot shows the Spring Initializr web application. At the top, it says "SPRING INITIALIZR bootstrap your application now". Below this, there's a header "Generate a" followed by a dropdown menu set to "Maven Project", then "with" followed by a dropdown menu set to "Java", and "and Spring Boot" followed by a dropdown menu set to "1.5.16 (SNAPSHOT)".

Below the header, there are two main sections: "Project Metadata" and "Dependencies".

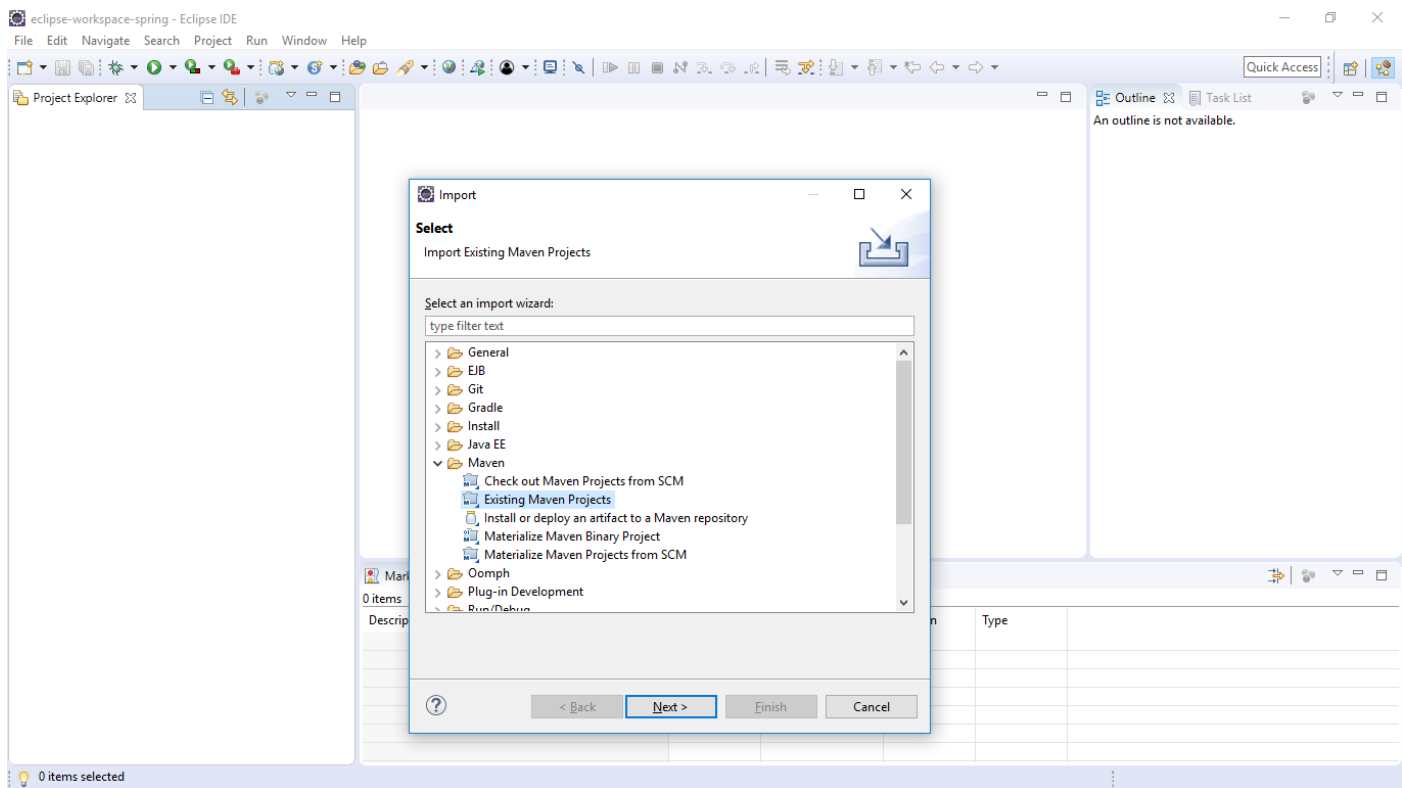
Project Metadata: It has a sub-header "Artifact coordinates". Under "Group", there's a text input field containing "com.abutua.springboot". Under "Artifact", there's a text input field containing "HelloSpringBootRest".

Dependencies: It has a sub-header "Add Spring Boot Starters and dependencies to your application". Below this is a "Search for dependencies" section with a text input field containing "Web, Security, JPA, Actuator, Devtools...". Below that is a "Selected Dependencies" section showing five green buttons: "DevTools", "Web", "H2", "Apache Derby", and "JPA". Each button has a small "x" icon to its right.

At the bottom of the form, there's a green button labeled "Generate Project" with a keyboard shortcut "alt + ⌘". Below the button, there's a link: "Don't know what to look for? Want more options? [Switch to the full version.](#)"

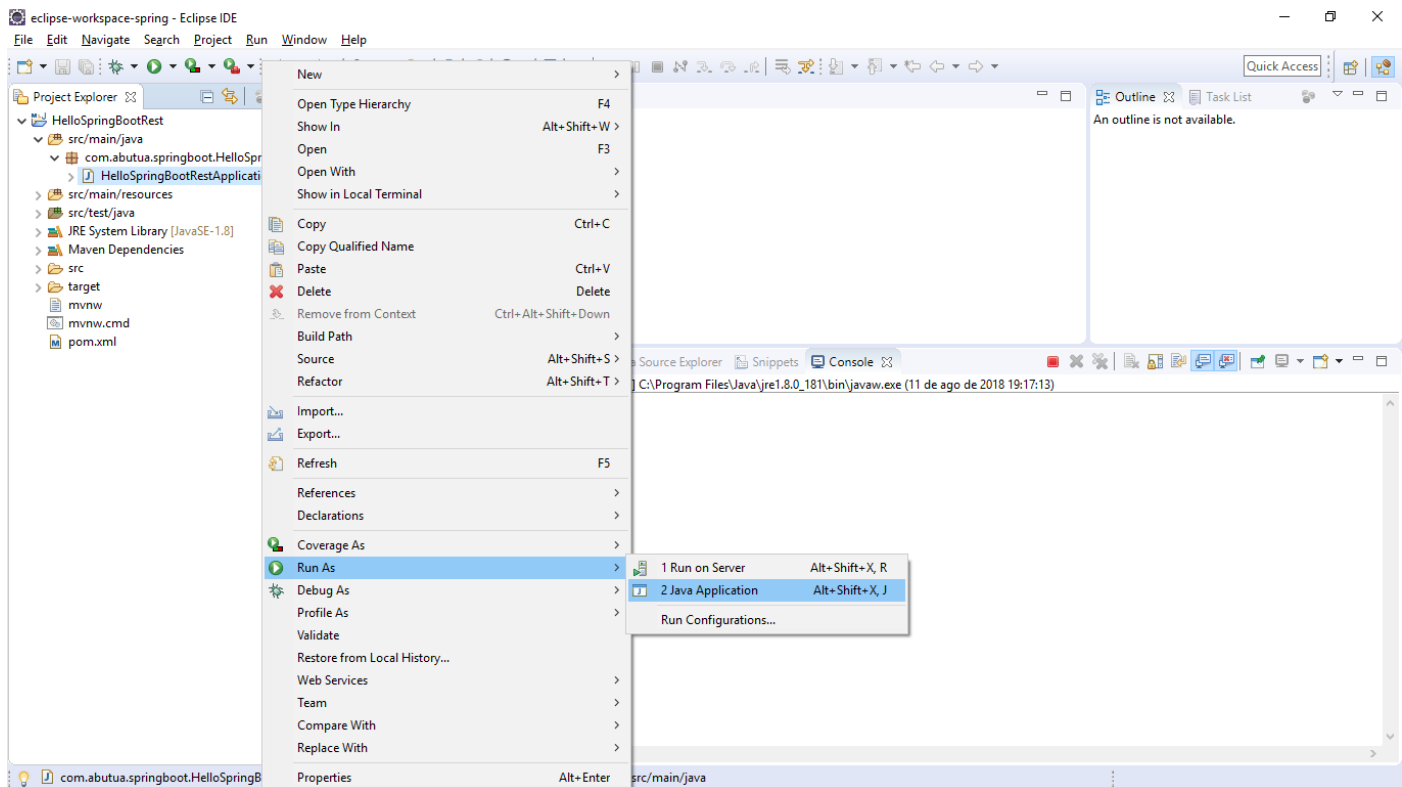
At the very bottom of the page, there's a footer: "start.spring.io is powered by [Spring Initializr](#) and [Pivotal Web Services](#)".

2. Descompacte o arquivo e importe dentro de Eclipse.
Menu: File->Import->Maven->Existing Maven Projects

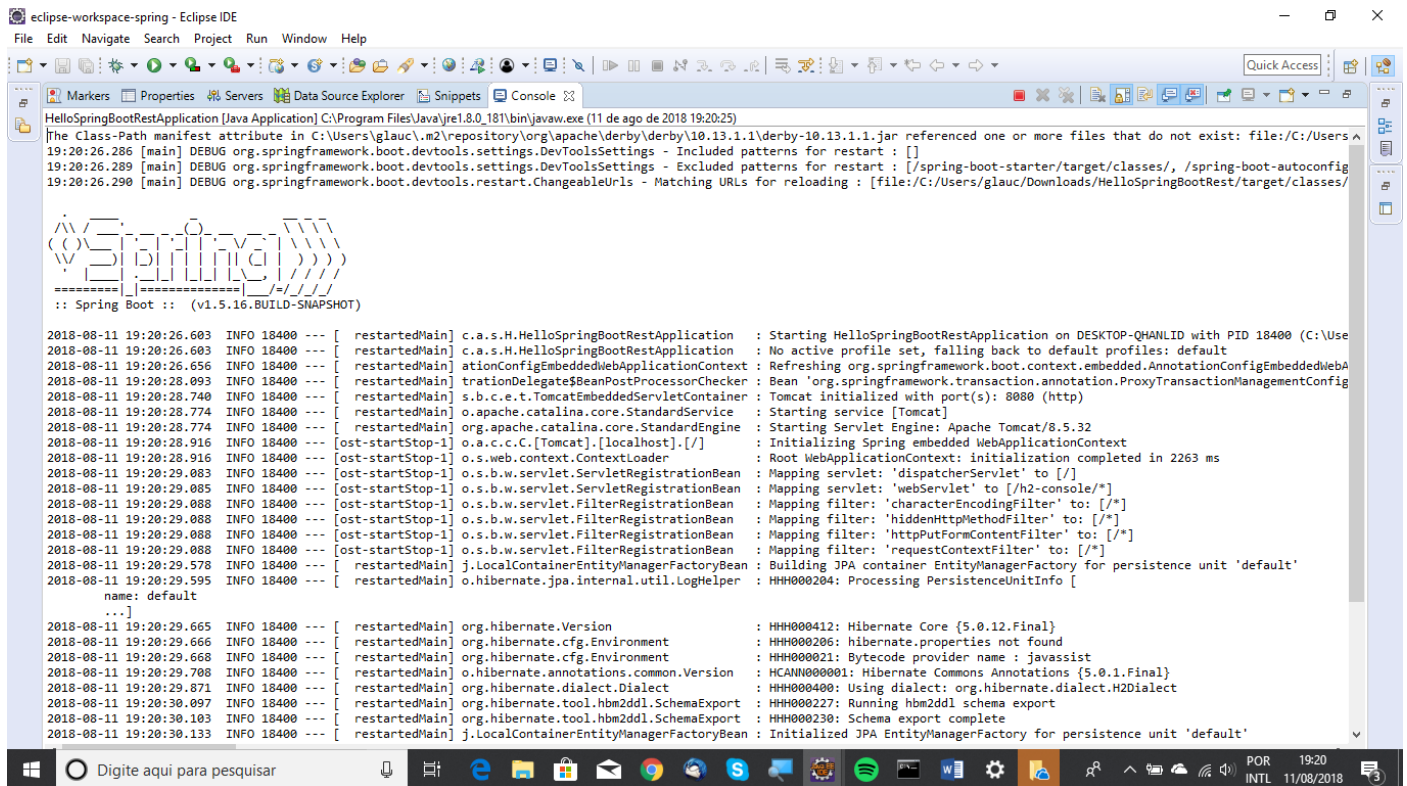


3. Espere o Eclipse fazer o download de todas as dependências do Maven. Isso pode demorar um pouco. Acompanhe o download clicando no canto inferior direito do Eclipse, na barra de progresso.

4. Execute a única classe do projeto para verificar se tudo está funcionando corretamente:



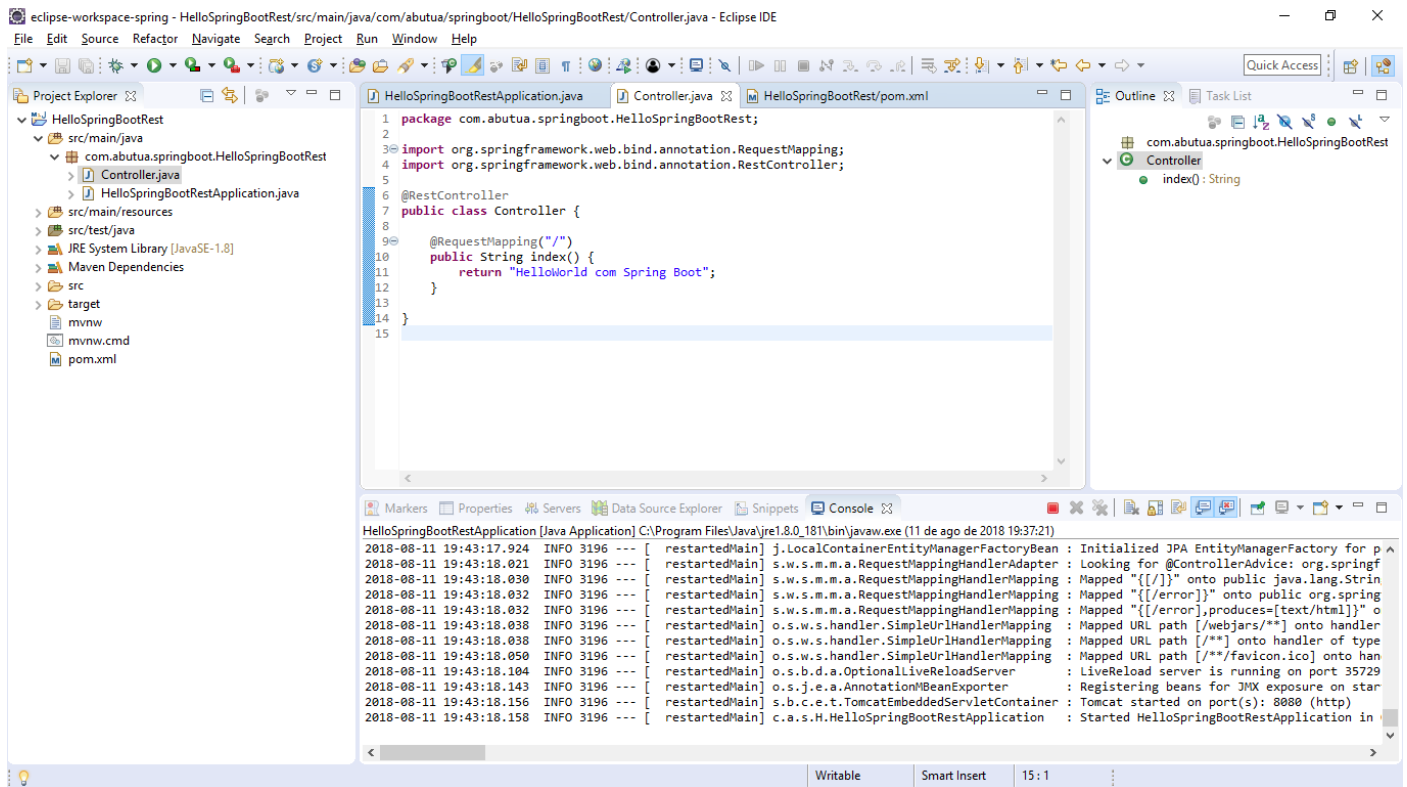
5. Se tudo estiver correto não teremos erros na console do Eclipse:



6. Se você abrir o navegador com o endereço “localhost:8080”, a seguinte tela será apresentada:



7. Vamos criar um serviço Rest simple. Para isso construa uma nova classe e codifique a mesma dessa forma apresentada da figura abaixo. Essa classe deve ficar no mesmo pacote da classe já existente no projeto.



8. Após a aplicação ser reiniciada automaticamente, teremos a seguinte tela no navegador:

