

Guia De Desenvolvimento
Spring MVC
Thymeleaf
Spring Boot
Git
2023
Parte 3

Introdução

Neste documento teremos o passo-a-passo para a construção de uma aplicação web utilizando as ferramentas Spring MVC, Thymeleaf, Spring Boot e Git.

Este guia se destina a dar suporte as aulas de LIP1 do segundo módulo do curso de informática noturno do ITB Belval, portanto, algumas explicações mais detalhadas serão omitidas pois serão apresentadas pelo professor durante a aula.

Esta é a segunda parte do guia e para acompanhá-la é necessário obter o projeto desenvolvido na parte 1 em <https://github.com/ProfNpc/CrudExampleParte3/tree/main/inicial>.

Parte 3

Nessa terceira parte de nosso guia continuaremos a implementar as ações do crud. No sentido de recuperar informações (**Recouver**), criaremos uma tela que nos permita ver a lista dos produtos inseridos. Para tanto, se faz necessário concluir a ação de criação (**Create**) de forma que o produto cujos dados foram preenchidos no formulário (desenvolvido na parte 2 desse guia) e enviados seja armazenado em algum lugar onde nossa aplicação possa acessar no momento que ela for solicitada a apresentar a lista dos produtos.

Numa primeira versão, armazenaremos os produtos em memória criando um atributo na classe ProdutoController do tipo List<Produto>.

Sobre Listas

Classes que implementam a interface List possuem a capacidade de armazenar conjuntos ordenados de objetos, isto é, o contrato da interface List pressupõem que haja uma ordem inerente aos objetos mantidos pela por um List. Essa característica se reflete no fato de que cada objeto armazenado possui um índice associado e pode ser recuperado através desse índice.

Listando os produtos

1 – Adicione o atributo listaProdutos do tipo List<Produto> à classe ProdutoController (Veja a seguir)

2 – Adicione o atributo proxId do tipo int à classe ProdutoController

```
@Controller
public class ProdutoController {

    private static List<Produto> listaProdutos = new ArrayList<Produto>();
    private static int proxId = 1;
```

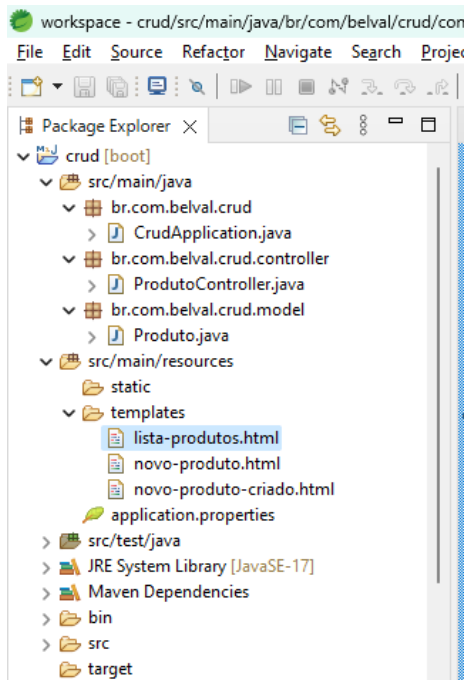
3 – Altere o método novo(Produto novo) e adicione as linhas abaixo

```
@PostMapping("/produto/novo")
public ModelAndView novo(Produto produto) {
    ModelAndView modelAndView = new ModelAndView("novo-produto-criado");
    modelAndView.addObject("novoProduto", produto);
    produto.setId(proxId++);
    listaProdutos.add(produto);
    return modelAndView;
}
```

4 – Adicione o método list() e associe ele à url “/produto/list”

```
@GetMapping("/produto/list")
public ModelAndView list() {
    ModelAndView modelAndView = new ModelAndView("lista-produtos");
    modelAndView.addObject("produtos", listaProdutos);
    return modelAndView;
}
```

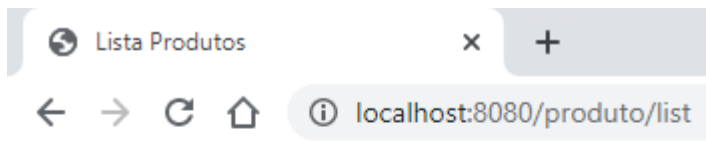
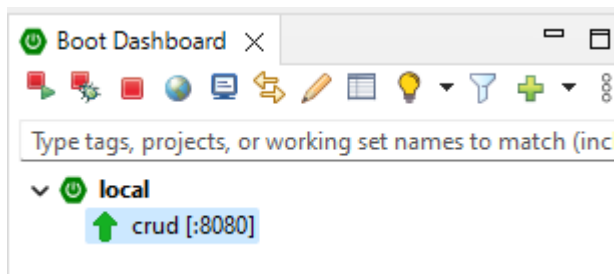
5 – Adicione o arquivo lista-produtos.html aos templates



```
<!DOCTYPE html>
<html>
  <head>
    <title>Lista Produtos</title>
    <meta charset="utf-8">
  </head>
  <body>
    <h1>Novo Produto</h1>

    <table border="1">
      <tr>
        <th>ID</th>
        <th>Nome</th>
        <th></th>
        <th></th>
      </tr>
      <tr th:each="prod : ${produtos}">
        <td>
          <span th:text="${prod.id}"></span>
        </td>
        <td>
          <span th:text="${prod.nome}"></span>
        </td>
        <td>
          <a th:href="@{/produto/{id} (id=${prod.id}) }">Detalhe</a>
        </td>
        <td>
          <a th:href="@{/produto/{id}/edit (id=${prod.id}) }">Editar</a>
        </td>
      </tr>
    </table>
    <a th:href="@{/produto/novo}">Novo</a>
  </body>
</html>
```

6 – Execute a aplicação e acesse a url <http://localhost:8080/produto/list>

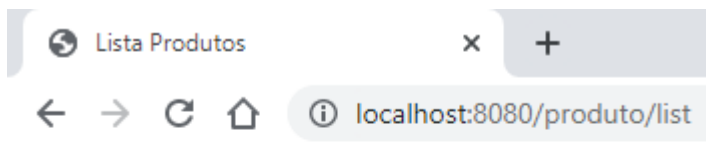


Novo Produto

ID	Nome		
----	------	--	--

[Novo](#)

7 – Clique no link “Novo”



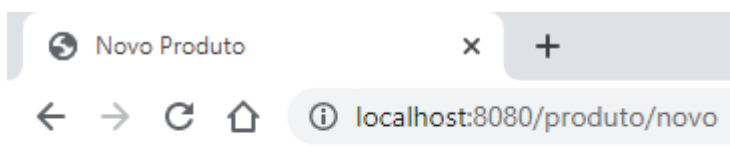
Novo Produto

ID	Nome		
----	------	--	--

[Novo](#)



8 – Preencha os campos do formulário e clique em “Enviar”

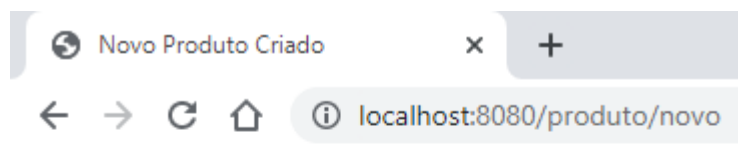


Novo Produto

Nome:

Descrição:

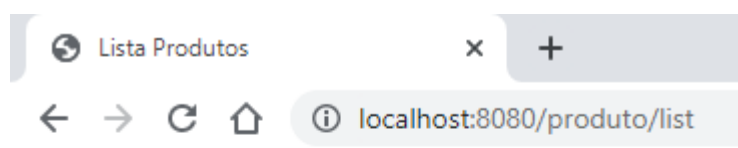
Preço:



Novo produto criado!

ID:	1
Nome:	Bala
Descrição:	Bala soft
Preço:	2.5

9 – Acesse novamente a url <http://localhost:8080/produto/list>



Novo Produto

ID	Nome		
1	Bala	Detalhe	Editar

[Novo](#)