

FIAP GRADUAÇÃO

SISTEMAS DE INFORMAÇÃO

MICROSERVICE AND WEB ENGINEERING

Prof^a. Aparecida Castello Rosa
profaparecida.rosa@fiap.com.br

■ Agenda

- Continuar com o projeto produto-mvc
- Aplicação de boas práticas.

Objetivos

- Continuar com o projeto produto-mvc
- Refatorar o código e utilizar as anotações do Spring para definir qual o método **HTTP** a ação representa.
- Refatorar o nome dos métodos.

I Recursos e URLs


- As funcionalidades/informações de um sistema web são organizadas na forma de **RECURSOS**.
- **RECURSOS** devem ser **substantivos** e **não verbos**.
- **URL** - *Universal Resource Locator*.
- **URL** deve acessar os recursos pelo nome:


GET: host:port/produtos	obter produtos
GET: host:port/produtos?page=3	obter produtos da página 3
GET: host:port/produtos/1	obter produto id 1
GET: host:port/produtos/1/categorias	obter categorias do produto id 1


Padrões de URL - Roteamento


- A ação desejada deve ser expressa pelo **verbo HTTP** e **não pela rota**.

ERRADO	CORRETO
GET : host:port/insertProduto	POST : host:port/produtos
GET : host:port/listarProdutos	GET : host:port/produtos

GET  http://localhost:8080/produtos

POST  http://localhost:8080/categorias

GET  http://localhost:8080/categorias/1

DELETE  http://localhost:8080/categorias/4

Métodos HTTP mais utilizados

Anotação (request)	Objetivo	Significado
@RequestMapping	Mapear URLs de acesso ao controller e aos métodos	
@GetMapping	Anota o método para responder como endpoint HTTP GET (select)	obter recurso
@PostMapping	Anota o método para responder como endpoint HTTP POST (insert)	criar novo recurso
@PutMapping	Anota o método para responder como endpoint HTTP PUT (update completo)	salvar recurso de forma idempotente
@PatchMapping	Anota o método para responder como endpoint HTTP PATCH (update parcial)	não é idempotente
@DeleteMapping	Anota o método para responder como endpoint HTTP DELETE (delete)	deletar recurso

Operação idempotente = não causa novos efeitos se executada mais de uma vez.

<https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Methods>

Refatorar class ProdutoController

- Alterar `@GetMapping("/novo")` para `@GetMapping("/form")`
- Alterar o nome do método `adicionarProduto` para `loadForm`


```
//URL - localhost:8080/produtos/novo  
@GetMapping("/novo")  
public String adicionarProduto(Model model) {  
    model.addAttribute("produto", new Produto())  
}
```


Refatorar class ProdutoController

```
//Form para cadastro  
//URL - localhost:8080/produtos/form  
//@GetMapping("/novo")  
@GetMapping("/form")  
public String loadForm(Model model) {  
    model.addAttribute("produto", new Produto());  
    return "produto/novo-produto";  
}
```

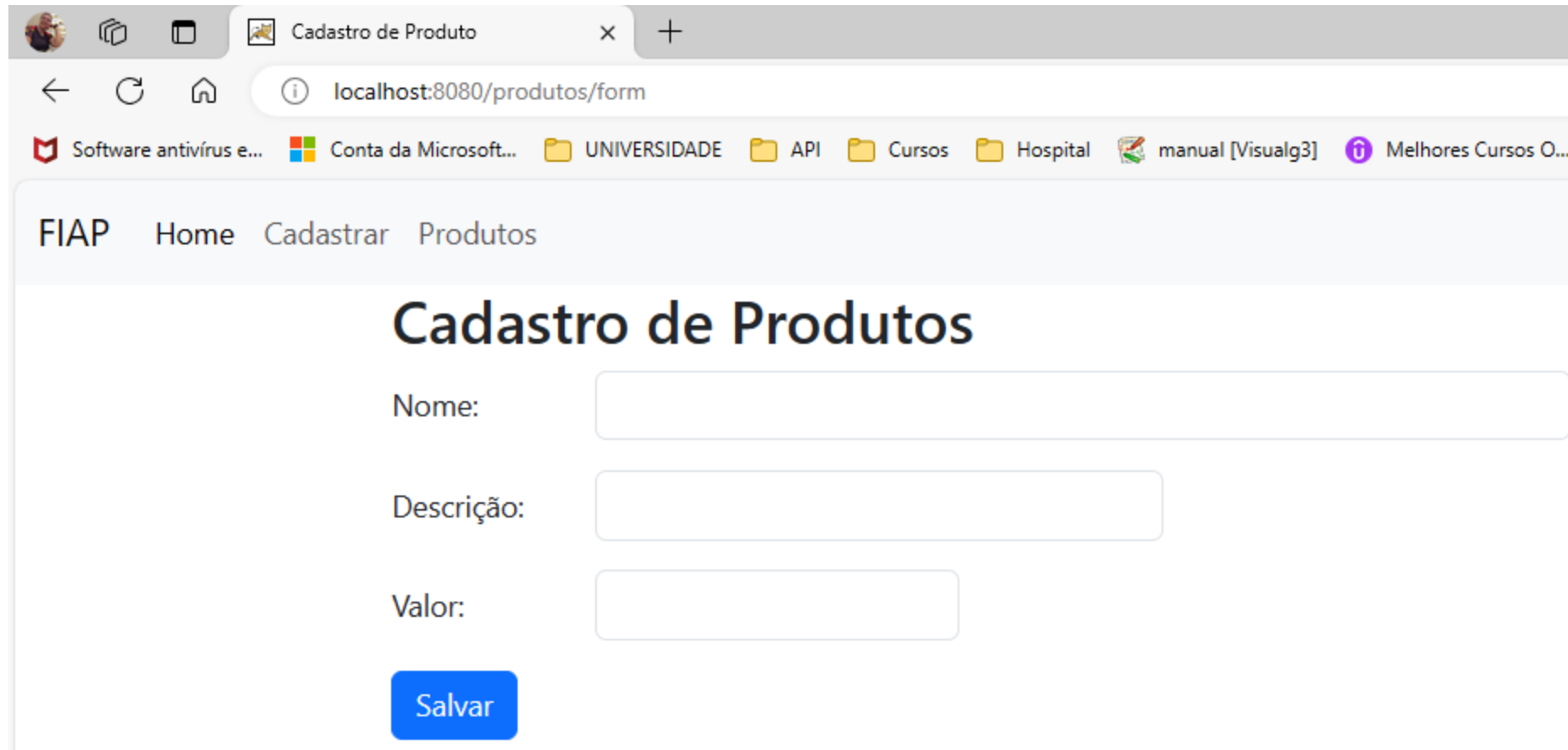
I Refatorar navbar.html

```
<li class="nav-item">  
  <a class="nav-link" href="/produtos/novo">Cadastrar</a>  
</li>
```



```
<li class="nav-item">  
  <a class="nav-link" href="/produtos/form">Cadastrar</a>  
</li>
```

■ Testar a aplicação



The screenshot shows a web browser window with the title 'Cadastro de Produto'. The address bar displays 'localhost:8080/produtos/form'. The browser's taskbar at the top shows various icons, including 'Software antivírus e...', 'Conta da Microsoft...', 'UNIVERSIDADE', 'API', 'Cursos', 'Hospital', 'manual [Visualg3]', and 'Melhores Cursos O...'. The web application has a navigation bar with links: 'FIAP', 'Home', 'Cadastrar', and 'Produtos'. The main content area is titled 'Cadastro de Produtos' and contains three input fields: 'Nome:', 'Descrição:', and 'Valor:'. Below these fields is a blue button labeled 'Salvar'.


Nome:

Descrição:

Valor:

Refatorar class ProdutoController

- Alterar `@PostMapping("/salvar")` para `@PostMapping()`
- Alterar o nome do método `insertProduto` para `insert`
- Alterar `"redirect:/produtos/novo";` para `"redirect:/produtos/form";`

```
@PostMapping("/salvar")   
@Transactional  
public String insertProduto(@Valid Produto produto,  
                             BindingResult result,  
                             RedirectAttributes attributes) {
```

Refatorar class ProdutoController

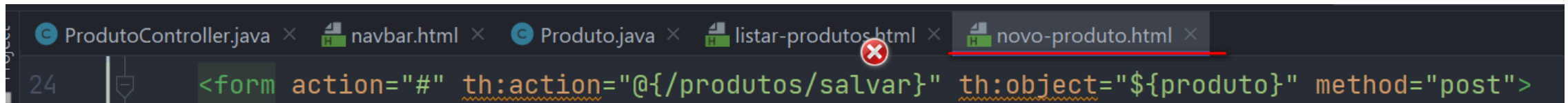
```
//@PostMapping("/salvar")
@PostMapping()
@Transactional
public String insert(@Valid Produto produto,
                      BindingResult result,
                      RedirectAttributes attributes) {

    if(result.hasErrors()){
        return "produto/novo-produto";
    }

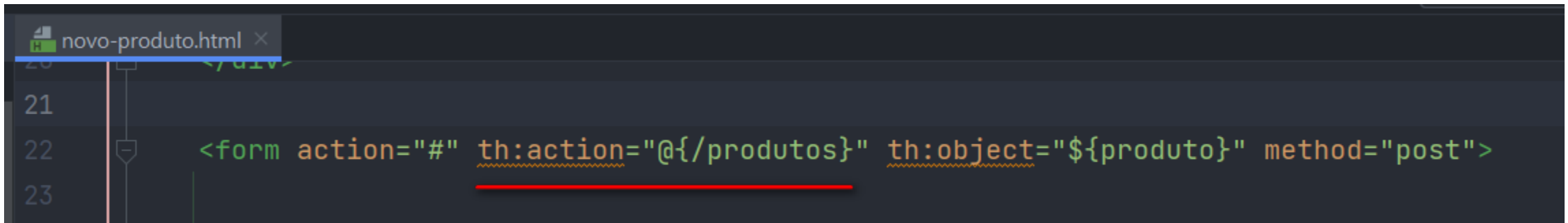
    repository.save(produto);
    attributes.addFlashAttribute("mensagem", "Produto salvo com sucesso");
    return "redirect:/produtos/form";
}
```

I Refatorar novo-produto.html

- Alterar o action do form `th:action="@{/produtos/salvar}"` para `th:action="@{/produtos}"`

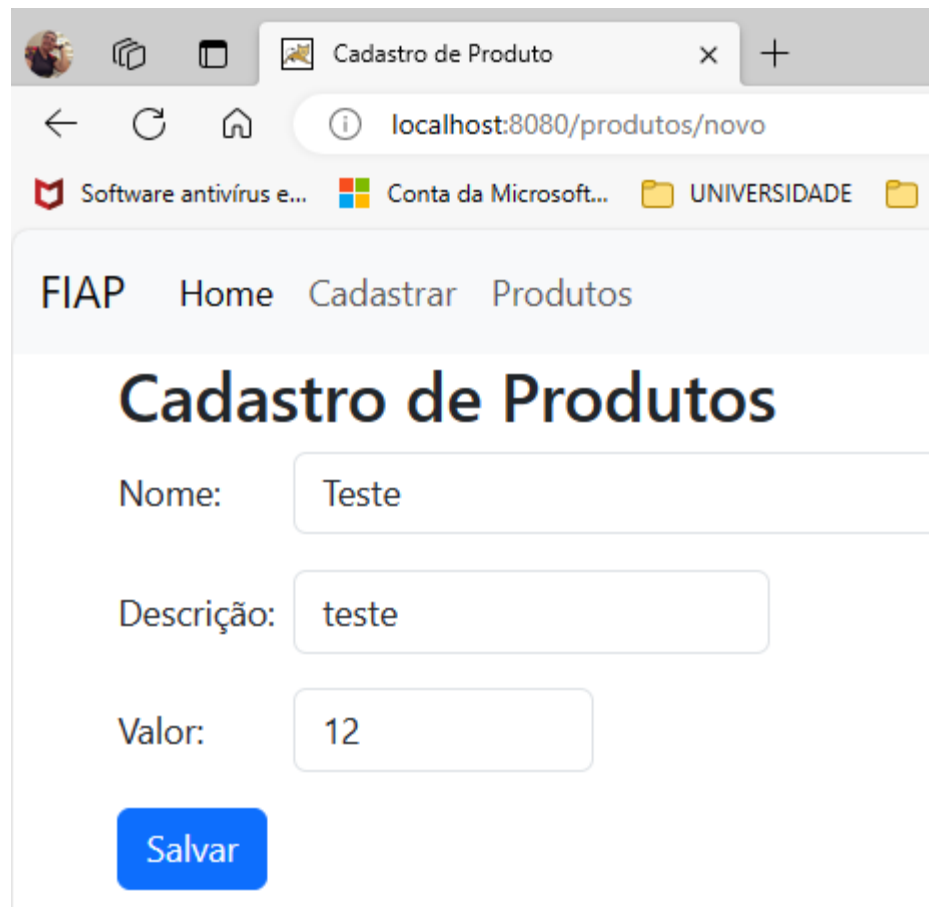


```
24 <form action="#" th:action="@{/produtos/salvar}" th:object="${produto}" method="post">
```



```
21  
22 <form action="#" th:action="@{/produtos}" th:object="${produto}" method="post">  
23
```

Testar a aplicação



Cadastro de Produto

localhost:8080/produtos/novo

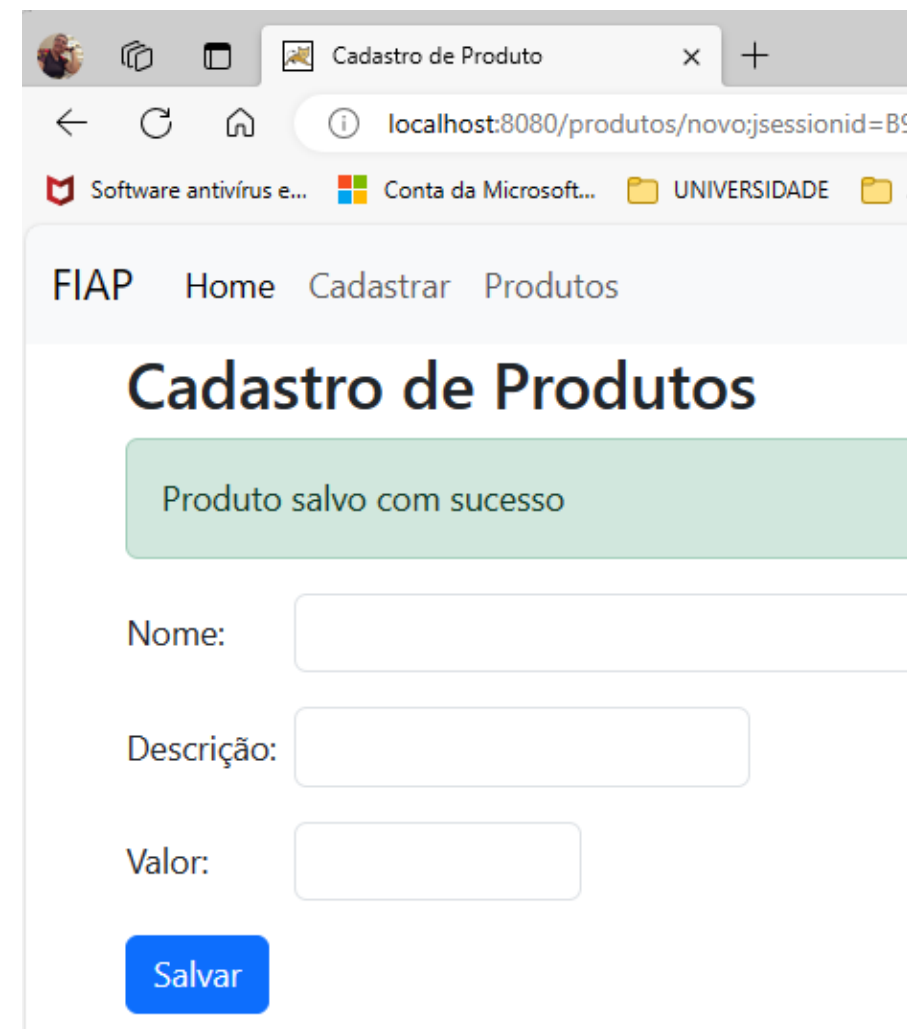
FIAP Home Cadastrar Produtos

Cadastro de Produtos

Nome:

Descrição:

Valor:



Cadastro de Produto

localhost:8080/produtos/novo;jsessionid=B9

FIAP Home Cadastrar Produtos

Cadastro de Produtos

Produto salvo com sucesso

Nome:

Descrição:

Valor:

Refatorar class ProdutoController

- Alterar `@GetMapping("/listar")` para `@GetMapping()`
- Alterar o nome do método `listarProdutos` para `findAll`

```
@GetMapping("/listar") ❌  
@Transactional(readOnly = true)  
public String listarProdutos(Model model){
```



Refatorar class ProdutoController

```
// HTTP - GET - http:localhost:8080/produtos
//@GetMapping("/listar")
@GetMapping()
@Transactional(readonly = true)
public String findAll(Model model){
    model.addAttribute("produtos", repository.findAll());
    return "/produto/listar-produtos";
}
```

I Refatorar navbar.html

- Alterar `Produtos` para `Produtos`

```
<li class="nav-item">  
  <a class="nav-link" href="/produtos/listar">Produtos</a>  
</li>
```



```
<li class="nav-item">  
  <a class="nav-link" href="/produtos">Produtos</a>  
</li>
```


Testar a aplicação

Lista de Produtos

Nome	Descrição	Valor	Ações
Mouse Microsoft	Mouse sem fio	250.0	<button>Excluir</button> <button>Editar</button>
Smartphone Samsung Galaxy A54 5G	Samsung Galaxy A54 5G	1799.0	<button>Excluir</button> <button>Editar</button>
Smart TV	Smart TV LG LED 65 polegadas	3999.0	<button>Excluir</button> <button>Editar</button>

Refatorar class ProdutoController

- Alterar `@GetMapping("/editar/{id}")` para `@GetMapping("/{id}")`
- Alterar o nome do método `editarProduto` para `findById`



```
@GetMapping("/editar/{id}")  
@Transactional(readOnly = true)  
public String editarProduto(@PathVariable ("id") Long id, Mo
```

Refatorar class ProdutoController

```
// HTTP - GET - http:localhost:8080/produtos/1
// @GetMapping("/editar/{id}")
@GetMapping("/{id}")
@Transactional(readonly = true)
public String findById(@PathVariable ("id") Long id, Model model ){

    Produto produto = repository.findById(id).orElseThrow(
        () -> new IllegalArgumentException("Produto inválido - id: " + id)
    );

    model.addAttribute("produto", produto);
    return "/produto/editar-produto";
}
```

Refatorar listar-produtos.html

```
listar-produtos.html x
37 </td>
38 <td>
39     <a class="btn btn-primary btn-xs"
40       th:href="@{/produtos/editar/{id}(id=${produto.id})}"> Editar</a>
41 </td>
```

```
listar-produtos.html x
37 </td>
38 <td>
39     <a class="btn btn-primary btn-xs"
40       th:href="@{/produtos/{id}(id=${produto.id})}"> Editar</a>
41 </td>
```

Testar a aplicação

Lista de Produtos

Nome	Descrição	Valor	Ações	
Mouse Microsoft	Mouse sem fio	250.0	Excluir	Editar
Smartphone Samsung Galaxy A54 5G	Samsung Galaxy A54 5G	1799.0	Excluir	Editar
Smart TV	Smart TV LG LED 65 polegadas	3999.0	Excluir	Editar

Alteração de Produto

Nome:

Descrição:

Valor:

[Alterar](#)

Problemas com o front-end

A tag `<form method="">` do **HTML** só entende os métodos **GET** e **POST**.

Precisamos, de alguma forma, enviar para o back-end a chamada aos métodos **PUT** e **DELETE**.

Para que seja possível fazer isso, precisamos configurar uma **propriedade** no arquivo `application.properties` para habilitar o filtro `spring.mvc.hiddenmethod.filter`, que por padrão vem desabilitado.

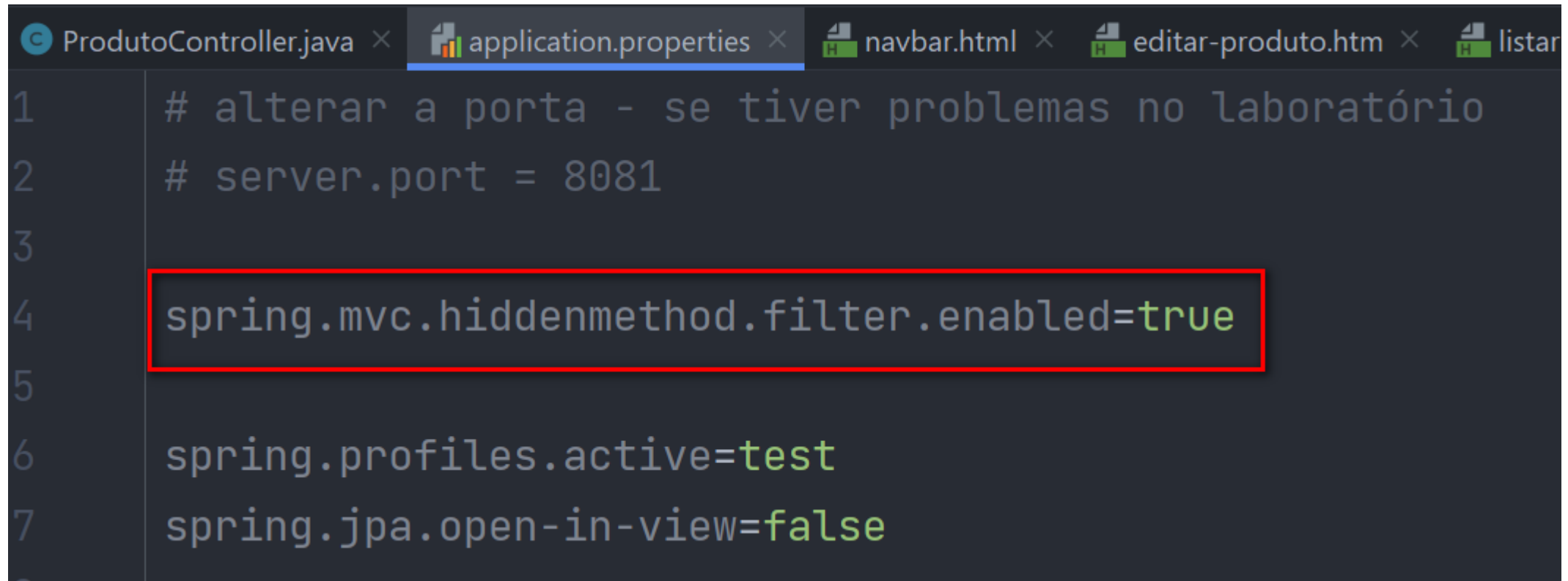
Configurando o filtro `spring.mvc.hiddenmethod.filter=true` o Spring MVC intercepta as requisições e examina o corpo da requisição para localizar o parâmetro oculto, que, geralmente é chamado `_method`, como, por exemplo, `<input type="hidden" name="_method" value="delete">`.

Encontrando esse parâmetro, o filtro altera o método da requisição para corresponder ao valor definido nesse parâmetro.

Para enviar a requisição usando o parâmetro oculto, precisamos utilizar um `<form>` HTML com a tag `<input type="submit">`.

Alterar application.properties

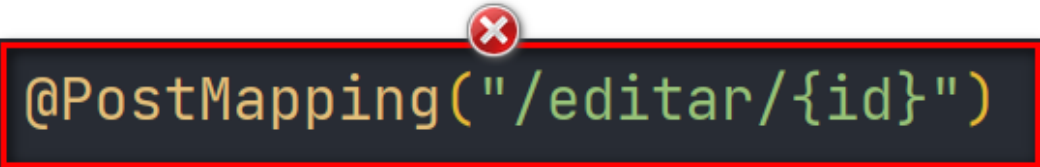
spring.mvc.hiddenmethod.filter.enabled=true



```
ProdutoController.java x application.properties x navbar.html x editar-produto.htm x listar
1 # alterar a porta - se tiver problemas no laboratório
2 # server.port = 8081
3
4 spring.mvc.hiddenmethod.filter.enabled=true
5
6 spring.profiles.active=test
7 spring.jpa.open-in-view=false
```

Refatorar class ProdutoController

- Alterar `@PostMapping("/editar/{id}")` para `@PutMapping("/{id}")`
- Alterar o nome do método `editarProduto` para `update`
- Alterar `return "redirect:/produtos/listar"` para `return "redirect:/produtos";`



```
@PostMapping("/editar/{id}")
```

```
@Transactional
```

```
public String editarProduto(@PathVariable("id") Long id, @Valid  
                            BindingResult result){
```

```
    if(result.hasErrors()){
```

Refatorar class ProdutoController

```
// HTTP - POST - http://localhost:8080/produtos/1
// @PostMapping("/editar/{id}")
@PutMapping("/{id}")
@Transactional
public String update(@PathVariable("id") Long id,
                      @Valid Produto produto,
                      BindingResult result){

    if(result.hasErrors()){
        produto.setId(id);
        return "/produto/editar-produto";
    }

    repository.save(produto);
    return "redirect:/produtos";
}
```

Alterar editar-produto.html

```
produtos.html x ProdutoController.java x application.properties x editar-produto.html x
<!--Editor -->
<form action="#" th:action="@{/produtos/editar/{id}(id=${produto.id})}" th:object="${produto}" method="post">
  <div class="row mb-3">
    <label for="nome" class="col-sm-1 col-form-label">Nome:</label>
```

```
Controller.java x application.properties x navbar.html x editar-produto.htm x listar-produtos.html x novo-produto.html x
<form action="#" th:action="@{/produtos/{id}(id=${produto.id})}" th:object="${produto}" method="post">
  <input type="hidden" name="_method" value="put">
  <div class="row mb-3">
    <label for="nome" class="col-sm-1 col-form-label">Nome:</label>
```

Refatorar class ProdutoController

- Alterar `@GetMapping("/deletar/{id}")` para `@DeleteMapping("/{id}")`
- Alterar o nome do método `deletarProduto` para `delete`
- Alterar `return "redirect:/produtos/listar"` para `return "redirect:/produtos";`

```
@GetMapping("/deletar/{id}")  
@Transactional  
public String deletarProduto(@PathVariable("id") Long id, Model  
    if(!repository.existsById(id)){  
        throw new IllegalArgumentException("Produto inválido -
```

Refatorar class ProdutoController

```
// @GetMapping("/deletar/{id}")
@DeleteMapping("/{id}")
@Transactional
public String delete(@PathVariable("id") Long id, Model model){
    if(!repository.existsById(id)){
        throw new IllegalArgumentException("Produto inválido - id: " + id);
    }
    try {
        repository.deleteById(id);
    } catch (Exception e){
        throw new IllegalArgumentException("Produto inválido - id: " + id);
    }

    return "redirect:/produtos";
}
```

Alterar listar-produtos.html

```
listar-produtos.html x
30 <td><span th:text="${produto.valor}"> Valor </span></td>
31 <td>
32     <a class="btn btn-danger btn-xs"-->
33         th:href="@{/produtos/deletar/{id}(id=${produto.id})}"-->
34         th:data-confirm-delete="|Deseja excluir ${produto.nome}?|"-->
35         onclick="if (!confirm(this.getAttribute('data-confirm-delete'))) return false"-->
36         Excluir</a>-->
37
38     <form th:action="@{/produtos/{id}(id=${produto.id})}" th:object="${produto}" method="post">
39         <input type="hidden" name="_method" value="delete">
40
41         <input class="btn btn-danger btn-xs" type="submit" value="Excluir"
42             th:data-confirm-delete="|Deseja excluir ${produto.nome}?|"
43             onclick="if (!confirm(this.getAttribute('data-confirm-delete'))) return false"/>
44     </form>
45 </td>
```

Alterar listar-produtos.html

```
listar-produtos.html x
46 <td>
47     <form action="" th:action="@{/produtos/{id}(id=${produto.id})}" th:object="${produto}" method="get">
48         <a class="btn btn-primary btn-xs"-->
49             th:href="@{/produtos/{id}(id=${produto.id})}"> Editar</a>-->
50         <input class="btn btn-primary btn-xs" type="submit" value="Editar">
51     </form>
52
```


Testar a aplicação

Lista de Produtos

Nome	Descrição	Valor	Ações
Mouse Microsoft	Mouse sem fio	250.0	<div>Excluir</div> <div>Editar</div>
Smartphone Samsung Galaxy A54 5G	Samsung Galaxy A54 5G	1799.0	<div>Excluir</div> <div>Editar</div>
Smart TV	Smart TV LG LED 65 polegadas	3999.0	<div>Excluir</div> <div>Editar</div>

Lista de Produtos

Nome	Descrição	Valor	Ações
Smartphone Samsung Galaxy A54 5G	Samsung Galaxy A54 5G	1799.0	<div>Excluir</div> <div>Editar</div>
Smart TV	Smart TV LG LED 65 polegadas	3999.0	<div>Excluir</div> <div>Editar</div>

localhost:8080/produtos

localhost:8080 diz
Deseja excluir Mouse Microsoft?

OK Cancelar

Lista de Produtos

Nome	Descrição	Valor	Ações
Mouse Microsoft	Mouse sem fio	250.0	<div>Excluir</div> <div>Editar</div>
Smartphone Samsung Galaxy A54 5G	Samsung Galaxy A54 5G	1799.0	<div>Excluir</div> <div>Editar</div>
Smart TV	Smart TV LG LED 65 polegadas	3999.0	<div>Excluir</div> <div>Editar</div>

Lista de Produtos

Nome	Descrição	Valor	Ações
Lista de produtos vazia			



Copyright © 2024
Prof^a. Aparecida de Fátima Castello Rosa

Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito, do Professor (autor).