



Salvando pedido

Transcrição

[00:00] Nesse vídeo vamos aprender a receber as informações do pedido e salvar no banco de dados. O que nós temos aqui é a página “home” e nós já criamos o menu naquele vídeo. Clicamos em “Novo” e vamos para a página de “Novo Pedido”, que ainda não está estilizada. Nós vamos fazer isso depois.

[00:16] E aí preenchemos as informações. Já estamos mandando essas informações que estamos preenchendo para o back-end”. Já estamos fazendo uma requisição. Se eu apertar a tecla “F12”, eu consigo ver a requisição feita.

[00:33] Aqui no início tem “/pedido/novo”, método “post” e as informações são passadas no “Form Data”. Então esses são os dados que preenchemos na tela, que o usuário preencheu. Ele vai com esses valores, “nomeProduto”, valor do “produto”, “UrlProduto” e tem isso que o usuário preencheu que estava no “input”.

[00:52] Como recebemos essas informações? Então, primeiro precisamos de alguém que responda a requisição “/pedido/novo”. Eu vou criar o método “public String” porque depois esse “String”, esse retorno tem haver com a página que vai aparecer depois que eu salvar o pedido. Vamos ver isso depois também. Então, “novo()”.

[01:10] O nome do método é “novo” e ele vai receber, não um “GetMapping”, mas um “PostMapping”. E o “PostMapping” é para “/pedido/novo”. Vamos retornar alguma coisa? Eu vou debugar só para nós percebermos que estamos realmente recebendo uma requisição.

[01:34] Eu salvei e ele já atualizou, ou seja, agora já tem uma “action” para receber a requisição feita no Browser. Então vamos limpar e voltar. Os dados estão preenchidos. Eu vou clicar em “Cadastrar”. “500”. Deve ter dado um erro. Vamos ver... Não existe “template” para vazio. Então vamos colocar “pedido/formulario”.

[02:02] Ou seja, voltamos para o formulário. É porque o debug não está pegando. Então eu vou subir no modo debug e vou abrir a página de novo. Esse aqui dá esse erro. Pronto! Estou na página e vou submeter a requisição de novo. Vamos soltar. Agora vai!

[02:38] Então bateu no debug. Eu estou batendo porque eu estou dizendo basicamente isso, que a requisição tem que bater nesse método. E aí, depois nós redirecionamos para o próprio formulário. Só que no formulário vai aparecer vazio, que é o que aconteceu aqui. Deixe-me fechar isso. Beleza, apareceu vazio. Pronto!

[02:54] O que nós precisamos fazer agora? Precisamos receber as informações que estão no “request”. Espere! Deixe-me voltar e fazer a requisição de novo com a tecla “F12” para nós vermos os dados de novo. Então dê um “Cadastrar”.

[03:11] A requisição foi feita e está lá debugando, está presa e ele já está acendendo e dizendo que o debug está ali. A requisição foi “/pedido/novo” e está aqui os valores, “nomeProduto”, “urlProduto”, “urlImagem” e “Descricao”. Agora nós temos que receber essas informações. O que vamos fazer?

[03:21] O Spring consegue pegar esses parâmetros que vem na requisição. Ele consegue pegar esses parâmetros e mapear isso para um objeto. Aí vamos precisar de uma classe. Essa classe que vamos criar agora tem que ter atributos com esses nomes aqui.

[03:38] Então, olhe o que eu vou fazer: eu vou dizer que esse aqui vai receber uma “RequisicaoNovoPedido”. Então “Requisicao”. Essa classe não existe. Eu vou criar dentro de um pacote, não no “controller”. Vamos colocar no “dto”. C

“dto data” transforma objetos. É uma classe que serve apenas transferir objetos.

[04:08] Então eu vou usar uma classe para receber informações e transferir dados em objetos. Para receber informações em um objeto dessa classe chamada de “RequisicaoNovoPedido”. “Finish” e ele criou esse pacote novo e está aqui a requisição. Aqui dentro dessa classe eu vou colocar os atributos.

[04:24] Os atributos vão ter os nomes que eu estou mandando na requisição. Ele perdeu porque ficou muito tempo debugando. Não sei se ainda está preso. Então, “novo” está aqui, “nomeProduto”, “urlProduto”, então volta para cá, “urlProduto”, “urlImagem” e uma “Drescricao”. Beleza!

[04:50] Então todas essas informações vem como String. Vou colocar “private String nomeProduto;”, “private String urlProduto;”, “private String urlImagem;” e “private String descricao;” e eu vou criar os “Getters and Setters”.

[05:05] Então apertei as teclas “Ctrl + 3” e “G + G + A + S”, “Generate Getters and Setters”, e “Enter”. E aí vai aparecer aquele campo. Como eu já tenho todos os “Getters and Setters”, ele não aparece. Então vou salvar, terminar e subir de novo. Então vamos subir o “MudiApplication.java”. E agora, a princípio, aqueles dados que vêm no parâmetro da requisição, vão ser colocados dentro de um objeto dessa classe. Então, vamos ver!

[05:36] Vamos voltar e fazer a requisição. Eu estou segurando, vou voltar a segurar aqui e vou fazer a requisição de novo. O que aconteceu? Não aconteceu nada. Agora foi. Pronto! Então esse aqui veio preenchido com as informações da requisição.

[05:57] Então, como fazemos para receber as informações que vem de um formulário? É só criar uma classe e o nome dos atributos da classe tem que ter o mesmo nome dos “names” dos “inputs”. De novo. “RequisicaoNovoPedido” - e olhe só o nome desses atributos aqui. Agora vamos no formulário e olhe o “name” dos “inputs”.

[06:19] “NomeProduto”, “UrlProduto”, “UrlImagem”, está certo? Então esse “name” que damos aos “inputs” é que é mapeado para uma determinada classe. E aí um objeto dessa classe vem com todas as informações dentro.

[06:33] O que temos que fazer aqui? Para que serve essa requisição “novoPedido”? Um “pedido” tem mais informações do que “NomeProduto”, “UrlProduto”, “UrlImagem” e “Descricao”. Também tem o preço, data de entrega, pode ter estado também e a situação do pedido. Mas na requisição são só essas informações.

[06:51] Então nós criamos um objeto para receber só essas informações. Uma outra coisa que nós vamos fazer nessa classe é o seguinte: vamos validar se esses aqui vieram preenchidos. Porque esses três campos são obrigatórios.

[07:02] Mas antes disso, vamos fazer o seguinte: vamos pedir para a requisição criar um “pedido” para nós. Um objeto do tipo “pedido” com os dados que vieram dentro dela. Então vamos fazer “Pedido pedido =” receber “requisicao.toPedido();”. Esse método “toPedido” é meio que padrão você usar o “to”.

[07:22] Basicamente, estamos convertendo uma requisição de “novo Pedido” para um “pedido” novo mesmo. Então ele está dando erro porque eu tenho que importar o “pedido” e está dando erro aqui porque esse método não existe. Aí eu criei esse método aqui embaixo, depois do “Getters and Setters”.

[07:38] Então, o que vamos fazer aqui embaixo? Nós vamos criar um “pedido”. “Pedido pedido = new Pedido();” e nós vamos passar os valores do atributo de um objeto dessa classe para dentro desse “new Pedido”. Então, “pedido.setDescricao(descricao);”.

[07:51] “Pedido.setNomeProduto(nomeProduto);” tem aqui também. “Pedido.set” e “urlImagem”. Tem que ter a “urlProduto” também. E só! São esses quatro valores. Retornamos o “pedido”. Vou derrubar, salvar e já tem um pedido.

[08:11] O que vou fazer com esse pedido? Ou seja, a requisição não me trouxe os valores que eu usei para criar um pedido novo, um objeto. Mas não salvei no banco de dados. Para salvar no banco de dados, o que nós fazemos?

[08:22] Nós criamos, delegamos para uma classe específica do tipo ou chamada de “repository”, “PedidoRepository” para fazermos todo esse trabalho de comunicação com o banco de dados. Então “private PedidoRepository”. Vamos injetar esse aqui e chamar de “pedidoRepository”. Quer dizer, o nome em letras minúsculas.

[08:43] E aí eu vou dar um “@Autowired”. Lembra para quê serve isso? Para poder pedir para o Spring injetar esse aqui. Então ele vai criar uma instância dessa classe e nós vamos utilizar essa classe para salvar esse pedido.

[08:55] Nós utilizamos essa classe no “homeController” para dar um “findAll” e pegar todos os pedidos. No futuro isso vai ser alterado para pegarmos só os pedidos do usuário logado. Mas nós não temos o usuário logado, então estou pegando todos os pedidos.

[09:10] E esse “findAll” nós não implementamos. Ele já veio no “JpaRepository”. Só abrir o “PedidoRepository” e já tem “JpaRepository”. Só que “JpaRepository” tem vários métodos. E um que é o que precisamos usar agora é método “save”. Tem “save” em “Flushes”, mas tem o “save” também.

[09:31] Deve estar aqui em alguma dessas interfaces. Então vamos fazer “PedidoRepository.save”. Vamos passar “pedido”. Na verdade, esse “save” é do “CrudRepository”. Mas com isso nós conseguimos salvar o pedido. Apertei as teclas “Ctrl + S”.

[09:53] Vou subir de novo porque derrubamos. Estava dando algum problema e nós vamos ver que agora ele vai salvar no banco de dados aquele pedido. Então eu vou subir ele no modo debug. “MudiApplication.java”, vou soltar o debug porque ele estava travando. Aí eu cliquei aqui para ele soltar. Não sei por que ele faz isso, mas vamos lá!

[10:21] Agora, o que vamos fazer? Vamos passar informações que podemos pegar do produto. Por exemplo: “nome do produto” vai ser “produto”. Pronto! Só que eu não vou preencher a “URL”, nem “Imagem” e nem “Descricao”; apenas “Cadastrar”. O que vai acontecer?

[10:35] Deixe-me soltar o debug, fazer a requisição bater nesse aqui. Está pendente ainda, ele ainda não chegou aqui. Acho que o debug ficou um pouco lento. Pronto! Está funcionando. Então recebi a requisição e na requisição olhe o que preenchemos. Só o “nome do produto”.

[10:53] Ainda sim, ele vai criar um objeto do tipo “pedido” e esse objeto vai ter só o nome do produto. Nós vamos salvar no banco de dados com o “save”. Vamos ver. Apertei a tecla “F6”. Não deu erro e nem processou até agora. Agora apertei a tecla “F8”.

[11:10] Vamos soltar o debug. Está meio lento. Ele voltou para essa tela, mas se formos para a página “home”, vamos ver que tem um “pedido” lá. Só que o “pedido” está vazio, ele só tem o nome do produto, ele está colado com o de cima. Nós temos que colocar um “margin botton” para eles. Vamos fazer isso, mas eu vou deixar para um momento em que eu tiver que mexer nessa página de novo.

[11:36] Então, pronto! O produto está vazio e esse é um problema que nós temos que resolver agora. Não aceitar uma requisição de “new pedido” sem que o usuário tenha preenchido tudo. A “UrlImagem”, a “UrlProduto” e o “nomeProduto”. Esses três campos são obrigatórios. Mas isso nós vamos fazer no próximo vídeo. Até lá!