



Atualizando o registro

Transcrição

[00:00] Olá. Na aula anterior eu deixei como desafio para que você criasse a função de atualizar um registro de cargo. Agora eu vou mostrar como que eu vou desenvolver esse método. Basicamente o nosso método *save* dentro de Crud tem por objetivo tanto salvar quanto atualizar um registro. Basicamente o que muda é que para atualizar você precisa inserir o Id do registro que você deseja atualizar. Então vamos lá.

[00:33] Vou criar o `private void atualizar`. E para atualizar eu vou precisar também do Scanner. Para atualizar o usuário terá de me passar no Console duas informações agora, não mais só descrição. Ele terá de me informar o Id do registro que ele quer alterar na tabela. Para que eu possa interceptar o Id é o `scanner.nextInt` e eu vou precisar que ele informe também a nova descrição que ele deseja inserir.

[01:33] Então `system.out.println ("Descricao do Cargo")` e vou inserir também essa descrição dentro de uma variável, que é através do `scanner.next` conforme está ali em cima. Então já tenho a descrição e o Id que o usuário deseja atualizar. Agora eu tenho que criar o objeto. Então `cargo = new Cargo` e agora `cargo.setId`. Vou adicionar o Id que o cliente informou e vou colocar aqui o `setDescricao` que o cliente também informou.

[02:55] Feito isso, posso chamar também o `cargoRepository.save`, pois ele tem a função de atualizar também o registro caso informado o Id de registro da tabela. E também vou informar para o cliente uma mensagem para informar que o arquivo foi atualizado. Agora eu já tenho dois métodos aqui e eu não

posso mais colocar aqui o atualizar direto porque agora o cliente tem que saber qual é a ação que ele quer executar dentro da nossa classe de serviço.

[03:35] Se ele quer salvar ou se ele quer atualizar. Nós já fizemos algo muito parecido aqui na nossa classe inicial que o cliente pode escolher se ele quer sair ou se ele quer ir à ação de cargo. Então nós vamos ter que criar algo parecido com isso dentro da nossa classe de Service para que o cliente possa também selecionar qual é a ação que ele quer executar aqui.

[04:01] Então vou criar também a variável aqui Booleana de System que vai controlar quando o cliente vai sair daqui de dentro, quando ele não quiser mais executar nenhuma função de cargo. Vou atribuir essa variável com valor `true` e vou falar aqui que enquanto a variável for `System = true`, o cliente vai continuar aqui dentro executando funções de cargo. Então agora vamos passar um informativo para o cliente, um cabeçalho.

[04:40] Vamos falar para o cliente, `System.out.println ("Qual a ação de cargo deseja executar")`. O 0 vai ser sair. Então `System.out.println ("0 = sair")`, o 1 `System.out.println("1=salvar")` e o 2 `System.out.println("2 = atualizar")`. Agora eu preciso fazer uma validação de qual ação. Como nisso existem outras funções que podemos atribuir aqui dentro, eu não vou fazer *if*, vou fazer por *switch* para que possamos selecionar de uma maneira mais rápida qual é a ação que o cliente está querendo executar.

[05:54] Uma maneira mais organizada para selecionar o que o cliente quer executar. Então vamos aqui `int action = scanner.nextInt`. Agora vamos fazer *switch* e aqui vai ser a *action* e o *default* vai ser colocar a nossa variável System como sendo *false*. Então selecionando qualquer opção que não seja mapeada, ele sai de dentro do nosso contexto. Então se for 1, vamos chamar o "salvar" conforme havíamos visto na aula anterior.

[06:45] E se colocar 2, agora nós vamos para a função de "atualizar". Feito isso, vamos salvar e executar a nossa aplicação para ver se ela está funcionando. Então vamos aguardar o Console iniciar. Console iniciado. Enquanto ele

carrega aqui, vamos no DBeaver e vamos selecionar. Nós temos um registro na nossa tabela que é o ID3 e ele está como RH.

[07:29] Eu vou alterar o valor desse registro 3. Então é importante eu gravar o ID3 e colocar a nova descrição que eu desejo inserir aqui. Projeto iniciado. Vou em 1 e agora aqui eu quero atualizar. Então ele vai me perguntar qual é o Id. O Id conforme vimos no DBeaver é o 3. Vamos aqui, 3 e a descrição que eu quero é `recurso humano` . Atualizado.

[08:08] Agora podemos voltar no DBeaver, vamos dar um "Renovar", RECURSO HUMANO. Então já temos a nossa função de atualizar. Então muito obrigado por terem visto o vídeo. Até mais.