

Criando classe de configuração

Transcrição

[00:00] Nós criamos o usuário "aplicação-casadocodigo", e fornecemos o acesso para esse usuário de poder utilizar os recursos do "S3", para depois enviar as imagens dos livros para o bucket que a gente criou o serviço do "S3". Então quando criamos esse usuário pessoal, a Amazon forneceu para a gente as credenciais com a "Chave de Acesso" e a "Chave de Segredo" que a gente fez o download na etapa anterior. Então eu já tenho aqui, justamente, essas credenciais com a "Chave de Acesso" e a "Chave de Segredo" que está aqui.

[00:30] Agora a gente precisa colocar essas informações no nosso projeto da "Casa do Código" para que a gente consiga realizar a autenticação com a Amazon para depois a gente poder ir fazer o envio das imagens dos livros.

[00:42] Vamos voltar para o nosso projeto no Eclipse, e para poder realizar a integração dessa nossa aplicação, com o ambiente de "Cloud" da Amazon, vamos precisar adicionar mais uma dependência aqui no projeto. Vou vir aqui no "Pom.XML", e colocamos na aba do "Pom.XML", e logo antes de fechar a aba aqui "dependencies", a gente vai adicionar essa dependência para que a gente consiga realizar a integração com a Amazon.

[01:09] Então logo abaixo do vídeo nos exercícios, eu também deixo essa dependência para você adicionarem no projeto. Então vamos colocar Ctrl+Shift+F para identificar, Ctrl+S salvar e ele já começa aqui abaixar dependências necessárias para a gente ter, justamente, essa possibilidade de integrar com a Amazon.

[01:27] Então ele já baixou todas as dependências aqui, vamos no pacote "Configuration" e vamos criar, justamente a nossa classe, para realizar esse processo de autenticação com a Amazon e depois o envio das imagens dos livros para esse bucket.

[01:43] Vamos criar uma "classe" que vamos chamar de "Amazon configuration", e vamos passar justamente, aquelas "Chaves" que nós fizemos o download, as credenciais com a "Chave de Acesso" e a "Chave de Segredo".

[01:58] Vamos colocar essas informações como sendo um atributo estático da "classe"? Então vamos colocar: "private static final String ACCESS_KEY", e colocamos aqui a nossa "Chave de Acesso", que nós já vimos que essa daqui, só copiar e a gente cola ela aqui. E vamos fazer a mesma coisa para nossa "Chave de Segredos", só colocar "private static final String SECRET_KEY", e vamos na nossa "Chave de Segredos" e copiamos ela aqui.

[02:28] Agora temos essas credenciais para realizar autenticação com a Amazon. Só que tenho que especificar, onde que está o bucket que criamos, na Amazon também. Então vamos voltar para o "Painel de Console" somente para confirmar onde criamos o bucket.

[02:43] Voltamos na aba "services" e procura por "S3", e teremos aqui, justamente o bucket quem criamos anteriormente. Criamos esse bucket na região norte da Virgínia, nos Estados Unidos e esse acesso para essa região norte da Virgínia, temos que pesquisar na documentação da Amazon, qual que seria o código referente para poder acessar essa região norte da Virgínia.

[03:09] Então vou pesquisar aqui, na outra aba, e colocar "AWS", "regions" em inglês, as regiões da "AWS", para podermos ter um redirecionamento para o link da documentação. Acho que nesse primeiro link já deve ter a informação com esses códigos dessas regiões, nesses vários "Data Centers" que a Amazon tem espalhado pelo mundo.

[03:29] Essa região do norte da Virgínia nos Estados Unidos, para podermos acessar, temos que utilizar esse código aqui o "us-east-1". Vou copiar esse código e voltar para o nosso projeto, e vai especificar, juntamente que o nosso bucket está

nessa região, que é a "us-east-1" no norte da Virgínia. Também vou colocar aqui como sendo também atributo estático da classe, vou colocar "region" e "us-east-1".

[03:57] Agora, precisamos o que? Precisamos configurar de fato configurar a autenticação com a Amazon. E para poder configurar essa autenticação, vamos precisar de um objeto tipo "BasicAWSCredentials". Então vamos colocar o método que vai se retornar, justamente esse objeto do tipo "BasicAWSCredentials", só chamar ele aqui no "auto complete", e vamos colocar o nome do nosso método também como "basicAWSCredentials()".

[04:30] E agora vamos simplesmente o que? Vamos instanciar a classe "BasicAWSCredentials" e no construtor dessa classe vamos passar, justamente, a "Chave de Acesso" e a "Chave de Segredo". Então vamos passar aqui no primeiro parâmetro "Chave de Acesso" e no segundo parâmetro a "Chave de Segredo", e basta retornar esse nosso objeto aqui do tipo "BasicAWSCredentials". Ainda vamos pedir pessoal, uma vez já que estamos usando o "spring", para ele fazer o gerenciamento desse objeto para nós. Então vamos aqui, e anota esse método com o "@Bean", para o "spring" fazer o gerenciamento desse objeto.

[05:06] Com isso estamos o que? Estamos configurando esse processo de autenticação com a Amazon. Só que temos que a configuração também pessoal, para realizar o envio dos arquivos para Amazon. Então precisamos ter um objeto do tipo "AmazonS3", para que a gente chame os métodos respectivos para poder enviar os arquivos posteriormente.

[05:25] Então vou ter que criar um outro método pessoal, que vai retornar para gente, um objeto do tipo "AmazonS3", vou somente colocar aqui também o nome do método como sendo "AmazonS3", e só colocar o Ctrl+Shift+O, para importar todos os pacotes, e para a gente realizar a construção desse objeto "AmazonS3", vamos colocar aqui "AmazonS3ClientBuilder", e chamamos um método estático "standard", para termos a instância de criação desse objeto, e pessoal temos que especificar qual é região, no qual, vai enviar os arquivos.

[06:02] Então temos a informação que o bucket está na região do norte da Virgínia. Quando criar esse objeto "Amazon S3", eu tenho que especificar que a comunicação vai ser com a região do norte da Virgínia com código "us-east-1". Como é que seria uma forma de dizer em inglês com a região, é "withRegion". Então eu chamo o método "withRegion", e passa a região no qual a gente estará realizando essa comunicação com o bucket que está no norte da Virgínia. Então temos aqui que já configuramos no atributo estático da classe.

[06:32] Mas e aí quando a gente criar esse objeto "AmazonS3"? Ele tem as credenciais necessárias para poder realizar esse tipo de acesso na Amazon? Então temos que, justamente, quando criar esse objeto "AmazonS3", passar as devidas credenciais.

[06:46] Chamamos o método aqui "withCredentials" para passar as credenciais. Esse método pessoal, espera receber um objeto que implementa essa interface "AWSCredentialsProvider". Então vamos aqui instanciar classe "AWStaticCredentialsProvider".

[07:06] E no construtor da classe, basta o que? Chamar o método "BasicAWSCredentials", que já estamos instanciando, justamente, passando a "Chave de Acesso" e a "Chave de Segredo". Vamos vir aqui, e simplesmente chama esse método, e para finalizar a construção, vamos aqui e chamamos o método "build" para construir esse nosso objeto a "AmazonS3".

[07:29] E basta agora o que? Retornar aqui nesse nosso objeto "AmazonS3". E mesma coisa que nós fizemos no método acima, vamos pedir para o "Spring" gerenciar esse objeto aqui, e coloca a notação "@Bean".

[07:41] E para finalizar, para dizer que essa classe deve expor esses "Beans" que a criamos, basta colocar aqui a anotação "@Configuration". Com isso, fizemos as configurações necessárias para realizar a autenticação com a Amazon, e criamos o objeto "AmazonS3", para depois chamar os respectivos métodos e enviar os arquivos para o bucket que criamos na Amazon.

[08:07] Então a próxima etapa vai ser, justamente, fazer as alterações em nosso projeto para enviar o arquivo para o bucket da Amazon. Vamos fazer isso lá na próxima etapa vamos lá.