



Criando um projeto com Maven

Transcrição

[00:00] Boas-vindas de volta ao treinamento de Maven! Continuando. Agora que já conhecemos o Maven, já sabemos como é que ele funciona, vamos utiliza-lo diretamente da IDE para facilitarmos a integração que já vem, aqui no caso, do Eclipse.

[00:13] Eu estou com o meu Eclipse aberto, em um *workspace* zerado, sem nenhum projeto. Aqui no menu esquerdo, perceba que ele te dá algumas opções. Fala: você não tem nenhum projeto, quer importar um projeto, quer criar um projeto. A primeira opção que ele sugere: quer criar um projeto com Maven? “Create a Maven project”.

[00:30] Esse vai ser o objetivo da aula de hoje. Vamos entender como que fazemos para criar um projeto com o Maven. Você poderia clicar em “Create a Maven project” no lado esquerdo da tela ou acessar o menu “File > New > Maven Project” - vai dar na mesma, ele vai redirecionar para essa mesma tela.

[00:44] Aqui ele te pergunta se *workspace* usa esse *workspace* padrão. Tem essa primeira opção aqui: “Create a simple project (skip archetype selection)” - em português, “criar um projeto simples (pular a escolha de um arquetipo)”. Se você não marcar ela e clicar em “Next”, ele vai te direcionar para essa outra tela aqui, pedindo para você escolher um *archetype*.

[01:03] Esse *archetype* nada mais é do que um padrão de como que ele vai criar a estrutura do projeto dependendo do objetivo do seu projeto. Ele já tem aqui no catálogo, “Catalog”, alguns *archetypes* por padrão, que já vêm embutidos.

[01:18] Ele fala: você quer criar uma aplicação “J2ee”? Hoje em dia ninguém mais cria uma aplicação “J2ee”, mas se você quisesse criar uma aplicação “J2ee” você poderia clicar nessa opção, que ele já iria criar o projeto com toda a estrutura necessária para funcionar com o “J2ee”.

[01:36] Você quer criar um *plugin* no Maven? Se você quiser criar um *plugin* no Maven, você tem essa opção, que ele já cria a estrutura para um *plugin*.

[01:43] Perceba: é só uma comodidade, ele já cria o projeto com alguns arquivos e alguns diretórios baseado no tipo de aplicação que você quer construir. Aqui, por exemplo: quer criar uma aplicação Web, uma aplicação Java Web tradicional? Ele já cria o diretório “web.info”, cria uma “web.xml” ou cria um “jsp” de exemplo. Tem essas opções disponíveis.

[02:05] Se você não quiser nenhuma delas, se quiser criar uma aplicação Maven pura, sem *archetype* nenhum e sem nada configurado; você pode voltar clicando na opção “Back > marcar o checkbox da opção "Create a simple project (skiparchetype selection)” - que cria um projeto Java com Maven sem *archetype* nenhum, só com a estrutura padrão.

[02:20] E vai desenvolvendo, à medida que vai precisando das coisas você vai criando. Quando clica em “Next” ele te pede algumas informações na janela "New Maven Project". A primeira informação que ele pede é o “*Group Id*”. *Group Id* nada mais é do que a organização, o identificador único da sua organização, da sua empresa.

[02:40] Aqui no “*Group Id*”, lembra que no Java configuramos os pacotes assim: “br.com.alura”? Geralmente esse vai ser o Group Id: “br.com.alura”. Estou identificando unicamente a empresa Alura.

[02:54] “Artifact Id” -, qual é o Id do seu projeto, do artefato. Quando você for gerar o *build* - qual é o nome desse projeto - você estará nesse grupo, “br.com.alura”, no grupo Alura, mas Alura pode ter diversos projetos e diversas aplicações. Como ele vai diferenciar uma aplicação Maven da outra

pertencente a esse mesmo grupo? Eu preciso ter um Id que é único da aplicação.

[03:15] Vou colocar em “Artifact Id”, o nome loja. Imagine que nós vamos desenvolver uma aplicação para uma loja. Aqui seria o “Artifact Id”.

[03:22] Aqui é a versão, “Version”. Qual é a versão? O Maven tem um padrão de nomenclatura para versões, você pode trocar e colocar “1.0”, usar a nomenclatura padrão. Fica a seu critério, vou colocar 1.0.0.

[03:36] Qual é o tipo de empacotamento, `Packaging`. Essa sua aplicação, quando você for gerar o *build* dela vai ser um JAR que você vai gerar, é uma aplicação Java tradicional, é uma aplicação que usa o *spring boot*, por exemplo, que gera um JAR; ou é uma aplicação web tradicional, que você vai gerar um WAR. Você já pode configurar isso daqui. Vou deixar marcado a opção JAR. “`Packaging > jar`”.

[03:56] Tem alguns campos aqui que são opcionais, “Name” e “Description” - são só para você descrever mais o seu projeto, uma descrição do seu projeto, o nome ali do seu projeto. São informações opcionais.

[04:07] “Parent Project” - se o seu projeto Maven vai herdar de uma outra aplicação Maven. Depois vamos ver com calma essa situação. No caso aqui não, não quero herdar de outro projeto Maven. Preenchi as informações principais e vou clicar aqui no “Finish”.

[04:23] Ele vai fazer a criação do projeto, vai baixar algumas coisas da internet na primeira vez que você rodar. Ele criou aqui a aplicação Maven, criou o projeto “loja”.

[04:33] O projeto Maven já vem sem o *archetype*, o projeto padrão puro, digamos assim; ele vem com essa estrutura.

[04:41] Tem alguns *SourceFolder* - “src/main/java”, onde geralmente vão ficar suas classes Java. “src/main/resources”, arquivos de configuração dos seus *frameworks* e os *SourceFolder* para teste, “src/test/java” para classes de testes e “src/test/resources” para arquivos de configuração de teste.

[05:02] Aqui embaixo vem o arquivo extremamente importante no Maven, que é o “pom.xml” - que é um arquivo de configurações do Maven para esse projeto.

[05:12] No Maven, toda a configuração é feita nesse arquivo XML - que vamos explorar ao longo do treinamento. Por padrão, o que vem preenchido aqui no “pom”? Vem o *Group Id*, `<groupId>br.com.alura</groupId>`, conforme digitamos naquela tela de criação do projeto. *Artifact Id*, `<artifactId>loja</artifactId>`, e a versão, `<version>1.0.0</version>`.

[05:29] Esse *Model Version*, `<modelVersion>4.0.0</modelVersion>`, é o modelo, a versão do “pom.xml” do Maven, digamos assim.

[05:34] No momento de gravação do vídeo, o *Model Version* padrão atual é o 4.0. Recomendo que você utilize essa versão. Ele utiliza `http://maven.apache.org/POM/4.0.0` como *name space* do XML de configuração do “pom” na versão 4.0.

[05:50] Vou apertar as teclas “Ctrl + Shift + F” para formatar. Está aí o nosso “pom.xml” em branco, está aí o nosso projeto, essa é estrutura de diretórios. Está vazio, não tem nada aqui dentro, tem só as bibliotecas do próprio Java, do JRE.

[06:07] Está aí o nosso projeto novo, criado do zero com o Meven. Essa é a estrutura de diretórios padrão do Maven e esse é o arquivo XML onde fazemos as configurações do Maven.

[06:18] Não tem muito mistério, é bem simples, bem tranquilo você criar um aplicação do zero utilizando o Maven. Certo? A partir daí, conforme você for

desenvolvendo o projeto e for precisando – “como eu adiciono as dependências, como eu mudo alguma coisa no *build*?” - Conforme for surgindo a necessidade, você vai modificando o “pom.xml” para adicionar essas novas coisas. Tudo bem?

[06:41] Ao longo do curso nós vamos explorando outras funcionalidades e recursos do Maven. Vejo vocês na próxima aula. Um abraço e até lá!