

Herdando comportamento através de mixins e polimorfismo

Transcrição

[00:00] Mas repara como tem muito comportamento que é comum para todas essas classes, livro, revista e ebook. Vamos tirar isso, e jogar em alguém, um produto por exemplo: vou criar o módulo “produto”, “module Produto”.

[00:21] Vou pegar esse comportamento do “csv”, e vou jogar no meu produto. Vou pegar esse método “calcula_preço”, e também vou jogar no meu produto. Agora, volto para o meu livro, e claro: tenho que falar que ele se comporta como um produto.

[00:44] Incluir o modo de produto, falar que ele tem os comportamentos que um produto tem. Ele se comporta como um produto. Para isso, eu preciso dar um “require”, claro. “require_relative “produto” “.

[00:55] Vamos para o terminal, roda a aplicação, e tudo certo. Mas repara o nosso construtor de livro: ele chama o método “calcula_preço”. No nosso módulo “produto”, esse método é privado. Como pode ser que eu acesso ele?

[01:10] Acontece que o método privado é privado ao objeto, e não ao módulo. Eu posso chamar um método privado de um objeto, dentro do próprio objeto, mas não de fora.

[01:25] Por exemplo, vamos tentar um: “puts algoritmos.calcula_preço”. Eu não posso. Não posso acessar o método privado de fora do objeto. Mas dentro do objeto, não tem problema nenhum. Eu posso acessar esse método privado.

[01:38] Quem mais se comporta como um produto no nosso sistema? A resposta é fácil: a revista. Então vamos na revista, tiramos o método “to_csv”, tiramos o método privado de calcular o preço, e incluímos o módulo ‘produto’. E o ebook.

[02:01] Incluímos o produto, tiramos o método “to_csv”, e o método de calcular o preço. Agora aquele “require” tem que estar nos três arquivos: na revista, no ebook também. Vamos remover essa chamada de método e vamos dar uma olhada agora na classe livro.

[02:21] Reparem que o “attr_reader : :título, :preço, :ano_lançamento, :editora”, ainda é bem repetitivo. Vou pegar esses três primeiros campos e vou tirar (título, preço e ano_lançamento), vou jogar no meu produto, eu vou receber eles por herança, tanto título, quanto preço e ano de lançamento.

[02:34] Eu vou na revista e removo os três. Vou no ebook e removo os três. Reparem que na verdade, editora também está nessas três classes, então eu vou pegar editora também.

[02:44] Vou tirar editora daqui do ebook, o reader, e vou colocar no nosso produto. Com isso, eu posso tirar também, esse reader completamente do ebook, da revista, e do livro.

[02:57] Salvamos tudo e roda o programa. Está tudo funcionando. Sempre que eu me comporto como alguma coisa, e esse como está distribuído, está sendo utilizado em diversas classes, eu posso pegar esse comportamento e extrair para um módulo.

[03:16] É importante tomar alguns cuidados, uma vez que esse acoplamento que eu vou ter com esse módulo, vai refletir em todas as classes que usam ele.

