02

Herdando comportamento através de mixins e polimorfismo

Note como tem muito comportamento que ainda é comum a todas as classes: to_csv, calcula_preco, etc. Vamos extrair esse comportamento e jogar em algo que pode ser incluído em todas elas: um módulo. Nada mais justo que chamá-lo de Produto:

```
module Produto

  def to_csv
    "#{@titulo},#{@ano_lancamento},#{@preco}"
  end

  private

  def calcula_preco(base)
    if @ano_lancamento < 2006
      if @possui_reimpressao
        base * 0.9
      else
        base *0.95
      end
    elsif @ano_lancamento <= 2010
      base * 0.96
    else
      base
    end
  end

end
```

Agora voltando em cada uma das nossas classes devemos dizer que o Livro se comporta como um produto, incluindo o módulo em nossas classe:

```
class Livro
  include Produto
  # resto do codigo
end
```

E incluindo o require no arquivo:

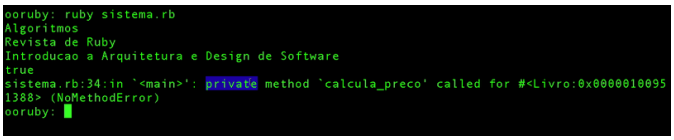
```
require_relative 'produto'
```

Mas repare que o nosso construtor de livro chama o método calcula_preco, que é definido como privado em Produto. Como pode ser que chamaremos um método privado? Acontece que um método privado em Ruby é privado ao objeto, e não ao módulo ou classe onde ele foi definido. Isto é, como o método foi definido em Produto, mas inserido na classe Livro, ao instanciarmos um Livro, esse objeto tem acesso a todos os métodos privados incluidos em seus módulos, independentemente de onde foram definidos.

Para visualizar que o método é privado de verdade, tentar chamar o método calcula_preco de fora do objeto não funciona:

```
puts algoritmos.calcula_preco(100)
```

Resultando no erro de acesso:



Agora vamos em Ebook e Revista remover os métodos to_csv e calcula_preco, adicionando os includes adequados:

```
class Revista
  include Produto
  # resto do codigo
end
class EBook
  include Produto
  # resto do codigo
end
```

Note que diversos attr_readers também podem ser movidos e é o que faremos, movendo-os para o módulo recém criado.

```
module Produto

  attr_reader :titulo, :preco, :ano_lancamento, :editora


  # to_csv e calcula_preco
end
```

Note como o programa continua funcionando normalmente até agora, apesar de todas essas refatorações.