

Ruby on Rails 5.x

Do início ao fim!

Módulos e Mixins



Módulos

Módulos

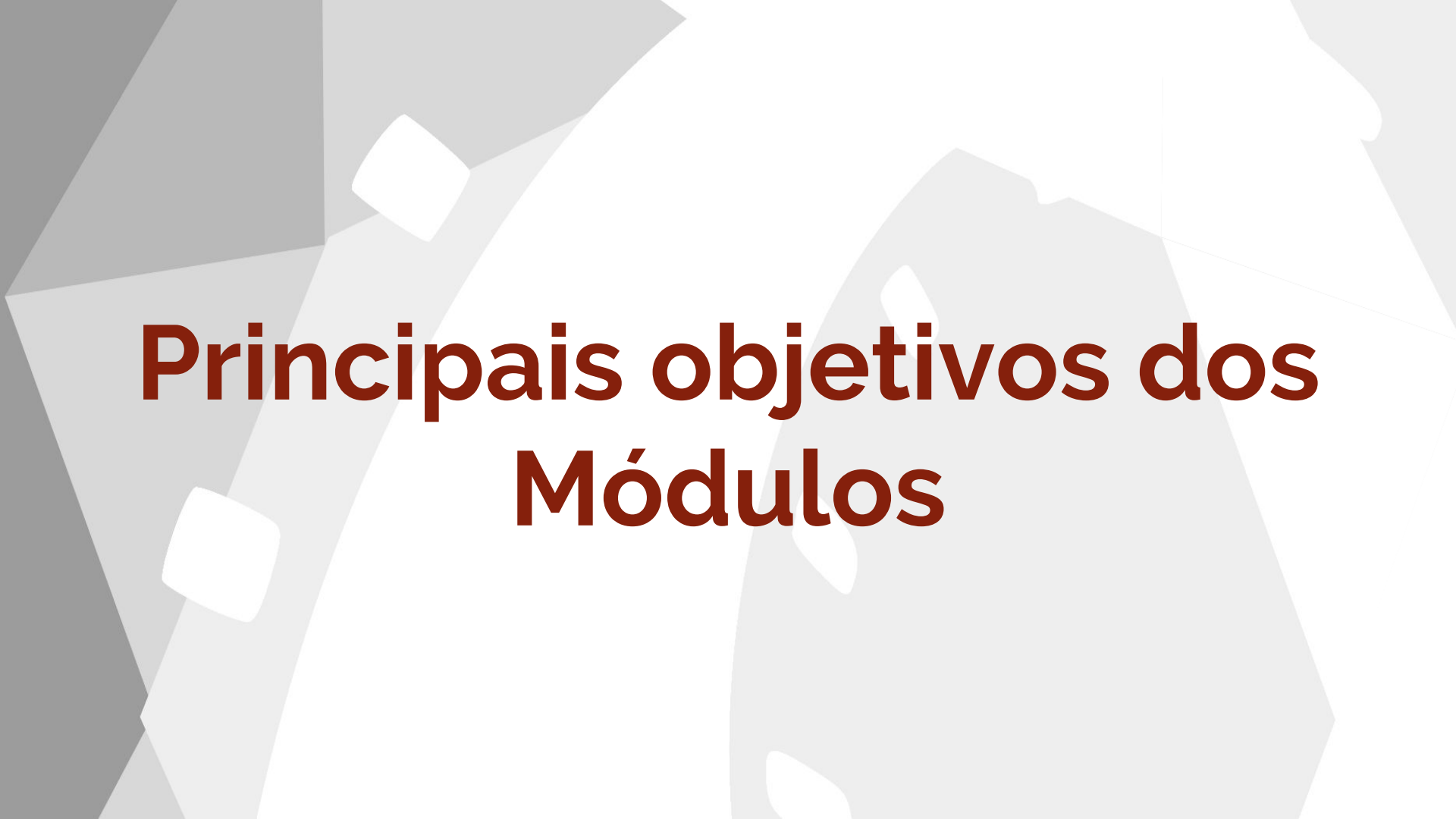
- Módulos Ruby são similares a classes em relação ao fato de que também armazenam uma coleção de **métodos**, **constantes** e outras definições de **módulos e classes**.

Módulos

- Entretanto, diferente das classes, você não pode criar objetos baseados em módulos nem pode criar módulos que herdaram desse módulo; ao invés disso, você especifica qual funcionalidade de um módulo específico você deseja adicionar a uma classe ou a um objeto específico.

Módulos

- Módulos permanecem sozinhos; não há hierarquia de módulos ou herança. Módulos são um bom lugar para armazenar constantes em um local centralizado.



Principais objetivos dos Módulos

Módulos

- Primeiro eles agem como *namespace*, permitindo que você defina métodos cujos nomes não irão colidir com aqueles definidos em outras partes de um programa.

Módulos e Mixins

- Em segundo lugar, permitem que você compartilhe funcionalidade entre classes – se uma classe “mistura” (**mixes in**) um módulo (isto é, o inclui), todos os métodos de instância do módulo se tornam disponíveis como se tivessem sido definidos na classe.

Módulos e Mixins

- Exemplo para Constantes

```
# pagamento.rb

module Pagamento
  PI = 3.14
end
```

```
# app.rb

require_relative 'pagamento'

include Pagamento

puts Pagamento::PI
puts PI
```

Módulos e Mixins

- Exemplo para Métodos

```
# pagamento.rb

module Pagamento
  def pagar(bandeira, numero, valor)
    "Pagando com o cartão #{bandeira} Número #{numero}, o valor de
    R$#{valor}..."
  end
end
```

Módulos e Mixins

- Exemplo para Métodos

```
# app.rb
require_relative 'pagamento'
include Pagamento

puts "Digite a bandeira do cartão:"
b = gets.chomp
puts "Digite a número do cartão:"
n = gets.chomp
puts "Digite a valor da compra:"
v = gets.chomp

puts pagar(b, n, v)
puts Pagamento::pagar(b, n, v)
```

Módulos e Mixins

- Exemplo para Classes

```
# pagamento.rb

module Pagamento
  class Visa
    def pagando
      "pagando..."
    end
  end
end
```

```
# app.rb

require_relative 'pagamento'

include Pagamento

p = Pagamento::Visa.new
puts p.pagando
```

Módulos e Mixins

- Exemplo para Módulos dentro de Módulos

```
# pagamento.rb

module Pagamento
  module Master
    def pagando
      "pagando..."
    end
  end
end
```

```
# app.rb

require_relative 'pagamento'

include Pagamento::Master

puts Pagamento::Master::pagando
```