LINGUAGEM RUBY

Nomes: Fábio França

Rebeca Santos

HISTÓRIA DA LINGUAGEM

- **Desenvolvida no Japão por Yukihiro Matsumo em 1995, o Ruby é a união das linguagens favoritas de seu criador. São elas:**
- **Perl, Smalltalk, Eiffel, Ada e Lisp.**
- **Mais poderosa do que Perl, e mais orientada a objetos do que Python.**
- **Escrita em C**
- **Equilibra a programação funcional com a programação imperativa.**

CARACTERÍSTICAS

- Linguagem interpretada;
- Orientada à objetos;
- Portável;
- Trabalha com herança, classes, métodos, polimorfismo e escalonamento;
- Sintaxe relativamente simples e de fácil compreensão;
- Tipagem dinâmica mas forte;
- A sintaxe é enxuta, quase não havendo necessidade de colchetes e outros caracteres;
- Linguagem de propósito geral

EXPRESSIVIDADE E MÉTODOS

- Os parênteses são opcionais nas chamadas de método

radio.play favorite_song

person.add_parents father, mother

- Métodos que retornam true ou false terminam com uma interrogação em seu

nome

radio.has? favorite_song

- Métodos que alteram o objeto que foi chamado terminam com uma

exclamação (Método Perigoso)

foo = "DOWNCASE"

foo.downcase # => returns a new string

foo.downcase! # => modifies the foo object

Programação Funcional com Ruby

No exemplo ao lado, o valor dentro de **x** foi transformado a partir do **map** e atribuido a **y**, que agora tem o valor [2,4,6,8], porém **x** continua sendo [1,2,3,4]. Em uma linguagem de programação funcional o valor de **x nunca** mudará, apenas se transformará.

Output

```
"x= 1,2,3,4"
"y= 2,4,6,8"
```

Editor

```
x = [1,2,3,4]

y = x.map {|x| x * 2}

p "x= #{x}"
p "y= #{y}"
```

METAPROGRAMAÇÃO

Para ler/alterar os valores das variáveis de instâncias no java existe getters e setters que são métodos para ler/alterar valores das variáveis. Em ruby é possíviel fazer da mesma maneira.

```
1.9.3p125 :001 > class Produto
1.9.3p125 :002?>     def initialize(nome, preco)
1.9.3p125 :003?>         @nome = nome
1.9.3p125 :004?>         @preco = preco
1.9.3p125 :005?>       end
1.9.3p125 :006?>
1.9.3p125 :007 >       def get_nome
1.9.3p125 :008?>         @nome
1.9.3p125 :009?>       end
1.9.3p125 :010?>
1.9.3p125 :011 >       def set_nome(nome)
1.9.3p125 :012?>         @nome = nome
1.9.3p125 :013?>       end
1.9.3p125 :014?>   end
 => nil
1.9.3p125 :015 >
1.9.3p125 :016 >   refri = Produto.new("refri", 2.50)
 => #<Produto:0x8bfd670 @nome="refri", @preco=2.5>
1.9.3p125 :017 > refri.get_nome
 => "refri"
1.9.3p125 :018 > refri.set_nome "refrigerante"
 => "refrigerante"
1.9.3p125 :019 > refri.get_nome
 => "refrigerante"
```

BIBLIOGRAFIA

https://www.ruby-lang.org/pt/about/https://www.devmedia.com.br/conhecendo-a-linguagem-ruby/8226

https://brizeno.wordpress.com/2012/03/27/metaprogramacao-em-ruby/

https://www.inf.ufes.br/~vitorsouza/wp-content/uploads/teaching-lp-20142-seminario-ruby.pdf

https://www.ime.usp.br/~esposte/documents/aula-ruby/aula01/Aula01.pdf

https://www.devmedia.com.br/conhecendo-a-linguagem-ruby/8226