

Programación para todos

5. Métodos, Bloques & Sorting



le wagon

@lewagonargentina

Métodos

Métodos (o funciones) son bloques de código reutilizables escritos para realizar una (repetitiva) tarea específica.

Como una función matemática, los métodos pueden tomar uno o múltiples argumentos (el **input**).

El rol del método es computar algunos cálculos (usando los inputs) y devolver un resultado (el **output**)

Porqué Métodos?

1

Son reusables y dinámicos (el output depende del input!)

2

Ayudan a mantener tu código organizado separando las diferentes partes de tu app:
un método específico ejecuta una específica tarea

3

Para “top-down” programación: como el código se va transformando cada vez más complejo, métodos pueden ayudar a resolver este gran problema (the “top”) dividiendolo en pequeños problemas

Sintaxis

Method name

Parameter(s)

```
def print_reversed_name (name)
  puts name.reverse
end
```

Method body

Cuando escribo un método (con las def-end keywords), no hay código ejecutado, porque solo estoy declarando (asignando).

Para ejecutarlo, tenemos que llamarlo (y especificar algún argumento, sino nos saldrá error!)

Argument(s)

```
print_reversed_name( "gabriele" )
```

Splat t

Qué pasa si nos sabes el número exacto de argumentos a esperar?

Splat arguments!

```
def print_reversed_names(*names)
  names.each do |name|
    name.reverse
  end
end
```

With a star
before!

Ahora puedo pasar uno o más argumentos, e irán todos
dentro de un array llamado `names`

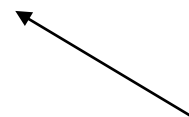
Return

Devolver (Return) un valor es diferente a imprimirlo (print)

Cuando devuelvo (return), el valor que obtengo está disponible en mi código, y puede ser asignado a una variable o usarlo

Cuando imprimo (print), solo muestro el valor en la consola (terminal), pero no está disponible para usarlo en mi código

```
def reverse_name(name)
  return name.reverse
end
```



Is not printing, but just giving me back the value in my code!

Bloques

Bloques (o funciones anonimas) son bloques de código ejecutados dentro de un metodo pre-definido
are block of codes executed inside a pre-defined method (como each, sort, etc.)

Son como métodos, pero no necesitan nombre o ser asignados porque ya estan dentro de otro (pre-definido) método.

```
names = [ "gabriele", "mariana", "shannon" ]
```

```
names.each do |name|
```

```
  reversed_name = name.reverse
```

```
  puts reversed_name
```

```
end
```



Nameless block go code

Gracias!!



le wagon

@lewagonargentina