

# Functions

Uma função é uma estrutura amplamente utilizada para otimizar o desenvolvimento, podemos compara-la como uma receita de bolo. Temos os ingredientes, misturamos eles em certas quantidades em uma determinada ordem e por fim temos o resultado que é o bolo.

No caso de uma função, recebemos argumentos que são processados de acordo com um objetivo e nos gera um resultado.

Uma função nos ajuda a reusar o código, quebrar grandes problemas em problemas menores, reduzir a quantidade de código escrita, melhorar a legibilidade do código e etc.

Um exemplo é a função `mean()` que calcula a média, sem ela o código ficaria assim:

```
nums <- 1:10
media <- sum(nums) / length(nums)
media
```

```
## [1] 5.5
```

Com a função:

```
mean(nums)
```

```
## [1] 5.5
```

No exemplo acima utilizamos uma função já existente no sistema, no entanto é possível criarmos nossas próprias funções.

Imagine que queremos ver qual vetor tem mais posições, é maior:

```
vetor1 <- c(1,2,3,4,5,6,7,8,9,10)
vetor2 <- c(10,20,30,40,50)

checa_tamanho <- function(primeiro_vetor, segundo_vetor){
  tamanho1 <- length(primeiro_vetor)
  tamanho2 <- length(segundo_vetor)

  if(tamanho1 > tamanho2){
    return("Vetor 1 maior")
  }else if(tamanho1 < tamanho2){
    return("Vetor 2 maior")
  }else{
    return("Ambos têm mesmo tamanho")
  }
}

checa_tamanho(vetor1, vetor2)
```

```
## [1] "Vetor 1 maior"
```

```
checa_tamanho(segundo_vetor = vetor2,
               primeiro_vetor = vetor1)
```

```
## [1] "Vetor 1 maior"
```

Acima podemos notar 2 pontos importantes:

- dentro da função criamos variáveis para auxiliar no cálculo, mas ao fim da execução da função não é mais possível acessá-las.

- fizemos a chamada dessa função de duas maneiras, uma respeitando a ordem os parâmetros, e a segunda forma e colocando na ordem que quisermos, com o detalhe que nesse caso é necessário nomear qual é o parâmetro que estamos preenchendo.

## Argumento Opcional

Quando por algum motivo é possível esperar que um dos argumentos não tenha valor fornecido durante a execução podemos deixar um valor padrão:

```
soma <- function(num1, num2 = 10){  
  return(num1+num2)  
}  
  
soma(1)
```

```
## [1] 11
```

```
soma(1,1)
```

```
## [1] 2
```

## Número variado de argumentos

Já por outro lado é possível que não conseguimos pré-determinar quantos argumentos vamos receber, para esse caso usamos os 3 pontos (...):

```
soma <- function(num1, num2, ...){  
  resultado = num1+num2  
  
  for(num in list(...)){  
    resultado <- resultado + num  
  }  
}
```

```
}  
  return(resultado)  
}
```

```
soma(1,2)
```

```
## [1] 3
```

```
soma(1,10,100,1000)
```

```
## [1] 1111
```