

## ARTIGO: 11400

### Dica em Groovy - Usos do método inject da Collection no Groovy

Aê galera, tudo bem?

Hoje quero falar sobre o método [inject](#) da classe `java.util.Collection` em Groovy. A classe `Collection` possui todos os métodos que possui em Java, obviamente, e também adiciona muitos outros, como o [inject](#).

Esse método funciona assim: itere pela coleção e execute a [Closure](#) enviando 2 parâmetros: o primeiro é o resultado da execução anterior e o segundo é o próximo elemento da coleção. Complicou? É mais simples que parece, vamos ver um exemplo, somatório.

Groovy Code

```
def collection = [1,2,3,4, 5]
def total = collection.inject{ somatorio, elemento ->
    somatorio += elemento // na primeira iteração, 1 + 2, na segunda 3 + 3, na quarta 6 + 4, ...
}

println total // mostra 15, ou seja 1 + 2 + 3 + 4 + 5
```

No primeiro elemento, o primeiro e segundo elemento são passados como parâmetros para execução da Closure.

Da segunda em diante, o resultado da operação é passado para o terceiro elemento, e depois o resultado da operação é passado para o quarto e assim por diante até que todos os elementos sejam iterados.

Com esse método é possível fazer várias coisas. Há alguns exemplos na documentação oficial na documentação do Groovy, como achar o maior elemento de uma coleção, por exemplo:

Groovy Code

```
def collection = [81711, 1,5,8,600, 330, 79]
def resultado = collection.inject{ max, elemento ->
    [max, elemento].max()
}

println resultado // mostra 81711, o maior elemento da coleção
```

O método `inject` é uma ferramenta indispensável para o programador Groovy, seus usos são vários.

Espero que tenha sido útil pra alguém.

Até a próxima.