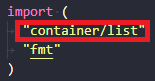
**Listas Ligadas em Go**

Para gerarmos uma lista lagada em Go primeiramente precisamos importar uma biblioteca chamada “container/list”:



Após isso, fazemos o encapsulamento do de uma nova lista, a partir do método list.New() em uma variável desejada qualquer:



Para inserirmos elementos nesta lista iremos utilizar os seguintes métodos:

1. .Pushback() – Método que recebe como parâmetro o valor que queremos inserir na lista na sua **ÚLTIMA POSIÇÃO (FIM):**

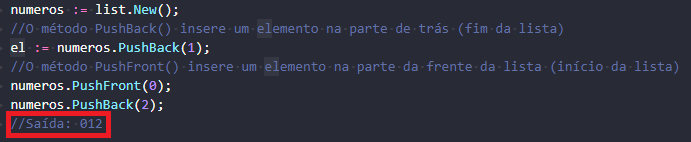


Neste exemplo em específico, encapsulamos o valor desejado numa variável para uso a posteriori.

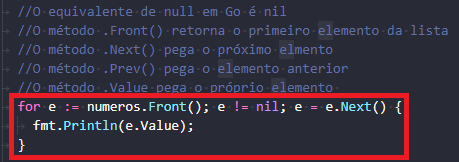
1. .PushFront() – Método que recebe como parâmetro o valor que querermos inserir na lista na sua **PRIMEIRA POSIÇÃO (INÍCIO):**



Por fim, acabamos criando uma lista de valores ordenados, em que cada valor conhece o seu antecessor e sucessor:



Para retornarmos os valores dessa lista, utilizaremos um loop for tradicional, utilizando os métodos a seguir:



**Métodos para ACESSAR valores em listas**

1. .Front() – Método que retorna o **PRIMEIRO ELEMENTO** da lista;
2. .Next() – Método que retorna o elemento **POSTERIOR** ao elemento referenciado. Ex.:  Neste caso em específico, a variável “e” vai encapsular o valor posterior ao elemento “e”.
3. .Prev() – Método que retorna o elemento **ANTERIOR** ao elemento referenciado.
4. .Value – Método que retorna o **PRÓPRIO** elemento referenciado.

Com isso, temos o seguinte retorno:

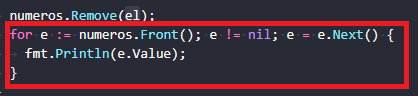


**Método que REMOVE valores em listas | .Remove() |**

O método .Remove() remove um elemento de uma lista. Segue Syntax:



No caso do nosso exemplo, utilizamos um loop for para retornar novamente a lista com seus valores atualizados:



E obtivemos o seguinte retorno:

