

```
//ARGUMENTS:
//A palavra reservada arguments - sempre usada dentro de funções - é um array interno da linguagem chamado "arguments" que existe por padrão dentro de todas as funções, esse array é responsável por guardar todos os parâmetros passados para a função, isso faz com que , mesmo que não tenhamos declarado nenhum parâmetro formal na assinatura da função ainda assim possamos recuperar parâmetros passados numa chamada da função usando o arguments.
//Veja como no exemplo abaixo:

//RECUPERANDO ARGUMENTOS DE UMA CHAMADA MESMO QUE A FUNÇÃO NÃO RECEBE ARGUMENTOS NA SUA ASSINATURA:
function semParam(){ //Perceba que não recebemos parâmetro nenhum...
    let soma = 0; //essa variável irá somar ou concatenar qualquer valor passado como parâmetro...
    for (let i in arguments) { //o arguments vai guardar qualquer valor passado...
        soma += arguments[i]; //Aqui recuperamos os valores passados e os somamos a variável soma, ou concatenamos se forem strings..
    }
    return soma;
}
console.log(semParam());
console.log(semParam(1));
console.log(semParam(1.1, 2.2, 3.3));
console.log(semParam(1.1, 2.2, "Teste"));
console.log(semParam('a', 'b', 'c'));
```

RESULTADO NO CONSOLE...

```
[Running] node "c:\Users\User\Documents\JAVASCRIPT\arquivos_das_aulas\062-Arguments__o_array_interno_das_funcoes.js"
0
1
6.6
3.3000000000000003Teste
0abc

[Done] exited with code=0 in 0.228 seconds
```