Tratamento de dados

1. Conversão
   1. Para número:

|  |  |  |
| --- | --- | --- |
| Number.parseInt(n) | Number.parseFloat(n) | Number(n) |

* 1. Para string:

|  |  |
| --- | --- |
| String(n) | n.toString() |

1. Formatação de String

|  |  |  |
| --- | --- | --- |
| s.length | s.toUpperCase() | s.toLowerCase() |

1. Formatação Number

* n.toFixed(x).replace(‘.’, ‘,’) => x é qntdd de casas decimais e trocou-se os separadores decimais . por , replace é facultativo o uso.
* n.toLocaleString(‘pt’, {style: ‘currency’, currency: ‘EUR’}) => transformar em moeda

1. Operadores de lógica

* n %= 5 ⬄ n=n%5 (% é o resto da divisão)
* n\*\*=3 ⬄ n=n\*\*3 (\*\* é elevado a)
* n++ ⬄ n=n+1 (funciona com n-- também)
* 5 == ‘5’ => True (o JVS testa a forma do q esta estrito)
* 5 === ‘5’ => False (=== sinal de idêntico, agora sim testa o type of também do q foi escrito)
* ! => não lógico, negação 1º na hierarquia de resolução
* && => e lógico, conjunção 2º na hierarquia de resolução
* || => ou lógico, disjunção 3º na hierarquia de resolução

**Operador Ternário**: teste ? resposta se true : resposta se false

Ex: media >= 7 ? ‘Aprovado’ : ‘Reprovado’

1. Captura de Elementos

* window.document.getElementsByX() => X pode ser:TagName, Id, Class, Name, Selector
* window.document.write(‘….' + X.innerHTML) =>pode usar Text no lugar do HTML, mas não pega a formatação em HTML do elemento, apenas o texto cru.
* window.document.querySelector(tag simbolo valor, por exem: article#artg1, artg1 é um id)