



Formatação de Numbers

Apesar de números não conterem métodos próprios pois não são objetos, é possível fazer modificações nos mesmos.

Métodos →

- **toString():**

Retorna um valor numérico como uma string. Podemos usá-lo em variáveis, números e até mesmo em expressões:

```
let numero = 15;

numero.toString();           // 15, valor da variável numero

(9).toString();              // 9, valor do inteiro 9

((9) + numero).toString();   // 24, valor da soma de ambos
```

- **toExponential():**

Retorna uma string com um número arredondado e com notação exponencial. O número de casas desejáveis deve ser passada por parâmetro:

```
let numero = 14;

numero.toExponential();      // 1.4e+1

(9).toExponential(2);        // 9.00e+0

((9) + numero).toExponential(4); // 2.3000e+1

Math.PI.toExponential(4);     // 3.1416e+0
```

- **toFixed():**

Retorna uma string com determinadas casas decimais. O número de casas decimais desejadas deve ser passada por parâmetro:

```
let numero = 10.989898989898989;

numero.toFixed(4);           // 10.9899

Math.PI.toFixed(2);          // 3.14
```

- **toPrecision():**

Retorna uma string com um número escrito conforme a quantidade especificada por parâmetro:

```
let numero = 20.987;

numero.toPrecision(2);       // 21

numero.toPrecision();         // 20.987

numero.toPrecision(3);       // 21.0

(Math.PI).toPrecision(3);     // 3.14
```

- **valueOf():**

Retorna um número como um número. No Javascript, um número pode ter dois tipos: valor primitivo e objeto. Esse método é usado internamente para converter objetos Number em valores primitivos, não há motivos para utilizá-lo em nossos códigos:

```
let numero = 15;

numero.valueOf();           // 15, valor da variável numero

((9) + numero).valueOf();    // 24, valor da soma de ambos

console.log((9).valueOf());  // 9, valor do inteiro 9
```

- **Number():**

Retorna um número a partir do valor atribuído à uma variável:

```
let boolean1 = true;
Number(boolean1);      // 1

let boolean2 = false;
Number(boolean2);      // 0

let data = new Date();
Number(data);          // 1630950195580

let string1 = 'Clóvis';
Number(string1);       // NaN

let string2 = '20';
Number(string2);       // 20
```

- **parseInt():**

Retorna um número inteiro a partir do valor atribuído à uma string. Somente o primeiro valor será retornado caso tenha mais de um:

```
parseInt("-0.999");      // -0
parseInt("9.999");      // 9
parseInt("7.8");        // 7
parseInt("7 9 10");     // 7
parseInt("clóvis 10");  // NaN
```

- **parseFloat():**

Retorna um número a partir do valor atribuído à uma string. Somente o primeiro valor será retornado caso tenha mais de um:

```
parseFloat("-9.876");    // -9.876
parseFloat("710");      // 710
parseFloat("7.10");     // 7.1
parseFloat("7 10");     // 7
parseFloat("clóvis 10"); // NaN
```

Bibliografias:

- https://www.w3schools.com/js/js_number_methods.asp;
- https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/Number/toFixed;
- https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/Number/toExponential;
- https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/Number/toPrecision;
- https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/Number/toString;