

## Para saber mais: Mais sobre números

Programas funcionam manipulando valores, como o número `3.14` ou o texto `Juliana e Leonardo`. Os tipos de valores que podem ser representados e manipulados em uma linguagem de programação são conhecidos como tipos, e uma das características mais fundamentais de uma linguagem de programação é o conjunto de tipos que ela suporta.

### Variáveis

Quando um programa precisa reter um valor para uso futuro, ele atribui o valor a (ou “armazena” o dado em) uma variável. As variáveis têm **nomes** e permitem o uso desses nomes em nossos programas para se referir a valores. A maneira como as variáveis funcionam é outra característica fundamental de qualquer linguagem de programação.

Os tipos de JavaScript podem ser divididos em duas categorias: tipos **primitivos** e tipos de **objetos**. Os tipos primitivos do JavaScript incluem números, palavras ou texto (conhecidas como **strings**) e valores booleanos (conhecidos como **booleanos**).

### Tipos numéricos

Como vimos em aula, podemos armazenar números de diferentes formas:

```
const idade = 28  
const pi = 3.14
```

[COPIAR CÓDIGO](#)

Dica: podemos utilizar o número [PI](https://pt.wikipedia.org/wiki/Pi) (<https://pt.wikipedia.org/wiki/Pi>) através do código `Math.PI` .

O ponto flutuante pode ter um ponto decimal; eles usam a sintaxe tradicional para números reais. Um valor real é representado como a parte integral do número, seguido por um ponto decimal e a parte fracionária do número.

Pontos flutuantes também podem ser representados usando notação exponencial: um número real seguido pela letra *e* (ou *E*), seguido por um sinal opcional de mais (+) ou menos (-), e por um expoente inteiro. Essa notação representa o número real multiplicado por 10 à potência do expoente.

Divisão por zero não é um erro em JavaScript: ele simplesmente retorna “*Infinity*”. No entanto, há uma exceção: zero dividido por zero não tem um valor bem definido e o resultado dessa operação é o valor especial não numérico NaN.

```
var a = 10
var b = 0
console.log(a/b) // Infinity
```

[COPIAR CÓDIGO](#)

```
var a = 0
var b = 0
console.log(a/b) // NaN
```

[COPIAR CÓDIGO](#)

## Links

- [Como formatar número com JavaScript](https://www.alura.com.br/artigos/formatando-numeros-no-javascript)  
(<https://www.alura.com.br/artigos/formatando-numeros-no-javascript>)

- [Como ordenar uma sequência de números no JavaScript](https://www.alura.com.br/artigos/ordenacao-de-numeros-no-javascript-nao-funciona)  
(<https://www.alura.com.br/artigos/ordenacao-de-numeros-no-javascript-nao-funciona>).