

console.log()

O `console.log` é um dos principais comandos utilizados para desenvolvimento de aplicações JavaScript. Seu funcionamento é bem simples, ele exibe o conteúdo no console.

Sintaxe:

```
console.log(texto);  
console.log(obj1[, obj2, ..., objN]);
```

Observe o exemplo:

```
1 console.log("cadeira");  
2 console.log(cadeira);
```

Explicando cada uma das linhas:

Na primeira linha, temos o termo a ser exibido entre aspas, e ao ser executado, vai exibir o texto **cadeira** no console.

Na segunda linha, o argumento - termo - a ser exibido está sem aspas, e isso pode ser confuso. Ao ser executado, vai exibir o valor da variável **cadeira** no console. Se você não definir a variável **borracha** antes do **console.log**, será exibido **undefined**, ou seja, não definido. Veja:

```
1 console.log(borracha);  
2  
3 var borracha = "caneta";  
4 console.log(borracha);
```

A primeira linha vai exibir **undefined** no console, pois ainda não foi definido o que é a variável **caneta**.

Na terceira linha, foi definida a variável **borracha**, e seu valor é o texto - ou string - caneta. Já na quarta linha, repetimos o comando, mas dessa vez vai exibir caneta no console, pois é o valor atribuído à variável **borracha**.

Para ver mais sobre **console.log** e variáveis, confira esses links úteis:

<https://developer.mozilla.org/pt-BR/docs/Web/API/console/log>

<https://pt.wikibooks.org/wiki/JavaScript/Vari%C3%A1veis>

<http://tableless.github.io/iniciantes/manual/js/variaveis.html>

<https://developer.mozilla.org/pt-BR/docs/Web/API/console/log>

https://www.w3schools.com/jsref/met_console_log.asp

function

Um dos principais comandos utilizados em JavaScript são as functions, ou funções. São conjuntos de comandos que são executados e podem retornar algum valor.

É como na matemática, veja só:

$$f(x) = x + 2$$

Foi criada uma função chamada **f**, que recebe o argumento **x**. Ao ser resolvida - ou executada - resulta no valor de **x** somado com 2.

$$f(5) = 5 + 2, \text{ ou seja, } 7$$

$$f(0) = 0 + 2, \text{ ou seja, } 2$$

$$f(10) = 10 + 2, \text{ ou seja, } 12$$

Confira agora como se fazem funções em JavaScript:

```
1 function f(x) {  
2     return x + 2;  
3 }
```

Na primeira linha, temos o início da função - **function** -, o nome da função - **f** -, o argumento ou parâmetro - **x** -, e a chave de abertura para o corpo da função.

Na segunda linha - e até fechar a chave - temos o comando a ser executado pela função. No exemplo acima, a função vai **retornar** $x + 2$.

A terceira linha é o fechamento da chave, e consequentemente o fim da função.

```
1 f(5);  
2 f(0);  
3 f(10);|
```

Seguindo o mesmo modo de funcionamento de uma função matemática, ao ser executada, a função criada vai **retornar** 7, 2, e 12 respectivamente.

Vale ressaltar que ao executar o exemplo acima, nada vai aparecer no console. Para fazer isso, é necessário usar o comando **console.log()**.

Veja o exemplo:

```
1 console.log(f(5));|
```

Agora sim, ao ser executado, o computador vai rodar de dentro para fora, então a primeira coisa que vai fazer é calcular a função. Essa função por sua vez vai retornar 7 ($5 + 2$). Por último, o computador vai exibir o resultado da função no console.