

# Chamada de função JavaScript

< Anterior

Próximo >

## Método de Reutilização

Com o `call()` método, você pode escrever um método que pode ser usado em diferentes objetos.

## Todas as funções são métodos

Em JavaScript, todas as funções são métodos de objeto.

Se uma função não é um método de um objeto JavaScript, é uma função do objeto global (consulte o capítulo anterior).

O exemplo abaixo cria um objeto com 3 propriedades, `firstName`, `lastName`, `fullName`.

### Exemplo

```
const myObject = {
  firstName:"John",
  lastName: "Doe",
  fullName: function () {
    return this.firstName + " " + this.lastName;
  }
}

// This will return "John Doe":
myObject.fullName();
```

Tente você mesmo "

## The `this` Keyword

In a function definition, `this` refers to the "owner" of the function.

In the example above, `this` is the **person object** that "owns" the **fullName** function.

In other words, `this.firstName` means the **firstName** property of **this object**.

Read more about the `this` keyword at [JS this Keyword](#).

## The JavaScript `call()` Method

The `call()` method is a predefined JavaScript method.

It can be used to invoke (call) a method with an owner object as an argument (parameter).

With `call()`, an object can use a method belonging to another object.

Este exemplo chama o método **fullName** de `person`, usando-o em **person1** :

### Exemplo

```
const person = {
  fullName: function() {
    return this.firstName + " " + this.lastName;
  }
}
const person1 = {
  firstName:"John",
  lastName: "Doe"
}
const person2 = {
  firstName:"Mary",
  lastName: "Doe"
}

// This will return "John Doe":
person.fullName.call(person1);
```

Tente você mesmo "

Este exemplo chama o método **fullName** de `person`, usando-o em **person2** :

### Exemplo

```
const person = {
  fullName: function() {
    return this.firstName + " " + this.lastName;
  }
}
const person1 = {
  firstName:"John",
  lastName: "Doe"
}
const person2 = {
  firstName:"Mary",
  lastName: "Doe"
}

// This will return "Mary Doe"
person.fullName.call(person2);
```

Tente você mesmo "

## O método `call()` com argumentos

O `call()` método pode aceitar argumentos:

### Exemplo

```
const person = {
  fullName: function(city, country) {
    return this.firstName + " " + this.lastName + ", " + city + ", " + country;
  }
}
const person1 = {
  firstName:"John",
  lastName: "Doe"
}

person.fullName.call(person1, "Oslo", "Norway");
```

Tente você mesmo "

< Anterior

Próximo >

Reportar erro

Fórum

Cerca de

Comprar

### Principais tutoriais

Tutorial HTML Tutorial  
CSS Tutorial  
JavaScript  
Como fazer Tutorial  
SQL Tutorial  
Python Tutorial  
W3.CSS Tutorial  
Bootstrap Tutorial  
PHP Tutorial  
Java Tutorial  
C ++ Tutorial  
jQuery Tutorial

### Top References

HTML Reference  
CSS Reference  
JavaScript Reference  
SQL Reference  
Python Reference  
W3.CSS Reference  
Bootstrap Reference  
PHP Reference  
HTML Colors  
Java Reference  
Angular Reference  
jQuery Reference

### Top Examples

HTML Examples  
CSS Examples  
JavaScript Examples  
How To Examples  
SQL Examples  
Python Examples  
W3.CSS Examples  
Bootstrap Examples  
PHP Examples  
Java Examples  
XML Examples  
jQuery Examples

### Web Courses

Curso de HTML Curso de  
CSS Curso de  
JavaScript Curso de  
front end Curso de  
SQL Curso de  
Python Curso de  
PHP Curso de  
jQuery Curso de  
Java Curso de  
C ++ Curso de  
C # Curso de  
XML

Obter certificação "

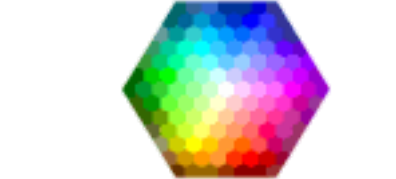
W3Schools é otimizado para aprendizagem e treinamento. Os exemplos podem ser simplificados para melhorar a leitura e o aprendizado. Tutoriais, referências e exemplos são constantemente revisados para evitar erros, mas não podemos garantir a correção total de todo o conteúdo. Ao usar o W3Schools, você concorda em ter lido e aceito nossos termos de uso , cookies e política de privacidade .

Copyright 1999-2021 de Refsnes Data. Todos os direitos reservados.  
W3Schools é desenvolvido por W3.CSS .



PROPAGANDA

COLOR PICKER



COMO NÓS



Obtenha a  
certificação  
completando  
um curso hoje!



iniciar

JOGO DE  
CÓDIGOS



Jogar um jogo