# Propriedades do objeto JavaScript

[❮ Anterior](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/js_object_definition.asp&usg=ALkJrhhnvVhz1XyDIfvFSu1XtdYmM-EmPA)[Próximo ❯](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/js_object_methods.asp&usg=ALkJrhjT3KiMekdZgJzjvqWfMIYEU121kw)

As propriedades são a parte mais importante de qualquer objeto JavaScript.

## Propriedades de JavaScript

As propriedades são os valores associados a um objeto JavaScript.

Um objeto JavaScript é uma coleção de propriedades não ordenadas.

As propriedades geralmente podem ser alteradas, adicionadas e excluídas, mas algumas são somente de leitura.

## Acessando propriedades de JavaScript

A sintaxe para acessar a propriedade de um objeto é:

*objectName.property*// person.age

ou

*objectName* [" *property* "]       // person["age"]

ou

*objectName* [ *expression* ]       // x = "age"; person[x]

A expressão deve avaliar para um nome de propriedade.

### Exemplo 1

person.firstname + " is " + person.age + " years old.";

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_properties1&usg=ALkJrhjkCMjLxUyE5Sp_QdvO-5OuykGPeA)

### Exemplo 2

person["firstname"] + " is " + person["age"] + " years old.";

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_properties2&usg=ALkJrhjSEoiEB63v-QhUR2MdoewgGmhHBA)

## JavaScript para ... em Loop

O JavaScript para ... na declaração faz o percurso através das propriedades de um objeto.

### Sintaxe

for ( *variable* in *object* ) {  
*code to be executed*   
}

O bloco de código dentro do for ... in loop será executado uma vez por cada propriedade.

Looping através das propriedades de um objeto:

### Exemplo

var person = {fname:"John", lname:"Doe", age:25};   
  
for (x in person) {  
    txt += person[x];  
}

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_properties_for_in&usg=ALkJrhiXA3pm65WwidfA-lXd8m0J_PM4LA)

## Adicionando novas propriedades

Você pode adicionar novas propriedades a um objeto existente simplesmente dando-lhe um valor.

Suponha que o objeto pessoa já existe - você pode dar-lhe novas propriedades:

### Exemplo

person.nationality = "English";

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_properties3&usg=ALkJrhjqgNoR4ga3mnjgggMTDVr9M9trUw)

Não é possível usar palavras reservadas para nomes de propriedade (ou método). Aplicam-se regras de nomeação de JavaScript.

## Eliminando Propriedades

A palavra-chave de **exclusão** exclui uma propriedade de um objeto:

### Exemplo

var person = {firstName:"John", lastName:"Doe", age:50, eyeColor:"blue"};  
delete person.age;   // or delete person["age"]; 

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_properties4&usg=ALkJrhg3pX0UnlxKz17V_FYvLMBugF8RDg)

A palavra-chave de exclusão exclui o valor da propriedade e a própria propriedade.

Após a exclusão, a propriedade não pode ser usada antes de ser adicionada novamente.

O operador de exclusão foi projetado para ser usado nas propriedades do objeto. Não tem efeito em variáveis ​​ou funções.

O operador de exclusão não deve ser usado em propriedades de objeto JavaScript predefinidas. Pode travar sua aplicação.

## Atributos da propriedade

Todas as propriedades têm um nome. Além disso, eles também têm um valor.

O valor é um dos atributos da propriedade.

Outros atributos são: enumerável, configurável e gravável.

Esses atributos definem como a propriedade pode ser acessada (é legível ?, É gravável?)

Em JavaScript, todos os atributos podem ser lidos, mas somente o atributo de valor pode ser alterado (e somente se a propriedade for gravável).

(ECMAScript 5 possui métodos para obter e definir todos os atributos de propriedade)

## Propriedades do protótipo

Os objetos JavaScript herdam as propriedades de seu protótipo.

A palavra-chave de exclusão não exclui propriedades herdadas, mas se você excluir uma propriedade de protótipo, afetará todos os objetos herdados do protótipo.