

- W3schools

Tutorials

References

Exercises
- Condições JS
- JS Switch
- JS Loop para
- JS Loop For In
- JS Loop para de
- JS Loop While
- JS Break
- JS Typeof
- Conversão de tipo JS
- JS Bitwise
- JS RegExp
- Erros JS
- Escopo JS
- JS Hoisting
- Modo JS Strict
- JS esta palavra-chave
- Função de seta JS
- Classes JS
- JS JSON
- Depuração JS
- Guia de estilo JS
- JS Best Practices

Elevação de JavaScript

< Anterior

Próximo >

Declarações de JavaScript são içadas

Em JavaScript, uma variável pode ser declarada após ter sido usada.
Em outras palavras; uma variável pode ser usada antes de ser declarada.

O **Exemplo 1** dá o mesmo resultado do **Exemplo 2** :

Exemplo 1

```
x = 5; // Assign 5 to x

elem = document.getElementById("demo"); // Find an element
elem.innerHTML = x;                      // Display x in the element

var x; // Declare x
```

Tente você mesmo "

Exemplo 2

```
var x; // Declare x
x = 5; // Assign 5 to x

elem = document.getElementById("demo"); // Find an element
elem.innerHTML = x;                      // Display x in the element
```

Tente você mesmo "

Para entender isso, você deve entender o termo "içamento".

Hoisting is JavaScript's default behavior of moving all declarations to the top of the current scope (to the top of the current script or the current function).

The let and const Keywords

Variables defined with **let** and **const** are hoisted to the top of the block, but not *initialized*.
Meaning: The block of code is aware of the variable, but it cannot be used until it has been declared.

Using a **let** variable before it is declared will result in a **ReferenceError** .

The variable is in a "temporal dead zone" from the start of the block until it is declared:

Example

This will result in a **ReferenceError** :

```
carName = "Volvo";
let carName;
```

Try it Yourself »

Using a **const** variable before it is declared, is a syntax error, so the code will simply not run.

Example

This code will not run.

```
carName = "Volvo";
const carName;
```

Try it Yourself »

Read more about let and const in [JS Let / Const](#).

JavaScript Initializations are Not Hoisted

JavaScript only hoists declarations, not initializations.

Example 1 does **not** give the same result as **Example 2**:

Example 1

```
var x = 5; // Initialize x
var y = 7; // Initialize y

elem = document.getElementById("demo"); // Find an element
elem.innerHTML = x + " " + y;           // Display x and y
```

Try it Yourself »

Example 2

```
var x = 5; // Initialize x

elem = document.getElementById("demo"); // Find an element
elem.innerHTML = x + " " + y;           // Display x and y

var y = 7; // Initialize y
```

Try it Yourself »

Does it make sense that y is undefined in the last example?

This is because only the declaration (var y), not the initialization (=7) is hoisted to the top.

Because of hoisting, y has been declared before it is used, but because initializations are not hoisted, the value of y is undefined.

Example 2 is the same as writing:

Example

```
var x = 5; // Initialize x
var y;     // Declare y

elem = document.getElementById("demo"); // Find an element
elem.innerHTML = x + " " + y;           // Display x and y

y = 7;     // Assign 7 to y
```

Try it Yourself »

Declare Your Variables At the Top !

Hoisting is (to many developers) an unknown or overlooked behavior of JavaScript.

Se um desenvolvedor não entende de içamento, os programas podem conter bugs (erros).

Para evitar bugs, sempre declare todas as variáveis no início de cada escopo.

Visto que é assim que o JavaScript interpreta o código, é sempre uma boa regra.

JavaScript no modo estrito não permite que variáveis sejam usadas se não forem declaradas.
Estude **"usar estrito"** no próximo capítulo.

< Anterior

Próximo >



COLOR PICKER



COMO NÓS



Obtenha a
certificação
completando
um curso hoje!



iniciar

JOGO DE
CÓDIGOS



Jogar um jogo

Reportar erro

Fórum

Cerca de

Comprar

Principais tutoriais

HTML Tutorial
CSS Tutorial
JavaScript Tutorial
How To 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

HTML Course
CSS Course
JavaScript Course
Front End Course
SQL Course
Python Course
PHP Course
jQuery Course
Java Course
C++ Course
C# Course
XML Course

Get Certified »

