# JavaScript HTML DOM Navigation

[❮ 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_htmldom_eventlistener.asp&usg=ALkJrhg0-F2ZeMHSXCYVuCV8pAEAXo01Zw)[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_htmldom_nodes.asp&usg=ALkJrhiBw50eCQj-RMZVgb-xn7dzzv9HVw)

Com o HTML DOM, você pode navegar na árvore do nó usando os relacionamentos dos nós.

## Nós DOM

De acordo com o padrão W3C HTML DOM, tudo em um documento HTML é um nó:

* Todo o documento é um nó de documento
* Todo elemento HTML é um nó de elemento
* O texto dentro dos elementos HTML são nós de texto
* Cada atributo HTML é um nó de atributos (obsoleto)
* Todos os comentários são nós de comentários



Com o HTML DOM, todos os nós na árvore de nós podem ser acessados ​​pelo JavaScript.

Novos nós podem ser criados e todos os nós podem ser modificados ou excluídos.

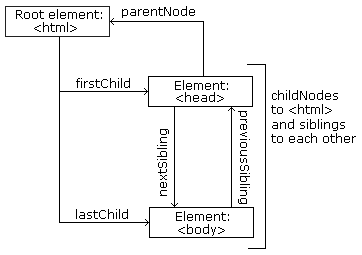
## Nó de Relacionamentos

Os nós na árvore de nós têm uma relação hierárquica entre si.

Os termos pai, filho e irmão são usados ​​para descrever os relacionamentos.

* Em uma árvore de nó, o nó superior é chamado de raiz (ou nó raiz)
* Cada nó tem exatamente um dos pais, exceto a raiz (que não tem pai)
* Um nó pode ter uma série de filhos
* Irmãos (irmãos ou irmãs) são nós com o mesmo pai

<html>  
  
  <head>  
      <title>DOM Tutorial</title>  
  </head>  
  
  <body>  
      <h1>DOM Lesson one</h1>  
      <p>Hello world!</p>  
  </body>  
  
</html>



Do HTML acima você pode ler:

* <html> é o nó raiz
* <html> não tem pais
* <html> é o pai de <head> e <body>
* <head> é o primeiro filho de <html>
* <body> é o último filho de <html>

e:

* <head> tem um filho: <title>
* <title> tem um filho (um nó de texto): "DOM Tutorial"
* <body> tem dois filhos: <h1> e <p>
* <h1> tem um filho: "DOM Lesson one"
* <p> tem um filho: "Olá mundo!"
* <h1> e <p> são irmãos

## Navegando entre nós

Você pode usar as seguintes propriedades de nó para navegar entre nós com JavaScript:

* parentNode
* childNodes [ nodenumber ]
* primeiro filho
* lastChild
* nextSibling
* irmã anterior

## Nodos infantis e valores de nó

Um erro comum no processamento do DOM é esperar que um elemento nó contenha texto.

### Exemplo:

<title id="demo">DOM Tutorial</title>

O nó do elemento <title> (no exemplo acima) **não** contém texto.

Contém um **nó de texto** com o valor "DOM Tutorial".

O valor do nó de texto pode ser acessado pela propriedade **innerHTML** do nó:

var myTitle = document.getElementById("demo").innerHTML;

Acessar a propriedade innerHTML é o mesmo que acessar o **nodeValue** do primeiro filho:

var myTitle = document.getElementById("demo").firstChild.nodeValue;

O acesso ao primeiro filho também pode ser feito assim:

var myTitle = document.getElementById("demo").childNodes[0].nodeValue;

Todos os (3) exemplos seguintes recuperam o texto de um elemento <h1> e o copiam para um elemento <p>:

### Exemplo

<html>  
<body>  
  
<h1 id="id01">My First Page</h1>  
<p id="id02"></p>  
  
<script>  
document.getElementById("id02").innerHTML = document.getElementById("id01").innerHTML;  
</script>  
  
</body>  
</html>

[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_dom_nav_innerhtml1&usg=ALkJrhhHyxp_tuohAEsNAyvvz1O8bFIATw)

### Exemplo

<html>  
<body>  
  
<h1 id="id01">My First Page</h1>  
<p id="id02"></p>  
  
<script>  
document.getElementById("id02").innerHTML = document.getElementById("id01").firstChild.nodeValue;  
</script>  
  
</body>  
</html>

[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_dom_nav_innerhtml2&usg=ALkJrhidI0eeatiXJ0fvZby8IJcMlRVUTw)

### Exemplo

<html>  
<body>  
  
<h1 id="id01">My First Page</h1>  
<p id="id02">Hello!</p>  
  
<script>  
document.getElementById("id02").innerHTML = document.getElementById("id01").childNodes[0].nodeValue;  
</script>  
  
</body>  
</html>

[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_dom_nav_innerhtml3&usg=ALkJrhi4xB9weuxt9nq4TYf-eH703a4T0A)

## InnerHTML

Neste tutorial, usamos a propriedade innerHTML para recuperar o conteúdo de um elemento HTML.

No entanto, aprender os outros métodos acima é útil para entender a estrutura da árvore e a navegação do DOM.

## Nódulos Raiz DOM

Existem duas propriedades especiais que permitem o acesso ao documento completo:

* document.body - O corpo do documento
* document.documentElement - O documento completo

### Exemplo

<html>  
<body>  
  
<p>Hello World!</p>  
<div>  
<p>The DOM is very useful!</p>  
<p>This example demonstrates the <b>document.body</b> property.</p>  
</div>  
  
<script>  
alert(document.body.innerHTML);  
</script>  
  
</body>  
</html>

[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_dom_body&usg=ALkJrhjJHhXUZKYOjMBJrbbL80G-uEC2Dw)

### Exemplo

<html>  
<body>  
  
<p>Hello World!</p>  
<div>  
<p>The DOM is very useful!</p>  
<p>This example demonstrates the <b>document.documentElement</b> property.</p>  
</div>  
  
<script>  
alert(document.documentElement.innerHTML);  
</script>  
  
</body>  
</html>

[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_dom_document&usg=ALkJrhgj5o4Fz9vW7Pn5yPFAloe7oG6IGg)

## A propriedade nodeName

A propriedade nodeName especifica o nome de um nó.

* nodeName é somente leitura
* nodeName de um nó de elemento é o mesmo que o nome da etiqueta
* nodeName de um nó de atributos é o nome do atributo
* nodeName de um nó de texto é sempre #text
* nodeName do nó do documento é sempre #document

### Exemplo

<h1 id="id01">My First Page</h1>  
<p id="id02"></p>  
  
<script>  
document.getElementById("id02").innerHTML = document.getElementById("id01").nodeName;  
</script>

[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_dom_nav_nodename&usg=ALkJrhgAMUl9KgfUaT-pAgd82xr70JW38A)

**Nota:** nodeName sempre contém o nome da tag em maiúscula de um elemento HTML.

## A propriedade nodeValue

A propriedade nodeValue especifica o valor de um nó.

* nodeValue para nós de elemento é indefinido
* nodeValue para nós de texto é o próprio texto
* nodeValue para nós de atributo é o valor do atributo

## A propriedade nodeType

A propriedade nodeType é somente leitura. Ele retorna o tipo de um nó.

### Exemplo

<h1 id="id01">My First Page</h1>  
<p id="id02"></p>  
  
<script>  
document.getElementById("id02").innerHTML = document.getElementById("id01").nodeType;  
</script>

[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_dom_nav_nodetype&usg=ALkJrhh_YwkOEBM5tUIULHo_e6e4r-2Pzg)

As propriedades nodeType mais importantes são:

|  |  |  |
| --- | --- | --- |
| **Node** | **Type** | **Example** |
| ELEMENT\_NODE | 1 | <h1 class="heading">W3Schools</h1> |
| ATTRIBUTE\_NODE | 2 | class = "heading" (deprecated) |
| TEXT\_NODE | 3 | W3Schools |
| COMMENT\_NODE | 8 | <!-- This is a comment --> |
| DOCUMENT\_NODE | 9 | The HTML document itself (the parent of <html>) |
| DOCUMENT\_TYPE\_NODE | 10 | <!Doctype html> |

O tipo 2 está obsoleto no HTML DOM (mas funciona). Não está obsoleto no XML DOM.