# JavaScript HTML DOM Elements (Nodes)

[❮ 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_navigation.asp&usg=ALkJrhg96aZ_EKauFuKYpnHbjbEEkhTDpA)[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_collections.asp&usg=ALkJrhhNCaC15EZqF2Ls5Vg2nxUj3l2yWA)

Adicionando e removendo nós (elementos HTML)

## Criando novos elementos HTML (nós)

Para adicionar um novo elemento ao DOM HTML, você deve primeiro criar o elemento (nó do elemento) e, em seguida, anexá-lo a um elemento existente.

### Exemplo

<div id="div1">  
<p id="p1">This is a paragraph.</p>  
<p id="p2">This is another paragraph.</p>  
</div>  
  
<script>  
var para = document.createElement("p");  
var node = document.createTextNode("This is new.");  
para.appendChild(node);  
  
var element = document.getElementById("div1");  
element.appendChild(para);  
</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_elementcreate&usg=ALkJrhhyBW6u7plPb4aoNDj-UMZldsQPrg)

## Exemplo Explicado

Este código cria um novo elemento <p>:

var para = document.createElement("p");

Para adicionar texto ao elemento <p>, primeiro você deve criar um nó de texto. Este código cria um nó de texto:

var node = document.createTextNode("This is a new paragraph.");

Em seguida, você deve anexar o nó de texto ao elemento <p>:

para.appendChild(node);

Finalmente, você deve anexar o novo elemento a um elemento existente.

Este código encontra um elemento existente:

var element = document.getElementById("div1");

Esse código adiciona o novo elemento ao elemento existente:

element.appendChild(para);

## Criando novos elementos HTML - insertBefore ()

O método appendChild () no exemplo anterior, anexou o novo elemento como o último filho do pai.

Se você não quer que você possa usar o método insertBefore ():

### Exemplo

<div id="div1">  
<p id="p1">This is a paragraph.</p>  
<p id="p2">This is another paragraph.</p>  
</div>  
  
<script>  
var para = document.createElement("p");  
var node = document.createTextNode("This is new.");  
para.appendChild(node);  
  
var element = document.getElementById("div1");  
var child = document.getElementById("p1");  
element.insertBefore(para, child);  
</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_elementcreate2&usg=ALkJrhhs3JBS6KUYl_S0s73--f_OSpJ3mA)

## Removendo elementos HTML existentes

Para remover um elemento HTML, você deve conhecer o pai do elemento:

### Exemplo

<div id="div1">  
<p id="p1">This is a paragraph.</p>  
<p id="p2">This is another paragraph.</p>  
</div>  
  
<script>  
var parent = document.getElementById("div1");  
var child = document.getElementById("p1");  
parent.removeChild(child);  
</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_elementremove&usg=ALkJrhh7imRfLUiMXrb3JglQfvlhZ07AuA)

O método node.remove () é implementado na especificação DOM 4.   
Mas devido ao suporte do navegador pobre, você não deve usá-lo.

## Exemplo Explicado

Este documento HTML contém um elemento <div> com dois nós secundários (dois elementos <p>):

<div id="div1">  
<p id="p1">This is a paragraph.</p>  
<p id="p2">This is another paragraph.</p>  
</div>

Encontre o elemento com id = "div1":

var parent = document.getElementById("div1");

Encontre o elemento <p> com id = "p1":

var child = document.getElementById("p1");

Remova o filho do pai:

parent.removeChild(child);

Seria bom poder remover um elemento sem se referir ao pai.   
Mas desculpa. O DOM precisa conhecer tanto o elemento que deseja remover quanto o pai.

Aqui está uma solução comum: encontre a criança que deseja remover e use sua propriedade parentNode para encontrar o pai:

var child = document.getElementById("p1");  
child.parentNode.removeChild(child);

## Substituindo elementos HTML

Para substituir um elemento pelo DOM HTML, use o método replaceChild ():

### Exemplo

<div id="div1">  
<p id="p1">This is a paragraph.</p>  
<p id="p2">This is another paragraph.</p>  
</div>  
  
<script>  
var para = document.createElement("p");  
var node = document.createTextNode("This is new.");  
para.appendChild(node);  
  
var parent = document.getElementById("div1");  
var child = document.getElementById("p1");  
parent.replaceChild(para, child);  
</script>