

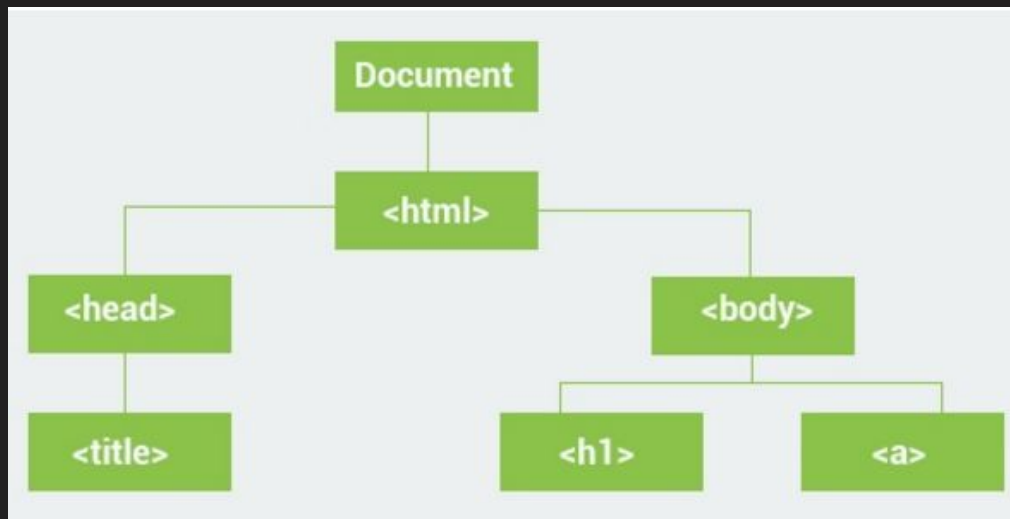
Este curso é de propriedade do aplicativo SoloLearn e foi traduzido por Ana Beatriz Augusto usando os recursos Reverso Context e Google Tradutor.

# O DOM

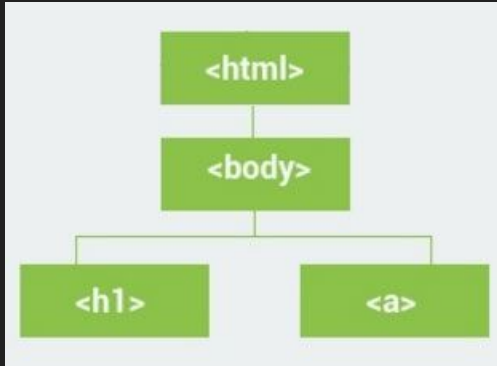
Quando você abre qualquer página pelo navegador, o HTML da página é carregado e mostrado na tela.

Para fazer isso, o navegador constrói o DOM (Documento Objeto Modelo) da página, que é um modelo orientado a objetos de sua estrutura lógica.

O DOM de um documento HTML pode ser representado como um conjunto aninhado de caixas:



Por exemplo, vamos usar um HTML que é representado pela seguinte estrutura:



O elemento `html` é pai do elemento `body` e parente de todos os elementos abaixo dele.

O elemento `body` é pai dos elementos `h1` e `a`.

Os elementos `h1` e `a` são filhos do elemento `body` e descendentes do elemento `html`.

Os elementos `h1` e `a` são irmãos (tem o mesmo pai).

Resumo:

Um ancestral é um pai, vô, bisavô e assim em diante.

Um descendente é um filho, neto, bisneto e assim em diante.

Irmãos têm o mesmo pai.

Entender a relação entre os elementos DOM é importante para se ter a capacidade de percorrer o DOM corretamente.

# Percorrendo o DOM

O jQuery tem muitos métodos úteis para se percorrer o DOM.

O método `parent()` retorna o direto elemento pai do elemento selecionado.

Por exemplo:

HTML:

```
<div> Elemento div  
  <p> Paragrafo </p>  
</div>
```

JavaScript:

```
var e = $("p").parent();  
e.css("border", "2px solid red");
```

O código acima seleciona o elemento pai do parágrafo e define uma borda vermelha a ele.

O método `parent()` apenas pode percorrer um único nível na árvore DOM. Para pegar todos os ancestrais do elemento selecionado, você pode usar o método `parents()`. Por exemplo:

HTML:

```
<body>  body
  <div style="width:300px;"> div
    <ul> ul
      <li> li
        <p> Parágrafo </p>
      </li>
    </ul>
  </div>
</body>
```

Este código define uma borda vermelha para todos os pais do parágrafo.

JavaScript:

```
$(function(){
  var e =
$("p").parents();
  e.css("border", "2px
solid red");
});
```

Alguns dos métodos mais usados para percorrer o DOM são os seguintes:

Método	O que ele retorna
parent()	O direto elemento pai do elemento selecionado.
parents()	Todos os elementos ancestrais do elemento selecionado.
children()	Todos os diretos elementos filhos do elemento selecionado.
siblings()	Todos os elementos irmãos.
next() / nextAll()	O próximo/todos os próximos elementos irmãos.
prev() / prevAll()	O anterior/todos os elementos anteriores do elemento selecionado.
eq()	Um elemento com um índice numérico específico do elemento selecionado.

O elemento `eq()` pode ser usado para selecionar um elemento específico entre múltiplos elementos.

Por exemplo, se a página contém múltiplos elementos `div` e queremos selecionar a terceira `div`:

```
$("div").eq(2);
```

Os números dos índices começam em 0, então o primeiro elemento tem índice 0.



# Remova elementos

Removemos elementos específicos do DOM usando o método `remove()`. Por exemplo:

HTML:

```
<p style="color:red">Vermelho</p>
<p style="color:green">Verde</p>
<p style="color:blue">Azul</p>
```

JavaScript:

```
$("p").eq(1).remove();
```

Este código remove o segundo elemento `p`.

Você também pode usar o método `remove()` em múltiplos elementos, por exemplo `$("p").remove()` remove todos os parágrafos.

# Removendo Conteúdo

O método `empty()` é usado para remover os elementos filhos do(s) elemento(s) selecionado(s). Por exemplo:

HTML:

```
<div>  
  <p style="color:red"> Vermelho </p>  
  <p style="color:green"> Verde </p>  
  <p style="color:blue"> Azul </p>  
</div>
```

JavaScript:

```
$("#div").empty();
```

Este código remove todos os três elementos da `div`, a deixando vazia.

Este curso é de propriedade do aplicativo SoloLearn e foi traduzido por Ana Beatriz Augusto usando os recursos Reverso Context e Google Tradutor.