



Referências

[https://pt.wikipedia.org/wiki/Mashup_\(aplica%C3%A7%C3%A3o_web\)](https://pt.wikipedia.org/wiki/Mashup_(aplica%C3%A7%C3%A3o_web))

<https://www.w3schools.com/html/default.asp>

Jon Duckett - HTML & CSS Design and Build Websites

jQuery e Manipulação do DOM

Professor Rafael Escalfoni

jQuery e Manipulação do DOM

1. Manipulação do DOM
2. Adicionando/ Removendo elementos
3. Navegando no DOM



1

Manipulação do DOM

Métodos `html()`, `text()` e `val()`

Manipulação de DOM

(Document Object Model)

Obtendo conteúdo

- `text()` – texto de um elemento selecionado
- `html()` – conteúdo do elemento, incluindo marcações de tags filhas
- `val()` – valor de campos de formulário

```
$(seletor).text();
```

```
$(seletor).html();
```

```
$(seletor).val();
```

- `attr(nomeAtrib)` – valor de um atributo

```
$(seletor).attr(nomeAtrib);
```

Alterando conteúdo

Obtendo conteúdo

- `text(val)` – onde `val` é somente texto
- `html(val)` – onde `val` pode ser texto e tags
- `val(val)` – onde `val` deve ser compatível com o type do elemento de formulário

```
$(seletor).text("valor exemplo");
```

```
$(seletor).html("<h1>valor exemplo</h1>");
```

```
$(seletor).val("valor exemplo");
```

- `attr(nomeAtrib, val)` – valor de um atributo

```
$(seletor).attr(nomeAtrib, "valor atrib");
```

exemplo:

```
$("#w3s").attr({  
  "href" : "http://www.w3schools.com/jquery",  
  "title" : "W3Schools jQuery Tutorial"  
});
```

Setando attr com callback

Função callback tem 2 parâmetros:

- `i` – índice do elemento atual na lista de elementos selecionados
- `origValue` – valor original do atributo

```
$("#w3s").attr("href", function(i, origValue){  
    return origValue + "/jquery";  
});
```



2

**Adicionando/
Removendo
elementos**

Adicionando conteúdo HTML

`append(val)` – insere **val** no fim do elemento selecionado

`prepend(val)` – insere **val** no início do elemento selecionado

`after(val)` – insere **val** depois do elemento selecionado

`before(val)` – insere **val** antes do elemento selecionado

Removendo elementos

`remove()` – remove elemento selecionado

`empty()` – apaga os filhos e conteúdo

```
$(seletor).remove();
```

```
$(seletor).empty();
```

`remove(filtro)` – remove aceita filtros seletores

```
$("p").remove(".test, #demo");
```

Manipulando CSS

`addClass(className)` – adiciona classe(s)

`removeClass(className)` – remove classe(s)

`toggleClass(className)` – adiciona/remove classe(s)

`css()` – seta/retorna atributo estilo

```
$(seletor).css("propriedadeCSS");
```

```
$(seletor).css("propriedadeCSS", "valor");
```

```
$(seletor).css({"propriedadeCSS1":"valor1", "propriedadeCSS2":"valor2", ...})
```

Métodos de dimensionamento

`width()`: retorna a largura de um elemento

`height()`: retorna a altura de um elemento

`innerWidth()`: retorna a largura de um elemento (inclui padding)

`innerHeight()`: retorna a altura de um elemento (inclui padding)

`outerWidth()`: retorna a largura de um elemento (inclui padding+borda)

`outerHeight()`: retorna a altura de um elemento (inclui padding+borda)

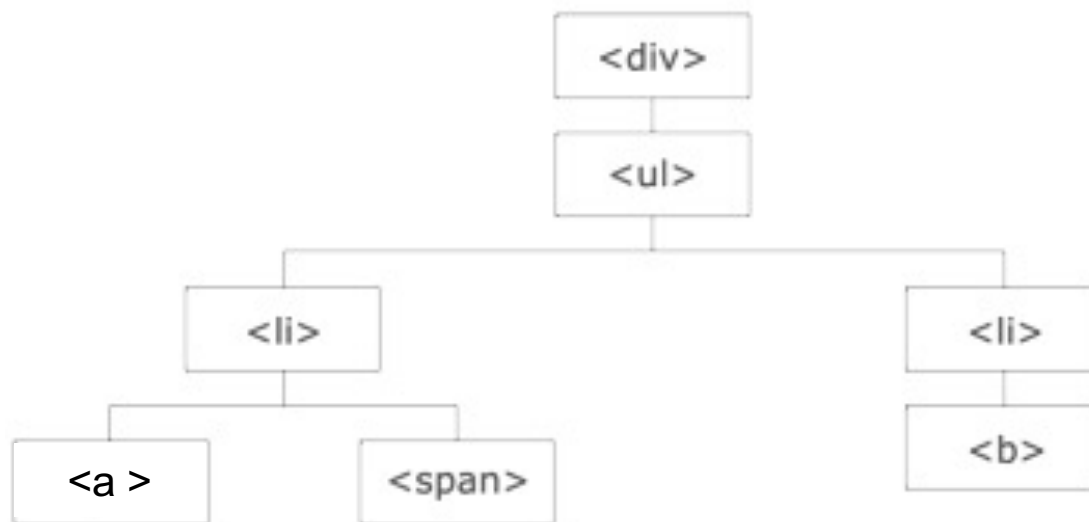
A decorative graphic on the left side of the slide. It consists of several horizontal bands: a thin dark teal band at the top, a large dark teal band in the middle containing a large, light teal number '3', a thin dark green band below the '3', and a light green band at the bottom.

3

Navegando no DOM

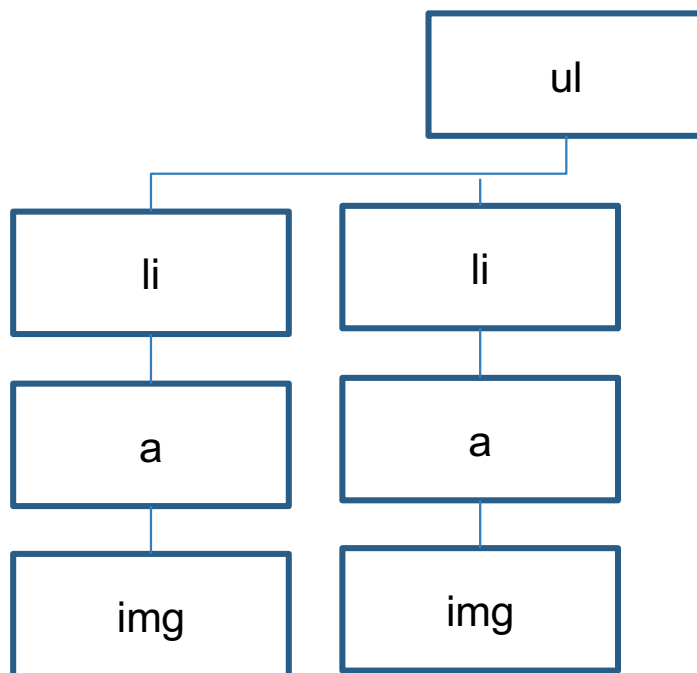
Cruzando o DOM (traversing)

Por vezes, precisamos ir de um ramo para outro na árvore DOM



Cruzando o DOM (traversing)

Por vezes, precisamos ir de um ramo para outro na árvore DOM



“Subindo” na árvore

`parent()` – retorna o pai do nó selecionado

```
$(seletor).parent();
```

`parents()` – retorna todos os ancestrais de um nó selecionado

```
$(seletor).parents();
```

```
$(seletor1).parents(seletor2); /* todos os ancestrais que satisfaçam o  
seletor2 */
```

`parentsUntil()` – retorna todos os ancestrais entre o nó e o argumento

```
$(seletor1).parentsUntil(seletor2); /* todos os nós entre o seletor1 e  
seleto2 */
```

“Descendo” a árvore

`children()`: retorna os nós filhos **diretos** do elemento selecionado

```
$(seletor).children();
```

```
$(seletor).children(seletor2); /* filhos de seletor que satisfaçam  
seletor2 */
```

`find()`: retorna todos os nós filhos do elemento selecionado que satisfaça a condição

```
$(seletor).find(seletor2);
```


Nós irmãos

`siblings()` – todos os irmãos

`next()` – próximo imediato

`nextAll()` – todos os próximos

`nextUntil()` – todos os próximos até condição de parada

`prev()` – anterior imediato

`prevAll()` – todos os anteriores

`prevUntil()` – todos os anteriores até condição de parada

Filtros de busca

`first()` – primeiro que satisfaça a condição

`last()` – último que satisfaça a condição

```
$(seletor).first();
```

```
$(seletor).last();
```

`eq()` – com índice igual

```
$(seletor).eq(indice);
```

`filter()` – satisfaça o seletor2:

```
$(seletor).filter(seletor2);
```

`not()` – todos que não satisfaçam o critério

```
$(seletor).not(seletor2);
```