

11

Introdução às
Tecnologias Web

jQuery II



Perguntas

Qual o principal passo numa instrução jQuery?

Diferença entre *getters* e *setters*?

**A ordem dos métodos no *chaining*
(encadeamento) é relevante? Porquê?**

Qual o significado?

```
35 <ul>
36   <li>Home Page</li>
37   <li>Instruções do Jogo</li>
38   <li>Quadro de Honra</li>
39   <li>Lista de Classificações</li>
40   <li>Jogar</li>
41   <li>Acerca dos Autores</li>
42   <li>Contactos</li>
43 </ul>
```

```
11 .azul {
12   color: blue;
13   border: 1px solid blue;
14   background-color: white;
15 }
16 .verde {
17   color: green;
18   border: 1px solid green;
19   background-color: white;
20 }
21 .invAzul {
22   color: white;
23   border: 1px solid white;
24   background-color: blue;
25 }
26 .invVerde {
27   color: white;
28   border: 1px solid white;
29   background-color: green;
30 }
```

```
46 function resetColor () {
47   $("li:odd").addClass("azul");
48   $("li:even").addClass("verde");
49   $("li").removeClass("invAzul");
50   $("li").removeClass("invVerde");
51 }
52
53 function ff () {
54   resetColor();
55   if($(this).hasClass("azul")) {
56     $(this).removeClass("azul");
57     $(this).addClass("invAzul");
58   } else {
59     $(this).removeClass("verde");
60     $(this).addClass("invVerde");
61   }
62 };
63
64 $(document).ready(function(){
65   resetColor();
66   $("li").click(ff);
67 });
```

Qual o significado?

```
35 <ul>
36   <li>Home Page</li>
37   <li>Instruções do Jogo</li>
38   <li>Quadro de Honra</li>
39   <li>Lista de Classificações</li>
40   <li>Jogar</li>
41   <li>Acerca dos Autores</li>
42   <li>Contactos</li>
43 </ul>
```

```
11 .azul {
12   color: blue;
13   border: 1px solid blue;
14   background-color: white;
15 }
16 .verde {
17   color: green;
18   border: 1px solid green;
19   background-color: white;
20 }
21 .invAzul {
22   color: white;
23   border: 1px solid white;
24   background-color: blue;
25 }
26 .invVerde {
27   color: white;
28   border: 1px solid white;
29   background-color: green;
30 }
```

• Home Page • Home Page
• Instruções • Instruções do Jogo
• Quadro de • Quadro de Honra
• Lista de Cl • Lista de Classificações

```
46 function resetColor () {
47   $("li:odd").addClass("azul");
48   $("li:even").addClass("verde");
49   $("li").removeClass("invAzul");
50   $("li").removeClass("invVerde");
51 }
52
53 function ff () {
54   resetColor();
55   if($(this).hasClass("azul")) {
56     $(this).removeClass("azul");
57     $(this).addClass("invAzul");
58   } else {
59     $(this).removeClass("verde");
60     $(this).addClass("invVerde");
61   }
62 };
63
64 $(document).ready(function(){
65   resetColor();
66   $("li").click(ff);
67 });
```

Resumo Aula Anterior

Introdução ao jQuery

Biblioteca JavaScript, facilita código

Executa ações sobre conjuntos de elementos

Seletores

Base do jQuery

Iguais aos usados em CSS

Tipos, Métodos e Eventos

Mesmo método devolve e altera valores

Eventos iguais aos do JS

CSS e Posicionamento

Absolutos (offset) e relativos (position)

Sumário

jQuery e DOM

jQuery e Forms

Animação

Eventos

01

JQUERY E DOM

jQuery e DOM

Leitura

Filtragem e navegação no DOM

Leitura de atributos/propriedades

Escrita

Adição, remoção, alteração e cópia de elementos

Edição de propriedades

Obter Conteúdo Elementos

`$(...).html()` – Devolve o conteúdo HTML do elemento

```
<div id="d1">  
  <div id="d2">Azul, Verde, Vermelho</div>  
</div>
```

`$('#d2').html();` -> Azul, Verde, Vermelho

`$('#d1').html();` -> <div id="d2"> Azul, Verde, Vermelho</div>

Definir Conteúdo Elementos

`$(...).html(novoHTML)` – Altera o conteúdo HTML de cada um dos elementos da seleção

```
<div id="d1">  
  <div id="d2">Azul, Verde, Vermelho</div>  
</div>
```

`$('#d1').html('<p>teste do html</p>');`

```
<div id="d1">  
  <p>teste do html</p>  
</div>
```

Obter/Definir Texto de Elementos

`$(...).text()` – Devolve o texto do elemento HTML

`$(...).text(novoTexto)` – Altera texto do elemento HTML

```
<div id="d1">  
  <div id="d2">Azul, Verde, Vermelho</div>  
</div>
```

`$('#d2').text();` -> Azul, Verde, Vermelho

`$('#d2').text('Verde e Branco');`

.html() <> .text()

`$('#d1').html('<p>teste do html</p>');`

```
<div id="d1">  
  <p>teste do html</p>  
</div>
```

`$('#d1').text('<p>teste do html</p>');`

```
<div id="d1">  
  &lt;p&gt;teste do html&lt;/p&gt;  
</div>
```

html() e text()

E se `$(...)` devolver vários elementos?

Setter do `html` e `text` – Altera todos para o mesmo texto

Getter do `html` e `text` – Concatena os textos de todos

```
<div id="d1">  
  <p>Parágrafo um</p>  
  <p>Segundo parágrafo</p>  
  <p>E mais um parágrafo</p>  
</div>
```

```
alert($('p').text());
```



JavaScript Alert

Parágrafo umSegundo parágrafoE mais um parágrafo

☐ Prevent this page from creating additional dialogs.

OK

Iterar numa Seleção usando each()

Qual o resultado disto?

```
alert($('li').offset().top);
```

Escreve a coordenada top do 1º elemento da seleção

Como escrevo as coordenadas top de todos os elementos da seleção?

Iterar numa Seleção usando each()

`$(...).each(func)` – Percorre os vários elementos da seleção, executando uma função com cada um deles

Índice do elemento da seleção

```
function showEach(idx) {  
    console.log(idx + ": " +  
        $(this).offset().top);  
}  
$('li').each(showEach);
```

Refere-se ao elemento da seleção

ADICIONAR ELEMENTOS

Criar Elementos

\$() também permite criar elementos

```
$(' <p id="p1">Parágrafo</p>');
```

Criar elemento com atributos

Tag + atributos

```
$('<p/>', { 'id': 'p1', 'text': 'Parágrafo' });
```

Adicionar Elementos (como filhos)

\$(s1).append(novoElem) – Adiciona novoElem à seleção s1 no fim

```
$('#dd1').append('<p id="p1">Parágrafo</p>');
```

```
$('#dd1').append($('<p/>', { 'id': 'p1',  
    'text': 'Parágrafo' }));
```

\$(novoElem).appendTo(s1) – Adiciona novoElem à seleção s1 no fim

```
$('<p/>', { 'id': 'p1',  
    'text': 'Parágrafo' }).appendTo('#dd1');
```

Adicionar Elementos (como filhos)

`$(s1).prepend(novoElem)`

`$(novoElem).prependTo(s1)`

Iguais aos anteriores, mas inserem no início

Inserir Antes e Depois (como irmãos)

`$(s1).after(novoElem)`

`$(novoElem).insertAfter(s1)`

Adiciona novoElem depois da seleção s1

`$(s1).before(novoElem)`

`$(novoElem).insertBefore(s1)`

Adiciona novoElem antes da seleção s1

REMOVER ELEMENTOS

Remover Elementos

```
<div id="dd1">  
  <p>Parágrafo um</p>  
  <p>Segundo parágrafo</p>  
</div>
```

\$(s1).empty() – Remove todos os filhos da seleção s1

```
$( '#dd1' ).empty(); ->  
  <div id="dd1">  
  </div>
```

\$(s1).remove() – Remove a seleção s1 e todos os seus filhos

```
$( '#dd1' ).remove(); -> (vazio)
```

PROPRIEDADES E ATRIBUTOS

Propriedades

`$(...).prop(nomeProp)` – Devolve o valor da propriedade para o primeiro elemento da seleção

Ex: `var src = $('img').prop('src');`

`$(...).prop(nomeProp, valor)` – Atribui um valor à propriedade nomeProp, para todos os elementos da seleção

Ex: `$('img').prop('alt', 'Imagem');`

`$(..).removeProp(nomeProp)` – Remove a propriedade nomeProp, de todos os elementos da seleção

Atributos

Métodos semelhantes, mas para atributos

`$(...).attr(nomeAttr)`

Ex: `var cc = $('div').attr('class');`

`$(...).attr(nomeAttr, valor)`

Ex: `$('div').attr('class', 'verde');`

`$(..).removeAttr(nomeAttr)`

Recomenda-se o uso de **`prop()`** em vez de **`attr()`**

02

JQUERY E FORMS

jQuery e Forms

Seletores especiais p/ formulários

Métodos de acesso e alteração de valores

Formas de detetar estados dos elementos

Obter/Definir Valores de Elementos

Elementos de input de um formulário

`<input>`, `<select>`, `<textarea>`

`$(...).val()` – Devolve o valor do primeiro elemento da seleção

`$(...).val(novoValor)` – Altera o valor do campo em todos os elementos da seleção

```
// Devolve valor do campo nome
$( 'input#nome' ).val();
// Altera valor do campo nome
$( 'input#nome' ).val( 'Manuel' );
```

Seletores para Formulários

Seletor	Seleção
<code>':enabled'</code>	Elementos que estão ativos
<code>':disabled'</code>	Elementos desativados
<code>':focus'</code>	Elemento que tem o foco
<code>':checked'</code>	<i>Checkboxes</i> e <i>radio buttons</i> que estão selecionados
<code>':selected'</code>	Opção ou opções selecionadas num <i>select</i>

Concatenar estes seletores com o formulário

```
$('#registo .genero:checked').val()  
$('#registo .paises:selected').val()  
$('#registo input:radio[name=gender]:checked').val()
```

Ativar e Desativar Elementos

Pode-se fazer usando **is()** e os seletores **':enabled'** e **':disabled'**

`$(...).is()` – Compara a seleção com um seletor, elemento ou objeto jQuery (devolve um booleano)

```
var mForm = $('#registo');  
mForm.find('[name=gender]').is(':enabled');  
mForm.find('[name=gender']).prop('enabled', true);  
  
mForm.is('img');
```

Manipular Elementos Select

Alteração dinâmica dos elementos de um *select*
Podemos usar os métodos da família *append()*

```
var paises = $('#paises');  
// Adicionar opção no fim da lista  
paises.append('<option value="pt">Portugal</option>');  
// Adicionar opção no início da lista  
paises.prepend('<option value="pt">Portugal</option>');
```

03

ANIMAÇÃO

Efeitos Básicos

`$(...).show()` e `$(...).hide()` – Mostra / esconde os elementos de uma seleção

Podem ser animados

Usar tempo como parâmetro

Podem chamar uma função no final

Event handler

```
$('#btn').show();  
$('#btn').show('slow');  
$('#btn').show(700, fimShow);  
$('#li').hide();
```

Fading (desvanecimento)

`$(...).fadeIn()` e `$(...).fadeOut()` – Mostra / esconde os elementos de uma seleção usando opacidade/transparência

```
$('#btn').fadeIn(3000);  
$('#btn').fadeOut('slow');
```

`$(...).fadeTo()` – Faz *fading* para um nível de opacidade

```
$('#btn').fadeTo('slow', 0.5);
```

Sliding (deslizamento)

`$(...).slideDown()` e **`$(...).slideUp()`** – Estica / encolhe os elementos de uma seleção na vertical

```
$('#img1').slideUp(1000);  
$('#img1').slideDown('fast');
```

Animações Personalizadas

`$(...).animate(valsObjetivo, duração, ...)` – Anima os elementos da seleção, durante um tempo, até atingirem os valsObjetivo (CSS)

```
$('img').animate({ padding: '30', width: '300px'},  
2000);
```

Pode-se chamar uma função no final

```
$('img').animate({ padding: '-=30px',  
width: '+=100px'}, 2000, fimAnim);
```

Parar uma Animação

```
$(...).stop()
```

```
$('img').fadeOut(5000);
```

```
$('img').stop();
```

04

EVENTOS

Associar Eventos a elementos HTML

```
$( "p" ).click(function() {  
    // código para o click  
});  
  
$( "p" ).on( "click", function() {  
    // código para o click  
});
```

Associar Eventos a elementos HTML

```
$( "p" ).on( {  
    mouseenter: function() {  
        $(this).css( "background-color",  
        "lightgray" ); },  
    click: function() {  
        $(this).css( "background-color",  
        "yellow" ); }  
});
```

Associa um ou mais *Event Handlers* aos elementos selecionados

http://www.w3schools.com/jquery/jquery_ref_events.asp

Remover Eventos a elementos HTML

```
var f1 = function() {  
    console.log("teste 1");}  
var f2 = function() {  
    console.log("teste 2");}  
$('p').on({click: f1, click: f2});  
$('p').off('click');  
$('p').off('click', f2);
```

Remove **todos** os event handlers para click

Remove **apenas** o event handler **f2**

O Objeto *Event*

Criado quando ocorre um evento

Tem informação sobre: o quê, quando e onde

```
$("#p").on('click', function(event) {  
    console.log(event.pageX);  
});
```

```
function myFunc(event) {  
    console.log(event.pageX);  
};  
$("#p").on('click', myFunc);
```

Propriedades do objeto *Event*

Propriedade	Descrição
type	O tipo de evento (ex. click)
target	O elemento que desencadeou o evento
data	Dados extra opcionalmente passados ao evento
pageX pageY	Posição do rato onde o evento ocorreu
timeStamp	Data em que o evento ocorreu (milissegundos desde 1-Jan-1970)
which	Botão do rato premido (em eventos do rato) Código da tecla (em eventos do teclado)

05

RESUMINDO

Resumo

jQuery e DOM

- Adicionar e remover elementos
- Obter e alterar propriedades/atributos

jQuery e Forms

- Obter valores de elementos do formulário
- Seletores para formulários

Animação

- Efeitos básicos, *fading* e *sliding*

Eventos

- Associar e remover eventos
- O objeto *Event* e a sua informação

Próxima Aula

