



Referências

<https://w3schools.com/jquery>

Curso WD-43 – Desenvolvimento Web com HTML, CSS e JavaScript CAELUM

Jon Duckett – Javascript & jQuery – interactive front-end web development

D. S. McFarland – JavaScript & jQuery – the missing manual

jQuery

Biblioteca JavaScript cross-browser

jQuery is a lightweight, “write less, do more”, JavaScript library

Tabelas e Frames

1. Introdução
2. Aplicações do jQuery
3. Efeitos



1

jQuery

Introdução

JQuery

O JQuery é uma biblioteca criada para simplificar scripts client-side que interagem com HTML

- Lançada em 2006
- Resolve os problemas:
 - Incompatibilidade entre navegadores
 - Reduz código
 - Promove a reutilização de código através de plugins
 - Componentização através de plugins
 - Trabalha com AJAX e DOM

Simplificar código JavaScript

JavaScript puro:

```
document.getElementById("teste").value = 5;
```

Mesmo código em JQuery:

```
$("#teste").val(5);
```

Como usar JQuery

Faça o download da versão atual estável em jquery.com

Acrescente o arquivo jquery.js na sua página.

```
<script src="jquery.min.js"></script>
```

o atributo `type="text/javascript"` não é mais necessário em HTML5

A decorative graphic on the left side of the slide. It consists of several horizontal bands of color: a thin dark teal band at the top, a large dark teal band in the middle containing a large, light teal number '2', a thin dark green band below the '2', and a light green band at the bottom.

2

jQuery

Aplicações

Sintaxe básica jQuery

A maior parte do uso do Jquery trata a manipulação de elementos HTML ao executar algum evento

- Eventos
- Efeitos
- Manipulação de DOM
- AJAX
- Plugins

Seletores JQuery

`$(seletor).metodo()`

- `$` - define o acesso ao objeto jQuery
- `seletor` – elemento a ser manipulado
- `metodo()` – método/ação a ser executado
- Exemplos:

```
$(this).hide(); //elemento atual
```

```
$("p").hide(); //todos os parágrafos da página
```

```
$(".test").hide(); //classe test
```

```
$("#test").hide(); //id=test
```

Os seletores são os mesmos do CSS

Document Ready Event

Para que os métodos jQuery executem apenas quando a página estiver carregada (ready), coloque seu código dentro do evento document ready event:

```
$(document).ready(function(){  
    // seu código aqui  
});
```

Seu uso é tão comum que foi simplificado para:

```
$(function(){  
    // seu código aqui  
});
```

Eventos JQuery

Mouse Events	Keyboard Events	Form Events	Document/Window Events
click	keypress	submit	load
dblclick	keydown	change	resize
mouseenter	keyup	focus	scroll
mouseleave		blur	unload

- Um evento representa o momento preciso quando alguma coisa acontece
 - Sintaxe para eventos:
 - Para a maioria dos eventos DOM tem um evento jquery equivalente:
- ```
$("#p").click(//fazer algo aqui!!!);
```

## O método on()

Organize seu código com o método `on()`. Assim, você poderá listar todos os eventos de um elemento:

```
$(“p”).on(“click”, function({
 alert(“callback”);
}));
```

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

3

**jQuery**

Efeitos

## Efeitos JQuery

| Method                               | Description                                                                                         |
|--------------------------------------|-----------------------------------------------------------------------------------------------------|
| <a href="#"><u>animate()</u></a>     | Runs a custom animation on the selected elements                                                    |
| <a href="#"><u>clearQueue()</u></a>  | Removes all remaining queued functions from the selected elements                                   |
| <a href="#"><u>delay()</u></a>       | Sets a delay for all queued functions on the selected elements                                      |
| <a href="#"><u>dequeue()</u></a>     | Removes the next function from the queue, and then executes the function                            |
| <a href="#"><u>fadeIn()</u></a>      | Fades in the selected elements                                                                      |
| <a href="#"><u>fadeOut()</u></a>     | Fades out the selected elements                                                                     |
| <a href="#"><u>fadeTo()</u></a>      | Fades in/out the selected elements to a given opacity                                               |
| <a href="#"><u>fadeToggle()</u></a>  | Toggles between the <a href="#"><u>fadeIn()</u></a> and <a href="#"><u>fadeOut()</u></a> methods    |
| <a href="#"><u>finish()</u></a>      | Stops, removes and completes all queued animations for the selected elements                        |
| <a href="#"><u>hide()</u></a>        | Hides the selected elements                                                                         |
| <a href="#"><u>queue()</u></a>       | Shows the queued functions on the selected elements                                                 |
| <a href="#"><u>show()</u></a>        | Shows the selected elements                                                                         |
| <a href="#"><u>slideDown()</u></a>   | Slides-down (shows) the selected elements                                                           |
| <a href="#"><u>slideToggle()</u></a> | Toggles between the <a href="#"><u>slideUp()</u></a> and <a href="#"><u>slideDown()</u></a> methods |
| <a href="#"><u>slideUp()</u></a>     | Slides-up (hides) the selected elements                                                             |
| <a href="#"><u>stop()</u></a>        | Stops the currently running animation for the selected elements                                     |
| <a href="#"><u>toggle()</u></a>      | Toggles between the <a href="#"><u>hide()</u></a> and <a href="#"><u>show()</u></a> methods         |

## Efeitos Hide & Show, Toggle

Sintaxe:

```
$(seletor).hide(speed*, callback*);
```

```
$(seletor).show(speed*, callback*);
```

```
$(seletor).toggle(speed*, callback*);
```

speed em milisegundos ou [“slow” | ”fast”]

\*opcionais

## Efeito fade

fadeIn

fadeOut

fadeToggle

- `$(seletor).fadeIn(speed*, callback*);`
- `$(seletor).fadeOut(speed*, callback*);`
- `$(seletor).fadeToggle(speed*, callback*);`

fadeTo

- `$(seletor).fadeTo(speed*, opacity*, callback*);`

## Efeito Slide

slideDown

slideUp

slideToggle

```
$(seletor).slideDown(speed*, callback*);
```

```
$(seletor).slideUp(speed*, callback*);
```

```
$(seletor).slideToggle(speed*, callback*);
```

## Efeito Animation

```
$(selector).animation({params}, speed, callback);
```

Exemplo:

```
$("#button").click(function(){
 $("#div").animate({
 left: '250px',
 opacity: '0.5',
 height: '150px',
 width: '150px'
 });
});
```

```
$("#button").click(function(){
 $("#div").animate({
 height: 'toggle' //hide ou show
 });
});
```

## Efeito stop

Parar um efeito antes do seu término previsto

```
$(seletor).stop(stopAll, goToEnd);
```

- **stopAll** – se houver uma fila de efeitos, para todos. Por default, é false.
- **goToEnd** – encerra a animação mas leva ao fim do efeito. default = false;

```
$("#start").click(function(){
 $("#div").animate({left: '100px'}, 5000);
 $("#div").animate({fontSize: '3em'}, 5000);
});
```

```
$("#stop").click(function(){
 $("#div").stop();
});
```

```
$("#stop2").click(function(){
 $("#div").stop(true);
});
```

```
$("#stop3").click(function(){
 $("#div").stop(true, true);
});
```



## Referências

<https://w3schools.com/jquery>

Curso WD-43 – Desenvolvimento Web com HTML, CSS e JavaScript CAELUM

Jon Duckett – Javascript & jQuery – interactive front-end web development

D. S. McFarland – JavaScript & jQuery – the missing manual

# jQuery

## Biblioteca JavaScript cross-browser

jQuery is a lightweight, “write less, do more”, JavaScript library