



# 01- Introdução à biblioteca JavaScript jQuery

Prof. Alan

# Introdução ao jQuery



---

jQuery, uma biblioteca JavaScript desenvolvida por John Resig, e que se tornou uma das bibliotecas JavaScript mais populares na internet.



---

Sua criação teve como foco a simplicidade e o objetivo de facilitar nossa vida no desenvolvimento de aplicações que necessitariam de linhas e mais linhas de código para obtermos um determinado efeito, ou efetuar uma requisição Ajax. Com jQuery esse trabalho é substituído por poucas instruções, o que faz da jQuery uma ferramenta ideal para aqueles designers e desenvolvedores com pouco conhecimento em JavaScript.

# O que é a jQuery?



---

jQuery é uma biblioteca de funções em JavaScript que interage com o HTML, desenvolvida para simplificar os scripts interpretados no navegador do usuário (client-side). Criada em dezembro de 2006 no BarCamp de Nova York por John Resig. Usada por cerca de 77% dos 10 mil sites mais visitados do mundo, jQuery é a mais popular das bibliotecas JavaScript

# jQuery é uma biblioteca de código aberto



---

(open source) A sintaxe do jQuery foi desenvolvida para simplificar a navegação em documentos HTML, a seleção de elementos DOM, criar animações, manipular eventos, desenvolver aplicações AJAX e criação de plugins sobre ela. Permitindo aos desenvolvedores criarem camadas de abstração para interações de baixo nível de modo simplificado em aplicações web de grande complexidade



---

A jQuery é leve, seu tamanho é em torno de 30kb, extensível, oferece suporte a plug-ins e conta ainda com uma grande equipe de desenvolvedores que vem diariamente adicionando novos recursos e funções a esta biblioteca, nos disponibilizando uma grande quantidade de controles para interface.

# Utilidade do jQuery



---

Podemos utilizar a jQuery para:

- Adicionarmos efeitos visuais e animações;
- Acessarmos e manipularmos o DOM;
- Carregarmos componentes Ajax;
- Provermos interatividade;
- Fazer alteração de conteúdo;
- Simplificarmos tarefas JavaScript.

# Suporte ao jQuery



---

A jQuery foi desenvolvida para ser uma biblioteca com suporte a qualquer navegador Web. Ela facilita a nós desenvolvedores a muitas vezes difícil tarefa de desenvolvermos aplicações em JavaScript, tendo que atingir a enorme quantidade de navegadores em que nossa programação poderá ser executada. Como sabemos, cada navegador possui seu próprio conjunto de características de implementação que ainda pode dificultar mais ainda, de acordo com a variação de plataforma e sistema operacional onde esteja executando. Já com a jQuery, nossa programação é única e transparente.





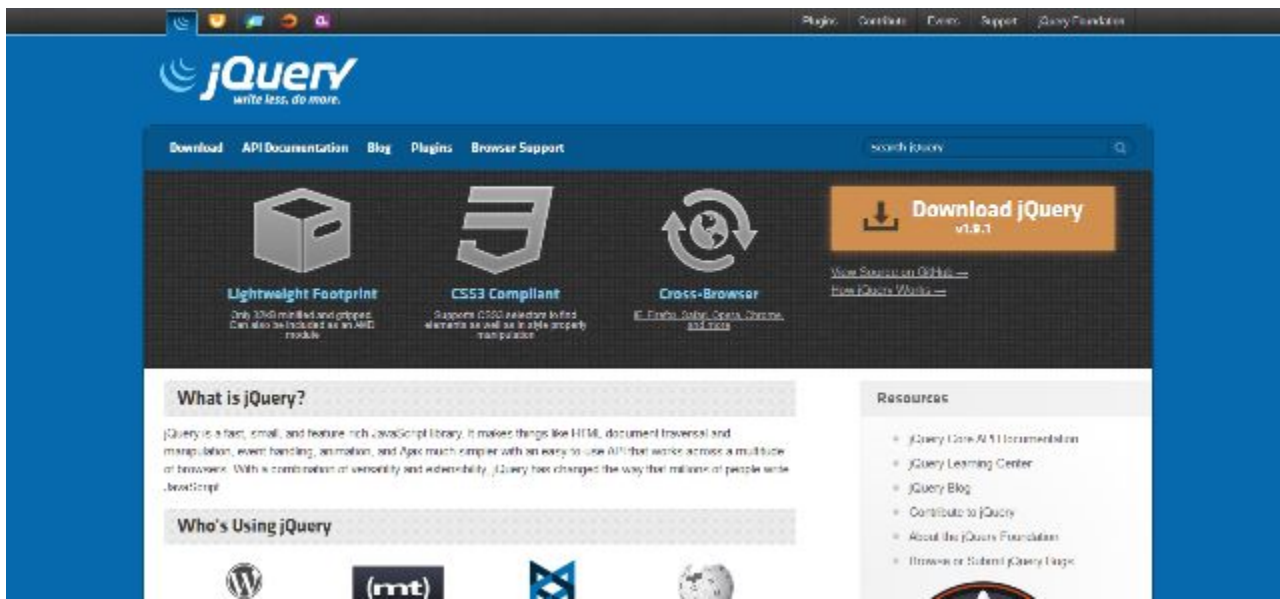
---

Com a jQuery possuímos suporte também às CSS3, onde podemos utilizar seletores CSS3 mesmo que o navegador não tenha suporte a esta folhas de estilo. Isso é possível porque a própria jQuery implementa os seletores CSS3, o que faz com que ela seja independente do navegador em que estiver sendo executada.

# Instalação do jQuery



Podemos estar obtendo a jQuery gratuitamente no site.  
<https://jquery.com/>



# diferença entre as versões



---

**Production:** Esta versão serve para ambientes de produção, é uma versão com o código em formato compactado, sem quebra de linhas e com o código sem comentários, possuindo apenas 15% do tamanho da que veremos a seguir.

**Development:** Esta versão é para desenvolvimento, é a versão não compactada da biblioteca e possui o código todo comentado. Sendo ideal para ambientes de desenvolvimento, por se integrar facilmente com as ferramentas e IDEs.



---

Versão production: `jquery.min.js`;

Versão Development: `jquery.js`.

# 01 Incluindo arquivo jQuery no HTML



---

Agora que já fizemos o download e salvamos a jQuery dentro da pasta do nosso projeto, vamos referenciar a mesma utilizando o parâmetro SRC da tag SCRIPT dentro da tag HEAD.

```
1 <html>
2 <head>
3 <title>...</title>
4 <!-- Versão development jQuery -->
5 <script type="text/javascript" src="jquery.js"></script>
6 </head>
```

# Construtor jQuery



---

Com a jQuery temos que utilizar a função `$()` para encontrarmos um elemento HTML dentro da aplicação e utilizarmos as funções da biblioteca. Essa função [e tecnicamente conhecida como construtor ou função construtora e ela estará presente em todas as aplicações que utilizarmos a jQuery.



---

O construtor faz uso do seguinte parâmetro, onde alvo é um seletor CSS para TAG, ID ou classe.

`$ (alvo)`

Vejamos alguns **exemplos de sua utilização.**

`$('p') //tag`

`$('#conteudo') //identificador`

`$('.teste') //classe`



---

Como podemos ver, no exemplo acima estamos fazendo com que a jQuery encontre os elementos, H1, P, e os elementos com `id="conteúdo"` e a `class="teste"`. O requisito mínimo para a utilização da jQuery é saber utilizar os seletores CSS.



# Função `$(document).ready()`



---

Essa função é uma das primeiras coisas que devemos aprender na jQuery. De forma simples podemos dizer que essa função é responsável por executar o conteúdo do método `ready()`, tão logo o navegador tenha carregado todos os elementos HTML

---

O modo mais comum de utilizarmos ela é em conjunto com uma função anônima, contendo os comandos que desejamos executar. Vejamos um exemplo.

```
1 <script type="text/javascript">
2     $(document).ready(function(){
3
4     //o nosso código jquery vai aqui.
5
6     });
7 </script>
```

```
1 <head>
2 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
3 <title>Introdução jQuery </title>
4 <!-- Referencia a biblioteca jquery (JS) -->
5 <script type="text/javascript" src="jquery.js"></script>
6 <!-- Funcionalidades da página -->
7 <script type="text/javascript">
8 $(document).ready(function(){
9 alert("página carregada")
10 });
11 </script>
12 </head>
13 <body>
14 <h1>Página demonstração</h1>
15 <h2>jQuery biblioteca javascript.</h2>
16 </body>
17 </html>
```



## 02 - Funções básicas - jQuery

Prof. Alan



---

Como quase tudo o que fazemos quando estamos utilizando o jQuery lê ou manipula um modelo de objeto de documento (DOM), precisamos nos certificar que começamos adicionado eventos etc tão logo o DOM esteja pronto.

Crie um novo arquivo chamado index.html, abra este arquivo crie uma estrutura HTML e faça a instância da biblioteca jQuery:

```
1 $(document).ready(function() {  
2     alert("Olá mundo com jquery")  
3 });
```



---

Colocar um alert nesta função não faz muito sentido, pois o alert não requer que o DOM esteja carregado. Então vamos tentar algo mais sofisticado: Mostrar um alert quando clicarmos o link.

Adicione o seguinte código dentro do seu <body>:



```
<a href="#">Meu link</a>
```

Agora vamos criar uma lógica para quando clicarmos neste elemento “a” exibiremos o nosso alert:

```
<body>
  <a href="#">Meu link</a>

  <script src="jquery.js"></script>
  <script>
    $(document).ready(function() {
      $("a").click(function() {
        alert("Olá mundo!");
      });
    });
  </script>
</body>
```

```
$().click(function() {
```

```
});
```

Ao clicar no elemento  
realizará a função.



# Após clicar em “Meu link” você verá uma saída parecida com esta:



[Meu link](#)

Essa página diz  
Olá mundo!

OK



---

Vamos dar uma olhada no que estamos fazendo: `$("a")` é um seletor do jQuery, neste caso, ele seleciona todos os elementos `a`. O `$` por si só é um alias para a “classe” jQuery, por outro lado o `$()` constrói um novo objeto jQuery. A função `click()` que chamamos depois é um método do objeto jQuery. Ele liga o evento clique a todos os elementos selecionados (neste caso, um único elemento `a`) e executa a função fornecida quando o evento ocorre.



---

Isto é similar ao seguinte código:

```
<a href="#" onclick="alert('Olá mundo!')">Link</a>
```

Nós não precisamos escrever onclick para todo elemento. Nós temos uma separação clara de estrutura (HTML) e comportamento (JS), assim como separamos estrutura e formatação utilizando CSS.

# Usando seletores e eventos



---

O jQuery provê duas maneiras de selecionar elementos. A primeira utiliza uma combinação de seletores CSS e XPath passados como uma string para o construtor do jQuery (ex. `$("div > ul a")`). A segunda utiliza vários métodos do objeto jQuery. Ambas podem ser combinadas.

---

Primeiramente queremos selecionar a própria lista. A lista tem um ID “listaOrdenada”. No javascript clássico, você pode selecioná-la usando

`document.getElementById(“listaOrdenada”)`. Com o jQuery, nós fazemos isso assim:

```
1 $(document).ready(function() {  
2     $("#listaOrdenada").css("color", "red")  
3 });
```

-- --

```
<body>
  <ul id="listaOrdenada">
    <li>Link 01</li>
    <li>Link 02</li>
  </ul>
```



- Link 01
- Link 02



---

Desta forma nós estilizamos nossa página utilizando o jQuery ao invés de CSS. No exemplo acima o `css()` é uma função jQuery que adiciona um atributo CSS a um elemento neste caso o “listaOrdenada”.

Caso o “listaOrdenada” fosse uma classe selecionaríamos o elemento desta maneira:

```
$(".listaOrdenada").css("color", "red");
```

Veja que utilizamos o `.` no lugar de `#`. Isto significa que para selecionarmos uma classe com jQuery basta utilizar: `$(".nomeClasse")` e para selecionarmos um id: `$("#meuID")`





# `$(seletor).html();` Retornar um HTML

---

Adiciona o html dentro do código:

```
$(document).ready(function() {  
  $('.element').html("Meu texto adicionado via jquery!");  
});
```

E crie uma div com a **class element**:

```
<div class="element"></div>
```



---

Repare que não existe conteúdo dentro desta div, mas como utilizamos uma método `html()` você verá algo parecido com isto:

# Plugins JQuery



---

É a biblioteca Javascript mais famosa e mais utilizada em todo mundo, por isso também é a que mais possui plugins e variações de possibilidades para se desenvolver.

# Inputmask



---

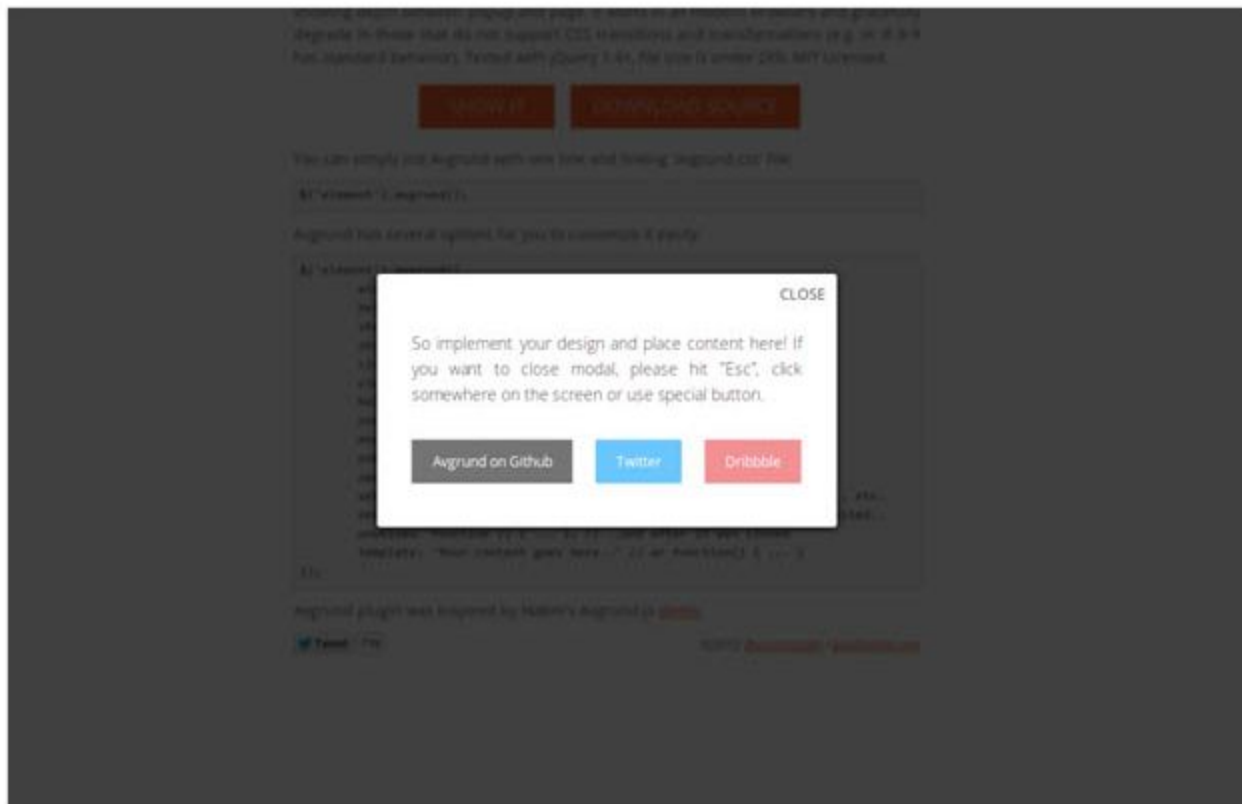
Não é tão rico em recursos, mas tem o fator surpreendente que as suas necessidades de aplicativos web. A caixa de diálogo é mostrada pela animação que traz em um ponto focalizado, enquanto o fundo é borrado e escurecido.

# jQuery Avgrund



---

Não é tão rico em recursos, mas tem o fator surpreendente que as suas necessidades de aplicativos web. A caixa de diálogo é mostrada pela animação que traz em um ponto focalizado, enquanto o fundo é borrado e escurecido.



# jCarousel

---

jCarousel é um plugin jQuery para controlar uma lista de itens em ordem horizontal ou vertical.

Ele fornece um conjunto de ferramentas flexível e cheio de recursos para navegar em qualquer conteúdo baseado em HTML em um carrossel.



## jCarousel

Riding carousels with jQuery



# Exercício Prático

— — —

