

FIAP



OPTIMIZATION TECHNOLOGY

1^o
semestre
2017

www.fiap.com.br

RENATO BORTOLIN
renatoboschini@gmail.com

Agenda

1. jQuery

O que é jQuery?

Lançada em 2006, **jQuery** é uma biblioteca JavaScript cross-browser desenvolvida para simplificar os scripts client side que interagem com o HTML.

Criada por **John Resig**, era um desenvolvedor web que trabalhava em seus próprios projetos. Ele estava frustrado com a dificuldade de escrever JavaScript que funcionasse em vários navegadores, e decidiu escrever sua própria biblioteca em JS. Com o lema "***Write less, do more.***" (escreva menos, faça mais), jQuery revolucionou o desenvolvimento web, sendo presente em inúmeros projetos atualmente.



Mas, por que é tão utilizada?

Exatamente pelo seu lema. Com **jQuery** é possível fazer diversos efeitos com poucas linhas e, que custariam dezenas de linhas em JavaScript puro.

Alguns recursos oferecidos facilmente pelo **jQuery**:

- Seleção e manipulação de elementos HTML;
- Manipulação de CSS;
- Efeitos e animações;
- *Navegação* pelo DOM;
- Ajax;
- Eventos;

Mas, como eu uso?

Igualmente qualquer arquivo JavaScript você precisa inseri-lo na página com as tags script.

Você pode fazer o link para uma **CDN** como a do [Google](#) ou fazer o download do framework no seu projeto e linkar direto (vale lembrar que se optar por essa opção, faça o link para a versão *minificada*).

```
<head>  
<script src="jquery-3.2.0.min.js"></script>  
</head>
```

```
<head>  
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.0/jquery.min.js"></script>  
</head>
```

Mas, como eu uso?

Antes de mais nada é necessário incluir o jQuery em nossa página. Só assim o navegador executará seu código para que possamos utilizar suas funcionalidades em nosso código.

Por isso é necessário que a tag `<script>` do jQuery seja a primeira de todas na ordem de nosso documento:

```
<script src="scripts/jquery.js"></script>  
<!-- só podemos utilizar o jQuery após sua  
importação -->  
<script src="scripts/meuscript.js"></script>  
<script src="scripts/meuoutroscript.js"></script>
```


Sintaxe

O jQuery é uma grande biblioteca que contém diversas funções que facilitam a vida do programador.

A mais importante delas, que inicia a maioria dos códigos, é a função \$.

Com ela é possível selecionar elementos com maior facilidade, maior compatibilidade, e com menos código.

Por exemplo:

```
document.getElementById('Teste').value = 5;
```

```
$('#Teste').val(5);
```

Sintaxe

Por exemplo:

```
document.getElementsByTagName("p")
```

```
$("p")
```

```
document.getElementById("um").setAttribute("class", "cor")
```

```
$("#um").attr("class", "cor")
```


Sintaxe

Com ela é possível selecionar elementos com maior facilidade, maior **compatibilidade**, e com menos código. Por exemplo:

```
// JavaScript "puro"
var cabecalho = document.getElementById("cabecalho");

if (cabecalho.attachEvent) {
    cabecalho.attachEvent("onclick", function (event) {
        alert("Você clicou no cabeçalho, usuário do IE!");
    });
} else if (cabecalho.addEventListener) {
    cabecalho.addEventListener("click", function (event) {
        alert("Você clicou no cabeçalho!")
    }, false);
}
```

```
// jQuery
$("#cabecalho").click(function (event) {
    alert("Você clicou no cabeçalho!");
});
```

Sintaxe

Existem diversas funções que o jQuery permite que utilizemos para alterar os elementos que selecionamos pela função \$, e essas funções podem ser encadeadas, por exemplo:

```
$("#cabecalho").css({"margin-top": "20px", "color": "#333333"}).addClass("selecionado");
```

No código acima, primeiramente chamamos a função \$ e passamos como argumento uma String idêntica ao seletor CSS que utilizaríamos para selecionar o elemento de id cabeçalho. Na sequência chamamos a função css e passamos um objeto como argumento, essa função adicionará ou alterará as informações desse objeto como propriedades de estilo do elemento que selecionamos com a função \$. Em seguida chamamos mais uma função, a addClass, que vai adicionar o valor "selecionado" ao atributo class do elemento com o id "cabeçalho".



Sintaxe

jQuery Selectors

Um dos maiores poderes do jQuery está na sua capacidade de selecionar elementos a partir de seletores CSS.

Por exemplo, se quisermos esconder todas as tags `<td>` filhas de um `<tbody>`, basta:

```
$('tbody td').hide();
```

Sintaxe

jQuery Selectors

Pinta o fundo do formulário com id "form" de preto

```
$('#form').css('background', 'black');
```

Esconde todos os elementos com o atributo "class" igual a "headline"

```
$('.headline').hide();
```

muda o texto de todos os parágrafos

```
$('p').text('alô :D');
```

Executar somente após carregar

Como estamos constantemente selecionando elementos do documento e alterando suas características, é importante garantir que os elementos que pretendemos utilizar já tenham sido carregados pelo navegador.

A melhor maneira de garantir isso é somente executar nosso script após o término do carregamento total da página com a função `$` dessa maneira:

```
$(document).ready(function() {  
  $("#cabecalho").css({"background-color": "#000000"});  
})
```

Essa função `$` que recebe uma função anônima como argumento garante que o código dentro dela só será executado ao fim do carregamento da página.

Executar somente após carregar

É uma boa prática esperar que o documento seja totalmente carregado e pronto antes de trabalhar com ele. Isso também permite que você tenha seu código JavaScript antes do corpo do seu documento, na seção de cabeçalho.

E ainda existe um método mais curto que seria:

```
$(function() {  
    $("#cabecalho").css({"background-color":  
"#000000"});  
})
```

Use a sintaxe que você preferir!

Vamos por em prática

Criei um projero e vamos fazer o **download** da biblioteca no site jQuery acessando o endereço <https://jquery.com/>

A screenshot of the jQuery website homepage. The browser's address bar shows "Seguro | https://jquery.com". The page has a blue header with the jQuery logo and the tagline "write less, do more.". Below the header is a navigation bar with links: Download, API Documentation, Blog, Plugins, and Browser Support. A search bar is also present. The main content area features three columns: "Lightweight Footprint" with a box icon, "CSS3 Compliant" with a stylized 'E' icon, and "Cross-Browser" with a circular arrow icon. A red rectangle highlights the "Download jQuery v3.2.0" button, which includes a download icon. Below this button are links for "View Source on GitHub" and "How jQuery Works".

← → ↻ Seguro | https://jquery.com

Apps ASUS E-Service Empire FT Filmes Torrent | Comod modem Chromebook wdmcloud/UI/

jQuery Foundation

Your donations help fund the continuous development and growth of jQuery.

SUPPORT THE PROJECT

Download API Documentation Blog Plugins Browser Support Search

Download jQuery
v3.2.0

View Source on GitHub →
How jQuery Works →

Lightweight Footprint
Only 32kB minified and gzipped. Can also be included as an AMD module

CSS3 Compliant
Supports CSS3 selectors to find elements as well as in style property manipulation

Cross-Browser
Chrome, Edge, Firefox, IE, Safari, Android, iOS, and more

Vamos por em prática



Vamos criar uma página simples em html

```
1  <!DOCTYPE html>
2  <html lang="en">
3      <head>
4          <meta charset="UTF-8">
5          <title>jQuery</title>
6      </head>
7      <body>
8          <h1>Minha primeira página utilizando jQuery</h1>
9
10         <p>jQuery é uma biblioteca de funções JavaScript que interage
            com o HTML</p>
11
12         <button>Clique aqui</button>
13     </body>
14 </html>
```

Vamos por em prática



Vamos fazer uma chamada com o jQuery

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8">
5     <title>jQuery</title>
6
7     <script src='js/jquery-3.2.0.min.js'></script>
8     <script src='js/meuScript.js'></script>
9
10  </head>
```

Ou podemos usar o acesso via **CDN**

```
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.2
.0/jquery.min.js">
</script>
```

Vamos por em prática

Na página de script **meuScript.js**, vamos implementar o código abaixo:



```
1 ▾ $(document).ready(function(){  
2     $("button").click(function(){  
3         $("p").hide();  
4     });  
5 });|
```

Vamos por em prática

Agora vamos inserir uma fonte nova para o título e alterar a sua cor para vermelho



```
exec1.html x meuScript.js x
1 ▾ $(document).ready(function(){
2 ▾     $("button").click(function(){
3     $("p").hide();
4
5     $("h1").css({"font-family":"arial","color":"red"});
6     });
7 });
8
```

Vamos por em prática

Utilizando um **seletor ID**, vamos implementar o HTML inserindo mais uma linha

```
<body>
  <h1>Minha primeira página utilizar
  <p>jQuery é uma biblioteca de func
  <div id='caixa'></div>
  <button>Clique aqui</button>
</body>
```



Vamos por em prática

Via script vamos implementar o novo elemento html



```
$(document).ready(function(){  
    $("button").click(function(){  
        $("p").hide();  
  
        $("h1").css({"font-family":"arial","color":"  
            red" });  
  
        $("#caixa").css({"width":"100px", "height":"  
            100px", "background-color":"blue"})  
    });  
});
```

Vamos por em prática

Utilizando um **seletor CLASS**, vamos implementar o HTML inserindo mais uma linha



```
15
16     <div id='caixa'></div>
17
18     <div class="painel"></div>
19     <div class="painel"></div>
20     <div class="painel"></div>
21
22     <button>Clique aqui</button>
```


Vamos por em prática

Via script vamos implementar o novo elemento html

```
$( ".paine1" ).css({ "border": "1px dashed red", "width": "50px", "height": "50px" });
```



Eventos

Todas as diferentes ações do visitante que uma página da Web pode responder são chamadas de **eventos**

Aqui estão alguns eventos DOM comuns:

Evento MOUSE

- *click*
- *dblclick*
- *mouseenter*
- *mouseleave*

Evento KEYBOARD

- *keypress*
- *keydown*
- *keyup*

Evento Form

- *submit*
- *change*
- *focus*
- *blur*

Evento Document/Window Events

- *load*
- *resize*
- *scroll*
- *unload*

Eventos

Evento mouse **click()**:

No HTML insira as tags

```
<p>Primeiro</p>  
<p>Segundo</p>  
<p>Terceiro</p>  
<p>Quarto</p>  
<p>Quinto</p>
```

Dentro da chamada **\$(document).ready(function(){**

```
· $("p").click(function(){  
· . . . . $(this).hide();  
· . . . })
```

Eventos

Evento mouse **dblclick()**:

No HTML insira as tags

```
<p>Primeiro</p>  
<p>Segundo</p>  
<p>Terceiro</p>  
<p>Quarto</p>  
<p>Quinto</p>
```

Dentro da chamada `$(document).ready(function(){`

```
$("p").dblclick(function(){  
    $(this).hide();  
});
```

FIAP



Copyright © **2017** Prof. Renato Bortolin Boschini

Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito, do Professor (autor).