

JQuery



Profª Anna Giselle Ribeiro
anna.ribeiro@imd.ufrn.br

Agenda

- O que é jQuery?
- Por que usar?
- Como inserir em minhas páginas?
- Hello Word jQuery

O que é jQuery?

- jQuery é uma biblioteca JavaScript.
 - Provavelmente a biblioteca JS mais utilizada da web.
- É uma compilação de métodos e funções que permitem agilizar a vida do Desenvolvedor Web quando se trabalha com JavaScript.
- Conhecida também como uma biblioteca cross-browser, para simplificar os scripts no lado do cliente.

O que é jQuery?

- Permite realizar várias tarefas comuns com pouquíssimas linhas de código.
- Além disso, contém recursos para:
 - Manipulação HTML / DOM
 - Manipulação de CSS
 - Métodos de eventos HTML
 - Efeitos e animações
 - AJAX
 - Etc.

Por que usar jQuery?

- Uma 'ótima' vantagem do jQuery é que basicamente todos os seus métodos deverão funcionar em praticamente todos os browsers.
- Além disso foram implementados buscando o máximo de desempenho e melhor experiência do usuário.
- E, possivelmente, a maior vantagem é simplificação de códigos.
 - Faça com jQuery em muito menos linhas que em JavaScript puro.

jQuery

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

// Se I.E.
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

- Agora o mesmo “código em jQuery”.

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

jQuery

“Escreva menos
e faça mais”

Obtendo o jQuery

- Você pode obter o jQuery diretamente do site oficial jquery.com

OU

- Utilizar os denominados CDNs

Obtendo o jQuery

- Baixe o arquivo js do site: <http://jquery.com/download/>

```
<head>
<script src="jquery-3.3.1.min.js"></script>
</head>
```

Obtendo o jQuery

- Usando CDN do Google
- Adicione a tag script no head de sua página, com o valor da propriedade src igual a:

<https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js>

OU

<https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.js>

```
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js">
</script>
</head>
```

Sintaxe jQuery?

- A sintaxe do jQuery é muito semelhante ao de seletores do CSS ou JavaScript, associada adição de uma ação.
- A sintaxe basicamente é:
 $\$(\text{seletor}).\text{acao}()$
- O **\$** (cifrão) é uma função do jQuery para selecionar um elemento.
- **seletor** é a "query" usada para encontrar os elementos.
- A **acao()** é o que deverá ser realizado sobre o elemento localizado.

jQuery

Função de seleção do jQuery

```
//jQuery  
$(document).ready(
```

```
)
```

jQuery

Função de seleção do jQuery

Elemento a ser buscado

```
//jQuery  
$(document).ready(  
    )
```

jQuery

Função de seleção do jQuery

Elemento a ser buscado

Ação que será
configurada

```
//jQuery  
$(document).ready(
```

)

jQuery

Função de seleção do jQuery

Elemento a ser buscado

Ação que será
configurada

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

Conteúdo da ação configurada!

jQuery

Função de seleção do jQuery

Elemento a ser buscado

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

Ação que será
configurada

jQuery

Função de seleção do jQuery

Elemento a ser buscado

Ação que será
configurada

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

jQuery

Função de seleção do jQuery

Elemento a ser buscado

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

Ação que será
configurada

jQuery

Função de seleção do jQuery

Elemento a ser buscado

Ação que será
configurada

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

Conteúdo da ação configurada!

jQuery

Função de seleção do jQuery

Elemento a ser buscado

Ação que será
configurada

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

Exemplos de código

```
$(document).ready(  
    function() {  
        $("#bloco").click(  
            function() {  
                $(this).fadeOut();  
            }  
        );  
    }  
)
```

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

Seletores jQuery

- Os seletores do jQuery permitem selecionar praticamente qualquer elemento em uma página.
- Os 3 tipos de seleções básicas são:
 - Por id;
 - Pela classe;
 - Pelo nome do elemento.

Seletores jQuery

- Seleção do elemento por ID:
 - `$("#cadastros")`
- Seleção de elementos **s** pelo nome da classe:
 - `(".cadastros")`
- Seleção de elementos **s** pelo nome da tag:
 - `("div")`
 - `("p")`
 - `("h1")`

Seletores jQuery

- Há outras formas de seleção mais “elegantes” :D

Código	Resultado
<code>\$("*")</code>	Seleção de todos os elementos da página
<code>\$(this)</code>	Seleciona o elemento “atual”.
<code>\$("p.titulo")</code>	Seleciona todos os <code><p></code> com class=“titulo”
<code>\$("p:first")</code>	Seleciona o primeiro parágrafo
<code>\$("ul li:first")</code>	Seleciona o primeiro <code></code> de uma primeira lista <code></code>
<code>\$("ul li:first-child")</code>	Seleciona o primeiro <code></code> de cada lista <code></code>

Seletores jQuery

Código	Resultado
<code>\$(".[href]")</code>	Seleciona todos os elementos com o atributo href
<code>\$("a[target='_blank']")</code>	Seleciona todos os elementos <a> com o atributo target=''_blank'
<code>\$("a[target!= '_blank']")</code>	Seleciona todos os elementos <a> com o atributo target com valor diferente de _blank
<code>\$(".:button")</code>	Seleciona todos os elementos <button> e <input type="button">
<code> \$("tr:even")</code>	Seleciona todos os elementos <tr> em linhas pares
<code> \$("tr:odd")</code>	Seleciona todos os elementos <tr> em linhas ímpares

CSS com jQuery

- Para alterar as propriedades de estilo com jQuery, você deve utilizar a função **css()**
 `$(“seletor”).css(“propriedade”, “valorDaPropriedade”)`
- Exemplo:
`$("#bloco").css("background-color", "gray");`

CSS com jQuery

- Código de exemplo completo:

```
$(document).ready(  
    function() {  
        $("#bloco").click(  
            function() {  
                $(this).css("background-color", "lime");  
            }  
        );  
    }  
)
```

CSS com jQuery

```
$(document).ready(function(){
    $("p").css({
        "background-color": "green",
        "color": "white",
        "font-size": "25px",
        "padding": "15px"
    });
});
```

Alterando conteúdo HTML com jQuery

- Para alterar conteúdos de elementos com jQuery, você deve utilizar a função **html()**

```
$(“seletor”).html(“Novo texto”)
```

- Exemplo 1:

```
$(“p#opa”).html(“Eu sou o pato Donald!”);
```

- Exemplo 2:

```
$(“#bloco”).html(“Também sou o pato Donald!”);
```

- Exemplo 3:

```
$(“p.inicial”).html(“Não importa”);
```

Alterando conteúdo HTML com jQuery

- Código de exemplo completo:

```
$(document).ready(  
    function() {  
        $("#opa").click(  
            function() {  
                $(this).html("O pato Donald sou eu! \o/");  
            }  
        );  
    }  
)
```

Eventos

- O jQuery possui funções que podem ser utilizados para manipular suas páginas.
- A essas funções podemos associar efeitos do próprio jQuery, como o hide e fadeOut.

Eventos

- Algumas das funções disponíveis no jQuery são:

Evento	Descrição
.click()	Executa quando o elemento recebe um clique simples.
.dblclick()	Executa quando o elemento recebe um <u>duplo</u> clique.
.unload()	Executa quando a janela do site for “descarregada”.
.mouseenter()	Executa quando o mouse estiver sobre o elemento.
.hover()	Executa quando o mouse estiver sobre o elemento e quando for retirado de cima do elemento.
.submit()	Executa quando o formulário recebe o comando de submit.



Eventos

- A sintaxe para definir uma ação a partir de um evento de um elemento é:

`$(elemento).evento()`

- Veja o exemplo:

```
$(document).ready(function(){
    $("#divTitulo").click(function(){
        //Ação para executar no clique.
    });
});
```

Eventos

- De forma geral, as funções de evento do jQuery funcionam da mesma maneira.
- Consulte a documentação para todos os eventos em:

<https://api.jquery.com/category/events/>

Manipulação

- Além dos eventos, o jQuery possui uma categoria de funções voltadas para manipulação do DOM.
 - Já vimos anteriormente a função `html()`.
- Duas funções bastante utilizadas dessa categoria são `.addClass()` e `.removeClass()`.

Manipulação

- As funções de `.addClass()` e `.removeClass()` são responsáveis pela adição ou remoção de uma **class** de CSS ao elemento selecionado.
- Vale ressaltar que ao adicionar uma classe ao elemento, a função **não irá remover classes já atribuídas** anteriormente a esse elemento.
- Veja o exemplo:

Manipulação

```
$(document).ready(function(){
    $("p").click(function(){
        $(this).addClass("formatacaoBack");
    });
});

$(document).ready(function(){
    $("p").dblclick(function(){
        $(this).removeClass("formatacaoBack");
    });
});
```

Manipulação

- Uma lista com todos os métodos do jQuery podem ser encontrados na documentação, em:
<https://api.jquery.com/category/manipulation/>

GET

Get Content - `text()`, `html()`, and `val()`

`text()` – Retorna ou atribui o text dos elementos selecionados

`html()` – Retorna ou atribui o conteúdo do elemento selecionado (incluindo marcações HTML)

`val()` – Retorna ou atribui o valor/value do elemento selecionado

```
<script>
$(document).ready(function(){
    $("#btn1").click(function(){
        alert("Text: " + $("#test").text());
    });
    $("#btn2").click(function(){
        alert("HTML: " + $("#test").html());
    });
});
</script>
```

SET

```
<script>
$(document).ready(function(){
    $("#btn1").click(function(){
        $("#test1").text("Hello world!");
    });
    $("#btn2").click(function(){
        $("#test2").html("<b>Hello world!</b>");
    });
    $("#btn3").click(function(){
        $("#test3").val("Mickey Mouse");
    });
});
</script>
```

Callbacks

- *Callback* é a “função que se executa depois de outra”.
- Ou seja, é a chamada da próxima função quando a função atual for concluída completamente.
- A sintaxe básica para usar callback é:

`minhaFuncao ("parametros da minhaFuncao", functionCallback);`

Callbacks

```
$( "button" ).click(function(){
    $("p").hide("slow", function(){
        alert("The paragraph is now hidden");
    });
});
```

Exemplos

Adicionando novos elementos:

https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_html_append

Manipulando CSS

https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_dom_addclass

Exercício 1

- Faça uma página com conteúdo igual ao código seguinte e então...

```
<p>Clique-me</p>
<div id="bloco"></div>
<p id="opa">Opa</p>
<a href="http://portal.imd.ufrn.br">Saindo...</a>
```

Exercício 1

- Escreva funções jQuery e JavaScript em um arquivo meusScripts.js para que:
 1. Ao clicar no bloco primeiro parágrafo, ele mude para cor verde a **div** de **id=“bloco”**.
 2. Ao clicar no bloco, ele “desapareça lentamente em 10 segundos”.
 3. Ao clicar em “Opa” o bloco retorne na **cor laranja**.
 4. Ao clicar em “Saindo...” a página exiba um alerta perguntando: “Vai para onde?”

Exercício 2

- Baixe o arquivo elementos.html no sigaa e acrescente funções para fazer os botões funcionarem.

Dúvidas?

