

Aula 9: Fundamentos do JQuery

Fonte: Plano de Aula Oficial da Disciplina

Objetivo: Capacitar o aluno para o trabalho com o framework JQuery.

INTRODUÇÃO

JQuery é uma biblioteca (ou framework) "projetado para mudar sua maneira de escrever javascript"; É uma biblioteca JavaScript rápida e concisa que simplifica manipulação de eventos, animação e interações nos documentos HTML.

Foi lançada em Janeiro de 2006 por John Resig e é utilizada por mais de 31% dos 10.000 sites mais visitados, jQuery é atualmente a biblioteca JavaScript mais popular do mundo.

1. A BIBLIOTECA JQUERY

A biblioteca jQuery oferece uma camada de abstração de uso geral para scripts web, e por isso é útil em quase todas as situações que se utilizam scripts. Sua natureza extensível significa que plugins estão sendo constantemente desenvolvidos para acrescentar novas funcionalidades.

Características

- Cross browser;
- Controle total sobre o DOM;
- Manipulação de eventos javascript;
- Manipulação de regras CSS;
- Aplicação de efeitos visuais;
- Uso de AJAX;

2. O QUE O JQUERY OFERECE?

2.1. Modificar a aparência de uma página web.

A CSS oferece um método poderoso de influenciar a forma como o documento é processado, mas fica prejudicado quando os navegadores web não dão todo o suporte para utilização de seus recursos. Com jQuery, os desenvolvedores não tem essa preocupação, pois fica a encargo do jQuery a equivalencia de padrões em todos os navegadores. Além disso, jQuery pode alterar as classes ou propriedades individuais de estilo aplicado para uma parte do documento, mesmo após a página ter sido carregada.

2.2. Alterar o conteúdo de um documento.

jQuery não se limita a simples mudanças na apresentação de um documento, jQuery pode modificar o conteúdo de um documento. O texto pode ser alterado, as imagens podem ser inseridos ou trocados, as listas podem ser reordenadas, ou toda a estrutura do HTML pode ser reescrita e ampliada, tudo isso com uma única API (Application Programming Interface).

2.3. Responder a interação do usuário.

Mesmo os mais elaborados e poderosos comportamentos não são úteis se não podemos controlar quando ocorrem. A biblioteca jQuery oferece uma maneira elegante para interceptar uma grande variedade de eventos, tais como clique em um link, sem necessidade de sobrecarregar o código HTML com manipuladores de eventos. Ao mesmo tempo, a API de manipulação de eventos remove as incoerências do navegador que normalmente afetam os desenvolvedores web.

2.4. Animações

Para implementar de forma eficaz esses comportamentos interativos, um designer também deve fornecer feedback visual para o usuário. A biblioteca jQuery facilita isso fornecendo uma série de efeitos.

2.5. Simplificar tarefas comuns de JavaScript

Além de todas essas características a biblioteca jQuery oferece aprimoramentos para construções básicas em JavaScript, como interação e manipulação de array.

3. USANDO O JQUERY

3.1. Download jQuery

O site da jQuery Oficial (<http://jquery.com/>) tem sempre a versão mais atualizada, além disso, possui recursos como códigos e notícias relacionadas com a biblioteca.

Para começar, precisamos de uma cópia da jQuery, que pode ser baixada na página inicial do site. Diversos versões da jQuery podem estar disponível a qualquer momento, a mais adequado para nós, como desenvolvedores de site vai ser a última versão descompactada da biblioteca. Nenhuma instalação é necessária. Para usar o jQuery, só precisamos colocá-lo em nosso site. Desde que o JavaScript é uma linguagem interpretada, não há compilação ou fase de construção para se preocupar. Sempre que precisamos de uma página para ter jQuery disponível, vamos simplesmente referir-se a localização do arquivo em nosso documento HTML.

A referência a biblioteca jQuery pode ser incluída em uma página web usando a seguinte marcação:

```
<script type="text/javascript" src="jquery.js"> </script>
```

Onde src é o endereço do arquivo **jquery.js**.

O jQuery também pode ser carregado de outras formas, como por meio do Google AJAX Libraries API , com a seguinte marcação:

```
<script type="text/javascript" src="http://www.google.com/jsapi"> </script>
```

ou

```
<script>google.load(jQuery " , "1.4.2"); </script>
```

Ou, ainda, por HTTPS e HTTP:

```
<script type="text/javascript" src="http://ajax.googleapis.com/ajax/libs/jquery/1.4.2/jquery.min.js"> </script>
```

A biblioteca pode ser carregada, ainda, usando o Microsoft hosts. O jQuery encontra-se em sua AJAX CDN (Content rede de distribuição), tornando mais fácil adicionar o suporte para a biblioteca jQuery. CDN serve bibliotecas JavaScript de um dos milhares servidores da Microsoft ao redor do mundo.

```
<script src="http://ajax.microsoft.com/ajax/jquery/jquery-1.4.2.min.js" type="text/javascript">
</script>
```

3.2. A Função de Busca \$()

Não importa qual o tipo de seleção queremos utilizar em jQuery, sempre começamos com o cifrão e parênteses: \$ (). Qualquer coisa que possa ser usada em um estilo também pode ser envolvida em aspas e colocados dentro dos parênteses, permitindo aplicar métodos jQuery para o conjunto de elementos combinados.

3.2.1. Método ready()

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. Para fazer isso, nós registramos o evento ready (pronto) para o documento.

```
$(document).ready(init);
function init() {
    // faça alguma coisa quando o DOM estiver pronto
}
```

Esta notação é conhecida como **sintaxe formal** e também pode ser indicada de maneira "anônima":

```
$(document).ready( function() {
    // faça alguma coisa quando o DOM estiver pronto
});
```

O método jQuery ready() oferece duas vantagens sobre seu equivalente em JavaScript:

- O script está pronto para funcionar tão logo a árvore do documento tenha sido carregada. Não é necessário que todos os componentes da página, tais como imagens, folhas de estilo, animações e mídias em geral, tenham sido carregadas. Basta que a estrutura de marcação da página esteja disponível e o script já poderá funcionar.
- Não há variações funcionais para o método e pelo fato de jQuery não admitir mistura de script com marcação, utiliza-se a sintaxe mostrada para servir de container aos scripts.

3.2.2. Seletores CSS

A biblioteca jQuery suporta quase todos os selecionadores incluídos nas especificações CSS 1 a 3, conforme descrito no site da World Wide Web Consortium: <http://www.w3.org/Style/CSS/#>. Este suporte permite que os desenvolvedores melhorem seus sites sem se preocupar com que os navegadores (especialmente Internet Explorer 6) podem compreender.

Para adicionar estilos em uma seleção jQuery podemos usar a seguinte sintaxe:

```
seletorjQuery.css('propriedade','valor')
```

3.2.3. Seleção de elementos

Existem três formas de selecionar elementos com jQuery:

- Nome da <tag>
- ID
- Class

Eles podem ser utilizados tanto sozinhas ou em combinação com outros selecionadores.

Sintaxe:

Seletor	Css	JQuery	Descrição
elemento	p	\$('p')	Seleciona todos os paragrafos no documento
ID	#meuID	\$('#meuID')	Seleciona o único elemento no documento lque possui o id "meuID"
class	.minhaClass	\$('.minhaClass')	Seleciona todos os elementos no documento que tem a classe "minhaClass"

\$('#id')

O seletor id acessa o elemento cujo valor do atributo id tenha sido especificado no argumento. Exemplo:

```
$(document).ready(function(){  
    $('#button').click(function(){  
        $('#diferente').css('background','red');  
    });  
});
```

\$('.class')

O seletor id acessa o elemento cujo valor da atributo class tenha sido especificado no argumento. Exemplo:

```
$(document).ready(function(){  
    $('#button').click(function(){  
        $('.verde').css('background','green');  
    });  
});
```

\$('#elemento')

O seletor de elementos acessa todos os elementos especificados no argumento.

Exemplo:

```
$(document).ready(function){  
    $('#button').click(function){  
        $('#div').css('background','yellow');  
    };  
};
```

4. SINTAXE AVANÇADA

4.1. Encadeamento

Denomina-se encadeamento a característica de se poder encadear diversos métodos em uma declaração. Isso é possível porque se criou o jQuery de modo a que cada método em uma declaração retorne um objeto pronto para receber outro método, que, por sua vez, retornará outro objeto, e assim por diante, permitindo ao desenvolvedor construir uma cadeia de objetos e métodos.

Exemplo:

```
$('#div').children('p').fadeOut().addClass('xpto')
```

4.2. Grupamento de Seletores

O grupamento de seletores permite acessar um grupo de seletores. O argumento é a lista de seletores que se deseja acessar separados por vírgula. Sintaxe:

```
$(seletor1, seletor2, seletor3, seletorN)
```

Exemplo:

```
$(document).ready(function){  
    $('#button').click(function){  
        $('#p, #diferente, .verde, li').css('background','gray');  
    };  
};
```

4.3. Seletores compostos

Os seletores compostos são aqueles constituídos pela combinação de dois ou mais seletores simples. A combinação entre seletores simples para criar um seletor composto segue uma sintaxe própria e é feita com o emprego de três sinais de combinação.

Sinal de combinação	Exemplo	Nota
Espaço em branco	div p em	Obrigatório usar um ou mais espaços em branco entre os seletores
Sinal maior que ">"	div > p ou div>p	Espaço facultativo entre os seletores
Sinal de adição "+"	p + a ou p+a	Espaço facultativo entre os seletores

5. FILTROS

Denominam-se seletores filtros os seletores do tipo pseudosseletores que se destinam a filtrar uma condição particular de um seletor simples ou composto. Os filtros simplesmente filtram um seletor, para que ele atinja somente determinados elementos.

`$(elemento:first')`

Seleciona a primeiro elemento do documento

`$(elemento:last')`

Seleciona a último elemento do documento

`$(elemento:even')`

Seleciona os elementos com índice par do documento.

`$(elemento:odd')`

Seleciona os elementos com índice ímpar do documento.

`$(elemento:eq(4)')`

Seleciona o elemento com índice 4 do documento

`$(elemento:contains(texto)')`

Esse seletor é exclusivo da biblioteca jQuery. Acessa o texto definido no parâmetro. No exemplo seleciona todos os elementos que tem a string "texto" em seu conteúdo

\$('elemento:hidden')

Seleciona todos os elementos ocultos do documento

 \$(':visible')

Seleciona tudo que está visível no documento

 \$('input:empty')

Seleciona todos os inputs vazios do documento

 \$('elemento:has(p)')

Seleciona todas as elementos que contenham parágrafos em seu conteúdo

5.1. Seletores de atributo

São os seletores que utilizam de atributos (href, id, class, etc.) para encontrar seus alvos.

 \$('elemento[id]')

Seleciona todos os elementos que tenham um atributo "id"

 \$('elemento[id = "valor"]')

Seleciona todos os elementos com id = "valor"

 \$('elemento[id != "valor"]')

Seleciona todos os elementos com id diferente de "valor". Você também pode acumular atributos, assim:

```
 $('elementos[id = "valor"][href = "link"]')
```

5.2. Seletores para formulários

São seletores criados especialmente para uso em formulários.

 \$(':input')

Seleciona todos os inputs input, textarea, button e select do documento

 \$(':text')

Seleciona todos os inputs com type = 'text' do documento

 \$(':password')

Seleciona todos os inputs type = 'password' do documento

\$(':checkbox')

Seleciona todos os checkboxes; o mesmo serve para inputs do tipo radio, button, submit, reset, image e file.

\$(':enabled')

Seleciona todos os elementos do formulário que estejam habilitados

\$(':disabled')

Seleciona todos os elementos do formulário que estejam desabilitados

\$(':checked')

Seleciona os elementos selecionados em campos do tipo radio e checkbox

\$(':selected')

Seleciona os elementos selecionados nos selects do formulário

6. BIBLIOGRAFIA

Plano de Aula Oficial da Disciplina "CCT0081 - Programação para Internet Rica",
Centro Universitário Estácio-Radial.