

---

# Sistemas Web III

jQuery: Introdução  
"write less, do more"

**Tiago Cruz de França**

[tcruz.franca@gmail.com](mailto:tcruz.franca@gmail.com)

# jQuery, O que é?

## @ jQuery é uma biblioteca javascript

- @ Simples de programar e de se aprender
- @ O objetivo é tornar mais simples o uso de javascript nas páginas web

## @ Requisitos para aprender, usar e entender:

- @ Conhecer:
  - ▶ HTML
  - ▶ CSS
  - ▶ javascript

# jQuery, O que é?

## No cabeçalho inclua:

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>  
<script>
```

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

## No HTML inclua:

```
<p>If you click on me, I will disappear.</p>  
<p>Click me away!</p>  
<p>Click me too!</p>
```



Teste você mesmo

# Benefícios ao usar jQuery

## @ **Escreva menos linhas de código**

- ⌚ Tarefas complicadas e repetitivas resolvidas com poucas linhas quando se usa a biblioteca (wrap code)
  - ▶ Menos javascript para desenvolver, maior produtividade, reuso de funcionalidades testadas
- ⌚ Simplifica tarefas realizadas com javascript
  - ▶ Manipulação do DOM e AJAX

# Features jQuery

**Manipulação HTML/DOM, CSS, métodos de eventos  
HTML, efeitos e animações, AJAX, além de outras  
utilidades**

**Selecione e realize ações sobre elementos HTML  
usando jQuery**

# Só existe jQuery?

## @ Não, mas...

@ É uma biblioteca bastante utilizada e grandes companhias a usam

▶ Ex: Google, IBM, Netflix, etc.

@ jQuery funciona até no IE6!

▶ Se não funcionar em algum navegador, mude de navegador, provavelmente o seu navegador é que não presta

▶ Ou seja, vai funcionar bem nos navegadores mais comuns

# Adicionando jQuery página

## 1. Baixe uma biblioteca jQuery em [jQuery.com](https://jquery.com)

@ Duas versões disponíveis

- ▶ Versão de produção: esta é a versão que você deve usar na página, ela tem só o necessário
- ▶ Versão de desenvolvimento: para desenvolvimento e teste (código legível e sem compressão)

@ A biblioteca tem um arquivo e deve ser incluída na página usando a tag `<script>` (geralmente no head da página)

```
<head>  
<script src="jquery-1.12.0.min.js"></script>  
</head>
```

# Adicionando jQuery página

## 1. jQuery usando Content Delivery Network (CDN)

@ Não precisar fazer download da biblioteca

@ Google CDN

```
<head>  
<script  
src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>  
</head>
```

@ Microsoft CDN

```
<head>  
<script src="http://ajax.aspnetcdn.com/ajax/jQuery/jquery-1.12.0.min.js"></script>  
</head>
```



# Sintaxe

## @ Sintaxe Básica: \$(selector).action()

- @ Um “\$” define o acesso jQuery
- @ Um seletor de elemento HTML
  - ▶ Consulta ou encontra o elemento
- @ Uma ação jQuery que será realizada sobre o elemento
- @ Exemplos:
  - ▶ \$(this).hide() – esconde o elemento
  - ▶ \$("p").hide() – esconde todos elementos <p>
  - ▶ \$(".test").hide() – esconde elementos com class="test".
  - ▶ \$("#test").hide() – esconde elementos com id="test".

# Evento **Ready** do document

```
$(document).ready(function(){  
    // jQuery methods go here...  
});
```

- Ⓢ **Toda operação dos exemplos mostrados até aqui estão dentro do evento **ready****
  - Ⓢ Por quê? Isso previne que o código jQuery seja executado antes do documento ser aberto
    - ▶ Nos exemplos com `hide()`, os campos escondidos podem nem aparecer, antes que isso aconteça eles já se tornam ocultos
  - Ⓢ O outro motivo é que o código javascript já estará disponível quando a página for inteiramente carregada

# Evento **Ready** do document<sup>2</sup>

- ⌚ **Exemplos do que pode acontecer se o documento for executado antes da página ser aberta**
  - ⌚ Esconder um elemento que ainda não foi nem criado
  - ⌚ Tentar pegar o tamanho de uma imagem que ainda não foi carregada

# Evento **Ready** do document<sup>3</sup>

## @ Incluindo tarefa a ser executada

@ Use o método conforme sintaxe abaixo

▶ Ela pode ser usada opcionalmente no lugar do Ready

```
$(function(){  
    // jQuery methods go here...  
});
```

# Seletores jQuery

## 🕒 Os seletores são uma das partes mais importantes da biblioteca jQuery

- 🕒 O seletor começa com o sinal “\$” e parenteses. Ex: \$()
- 🕒 Eles permitem que o elemento jQuery seja selecionado e manipulado
  - ▶ Os elementos podem ser selecionados (ou encontrados) usando:
    - Nome, id, classe, tipo, atributos, valores de atributos, etc.
    - Ele se baseia nos seletores usados no CSS e usa alguns outros próprios
  - ▶ Lembra do DOM (web-1), um bom entendimento sobre o DOM ajuda a entender como é possível encontrar e manipular os elementos HTML

# Seletor de elemento

## @ Selecionando baseado no nome

@ Foi que fizemos com o elemento <p>: \$("p")

## @ Selec. por id: \$("#test")

@ Usando um botão com “id=test”

```
$(document).ready(function(){  
    $("button").click(function(){    $("#test").hide();    });  
});
```

## @ Selecionando pela classe: \$(".test")

@ Usando um botão com o “class=test”

```
$(document).ready(function(){  
    $("button").click(function(){ $(".test").hide(); });  
});
```

Para ver mais seletores...

@ <http://www.w3schools.com/jquery/tryssel.asp>

@ [\*\*http://www.w3schools.com/jquery/jquery\\_ref\\_selectors.asp\*\*](http://www.w3schools.com/jquery/jquery_ref_selectors.asp)

# Usando arquivos com funções jQuery

## Ⓢ Permite o uso da função jQuery definida em várias páginas

🌀 Ideia semelhante a do uso de javascript externo

```
<head>  
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js">  
</script>  
<script src="my_jquery_functions.js"></script>  
</head>
```

