

Introduction To jQuery



What You Should Already Know

- ▶ Before you start studying **jQuery**, you should have a basic knowledge of:
- ▶ HTML
- ▶ CSS
- ▶ JavaScript

What is jQuery?

- ▶ Open source JavaScript library/framework
- ▶ Created by John Resig in Jan. 2006
- ▶ Cross-browser compatible
 - IE 6+, Firefox 2+, Safari 2.0+, Opera 9+
- ▶ Fully supported by Microsoft .
- ▶ Supports AJAX technology
- ▶ jQuery is a lightweight, "write less, do more"

What is jQuery?

- ▶ Built with plug-in architecture
- ▶ Easy to learn / use
- ▶ Chaining methods
- ▶ Unobtrusive Javascript
- ▶ CSS v1–3 compatible
- ▶ Designed to change the way you write Javascript

How Does It work?

- ▶ jQuery() function
 - Everything starts with a call to jQuery()
 - Since it's called so often, the **\$** variable is set up as an alias to jQuery().
- ▶ Basic structure:

FindSomething.DoSomething

\$(SomethingToFind).DoSomething();

jQuery Install

- Adding jQuery to Your Web Pages
- There are several ways to start using jQuery on your web site. You can:
 - Download the jQuery library from **jQuery.com**
 - Include jQuery from a CDN, like Google
- **Downloading jQuery**

```
<head>  
<script src="jquery-1.11.1.min.js"></script>  
</head>
```

- **jQuery CDN** (Content Delivery Network)

Google CDN:

```
<head>  
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js"></script>  
  
</head>
```

jQuery Selectors

- ▶ **jQuery selectors** allow you to select and manipulate HTML element(s).
- ▶ jQuery selectors are used to "find" (or select) HTML elements based on their name, id, classes, types, attributes, values of attributes and much more. It's based on the existing CSS Selectors, and in addition, it has some own custom selectors.
- ▶ All selectors in jQuery start with the dollar sign and parentheses: \$().
- ▶ syntax:
 - \$(css selector)

The Ready Function

- ▶ Set up a basic HTML page
- ▶ add jQuery library
- ▶ Create a “ready” function
- ▶ Call a function
- ▶ 2 ways to specify the ready function
 - `$(function(){...});`
 - `$(document).ready(function(){...});`

jQuery – css() Method

- ▶ The **css()** method sets or returns one or more style properties for the selected elements.
- ▶ **Set a CSS Property**
- ▶ To set a specified CSS property, use the following syntax:
 - ▶ `css("propertyname", "value");`
- ▶ **Set Multiple CSS Properties**
- ▶ To set multiple CSS properties, use the following syntax:
 - ▶ `css({"propertyname": "value", "propertyname": "value", ...});`

jQuery – css() Method(2)

- ▶ **Return a CSS Property**
- ▶ To return the value of a specified CSS property, use the following syntax:
- ▶ `css(propertyname);`
- ▶ The following example will return the **background-color** value of the **FIRST** matched element:
- ▶ `$("p").css("background-color");`

Simple Selector Examples

- ▶ **`$ ("img")`**
 - Selects all images
- ▶ **`$ ("p a")`**
 - Selects all links nested under a paragraph
- ▶ **`$ ("div.myClass")`**
 - Selects all divs with a class of "myClass"
- ▶ **`$ ("#myElement")`**
 - Selects element with an id of "myElement"
- ▶ **`$ ("#nav li.current a")`**
 - Selects all links under any list item with class="current" that exist under the element that have "nav" id

Custom Selectors

<code>:checkbox</code>	Selects checkboxes
<code>:checked</code>	Selects checkboxes or radio buttons that are checked
<code>:contains(foo)</code>	Selects elements containing the text “foo”
<code>:disabled</code>	Selects disabled form elements
<code>:header</code>	Selects elements that are headers (<h1>, etc)
<code>:input</code>	Selects form elements (input, select, textarea, button)
<code>:not</code>	Negates the specified filter
<code>:selected</code>	Selects option elements that are selected
<code>:submit</code>	Selects submit buttons

These are just a few of the 20+ available

Using Custom Selectors

- ▶ **:checkbox:checked:enabled**
 - Selects checkboxes that are enabled and checked
- ▶ **input:not(:checkbox)**
 - Selects non-checkbox <input> elements
- ▶ **div p:not(:hidden)**
 - Selects all visible <p> elements nested under a div
 - **:checkbox:checked**
 - `$(":checkbox:checked").wrap("<span style='background-color:red' />");`
 - **:header:not(h5)**
 - `$(':header:not(h5)').css("background-color","#3F0");`

Selecting By Position

<code>:first</code>	<code>:last-of-type</code>	<code>:nth-child(Xn+Y)</code>
<code>:last</code>	<code>:only-child</code>	<code>:even</code> and <code>:odd</code>
<code>:first-child</code>	<code>:only-of-type</code>	<code>:eq(n)</code>
<code>:last-child</code>	<code>:nth-child(n)</code>	<code>:gt(n)</code>
<code>:first-of-type</code>	<code>:nth-child(even odd)</code>	<code>:lt(n)</code>

- `p:odd`
 - Returns every odd paragraph element
- `li:last-child`
 - Returns last item of each list (last child of parent element)
- `td:eq(2)`
 - Returns third table cell
- Selectors start counting from 0 (except `:nth-child`, which starts from 1 for CSS compatibility)

More Selector Examples

<code>\$ ("li:last")</code>	Selects last list item
<code>\$ ("tr:nth-child(1)")</code>	Selects the first row of each table
<code>\$ ("tr:nth-child(even)")</code>	Selects even numbered table rows
<code>\$ ("body > div:has(a)")</code>	Selects direct <div> children of <body> containing links
<code>\$ ("img[alt]")</code>	Selects images with ALT attribute
<code>\$ ("a[href*=jquery.com]")</code>	Matches all <a> elements that reference the jQuery site
<code>\$ ("a[href\$=pdf]")</code>	Selects links to PDF files
<code>\$ ("a[href^=https://]")</code>	Selects all links starting with "https://"

Selectors Summary

- ▶ Can leverage your knowledge of CSS selectors to get up and running fast
- ▶ Learn about the more advanced selectors jQuery supports:
 - <http://docs.jquery.com/selectors>

Selectors Demo



jQuery – Set Content and Attributes

- ▶ **Set Content – text(), html(), and val(), attr()**
- ▶ **text()** – Sets the text content of selected elements
- ▶ **html()** – Sets the content of selected elements (including HTML markup)
- ▶ **val()** – Sets the value of form fields
- ▶ **attr()** – method is also used to set/change attribute values.

Set Content - text(), html(), and val()

▶ Set text:

- `$ ("span#msg") .text ( 'The thing was updated' ) ;`

▶ Set HTML:

- `$ ("div.intro") .html ("<h1>Look, HTML</h1>") ;`

▶ Set val (of first matched element):

- `$ ("#text1") .val ("text") ;`

jQuery – get Content and Attributes

- ▶ We will use the same three methods from the previous page to **set content** but do not set the value or single or double quotes of the selectors:
- ▶ **text()** – returns the text content of selected elements
- ▶ **html()** – returns the content of selected elements (including HTML markup)
- ▶ **val()** – returns the value of form fields
- ▶ **attr()** – method is also used to get attribute values.

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

► Get text:

- `alert("Text: " + $("span#msg").text());`
- Get HTML (of first matched element):
- `Var str=$("div.intro").html();`

► Get val:

- `alert("Value: " + $("#test").val());`
- Get Attribute :
- `alert($("#w3s").attr("href"));`

Attributes

- ▶ Get Attributes (for first matching element):
 - `var href = $("a.nav").attr("href");`
- ▶ Set Attributes:
 - `$("a.nav").attr("href", "http://flickr.com");`
 - Set multiple Attributes:
 - `$("a.nav").attr({
 "href": "http://westlaw.com",
 "id": "westlaw" });`
- ▶ Remove attribute:
 - `$("#intro").removeAttr("id");`

jQuery – Get and Set CSS Classes

- ▶ **jQuery Manipulating CSS**
- ▶ jQuery has several methods for CSS manipulation. We will look at the following methods:
- ▶ **addClass()** – Adds one or more classes to the selected elements
- ▶ **removeClass()** – Removes one or more classes from the selected elements
- ▶ **toggleClass()** – Toggles between adding/removing classes from the selected elements

CSS

- ▶ `$("#intro").addClass("highlighted");`
- ▶ `$("#intro").removeClass("highlighted");`
- ▶ `$("#intro").toggleClass("highlighted");`
- ▶ `$("div").hasClass("highlighted");`
 - True if at least 1 element has the class
- Set multiple CSS Attributes :
 - ▶ `$("p").css("font-size", "20px");`
 - ▶ `$("p").css({
 "font-size": "20px",
 "color": "red"
});`

Add New HTML Content

- ▶ We will look at four jQuery methods that are used to add new content:
- **append()** - Inserts content at the **end** of the selected elements
- **prepend()** - Inserts content at the **beginning** of the selected elements
- **after()** - Inserts content **after** the selected elements
- **before()** - Inserts content **before** the selected elements



Add New HTML Content

- ▶ `$("p").append("Some appended text.");`
- ▶ `$("p").prepend("Some prepended text.");`
- ▶ `$("img").after("Some text after");`
- ▶ `$("img").before("Some text before");`

Remove Elements / Content

- ▶ To remove elements and content, there are mainly two jQuery methods:
 - **remove()** - Removes the selected element (and its child elements)
 - ▶ The jQuery **remove()** method removes the selected element(s) and its child elements.
 - **empty()** - Removes the child elements from the selected element

Remove Elements / Content

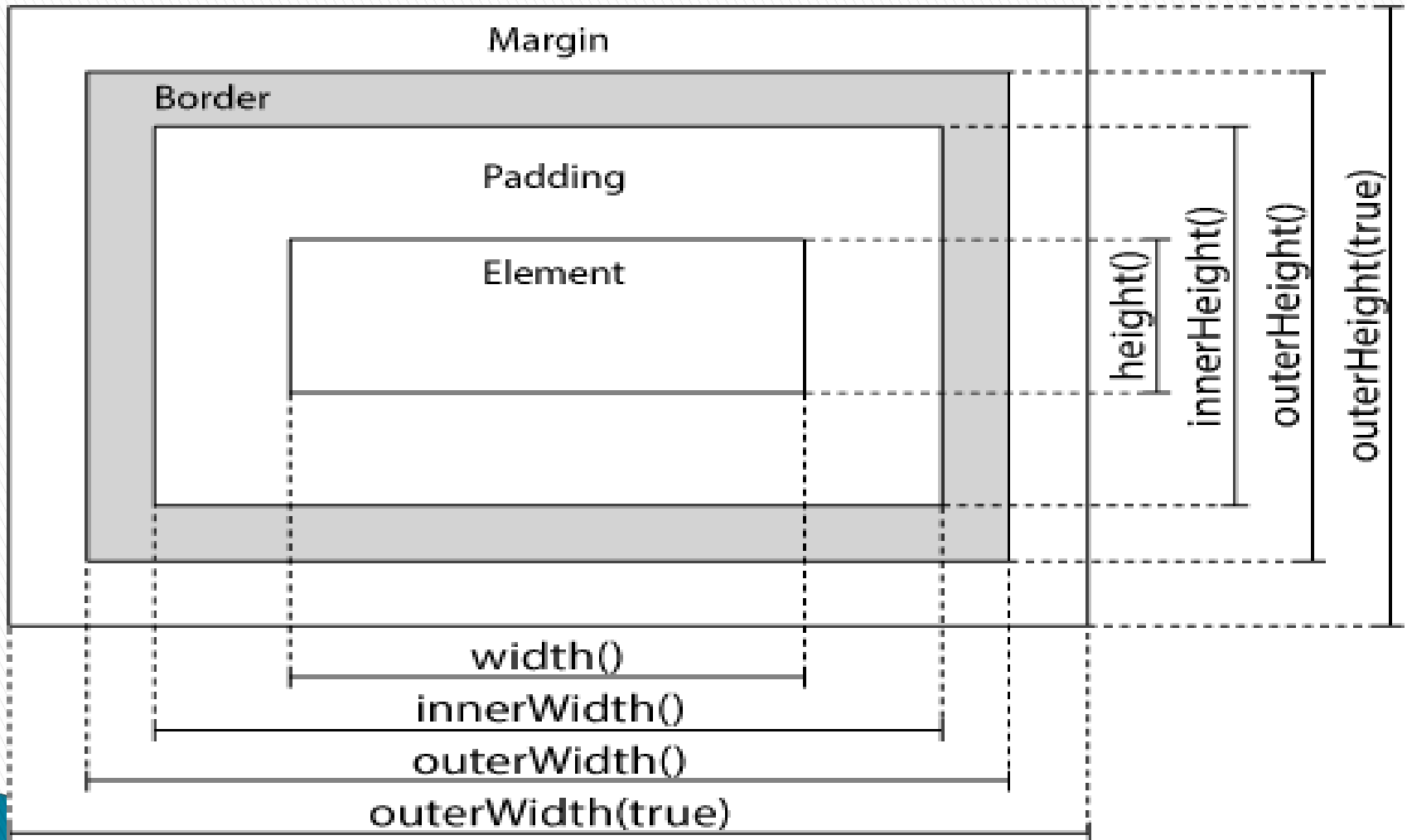
- ▶ `$("#div1").remove();`
 - ▶ `$("p").remove(".test");`
 - ▶ `$("p").remove(".test, .demo");`
-
- ▶ `$("#div1").empty();`

jQuery - Dimensions

jQuery Dimension Methods

- ▶ jQuery has several important methods for working with dimensions:
 - width()
 - height()
 - innerWidth()
 - innerHeight()
 - outerWidth()
 - outerHeight()

jQuery - Dimensions



jQuery - Dimensions

```
▶ $("button").click(function(){  
    var txt = "";  
    txt += "Document width/height: " +  
$(document).width();  
    txt += "x" + $(document).height() + "\n";  
    txt += "Window width/height: " +  
$(window).width();  
    txt += "x" + $(window).height();  
    alert(txt);  
});
```

‘Event’ properties

- ▶ `event.target` ref to element triggering event
- ▶ `Event.target.id` id of element triggering event
- ▶ `event.currentTarget`
- ▶ `event.type` type of event triggered
- ▶ `event.data` second parm in the `bind()` func
- ▶ Various mouse coordinate properties
- ▶ Various keystroke related properties

Commands Demo



Questions?

