



**Wintellect®**  
Know how.

## Introducing jQuery

Jeff Proise  
jeffpro@wintellect.com

**Wintellect NOW** Consulting/Training 


1

## About Wintellect

consulting/training

**who we are**  
Founded by top experts on Microsoft – Jeffrey Richter, Jeff Proise, and John Robbins, along with a team of industry leaders – we pull out all the stops to help our customers achieve their goals through advanced software-based consulting and training solutions.

<p><b>consulting</b></p> <p><i>Wintellect helps you build better software, faster, tackling the tough projects and solving the software and technology questions that help you transform your business.</i></p> <ul style="list-style-type: none"> <li>• Architecture, Analysis and Design</li> <li>• Full lifecycle software development</li> <li>• Debugging and Performance tuning</li> <li>• Database design and development</li> </ul>	<p><b>training</b></p> <p><i>Wintellect's courses are written and taught by some of the most renowned and respected names in the Microsoft programming industry.</i></p> <ul style="list-style-type: none"> <li>• Learn from the best. Access the same training Microsoft's worldwide developers enjoy.</li> <li>• Real world knowledge and solutions on both current and cutting edge technologies</li> <li>• Flexibility in training options – onsite, virtual, on demand</li> </ul>
---	--


**Wintellect NOW** 

2

## jQuery

- Cross-browser JavaScript library
  - Created by John Resig in 2006
  - Compact and CSS3-compliant
  - Free, open-source, and dual-licensed
    - MIT license and GNU general public license v2
  - Abstracts browser differences
  - Do more with less code
- Most popular JavaScript library on the planet


<http://jquery.com/>

**Wintellect NOW** 

3

## jQuery Features


- Simplified DOM navigation
- Simplified DOM and CSS manipulation
- Utility methods and properties
- Enhanced DOM event handling
- Effects and animations
- Simplified AJAX
- Extensibility through plugins

**Wintellect NOW** 

4

## Why jQuery?

- Because you'll write a lot less code
- Because your code will work in a lot more places
  - Let jQuery handle differences between browsers
- Because jQuery brings the power of CSS3 selectors to browsers that don't support CSS3
  - Delegates to browser when CSS3 is supported
  - Does the work itself when it's not
- Because jQuery add-ons such as jQuery Mobile and jQuery UI make jQuery even better

**Wintellect NOW** 


5

## JavaScript without jQuery

```
// Works in IE9+ and most non-IE browsers
var table = document.getElementById("expandotable");

for (var i = 0; i < table.rows.length; i++) {
    table.rows[i].cells[1].addEventListener("click", onClick, false);
}

function onClick(event) {
    var row = event.target;
    var div = row.getElementsByTagName("div")[0];
    var style = window.getComputedStyle(div, null).getPropertyValue("display");
    div.style.display = (style == "none") ? "block" : "none";
}
```

**Wintellect NOW** 

6

## JavaScript without jQuery, Cont.

```
// Works in older versions of IE (e.g., 6, 7, and 8)
var table = document.getElementById("expandotable");

for (var i = 0; i < table.rows.length; i++) {
    table.rows[i].cells[1].attachEvent("onclick", onClick);
}

function onClick() {
    var row = window.event.srcElement;
    var div = row.getElementsByTagName("div")[0];
    var style = div.currentStyle["display"];
    div.style.display = (style == "none") ? "block" : "none";
}
```



7

## Making It Browser-Neutral

```
var table = document.getElementById("expandotable");

for (var i = 0; i < table.rows.length; i++) {
    if (window.addEventListener)
        table.rows[i].cells[1].addEventListener("click", onClick, false);
    else
        table.rows[i].cells[1].attachEvent("onclick", onClick);
}

function onClick(event) {
    if (!event)
        event = window.event;
    var row = (event.target) ? event.target : event.srcElement;
    var div = row.getElementsByTagName("div")[0];
    var style;
    if (window.getComputedStyle)
        style = window.getComputedStyle(div, null).getPropertyValue("display");
    else
        style = div.currentStyle["display"];
    div.style.display = (style == "none") ? "block" : "none";
}
```



8

## JavaScript with jQuery

Select all TD elements representing the second column in the table whose id is "expandotable"

Attach a click handler to each TD element selected

```
$("#expandotable tr td:nth-child(2)").click(function() {
    $(this).find("div:first").toggle();
});
```

Find the first DIV that's a child of the TD that was clicked

Alternately toggle the DIV's visibility on and off



9

## jquery.[min].js

- Script file containing jQuery code
  - Download private copy from URL below
  - Or load from jQuery CDN, Microsoft CDN, or Google CDN so file can be shared through browser cache
- Library forked for optimization
  - Version 1.x supports IE6 and higher (latest is 1.10.2)
  - Version 2.x supports IE9 and higher (latest is 2.0.3)
- Minified version approx. 32K after GZIPping

[http://docs.jquery.com/Downloading\\_jQuery](http://docs.jquery.com/Downloading_jQuery)



10

## Loading jQuery

```
// Load private copy from your own site
<script src="jquery-1.8.3.min.js"></script>

// Load shared copy from jQuery CDN
<script src="http://code.jquery.com/jquery-1.8.3.min.js">
</script>

// Load shared copy from Microsoft CDN
<script src="http://ajax.microsoft.com/ajax/jquery/jquery-1.8.3.min.js">
</script>

// Load shared copy from Google CDN
<script
    src="http://ajax.googleapis.com/ajax/libs/jquery/1.8.3/jquery.min.js">
</script>
```



11

## The .ready() Method

- Invokes callback function ("ready handler") when DOM is loaded but before images are loaded

```
// Formal syntax
$(document).ready(function() {
    // Code to execute when DOM is ready
});
```

```
// Shorthand form
$(function() {
    // Code to execute when DOM is ready
});
```

<http://api.jquery.com/ready/>



12

## Using the .ready() Method

```
$(document).ready(function() {
  // Set background color of alternating rows to light green
  $("table tr:odd").css("background-color", "#ccffcc");
});
```

Lorem ipsum	Nunc imperdiet turpis sed mi accumsan et aliquet tortor volutpat
Maecenas non	Donec odio tortor, varius vel iaculis sed, tincidunt quis neque
Vestibulum ante	Mauris quam erat, mattis non ultricies id, ornare sed felis
Maecenas eleifend	Nunc ante ante, sollicitudin quis consectetur in, sodales et neque
Maecenas et	Sed nibh justo, luctus non scelerisque eu, molestie id magna



13

## DEMO

The .ready() Method



Consulting/Training



14

## The jQuery() Function

- Global function at heart of jQuery library
  - Aliased with \$ symbol - jQuery("div") == \$("div")
- Callable in four ways
  - Input a selector string
  - Input an object reference
  - Input an HTML string
  - Input a callback function
- Also contains utility methods and properties

<http://api.jquery.com/jquery/>



15

## Passing a Selector

```
// Set background color of alternating table rows to light green
$("table tr:odd").css("background-color", "#ccffcc");
```



16

## Passing a DOM Element

```
// In an event handler, pass "this" to jQuery() so jQuery
// methods can be called on the element that fired the event
$("table tr").click(function(e) {
  $(this).css("background-color", "red");
});
```

*"this" refers to TR element that was clicked*



17

## Passing an HTML String

```
// Create an image and append it to the document body
$("<img src='flower.jpg' />").appendTo("body");
```



18

## Passing a Callback Method

```
$(function() {
    // Set background color of alternating rows to light green
    $("table tr:odd").css("background-color", "#ccffcc");
});
```



19

## Selectors

- jQuery() function supports a rich selector syntax
  - Accepts a CSS-style selector string
  - Returns jQuery object wrapping the selected elements
    - Also known as a "wrapped set"
    - jQuery() function != jQuery object
- Supports most CSS1, CSS2, and CSS3 selectors
  - Even in browsers that don't support CSS3
- Adds a few selectors of its own

<http://api.jquery.com/category/selectors/>



20

## Simple Selectors

```
// Select all <div> elements in the document
var results = $("div");

// Select the element whose id is "output"
var results = $("#output");

// Select all elements assigned the CSS class named "shaded"
var results = $(".shaded");

// Select all <div> elements assigned the CSS class "shaded"
var results = $("div.shaded");

// Select all <h1>, <h2>, and <h3> elements
var results = $("h1, h2, h3");
```



21

## Selector Combinations

```
// Select all <a> elements that are descendants of <div>s
var results = $("div a");

// Select all <a> elements that are direct children of <div>s
var results = $("div > a");

// Select all <a> elements that occur immediately after <div>s
var results = $("div + a");

// Select all sibling <a>s that occur immediately after <div>s
var results = $("div ~ a");

// Select all <p>s that occur immediately after <a>s in <div>s
var results = $("div a + p");
```



22

## Attribute Selectors

Selector	Description
[attr]	Selects elements with an <i>attr</i> attribute
[attr=val]	Selects elements with an <i>attr</i> attribute whose value is <i>val</i>
[attr~ =val]	Selects elements with an <i>attr</i> attribute whose value is a space-separated list of words, one of which is <i>val</i>
[attr =val]	Selects elements with an <i>attr</i> attribute whose value is <i>val</i> or begins with <i>val</i> and is immediately followed by a hyphen
[attr^ =val]	Selects elements with an <i>attr</i> attribute whose value begins with <i>val</i>
[attr\$ =val]	Selects elements with an <i>attr</i> attribute whose value ends with <i>val</i>
[attr* =val]	Selects elements with an <i>attr</i> attribute whose value contains <i>val</i>
[attr!=val]	Selects elements with no <i>attr</i> attribute or whose <i>attr</i> attribute is not <i>val</i>



23

## Using Attribute Selectors

```
// Select all <img> elements with alt attributes
var results = $("img[alt]");

// Select all <img> elements with alt="photo" attributes
var results = $("img[alt=photo]");

// Select all <img> elements with src attributes whose value
// ends with ".png"
var results = $("img[src$=\\.png]");

// Select all <img> elements with src attributes whose value
// ends with ".png" and with alt="photo" attributes
var results = $("img[src$=\\.png][alt=photo]");

// Select all <img> elements that lack alt attributes
var results = $("img:not([alt])");
```



24

## Pseudo-Class Selectors (Filters)

Selector	Description
:animated	Selects elements that are currently being animated by jQuery
:button	Selects <button> and type="button" elements
:checkbox	Selects type="checkbox" elements
:checked	Selects checkbox and radio-button elements that are checked
:contains()	Selects elements that contain the specified text (case-sensitive)
:disabled()	Selects elements that are disabled
:empty	Selects elements that have no children
:enabled	Selects elements that are enabled
:eq()	Selects the element at the specified index within the matched set
:even	Selects elements with even indexes within the matched set



25

## Pseudo-Class Selectors, Cont.

Selector	Description
:file	Matches type="file" elements
:first	Selects the first element in the matched set
:first-child	Selects elements that are the first child of their parent
:focus	Selects elements that currently have the input focus
:gt()	Selects elements whose index is greater than the one specified
:has()	Selects elements which contain at least one element that matches the specified selector
:header	Selects header elements (h1, h2, h3, and so on)
:hidden	Selects elements that are hidden
:image	Selects <img> elements
:input	Selects <input>, <textarea>, <select>, and <button> elements



26

## Pseudo-Class Selectors, Cont.

Selector	Description
:last	Selects the last element in the matched set
:last-child	Selects elements that are the last child of their parent
:lt()	Selects elements whose index is less than the one specified
:not()	Selects elements that do not match the specified selector
:nth()	Selects the element at the specified index within the matched set (synonym for :eq())
:nth-child()	Selects elements that are the nth child of their parent
:odd	Selects elements with odd indexes within the matched set
:only-child	Selects elements that are the only child of their parent
:parent	Selects elements that are the parent of other elements
:password	Matches type="password" elements



27

## Pseudo-Class Selectors, Cont.

Selector	Description
:radio	Selects type="radio" elements
:reset	Selects type="reset" elements
:selected	Selects elements that are currently selected
:submit	Selects type="submit" elements
:text	Selects type="text" elements
:visible	Selects elements that are visible



28

## Using Pseudo-Class Selectors

```
// Select odd-numbered table rows
var results = $("table tr:odd");

// Select every third table row
var results = $("table tr:nth-child(3n)");

// Select all checkboxes and radio buttons that are checked
var results = $("input:checked");

// Select all checkboxes that are checked
var results = $("input:checkbox:checked");

// Select all checkboxes that are not checked
var results = $("input:checkbox:not(:checked)");
```



29

## DEMO

Selectors



Consulting/Training



30