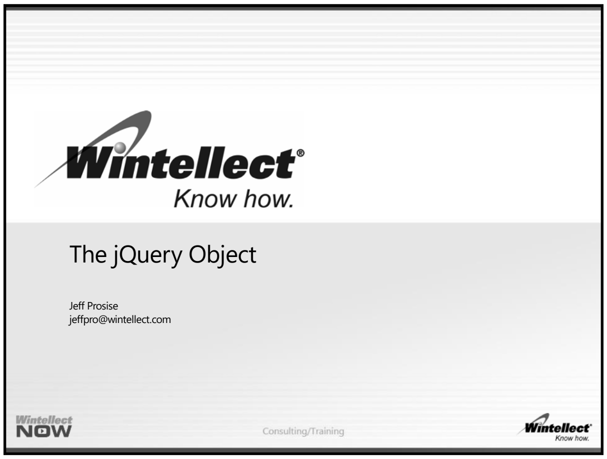


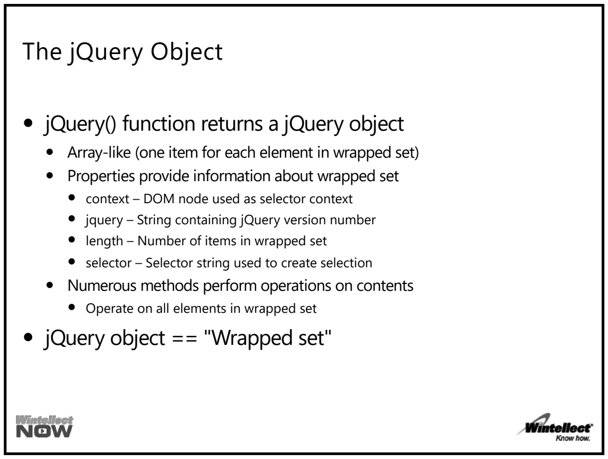
Wintellect® Know how.

## The jQuery Object

Jeff Prosise
jeffpro@wintellect.com

Wintellect NOW Consulting/Training Wintellect Know how.

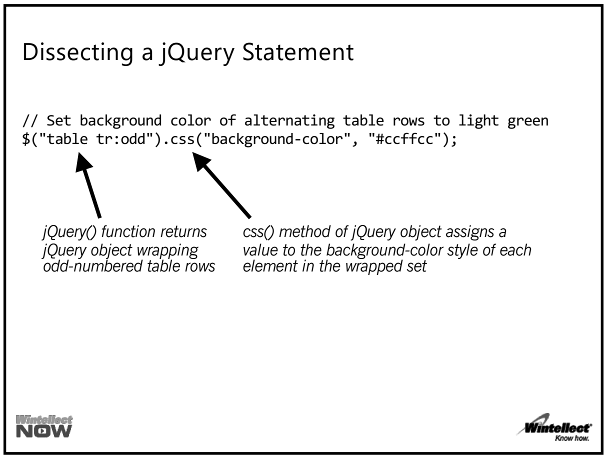
1



## The jQuery Object

- jQuery() function returns a jQuery object
  - Array-like (one item for each element in wrapped set)
  - Properties provide information about wrapped set
    - context – DOM node used as selector context
    - jquery – String containing jQuery version number
    - length – Number of items in wrapped set
    - selector – Selector string used to create selection
  - Numerous methods perform operations on contents
    - Operate on all elements in wrapped set
- jQuery object == "Wrapped set"

2



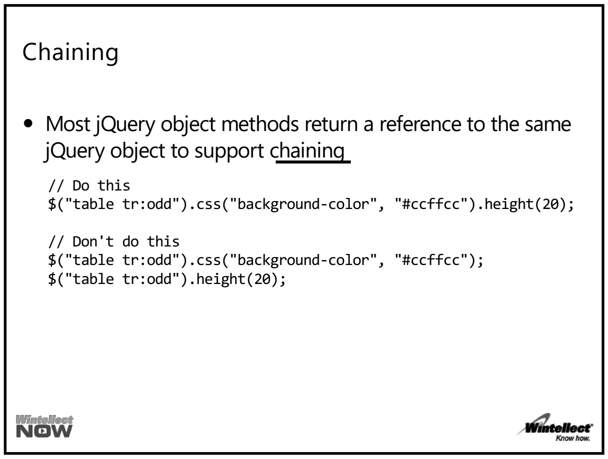
## Dissecting a jQuery Statement

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

*jQuery() function returns jQuery object wrapping odd-numbered table rows*

*css() method of jQuery object assigns a value to the background-color style of each element in the wrapped set*

3



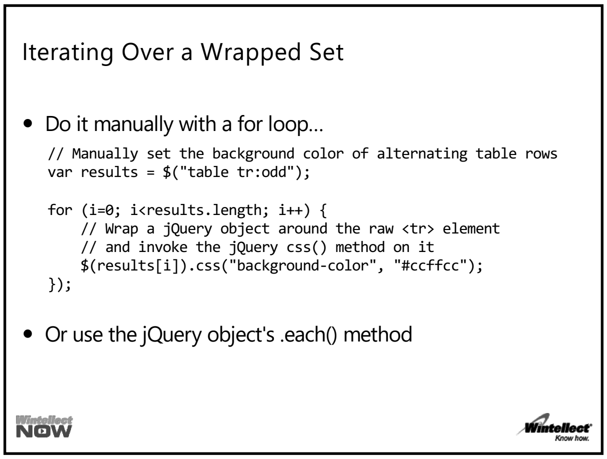
## Chaining

- Most jQuery object methods return a reference to the same jQuery object to support chaining

```
// Do this
$("table tr:odd").css("background-color", "#ccffcc").height(20);

// Don't do this
$("table tr:odd").css("background-color", "#ccffcc");
$("table tr:odd").height(20);
```

4



## Iterating Over a Wrapped Set

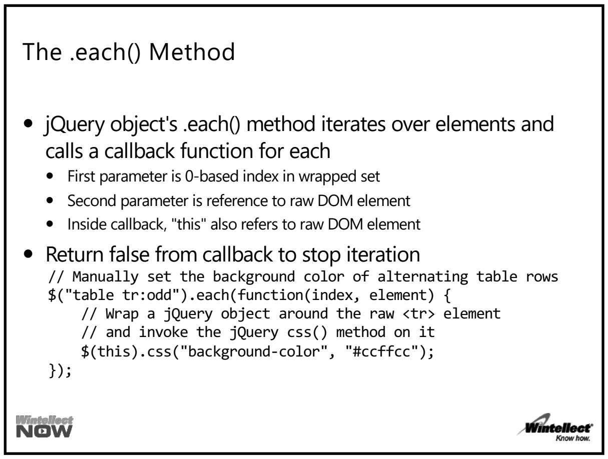
- Do it manually with a for loop…

```
// Manually set the background color of alternating table rows
var results = $("table tr:odd");

for (i=0; i<results.length; i++) {
    // Wrap a jQuery object around the raw <tr> element
    // and invoke the jQuery css() method on it
    $(results[i]).css("background-color", "#ccffcc");
});
```

- Or use the jQuery object's .each() method

5

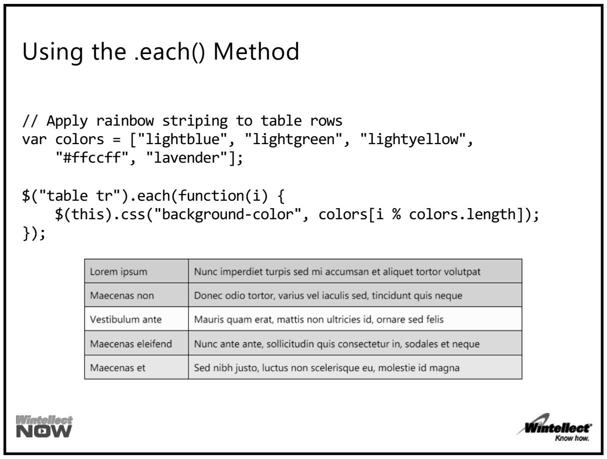


## The .each() Method

- jQuery object's .each() method iterates over elements and calls a callback function for each
  - First parameter is 0-based index in wrapped set
  - Second parameter is reference to raw DOM element
  - Inside callback, "this" also refers to raw DOM element
- Return false from callback to stop iteration

```
// Manually set the background color of alternating table rows
$("table tr:odd").each(function(index, element) {
    // Wrap a jQuery object around the raw <tr> element
    // and invoke the jQuery css() method on it
    $(this).css("background-color", "#ccffcc");
});
```

6



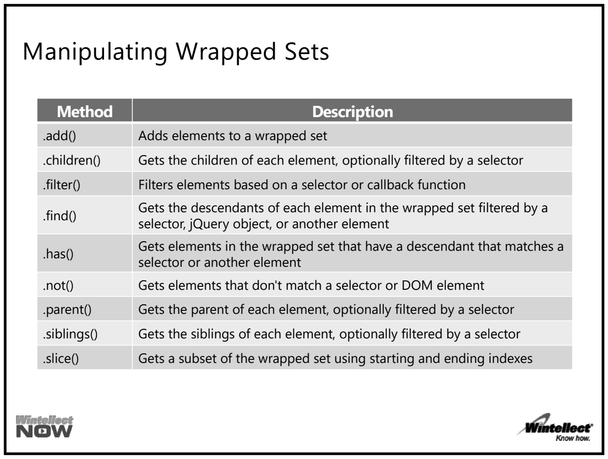
## Using the .each() Method

```
// Apply rainbow striping to table rows
var colors = ["lightblue", "lightgreen", "lightyellow",
    "#ffccff", "lavender"];

$("table tr").each(function(i) {
    $(this).css("background-color", colors[i % colors.length]);
});
```

| | |
|---|---|
| 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 |

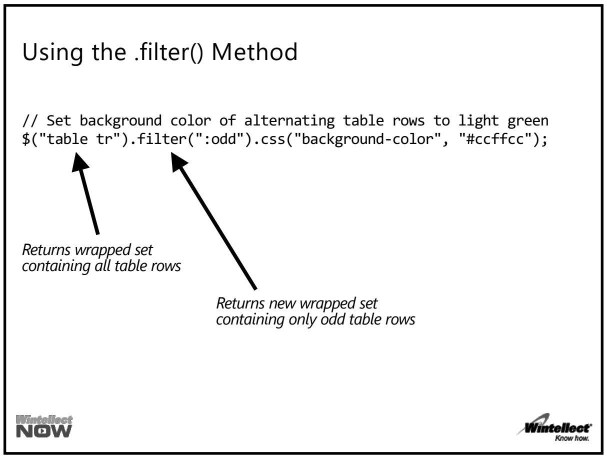
7



## Manipulating Wrapped Sets

| Method | Description |
|---|---|
| .add() | Adds elements to a wrapped set |
| .children() | Gets the children of each element, optionally filtered by a selector |
| .filter() | Filters elements based on a selector or callback function |
| .find() | Gets the descendants of each element in the wrapped set filtered by a selector, jQuery object, or another element |
| .has() | Gets elements in the wrapped set that have a descendant that matches a selector or another element |
| .not() | Gets elements that don't match a selector or DOM element |
| .parent() | Gets the parent of each element, optionally filtered by a selector |
| .siblings() | Gets the siblings of each element, optionally filtered by a selector |
| .slice() | Gets a subset of the wrapped set using starting and ending indexes |

8



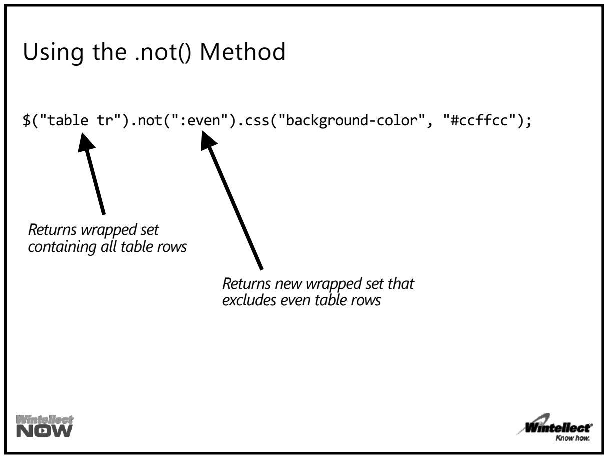
## Using the .filter() Method

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

*Returns wrapped set containing all table rows*

*Returns new wrapped set containing only odd table rows*

9



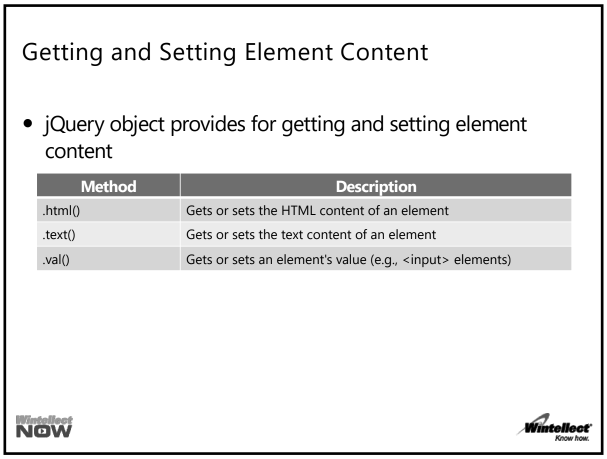
## Using the .not() Method

```
$("table tr").not(":even").css("background-color", "#ccffcc");
```

*Returns wrapped set containing all table rows*

*Returns new wrapped set that excludes even table rows*

10

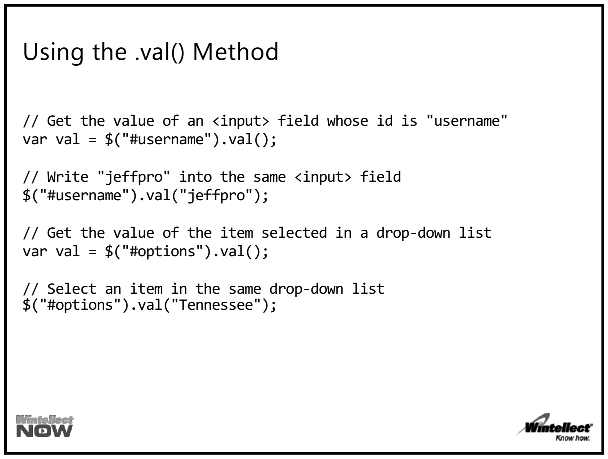


## Getting and Setting Element Content

- jQuery object provides for getting and setting element content

| Method | Description |
|---|---|
| .html() | Gets or sets the HTML content of an element |
| .text() | Gets or sets the text content of an element |
| .val() | Gets or sets an element's value (e.g., <input> elements) |

11



## Using the .val() Method

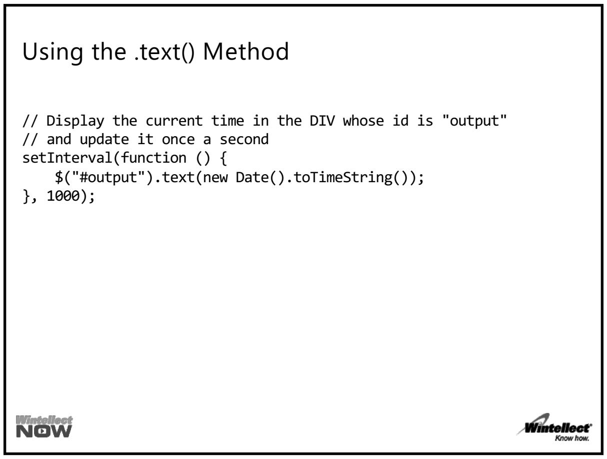
```
// Get the value of an <input> field whose id is "username"
var val = $("#username").val();

// Write "jeffpro" into the same <input> field
$("#username").val("jeffpro");

// Get the value of the item selected in a drop-down list
var val = $("#options").val();

// Select an item in the same drop-down list
$("#options").val("Tennessee");
```

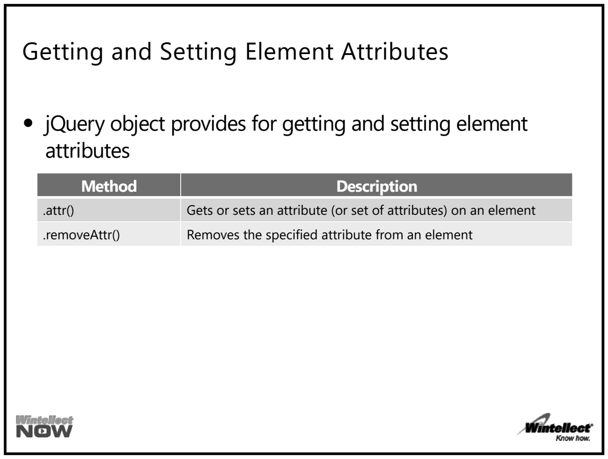
12



## Using the .text() Method

```
// Display the current time in the DIV whose id is "output"
// and update it once a second
setInterval(function () {
    $("#output").text(new Date().toTimeString());
}, 1000);
```

13

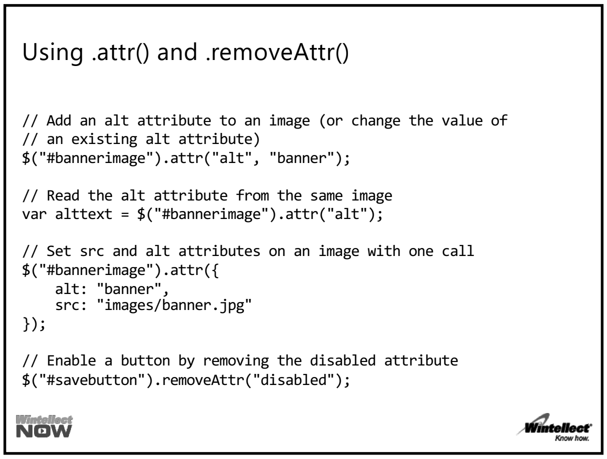


## Getting and Setting Element Attributes

- jQuery object provides for getting and setting element attributes

| Method | Description |
|---|---|
| .attr() | Gets or sets an attribute (or set of attributes) on an element |
| .removeAttr() | Removes the specified attribute from an element |

14



## Using .attr() and .removeAttr()

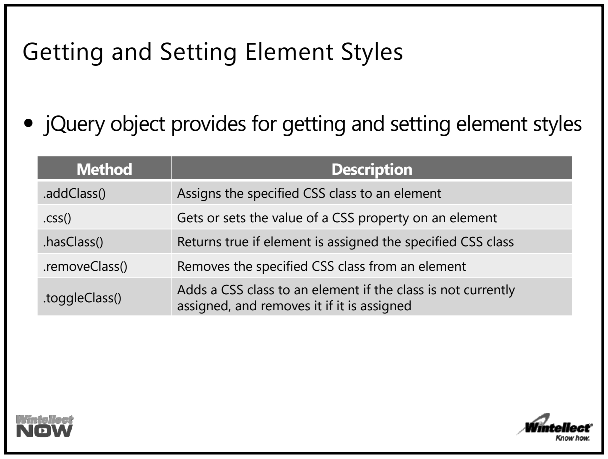
```
// Add an alt attribute to an image (or change the value of
// an existing alt attribute)
$("#bannerimage").attr("alt", "banner");

// Read the alt attribute from the same image
var alttext = $("#bannerimage").attr("alt");

// Set src and alt attributes on an image with one call
$("#bannerimage").attr({
    alt: "banner",
    src: "images/banner.jpg"
});

// Enable a button by removing the disabled attribute
$("#savebutton").removeAttr("disabled");
```

15

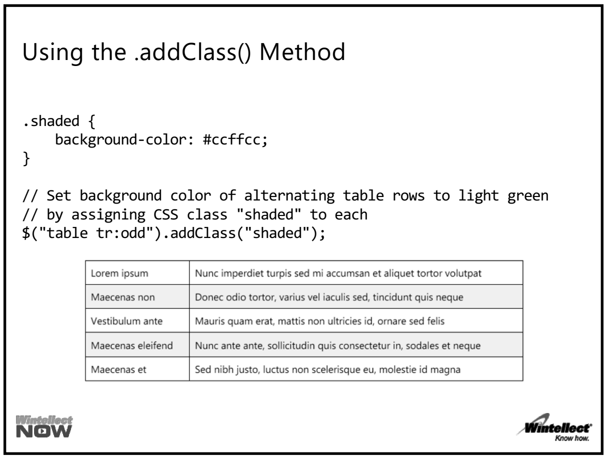


## Getting and Setting Element Styles

- jQuery object provides for getting and setting element styles

| Method | Description |
|---|---|
| .addClass() | Assigns the specified CSS class to an element |
| .css() | Gets or sets the value of a CSS property on an element |
| .hasClass() | Returns true if element is assigned the specified CSS class |
| .removeClass() | Removes the specified CSS class from an element |
| .toggleClass() | Adds a CSS class to an element if the class is not currently assigned, and removes it if it is assigned |

16



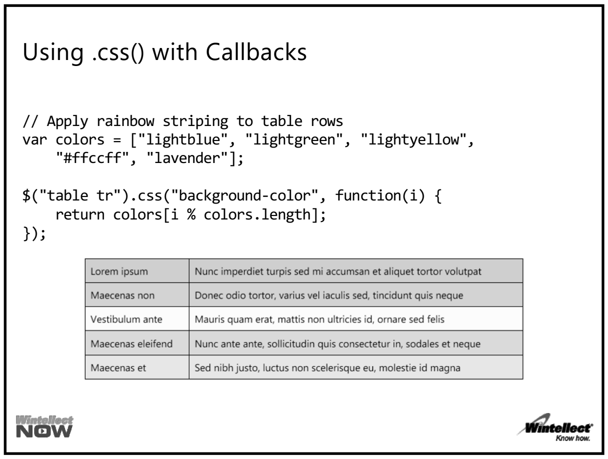
## Using the .addClass() Method

```
.shaded {
    background-color: #ccffcc;
}

// Set background color of alternating table rows to light green
// by assigning CSS class "shaded" to each
$("table tr:odd").addClass("shaded");
```

| | |
|---|---|
| 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 |

17



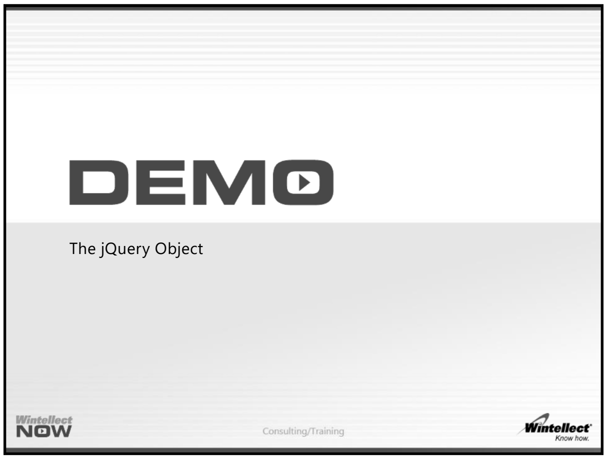
## Using .css() with Callbacks

```
// Apply rainbow striping to table rows
var colors = ["lightblue", "lightgreen", "lightyellow",
    "#ffccff", "lavender"];

$("table tr").css("background-color", function(i) {
    return colors[i % colors.length];
});
```

| | |
|---|---|
| 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 |

18

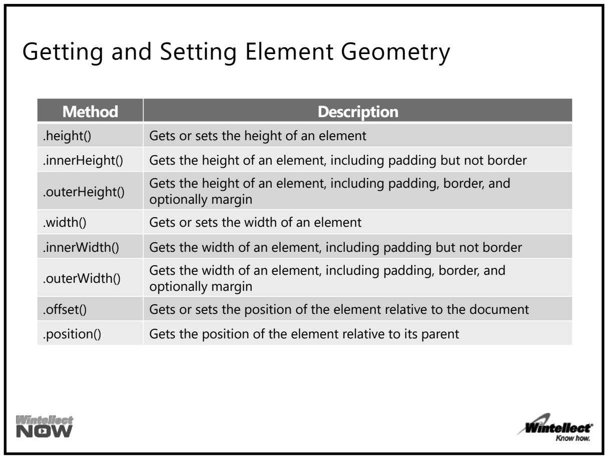


# DEMO

The jQuery Object

Consulting/Training

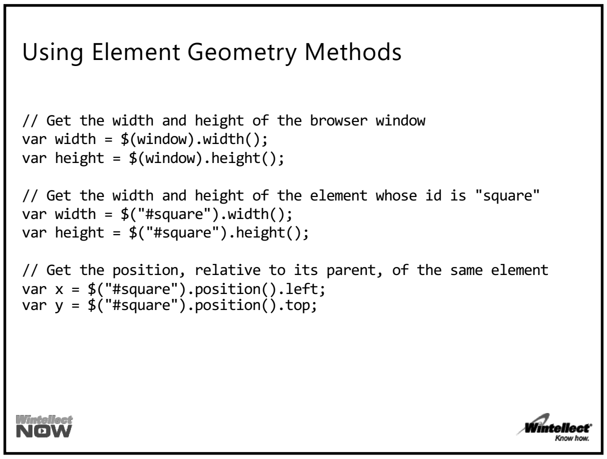
19



## Getting and Setting Element Geometry

| Method | Description |
|---|---|
| .height() | Gets or sets the height of an element |
| .innerHeight() | Gets the height of an element, including padding but not border |
| .outerHeight() | Gets the height of an element, including padding, border, and optionally margin |
| .width() | Gets or sets the width of an element |
| .innerWidth() | Gets the width of an element, including padding but not border |
| .outerWidth() | Gets the width of an element, including padding, border, and optionally margin |
| .offset() | Gets or sets the position of the element relative to the document |
| .position() | Gets the position of the element relative to its parent |

20



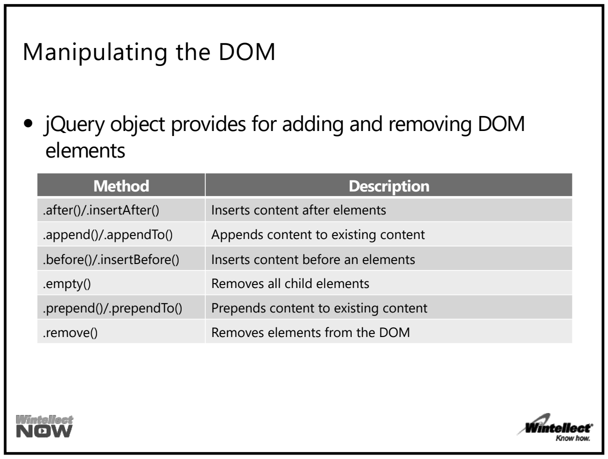
## Using Element Geometry Methods

```
// Get the width and height of the browser window
var width = $(window).width();
var height = $(window).height();

// Get the width and height of the element whose id is "square"
var width = $("#square").width();
var height = $("#square").height();

// Get the position, relative to its parent, of the same element
var x = $("#square").position().left;
var y = $("#square").position().top;
```

21

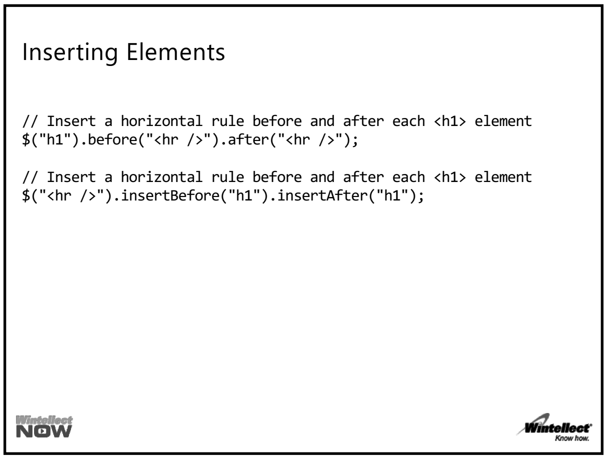


## Manipulating the DOM

- jQuery object provides for adding and removing DOM elements

| Method | Description |
|---|---|
| .after()/.insertAfter() | Inserts content after elements |
| .append()/.appendTo() | Appends content to existing content |
| .before()/.insertBefore() | Inserts content before an elements |
| .empty() | Removes all child elements |
| .prepend()/.prependTo() | Prepends content to existing content |
| .remove() | Removes elements from the DOM |

22

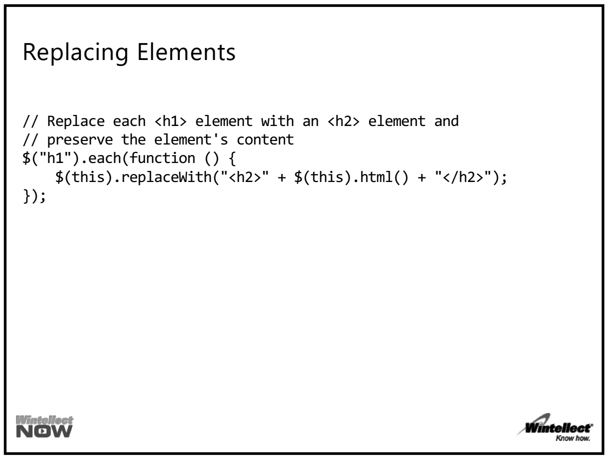


## Inserting Elements

```
// Insert a horizontal rule before and after each <h1> element
$("h1").before("<hr />").after("<hr />");

// Insert a horizontal rule before and after each <h1> element
$("<hr />").insertBefore("h1").insertAfter("h1");
```

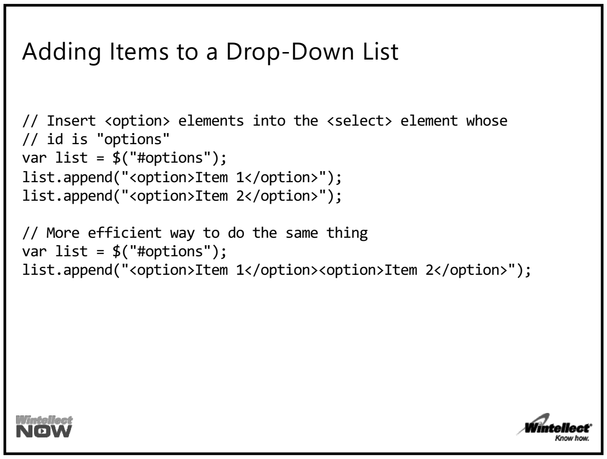
23



## Replacing Elements

```
// Replace each <h1> element with an <h2> element and
// preserve the element's content
$("h1").each(function () {
    $(this).replaceWith("<h2>" + $(this).html() + "</h2>");
});
```

24

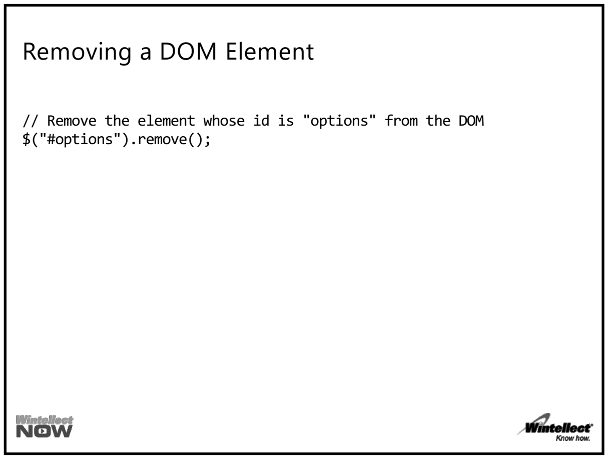


## Adding Items to a Drop-Down List

```
// Insert <option> elements into the <select> element whose
// id is "options"
var list = $("#options");
list.append("<option>Item 1</option>");
list.append("<option>Item 2</option>");

// More efficient way to do the same thing
var list = $("#options");
list.append("<option>Item 1</option><option>Item 2</option>");
```

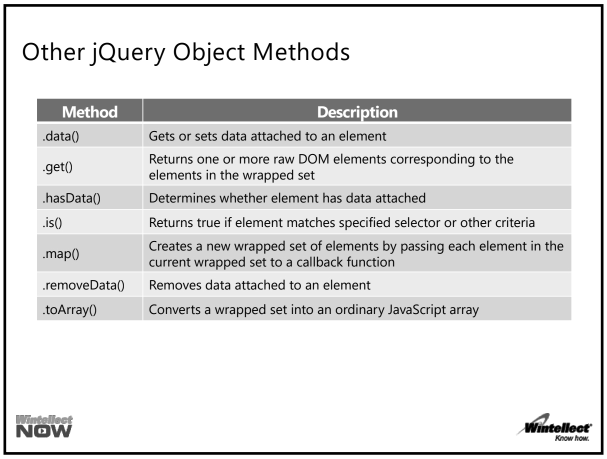
25



## Removing a DOM Element

```
// Remove the element whose id is "options" from the DOM
$("#options").remove();
```

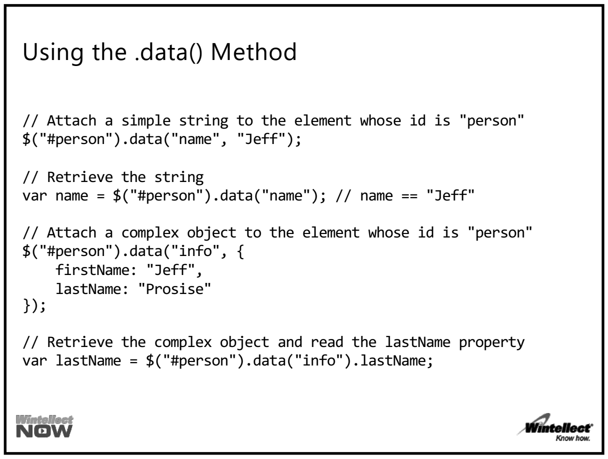
26



## Other jQuery Object Methods

| Method | Description |
|---|---|
| .data() | Gets or sets data attached to an element |
| .get() | Returns one or more raw DOM elements corresponding to the elements in the wrapped set |
| .hasData() | Determines whether element has data attached |
| .is() | Returns true if element matches specified selector or other criteria |
| .map() | Creates a new wrapped set of elements by passing each element in the current wrapped set to a callback function |
| .removeData() | Removes data attached to an element |
| .toArray() | Converts a wrapped set into an ordinary JavaScript array |

27



## Using the .data() Method

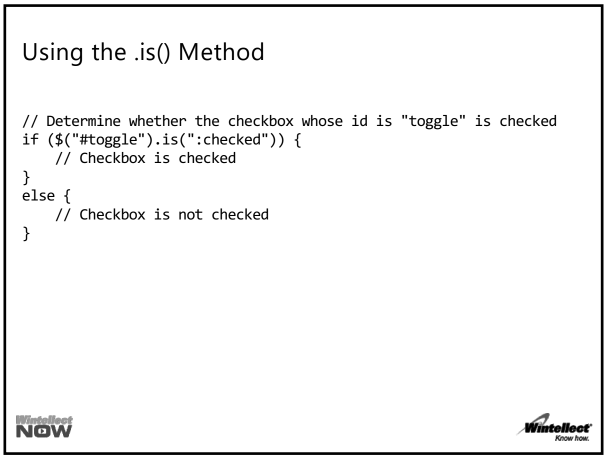
```
// Attach a simple string to the element whose id is "person"
$("#person").data("name", "Jeff");

// Retrieve the string
var name = $("#person").data("name"); // name == "Jeff"

// Attach a complex object to the element whose id is "person"
$("#person").data("info", {
    firstName: "Jeff",
    lastName: "Prosise"
});

// Retrieve the complex object and read the lastName property
var lastName = $("#person").data("info").lastName;
```

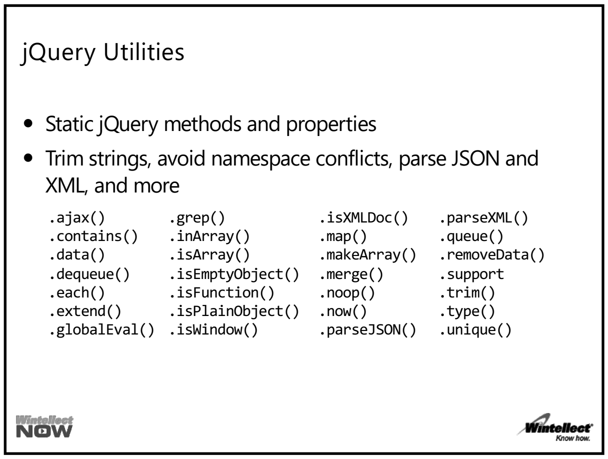
28



## Using the .is() Method

```
// Determine whether the checkbox whose id is "toggle" is checked
if ($("#toggle").is(":checked")) {
    // Checkbox is checked
}
else {
    // Checkbox is not checked
}
```

29

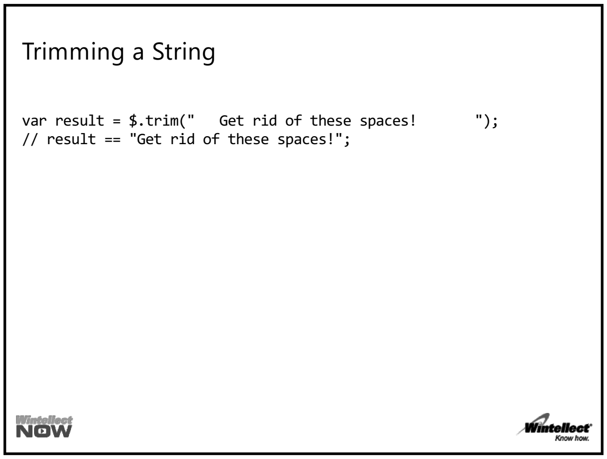


## jQuery Utilities

- Static jQuery methods and properties
- Trim strings, avoid namespace conflicts, parse JSON and XML, and more

```
.ajax()        .grep()           .isXMLDoc()    .parseXML()
.contains()    .inArray()        .map()         .queue()
.data()        .isArray()        .makeArray()   .removeData()
.dequeue()     .isEmptyObject()  .merge()       .support
.each()        .isFunction()     .noop()        .trim()
.extend()      .isPlainObject()  .now()         .type()
.globalEval()  .isWindow()       .parseJSON()   .unique()
```

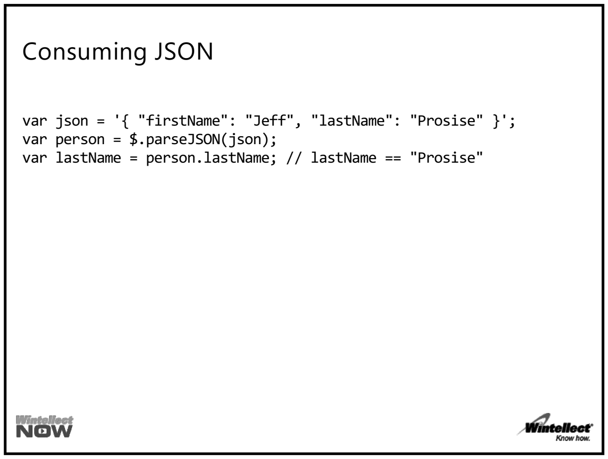
30



## Trimming a String

```
var result = $.trim("   Get rid of these spaces!       ");
// result == "Get rid of these spaces!";
```

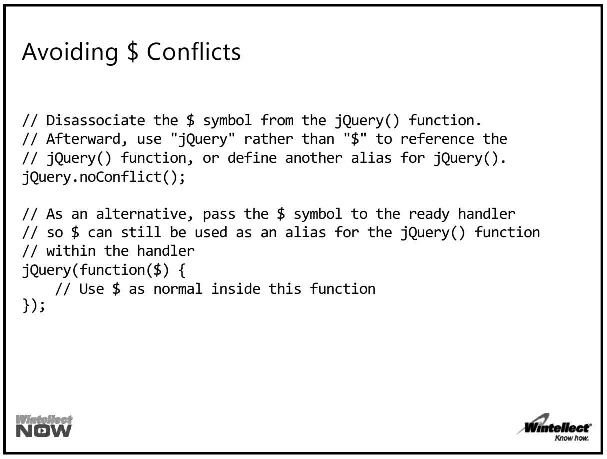
31



## Consuming JSON

```
var json = '{ "firstName": "Jeff", "lastName": "Prosise" }';
var person = $.parseJSON(json);
var lastName = person.lastName; // lastName == "Prosise"
```

32

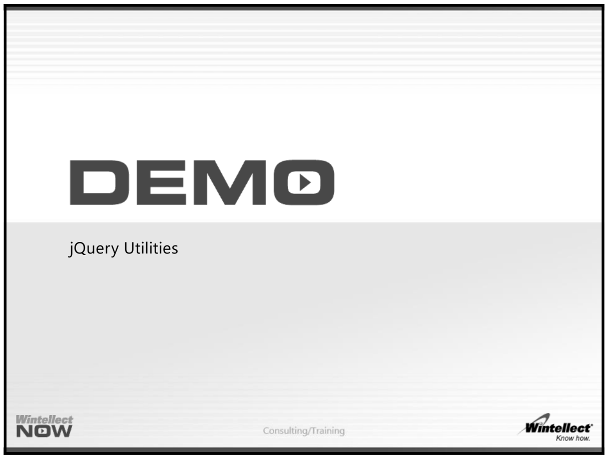


## Avoiding $ Conflicts

```
// Disassociate the $ symbol from the jQuery() function.
// Afterward, use "jQuery" rather than "$" to reference the
// jQuery() function, or define another alias for jQuery().
jQuery.noConflict();

// As an alternative, pass the $ symbol to the ready handler
// so $ can still be used as an alias for the jQuery() function
// within the handler
jQuery(function($) {
    // Use $ as normal inside this function
});
```

33



DEMO

jQuery Utilities

Consulting/Training

34