

Using jQuery Selectors

Dan Wahlin

www.pluralsight.com



Agenda

- What are Selectors?
- Selecting Nodes by Tag Name
- Selecting Nodes by ID
- Selecting Nodes by Class Name
- Selecting Nodes by Attribute Value
- Selecting Input Nodes
- Additional Selector Features

Introduction to Selectors

- Selectors allow page elements to be selected
- Single or multiple elements are supported
- A selector identifies an HTML element/tag that you will manipulate with jQuery Code

```
<div id="CustomersDiv" class="Bright">  
  <span class="Text">Welcome John</span>  
</div>
```

Selector Syntax

`$(selectorExpression)`

`jQuery(selectorExpression)`

Agenda

- What are Selectors?
- **Selecting Nodes by Tag Name**
- Selecting Nodes by ID
- Selecting Nodes by Class Name
- Selecting Nodes by Attribute Value
- Selecting Input Nodes
- Additional Selector Features

Selecting by Tag Name

`$('p')` selects all `<p>` elements

`$('a')` selects all `<a>` elements

Selecting Multiple Tags

- To reference multiple tags, use the , character to separate the elements:

`$('p, a, span')`

selects all paragraphs, anchors, and span elements

Selecting Descendants

- `$('ancestor descendant')` selects all descendants of the ancestor:

`$('table tr')`

Selects all `tr` elements that are descendants of the `table` element

- Descendants are children, grandchildren, etc of the designated ancestor element

Agenda

- What are Selectors?
- Selecting Nodes by Tag Name
- **Selecting Nodes by ID**
- Selecting Nodes by Class Name
- Selecting Nodes by Attribute Value
- Selecting Input Nodes
- Additional Selector Features

Selecting by Element ID

- Use the # character to select elements by ID:

`$('#myID')`

selects `<p id="myID">` element

Agenda

- What are Selectors?
- Selecting Nodes by Tag Name
- Selecting Nodes by ID
- **Selecting Nodes by Class Name**
- Selecting Nodes by Attribute Value
- Selecting Input Nodes
- Additional Selector Features

Selecting Elements by Class Name

- Use the `.` character to select elements by class name:

`$('.myClass')`

selects `<p class="myClass">` element

Selecting Multiple Class Names

- To reference multiple tags, use the , character to separate the class names:

`$('.BlueDiv, .RedDiv')`

selects all elements containing the class `BlueDiv` and `RedDiv`

Selecting by Tag Name and Class Name

- You can combine this with element names as well:

`$('a.myClass')`

selects only `<a>` tags with `class="myClass"`

Agenda

- What are Selectors?
- Selecting Nodes by Tag Name
- Selecting Nodes by ID
- Selecting Nodes by Class Name
- **Selecting Nodes by Attribute Value**
- Selecting Input Nodes
- Additional Selector Features

Selecting By Attribute Value

- Use brackets `[attribute]` to select based on attribute name and/or attribute value:

`$('a[title]')`

selects all `<a>` elements that have a title attribute

`$('a[title="Programming Info"]')`

selects all anchor elements that have a "Programming Info" title attribute value

Agenda

- What are Selectors?
- Selecting Nodes by Tag Name
- Selecting Nodes by ID
- Selecting Nodes by Class Name
- Selecting Nodes by Attribute Value
- **Selecting Input Nodes**
- Additional Selector Features

Selecting Input Elements

- `$(':input')` selects all input elements including: input, select, textarea, button, image, radio and more

`$(':input[type="radio"]')`

targets all radio buttons on the page...but is it the most efficient selector?

Agenda

- What are Selectors?
- Selecting Nodes by Tag Name
- Selecting Nodes by ID
- Selecting Nodes by Class Name
- Selecting Nodes by Attribute Value
- Selecting Input Nodes
- **Additional Selector Features**

Using Contains in Selectors

- **:contains()** will select elements that match the contents within the contains exception:

```
$('div:contains("pluralsight")')
```

selects **div's** that contain the text **pluralsight** (note that the match is case-sensitive)

```
<div>Expert pluralsight Training</div>
```

Selecting Even or Odd Rows in a Table

- `$('tr:odd')` and `$('tr:even')` is the jQuery syntax for selecting odd or even rows respectively
- Remember the index is 0 based - the first row in the table is 0:
 - Odd would return 1, 3, 5, 7, 9, etc
 - Even would return 0, 2, 4, 6, 8, etc

Selecting the First Child

- **`$('element:first-child')`** selects the first child of every element group:

`$('span:first-child')`

```
<div>
```

```
  <span>First Child, first group</span>
```

```
  <span>Second Child, first group</span>
```

```
</div>
```

```
<div>
```

```
  <span>First Child, second group</span>
```

```
  <span>Second Child, second group Child</span>
```

```
</div>
```

Using Starts With in Selectors

- `[attribute^="value"]` will select all elements with an attribute that begins with stated value:

`$('input[value^="Events"]')`

selects any input element whose value attribute begins with "Events":

`<input type="button" value="Events – World"/>`

`<input type="button" value="Events – National"/>`

`<input type="button" value="Events – Local"/>`

Using Ends With in Selectors

- `[attribute$="value"]` will select all elements with an attribute that ends with stated value:

`$(input[value$="Events"]')`

selects any input element whose value attribute ends with "Events":

```
<input type="button" value="World Events"/>
```

```
<input type="button" value = "National Events"/>
```

```
<input type="button" value = "Local Events"/>
```


Find Attributes Containing a Value

- `[attribute*="value"]` will select all elements with an attribute that contain the stated value:

`$('input[value*="Events"]')`

selects any input element whose value attribute contains "Events":

`<input type="button" value="World Events 2011"/>`

`<input type="button" value ="National Events 2011"/>`

`<input type="button" value ="Local Events 2011"/>`

Summary

- **Selectors allow any type of HTML element to be located in an HTML page**
- **Key jQuery selector characters include:**
 - # for ID selections
 - . for class selection
 - ^ for attributes starting with a value
 - \$ for attributes ending with a value
 - * for attributes containing a value
- **Mastering selectors is the key to being productive with jQuery!**

For more in-depth **online** developer **training** visit



on-demand content from authors you **trust**