

Retrieving Page Content

Kayartaya Vinod

jQuery selectors and filters:

Overview

- jQuery selectors and filters retrieve content from the documents so it can be manipulated using other functions (think of this as the “Query” part of the “jQuery”)
 - jQuery selectors return an array of objects that match the selection criteria
 - jQuery filters operate on a selector to further refine the results array that the selector returns
- This array is not a set of DOM elements
- It is a collection of jQuery objects that provide a large number of predefined functions for further operating on the objects

Using basic jQuery selectors

- CSS-style selectors and filters are based on familiar CSS syntax, and work pretty much the same way as CSS does
- The CSS selectors listed here correspond directly to their CSS counterparts

SELECTOR	PURPOSE
<code>tagname</code>	Finds all elements that are named <i>tagname</i>
<code>#identifier</code>	Finds all elements with ID of <i>identifier</i>
<code>.className</code>	Finds all elements that have class attribute with the value of <i>className</i>
<code>tag.className</code>	Gets elements of type <i>tag</i> that have a class attribute with the value of <i>className</i>
<code>tag#id.className</code>	Retrieves the <i>tag</i> element that has an ID of <i>id</i> and a class attribute with the value of <i>className</i>
<code>*</code>	Finds all of the elements on the page

Using basic jQuery selectors

- Using jQuery vs. using the plain browser DOM

```
<html>
<head>
<title>Document</title>
</head>
<body>
<ul id="list1">
  <li class="a">item 1</li>
  <li class="a">item 2</li>
  <li class="b">item 3</li>
  <li class="b">item 3</li>
</ul>
<p>This is paragraph 1</p>
<p>This is paragraph 2</p>
<p>This is paragraph 3</p>
<p>This is paragraph 4</p>
</body>
</html>
```

Get all <p> tags (DOM):

```
document.getElementsByTagName("p");
```

Get all <p> tags (jQuery):

```
$("p");
```

Get the tag with id "list1" (DOM):

```
document.getElementById("list1");
```

Get the tag with id "list1" (jQuery):

```
$("#list1");
```

Get all tags with class "a" (jQuery):

```
$("li.a");
```

Get all tags with class "b", but only if they are inside a (jQuery):

```
$("ul .b");
```

Using basic jQuery selectors

- The hierarchy and combination selectors allow you to get a little more advanced in selecting page content
- You can select elements based on hierarchical relationships or on a series of common criteria

SELECTOR	PURPOSE
<i>selector, selector, ...</i>	Finds all of the specified selectors
<i>.class1.class2</i>	Finds all elements with both <i>.class1</i> and <i>.class2</i> applied
<i>parent > child</i>	Finds all <i>child</i> elements that are direct children of elements of type <i>parent</i>
<i>ancestor descendant</i>	Finds all <i>descendant</i> elements that are contained within elements of type <i>ancestor</i>
<i>prev + next</i>	Finds all <i>next</i> elements that are next to a <i>prev</i> element
<i>prev ~ siblings</i>	Finds all sibling elements that come after <i>prev</i> and match the <i>siblings</i> selector

DEMO

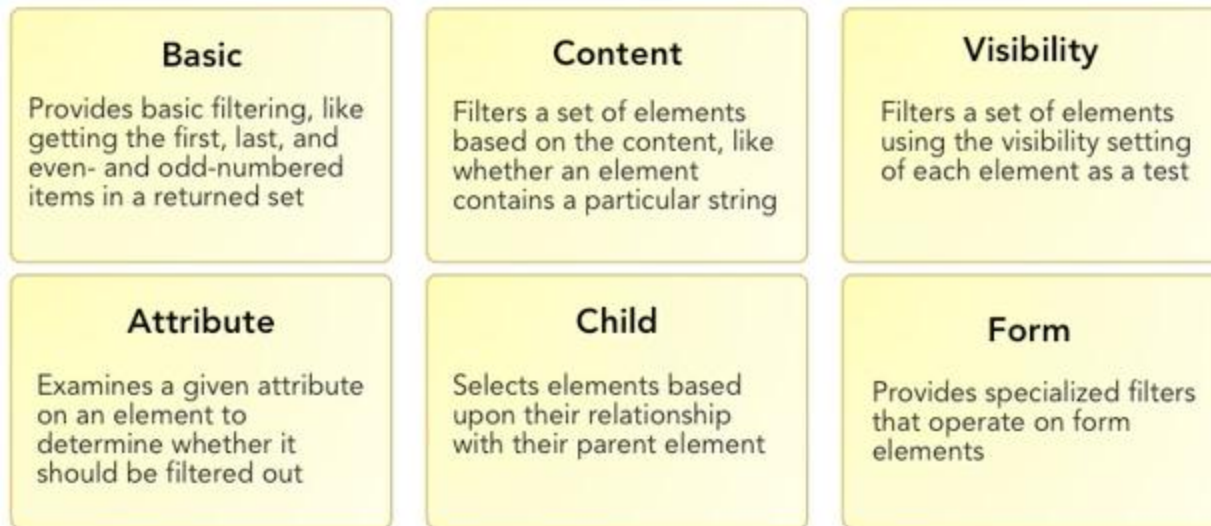


JQUERY SELECTORS AND FILTERS: USING FILTERS



jQuery Selectors and filters: Using Filters

- Filters work with selectors to provide even more fine-grained control over how elements are selected in the document
- jQuery filters fall into six different categories



Using basic jQuery filters

- You can refine a selector by including elements that match certain conditions, like position or index

FILTER	PURPOSE
:first	Selects only the first instance of the selector's returned set
:last	Selects only the last instance of the selector's returned set
:even	Selects only even-numbered elements in the selector's returned set
:odd	Selects only odd-numbered elements in the selector's returned set
:eq(n)	Filters out elements that are not positioned at the given index
:gt(n)	Includes elements that are past the given index
:lt(n)	Includes elements that are before the given index
:header	Selects all header elements (H1, H2, H3, etc)
:animated	Selects all elements that are currently being animated in some way
:not(selector)	Includes elements that do not match the given selector

DEMO



JQUERY SELECTORS AND FILTERS: ATTRIBUTE FILTERS

jQuery Selectors and filters:

Attribute Filters

- You can filter the results of a selector statement based on attribute content

FILTER	PURPOSE
<code>[attribute]</code>	Includes elements in the result set if they have the specified <i>attribute</i>
<code>[attribute=value]</code>	Includes elements in the result set if they have the specified <i>attribute</i> and it has the given <i>value</i>
<code>[attribute!=value]</code>	Includes elements in the result set only if they have the specified attribute and it doesn't have the given <i>value</i>
<code>[attribute^=value]</code>	Includes elements that have the specified <i>attribute</i> and it starts with the specified <i>value</i>
<code>[attribute\$=value]</code>	Includes elements that have the specified <i>attribute</i> and it ends with the specified <i>value</i>
<code>[attribute*=value]</code>	Includes elements that have the specified <i>attribute</i> and it contains the specified <i>value</i>
<code>[attrFilter1][attrFilterN]</code>	Includes elements that match all of the specified attribute filters

DEMO



JQUERY SELECTORS AND FILTERS: CONTENT AND VISIBILITY FILTERS

jQuery Selectors and filters: content and visibility filters

- You can examine the content of selected elements or their visibility property to determine whether they should be included or excluded from the final set

CONTENT FILTER	PURPOSE
<code>:contains(text)</code>	Filters the selection to only include elements that contain the <i>text</i> string
<code>:empty</code>	Filters the selection to only include empty elements
<code>:has(selector)</code>	Matches elements that contain at least one element that has the specified selector
<code>:parent</code>	Matches all elements that are parents (i.e. they contain at least one other element, including text)

VISIBILITY FILTER	PURPOSE
<code>:visible</code>	Filters the selection to only include visible elements
<code>:hidden</code>	Filters the selection to only include hidden elements

JQUERY SELECTORS AND FILTERS: CHILD FILTERS



jQuery Selectors and filters:

Child filters

- You can refine a selector by examining the relationship each element has with its parent element

FILTER	PURPOSE
<code>:nth-child(<i>index</i>)</code> <code>:nth-child(even)</code> <code>:nth-child(odd)</code> <code>:nth-child(<i>equation</i>)</code>	Matches elements at <i>index</i> , or even or odd increments, or who match an equation of the form $Xn+M$ (for example, $2n$ or $3n+1$)
<code>:first-child</code>	Matches elements who are the first child of their parent
<code>:last-child</code>	Matches elements who are the last child of their parent
<code>:only-child</code>	Matches elements who are the only child of their parent

DEMO



JQUERY SELECTORS AND FILTERS: FORM SELECTORS

jQuery Selectors and filters:

Form selectors

- You can use form selectors to deal with form elements

SELECTOR	PURPOSE
:input	Finds all input, select, textarea, and button elements
:text	Finds all text elements
:password	Finds all password elements
:radio	Finds all radio button elements
:checkbox	Finds all checkbox elements
:submit	Finds all submit elements
:reset	Finds all reset elements
:image	Finds all image elements
:button	Finds all button elements
:file	Finds all file upload elements

jQuery Selectors and filters:

Form filters

- You can perform additional filtering of form elements, such as whether items are checked, selected, or enabled

SELECTOR	PURPOSE
<code>:enabled</code>	Matches all form elements that are enabled
<code>:disabled</code>	Matches all form elements that are disabled
<code>:checked</code>	Matches all form elements that are checked (radiobuttons and checkboxes)
<code>:selected</code>	Matches all form elements that are selected

DEMO



TRAVERSING DOCUMENT INFORMATION



Traversing document information

- You can traverse the information returned from a document easily

FUNCTION / PROPERTY	PURPOSE
size(), length	The number of elements in the jQuery result set
get()	Returns an array of all matched DOM elements. Useful if you need to operate on the DOM elements themselves instead of using built-in jQuery functions
get(index)	Access a single matched DOM element at a specified index in the matched set
find(expression)	Searches for descendent elements that match the specified expression
each(fn)	Execute a function within the context of every matched element

DEMO




JQUERY STATEMENT CHAINING



jQuery statement chaining

- One of jQuery's most powerful features is its ability to chain multiple functions together to perform several operations in one line of code

```
$(selector).fn1().fn2().fn3();
```



Statement chain