

Manipulating Page Content

Kayartaya Vinod

Manipulating Content: Overview

- Once you have used selectors and filters to retrieve web page content, you usually want to do something with it
- Sometimes, you want to create new content to dynamically add into the page
- jQuery has functions for creating, copying, deleting, and moving content around, as well as wrapping page content in other content
- jQuery provides cross-browser support for working with CSS, including positioning and sizing information

Creating, setting, and getting content

- To create new HTML content, you simply pass a string containing new HTML to the `$()` function:

```
var newHeader = $("<h1>My New Header</h1>");  
var myStr = "<h1>My New Header</h1>";  
var newHeader = $(myStr);
```

- In addition to this method, you can use the `html()` and `text()` methods to get and set content on

FUNCTION	PURPOSE
<code>html()</code>	Returns the HTML content of the first matched element
<code>html(newcontent)</code>	Sets the HTML content of every matched element
<code>text()</code>	Returns the text content of the first matched element
<code>text(newtext)</code>	Sets the text content for all matched elements

DEMO



Manipulating attributes

- To inspect or change the value of attribute on elements, use jQuery's *attr* functions

FUNCTION	PURPOSE
<code>attr(name)</code>	Accesses property on the first matched element. This method makes it easy to retrieve a property value from the first matched element. If the element does not have an attribute with such a name, undefined is returned
<code>attr(properties)</code>	Sets a series of attributes on all matched elements using an object notation syntax. This is the best used for setting large numbers of properties at once <pre>\$("#img").attr({ src: "/images/hat.gif", title: "jQuery", alt: "jQuery Logo" });</pre>
<code>attr(key, value)</code>	Sets a single property to a value on all matched elements
<code>attr(key, fn)</code>	Sets a single property to a computed value, on all matched elements. Instead of supplying a string value, a function is provided that computes the value of the attribute
<code>removeAttr(name)</code>	Removes the named attribute from all matched elements

DEMO



Inserting content

- jQuery provides several functions for inserting content into the document, both before and after existing page elements

FUNCTION	PURPOSE
<code>append(content)</code>	Appends content to the inside of every matched element
<code>appendTo(selector)</code>	Appends all of the matched elements to another, specified, set of elements
<code>prepend(content)</code>	Prepends content to the inside of every matched element
<code>prependTo(selector)</code>	Prepends all the matched elements to another, specified, set of elements
<code>after(content)</code>	Inserts content after each of the matched elements
<code>before(content)</code>	Inserts content before each of the matched elements
<code>insertAfter(selector)</code>	Inserts all of the matched elements after another, specified, set of elements
<code>insertBefore(selector)</code>	Inserts all the matched elements before another, specified, set of elements

DEMO



Wrapping, replacing, and removing content

- jQuery can wrap existing content in the page, replace content, copy content and remove it

FUNCTION	PURPOSE
<code>wrap(html)</code>	Wraps each matched element with the specified HTML content
<code>wrap(element)</code>	Wraps each matched element with the specified element
<code>wrapAll(html)</code>	Wraps all the elements in the matched set with the specified HTML content
<code>wrapAll(element)</code>	Wraps all the elements in the matched set into a single wrapper element
<code>wrapInner(html)</code>	Wraps the inner child contents of each matched element (including text nodes) with an HTML structure
<code>wrapInner(element)</code>	Wraps the inner child contents of each matched element (including text nodes) with a DOM structure
<code>replaceWith(content)</code>	Replaces all matched elements with the specified HTML or DOM elements
<code>replaceAll(selector)</code>	Replaces the elements matched by the specified selector with the matched elements
<code>empty()</code>	Removes all child nodes from the set of matched elements
<code>remove()</code>	Removes all matched elements from the DOM
<code>clone()</code>	Clone matched DOM elements and selects the clones
<code>clone(bool)</code>	Clone matched DOM elements, and all their event handlers, and select the clones

DEMO



Working with CSS information

- jQuery's CSS functions provide easy, cross-browser access for setting properties and working with positioning and sizing information
- The `css()` function allows you to retrieve and set CSS styles for a set of matched elements

FILTER	PURPOSE
<code>css(name)</code>	Returns the value for the named CSS property for the first matched element
<code>css(properties)</code>	Sets the CSS properties of every matched element using an object-notation syntax: <pre>var cssObj = { 'background-color' : '#ddd', 'font-weight' : 'bold', 'color' : 'rgb(0,40,244)' } \$(this).css(cssObj);</pre>
<code>css(property, value)</code>	Sets a single style property to a value on all matched elements. If a number is provided, it is automatically converted into a pixel value, with the following exceptions:z-index, font-weight, opacity, zoom, and line-height

Working with CSS classes

- jQuery provides a set of functions for working with CSS classes on page elements
- Classes can be easily added, removed, toggled and detected

CSS FUNCTIONS	PURPOSE
<code>addClass(class)</code>	Adds the specified class(es) to each of the set of matched elements
<code>hasClass(class)</code>	Returns true if the specified class is present on at least one of the set of matched elements
<code>removeClass(class)</code>	Removes all the specified class(es) from the set of matched elements
<code>toggleClass(class)</code>	Adds the specified class if it is not present, removes the specified class if it is present
<code>toggleClass(class,switch)</code>	Adds the specified class if the switch is true, removes the specified class if the switch is false

Working with CSS positioning

- The CSS positioning functions provide cross-browser for figuring out the positions of elements

CSS FUNCTIONS	PURPOSE
<code>offset()</code>	Gets the current offset of the first matched element, in pixels, relative to the document
<code>offsetParent()</code>	Returns a jQuery collection with the positioned parent of the first matched element
<code>position()</code>	Gets the top and left position of an element relative to its offset parent
<code>scrollTop()</code>	Gets the scroll top offset of the first matched element
<code>scrollTop(val)</code>	Sets the scroll top offset to the given value on all matched elements
<code>scrollLeft()</code>	Gets the scroll left offset of the first matched element
<code>scrollLeft(val)</code>	Sets the scroll left offset to the given value on all matched elements

Working with CSS sizing information

- To retrieve cross-browser sizing information for elements, use the jQuery size related CSS functions

CSS FUNCTIONS	PURPOSE
<code>height()</code>	Gets the current computed, pixel, height of the first matched element
<code>height(val)</code>	Sets the CSS height of every matched element
<code>width()</code>	Gets the current computed, pixel, width of the first matched element
<code>width(val)</code>	Sets the CSS width of every matched element
<code>innerHeight()</code>	Gets the inner height (excluding the border and including the padding) for the first matched element
<code>innerWidth()</code>	Gets the inner width (excluding the border and including the padding) for the first matched element
<code>outerHeight(margin)</code>	Gets the outer height (includes the border and padding by default) for the first matched element. If the margin argument is true, then the margin values are also included
<code>outerWidth(margin)</code>	Gets the outer width (includes the border and padding by default) for the first matched element. If the margin argument is true, then the margin values are also included

DEMO

