



CHAPTER 7

JQUERY



WHAT IS JQUERY?



jQuery is a script.

It is written in JavaScript.



Once included in your page, it
is used to:

Select elements

Do things with the elements you selected



1

1

Select elements
using CSS-style
selectors



```
jQuery( 'li.hot' );
```



```
jQuery( 'li.hot' );
```

FUNCTION



```
jQuery( 'li.hot' );
```

CSS-STYLE SELECTOR



SHORTHAND FOR JQUERY FUNCTION

```
$ ( 'li.hot' );
```



When you select an element or set of elements, it creates a **jQuery object**.



That object contains
references to the elements.



Like any object, the jQuery object has properties and methods. They allow you to work with those elements.



A jQuery object with selected elements can be called a:

jQuery selection
or
matched set



2

1

Select elements
using CSS-style
selectors

2

Do something
using methods of
the jQuery object



HIDE ALL MATCHING LIST ITEMS

```
$('li.hot').hide();
```

METHOD OF THE JQUERY OBJECT



With jQuery, you can:

- Select or find elements
- Update their content / size / visibility
- Simplify event handling



**STORING SELECTIONS
IN VARIABLES**



When a variable holds a jQuery object, its name often begins with a \$ symbol.



```
var $titles = $('.title');
```



```
var $titles = $('.title');
```

└──────────┘
VARIABLE NAME



```
var $titles = $('.title');
```

└──────────┘
CSS-STYLE SELECTOR



CSS-STYLE SELECTORS

This variable holds a jQuery object containing the element whose `id` attribute has a value of `author`:

```
var $author = $('#author');
```



CSS-STYLE SELECTORS

This variable holds a jQuery object containing the first item from the unordered list whose `id` attribute has a value of `list`:

```
var $first = $('ul#list:first-child');
```



GETTING & SETTING INFORMATION



GETTING HTML CONTENT

The `.html()` method gets the content of the selection (including markup).

It only retrieves content from the *first element* in the matched set.



GETTING HTML CONTENT

This example retrieves the content of the first list item.

```
$('li').html();
```



GETTING TEXT CONTENT

The `.text()` method gets the *text content only* of the selection (excluding markup).



GETTING TEXT CONTENT

This example retrieves the text content of the first list item.

```
$('li').text();
```



SETTING CONTENT

New content is added inside the parentheses after the method name.

It updates *all* of the elements in the matched set (not just the first). This is known as **implicit iteration**.



SETTING HTML CONTENT

This example will replace the content of each list item with the word **Updated** in `` tags.

```
$('li').html('<b>Updated</b>');
```



SETTING TEXT CONTENT

This example will replace the text content of each list item with the word **Updated**.

```
$('li').text('Updated');
```



CHAINING

It is possible to call multiple methods on the same selection.

```
$('li').hide().fadeIn(500);
```



CHAINING

It is possible to call multiple methods on the same selection.

```
$('li').hide().fadeIn(500);
```

└──────────┘
FIRST METHOD



CHAINING

It is possible to call multiple methods on the same selection.

```
$( 'li' ).hide() .fadeIn(500);
```

SECOND METHOD



CHECKING A PAGE IS READY TO WORK WITH



jQuery's `.ready()` method checks that the page is ready for your code to work with.



```
$(document).ready(function() {  
    // code goes here  
});
```



Creates a jQuery object
containing the entire page

JQUERY OBJECT

```
$(document).ready(function() {  
    // code goes here  
});
```



Checks to see if the page
has loaded before...

JQUERY METHOD

```
$(document).ready(function() {  
    // code goes here  
});
```



...running the code inside
the parentheses

CODE

```
$(document).ready(function() {  
    // code goes here  
});
```



This is a shortcut for writing
the `.ready()` method.



```
$(document).ready(function() {  
    // code goes here  
});
```

```
$(function() {  
    // code goes here  
});
```

MORE ABOUT UPDATING ELEMENTS

UPDATE ELEMENTS

```
.html()           .text()  
.replaceWith()    .remove()
```

```
$('li#one').remove();
```

INSERT ELEMENTS

`.before()` `.after()`

` item `

`.prepend()` `.append()`

A diagram illustrating jQuery insertion methods. In the center is the HTML element ` item `. Above it, `.before()` has a red downward-pointing triangle, and `.after()` has a red downward-pointing triangle. Below it, `.prepend()` has a red upward-pointing triangle, and `.append()` has a red upward-pointing triangle.

GET & SET ATTRIBUTES

`.attr()` `.removeAttr()`
`.addClass()` `.removeClass()`

```
$('#a#top').addClass('button');
```



UPDATE CSS

```
.css({  
  'font-family': 'Arial',  
  'color': '#ffffff',  
  'font-size': '+=2'  
});
```



EACH ELEMENT IN A SELECTION

```
$('#li').each(function() {  
  var ids = this.id;  
  $(this).append(' ' + ids);  
});
```



EVENTS



EVENT METHODS

```
$('#li').on('click', function() {  
    // code to happen on click  
});
```



EVENT METHODS

The `.on()` method handles events:

```
$('#li').on('click', function() {  
    // code to happen on click  
});
```



EVENT METHODS

The type of event is the first argument:

```
$('#li').on('click', function() {  
    // code to happen on click  
});
```



EVENT METHODS

When the event happens, an anonymous function is called:

```
$('#li').on('click', function() {  
    // code to happen on click  
});
```



EVENTS

focus, blur, change, input,
keydown, keyup, keypress,
click, dblclick, mouseup,
mousedown, mouseover,
mousemove, mouseout, hover,
submit, select, ready, load,
unload, error, resize, scroll



EVENT OBJECT

Passed into event handling function:

```
$('#li').on('click', function(e) {  
    var eventType = e.type;  
});
```



EFFECTS



ANIMATION

jQuery has methods that can:

- Show / hide a selection
- Fade in / out a selection
- Slide in / out a selection
- Animate numeric CSS properties



BASIC EFFECTS

```
.show()  
.hide()  
.toggle()
```



FADING EFFECTS

```
.fadeIn()  
.fadeOut()  
.fadeTo()  
.fadeToggle()
```



SLIDING EFFECTS

```
.slideUp()  
.slideDown()  
.slideToggle()
```



CUSTOM EFFECTS

```
.display()  
.stop()  
.animate()
```



ANIMATING CSS PROPERTIES

```
$( 'li' ).on( 'click', function() {  
  
    $( this ).animate( {  
        opacity: 0.0,  
        padding-left: '+=80',  
        500  
    } );  
  
});
```



TRAVERSING THE DOM

&

WORKING WITH SELECTIONS



DOM TRAVERSAL METHODS

<code>.find()</code>	<code>.closest()</code>
<code>.parent()</code>	<code>.children()</code>
<code>.parents()</code>	<code>.siblings()</code>
<code>.next()</code>	<code>.prev()</code>
<code>.nextAll()</code>	<code>.prevAll()</code>



ADD & FILTER ELEMENTS

```
.add()
```

```
.filter()
```

```
.find()
```

```
.not() / :not()
```

```
.has() / :has()
```

```
:contains
```

```
.is()
```



BOX DIMENSIONS



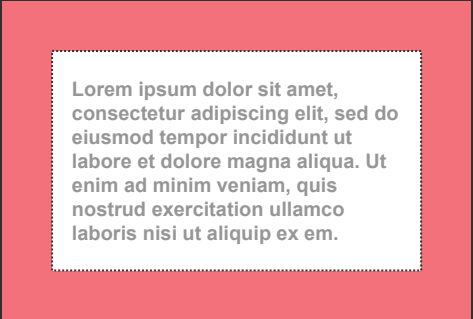
You can find the size of any box on the page or update its size.



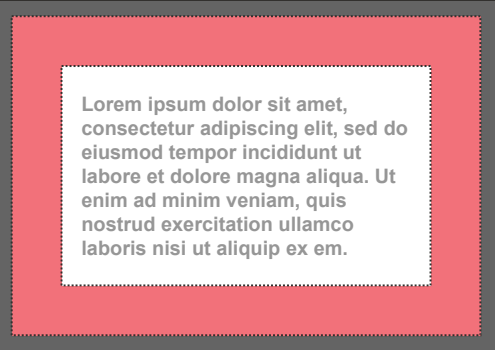
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex em.

● BOX

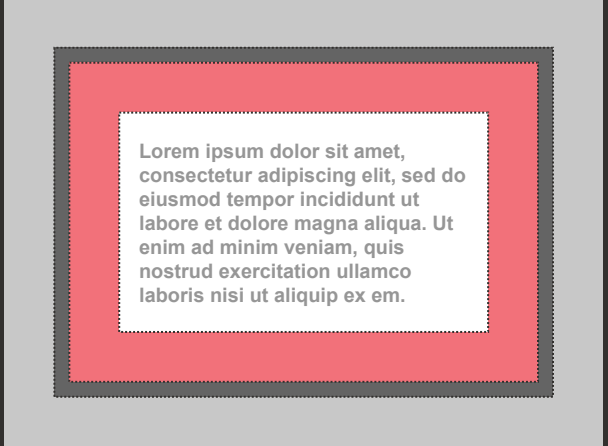




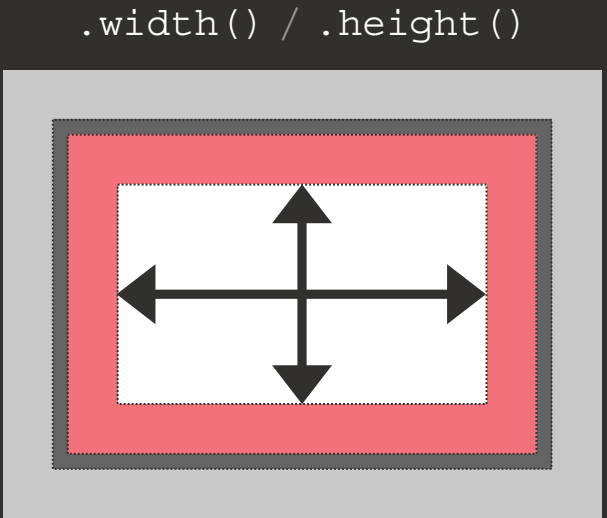
● BOX ● PADDING



● BOX ● PADDING ● BORDER



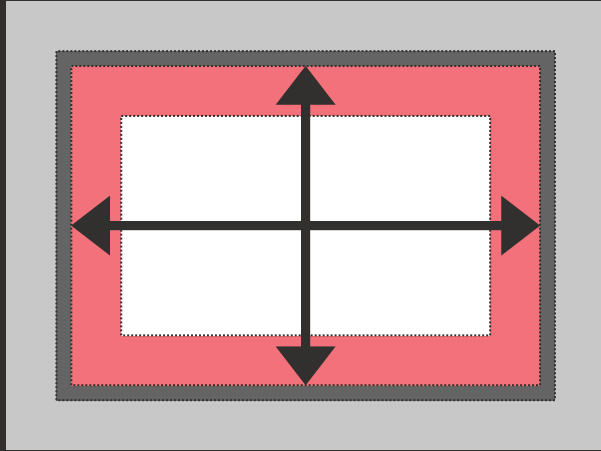
● BOX ● PADDING ● BORDER ● MARGIN



● BOX ● PADDING ● BORDER ● MARGIN



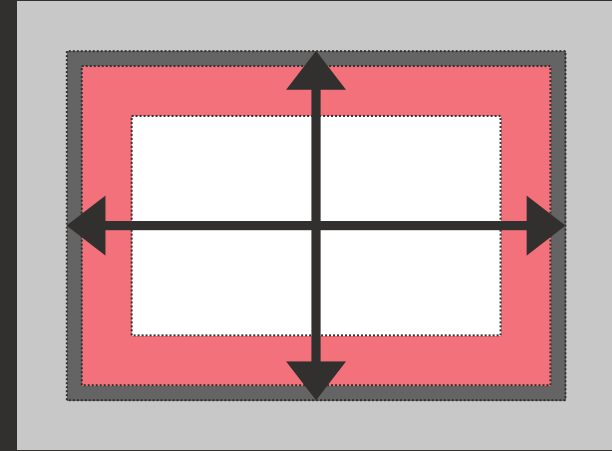
`.innerWidth() / .innerHeight()`



● BOX ● PADDING ● BORDER ● MARGIN



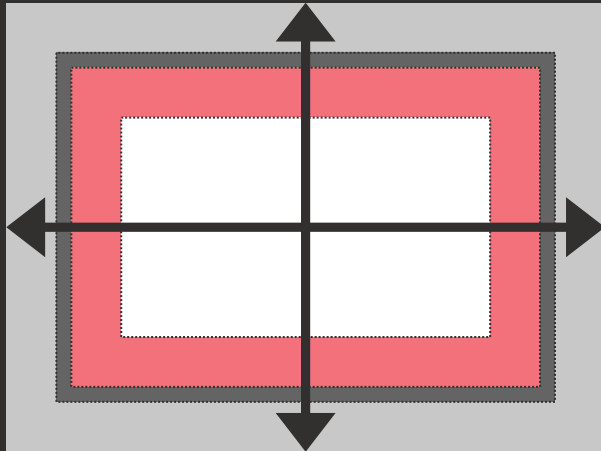
`.outerWidth() / .outerHeight()`



● BOX ● PADDING ● BORDER ● MARGIN



`.outerWidth(true) / .outerHeight(true)`



● BOX ● PADDING ● BORDER ● MARGIN



WINDOW & PAGE
DIMENSIONS



The `.height()` and `.width()` methods can be used to determine the dimensions of both the browser window and the HTML document.



```
1: $(document).height();  
2: $(window).height();
```



The `.scrollLeft()` and `.scrollTop()` methods allow you to get and set the position of the scrollbars.



FURTHER JQUERY



The jQuery documentation shows examples of every jQuery method.

<http://api.jquery.com>



jQuery plugins are often shared. They offer tools you may find helpful.

<http://plugins.jquery.com>

