

## Capítulo 4

# Trabajar con las hojas de estilo CSS

### 1. Introducción

¿Es necesario resaltar la importancia de las hojas de estilo en la escritura del código de las páginas web? Con jQuery, la modificación dinámica de las propiedades de estilo CSS se hace más fácil de implementar.

El método `css()` que estudiamos en los tres primeros puntos de este capítulo recuerda al método `attr()` del capítulo anterior.

### 2. Acceder a una propiedad de estilo

El método `css()` se usa de tres maneras diferentes. La primera solo permite acceder a la propiedad de estilo CSS de un elemento determinado.

#### **`css(nombre)`**

Este método permite acceder a la propiedad de estilo CSS del primer elemento encontrado.

El nombre es una cadena de caracteres con el nombre de la propiedad de estilo CSS a la que se va a acceder.

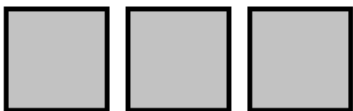
Este método devuelve una cadena de caracteres (*string*).

```
$("#p").css("color");
```

En este ejemplo, el método devuelve el valor de la propiedad CSS `color` del párrafo `<p>`.

### Ejemplo

Supongamos que tenemos una página con tres cajas `<div>` idénticas:



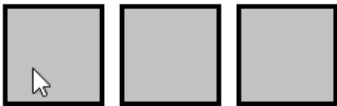
El documento HTML completo:

```
doctype html>
<html lang="es">
<head>
<meta charset="utf-8">
<title>jQuery</title>
<script src="jquery.js" ></script>
<script>
    $(document).ready(function(){
        $("#div1").click(function () {
            var color = $(this).css("background-color");
            $("#resultado").html("El color de fondo es <strong>" +
color + "</strong>.");
        });
        $("#div2").click(function () {
            var borde = $(this).css("border-color");
            $("#resultado").html("El color del borde es <strong>" +
borde + "</strong>.");
        });
        $("#div3").click(function () {
            var ancho = $(this).css("width");
            $("#resultado").html("El ancho es <strong>" + ancho
+ "</strong>.");
        });
    });
</script>
<style>
div {
```

```
        width: 60px;
        height: 60px;
        margin: 5px;
        float: left;
        background-color: #9cf;
        border: 3px solid black;
    }
    p {
        clear: both;
    }
</style>
</head>
<body>
<div id="div1">&nbsp;</div>
<div id="div2">&nbsp;</div>
<div id="div3">&nbsp;</div>
<p id="resultado">&nbsp;</p>
</body>
</html>
```

El archivo a descargar es el **C04-2.html**.

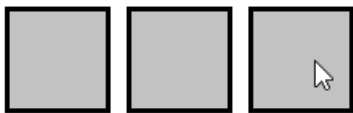
Al hacer clic en las tres cajas `<div>` el script mostrará respectivamente, de izquierda a derecha, el color de fondo, el color del borde y el ancho.



El color de fondo es **rgb(153, 204, 255)**.



El color del borde es **rgb(0,0,0)**.



El ancho es **60px**.

Detallemos el script jQuery:

```
$(document).ready(function() {
```

Tan pronto como se carga el DOM.

```
$("#div1").click(function () {  
    var color = $(this).css("background-color");
```

El clic del ratón en la primera caja (id="div1") carga en la variable color la propiedad del color de fondo de este elemento, usando css ("background-color").

```
$("#resultado").html("El color de fondo es <strong>" + color +  
"</strong>.");  
});
```

El valor de la propiedad de estilo se muestra como HTML en el elemento <p>, identificado por resultado.

```
$("#div2").click(function () {  
    var border = $(this).css("border-color");
```

El clic del ratón en la segunda caja (id="div2") carga en la variable borde el color del borde de este elemento, usando css ("border-color").

```
$("#resultado").html("El color del borde es <strong>" + borde +  
"</strong>.");  
});
```

El valor se mostrará en la página HTML.

```
$("#div3").click(function () {  
    var ancho = $(this).css("width");
```

El clic del ratón en la tercera caja (id="div3") carga en la variable ancho el ancho de este elemento, usando css ("width").

```
$("#resultado").html("El ancho es <strong>" + ancho +  
"</strong>.");  
});
```

El valor se mostrará en la página.

```
});
```

Fin del script.

### 3. Modificar las propiedades de estilo

La función `css()`, que admite parámetros, también permite modificar las propiedades de estilo de los elementos de la página.

#### **`css({propiedad de estilo})`**

Esta función modifica las propiedades de estilo CSS de un elemento concreto usando la notación CSS clave/valor para las propiedades de estilo a modificar.

```
$("#p").css({ color: "red", background: "blue" });
```

Observe la presencia de las etiquetas habituales para las declaraciones de estilo.

Si la clave contiene un guion de unión, como por ejemplo `background-color`, esta se debe ubicar entre comillas ("`background-color`").

También se puede adoptar la notación JavaScript (*CamelCase*), es decir `backgroundColor`, en lugar de `background-color`.

Este método devuelve un objeto jQuery.

Desde la versión 1.6 de jQuery, podemos usar valores relativos para modificar el valor de una propiedad de estilo CSS. Esto se puede codificar con `"+="` o `"-="`, con respecto al valor actual.

```
$("#item").css("left", "+=10px")
```

En este ejemplo desplazamos el elemento `item` 10 píxeles a la izquierda.

### Ejemplo

Vamos a ilustrar este método de jQuery con un ejemplo. Al pasar el ratón por encima de un párrafo, este tomará un color de fondo y el tipo de letra se mostrará en itálica.

Tras cargar la página:

Pasar el ratón por encima de este texto.

Ídem que para el texto anterior.

Al pasar por encima del primer párrafo:

*Pasar el ratón por encima de este texto.*

Ídem que para el texto anterior.

El documento HTML:

```
doctype html>
<html lang="es">
<head>
<meta charset="utf-8">
<title>jQuery</title>
<script src="jquery.js" ></script>
<script>
    $(document).ready(function(){
        $("p").mouseover(function () {
            $(this).css({'background-color': '#9cf', 'font-style':
'italic'});
        });
        $("p").mouseout(function () {
            $(this).css({'background-color': '', 'font-style': ''});
        });
    });
</script>
<style>
    p {
        font: Arial sans-serif;
    }
</style>
</head>
```