

Introducción a CSS | LMTW

Víctor Peinado

3-4 de mayo de 2016

Referencias

Hay numerosas referencias válidas en internet para aprender CSS. En esta presentación, vamos a basarnos en las dos siguientes:

- Tutorial de CSS en W3Schools¹
- Tutorial de CSS en HTML.net²
- Curso de HTML + CSS en Codecademy³

¹<http://www.w3schools.com/css/>

²<http://es.html.net/tutorials/css/>

³<http://www.codecademy.com/tracks/web>

Introducción a CSS

CSS (*Cascading Style Sheets*) es un lenguaje de estilo que define la presentación de los documentos HTML. CSS abarca cuestiones relativas a fuentes, colores, márgenes, líneas, altura, anchura, imágenes de fondo, posicionamiento avanzado y muchos otros temas

Como hemos visto, es posible usar HTML para añadir formato a los sitios web. Sin embargo, CSS ofrece más opciones y es más preciso y sofisticado.

Lo ideal actualmente es utilizar HTML para estructurar el contenido y CSS para formatear el contenido y configurar la vista.

Ventajas

Entre los beneficios de esta aproximación usando CSS, podemos mencionar:

- Control de la presentación de muchos documentos desde una única hoja de estilo.
- Control más preciso de la presentación.
- Aplicación de diferentes presentaciones a diferentes tipos de medios (pantalla, impresión, etc.),
- Numerosas técnicas avanzadas y sofisticadas.

Sintaxis de CSS

La sintaxis básica de CSS es:

```
elemento { propiedad: valor }
```

```
elemento {
  propiedad1: valor1;
  propiedad2: valor2;
}
```

Si te fijas en el ejemplo anterior, los elementos de la sintaxis de CSS son tres:

1. el nombre del elemento HTML para el que especificamos el formato.
2. entre llaves, especificamos el nombre de la propiedad que queremos modificar.
3. y separado con dos puntos :, el valor que asignamos a dicha propiedad.

Aplicando una CSS

Una vez tengamos definido nuestro formato de visualización en formato CSS, podemos aplicarlo a un documento HTML de tres maneras diferentes, a saber:

1. En línea, a través del atributo `style` dentro del propio código HTML:

```
<body style="background-color: #FF0000;">
```

El primero es apropiado si solo necesitamos modificar el formato de visualización de un elemento HTML.

2. Interno, con el elemento `style` dentro del `head` del documento HTML:

```
<head>
  <title>Ejemplo</title>
  <style type="text/css">
    body { background-color: #FF0000; }
  </style>
</head>
```

El segundo es suficiente si solo manejamos un documento HTML y la hoja de estilo es sencilla.

3. O bien de manera externa, en un fichero `.css` aparte y añadiendo un enlace con el elemento `link` que apunte a nuestra hoja de estilo, desde dentro del elemento `head` del documento HTML:

```
<link rel="stylesheet" type="text/css" href="style/style.css" />
```

El tercero es la solución idónea si la hoja de estilo es grande o si necesitamos aplicarla a más de un documento HTML.

Nuestra primera prueba

Por un lado, tenemos el siguiente documento `index.html`:

```
<html>
  <head>
    <title>Mi documento</title>
    <!-- enlazamos la hoja CSS -->
    <link rel="stylesheet" type="text/css" href="style.css" />
  </head>
  <body>
    <h1>Mi primera hoja de estilo</h1>
  </body>
</html>
```

Por otro lado, creamos un el documento `style.css` que contiene la hoja de estilo CSS:

```
body {
  background-color: #FF0000;
}
```

Propiedades de formato

Veamos cómo modificar otros elementos:

- Colores y fondos.⁴
- Tipos de fuente.⁵
- Propiedades del texto.⁶
- Enlaces.⁷

⁴<http://es.html.net/tutorials/css/lesson3.php>

⁵<http://es.html.net/tutorials/css/lesson4.php>

⁶<http://es.html.net/tutorials/css/lesson5.php>

⁷<http://es.html.net/tutorials/css/lesson6.php>

Agrupación de elementos

El atributo `class`

A veces resulta interesante definir distintas *clases* de elementos HTML para aplicar a cada clase un estilo diferente. Esto lo podemos conseguir a través del atributo `class`⁸.

Imagina el siguiente código de ejemplo:

```
<p>Mi lista de la compra</p>
<ul>
  <li class="red">pimiento rojo</li>
  <li class="red">tomate</li>
```

⁸<http://es.html.net/tutorials/css/lesson7.php>

```
<li class="yellow">pimiento amarillo</li>
<li class="yellow">limón</li>
<li class="green">pimiento verde</li>
<li class="green">judías verdes</li>
</ul>
```

Y el siguiente código en la hoja de estilo CSS:

```
li { color: black; }

li.red { color: red; }

li.yellow { color: yellow; }

li.green { color: green; }
```

El atributo *id*

El atributo `id`⁹ tiene una funcionalidad similar, con la única diferencia que mientras `class` permite definir estilos para *clases de elementos*, con `id` podemos definir las propiedad de un único elemento.

⁹ <http://es.html.net/tutorials/css/lesson7.php>

```
<p>Mi lista de la compra</p>
<ul>
  <li id="red">pimiento rojo</li>
  <li id="yellow">pimiento amarillo</li>
  <li id="green">pimiento verde</li>
</ul>
```

Y el siguiente código en la hoja de estilo CSS:

```
#red { color: red; }

#yellow { color: yellow; }

#green { color: green; }
```

Los elementos *span* y *div*

Los elementos `span` y `div` se usan para agrupar y estructurar un documento, y se usarán, a menudo, junto con los atributos `class` e `id`. Veamos cómo funcionan:

El elemento `span`¹⁰ es un elemento neutro que no añade nada al documento en sí. Pero podemos utilizarlo para añadir características visuales distintivas a partes específicas de texto en los documentos.

¹⁰ <http://es.html.net/tutorials/css/lesson8.php>

En el documento HTML, incluimos un elemento `span` con una clase concreta y etiquetamos determinada porción de texto.

```
<p>Esto es un ejemplo de
<span class="negrita">palabras en negrita</span>.</p>
```

Y en la hoja de estilo, podemos definir las características visuales que queramos para esa porción de texto.

```
span.negrita {
  color: red;
  font-weight: bold;
}
```

El elemento `div`¹¹ es también un elemento neutro. Mientras que `span` se utiliza para marcar porciones de texto dentro de un elemento, `div` se utiliza para etiquetar porciones de documento HTML que incluyan distintos elementos.

¹¹<http://es.html.net/tutorials/css/lesson8.php>

En el documento HTML, incluimos un elemento `div` con una clase concreta y etiquetamos determinada porción del documento, elementos incluidos.

```
<div id="democrats">
  <ul>
    <li>Franklin D. Roosevelt</li>
    <li>Bill Clinton</li>
    <li>Barack Obama</li>
  </ul>
</div>

<div id="republicans">
  <ul>
    <li>Dwight D. Eisenhower</li>
    <li>Richard Nixon</li>
    <li>George W. Bush</li>
  </ul>
</div>
```

Y en la hoja de estilo, podemos definir las características visuales que queramos para esa porción de texto.

```
#democrats {
  background-color: blue;
}

#republicans {
  background-color: red;
}
```