



APRENDERAPROGRAMAR.COM

¿QUÉ ES Y PARA QUÉ SIRVE EL LENGUAJE CSS (CASCADING STYLE SHEETS – HOJAS DE ESTILOS EN CASCADA)? (DV00203A)

Sección: Divulgación

Categoría: Lenguajes y entorno

Fecha revisión: 2029

Resumen: ¿Qué es CSS? ¿Para qué se utiliza? Veremos ejemplos sencillos y prácticos de este lenguaje para entender su utilidad dentro de internet en el diseño y programación de páginas web.

Autor: Manuel Sierra

DEFINICIÓN O CONCEPTO DE LENGUAJE CSS

CSS es un lenguaje utilizado en la presentación de documentos HTML. Un documento HTML viene siendo coloquialmente “una página web”. Entonces podemos decir que el lenguaje CSS sirve para organizar la presentación y aspecto de una página web. Este lenguaje es principalmente utilizado por parte de los navegadores web de internet y por los programadores web informáticos para elegir multitud de opciones de presentación como colores, tipos y tamaños de letra, etc.



La filosofía de CSS se basa en intentar separar lo que es la estructura del documento HTML de su presentación. Por decirlo de alguna manera: la página web sería lo que hay debajo (el contenido) y CSS sería un cristal de color que hace que el contenido se vea de una forma u otra. Usando esta filosofía, resulta muy fácil cambiarle el aspecto a una página web: basta con cambiar “el cristal” que tiene delante. Piensa por ejemplo qué ocurre si tienes un libro de papel y lo miras a través de un cristal de color azul: que ves el libro azul. En cambio, si lo miras a través de un cristal amarillo, verás el libro amarillo. El libro (el contenido) es el mismo, pero lo puedes ver de distintas maneras.

Algunas opciones básicas del lenguaje CSS por ejemplo pueden ser el poder cambiar el color de algunas típicas etiquetas HTML como `<H1>` (h1 es una etiqueta en el lenguaje HTML destinada a mostrar un texto como encabezado, en tamaño grande). Pero también hay funciones algo más complejas, como introducir espaciado entre elementos `<DIV>` (div es una etiqueta HTML para identificar una determinada región o división de contenido dentro de una página web) o establecer imágenes de fondo.

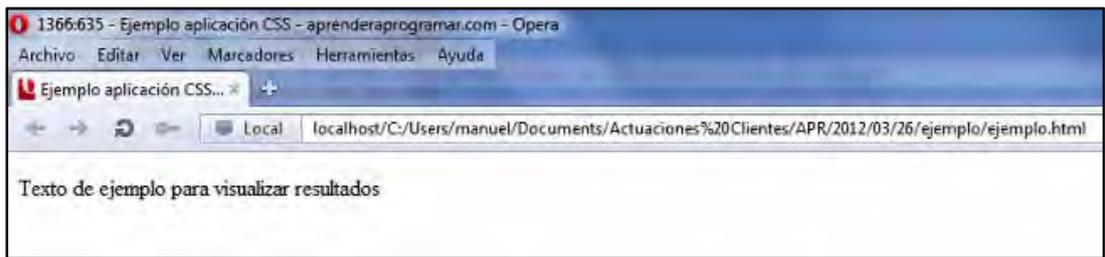
CSS es muy intuitivo y sencillo una vez se llega a aprender, ya que para su definición siempre se hace uso de un identificador de etiqueta HTML (como por ejemplo `<H1>`), y luego indicamos con qué aspecto queremos que se muestren todas las etiquetas `<H1>` que aparezcan en un documento. Al igual que con `<H1>` podemos definir cómo queremos que se muestren las distintas partes del documento HTML, pudiendo en cada caso definir sus propiedades (color, tipo de fuente, tamaño, espacio, imagen) con algún determinado valor deseado.

Vamos a partir de un ejemplo muy sencillo, que tratará de una página web o archivo HTML donde tan solo tendremos un párrafo de texto y sobre cuya etiqueta `<p>` iremos realizando cambios e iremos aplicando los conocimientos que vamos a ir adquiriendo y posteriormente veremos los resultados que obtenemos.

Nuestro documento html contendrá el siguiente texto de partida (en este caso se ha llamado ejemplo.html):

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <title>Ejemplo aplicación CSS - aprenderaprogramar.com</title>
  </head>
  <body>
    <p>Texto de ejemplo para visualizar resultados </p>
  </body>
</html>
```

Una vez creado el documento ejemplo.html en nuestro ordenador, lo abriremos con un navegador web, cuyo resultado será similar a éste dependiendo del navegador que utilicemos:



Una vez hemos visualizado este ejemplo tenemos una página web que tan solo tiene un párrafo ("Texto de ejemplo para visualizar resultados"). Este aparece en color negro por defecto y nosotros, para ver la utilidad de CSS deseamos mostrar el texto en rojo. En realidad con CSS podemos hacer cosas mucho más complejas, pero solo queremos poner un ejemplo para mostrar la utilidad de CSS.

Para ello vamos a crear nuestro archivo de hojas de estilo CSS que llamaremos estilos.css, que crearemos en el mismo directorio donde tengamos el archivo ejemplo.html y que contendrá lo siguiente:

```
/* Comentario en nuestro primer archivo css estilos.css - www.aprenderaprogramar.com */
p {color:red;}
```

Para crear el archivo basta abrir un editor de texto como el bloc de notas como Notepad++ o el simple bloc de notas de Windows, escribir el texto y guardar el archivo con el nombre adecuado.

En la primera línea hemos insertado un comentario CSS. Para insertar un comentario usamos la cadena de caracteres "/*" seguido del comentario que queremos y cerramos la cadena con "*/".

La segunda línea es propiamente donde definimos el estilo que queremos mostrar. En este caso, `p{color:red;}` que indica que queremos un estilo para la etiqueta `<p>` cuya propiedad `color` sea igual a `red` (rojo en inglés).

El patrón será el siguiente: nombre etiqueta HTML (en este caso `p` para la etiqueta `<p>` de HTML) seguido de `{`, seguido de una lista de propiedades-valores separadas por `;` y finalizamos con `}`. A su vez la lista de propiedades-valores será un conjunto de palabras que indicará la propiedad que queremos cambiar, en nuestro caso `color`, seguido de `:` y el valor que queremos asignar para esa propiedad, en este caso `red` ya que queremos mostrar el texto en rojo. Además de color podríamos establecer tipo de fuente, tamaño de fuente, y distintas propiedades.

Pues bien una vez tenemos nuestro archivo de estilos, `estilos.css`, debemos de importarlo para su uso a nuestra página web o documento html, `ejemplo.html`. Para ello ponemos lo siguiente entre las etiquetas `<head>` y `</head>` de nuestro `ejemplo.html`.

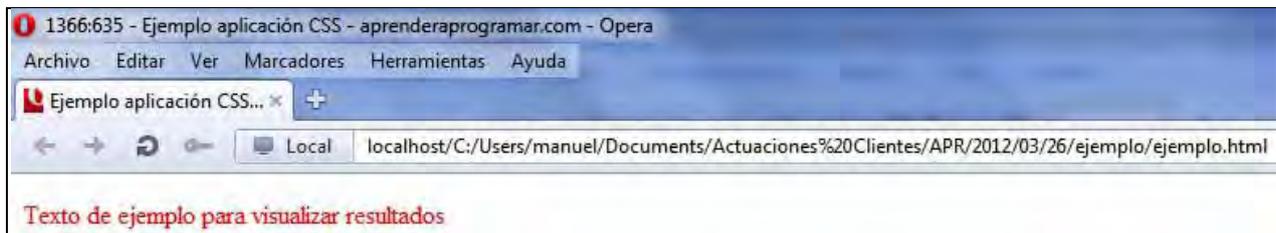
```
<link rel="stylesheet" type="text/css" href="estilos.css">
```

Este código lo que hace es indicar que existe un archivo con una definición de estilos u hoja de estilos (stylesheet) que contiene código css (text/css) y que se encuentra en la ruta `estilos.css`. Si el fichero se encontrara en otro directorio habría que especificar la ruta, por ejemplo : `templates/estilos.css`.

Así tendremos nuestro archivo `ejemplo.html` con el siguiente texto:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <title>Ejemplo aplicación CSS - aprenderaprogramar.com</title>
    <link rel="stylesheet" type="text/css" href="estilos.css">
  </head>
  <body>
    <p>Texto de ejemplo para visualizar resultados </p>
  </body>
</html>
```

Y una vez que guardamos el archivo html con este nuevo código, recargamos la página web en el navegador obteniendo el texto de la etiqueta `<p>` en rojo:



Aquí podemos ver la facilidad de CSS para cambiar la presentación ya que si ahora quisiéramos cambiar el color, fuente, tamaño, etc, de la etiqueta `<p>` para nuestra página web, tan solo deberíamos de cambiar las propiedades en el fichero `estilos.css` sin alterar nada en la página web siendo más fácil, rápido y legible que tener que estar modificando todas y cada una de las etiquetas `<p>` que aparecieran en nuestra página web.

Ahora bien, imaginemos que tenemos una página web con 3 etiquetas `<p>` pero no deseamos que todas tengan la misma presentación. Pues para estas situaciones CSS incorpora los llamados selectores. Estos selectores, nos indican para qué elementos debemos de adoptar los estilos definidos en ellos.

Principalmente hay 2 tipos de selectores, el punto y la almohadilla:

- .
- #

El selector `.` afecta a todas las etiquetas HTML que tengan un atributo en HTML del tipo `class`. Por ejemplo para `<p class="azul">` podríamos definir un estilo CSS del tipo `.azul{color:blue;}`. Escribiendo esto en el archivo `css` lo que hacemos es definir que cada vez que se indique que una etiqueta es de clase "azul" la propiedad de estilo `color` será establecida a `blue` (blue es azul en inglés). Así si escribiéramos `<p class="azul">Texto de ejemplo para visualizar resultados </p>`, el navegador busca en el `css` cómo se ha definido que deben ser todos los elementos en los que se indique que usen la clase "azul". Esta clase además de un color podría establecer tamaño de texto, interlineado, tipo de fuente y múltiples atributos.

El otro selector más utilizado es `#`. Este afecta a todas las etiquetas HTML que tengan un atributo en HTML del tipo `id`. Por ejemplo `<p id="verde"> Este texto está identificado </p>` significa que esta etiqueta de HTML lleva un "identificador" o nombre único en esa página web, en este caso "verde". Podríamos definir un estilo CSS del tipo `#verde{color:green;}`. Con este código hemos definido un estilo para la etiqueta con `id verde`, de modo que la propiedad `color` tomará el valor `green` (green es verde en inglés). Es decir, el elemento de la página web que lleve el identificador indicado tomará las propiedades de estilo que se hayan especificado. El `id` es único dentro de una página web: no puede haber dos etiquetas cuyo `id` sea el mismo.

Normalmente el selector `"#"` se utiliza para un solo elemento (por ejemplo, el encabezado o el pie) en toda la página web identificado por su atributo `id`, mientras que el selector `"."` también llamado selector de clase es utilizado para un grupo más amplio de elementos en concreto todos los que tengan sus atributos `class` iguales que los definidos por el selector `"."`.

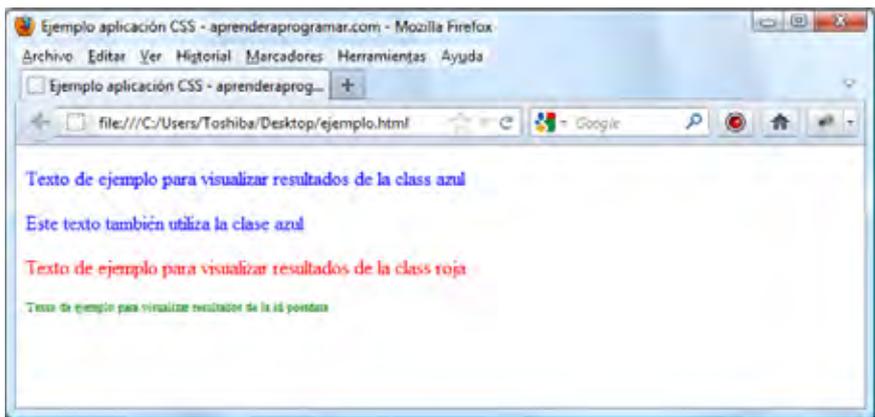
Vamos ahora a realizar algunos cambios en nuestros documentos, tanto `ejemplo.html` como `sobre estilos.css` y veremos más claro lo anteriormente explicado.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <title>Ejemplo aplicación CSS - aprenderaprogramar.com</title>
    <link rel="stylesheet" type="text/css" href="estilos.css">
  </head>
  <body>
    <p class="azul">Texto de ejemplo para visualizar resultados de la class azul </p>
    <p class="azul">Este texto también utiliza la clase azul </p>
    <p class="roja">Texto de ejemplo para visualizar resultados de la class roja </p>
    <p id="postdata">Texto de ejemplo para visualizar resultados de la id postdata </p>
  </body>
</html>
```

También deberemos de cambiar nuestro fichero estilos.css para definir las propiedades de las etiquetas correspondientes, quedando así:

```
/* Comentario en nuestro primer archivo css estilos.css – www.aprenderaprogramar.com */
.azul{color:blue;}
.roja{color:red;}
#postdata{color:green;font-size:10px;}
```

Y al refrescar nuestro navegador obtendríamos el siguiente resultado:



Como podemos observar hemos modificado nuestro documento html para que tuviera 4 párrafos. Los 2 primeros pertenecen a la misma clase *azul* ya que su atributo *class* se corresponde con el estilo *azul* que se ha definido en la hoja de estilos (en el archivo estilos.css).

El tercero pertenece a la clase "roja", mientras que el cuarto es un párrafo con *id postdata* y por tanto pertenece al estilo del selector único *postdata*.

AMBITO DE DEFINICION PARA CSS

La forma de definición de estilos css que hemos explicado en este artículo es la que más separa la presentación de la información en una página web. Es muy útil porque tenemos los estilos por un lado y los contenidos por otro. Si en un momento dado queremos cambiar la forma en que se ve la página pero no sus contenidos, únicamente tendríamos que modificar los archivos css. Pero también existen estas otras 2 formas de aplicar estilos.

- 1- **Definición en línea:** En este método se define las propiedades de la etiqueta HTML con el atributo *style* dentro del mismo archivo HTML y se le asigna una determinada cadena que definiría el comportamiento. Ejemplo:

```
<p style="color:red;">
```

En este ejemplo asignamos en línea a la etiqueta *p* el *style* (style es estilo en inglés) definido, en el que la propiedad *color* toma el valor *red* de modo que el texto se mostrará en rojo. Esta opción de definición, es rápida y para pequeñas cosas o reparaciones de mantenimiento web puede ser útil, pero si se abusa de esta técnica tendremos una página web muy poco legible ya que tendremos en un mismo archivo el contenido de la página web junto con su presentación. Hacer cambios en la presentación se hace entonces más complicado, al tener que ir revisando todas las etiquetas que queramos cambiar una por una.

- 2- **Definición hoja de estilos interna:** En este método se definen las propiedades de estilo de la misma manera que hacíamos en el archivo css independiente, pero en este caso se hace en la misma página entre las etiquetas `<head>` y `</head>`, como vemos en el siguiente ejemplo.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <title>Ejemplo aplicación CSS - aprenderaprogramar.com</title>
    <style type="text/css">
      p{color:red;}
    </style>
  </head>
  <body>
    <p>Texto ejemplo </p>
  </body>
</html>
```

Como podemos observar definimos el estilo dentro de las etiquetas <head> y </head>. Para delimitar lo que es definición de estilos se usa la etiqueta <style> y </style> a modo de apertura y cierre de la definición de estilos. El código se escribe igual que en el archivo css y aunque no es igual de independiente, esta forma de definición al menos es algo más separada y legible que la anterior opción ya que la definición de la presentación al menos no se mezcla con el contenido de la página web.

ESTANDARIZACIÓN

Tras la aparición del lenguaje CSS surgió la necesidad de estandarizar su uso para todos los navegadores y los desarrolladores de software, ya que existían muchas formas de escribir código CSS. Además, los navegadores interpretaban algunas definiciones de estilo de distintas maneras y esto hacía más complicado el desarrollo de páginas web. De hecho, las personas que trabajan en desarrollos web muchas veces se enfrentan a que un navegador muestra una página web de distinta manera aunque el código sea el mismo, simplemente por la forma en que el navegador interpreta el código. El organismo encargado de la estandarización al respecto es el llamado [W3C](#) que definió la primera versión CSS1 en 1996. Posteriormente se han desarrollado las revisiones 2, 2.1 y 3 que es la más actual. El lenguaje CSS seguirá evolucionando, pero hoy día puede decirse que ha sido un éxito al simplificar el mantenimiento de páginas web. Antes, los contenidos y los estilos de presentación estaban mezclados. Con CSS, quedan separados y se hace más fácil el diseño y mantenimiento de páginas web.

SABER MÁS. APRENDER CSS

Quienes quieran trabajar como programadores, maquetadores o diseñadores web deben conocer CSS pues es un lenguaje imprescindible para este tipo de actividad profesional. Para formarte en CSS dispones del curso "Tutorial básico del programador web: CSS desde cero" de [aprenderaprogramar.com](#), al que puedes acceder en:

http://aprenderaprogramar.com/index.php?option=com_content&view=category&id=75&Itemid=203

Si tienes interés por otros cursos en el área de programación web o programación en general consulta la oferta formativa completa en:

http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=57&Itemid=86