

## 2 > Hojas de estilos CSS

Dado que en HTML no solo se indican los contenidos que se quieren visualizar, sino que se mezclan con las instrucciones de formato y estilo, el W3C creó un sistema mediante el cual las instrucciones del estilo y formato estuvieran separadas del contenido. Así nació CSS (*Cascading Style Sheets*, hojas de estilo en cascada.), un mecanismo para dotar de estilo a los elementos HTML, que desaconseja mezclar su contenido con la forma en que estos son presentados.

### 2.1 > Introducción

Por tanto, el objetivo fundamental de CSS es **separar el contenido y la estructura de un documento HTML, de su estilo y formato de presentación**. La versión recomendada era CSS 2.1, pero actualmente está en desarrollo el nuevo estándar CSS3, y el W3C recomienda el uso de los módulos CSS3 ya implementados.

Para entender el objetivo de CSS, pongamos un ejemplo: sabemos que el elemento de HTML `<h1>` indica que un bloque de texto es un encabezamiento y que es más importante (de mayor tamaño) que un bloque etiquetado como `<h2>`. Versiones más antiguas de HTML permitían atributos dentro de la etiqueta para añadir estilo extra (como el color o el tamaño de fuente). Pero cuando se utiliza CSS, la etiqueta `<h1>` no debería proporcionar información sobre cómo va a ser visualizado, solamente indicar la estructura del documento. La información de estilo, recogida por ejemplo en una hoja de estilos, debe especificar cómo se ha de mostrar `<h1>`, es decir, su color, fuente, alineación del texto, tamaño y otras características visuales.

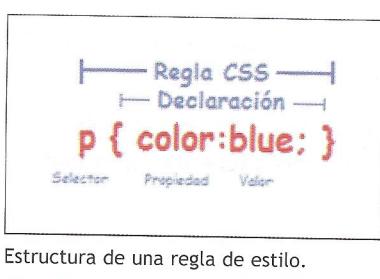
Por consiguiente, CSS es la herramienta con la que se debe dar formato y estilo a los elementos de HTML. La información de estilo puede ser recogida tanto en un documento separado del HTML, opción recomendada, como en el propio documento HTML. En este último caso pueden definirse estilos generales en la cabecera del documento o en cada etiqueta particular mediante el atributo `style`, como más adelante se explica.

### 2.2 > Hoja de estilos

Una hoja de estilos es un conjunto de especificaciones que definen el formato con el que presentar los elementos HTML de una página web.

Está compuesta por **reglas de estilo**, cada una de las cuales declara los formatos que adoptarán los elementos de una página web. Así, un conjunto de reglas de estilo conforman una hoja de estilos, información que normalmente se recoge en un fichero con extensión `.css`, aunque existen otras maneras de recoger las reglas que den estilo a los elementos que componen un documento HTML.

Una regla de estilo tiene dos partes: el **selector**, que determina a qué elementos se asignará el estilo definido en la regla, y la **declaración**, que está compuesta por uno o más pares (`propiedad:valor;`), que asignan un valor a la propiedad de los elementos seleccionados por el selector.



## 2.3 > Formas de aplicar estilo

CSS proporciona tres formas diferentes para aplicar las reglas de estilo a los elementos HTML de una página web:

### Estilo local o en línea

Este método permite insertar el estilo, directamente, dentro de cada etiqueta HTML, añadiendo en ellas el atributo común `style`.

Es la forma menos adecuada, dado que especificar el estilo dentro de cada etiqueta del documento HTML, es contrario al objetivo de CSS, y además se convierte en una tarea larga, tediosa y poco elegante de expresar el formato y estilo de una página web. Solo se recomienda usar este modo de trabajo, de manera excepcional, por ejemplo cuando se necesita aplicar un estilo o formato de manera urgente en una página.

### Hoja de estilos interna

Es una hoja de estilo que está incrustada (etiqueta `<style>`) dentro de un documento HTML. Suele ponerse dentro del elemento `<head>`, aunque puede ir en cualquier parte de la página. Esta forma permite agrupar toda la información sobre el estilo, pero dentro del mismo fichero HTML que contiene el contenido. Una desventaja es que hay que modificar todas las páginas que conforman un sitio cuando se quiera realizar un cambio que afecte a todas ellas; por lo tanto, será útil cuando se aplique estilo a una única página.

### Hoja de estilos externa

Es una hoja de estilo que está almacenada en un archivo diferente al archivo donde se almacena el código HTML de la página web. Esta es la mejor manera de aplicar estilo y la que sigue más fielmente el objetivo de CSS, porque separa completamente las reglas de estilo y formato para la página HTML, de su contenido y estructuración básica.

La etiqueta `<link>` se coloca en la cabecera `<head>` del documento y permite enlazarlo con otro recurso.

Se debe indicar el atributo `rel`, para especificar la relación entre el documento y el recurso.

El atributo opcional `type` especifica el tipo del documento a vincular, para una hoja de estilos “`text/css`”.

El atributo `href` debe indicar una URL, la del fichero CSS que tenga la definición del estilo.

Es recomendable utilizar hojas de estilo externas, pero sobre todo en sitios web que utilizan un mismo estilo para todas sus páginas, dado que una modificación de una regla de estilo afectará a todas ellas.

### Uso de hoja de estilo externa

#### `<link>`

```
<html>
<head>
...
<link rel="stylesheet" type="text/css" href="hojaEstilo.css"/>
...
</head>
<body>
...
</body>
</html>
```

### Comentarios

Cuando se crea una hoja de estilos (interna o externa), para facilitar la revisión del código CSS es conveniente añadir comentarios entre las reglas de estilo. Para incluir un comentario es necesario incluir el texto a comentar entre los caracteres `/*` y `*/`, por ejemplo:

```
/* Esta es la forma de incluir
un comentario válido entre las
reglas de estilo css */
```

## Actividades propuestas

- 4.. Indica una etiqueta HTML que permita vincular la hoja de estilos del fichero “miHoja.css”.

## 2.4 > Conceptos fundamentales

Los conceptos fundamentales que permiten realizar hojas de estilos en CSS son:

### Identificadores

Todos los elementos HTML tienen un atributo opcional llamado identificador (*id*), cuya finalidad es poder identificar a cada uno de los diferentes elementos que componen la página web. Con estos identificadores podemos aplicar un estilo exclusivo a cada etiqueta, según se seleccione su identificador en el selector de una regla CSS.

Para identificar un determinado elemento en el código HTML, es necesario darle un nombre por medio del atributo *id*, y, posteriormente, será necesario acceder a este elemento desde una regla CSS para aplicarle el diseño. Esto se hace a través del selector, que se formará con el nombre de la etiqueta seguido del símbolo # y del nombre del atributo *id*.

### Clases

Existe otro atributo que se puede utilizar en todos los elementos HTML, el atributo *class*. Su utilidad es la de poder agrupar los elementos por clases o grupos para que, posteriormente, CSS los pueda seleccionar y distinguirlos de los demás. El uso de *class* también es opcional, pero, como el atributo *id*, es muy recomendable y permite precisar selectores.

### Alto y ancho de objetos CSS

Las propiedades generales *height* y *width* permiten establecer el alto y el ancho de los elementos.

```
#tamaño100
{height:100px;
 width:100px;}
```

## Ejemplos

### Aplicar estilo de manera local, con una hoja de estilos interna y a través de clases

Este es un ejemplo de código HTML que utiliza una hoja de estilo local al párrafo, haciendo uso del atributo *style*:

```
<html> <body> <p style="color:green"> HOLA MUNDO </p>
</body> </html>
```

HOLA MUNDO

Este ejemplo utiliza una hoja de estilos interna, a través de la etiqueta *<style>*:

```
<html> <head> <title>Mi primera página con estilo</title>
      <style type="text/css">
        body {color: purple;
              background-color: #d8da3d}</style>
      </head> <body>
```

HOLA MUNDO

En este otro ejemplo utiliza **clases** para aplicar estilo, a través del atributo *class*; su sintaxis en HTML sería:

```
<p class="bordeRojo">HOLA</p>
```

Un elemento puede pertenecer a varias clases a la vez, ya que estas no son excluyentes. Para ello se especifica dentro del atributo *class* los nombres de las clases separados por espacios:

```
<p class="letraAzul bordeAzul">HOLA</p>
```

Si un elemento HTML ha sido identificado siguiendo este ejemplo, puedes darle formato con reglas de estilo así:

```
.bordeRojo {
  border:solid 1px #000000;
  border-color:red; }

.bordeAzul {
  border:solid 1px #000000;
  border-color:blue; }

.letraAzul {color:blue;}
```

## Casos prácticos

4

### Creación de estilo usando identificadores

- Crea un estilo, poniendo en negrita el primer párrafo, utilizando el atributo *id* dentro del elemento *<p>*.

**Solución** • Para probar cómo dar estilo a elementos, abre un editor de texto (Bloc de notas, Gedit o similar) y edita dos documentos, uno que incluya las líneas de "Código HTML", y el otro con el "Código CSS" indicado. Hay que identificar con un nombre a la etiqueta *<p>* del primer párrafo y añadir un identificador al selector del código CSS.

#### Código HTML:

```
<link rel="stylesheet"
      type="text/css"
      href="negrita.css">...
<p id="EnNegrita">
    Párrafo 1</p>
<p>Párrafo 2</p>
<p>Párrafo 3</p>...
```

#### Código CSS:

```
p#EnNegrita {font-weight:bold;}
```

Párrafo 1  
Párrafo 2  
Párrafo 3

Comprueba que incluyes el atributo *id* en el código HTML y utiliza su valor en el código CSS, incluyéndolo en el selector. Al ser un identificador, tienes que anteponerle la almohadilla al indicarlo en la regla de estilo. Así se puede dar un diseño exclusivo a cualquier elemento de una página web, tablas, listas, etc.; en este caso práctico se hace solo sobre el primer párrafo. Crea identificadores para los otros párrafos y aplícales reglas de estilo CSS.

## Pseudo-classes

Las pseudo-classes permiten clasificar los elementos HTML según el estado en el que se encuentren, es decir, los elementos serán seleccionados por una regla de estilo CSS en función de otros sucesos que hayan ocurrido, por ejemplo haber visitado un vínculo, o que el cursor del ratón se sitúe sobre un elemento, o en función del idioma especificado con el atributo común *lang*, en el elemento HTML.

El elemento *<a>* de HTML, utilizado para crear hiperenlaces, tiene definidas las siguientes pseudo-classes:

<pre>a:visited {color:blue;} a:hover {color:blue;} a:link {color:blue;} a:focus {color:blue;} a:active {color:blue;}</pre>	<p>Vínculo visitado Vínculo donde está el ratón Vínculo no visitado Vínculo seleccionado por teclado Vínculo que recibe clic</p>
--	--

La pseudo-clase para dar estilo a párrafos según el idioma sería así:

```
p:lang(en){color:blue;}
```

Y la pseudo-clase para asignar un estilo al primer hijo de un elemento:

```
p:first-child {color:blue;}
```

### Selectores CSS3

Una de las diferencias en el uso de los selectores CSS3 es utilizar el símbolo *(::)*, doble dos puntos, en lugar de los dos puntos *(:)* que se usaba para los pseudo-elementos en CSS 2.1. Algunos ejemplos son:

```
p::first-letter
      {color:blue}
```

Aplica la propiedad a la primera letra de cada párrafo.

```
p::first-line {color:blue}
```

Aplica la propiedad a la primera línea de cada párrafo.

## Pseudo-elementos

Los pseudo-elementos son una parte de algún elemento HTML, que, aunque no tienen identidad propia, pueden ser identificados por las reglas de estilos y ser formateados de manera distinta a los elementos a los que pertenecen; algunos son: `::first-letter` y `::first-line`.

## 2.5 > Criterios de selección, selectores

Con los criterios de selección podemos indicar cuál es el selector de una regla de estilo CSS y así expresar a qué elementos de HTML se le aplicará el valor sobre la propiedad indicada en dicha regla CSS. Los diferentes criterios establecen, por lo tanto, distintos tipos de selectores:

### Selector universal

Si se utiliza el asterisco (\*) como selector, las propiedades especificadas afectarán a todos los elementos de la página; el asterisco es el selector universal. Para declarar el color azul a todos los elementos sería:

```
* {color:blue}
```

### Selector de tipo o nombre de elemento

Permite aplicar un determinado estilo a todos los elementos del tipo seleccionado, con independencia de posibles clases, identificadores, etc. Para poder afinar más la selección se deben utilizar otros criterios de selección. Un ejemplo que declara el color azul a todos los párrafos sería:

```
p {color:blue}
```

### Selector por combinación de elementos

Este criterio permite seleccionar en función de los elementos ascendientes/descendientes, padres/hijos o hermanos del elemento al cual se quiere dar un estilo. Estos elementos se pueden definir como:

- **Ascendientes/Descendientes:** ascendiente es el elemento que contiene al que se desea formatear, el descendiente. Puede haber más de un elemento entre uno y otro. Si B es un descendiente arbitrario de algún elemento antepasado A, entonces A se considera ascendiente de B. Para seleccionar elementos en función del ascendiente, se usan dos selectores separados por un espacio ( ), siendo el primero de ellos el ascendiente y el segundo el descendiente. Por ejemplo, si se quiere formatear de color rojo los elementos `p` que estén dentro de un elemento `div`, se debe escribir la siguiente regla:

```
div p {color : #00FF00}
```

- **Padres/Hijos:** padre es el elemento que contiene directamente al que se desea formatear, al que llamamos hijo. Para seleccionar elementos en función del parente, se usan dos selectores separados por el símbolo “mayor que” (>), siendo el primero el parente y el segundo el hijo. Por ejemplo, con la siguiente declaración solo se verían afectados los elementos `p` que estuvieran directamente dentro de un elemento `div`, es decir, los elementos `p` que son hijos de `div`, los cuales serán seleccionados para ponerles el color verde; sería así:

```
div > p {color : #00FF00;}
```