



ELEARNING TOTAL

Programador Web / Nivel 1 – Unidad 3

Programador Web – Nivel 1

Unidad 3: Estilos CSS





Indice

Unidad 3: Estilos CSS

Texto

Enlaces

Listas

Tablas

Fondos



Objetivos

Que el alumno logre:

- Conocer el lenguaje de hojas de estilo y su implementación en el código HTML.



Texto

CSS define numerosas propiedades para modificar la apariencia del texto. A pesar de que no dispone de tantas posibilidades como los lenguajes y programas específicos para crear documentos impresos, CSS permite aplicar estilos complejos y muy variados al texto de las páginas web.

La propiedad básica que define CSS relacionada con la tipografía se denomina color y se utiliza para establecer el color de la letra.

Atributos:

.: color: Establece el color de letra utilizado para el texto Ej: `h1 { color: #B1251E; }`

.: font-family: Establece el tipo de letra utilizado para el texto

El tipo de letra del texto se puede indicar de dos formas diferentes:

.: Mediante el nombre de una familia tipográfica: en otras palabras, mediante el nombre del tipo de letra, como por ejemplo “Arial”, “Verdana”, “Garamond”, etc.

.: Mediante el nombre genérico de una familia tipográfica: los nombres genéricos no se refieren a ninguna fuente en concreto, sino que hacen referencia al estilo del tipo de letra. Las familias genéricas definidas son serif (tipo de letra similar a Times New Roman), sans-serif (tipo Arial), cursive (tipo Comic Sans), fantasy (tipo Impact) y monospace (tipo Courier New).

Ej: `p {font-family: Arial, Helvetica, sans-serif};`

.: font-size: Establece el tamaño de letra utilizado para el texto

Además de todas las unidades de medida relativas y absolutas y el uso de porcentajes, CSS permite utilizar una serie de palabras clave para indicar el tamaño de letra del texto:



.: tamaño_absoluto: indica el tamaño de letra de forma absoluta mediante alguna de las siguientes palabras clave: xx-small, x-small, small, medium, large, x-large, xx-large.

.: tamaño_relativo: indica de forma relativa el tamaño de letra del texto mediante dos palabras clave (larger, smaller) que toman como referencia el tamaño de letra del elemento padre.

Ej: p {font-size: medium}

.: font-style: Establece el estilo de la letra utilizada para el texto

Valores: normal | italic | oblique | inherit

Ej: h1 {font-style: italic}

Apariencia

Además de las propiedades relativas a la tipografía del texto, CSS define numerosas propiedades que determinan la apariencia del texto en su conjunto. Estas propiedades adicionales permiten controlar al alineación del texto, el interlineado, la separación entre palabras, etc.

Atributos:

.: text-align: Establece la alineación del contenido del elemento

Valores: left | right | center | justify | inherit

Ej: p {text-align: center}

.: line-height: Permite establecer la altura de línea de los elementos

Valores: normal | <numero> | <medida> | <porcentaje> | inherit

Ej: p { line-height: 120%}



.: text-decoration: Establece la decoración del texto (subrayado, tachado, parpadeante, etc.)

Valores: none | (underline || overline || line-through || blink) | inherit

.: text-transform: Transforma el texto original (lo transforma a mayúsculas, a minúsculas, etc.)

Valores: capitalize | uppercase | lowercase | none | inherit

Ej: `<div style="text-transform: none"><h1>Original</h1>Lorem ipsum dolor
sit amet...</div>`

.: vertical-align: Determina la alineación vertical de los contenidos de un elemento

Valores: baseline | sub | super | top | text-top | middle | bottom | text-bottom | inherit |
<porcentaje> | <medida>

.: letter-spacing: Permite establecer el espacio entre las letras que forman las palabras del texto

Valores: normal | <medida> | inherit

.: word-spacing: Permite establecer el espacio entre las palabras que forman el texto

Valores: normal | <medida> | inherit



Enlaces

ESTILOS BÁSICOS

Los estilos más sencillos que se pueden aplicar a los enlaces son los que modifican su tamaño de letra, su color y la decoración del texto del enlace. Utilizando las propiedades text-decoration y font-weight se pueden conseguir estilos como los que se muestran en la siguiente imagen:



A continuación se muestran las reglas CSS del ejemplo anterior

```
a { margin: 1em 0; }  
.alternativo {color: #CC0000;}  
.simple {text-decoration: none;}  
.importante {font-weight: bold; font-size: 1.3em;}  
.raro {text-decoration:overline;}  
<a href="#">Enlace con el estilo por defecto</a>  
<a class="alternativo" href="#">Enlace de color rojo</ a>
```



`Enlace sin subrayado`
`Enlace muy importante`
`Enlace con un estilo raro`

PSEUDO-CLASES

CSS también permite aplicar diferentes estilos a un mismo enlace en función de su estado. De esta forma, es posible cambiar el aspecto de un enlace cuando por ejemplo el usuario pasa el ratón por encima o cuando el usuario pincha sobre ese enlace.

Como con los atributos `id` o `class` no es posible aplicar diferentes estilos a un mismo elemento en función de su estado, CSS introduce un nuevo concepto llamado pseudo-clases. En concreto, CSS define las siguientes cuatro pseudo-clases:

`:link`, aplica estilos a los enlaces que apuntan a páginas o recursos que aún no han sido visitados por el usuario.

`:visited`, aplica estilos a los enlaces que apuntan a recursos que han sido visitados anteriormente por el usuario. El historial de enlaces visitados se borra automáticamente cada cierto tiempo y el usuario también puede borrarlo manualmente.

`:hover`, aplica estilos al enlace sobre el que el usuario ha posicionado el puntero del ratón.

`:active`, aplica estilos al enlace que está pinchando el usuario. Los estilos sólo se aplican desde que el usuario pincha el botón del ratón hasta que lo suelta, por lo que suelen ser unas pocas décimas de segundo.

Como se sabe, por defecto los navegadores muestran los enlaces no visitados de color azul y subrayados y los enlaces visitados de color morado.

Las pseudo-clases anteriores permiten modificar completamente ese aspecto por defecto y por eso casi todas las páginas las utilizan.

El siguiente ejemplo muestra cómo ocultar el subrayado cuando el usuario pasa el ratón por encima de cualquier enlace de la página:



```
a:hover { text-decoration: none; }
```

Aplicando las reglas anteriores, los navegadores ocultan el subrayado del enlace sobre el que se posiciona el ratón:





LISTAS

Viñetas personalizadas

Por defecto, los navegadores muestran los elementos de las listas no ordenadas con una viñeta formada por un pequeño círculo de color negro. Los elementos de las listas ordenadas se muestran por defecto con la numeración decimal utilizada en la mayoría de países.

No obstante, CSS define varias propiedades para controlar el tipo de viñeta que muestran las listas, además de poder controlar la posición de la propia viñeta. La propiedad básica es la que controla el tipo de viñeta que se muestra y que se denomina `list-style-type`.

.: **list-style-type**: Permite establecer el tipo de viñeta mostrada para una lista

Valores: `disc` | `circle` | `square` | `decimal` | `decimal-leading-zero` | `lower-roman` | `upper-roman` | `lower-greek` | `lower-latin` | `upper-latin` | `armenian` | `georgian` | `lower-alpha` | `upper-alpha` | `none` | `inherit`

En primer lugar, el valor `none` permite mostrar una lista en la que sus elementos no contienen viñetas, números o letras. Se trata de un valor muy utilizado, ya que es imprescindible para los menús de navegación creados con listas, como se verá más adelante.

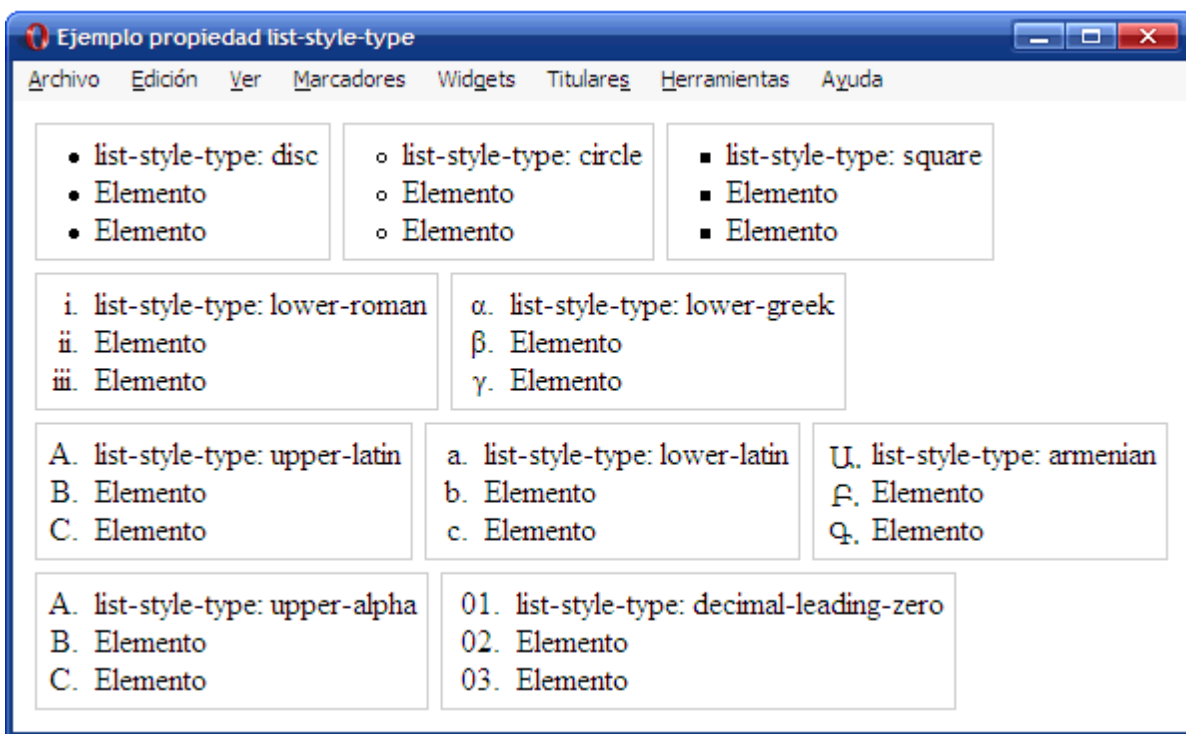
El resto de valores de la propiedad `list-style-type` se dividen en tres tipos: gráficos, numéricos y alfabéticos.

.: **Los valores gráficos** son `disc`, `circle` y `square` y muestran como viñeta un círculo relleno, un círculo vacío y un cuadrado relleno respectivamente.

.: **Los valores numéricos** están formados por `decimal`, `decimal-leading-zero`, `lower-roman`, `upper-roman`, `armenian` y `georgian`.

.: Por último, los **valores alfanuméricos** se controlan mediante `lower-latin`, `lower-alpha`, `upper-latin`, `upper-alpha` y `lower-greek`.

La siguiente imagen muestra algunos de los valores definidos por la propiedad `list-style-type`:



El código CSS de algunas de las listas del ejemplo anterior se muestra a continuación:

```
<ul style="list-style-type: square">
<li>list-style-type: square</li>
<li>Elemento</li>
<li>Elemento</li>
</ul>
<ol style="list-style-type: lower-roman">
<li>list-style-type: lower-roman</li>
<li>Elemento</li>
<li>Elemento</li>
</ol>
<ol style="list-style-type: decimal-leading-zero; padding-left: 2em;">
<li>list-style-type: decimal-leading-zero</li>
<li>Elemento</li>
<li>Elemento</li>
</ol>
```



Cuando se requiere personalizar el aspecto de las viñetas, se debe emplear la propiedad `list-style-image`, que permite mostrar una imagen propia en vez de una viñeta automática.

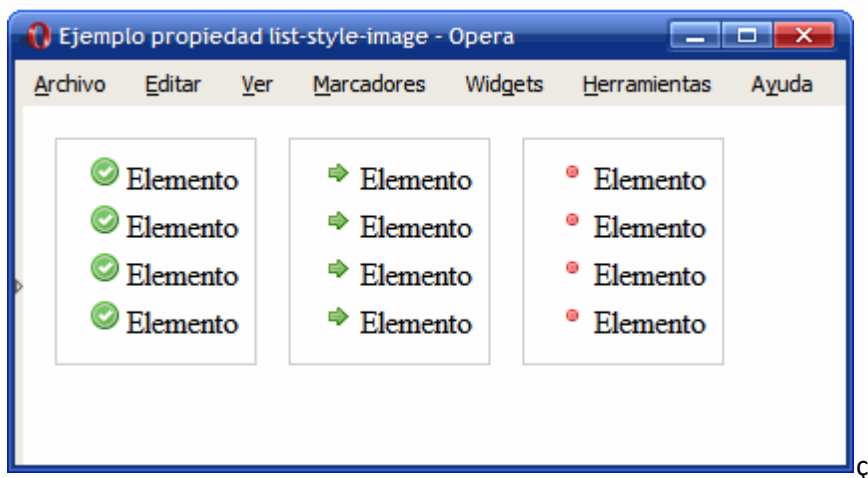
.: `list-style-image`: Permite reemplazar las viñetas automáticas por una imagen personalizada

Valores: `<url>` | `none` | `inherit`

Las imágenes personalizadas se indican mediante la **URL de la imagen**.

Si no se encuentra la imagen o no se puede cargar, se muestra la viñeta automática correspondiente (salvo que explícitamente se haya eliminado mediante la propiedad `list-style-type`).

La siguiente imagen muestra el uso de la propiedad `list-style-image` mediante tres ejemplos sencillos de listas con viñetas personalizadas:

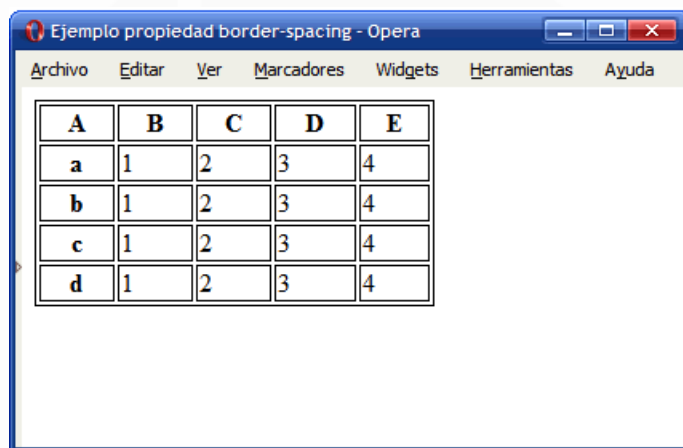




Tablas

ESTILOS BÁSICOS

Cuando se aplican bordes a las celdas de una tabla, el aspecto por defecto con el que se muestra en un navegador es el siguiente:



A	B	C	D	E
a	1	2	3	4
b	1	2	3	4
c	1	2	3	4
d	1	2	3	4

El código HTML y CSS del ejemplo anterior se muestra a continuación (ver ejemplo16.html):

```
.normal {  
width: 250px;  
border: 1px solid #ff0000;  
}  
.normal th, .normal td {  
border: 1px solid #00ff00;  
}  
<table class="normal">  
<tr>  
<th>A</th>  
<th>B</th>  
<th>C</th>  
<th>D</th>
```



```
<th>E</th>
</tr>
</table>
```

Bordes

El estándar CSS 2.1 define dos modelos diferentes para el tratamiento de los bordes de las celdas. La propiedad que permite seleccionar el modelo de bordes es border-collapse:



A	B	C	D	E
a	1	2	3	4
b	1	2	3	4
c	1	2	3	4
d	1	2	3	4

El modelo collapse fusiona de forma automática los bordes de las celdas adyacentes, mientras que el modelo separate fuerza a que cada celda muestre sus cuatro bordes. Por defecto, los navegadores utilizan el modelo separate, tal y como se puede comprobar en el ejemplo anterior.

En general, los diseñadores prefieren el modelo collapse porque estéticamente resulta más atractivo y más parecido a las tablas de datos tradicionales. El ejemplo anterior se puede rehacer para mostrar la tabla con bordes sencillos y sin separación entre celdas.

Pseudo-clase

La pseudo-clase :hover permite añadir fácilmente esta característica:

La regla CSS necesaria se muestra a continuación:



```
table tr:hover {  
  background: #FFFF66;  
}
```



Cambio	Compra	Venta	Máximo	Mínimo
€ Euro/Dolar	1.2524	1.2527	1.2539	1.2488
\$ Dolar/Yen	119.01	119.05	119.82	119.82
£ Libra/Dolar	1.8606	1.8611	1.8651	1.8522
¥ Yen/Euro	0.6711	0.6705	0.6676	0.6713



FONDOS

El último elemento que forma el box model es el fondo de la caja del elemento. El fondo puede ser un color simple o una imagen. El fondo solamente se visualiza en el área ocupada por el contenido y su relleno, ya que el color de los bordes se controla directamente desde los bordes y las zonas de los márgenes siempre son transparentes.

Para establecer un color o imagen de fondo en la página entera, se debe establecer un fondo al elemento `<body>`. Si se establece un fondo a la página, como el valor inicial del fondo de los elementos es transparente, todos los elementos de la página se visualizan con el mismo fondo a menos que algún elemento especifique su propio fondo.

CSS define cinco propiedades para establecer el fondo de cada elemento (`background-color`, `background-image`, `background-repeat`, `background-attachment`, `background-position`)

En ocasiones, es necesario crear un fondo más complejo que un simple color. CSS permite mostrar una imagen como fondo de cualquier elemento:

background-image: Establece una imagen como fondo para los elementos

Valores: `<url>` | `none` | `inherit`

CSS permite establecer de forma simultánea un color y una imagen de fondo. En este caso, la imagen se muestra delante del color, por lo que solamente si la imagen contiene zonas transparentes es posible ver el color de fondo.

Las imágenes de fondo se indican a través de su URL, que puede ser absoluta o relativa. Suele ser recomendable crear una carpeta de imágenes que se encuentre en el mismo directorio que los archivos CSS y que almacene todas las imágenes utilizadas en el diseño de las páginas.

Así, las imágenes correspondientes al diseño de la página se mantienen separadas del resto de imágenes del sitio y el código CSS es más sencillo (por utilizar URL relativas) y más fácil de mantener (por no tener que actualizar URL absolutas en caso de que se cambie la estructura del sitio web).



Por otra parte, suele ser habitual indicar un color de fondo siempre que se muestra una imagen de fondo. En caso de que la imagen no se pueda mostrar o contenga errores, el navegador mostrará el color indicado (que debería ser, en lo posible, similar a la imagen) y la página no parecerá que contiene errores.

Si la imagen que se quiere mostrar es demasiado grande para el fondo del elemento, solamente se muestra la parte de imagen comprendida en el tamaño del elemento. Si la imagen es más pequeña que el elemento, CSS la repite horizontal y verticalmente hasta llenar el fondo del elemento.

Este comportamiento es útil para establecer un fondo complejo a una página web entera.

El siguiente ejemplo utiliza una imagen muy pequeña para establecer un fondo complejo a toda una página:

Imagen original



Reglas CSS

```
body {  
    background-image:url(imagenes/fondo.gif);  
}
```

Con una imagen muy pequeña (y que por tanto, se puede descargar en muy poco tiempo) se consigue cubrir completamente el fondo de la página, con lo que se consigue un gran ahorro de ancho de banda.

En ocasiones, no es conveniente que la imagen de fondo se repita horizontal y verticalmente. Para ello, CSS introduce la propiedad **background-repeat** que permite controlar la forma de repetición de las imágenes de fondo.

El valor **repeat** indica que la imagen se debe repetir en todas direcciones y por tanto, es el comportamiento por defecto.

El valor **no-repeat** muestra una sola vez la imagen y no se repite en ninguna dirección.

El valor **repeat-x** repite la imagen sólo horizontalmente y el valor **repeat-y** repite la imagen solamente de forma vertical.



```
body {  
  background-image:url(imagenes/fondo.gif);  
  background-repeat: repeat-x;  
}
```





Resumen

En esta Unidad...

En la presente unidad desarrollamos los conceptos necesarios para incorporar los atributos gráficos a nuestras estructuras de HTML utilizando el lenguaje CSS

Con las propiedades propuestas podemos comenzar a plantear la estructura gráfica de una página web.

En la próxima Unidad...

En la próxima unidad vamos a comenzar a trabajar con los nuevos elementos incorporados en la versión HTML5 para maquetar sitios web.