

# ÍNDICE

CSS: Estilos de texto



Tipografía	4
Aplicación de estilos a los enlaces	22
Imágenes	28
Creación de listas	32



## CSS: ESTILOS DE TEXTO

A pesar de que con el paso de los años las imágenes, los videos y las animaciones han tomado gran relevancia como medios de contenido por Internet, el texto sigue representando la principal fuente de información y la de uso más común.

Un diseñador web debe tener siempre en cuenta los múltiples problemas que pueden surgir desde el momento que decide establecer las diferentes tipografías.

⚠ Este manual no es una referencia de propiedades CSS. Para una consulta exhaustiva y actualizada de propiedades y valores CSS por favor consultar:

En inglés:

<https://www.w3schools.com/cssref/default.asp>

En castellano:

[https://developer.mozilla.org/es/docs/Web/CSS/Referencia\\_CSS](https://developer.mozilla.org/es/docs/Web/CSS/Referencia_CSS)

# 01

## TRATAMIENTO DEL TEXTO

CSS permite aplicar estilos complejos y muy variados al texto de las páginas web

### CSS: Estilos de 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.

Con la llegada de CSS muchas de estas etiquetas fueron quedando enterradas y en el olvido, ya que los estilos podían replicar lo que se hacía con ellas.

## TIPOGRAFÍA

### Color

La propiedad básica que define CSS relacionada con la tipografía se denomina **color** y se utiliza para establecer el color de la letra. Aunque el color por defecto del texto depende del navegador, todos los navegadores principales utilizan el color negro.

Para establecer el color se puede utilizar cualquiera de las cinco formas que incluye CSS para definir un color. Estas son varias reglas CSS que establecen el color del texto de diferentes formas:

```
h1 { color: #369; }
p { color: black; }
a, span { color: #B1251E; }
div { color: rgb(71, 98, 176); }
```

Como el valor de la propiedad **color** se hereda, normalmente se establece la propiedad color en el elemento **body** para establecer el color de letra de todos los elementos de la página. todos los elementos de la página muestran el mismo color de letra salvo que establezcan de forma explícita otro color. La única excepción son los enlaces que se crean con la etiqueta `<a>`.

Aunque el color de la letra se hereda de los elementos padre a los elementos hijo, con los enlaces no sucede lo mismo, por lo que es necesario indicar su color de forma explícita:

```
/* Establece el mismo color a todos los elementos de
la página salvo los enlaces */
body { color: #777; }
```

```
/* Establece el mismo color a todos los elementos de
la página, incluyendo los enlaces */
body, a { color: #777; }
```

### Font-family

La otra propiedad básica que define CSS relacionada con la tipografía se denomina **font-family** y se utiliza para indicar el tipo de letra con el que se muestra 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.

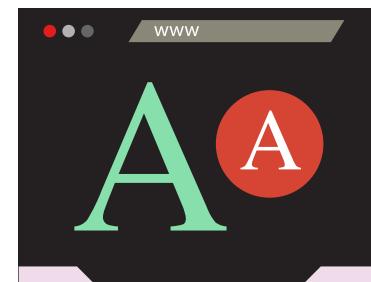


■ Mediante el nombre genérico de una familia tipográfica: los nombres genéricos no se refieren a ninguna fuente en concreto, sino que hacen referencia al estilo del tipo de letra. Las familias genéricas definidas son serif (tipo de letra similar a Times New Roman), sans-serif (tipo Arial), cursive (tipo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

**Los navegadores muestran el texto de las páginas web utilizando los tipos de letra instalados en el ordenador o dispositivo del propio usuario.** De esta forma, si el diseñador indica en la propiedad **font-family** que el texto debe mostrarse con un tipo de letra especialmente raro o rebuscado, casi ningún usuario dispondrá de ese tipo de letra.

Para evitar el problema común de que el usuario no tenga instalada la fuente que quiere utilizar el diseñador, **CSS permite indicar en la propiedad font-family más de un tipo de letra.**

El navegador probará en primer lugar con el primer tipo de letra indicado. Si el usuario la tiene instalada, el texto se muestra con ese tipo de letra.



Por defecto, si el usuario no dispone del primer tipo de letra indicado, el navegador irá probando con el resto de tipos de letra hasta que encuentre alguna fuente que esté instalada en el ordenador del usuario.

Evidentemente, el diseñador no puede indicar para cada propiedad font-family tantos tipos de letra como posibles fuentes parecidas existan. Para solucionar este problema se utilizan las familias tipográficas genéricas.

Por todo ello, el valor de **font-family** suele definirse como una lista de tipos de letra alternativos separados por comas. El último valor de la lista es el nombre de la familia tipográfica genérica que más se parece al tipo de letra que se quiere utilizar.

Las listas de tipos de letra más utilizadas son las siguientes:

```
font-family: Arial, Helvetica, sans-serif;  
font-family: "Times New Roman", Times, serif;  
font-family: "Courier New", Courier, monospace;  
font-family: Georgia, "Times New Roman", Times, serif;  
font-family: Verdana, Arial, Helvetica, sans-serif;
```

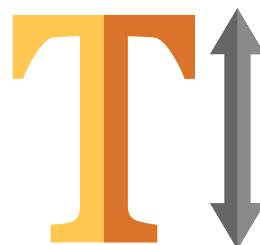
Cuando se quiere disponer de un diseño complejo con fuentes muy especiales, se debe recurrir a soluciones alternativas: crear imágenes en las que se muestra el texto con la fuente deseada. Esta técnica solamente es viable para textos cortos (por ejemplo los titulares) y puede ser manual (creando las imágenes una por una) o automática, utilizando JavaScript, PHP y/o CSS.

Otra alternativa es la de la sustitución automática de texto basada en Flash: la técnica más conocida es la de sIFR.

## Font-size

Una vez seleccionado el tipo de letra, **se puede modificar su tamaño mediante la propiedad font-size**. 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.

CSS recomienda indicar el tamaño del texto en la unidad em o en porcentaje (%). Además, es habitual indicar el tamaño del texto en puntos (pt) cuando el documento está específicamente diseñado para imprimirlo.

Por defecto los navegadores asignan los siguientes tamaños a los títulos de sección: <h1> = xx-large, <h2> = x-large, <h3> = large, <h4> = medium, <h5> = small, <h6> = xx-small.

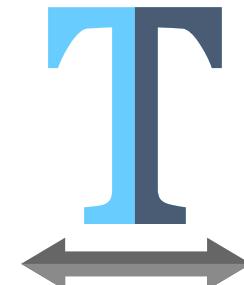
## Font-weight

Una vez indicado el tipo y el tamaño de letra, es habitual modificar otras características como su grosor (texto en negrita) y su estilo (texto en cursiva). **La propiedad que controla la anchura de la letra es font-weight**.

Los valores que normalmente se utilizan son **normal** (el valor por defecto) y **bold** para los textos en negrita. El valor normal equivale al valor numérico 400 y el valor bold al valor numérico 700.

Por defecto, los navegadores muestran el texto de los elementos <em> en cursiva y el texto de los elementos <strong> en negrita. La propiedad **font-weight** permite alterar ese aspecto.

```
#especial em {  
    font-weight: bold;  
}  
  
#especial strong {  
    font-weight: normal;  
    background-color: #FFFF66;  
    padding: 2px;  
}
```



## Font-style

Además de la anchura de la letra, CSS permite variar su estilo mediante la propiedad **font-style**. Normalmente la propiedad **font-style** se emplea para mostrar un texto en cursiva mediante el valor **italic**.

```
#especial em {  
    font-weight: bold;  
    font-style: normal;  
}  
  
#especial strong {  
    font-weight: normal;  
    font-style: italic;  
    background-color:#FFFF66;  
    padding: 2px;  
}
```

## Font-variant

Por último, CSS permite otra variación en el estilo del tipo de letra, controlado mediante la propiedad **font-variant**.

La propiedad **font-variant** no se suele emplear habitualmente, ya que sólo permite mostrar el texto con letra **versal** (mayúsculas pequeñas). Solamente es necesario añadir una regla a los estilos CSS:

```
#especial {  
    font-variant: small-caps;  
}
```

## Font

Por otra parte, CSS proporciona una propiedad tipo "shorthand" denominada **font** y que permite indicar de forma directa algunas o todas las propiedades de la tipografía de un texto. Este es el orden en el que se deben indicar las propiedades del texto:

- 1 En primer lugar y de forma opcional se indican el **font-style**, **font-variant** y **font-weight** en cualquier orden.
- 2 A continuación, se indica obligatoriamente el valor de **font-size** seguido opcionalmente por el valor de **line-height**.
- 3 Por último, se indica obligatoriamente el tipo de letra a utilizar.

Aunque su uso no es muy común, la propiedad **font** también permite indicar el tipo de letra a utilizar mediante una serie de palabras clave: **caption, icon, menu, message-box, small-caption, status-bar**.

Si por ejemplo se utiliza la palabra **status-bar**, el navegador muestra el texto con el mismo tipo de letra que la que utiliza el sistema operativo para mostrar los textos de la barra de estado de las ventanas. La palabra **icon** se puede utilizar para mostrar el texto con el mismo tipo de letra que utiliza el sistema operativo para mostrar el nombre de los iconos y así sucesivamente.

## TEXTO

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.**



### Text-align

La **propiedad que define la alineación del texto se denomina text-align**. Los valores definidos por CSS permiten alinear el texto según los valores tradicionales:

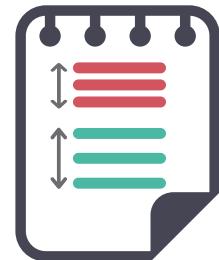


La propiedad **text-align** no sólo alinea el texto, sino que también alinea todos sus contenidos, como las imágenes.

### Line-height

El interlineado de un texto se controla mediante la propiedad **line-height**, que **permite controlar la altura ocupada por cada línea de texto**. Además, permite indicar un número sin unidades que se interpreta como el múltiplo del tamaño de letra del elemento.

Siempre que se utilice de forma moderada, el interlineado mejora notablemente la legibilidad de un texto.



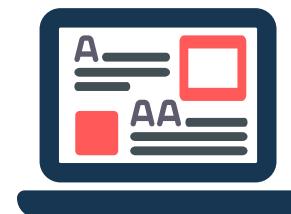
## Text-decoration

Además de la decoración que se puede aplicar a la tipografía que utilizan los textos, **CSS define otros estilos y decoraciones para el texto en su conjunto**. La propiedad que decora el texto se denomina **text-decoration** y establece la decoración del texto (subrayado, tachado, parpadeante, etc.).

El valor **underline** subraya el texto, por lo que puede confundir a los usuarios haciéndoles creer que se trata de un enlace. El valor **overline** añade una línea en la parte superior del texto, un aspecto que raramente es deseable. El valor **line-through** muestra el texto tachado con una línea continua, por lo que su uso tampoco es muy habitual. Por último, el valor **blink** muestra el texto parpadeante y se recomienda evitar su uso por las molestias que genera a la mayoría de usuarios.

## Text-transform

Una de las propiedades de CSS más desconocidas y que puede ser de gran utilidad es la propiedad **text-transform**, que puede variar de forma sustancial el aspecto del texto.



La propiedad **text-transform** permite mostrar el texto original transformado en un texto completamente en mayúsculas (**uppercase**), en minúsculas (**lowercase**) o con la primera letra de cada palabra en mayúscula (**capitalize**):

### Original:

Id excerios similique es nialtam sunt eos volorit quiasin vereiuntium sum repra similit untio es ut estem. Harum explique voluptatquos exeritate in porporia.

### Uppercase:

ID EXCERIOS SIMILIQUE ES NIAITAM SUNT EOS VOLORIT QUIASIN VEREIUNTIUM SUM REPRA SIMILIT UNTIO ES UT ESTEM. HARUM EXPLIQUE VOLUPTATQUOS EXERITATE IN PORPORIA.

### Lowercase:

id excerios similique es nialtam sunt eos volorit quiasin vereiuntium sum repra similit untio es ut estem. harum explique voluptatquos exeritate in porporia.

### Capitalize:

Id Excerios Similique Es Nialtam Sunt Eos Volorit Quiasin Vereiuntium Sum Repra Similit Untio Es Ut Estem. Harum Explique Voluptatquos Exeritate In Porporia.

## Vertical-align

Uno de los principales problemas del diseño de documentos y páginas mediante CSS consiste en **la alineación vertical en una misma línea de varios elementos diferentes como imágenes y texto**. Para controlar esta alineación, CSS define la propiedad **vertical-align**.

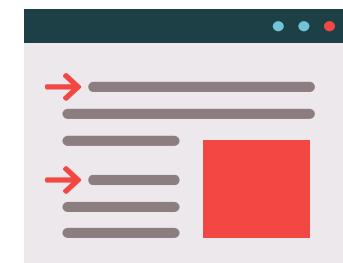
El valor por defecto es **baseline** y el valor más utilizado cuando se establece la propiedad **vertical-align** es **middle**.



## Text-indent

En muchas publicaciones impresas suele ser habitual tabular la primera línea de cada párrafo para facilitar su lectura. **CSS permite controlar esta tabulación mediante la propiedad text-indent**.

El uso de la tabulación ayuda a mejorar los márgenes y la organización del documento que te encuentras realizando.



## Letter-spacing y word-spacing

CSS también permite controlar la separación entre las letras que forman las palabras y la separación entre las palabras que forman los textos. La propiedad que controla la separación entre letras se llama **letter-spacing** y la separación entre palabras se controla mediante **word-spacing**.

Cuando se utiliza un valor numérico en las propiedades **letter-spacing** y **word-spacing**, se interpreta como la separación adicional que se añade (si el valor es positivo) o se quita (si el valor es negativo) a la separación por defecto entre letras y palabras respectivamente.

Como ya se sabe, el tratamiento que hace HTML de los espacios en blanco es uno de los aspectos más difíciles de comprender cuando se empiezan a crear las primeras páginas web. Básicamente, HTML elimina todos los espacios en blanco sobrantes, es decir, todos salvo un espacio en blanco entre cada palabra.

Para forzar los espacios en blanco adicionales se debe utilizar la entidad HTML **&ampnbsp**; y para forzar nuevas líneas, se utiliza el elemento **<br/>**. Además, HTML proporciona el elemento **<pre>** que muestra el contenido tal y como se escribe, respetando todos los espacios en blanco y todas las nuevas líneas.



## White-space

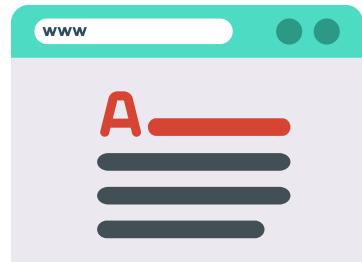
CSS también permite controlar el tratamiento de los espacios en blanco de los textos mediante la propiedad white-space. Tiene varios valores asociados cuyo significado es el siguiente:

- **Normal**: comportamiento por defecto de HTML.
- **Pre**: se respetan los espacios en blanco y las nuevas líneas (exactamente igual que la etiqueta `<pre>`). Si la línea es muy larga, se sale del espacio asignado para ese contenido.
- **Nowrap**: elimina los espacios en blanco y las nuevas líneas. Si la línea es muy larga, se sale del espacio asignado para ese contenido.
- **Pre-wrap**: se respetan los espacios en blanco y las nuevas líneas, pero ajustando cada línea al espacio asignado para ese contenido.
- **Pre-line**: elimina los espacios en blanco y respeta las nuevas líneas, pero ajustando cada línea al espacio asignado para ese contenido.

## CSS: Estilos de texto

### Pseudo-elementos

Por último, CSS define unos elementos especiales llamados **"pseudo-elementos"** que permiten aplicar estilos a ciertas partes de un texto. En concreto, CSS permite definir estilos especiales a la primera frase de un texto y a la primera letra de un texto.



El pseudo-elemento **first-line** permite aplicar estilos a la primera línea de un texto. Las palabras que forman la primera línea de un texto dependen del espacio reservado para mostrar el texto o del tamaño de la ventana del navegador, por lo que CSS calcula de forma automática las palabras que forman la primera línea de texto en cada momento.

De la misma forma, CSS permite aplicar estilos a la primera letra del texto mediante el pseudo-elemento **first-letter**.

# 02

## APLICACIÓN DE ESTILOS A LOS ENLACES

El aspecto de los enlaces también pueden modificarse, ya sea de forma básica o avanzada

### CSS: Estilos de texto

#### ESTILOS BÁSICOS

##### Tamaño, color y decoración

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**. Para ello se utilizan las propiedades text-decoration y font-weight.



■ Ejemplo de enlaces con estilos diferentes

## 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.

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 ese aspecto** y por eso casi todas las páginas las utilizan.

Las **pseudo-clases siempre incluyen dos puntos (:) por delante de su nombre y siempre se añaden a continuación del elemento al que se aplican**, sin dejar ningún espacio en blanco por delante. Además, se pueden combinar con cualquier otro selector complejo.

Cuando se aplican varias pseudo-clases diferentes sobre un mismo enlace, se producen colisiones. Si se pasa por ejemplo el ratón por encima de un enlace visitado, se aplican los estilos de las pseudo-clases **:hover** y **:visited**. Si el usuario pincha sobre un enlace no visitado, se aplican las pseudo-clases **:hover**, **:link** y **:active** y así sucesivamente.

Si se definen varias pseudo-clases sobre un mismo enlace, el único orden que asegura que todos los estilos de las pseudo-clases se aplican de forma coherente es el siguiente: **:link**, **:visited**, **:hover** y **:active**.

Las pseudo-clases **:link** y **:visited** solamente están definidas para los enlaces, pero las pseudo-clases **:hover** y **:active** se definen para todos los elementos HTML. Desafortunadamente, Internet Explorer 6 y sus versiones anteriores solamente las soportan para los enlaces.

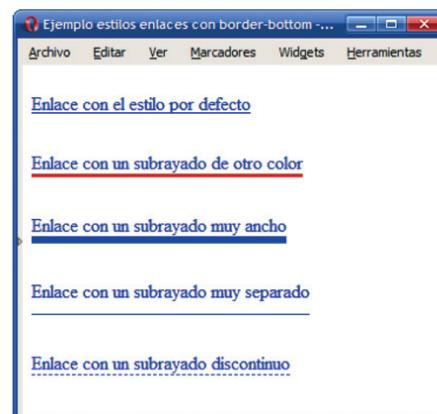


## ESTILOS AVANZADOS

### Decoración personalizada

La propiedad **text-decoration** permite añadir o eliminar el subrayado de los enlaces. Sin embargo, el aspecto del subrayado lo controla automáticamente el navegador, por lo que su color siempre es el mismo que el del texto del enlace y su anchura es proporcional al tamaño de letra.

No obstante, utilizando la propiedad **border-bottom** es posible añadir cualquier tipo de subrayado para los enlaces de las páginas. El siguiente ejemplo muestra algunos enlaces con el subrayado personalizado:



■ Enlaces con subrayado personalizado mediante la propiedad border

### Imágenes según el tipo de enlace

En ocasiones, puede resultar útil incluir un pequeño ícono al lado de un enlace para indicar el tipo de contenido que enlaza, como por ejemplo un archivo comprimido o un documento con formato PDF. Este tipo de imágenes son puramente decorativas, por lo que se deberían añadir con CSS y no con elementos de tipo **<img>**.

Utilizando la propiedad **background** y **background-image** se puede personalizar el aspecto de dichos enlaces. La técnica consiste en mostrar una imagen de fondo sin repetición en el enlace y añadir el padding necesario al texto del enlace para que no se solape con la imagen de fondo.



■ Personalizando el aspecto de los enlaces en función de su tipo

### Mostrar los enlaces como si fueran botones

Las propiedades definidas por CSS permiten mostrar los enlaces con el aspecto de un botón, lo que puede ser útil en formularios en los que no se utilizan elementos específicos para crear botones.



# 03

## IMÁGENES

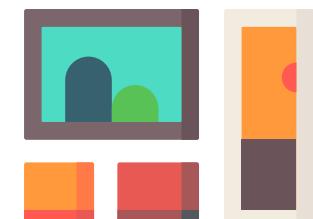
Con el código CSS también podemos controlar las propiedades de las imágenes que aparecen

### ESTILOS BÁSICOS

#### Establecer la anchura y altura de las imágenes

Utilizando las propiedades **width** y **height**, es posible mostrar las imágenes con cualquier altura/anchura, independientemente de su altura/anchura real. No obstante, si se utilizan alturas/anchuras diferentes de las reales, el navegador deforma las imágenes y el resultado estético es muy desagradable.

Por otra parte, establecer la altura/anchura de todas las imágenes mediante CSS es una práctica poco recomendable. El motivo es que normalmente dos imágenes diferentes no comparten la misma altura/anchura. Por lo tanto, se debe crear una nueva regla CSS (y un nuevo selector) para cada una de las diferentes imágenes del sitio web.

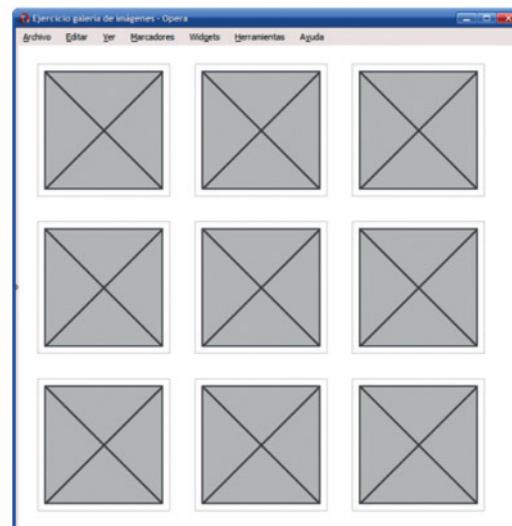


En la práctica, esta forma de trabajo produce una sobrecarga de estilos inviable. Por este motivo, aunque no respeta la separación completa entre contenidos (XHTML) y presentación (CSS), se recomienda establecer la altura/anchura de las imágenes mediante los atributos **width** y **height** de la etiqueta `<img>`.

## Eliminar el borde de las imágenes con enlaces

Cuando una imagen forma parte de un enlace, los navegadores muestran por defecto un borde azul grueso alrededor de las imágenes. Por tanto, una de las reglas más utilizadas en los archivos CSS es la que elimina los bordes de las imágenes con enlaces:

```
img {  
    border: none;  
}
```



■ Galería de imágenes construida con CSS

## ESTILOS AVANZADOS

### Sombra (drop shadow)

La mayoría de aplicaciones de diseño gráfico permiten añadir una sombra (drop shadow) a las imágenes. CSS no incluye propiedades que permitan mostrar de forma sencilla sombras en los elementos.



Existen varias técnicas sencillas y otras más avanzadas que **permiten crear sombras que se adapten a cualquier imagen o elemento**. Una de ellas consiste en la utilización de un par de elementos `<div>` alrededor del elemento que va a mostrar una sombra. La sombra se consigue mediante una imagen muy grande que contiene una sombra orientada hacia el lado derecho e inferior. La ventaja de este método es que es sencillo y solamente requiere añadir un par de `<div>` por cada elemento. Además, como la sombra es una imagen muy grande, el efecto funciona con elementos de cualquier tamaño.

El mayor inconveniente de este método es que la sombra siempre se muestra en la misma dirección (derecha inferior) y que en Internet Explorer 6 y versiones anteriores sólo muestran la sombra correctamente cuando el color de fondo de la página es blanco (ya que Internet Explorer 6 y versiones anteriores no soportan imágenes en formato PNG con transparencias).

# 04

## CREACIÓN DE LISTAS

Subtítulo de 1 a 3 líneas

### ESTILOS BÁSICOS

#### 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.

### ■ List-style-type

La propiedad básica es la que controla el tipo de viñeta que se muestra y que se denomina **list-style-type**. 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.**

El resto de valores de la propiedad **list-style-type** se dividen en tres tipos:

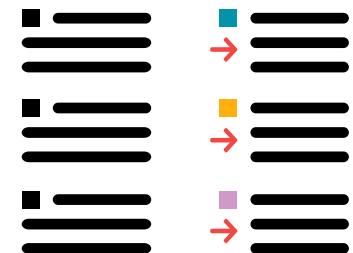
- Los **valores gráficos** son **disc**, **circle** y **square** y muestran como viñeta un círculo relleno, un círculo vacío y un cuadrado relleno respectivamente.
- Los **valores numéricos** están formados por **decimal**, **decimal-leading-zero**, **lower-roman**, **upper-roman**, **armenian** y **georgian**.
- Los **valores alfanuméricos** se controlan mediante **lower-latin**, **lower-alpha**, **upper-latin**, **upper-alpha** y **lower-greek**.



### ■ List-style-position

La propiedad **list-style-position** permite controlar la colocación de las viñetas. Tiene asignados dos valores: **outside** y **inside**. La diferencia entre los valores **outside** y **inside** se hace evidente cuando los elementos contienen mucho texto.

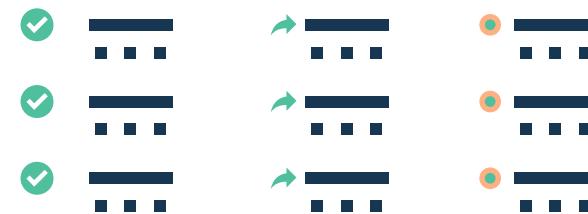
Utilizando las propiedades anteriores (**list-style-type** y **list-style-position**), se puede seleccionar el tipo de viñeta y su posición, pero no es posible personalizar algunas de sus características básicas como su color y tamaño.



### ■ List-style-image

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.**

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**).



### ■ List-style

Como es habitual, CSS define una propiedad de tipo "shorthand" que **permite establecer todas las propiedades de una lista de forma directa**. La propiedad se denomina **list-style**.

En la definición anterior, la notación **||** significa que el orden en el que se indican los valores de la propiedad es indiferente. El siguiente ejemplo indica que no se debe mostrar ni viñetas automáticas ni viñetas personalizadas:

```
ul { list-style: none }
```

Cuando se utiliza una viñeta personalizada, es conveniente indicar la viñeta automática que se mostrará cuando no se pueda cargar la imagen:

```
ul { list-style: url(imagenes/cuadrado_rojo.gif) square; }
```

### Menú vertical básico

Las listas HTML se suelen emplear, además de para su función natural, para la creación de menús de navegación verticales y horizontales. Los tipos de menús verticales que se pueden definir mediante las propiedades CSS son innumerables

El proceso de transformación de la lista en un menú requiere de los siguientes pasos:

- 1. Definir la anchura del menú.
- 2. Eliminar las viñetas automáticas y todos los márgenes y espaciados aplicados por defecto.
- 3. Añadir un borde al menú de navegación y establecer el color de fondo y los bordes de cada elemento del menú.
- 4. Aplicar estilos a los enlaces: mostrarlos como un elemento de bloque para que ocupen todo el espacio de cada **<li>** del menú, añadir un espacio de relleno y modificar los colores y la decoración por defecto.

## ESTILOS AVANZADOS

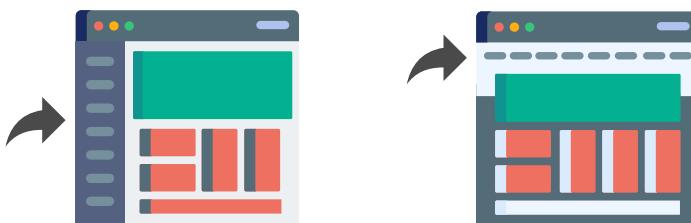
### Menú horizontal básico

A partir de un menú vertical sencillo, es posible crear un menú horizontal sencillo aplicando las propiedades CSS conocidas hasta el momento.

Los elementos del menú se definen en los elementos `<a>` en lugar de definirlas para los elementos `<li>`. En cualquier caso, es indiferente el elemento en el que se aplican los estilos que definen el aspecto de cada opción del menú.

El proceso de transformación del menú vertical en un menú horizontal consta de los siguientes pasos:

- 1. Eliminar la anchura y el borde del elemento `<ul>` y aplicar las propiedades `float` y `clear`.
- 2. La clave de la transformación reside en modificar la propiedad `float` de los elementos `<li>` del menú.
- 3. Modificar el `padding` y los bordes de los enlaces del menú.
- 4. Por último, se añade un borde izquierdo en el elemento `<ul>` para homogeneizar el aspecto de los elementos del menú.



### Menú horizontal con pestañas

Modificando los estilos de cada elemento del menú y utilizando imágenes de fondo y las pseudo-clases `:hover` y `:active`, se pueden crear menús horizontales complejos, incluso con el aspecto de un menú de solapas o pestañas:



Además de los menús horizontales de solapas, existen muchos otros tipos diferentes de menús horizontales (y verticales) que se pueden realizar empleando exclusivamente CSS. El código CSS de todos los ejemplos de la imagen anterior y muchos otros se pueden encontrar en: <http://alvit.de/css-showcase/css-navigation-techniques-showcase.php>

The logo consists of the word "nowe" in a black sans-serif font. The letter "o" features a blue wavy line through its center.

Espacio libre para tus ideas:

---