

CURSO DE PROGRAMACIÓN FULL STACK

# ANEXO CSS



## PROPIEDADES CSS

Tanto para practicar en tu aprendizaje como para trabajar con las CSS lo mejor es disponer de las distintas **propiedades y valores** de estilos que podemos aplicarles a las páginas web.

Aquí puedes ver las propiedades CSS más fundamentales para aplicar estilos a elementos básicos, que te vendrá perfecto para comenzar con las CSS.

## UNIDADES DE MEDIDA CSS

Los valores que se pueden asignar a los atributos de estilo se pueden ver en una tabla más adelante en la guía. Muchos de los valores que podemos asignarle son unidades de medida, por ejemplo, el valor del tamaño de un margen o el tamaño de la fuente. Las unidades de medida **CSS se pueden clasificar en dos grupos**, las **relativas y las absolutas**. Más la posibilidad de expresar valores en porcentaje.

**Absolutas:** las unidades absolutas son medidas fijas, que deberían verse igual en todos los dispositivos. Como los centímetros, que son una convención de medida internacional. Pese a que en principio pueden parecer más útiles, puesto que se verían en todos los sistemas igual, **tienen el problema de adaptarse menos a las distintas particularidades de los dispositivos que pueden acceder a una web y restan accesibilidad a nuestro web**. Puede que en tu ordenador 1 centímetro sea una medida razonable, pero en un móvil puede ser un espacio exageradamente grande, puesto que la pantalla es mucho menor. Se aconseja utilizar, por tanto, medidas relativas.

- **pt** (puntos): Un punto es 1/72 pulgadas.
- **in** (pulgadas)
- **cm** (centímetros)
- **mm** (milímetros)
- **px** (píxeles): Es la unidad mínima de resolución de la pantalla. En realidad suele considerársela una unidad absoluta, relativa o híbrida dependiendo del criterio que se analice. Un pixel equivale a 0.26 milímetros.

**Relativas:** se llaman así porque son unidades relativas al medio o soporte sobre el que se está viendo la página web, que dependiendo de cada usuario puede ser distinto, puesto que existen muchos dispositivos que pueden acceder a la web, como ordenadores o teléfonos móviles. En principio las unidades relativas son más aconsejables, porque se ajustarán mejor al medio con el que el usuario está accediendo a nuestra web. Son las siguientes:

### UNIDAD EM

La unidad **em** se utiliza para hacer referencia al tamaño actual de la fuente que ha sido establecida en el navegador, que habitualmente es un valor aproximado a 16px (salvo que se modifique por el usuario). De esta forma, podemos trabajar simplificando las unidades a medidas en base a ese tamaño.

Por ejemplo, imaginemos que el tamaño de la fuente establecida en el navegador del usuario es exactamente 16px. Una cantidad 1em equivaldría a 16px, mientras que una cantidad de 2em sería justo el doble: 32px. Por otro lado, una cantidad de 0.5em sería justo la mitad: 8px.



## UNIDAD PORCENTAJE

Porcentaje (%), es una de las unidades relativas más utilizadas. Su valor está calculado siempre en base a otro elemento. Si lo aplicamos sobre una fuente es relativo al tamaño de la fuente declarada en el contexto, pero si lo aplicamos al width de un elemento entonces es relativo al ancho de su contenedor.

El porcentaje se utiliza para definir una unidad en función de la que esté definida en un momento dado. Imaginemos que estamos trabajando en 12pt y definimos una unidad como 150%. Esto sería igual al 150% de los 12pt actuales, que equivale a 18pt.

## COLORES EN CSS

Con CSS se puede especificar colores para cada elemento HTML de la página, incluso hay elementos que podrían admitir varios colores, como el color de fondo o el color del borde. Pero bueno, vamos a ver ahora es las distintas maneras de escribir un color en una declaración CSS.

Porque lo más habitual es que especifiquemos un color con su valor RGB. Pero en CSS tenemos otras maneras de declarar colores que pueden interesarnos, como mínimo para poder entender el código CSS cuando lo veamos escrito.

## NOTACIÓN HEXADECIMAL RGB

Se especifican los tres valores de color (rojo, verde y azul) con valores en hexadecimal entre 00 y FF.

```
background-color: #ff8800;
```

## NOMBRE DEL COLOR

También podemos definir un color por su nombre. Los nombres de colores son en inglés, los mismos que sirven para especificar colores con HTML.

```
color: red;
```

```
border-color: lime;
```

## NOTACIÓN DE COLOR CON PORCENTAJES DE RGB

Se puede definir un color por los distintos porcentajes de valores RGB. Si todos los valores están al 100% el color es blanco. Si todos están al 0% obtendríamos el negro y con combinaciones de distintos porcentajes de RGB obtendríamos cualquier matiz de color.

```
color: rgb(33%, 0%, 0%);
```

## NOTACIÓN POR VALORES DECIMALES DE RGB, DE 0 A 255

De una manera similar a la notación por porcentajes de RGB se puede definir un color directamente con valores decimales en un rango desde 0 a 255.

```
color: rgb(200,255,0);
```

De entre todas estas notaciones podemos utilizar la que más nos interese o con la que nos sintamos más a gusto. Nosotros en nuestros ejemplos venimos utilizando la notación hexadecimal RGB por habernos acostumbrado a ella en HTML.

## COLOR TRANSPARENTE

Para finalizar, podemos comentar que también existe el color transparente, que no es ningún color, sino que especifica que el elemento debe tener el mismo color que el fondo donde está. Este valor, `transparent`, sustituye al color. Podemos indicarlo en principio sólo para fondos de elementos, es decir, para el atributo `background-color`.

```
background-color: transparent;
```

## DIMENSIONES EN CSS

### ANCHO

La propiedad CSS que controla la anchura de la caja de los elementos se denomina **width**.

|                      |   |
|----------------------|---|
| <b>width</b>         | Anchura   |
| <b>Valores</b>       | <medida>   <porcentaje>   auto   inherit  |
| <b>Se aplica a</b>   | Todos los elementos, salvo los elementos en línea que no sean imágenes, las filas de tabla y los grupos de filas de tabla |
| <b>Valor inicial</b> | auto  |
| <b>Descripción</b>   | Establece la anchura de un elemento   |

## ALTO

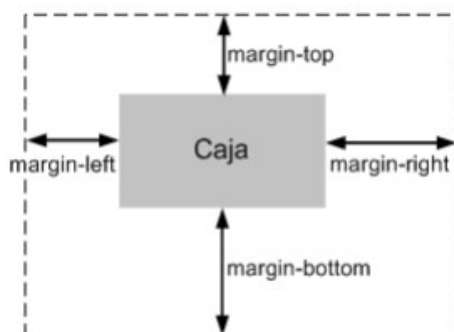
La propiedad CSS que controla la anchura de la caja de los elementos se denomina **height**.

|               |   |
|---------------|---|
| <b>height</b> | Altura  |
| Valores       | <medida>   <porcentaje>   auto   inherit  |
| Se aplica a   | Todos los elementos, salvo los elementos en línea que no sean imágenes, las columnas de tabla y los grupos de columnas de tabla |
| Valor inicial | auto  |
| Descripción   | Establece la altura de un elemento  |

## MARGEN

CSS define cuatro propiedades para controlar cada uno de los márgenes horizontales y verticales de un elemento.

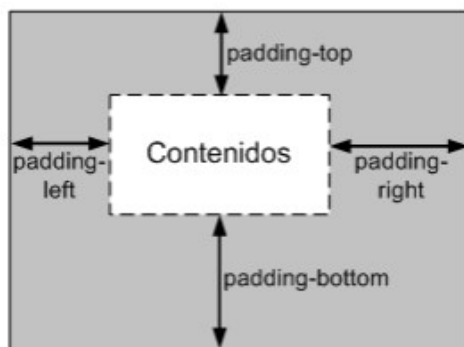
|  |  |
|--|--|
| <b>margin-top</b><br><b>margin-right</b><br><b>margin-bottom</b><br><b>margin-left</b> | Margen superior<br>Margen derecho<br>Margen inferior<br>Margen izquierdo   |
| Valores  | <medida>   <porcentaje>   auto   inherit   |
| Se aplica a  | Todos los elementos, salvo margin-top y margin-bottom que sólo se aplican a los elementos de bloque y a las imágenes |
| Valor inicial  | 0  |
| Descripción  | Establece cada uno de los márgenes horizontales y verticales de un elemento  |



## RELLENO

CSS define cuatro propiedades para controlar cada uno de los espacios de relleno horizontales y verticales de un elemento.

|  |  |
|--|--|
| <b>padding-top</b><br><b>padding-right</b><br><b>padding-bottom</b><br><b>padding-left</b> | Relleno superior<br>Relleno derecho<br>Relleno inferior<br>Relleno izquierdo                               |
| Valores  | <medida>   <porcentaje>   inherit  |
| Se aplica a  | Todos los elementos excepto algunos elementos de tablas como grupos de cabeceras y grupos de pies de tabla |
| Valor inicial  | 0  |
| Descripción  | Establece cada uno de los rellenos horizontales y verticales de un elemento                                |



caja

## BORDES EN CSS

### BORDES - TAMAÑO

CSS permite definir el aspecto de cada uno de los cuatro bordes horizontales y verticales de los elementos. Para cada borde se puede establecer su anchura, su color y su estilo.

|  |  |
|--|--|
| <b>border-top-width</b><br><b>border-right-width</b><br><b>border-bottom-width</b><br><b>border-left-width</b> | Anchura del borde superior<br>Anchura del borde derecho<br>Anchura del borde inferior<br>Anchura del borde izquierdo |
| Valores  | ( <medida>   thin   medium   thick )   inherit   |
| Se aplica a  | Todos los elementos  |
| Valor inicial  | Medium   |
| Descripción  | Establece la anchura de cada uno de los cuatro bordes de los elementos   |

## BORDES - COLOR

El color de los bordes se controla con las cuatro propiedades siguientes:

|  |  |
|--|--|
| <b>border-top-color</b><br><b>border-right-color</b><br><b>border-bottom-color</b><br><b>border-left-color</b> | Color del borde superior<br>Color del borde derecho<br>Color del borde inferior<br>Color del borde izquierdo |
| Valores  | <color>   transparent   inherit  |
| Se aplica a  | Todos los elementos  |
| Valor inicial  | -  |
| Descripción  | Establece el color de cada uno de los cuatro bordes de los elementos   |

## BORDES - ESTILO

El color de los bordes se controla con las cuatro propiedades siguientes:

|  |  |
|--|--|
| <b>border-top-color</b><br><b>border-right-color</b><br><b>border-bottom-color</b><br><b>border-left-color</b> | Color del borde superior<br>Color del borde derecho<br>Color del borde inferior<br>Color del borde izquierdo |
| Valores  | <color>   transparent   inherit  |
| Se aplica a  | Todos los elementos  |
| Valor inicial  | -  |
| Descripción  | Establece el color de cada uno de los cuatro bordes de los elementos   |

## BORDES - FORMA RESUMIDA

Todos los estilos de los bordes se controlan con la siguiente siguientes:

|               |   |
|---------------|---|
| <b>border</b> | Estilo completo de todos los bordes                               |
| Valores       | ( <medida_borde>    <color_borde>    <estilo_borde> )   inherit   |
| Se aplica a   | Todos los elementos   |
| Valor inicial | -   |
| Descripción   | Establece el estilo completo de todos los bordes de los elementos |

## FONDO EN CSS

### FONDO - COLOR

El color de fondo se establece con esta propiedad:

|                         |  |
|-------------------------|--|
| <b>background-color</b> | Color de fondo                                 |
| Valores                 | <color>   transparent   inherit                |
| Se aplica a             | Todos los elementos                            |
| Valor inicial           | transparent                                    |
| Descripción             | Establece un color de fondo para los elementos |

La imagen de fondo se establece con esta propiedad:

|                         |  |
|-------------------------|--|
| <b>background-image</b> | Imagen de fondo                                    |
| Valores                 | <url>   none   inherit                             |
| Se aplica a             | Todos los elementos                                |
| Valor inicial           | none   |
| Descripción             | Establece una imagen como fondo para los elementos |

### FONDO - REPETICIÓN

La repetición de la imagen de fondo se configura con esta propiedad:

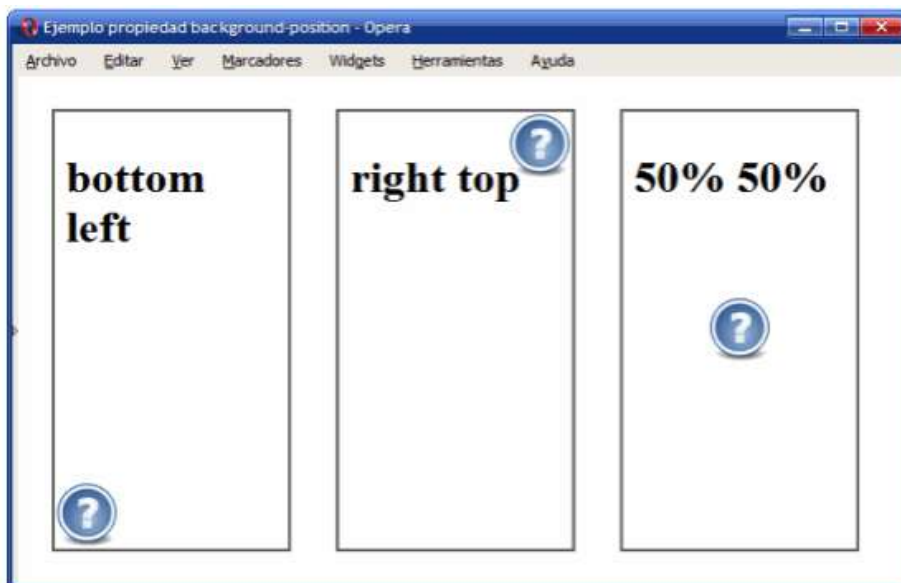
|                          |  |
|--------------------------|--|
| <b>background-repeat</b> | Repetición de la imagen de fondo                             |
| Valores                  | repeat   repeat-x   repeat-y   no-repeat   inherit           |
| Se aplica a              | Todos los elementos  |
| Valor inicial            | repeat   |
| Descripción              | Controla la forma en la que se repiten las imágenes de fondo |



## FONDO - POSICIÓN

La posición de la imagen de fondo se configura con esta propiedad:

| <b>background-position</b> | Posición de la imagen de fondo  |
|----------------------------|---|
| <b>Valores</b>             | ( ( <porcentaje>   <medida>   left   center   right ) ( <porcentaje>   <medida>   top   center   bottom )? )   ( ( left   center   right )    ( top   center   bottom ) )   inherit |
| <b>Se aplica a</b>         | Todos los elementos   |
| <b>Valor inicial</b>       | 0% 0%   |
| <b>Descripción</b>         | Controla la posición en la que se muestra la imagen en el fondo del elemento  |



## FONDO - IMAGEN DE FONDO

Para controlar la manera de visualizar la imagen de fondo:

| <b>background-attachment</b> | Comportamiento de la imagen de fondo   |
|------------------------------|--|
| <b>Valores</b>               | scroll   fixed   inherit   |
| <b>Se aplica a</b>           | Todos los elementos  |
| <b>Valor inicial</b>         | scroll   |
| <b>Descripción</b>           | Controla la forma en la que se visualiza la imagen de fondo: permanece fija cuando se hace scroll en la ventana del navegador o se desplaza junto con la ventana |

## FONDO - RESUMIDA

Establecer todas las propiedades de fondo:

|                      |   |
|----------------------|---|
| <b>background</b>    | Fondo de un elemento  |
| <b>Valores</b>       | ( <background-color>    <background-image>    <background-repeat>    <background-attachment>    <background-position> )   inherit |
| <b>Se aplica a</b>   | Todos los elementos   |
| <b>Valor inicial</b> | -   |
| <b>Descripción</b>   | Establece todas las propiedades del fondo de un elemento  |

## TIPOGRAFÍA EN CSS

### TIPOGRAFÍA - RESUMIDA

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.

|                      |   |
|----------------------|---|
| <b>color</b>         | Color del texto                                     |
| <b>Valores</b>       | <color>   inherit                                   |
| <b>Se aplica a</b>   | Todos los elementos                                 |
| <b>Valor inicial</b> | Depende del navegador                               |
| <b>Descripción</b>   | Establece el color de letra utilizado para el texto |

### TIPOGRAFÍA - FUENTE

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.

|                      |  |
|----------------------|--|
| <b>font-family</b>   | Tipo de letra  |
| <b>Valores</b>       | (( <nombre_familia>   <familia_generica> ) (,<nombre_familia>   <familia_generica>)* )   inherit |
| <b>Se aplica a</b>   | Todos los elementos  |
| <b>Valor inicial</b> | Depende del navegador  |
| <b>Descripción</b>   | Establece el tipo de letra utilizado para el texto   |

## TIPOGRAFÍA – TAMAÑO

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

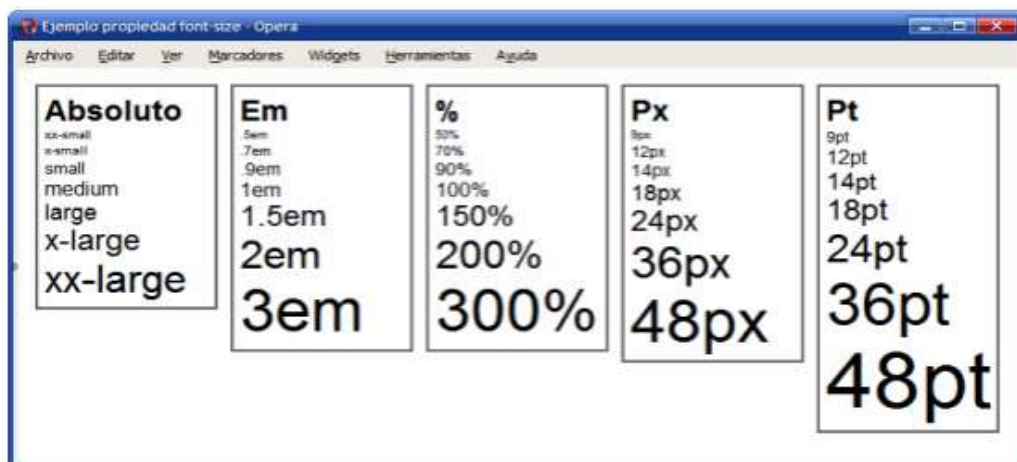
| font-size     | Tamaño de letra   |
|---------------|---|
| Valores       | <tamaño_absoluto>   <tamaño_relativo>   <medida>   <porcentaje>   inherit |
| Se aplica a   | Todos los elementos   |
| Valor inicial | medium  |
| Descripción   | Establece el tamaño de letra utilizado para el texto                      |

## TIPOGRAFÍA – TAMAÑO

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.



## TIPOGRAFÍA – GROSOR

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.

| font-weight   | Anchura de la letra  |
|---------------|--|
| Valores       | normal   bold   bolder   lighter   100   200   300   400   500   600   700   800   900   inherit |
| Se aplica a   | Todos los elementos  |
| Valor inicial | normal   |
| Descripción   | Establece la anchura de la letra utilizada para el texto   |

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.

|                   |   |
|-------------------|---|
| <b>font-style</b> | Estilo de la letra                                      |
| Valores           | normal   italic   oblique   inherit                     |
| Se aplica a       | Todos los elementos                                     |
| Valor inicial     | normal  |
| Descripción       | Establece el estilo de la letra utilizada para el texto |

## FORMATEO DE TEXTO CON CSS

### TEXTO - ALINEACIÓN

Para establecer la alineación del contenido del elemento. La propiedad text-align no sólo alinea el texto que contiene un elemento, sino que también alinea todos sus contenidos, como por ejemplo las imágenes.

|                   |  |
|-------------------|--|
| <b>text-align</b> | Alineación del texto                               |
| Valores           | left   right   center   justify   inherit          |
| Se aplica a       | Elementos de bloque y celdas de tabla              |
| Valor inicial     | left   |
| Descripción       | Establece la alineación del contenido del elemento |

### TEXTO - INTERLINEADO

El interlineado de un texto se controla mediante la propiedad line-height, que permite controlar la altura ocupada por cada línea de texto:

|                    |  |
|--------------------|--|
| <b>line-height</b> | Interlineado   |
| Valores            | normal   <numero>   <medida>   <porcentaje>   inherit  |
| Se aplica a        | Todos los elementos                                    |
| Valor inicial      | normal   |
| Descripción        | Permite establecer la altura de línea de los elementos |

## TEXTO - DECORACIÓN

El valor **underline** subraya el texto. El valor **overline** añade una línea en la parte superior del texto. El valor **line-through** muestra el texto tachado con una línea continua. El valor **blink** muestra el texto parpadeante.

| <b>text-decoration</b> | Decoración del texto   |
|------------------------|--|
| Valores                | <code>none   ( underline    overline    line-through    blink )   inherit</code> |
| Se aplica a            | Todos los elementos  |
| Valor inicial          | <code>none</code>  |
| Descripción            | Establece la decoración del texto (subrayado, tachado, parpadeante, etc.)        |

## TEXTO - TRANSFORMACIÓN

El valor `capitalize` transforma a mayúsculas la primera letra de las palabras del texto.

El valor `uppercase` transforma a mayúsculas todo el texto.

El valor `lowercase` transforma a minúsculas todo el texto.

| <b>text-transform</b> | Transformación del texto  |
|-----------------------|---|
| Valores               | <code>capitalize   uppercase   lowercase   none   inherit</code>              |
| Se aplica a           | Todos los elementos   |
| Valor inicial         | <code>none</code>   |
| Descripción           | Transforma el texto original (lo transforma a mayúsculas, a minúsculas, etc.) |

## IMÁGENES EN CSS

### ATRIBUTOS HEIGHT Y WIDTH

Estos atributos definen la altura y anchura respectivamente de la imagen en píxeles. Aunque estas dimensiones forman parte del estilo de la imagen, y por tanto podrían ir en el CSS, todavía puede ser interesante definir las dentro del HTML. Esto, ya no es tan indispensable, puesto que muchos sitios creados con *"Responsive Web Design"* prefieren que las imágenes se adapten al tamaño de la pantalla donde se va a visualizar.

Todos los archivos gráficos poseen unas dimensiones de ancho y alto. Estas dimensiones pueden obtenerse a partir del propio diseñador gráfico o bien haciendo clic con el botón derecho sobre la imagen, vista desde el explorador de archivos de tu ordenador, para luego elegir "propiedades" o "información de la imagen" sobre el menú que se despliega.

Un ejemplo de etiqueta `IMG` con sus valores de anchura y altura declarados te quedaría así:

```

```

## IMÁGENES – ANCHO / ALTURA

Utilizando las propiedades width y height, es posible mostrar las imágenes con cualquier altura/anchura, independientemente de su altura/anchura real:

```
#destacada {  
    width: 120px;  
    height: 250px;  
}  
  

```

No obstante, si se utilizan alturas/anchuras diferentes de las reales, el navegador deforma las imágenes y el resultado estético es muy desagradable.

## IMÁGENES – BORDES

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;  
}
```

## LISTAS EN CSS

### LISTAS – VIÑETAS PERSONALIZADAS

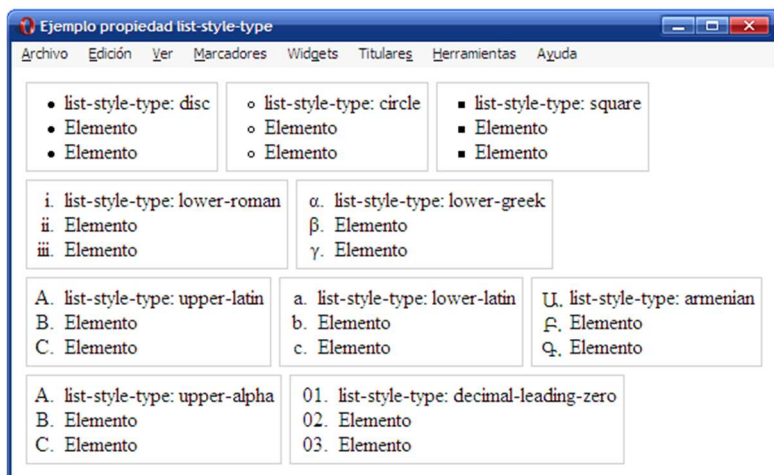
Por defecto, los navegadores muestran los elementos de las listas no ordenadas con una viñeta formada por un pequeño círculo de color negro. Los elementos de las listas ordenadas se muestran por defecto con la numeración decimal utilizada en la mayoría de países.

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.

| Propiedad     | list-style-type  |
|---------------|--|
| 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   <a href="#">inherit</a> |
| Se aplica a   | Elementos de una lista   |
| Valor inicial | disc   |
| Descripción   | Permite establecer el tipo de viñeta mostrada para una lista   |



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

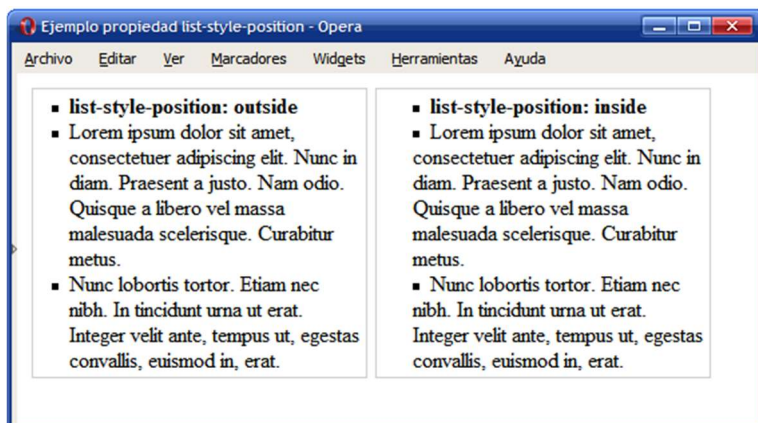


## LISTAS – VIÑETAS POSICIÓN

La propiedad list-style-position permite controlar la colocación de las viñetas.

| Propiedad     | <b>list-style-position</b>  |
|---------------|---|
| Valores       | inside   outside   <a href="#">inherit</a>                                |
| Se aplica a   | Elementos de una lista  |
| Valor inicial | outside   |
| Descripción   | Permite establecer la posición de la viñeta de cada elemento de una lista |

La diferencia entre los valores outside y inside se hace evidente cuando los elementos contienen mucho texto, como en la siguiente imagen:



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.

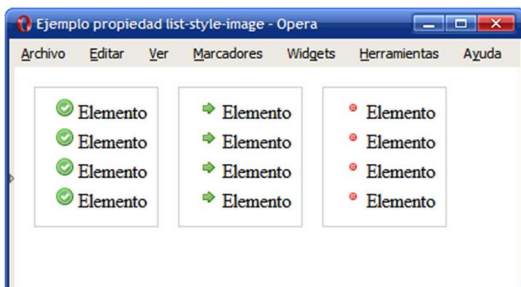
## LISTAS – VIÑETAS ASPECTO

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.

|                      |   |
|----------------------|---|
| <b>Propiedad</b>     | <b>list-style-image</b>   |
| <b>Valores</b>       | url   none   <a href="#">inherit</a>                                    |
| <b>Se aplica a</b>   | Elementos de una lista  |
| <b>Valor inicial</b> | none  |
| <b>Descripción</b>   | Permite reemplazar las viñetas automáticas por una imagen personalizada |

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:



Las reglas CSS correspondientes al ejemplo anterior se muestran a continuación:

```
ul.ok {  
  list-style-image: url("imagenes/ok.png");  
}  
  
ul.flecha {  
  list-style-image: url("imagenes/flecha.png");  
}  
  
ul.circulo {  
  list-style-image: url("imagenes/circulo_rojo.png");  
}
```



## LISTAS – MENÚ VERTICAL

Los sitios web correctamente diseñados emplean las listas de elementos para crear todos sus menús de navegación. Utilizando la etiqueta `<ul>` de HTML se agrupan todas las opciones del menú y haciendo uso de CSS se modifica su aspecto para mostrar un menú horizontal o vertical.

A continuación, se muestra la transformación de una lista sencilla de enlaces en un menú vertical de navegación.

`<ul>`

`<li><a href="#">Elemento 1</a></li>`

`<li><a href="#">Elemento 2</a></li>`

`<li><a href="#">Elemento 3</a></li>`

`<li><a href="#">Elemento 4</a></li>`

`<li><a href="#">Elemento 5</a></li>`

`<li><a href="#">Elemento 6</a></li>`

`</ul>`

Aspecto final del menú vertical:



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

- 1) Definir la anchura del menú:

```
ul.menu { width: 180px; }
```

- 2) Eliminar las viñetas automáticas y todos los márgenes y espaciados aplicados por defecto:

```
ul.menu {  
    list-style: none;  
    margin: 0;  
    padding: 0;  
    width: 180px;  
}
```

- 3) Añadir un borde al menú de navegación y establecer el color de fondo y los bordes de cada elemento del menú:

```
ul.menu {  
    border: 1px solid #7C7C7C;  
    border-bottom: none;  
    list-style: none;  
    margin: 0;  
    padding: 0;  
    width: 180px;  
}  
  
ul.menu li {  
    background: #F4F4F4;  
    border-bottom: 1px solid #7C7C7C;  
    border-top: 1px solid #FFF;  
}
```

- 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:

```
ul.menu li a {  
    color: #333;  
    display: block;  
    padding: .2em 0 .2em .5em;  
    text-decoration: none;  
}
```

## TABLAS EN CSS

### ATRIBUTOS PARA TABLAS, FILAS Y CELDAS

A partir de esta idea simple y sencilla, las tablas adquieren otra magnitud cuando les incorporamos toda una cantidad de atributos aplicados sobre cada tipo de etiquetas que las componen.

En cuanto a atributos para tabla hay unos cuantos. Muchos los conoces ya de otras etiquetas, como width, height, align, etc. Hay otros que son especialmente creados para las etiquetas TABLE.

- **cellspacing:** es el espacio entre celdas de la tabla.
- **cellpadding:** es el espacio entre el borde de la celda y su contenido.
- **border:** es el número de píxeles que tendrá el borde de la tabla.
- **bordercolor:** es el rgb que le vas a asignar al borde de la tabla.

En cuanto a las etiquetas "interiores" de una tabla, nos referimos a TR y TD, ten en cuenta:

- Podemos usar prácticamente cualquier tipo de etiqueta dentro de la etiqueta TD para, de esta forma, escribir su contenido.
- Las etiquetas situadas en el interior de la celda no modifican el resto del documento.
- Las etiquetas de fuera de la celda no son tenidas en cuenta por ésta.

Así pues, podemos especificar el formato de nuestras celdas a partir de etiquetas introducidas en su interior o mediante atributos colocados dentro de la etiqueta de celda TD o bien, en algunos casos, dentro de la etiqueta TR, si deseamos que el atributo sea válido para toda la línea. La forma más útil y actual de dar forma a las celdas es a partir de las hojas de estilo en cascada que ya tendréis la oportunidad de abordar más adelante.

Veamos a continuación algunos atributos útiles para la construcción de nuestras tablas. Empecemos viendo atributos que nos permiten modificar una celda en concreto o toda una línea:

- **align:** Justifica el texto de la celda del mismo modo que si fuese el de un párrafo.
- **valign:** Podemos elegir si queremos que el texto aparezca arriba (top), en el centro (middle) o abajo (bottom) de la celda.
- **bgcolor:** Da color a la celda o línea elegida.
- **bordercolor:** Define el color del borde.

Otros atributos que pueden ser únicamente asignados a una celda y no al conjunto de celdas de una línea son:

- **background:** Nos permite colocar un fondo para la celda a partir de un enlace o una imagen.
- **height:** Define la altura de la celda en píxeles o porcentaje.

- **width:** Define la anchura de la celda en pixeles o porcentaje.
- **colspan:** Expande una celda horizontalmente.
- **rowspan:** Expande una celda verticalmente.

Estos últimos cuatro atributos descritos son de gran utilidad. Concretamente, height y width nos ayudan a definir las dimensiones de nuestras celdas de una forma absoluta (en pixeles o puntos de pantalla) o de una forma relativa, es decir por porcentajes referidos al tamaño total de la tabla.

Los atributos rowspan y colspan son también utilizados frecuentemente. Gracias a ellos es posible expandir celdas fusionando éstas con sus vecinas. El valor que pueden tomar estas etiquetas es numérico. El número representa la cantidad de celdas fusionadas.

Así:

```
<td colspan="2">
```

Esta celda tiene un colspan 2

Celda 1, línea 2 Celda 2, línea 2

Por otro lado:

```
<td rowspan="2">
```

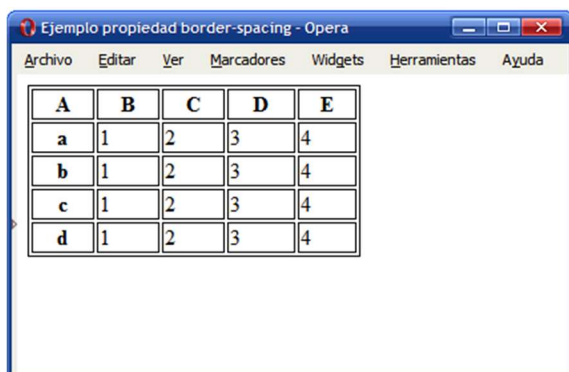
Esta celda tiene rowspan="2",  
por eso tiene fusionada la  
celda de abajo.

Celda  
Normal  
Otra  
celda  
normal

El resto de los atributos presentados presentan una utilidad y uso bastante obvios. Los dejamos a vuestra propia investigación.

## TABLAS – BORDES CELDAS

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:



El código HTML y CSS del ejemplo anterior se muestra a continuación:

#### CSS.

```
.normal {  
    width: 250px;  
    border: 1px solid #000;  
}  
  
.normal th, .normal td {  
    border: 1px solid #000;  
}
```

#### HTML.

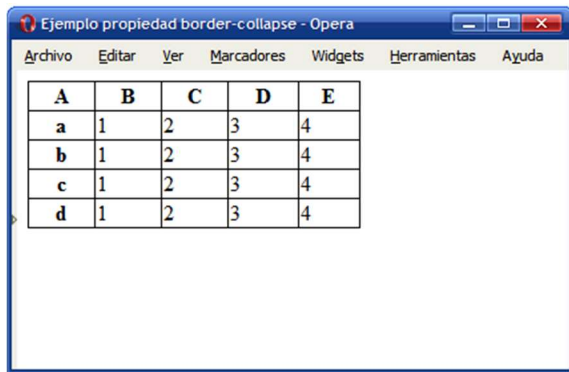
```
<table class="normal" summary="Tabla genérica">  
    <tr>  
        <th scope="col">A</th>  
        <th scope="col">B</th>  
        <th scope="col">C</th>  
        <th scope="col">D</th>  
        <th scope="col">E</th>  
    </tr>  
</table>
```

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:

|               |   |
|---------------|---|
| Propiedad     | <b>border-collapse</b>  |
| Valores       | collapse   separate   <a href="#">inherit</a>                                     |
| Se aplica a   | Todas las tablas  |
| Valor inicial | separate  |
| Descripción   | Define el mecanismo de fusión de los bordes de las celdas adyacentes de una tabla |

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.

Ejemplo 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 código completo del ejemplo anterior se muestra a continuación:

### CSS.

```
.normal {  
    width: 250px;  
    border: 1px solid #000;  
    border-collapse: collapse;  
}  
  
.normal th, .normal td {  
    border: 1px solid #000;  
}
```

### HTML.

```
<table class="normal" summary="Tabla genérica">  
<tr>  
    <th scope="col">A</th>  
    <th scope="col">B</th>  
    <th scope="col">C</th>  
    <th scope="col">D</th>  
    <th scope="col">E</th>  
</tr>  
</table>
```

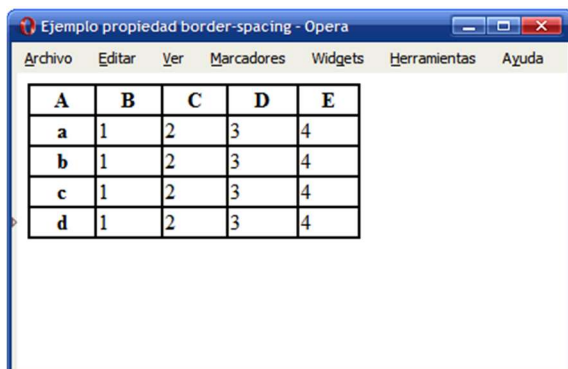
## Border-spacing

Si se opta por el modelo *separate* (que es el que se aplica si no se indica lo contrario) se puede utilizar la propiedad `border-spacing` para controlar la separación entre los bordes de cada celda.

|                      |   |
|----------------------|---|
| <b>Propiedad</b>     | <b>border-spacing</b>   |
| <b>Valores</b>       | <a href="#">unidad de medida</a> <a href="#">unidad de medida</a> ?   <a href="#">inherit</a> |
| <b>Se aplica a</b>   | Todas las tablas  |
| <b>Valor inicial</b> | 0   |
| <b>Descripción</b>   | Establece la separación entre los bordes de las celdas adyacentes de una tabla                |

Si solamente se indica como valor una medida, se asigna ese valor como separación horizontal y vertical. Si se indican dos medidas, la primera es la separación horizontal y la segunda es la separación vertical entre celdas.

La propiedad `border-spacing` sólo controla la separación entre celdas y por tanto, no se puede utilizar para modificar el tipo de modelo de bordes que se utiliza. En concreto, si se establece un valor igual a 0 para la separación entre los bordes de las celdas, el resultado es muy diferente al modelo *collapse*:



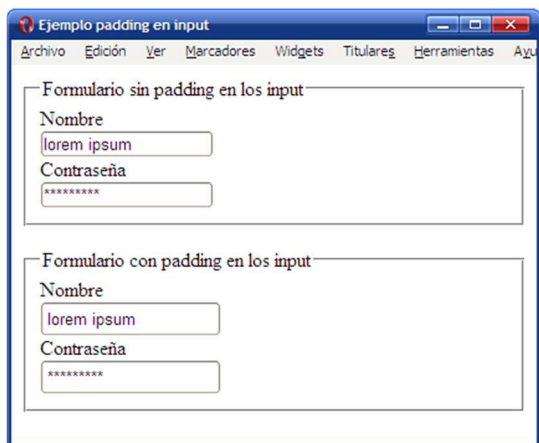
| A | B | C | D | E |
|---|---|---|---|---|
| a | 1 | 2 | 3 | 4 |
| b | 1 | 2 | 3 | 4 |
| c | 1 | 2 | 3 | 4 |
| d | 1 | 2 | 3 | 4 |

## FORMULARIOS EN CSS

### FORMULARIOS – CAMPOS DE TEXTO

Por defecto, los campos de texto de los formularios no incluyen ningún espacio de relleno, por lo que el texto introducido por el usuario aparece pegado a los bordes del cuadro de texto.

Añadiendo un pequeño padding a cada elemento `<input>`, se mejora notablemente el aspecto del formulario:

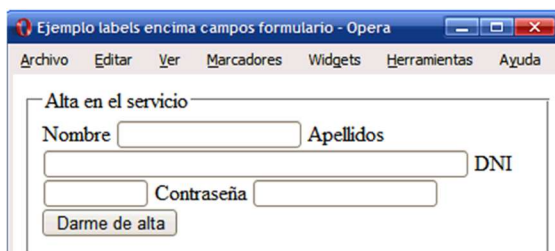


La regla CSS necesaria para mejorar el formulario es muy sencilla:

```
form.elegante input {  
    padding: .2em;  
}
```

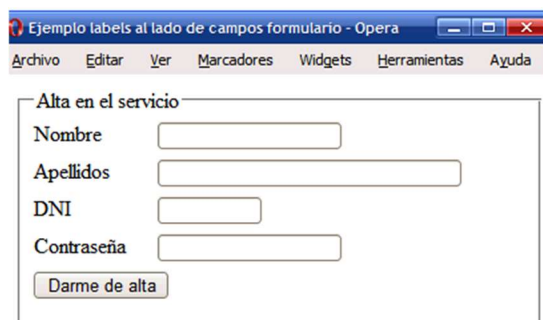
### FORMULARIOS – LABELS ALINEADAS Y FORMATEADAS

Los elementos `<input>` y `<label>` de los formularios son elementos en línea, por lo que el aspecto que muestran los formularios por defecto, es similar al de la siguiente imagen:

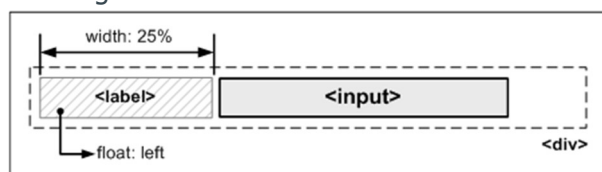


Aprovechando los elementos `<label>`, se pueden aplicar unos estilos CSS sencillos que permitan mostrar el formulario con el aspecto de la siguiente imagen:





Para mostrar un formulario tal y como aparece en la imagen anterior no es necesario crear una tabla y controlar la anchura de sus columnas para conseguir una alineación perfecta. Sin embargo, sí que es necesario añadir un nuevo elemento (por ejemplo un `<div>`) que encierre a cada uno de los campos del formulario (`<label>` y `<input>`). El esquema de la solución propuesta es el siguiente:



Por tanto, en el código HTML del formulario se añaden los elementos `<div>`:

```
<form>
```

```
  <fieldset>
```

```
    <legend>Alta en el servicio</legend>
```

```
    <div>
```

```
      <label for="nombre">Nombre</label>
```

```
      <input type="text" id="nombre" />
```

```
    </div>
```

```
  <div>
```

```
    <label for="apellidos">Apellidos</label>
```

```
    <input type="text" id="apellidos" size="35" />
```

```
  </div>
```

```
</fieldset>
```

```
</form>
```

Y en el código CSS se añaden las reglas necesarias para alinear los campos del formulario:

```
div { margin: .4em 0; }
```

```
div label { width: 25%; float: left; }
```