

INTRODUCCIÓN A CSS

3. El modelo de cajas

CONTENIDO

1. Introducción	1
2. Descomposición en capas de una caja	2
3. Propiedades de las cajas	4
3.1. Anchura	4
3.2. Altura	5
3.3. Margen	6
3.4. Relleno	10
3.5. Bordes	12
3.6. Color e imagen de fondo	15
4. El tamaño de la caja visualizada	20
5. El modo 'quirks'	20

1. Introducción

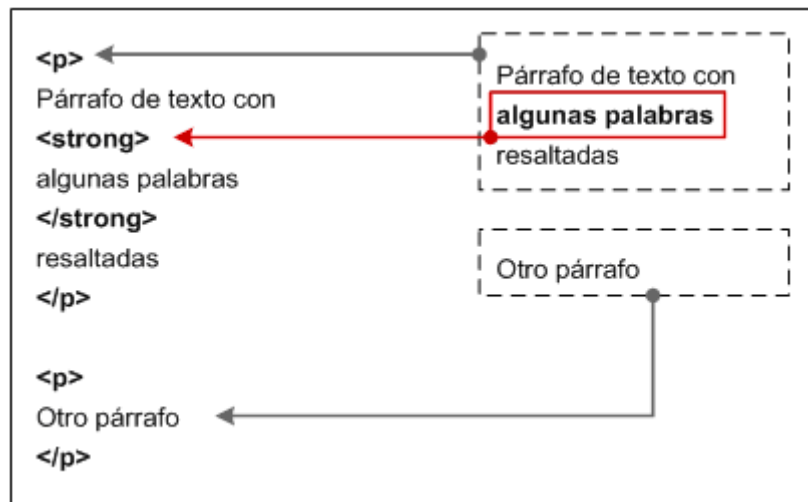
El modelo de cajas o “box model” es el comportamiento de CSS que hace que todos los elementos incluidos en una página HTML se representen mediante cajas rectangulares. CSS permite controlar el aspecto de cada caja.

Es posible definir la altura y anchura de cada caja, el margen existente entre las cajas y el espaciado de relleno entre el contenido de la caja y su borde. Además CSS permite controlar la forma en la que se visualizan las cajas, permitiendo ocultarlas o desplazarlas respecto a su posición original.

Como la mayoría de etiquetas no muestran un color de fondo ni un borde, las cajas no son visibles a simple vista. No obstante, si incluyésemos los estilos necesarios para mostrar los bordes, veríamos algo similar a esto:



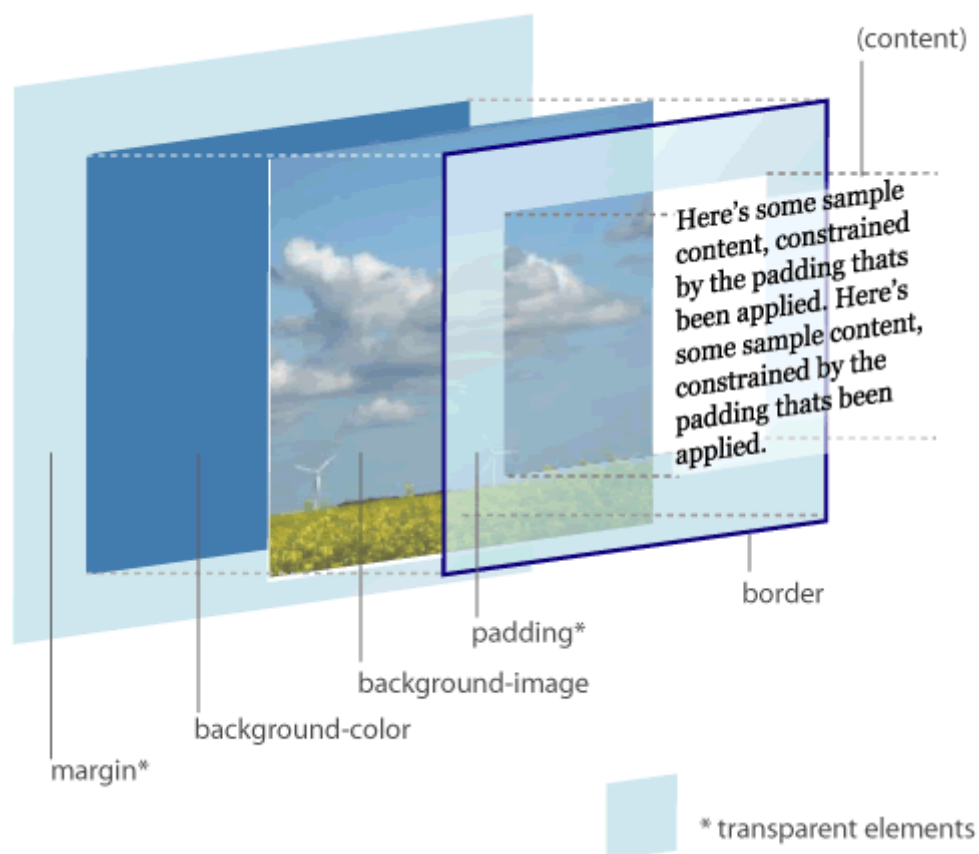
Las cajas correspondientes a cada etiqueta se crean automáticamente al insertar la etiqueta en el código fuente:



2. Descomposición en capas de una caja

Cada caja correspondiente a una etiqueta está formada por seis capas superpuestas, tal y como se muestra en la siguiente figura:

THE CSS BOX MODEL HIERARCHY



Esta es la descripción de cada una de estas capas y su orden de visualización:

- **Contenido (*content*):** se trata del contenido HTML del elemento (las palabras de un párrafo, una imagen, el texto de una lista de elementos, etc.)
- **Relleno (*padding*):** espacio libre opcional entre el contenido y el borde que lo encierra.
- **Borde (*border*):** línea que encierra completamente el contenido y su relleno.
- **Imagen de fondo (*background image*):** imagen que se muestra por detrás del contenido y el espacio de relleno.
- **Color de fondo (*background color*):** color que se muestra por detrás del contenido y el espacio de relleno.
- **Margen (*margin*):** espacio libre entre la caja y las posibles cajas adyacentes.

El relleno y el margen son transparentes, por lo que en el espacio ocupado por el relleno se muestra el color o imagen de fondo (si están definidos) y en el espacio ocupado por el margen se muestra el color o imagen de fondo de su elemento padre (si están definidos). Si ningún elemento padre tiene definido un color o imagen de fondo, se muestra el color o imagen de fondo de la propia página (si están definidos).

Si una caja define tanto un color como una imagen de fondo, la imagen tiene más prioridad y es la que se visualiza. No obstante, si la imagen de fondo no cubre totalmente la caja del elemento o si la imagen tiene zonas transparentes, también se visualiza el color de fondo.

3. Propiedades de las cajas

Las propiedades que permiten modificar la apariencia de las cajas son el ancho y el alto del contenido, el relleno, el margen, el borde, el color y la imagen de fondo.

3.1. Anchura

La propiedad ***width*** permite controlar el ancho de la caja. Tiene las siguientes características:

<i>width</i>	Establece el ancho de la caja
Se aplica a:	<p>Todas las etiquetas excepto:</p> <ul style="list-style-type: none"> • Elementos en línea (aunque sí es aplicable a imágenes) • Filas y grupos de filas de una tabla
Valores admitidos:	<p><unidades relativas o absolutas> auto inherit</p> <p>No admite valores negativos.</p> <p>Si se especifican valores en porcentaje, dicho porcentaje es relativo al ancho del elemento padre.</p> <p>El valor auto, utilizado por defecto, indica que el navegador debe calcular automáticamente la anchura del elemento, teniendo en cuenta sus contenidos y el sitio disponible en la página.</p> <p>El valor inherit indica que la anchura del elemento se hereda de su elemento padre</p>

El siguiente ejemplo establece el valor de la anchura de una etiqueta <div>:

```
[...]
.lateral      { width: 200px; }
[...]
```

```
<div class="lateral">
  ...
</div>
```

CSS define también las propiedades ***min-width*** y ***max-width*** que veremos más adelante.

3.2. Altura

La propiedad **height** controla el alto de la caja. Tiene las siguientes características:

height	Establece el alto de la caja
Se aplica a:	Todas las etiquetas excepto : <ul style="list-style-type: none">• Elementos en línea (aunque sí es aplicable a imágenes)• Columnas y grupos de columnas de una tabla
Valores admitidos:	<unidades relativas o absolutas> auto inherit No admite valores negativos. Si se especifican valores en porcentaje, dicho porcentaje es relativo al ancho del elemento padre. El valor auto , utilizado por defecto, indica que el navegador debe calcular automáticamente la altura del elemento, teniendo en cuenta sus contenidos y el sitio disponible en la página. El valor inherit indica que la altura del elemento se hereda de su elemento padre.

El siguiente ejemplo establece el valor de la altura de una etiqueta <div>:

```
[...]  
.lateral      { height: 200px; }  
[...]  
  
<div class="lateral">  
    ...  
</div>
```

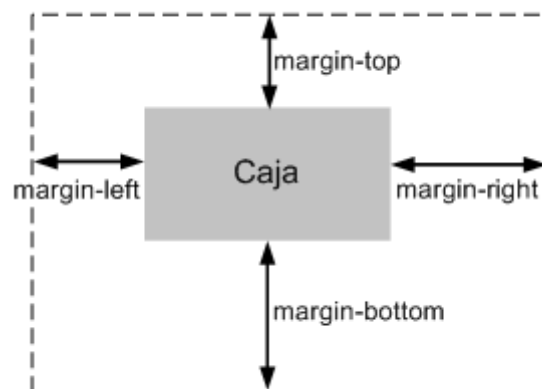
CSS define también las propiedades **min-height** y **max-height** que veremos más adelante.

3.3. Margen

El margen de una caja se establece mediante las propiedades **margin** siguientes:

margin-top	Establece el margen superior de la caja
margin-bottom	Establece el margen inferior de la caja
margin-left	Establece el margen izquierdo de la caja
margin-right	Establece el margen derecho de la caja
Shorthand	margin <[margin_sup] [margin_inf] [margin_der] [margin_der]>
Se aplica a:	Todas las etiquetas con las siguientes excepciones: <ul style="list-style-type: none"> <i>margin-top</i> y <i>margin-bottom</i> sólo se aplican a elementos de bloque y a imágenes.
Valores admitidos:	<unidades relativas o absolutas> auto inherit ¿admite valores negativos? Si se especifican valores en porcentaje, dicho porcentaje es relativo al ancho del elemento padre. El valor auto , utilizado por defecto, indica que el navegador debe calcular automáticamente la altura del elemento, teniendo en cuenta sus contenidos y el sitio disponible en la página. El valor inherit indica que la altura del elemento se hereda de su elemento padre.

Las unidades más utilizadas para indicar los márgenes de un elemento son los píxeles (cuando se requiere una precisión total), los em (para hacer diseños que mantengan las proporciones) y los porcentajes (para hacer diseños líquidos o fluidos).



El siguiente ejemplo añade un margen izquierdo al segundo párrafo:

3. El modelo de cajas

```
[...]
.destacado      { margin-left: 2em; }
[...]
```

<p> Este párrafo no tiene estilos aplicados.

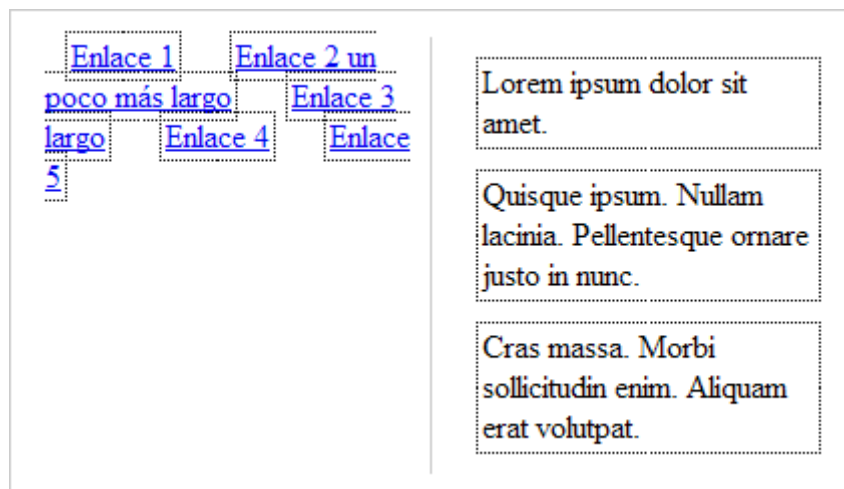
</p>

<p class="destacado">

 Este párrafo tiene definido un margen de 2em.

</p>

Los márgenes verticales (*margin-top* y *margin-bottom*) sólo se pueden aplicar a los elementos de bloque y las imágenes, mientras que los márgenes laterales (*margin-left* y *margin-right*) se pueden aplicar a cualquier elemento, tal y como muestra la siguiente imagen:



La imagen anterior muestra el resultado de aplicar los mismos márgenes a varios enlaces (elementos en línea) y varios párrafos (elementos de bloque). En los elementos en línea los márgenes verticales no tienen ningún efecto, por lo que los enlaces no muestran ninguna separación vertical, al contrario de lo que sucede con los párrafos. Sin embargo, los márgenes laterales funcionan sobre cualquier tipo de elemento, por lo que los enlaces se muestran separados entre sí y los párrafos aumentan su separación con los bordes laterales de su elemento contenedor.

Además de las cuatro propiedades que controlan cada uno de los márgenes del elemento, CSS define una propiedad que permite establecer los cuatro márgenes de forma directa empleando una única propiedad denominada ***margin***. Este tipo de propiedades resumidas se denominan propiedades de tipo "***shorthand***" y CSS define varias propiedades de este tipo como veremos más adelante.

La sintaxis del shorthand admite entre uno y cuatro valores, con el siguiente significado:

- Si solo se indica un valor, todos los márgenes tienen ese valor.

3. El modelo de cajas

- Si se indican dos valores, el primero se asigna al margen superior e inferior y el segundo se asigna a los márgenes izquierdo y derecho.
- Si se indican tres valores, el primero se asigna al margen superior, el tercero se asigna al margen inferior y el segundo valor se asigna los márgenes izquierdo y derecho.
- Si se indican los cuatro valores, el orden de asignación es: margen superior, margen derecho, margen inferior y margen izquierdo.

Por ejemplo si tomamos el siguiente código:

```
div img
{
  margin-top: .5em;
  margin-bottom: .5em;
  margin-left: 1em;
  margin-right: .5em;
}
```

podríamos construir una alternativa equivalente así:

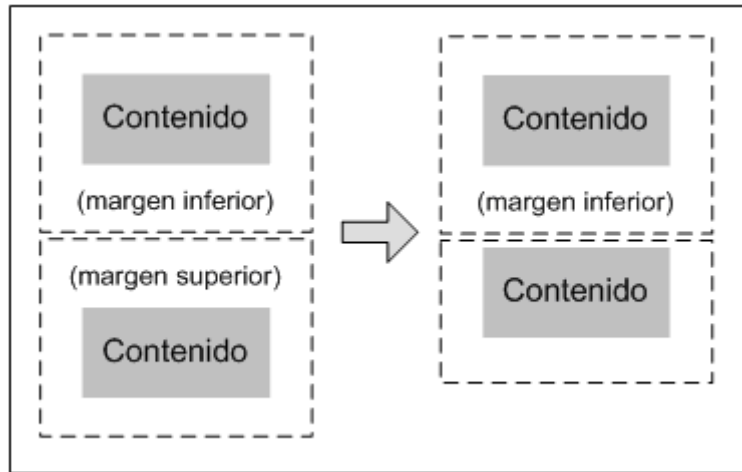
```
div img
{
  margin: .5em .5em .5em 1em;
}
```

o bien podríamos hacerlo así:

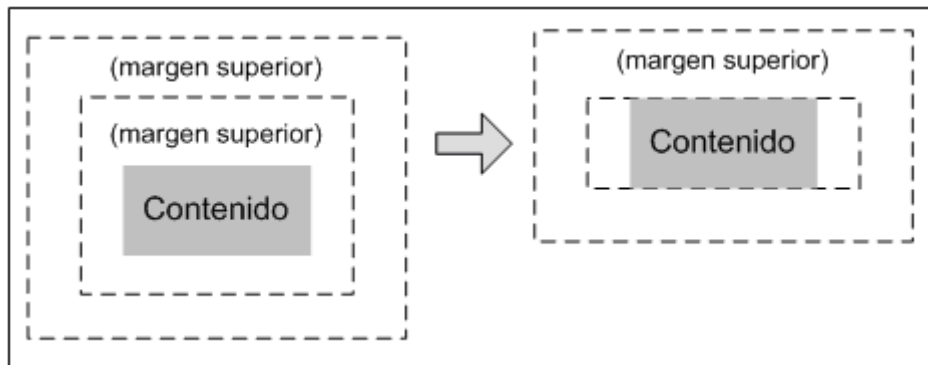
```
div img
{
  margin: .5em;
  margin-left: 1em;
}
```

El comportamiento de los márgenes verticales es más complejo de lo que se puede prever en principio. Cuando se juntan dos o más márgenes verticales, se fusionan de forma automática y la altura del nuevo margen será igual a la altura del margen más alto de los que se han fusionado.

3. El modelo de cajas



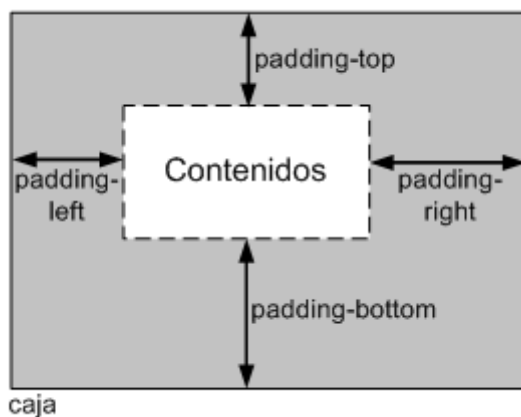
De la misma forma, si un elemento está contenido dentro de otro elemento, sus márgenes verticales se fusionan y resultan en un nuevo margen de la misma altura que el mayor margen de los que se han fusionado:



3.4. Relleno

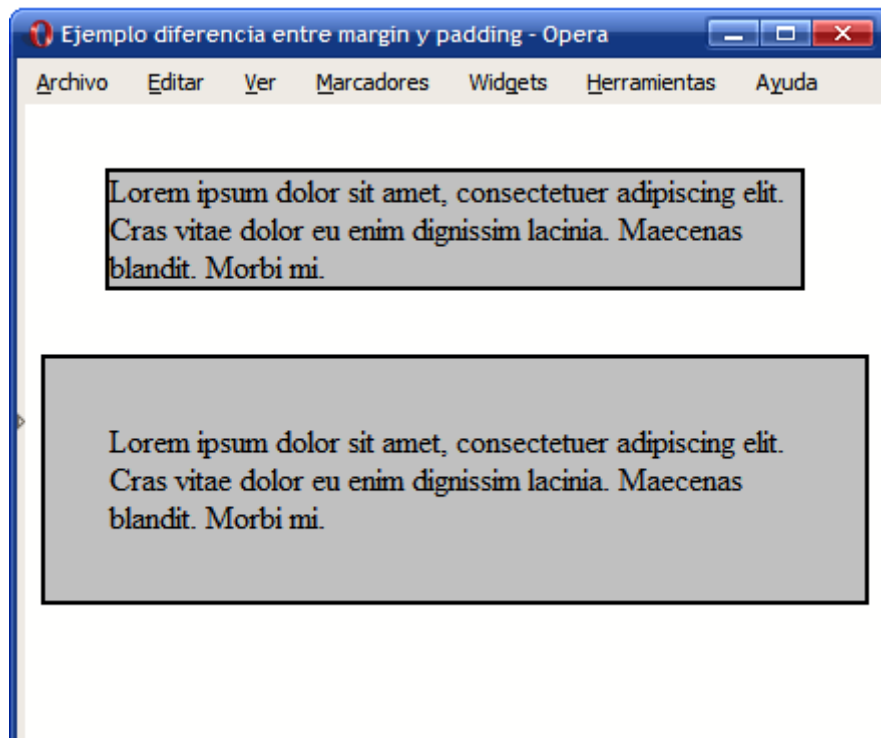
Para establecer el relleno se utiliza la propiedad padding:

padding-top	Establece el relleno superior de la caja (espacio entre el contenido y el borde superior de la caja)
padding-bottom	Establece el relleno inferior de la caja (espacio entre el contenido y el borde inferior de la caja)
padding-left	Establece el relleno izquierdo de la caja (espacio entre el contenido y el borde izquierdo de la caja)
padding-right	Establece el relleno derecho de la caja (espacio entre el contenido y el borde derecho de la caja)
Shorthand	padding <[margin_sup] [margin_inf] [margin_der] [margin_der]>
Se aplica a:	Todas las etiquetas con las siguientes excepciones: <ul style="list-style-type: none"> Algunos elementos de tablas como grupos de cabeceras y grupos de pies de tabla.
Valores admitidos:	<unidades relativas o absolutas> auto inherit ¿admite valores negativos? Si se especifican valores en porcentaje, dicho porcentaje es relativo al ancho del elemento padre. El valor auto , utilizado por defecto, indica que el navegador debe calcular automáticamente la altura del elemento, teniendo en cuenta sus contenidos y el sitio disponible en la página. El valor inherit indica que la altura del elemento se hereda de su elemento padre.



La siguiente imagen muestra la diferencia entre el margen y el relleno de los elementos:

3. El modelo de cajas



El código HTML / CSS del ejemplo se muestra a continuación:

```
.margin
{
    margin-top: 2em;
    margin-right: 2em;
    margin-bottom: 2em;
    margin-left: 2em;
}

.relleno
{
    padding-top: 2em;
    padding-right: 2em;
    padding-bottom: 2em;
    padding-left: 2em;
}

[...]
```

```
<p class="margin">
    Lorem ipsum dolor sit amet, consectetur adipiscing elit.
    Cras vitae dolor eu enim dignissim lacinia. Maecenas
    blandit. Morbi mi.
</p>

<p class="relleno">
    Lorem ipsum dolor sit amet, consectetur adipiscing
    elit.Cras vitae dolor eu enim dignissim lacinia.
    Maecenas blandit. Morbi mi.
</p>
```

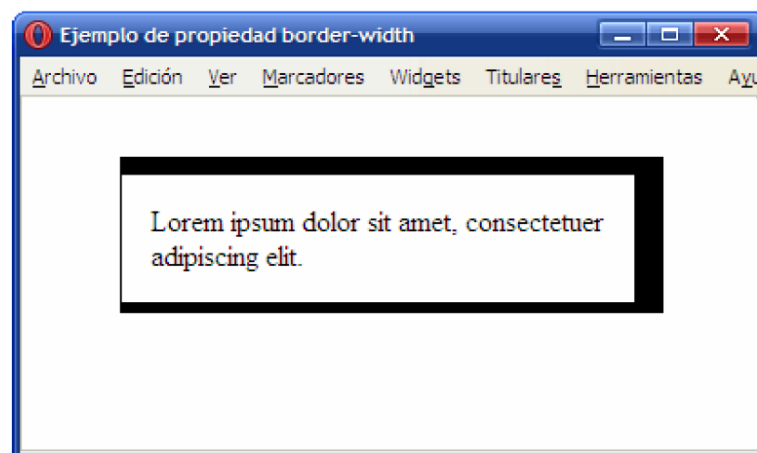
3.5. Bordes

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 ancho, color y estilo.

border-top-width	Establece el ancho del borde superior.
border-bottom-width	Establece el ancho del borde inferior.
border-left-width	Establece el ancho del borde izquierdo.
border-right-width	Establece el ancho del borde derecho.
Shorthand	border-width: <[borde_sup][borde_inf][borde_der] [borde_der]>
Se aplica a:	Todas las etiquetas
Valores admitidos:	<unidades relativas o absolutas> thin medium thick inherit

La medida más habitual para especificar el ancho del borde suele ser el pixel, ya que permite controlar con precisión el grosor. Las palabras clave *thin*, *medium* y *thick* apenas se usan puesto que el estándar no define con precisión cuál es su grosor y se visualizan de manera distinta en los navegadores.

El siguiente ejemplo muestra un div con bordes de diferentes anchos:



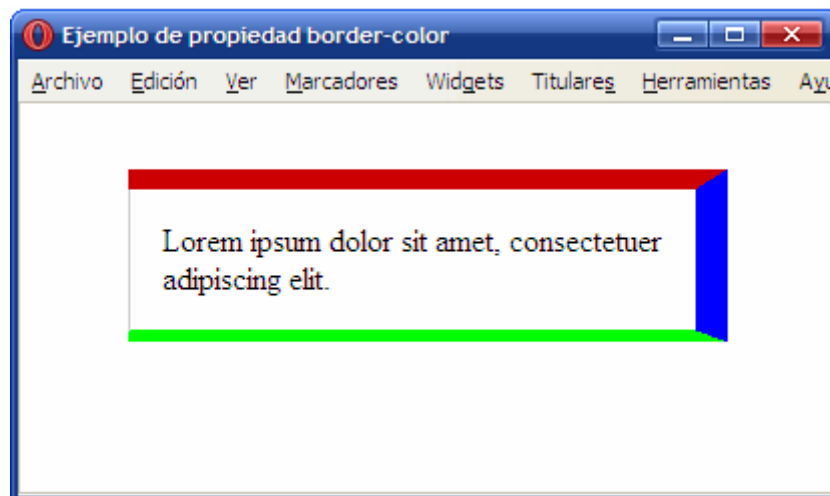
Las reglas utilizadas son las siguientes:

```
div
{
border-top-width: 10px;
border-right-width: 1em;
border-bottom-width: thick;
border-left-width: thin;
}
```

Otra de las propiedades que puede modificarse es el color del borde:

border-top-color	Establece el color del borde superior.
border-bottom-color	Establece el color del borde inferior.
border-left-color	Establece el color del borde izquierdo.
border-right-color	Establece el color del borde derecho.
Shorthand	border-color: <[borde_sup][borde_inf][borde_der] [borde_der]>
Se aplica a:	Todas las etiquetas
Valores admitidos:	<color> transparent inherit

El siguiente ejemplo muestra un div con bordes de distintos colores:



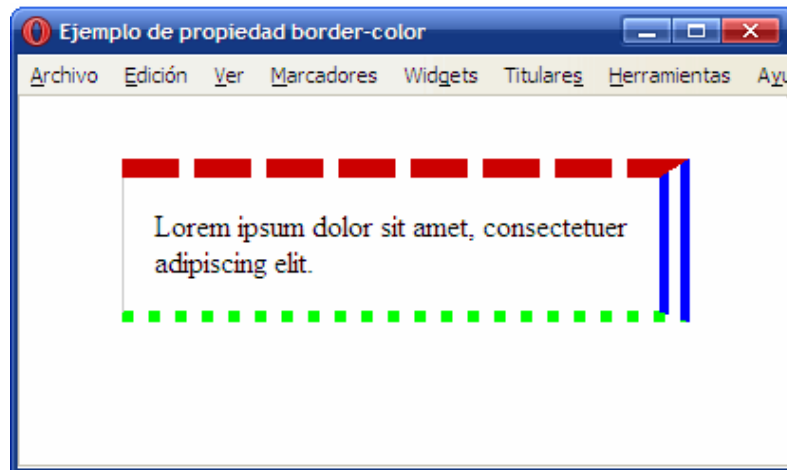
```
div
{
border-top-color: 10px;
border-right-color: 1em;
border-bottom-color: thick;
border-left-color: thin;
}
```

Para cambiar el estilo de cada uno de los bordes, CSS dispone de las siguientes propiedades:

border-top-style	Establece el estilo del borde superior.
border-right-style	Establece el estilo del borde inferior.
border-bottom-style	Establece el estilo del borde izquierdo.
border-left-style	Establece el estilo del borde derecho.
Shorthand	border-style: <[borde_sup][borde_inf][borde_der] [borde_der]>
Se aplica a:	Todas las etiquetas
Valores admitidos:	none hidden dotted dashed solid double groove ridge inset outset inherit

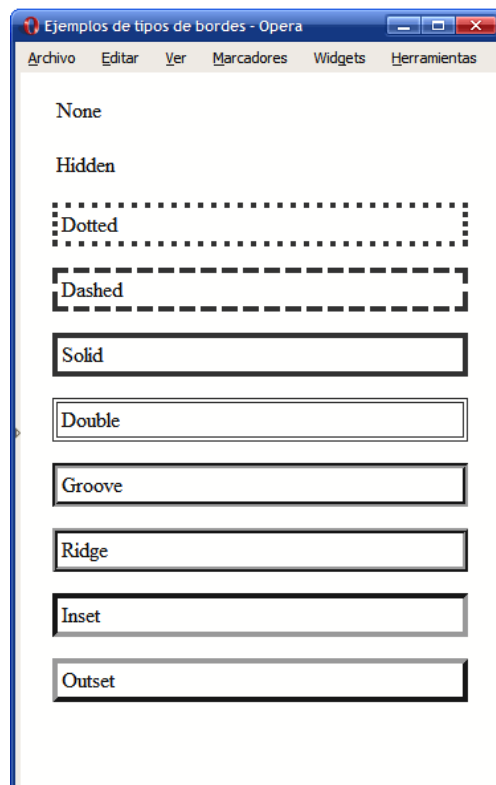
3. El modelo de cajas

El valor del estilo de los bordes sólo se puede indicar mediante alguna de las palabras reservadas definidas por CSS. Como el valor por defecto es *none*, los elementos no muestran ningún borde visible a menos que se establezca explícitamente un estilo de borde. En el siguiente ejemplo se muestran distintos estilos de borde:



```
div
{
border-top-style: dashed;
border-right-style: double;
border-bottom-style: dotted;
border-left-style: solid;
}
```

Y estas son las visualizaciones que muestran los navegadores con cada uno de los estilos:



Los estilos *none* y *hidden* son idénticos visualmente, pero se diferencian en la forma que los navegadores resuelven los conflictos entre los bordes de las celdas adyacentes en las tablas.

Para finalizar, CSS define un shorthand global que permite establecer el ancho, el color y el estilo del borde en una sola propiedad:

border	Establece el estilo completo de los bordes
Se aplica a:	Todas las etiquetas
Valores admitidos:	<ancho> <color> <estilo>

3.6. Color e imagen de fondo

Para establecer el color y la imagen de fondo contamos con las siguientes propiedades:

background-color	Establece el color del fondo de la capa.
Se aplica a:	Todas las etiquetas
Valores admitidos:	<color> transparent inherit

background-image	Establece la imagen del fondo de la capa.
Se aplica a:	Todas las etiquetas
Valores admitidos:	<url_foto> none inherit

El siguiente ejemplo muestra una imagen como fondo de la página:

```
body
{
  background-image: url("img/fondo.png");
}
```

Las imágenes de fondo se indican a través de su URL, que puede ser absoluta o relativa. Por otro lado, 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.

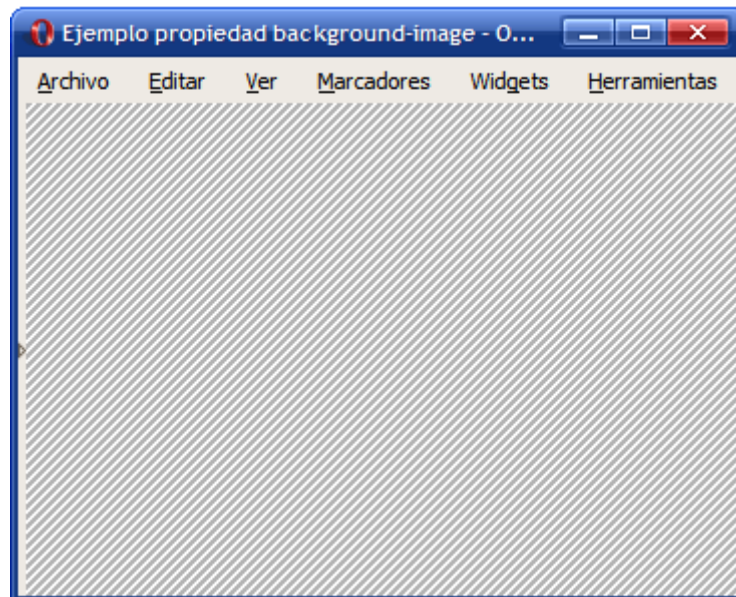
Si la imagen que se quiere mostrar es demasiado grande para el fondo del elemento, solamente se muestra la parte de imagen comprendida en el tamaño del elemento. Si la imagen es más pequeña que el elemento, ésta se repite horizontal y verticalmente hasta llenar el fondo del elemento. Este comportamiento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

3. El modelo de cajas

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

El resultado es el siguiente:



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

Sin embargo, 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:

background-repeat	Establece el modo de repetición de la imagen del fondo de la capa.
Se aplica a:	Todas las etiquetas
Valores admitidos:	repeat repeat-x repeat-y no-repeat inherit

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.

El siguiente sitio web utiliza el valor repeat-x para mostrar una imagen de fondo en la cabecera de la página:

3. El modelo de cajas



```
div#header
{
    background-image: url("img/fondo.gif");
    background-repeat: repeat-x;
    width: 100%;
}
```

Además de seleccionar el tipo de repetición de las imágenes de fondo, CSS permite controlar la posición de la imagen dentro del fondo del elemento mediante la propiedad **background-position**:

background-position	Establece la posición de la imagen del fondo de la capa.
Se aplica a:	Todas las etiquetas
Valores admitidos:	<ul style="list-style-type: none">< porcentaje medida left center right > < porcentaje medida top center bottom >< left center right > < top center bottom >inherit

La propiedad background-position permite indicar la distancia que se desplaza la imagen de fondo respecto de su posición original situada en la esquina superior izquierda.

Si se indican dos porcentajes o dos medidas, el primero indica el desplazamiento horizontal y el segundo el desplazamiento vertical respecto del origen (situado en la esquina superior izquierda del elemento contenedor). Si solamente se indica un porcentaje o una medida, se considera que dicha medida está referida al desplazamiento horizontal. En tal situación, al desplazamiento vertical se le asigna automáticamente el valor de 50%.

Cuando se utilizan porcentajes, su interpretación no es intuitiva: si el valor de la propiedad background-position se indica mediante dos porcentajes x% y%, el navegador coloca el punto (x%, y%) de la imagen de fondo en el punto (x%, y%) del elemento.

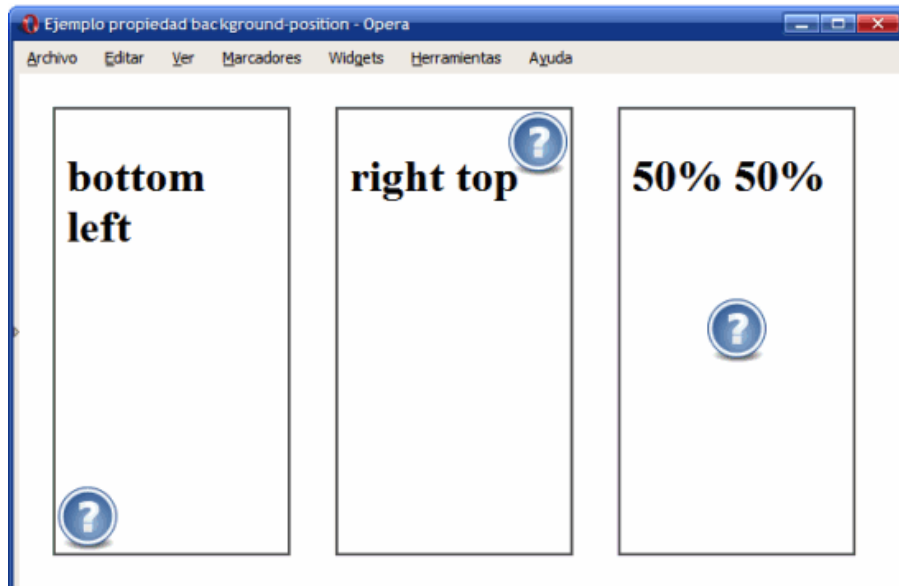
Las palabras clave permitidas son equivalentes a algunos porcentajes significativos:

3. El modelo de cajas

top = 0%, left = 0%, center = 50%, bottom = 100%, right = 100%

CSS permite mezclar porcentajes y palabras clave. Si se utilizan solamente palabras clave, el orden es indiferente y por tanto, es equivalente indicar top left y left top.

El siguiente ejemplo muestra una misma imagen de fondo (un interrogante) posicionada de tres formas diferentes:



Las reglas CSS del ejemplo anterior se muestran a continuación:

```
#caja1
{
  background-image: url("images/help.png");
  background-repeat: no-repeat;
  background-position: bottom left;
}
```

```
#caja2
{
  background-image: url("images/help.png");
  background-repeat: no-repeat;
  background-position: right top;
}
```

```
#caja3
{
  background-image: url("images/help.png");
  background-repeat: no-repeat;
  background-position: 50% 50%;
}
```

Opcionalmente, se puede indicar que el fondo permanezca fijo cuando la ventana del navegador se desplaza mediante las barras de scroll. Se trata de un comportamiento que en general no es deseable y que algunos navegadores no soportan correctamente. La propiedad que controla este comportamiento es **background-attachment**.

background-attachment	Establece el comportamiento de la imagen del fondo de la capa.
Se aplica a:	Todas las etiquetas
Valores admitidos:	< scroll fixed inherit >

Esta propiedad permite controlar si la imagen permanece fija cuando se hace scroll en la ventana del navegador o se desplaza junto con la ventana. Para hacer que una imagen de fondo se muestre fija al desplazar la ventana del navegador, se debe añadir la propiedad **background-attachment: fixed**.

Por último, CSS define una propiedad de tipo "shorthand" para indicar todas las propiedades de los colores e imágenes de fondo de forma directa. La propiedad se denomina **background**:

background	Establece las propiedades del color y la imagen de fondo.
Se aplica a:	Todas las etiquetas
Valores admitidos:	<color> <imagen> <repetición> <comportamiento> <posición> inherit

El siguiente ejemplo muestra la ventaja de utilizar la propiedad background:

```
/*
Color e imagen de fondo de la página mediante una propiedad
shorthand
*/

body
{
    background:  #222d2d url(./graphics/colorstrip.gif)
                repeat-x 0 0;
}

/*
La propiedad shorthand anterior es equivalente a las
siguientes propiedades
*/

body
{
    background-color: #222d2d;
    background-image: url("./graphics/colorstrip.gif");
    background-repeat: repeat-x;
    background-position: 0 0;
}
```

La propiedad background permite asignar todos o sólo algunos de todos los valores que se pueden definir para los fondos de los elementos.

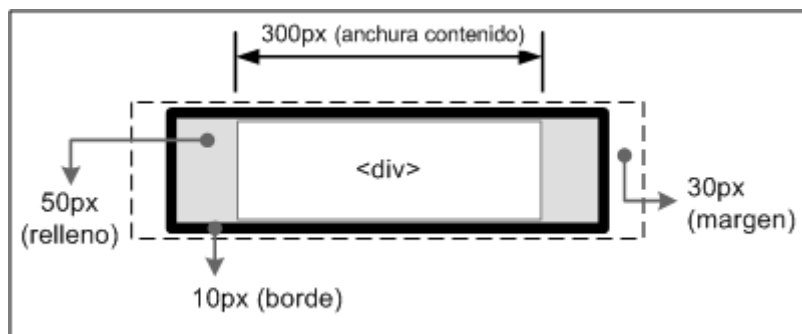
4. El tamaño de la caja visualizada

Cuando se establece el ancho y el alto de un elemento (con las propiedades *width* y *height* que acabamos de ver), estamos estableciendo el ancho y alto de la capa de contenido. Sin embargo, el ancho y alto total con el que el navegador muestra la caja incluye también el margen, el relleno y los bordes.

Por ejemplo, si tenemos definida la siguiente regla:

```
div
{
  width: 300px;
  padding-left: 50px;
  padding-right: 50px;
  margin-left: 30px;
  margin-right: 30px;
  border: 10px solid black;
}
```

La anchura total con la que se muestra el elemento no son los 300 píxeles indicados en la propiedad *width*, sino que se tienen en cuenta todos sus márgenes, rellenos y bordes, tal y como se muestra a continuación:



De esta forma, la anchura del elemento en pantalla sería igual a la suma de la anchura original, los márgenes, los bordes y los rellenos:

$$30px + 10px + 50px + 300px + 50px + 10px + 30px = 480 \text{ píxel}$$

5. El modo 'quirks'

Durante los años 90 los principales desarrolladores de navegadores web (Microsoft y Netscape) añadieron sus propias extensiones y mejoras en sus productos creando su propia versión propietaria de HTML. Posteriormente, aparecieron los estándares publicados por el W3C y los desarrolladores se encontraron con el problema de la incompatibilidad entre las implementaciones propietarias de HTML/CSS de los navegadores y las implementaciones que requerían los estándares.

La solución que adoptaron fue la de incluir en el navegador dos modos diferentes de funcionamiento: un modo compatible con las páginas antiguas (denominado "modo quirks" y que se podría traducir como "modo raro") y un modo compatible con los nuevos estándares (denominado "modo estándar").

El modo quirks es equivalente a la forma en la que se visualizaban las páginas en los navegadores Internet Explorer 4 y Netscape Navigator 4. La diferencia más notable entre los dos modos es el tratamiento del "box model", lo que puede afectar gravemente al diseño de las páginas HTML.

Los navegadores seleccionan automáticamente el modo en el que muestran las páginas en función del DOCTYPE definido por el documento. En general, los siguientes tipos de DOCTYPE activan el modo quirks en los navegadores:

- No utilizar ningún DOCTYPE
- DOCTYPE anterior a HTML 4.0
- En algunos navegadores, también activa el modo quirks los DOCTYPE XHTML 1.0 que incluyen la declaración de XML (`<?xml version="1.0" encoding="UTF-8"?>`) al principio de la página web.

Se pueden consultar todos los casos concretos que activan el modo quirks para cada navegador en la página <http://hsivonen.iki.fi/doctype/>