

Unidad Didáctica 5
Modelo de cajas

Contenido

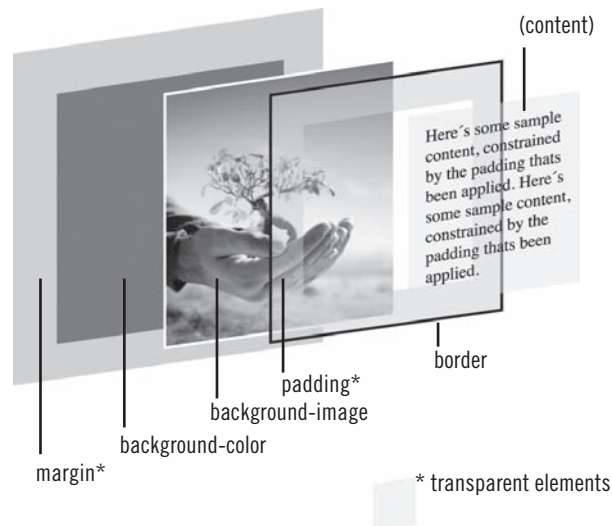
1. Introducción
2. Estructura y propiedades

1. Introducción

El modelo de cajas (*box model*) es, sin ninguna duda, una de las características más importantes de CSS ya que condiciona el diseño de la mayoría de las páginas Web que existen. El *box model* consiste en el comportamiento CSS que hace que todos los elementos incluidos en una página HTML presenten una estructura de caja rectangular. Esto permite y facilita el control del aspecto y posicionamiento de estos elementos.

2. Estructura y propiedades

Cualquier elemento HTML consiste en una caja con las siguientes partes:



- *content* (contenido): es el contenido del elemento. Por ejemplo: palabras de `<P>`, imagen contenida en ``.
- *padding* (relleno): Espacio entre el contenido y un borde que lo “cierre”. Es opcional y transparente.
- *border* (borde): Línea que encierra el contenido y el relleno.
- *background-image* (imagen de fondo): Es la imagen que aparece detrás del contenido y el relleno.

- *background-color* (color fondo): Lo mismo que el anterior pero con un color de fondo.
- *margin* (margen): Es el espacio que existe entre la caja y otros elementos adyacentes. Es transparente.

2.1. Anchura (*width*) y altura (*height*)

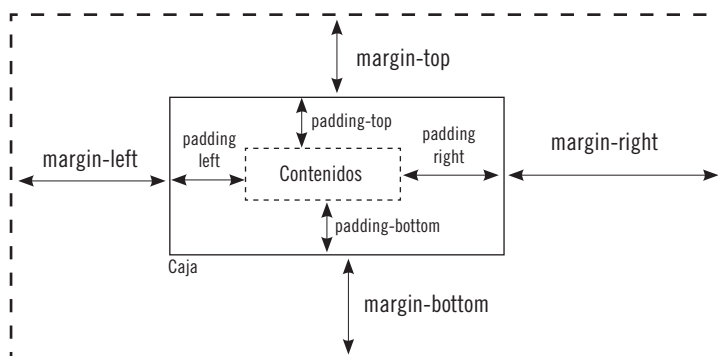
Las propiedades que controlan la anchura y altura de los elementos se denominan, respectivamente, *width* y *height*.

Cada una de estas propiedades puede tener los siguientes valores:

- *inherit*: La anchura/altura del elemento se hereda del elemento padre.
- *auto* (por defecto): El navegador calcula la anchura/altura del elemento en función del contenido y posicionamiento del elemento.
- Porcentaje y medida: La medida puede ser absoluta o relativa. El porcentaje se establece en función de la anchura/altura del elemento padre.

2.2. Margen (*margin*) y relleno (*padding*)

CSS establece una serie de propiedades para controlar el margen y el relleno de los elementos.



2.2.1 Margen

Existen propiedades que controlan el margen de una caja de texto: *margin-top* (margen superior), *margin-bottom* (margen inferior), *margin-left* (margen izquierdo) y *margin-right* (margen derecho).

Estas cuatro propiedades pueden ser aplicadas a todos los elementos, excepto *margin-top* y *margin-bottom* que se usan en elementos bloque e imágenes.

Respecto a las unidades, se suelen utilizar los *píxeles* y los *em* (aunque también se admiten porcentajes).

En el siguiente ejemplo se establecen márgenes a una imagen:

```
.margenes {
    margin-top: 100px;
    margin-bottom: 100px;
    margin-right: 100px;
    margin-left: 100px;
}

...
<IMG src="imagenes/img.jpeg" /><BR />
<IMG class="margenes" src="imagenes/
    img.jpeg" /><BR />
<IMG src="imagenes/img.jpeg" />
...
```

El estilo se aplica a la segunda imagen. Como puede ver es ésta la que se “separa” de las otras dos:

Definición



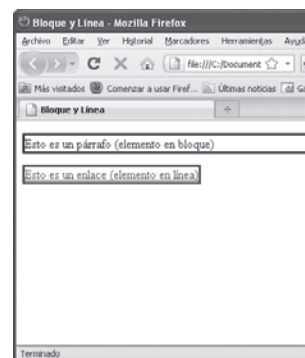
Los elementos en HTML se dividen en dos grandes grupos: elementos en bloque y elementos en línea.

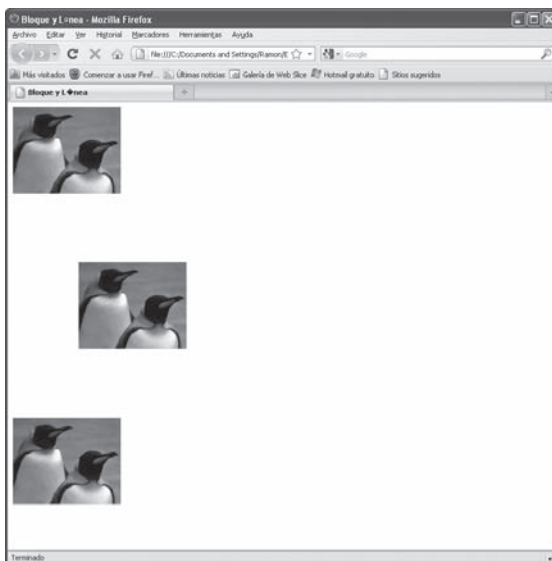
Cuando son mostrados en el navegador, los elementos en bloque aparecen en una nueva línea y ocupan todo el espacio de esta. Por otro lado, los elementos en línea aparecen a continuación del último elemento mostrado y sólo ocupan el espacio necesario para su visualización.

Por sus características, los elementos en bloque no pueden insertarse dentro de uno de línea mientras que éstos pueden ir dentro, tanto dentro de uno de línea como dentro de uno de bloque.

Ejemplos de elementos en línea: `img`, `a`, `input`, `select`, `textarea` etc.

Ejemplos de elementos en bloque: `FORM`, `H1` ~ `H6`, `HR`, `P`, `TABLE`, `UL` etc.





Aplicación práctica

Aplice el estilo CSS anterior a cada uno de los seis enlaces dispuestos con el siguiente código HTML. ¿Qué diferencia encuentra con el ejemplo anterior?:

```
...  
<A href="#">Enlace 1</A><A href="#">Enlace 2</A><BR />  
<A href="#">Enlace 3</A><A href="#">Enlace 4</A><BR />  
<A href="#">Enlace 5</A><A href="#">Enlace 5</A>  
...
```

SOLUCIÓN

El estilo se puede aplicar de la siguiente manera:

```
...  
<A class="margenes" href="#">Enlace 1</A><A class="margenes" href="#">Enlace 2</  
A><BR/>  
<A class="margenes" href="#">Enlace 3</A><A class="margenes" href="#">Enlace 4</  
A><BR/>
```

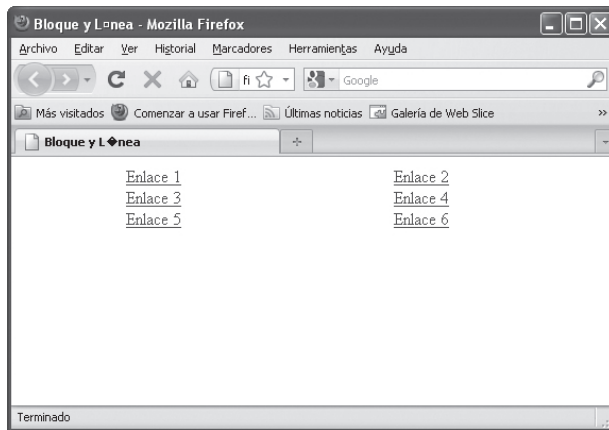
Continúa en página siguiente >>

<< Viene de página anterior

```
<A class="margenes" href="#">Enlace 5</A><A class="margenes" href="#">Enlace 6</A>
```

...

El resultado obtenido es:



En este caso, cada elemento es un enlace en lugar de una imagen. Como puede observar, los márgenes superior e inferior no funcionan (superior e inferiormente los enlaces continúan “pegados”). Esto se debe a que los enlaces no son elementos tipo bloque o imágenes como se comentó antes.

Existe una propiedad llamada *margin* que permite establecer, a la vez, los cuatro márgenes posibles. Esta es una propiedad que se denomina del tipo *shorthand*. Se pueden establecer de 1 a 4 márgenes, pero se deben de tener en cuenta las siguientes consideraciones:

- Si se declara un solo margen, este es el valor de los cuatro.
- Si se declaran dos, el primero es el valor del superior e inferior y el segundo es el de los laterales.

- Si se declaran tres, el primero es el valor del superior, el segundo es el del inferior y el tercero es el de los laterales.
- En el caso de indicar los cuatro márgenes, el formato debe de ser:
margin: valor_superior valor_derecho valor_inferior, valor_izquierdo

Por ejemplo, para el código CSS anterior, hubiera sido equivalente a poner (todos los márgenes son de 100 px):

```
.margenes { margin: 100px;  
            }
```



Consejo

Uno de los métodos más utilizados para centrar una página horizontalmente consiste en establecer los márgenes laterales (*margin-left*, *margin-right*) con valor *auto*. Esto se debe a que, cuando se asignan márgenes laterales automáticos a un elemento, los navegadores centran ese elemento respecto de su elemento padre. Es importante tener en cuenta que esto no funciona así con la alineación vertical.

2.2.2 Relleno

Para definir el relleno, CSS establece otras cuatro propiedades: *padding-top* (relleno superior), *padding-right* (relleno derecho), *padding-bottom* (relleno inferior) y *padding-left* (relleno izquierdo).

Estas propiedades establecen la separación entre el contenido y el borde de la caja y usan las mismas unidades que los márgenes.

Existe una propiedad llamada *padding* que permite establecer los cuatro rellenos a la vez. Funciona exactamente igual que *margin* ya que también es del tipo *shorthand*.

En el siguiente ejemplo se muestran tres párrafos que presentan diferentes comportamientos: el primero no tiene ni margen ni relleno, el segundo tiene margen y el tercero tiene relleno. Para una mejor visualización del ejemplo, se ha definido una propiedad que se verá más adelante: *border-style*.

```
.nada {
    border-style: solid;
}
.margenes {
    margin: 50px;
    border-style: solid;
}
.relleno {
    padding: 50px;
    border-style: solid;
}
```

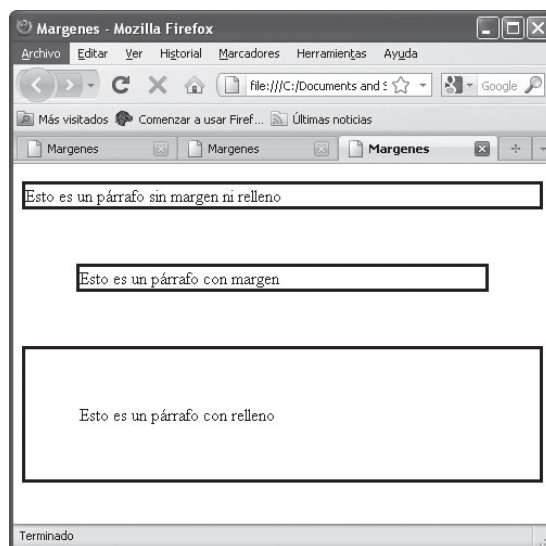
...

`<P class="nada">Esto es un párrafo sin margen ni relleno</P>`

`<P class="margenes">Esto es un párrafo con margen</P>`

`<P class="relleno">Esto es un párrafo con relleno</P>`

...



2.3. Bordes

CSS puede definir el aspecto de los cuatro bordes que forman parte de los elementos HTML. Se pueden establecer la anchura, el color y el estilo.

2.3.1 Anchura

La anchura de los bordes se controla con las propiedades: *border-top-width* (anchura del borde superior), *border-right-width* (anchura del borde derecho), *border-bottom-width* (anchura del borde inferior) y *border-left-width* (anchura del borde izquierdo).

La anchura de los bordes se puede indicar con las medidas absolutas o relativas correspondientes. También se pueden usar las palabras clave: *thin* (borde delgado), *medium* (borde normal, es el que está por defecto) y *thick* (borde ancho).

Existe también una propiedad que permite establecer, de una vez, la anchura de los bordes de la tabla. Esta es *border-width* y es del tipo *shorthand*.

2.3.2 Color

Las cuatro propiedades que controla el color de los bordes son: *border-top-color* (color borde superior), *border-right-color* (color borde derecho), *border-bottom-color* (color borde inferior) y *border-left-color* (color borde izquierdo).

Los colores se suelen identificar en formato hexadecimal o utilizando la definición correspondiente al color (*green*, *red* etc.). Para definir todos los colores en una propiedad se utiliza *border-color* (*shorthand*).

2.3.3 Estilo

Las siguientes propiedades establecen el estilo de los bordes de los elementos: *border-top-style* (estilo borde superior), *border-right-style* (estilo borde derecho), *border-bottom-style* (estilo borde inferior) y *border-left-style* (estilo borde izquierdo).

Los valores que pueden presentar son:

- *none*: No se muestra ningún borde. Es el valor por defecto.
- *hidden*: Visualmente es idéntico a *none*, ya que hace que no se muestre borde alguno. La única diferencia con el anterior es la manera en la que el navegador solventa los conflictos entre bordes adyacentes en las celdas de las tablas.
- *dotted*: El borde consiste en puntos separados por espacios.
- *dashed*: El borde consiste en trazos separados por espacios.
- *solid*: Borde en línea recta y continua.
- *double*: Borde formado por dos líneas rectas continuas y paralelas.
- *groove*: Borde “hundido”.
- *ridge*: Borde “saliente”.
- *inset*: Borde que da la sensación de que el elemento que encierra está hundido.
- *outset*: Borde que da la sensación de que el elemento que encierra “sobresale” de la pantalla.

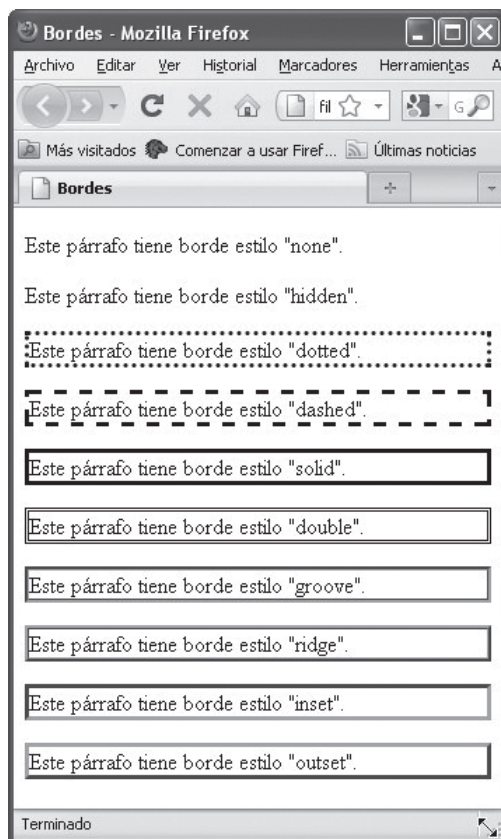
Con la propiedad *border-style*, se puede definir el aspecto de todos los bordes de una sola vez ya que se tratan de otra propiedad *shorthand*.

A continuación se va a ver un ejemplo de visualización de cada uno de estos bordes:

```
.none { border-style: none; }
.hidden { border-style: hidden; }
.dotted { border-style: dotted; }
.dashed { border-style: dashed; }
.solid { border-style: solid; }
.double { border-style: double; }
.groove { border-style: groove; }
.ridge { border-style: ridge; }
.inset { border-style: inset; }
.outset { border-style: outset; }
...
```

```
<P class="none">Este párrafo tiene borde estilo "none".</P>  
<P class="hidden">Este párrafo tiene borde estilo "hidden".</P>  
<P class="dotted">Este párrafo tiene borde estilo "dotted".</P>  
<P class="dashed">Este párrafo tiene borde estilo "dashed".</P>  
<P class="solid">Este párrafo tiene borde estilo "solid".</P>  
<P class="double">Este párrafo tiene borde estilo "double".</P>  
<P class="groove">Este párrafo tiene borde estilo "groove".</P>  
<P class="ridge">Este párrafo tiene borde estilo "ridge".</P>  
<P class="inset">Este párrafo tiene borde estilo "inset".</P>  
<P class="outset">Este párrafo tiene borde estilo "outset".</P>  
...
```

Este código dará como resultado:



2.3.4 Estilos por borde

Existen cuatro propiedades que permiten definir de forma completa cada uno de los cuatro bordes. Estas son: `border-top` (estilo completo borde superior), `border-right` (estilo completo borde derecho), `border-bottom` (estilo completo borde inferior) y `border-left` (estilo completo borde izquierdo). Estas propiedades son del tipo *shorthand*.

La sintaxis de los posibles valores que pueden contener estas propiedades es:

(medida_borde, color_borde, estilo_borde)

Por ejemplo, si deseamos definir el estilo del borde inferior como: anchura: thin; color: gris; estilo: solid:

```
.estilobordeinf {
    border-bottom: thin grey solid;
}
```

En el caso de que se desee que todos los bordes tengan el mismo aspecto, se puede utilizar la propiedad `border`. Por ejemplo:

```
.estilo {
    border: 1px solid red;
}
```

Este estilo establece que los cuatro bordes de la caja medirán 1 px de ancho, serán de color rojo y tendrán el estilo *solid*.

Importante



Tenga en cuenta que la anchura y altura final de un elemento no se determina únicamente con las propiedades `width` y `height`. Estas propiedades determinan el ancho y el alto del contenido sin contar con el relleno, borde y margen. Por ejemplo, fíjese en el siguiente estilo de párrafo:

```
P {
    width: 150px;
    padding-left: 25px;
    padding-right: 25px;
    margin-left: 25px;
    margin-right: 15px;
    border: 15px solid black;
}
```

El siguiente croquis explica cómo calcular la anchura total del un párrafo con el estilo anterior:

