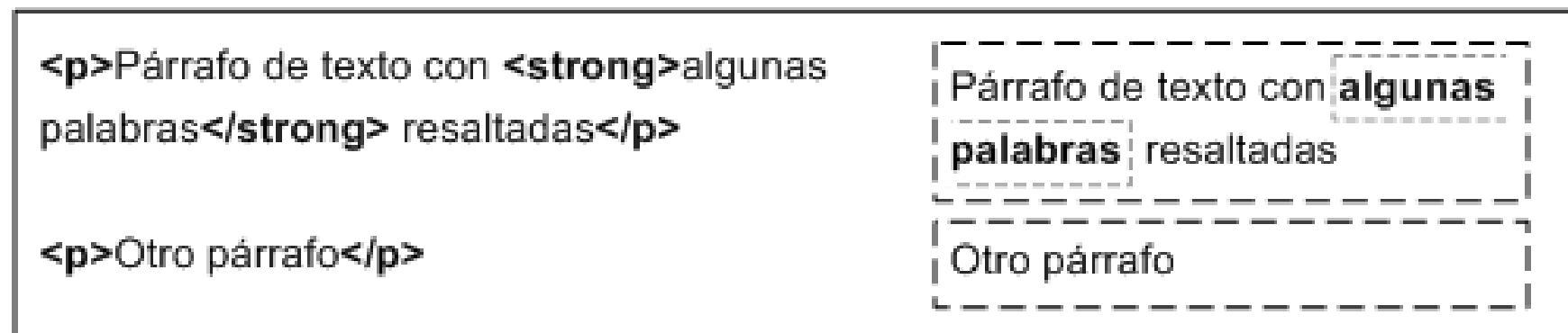


Modelo de Cajas

El modelo de cajas o "*box model*" es seguramente la característica más importante del lenguaje de hojas de estilos CSS, ya que condiciona el diseño de todas las páginas web. El modelo de cajas es el comportamiento de CSS que hace que todos los elementos de las páginas se representen mediante cajas rectangulares.

Las cajas de una página se crean automáticamente. Cada vez que se inserta una etiqueta HTML, se crea una nueva caja rectangular que encierra los contenidos de ese elemento. La siguiente imagen muestra las tres cajas rectangulares que crean las tres etiquetas HTML que incluye la página:



Hojas de estilo CSS

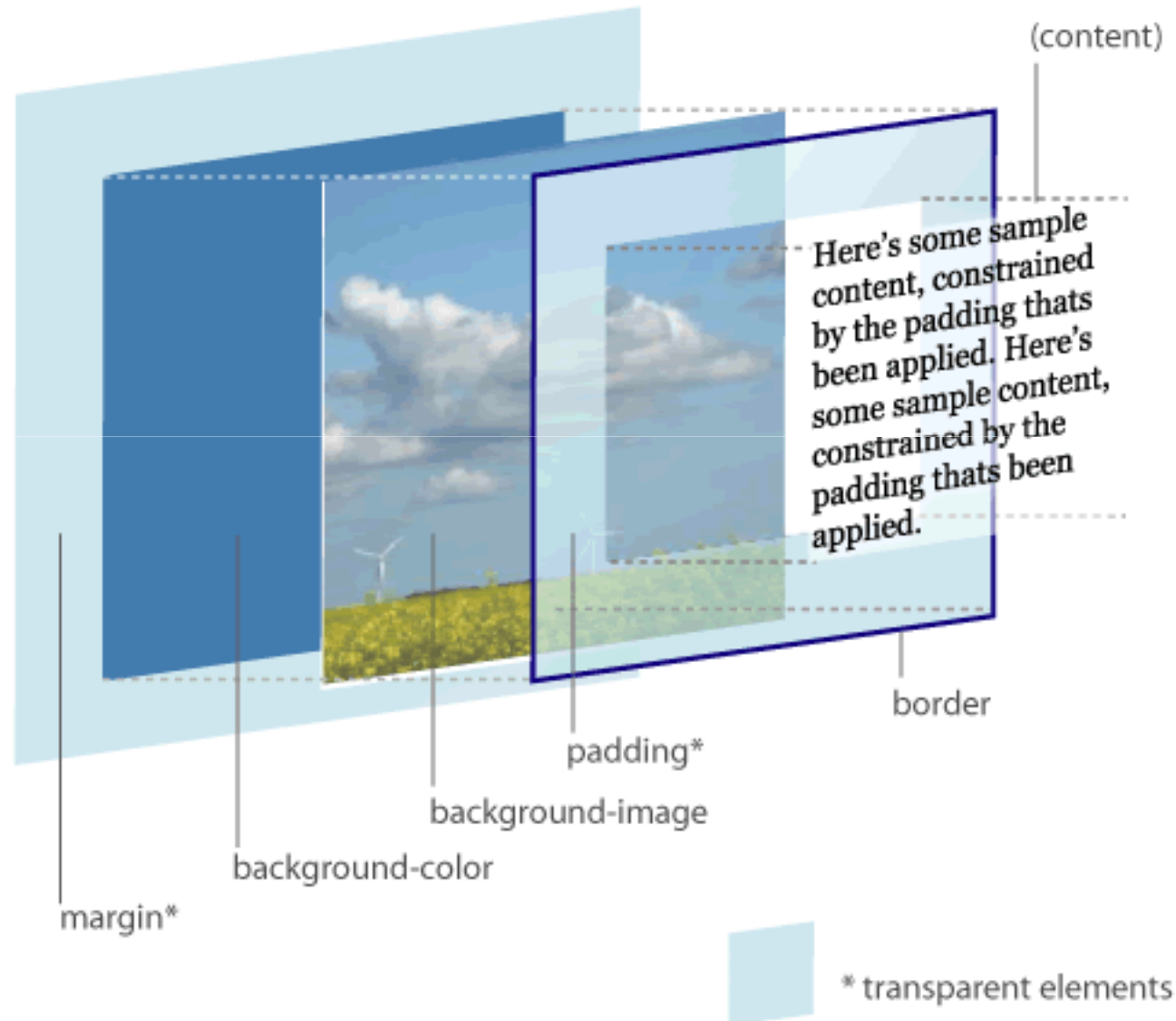
Las cajas se crean automáticamente al definir cada elemento HTML.

Las cajas de las páginas no son visibles a simple vista porque inicialmente no muestran ningún color de fondo ni ningún borde.

Los navegadores crean y colocan las cajas de forma automática, pero CSS permite modificar todas sus características. Cada una de las cajas está formada por seis partes, tal y como muestra la siguiente imagen:

Hojas de estilo CSS

Modelo de Cajas



Hojas de estilo CSS

Modelo de Cajas

Las partes que componen cada caja y su orden de visualización desde el punto de vista del usuario son las siguientes:

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 existente entre el contenido y el borde.

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**): separación opcional existente entre la caja y el resto de cajas adyacentes.

1. Anchura y altura

1.1. Anchura

La propiedad CSS que controla la anchura de la caja de los elementos se denomina **width**. Se mide en px o %.

El siguiente ejemplo establece el valor de la anchura del elemento <div> lateral:

```
#lateral { width: 200px; }
```

```
<div id="lateral">
```

```
  ...  
</div>
```

1.2. Altura

La propiedad CSS que controla la altura de los elementos se denomina **height**. Se mide en px o %.

El siguiente ejemplo establece el valor de la altura del elemento <div> de cabecera:

```
#cabecera { height: 60px; }
```

```
<div id="cabecera">
```

```
  ...  
</div>
```

También se pueden especificar una anchura y una altura mínima y/o máxima mediante las propiedades:

min-width

min-height

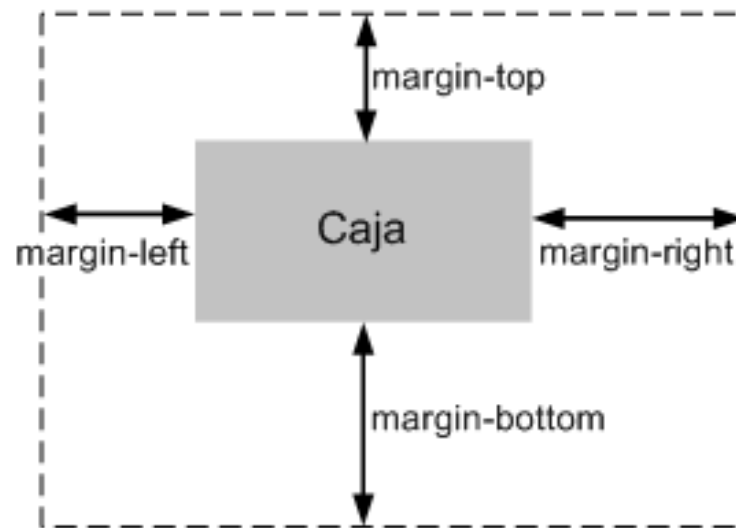
max-width

max-height

2. Margen y relleno

2.1. Margen

CSS define cuatro propiedades para controlar cada uno de los márgenes horizontales y verticales de un elemento. Cada una de las propiedades establece la separación entre el borde lateral de la caja y el resto de cajas adyacentes:



La propiedad margin establece el mismo margen en los 4 sentidos.

2. Margen y relleno

2.1. Margen

El siguiente ejemplo añade un margen izquierdo al segundo párrafo:

```
.destacado {  
    margin-left: 30px;  
}
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam et elit. Vivamus placerat lorem. Maecenas sapien. Integer ut massa. Cras diam ipsum, laoreet non, tincidunt a, viverra sed, tortor.

Vestibulum lectus diam, luctus vel, venenatis ultrices, cursus vel, tellus. Etiam placerat erat non sem. Nulla molestie odio non nisl tincidunt faucibus.

Aliquam euismod sapien eu libero. Ut tempor orci at nulla. Nam in eros egestas massa vehicula nonummy. Morbi posuere, nibh ultricies consectetur tincidunt, risus turpis laoreet elit, ut tincidunt risus sem et nunc.

¿Cuál es la diferencia?

margin: 5px;

margin: 5px;
margin-left: 30px;

margin: 5px, 5px, 5px, 30px;

margin: 5px, 10px;

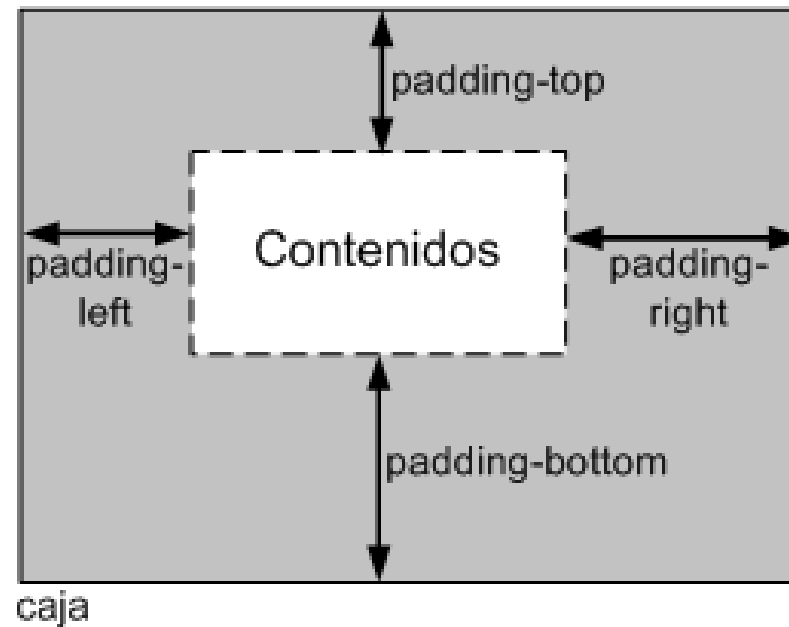
Al margen también se le puede asignar el valor auto, hace que se centre horizontalmente respecto a su contenedor.

```
div {  
    margin: auto;  
    width: 200px;  
}
```

2. Margen y relleno

2.2. Relleno

CSS define cuatro propiedades para controlar cada uno de los espacios de relleno horizontales y verticales de un elemento. Cada una de estas propiedades establece la separación entre el contenido y los bordes laterales de la caja del elemento.



La propiedad padding establece el mismo relleno en los 4 sentidos.

3. Bordes

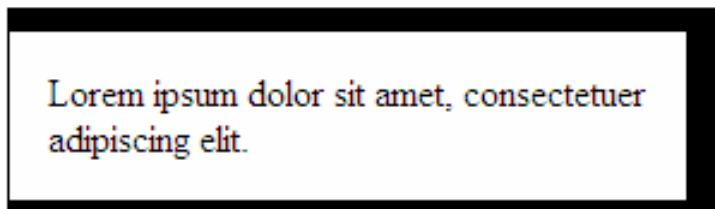
CSS permite modificar el aspecto de cada uno de los cuatro bordes de la caja de un elemento. Para cada borde se puede establecer su anchura o grosor, su color y su estilo.

3.1. Ancho

Propiedades: **border-width** (ancho de borde)
border-top-width (ancho de borde superior)
border-right-width (ancho de borde derecho)
border-bottom-width (ancho de borde inferior)
border-left-width (ancho de borde izquierdo)

Valor: <numero px|em> | thin | medium | thick

Ejemplo:



Las reglas CSS utilizadas se muestran a continuación:

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

3. Bordes

3.2. Color

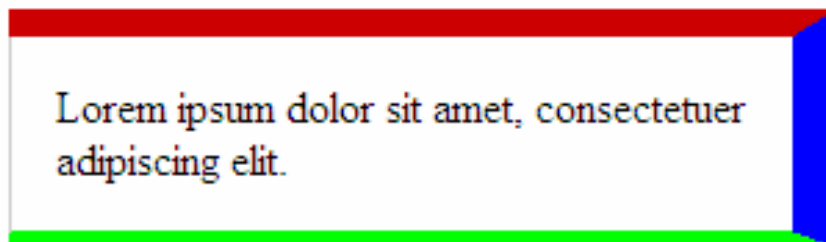
Seleccionador de colores

Nombres de colores

Propiedades: **border-color** (color de borde)
border-top-color (color de borde superior)
border-right-color (color de borde derecho)
border-bottom-color (color de borde inferior)
border-left-color (color de borde izquierdo)

Valores: <color> | transparent

Ejemplo:



```
div {  
    border-top-color: #CC0000;  
    border-right-color: blue;  
    border-bottom-color: #00FF00;  
    border-left-color: #CCC;  
}
```

https://www.w3schools.com/cssref/playit.asp?filename=playcss_border-color

3. Bordes

3.3. Estilo

Propiedades:

border-style

border-top-style

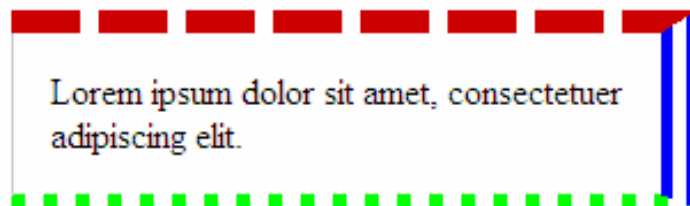
border-right-style

border-bottom-style

border-left-style

Valores: none | hidden | dotted | dashed | solid | double | groove | ridge | inset | outset

Ejemplo:



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

https://www.w3schools.com/cssref/playit.asp?filename=playcss_border-style

3. Bordes

3.4. Radio

Propiedades: **border-radius**
border-top-left-radius
border-top-right-radius
border-bottom-right-radius
border-bottom-left-radius

Valores: <numero px|em> | % | initial

https://www.w3schools.com/cssref/playit.asp?filename=playcss_border-radius

Las propiedades de borde width, style y color (radius no) también se pueden especificar de forma compacta de la siguiente forma:

border: 5px solid blue;

¿Cuál es la diferencia?

border-radius: 15px;

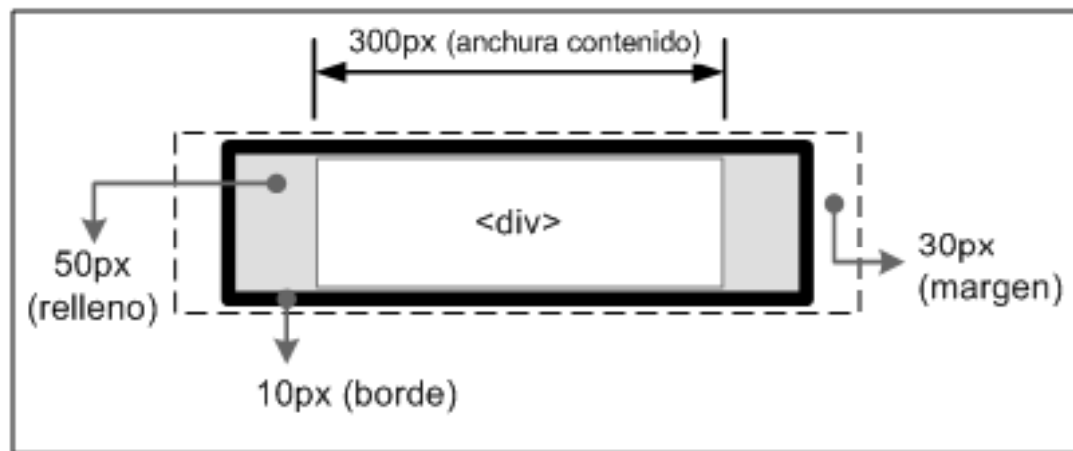
border-radius: 15px 50px 30px 5px;

border-radius: 15px 50px 30px;

border-radius: 15px 50px;

4. Margen, relleno, bordes y modelo de cajas

La anchura y altura de un elemento no solamente se calculan teniendo en cuenta sus propiedades `width` y `height`. El margen, el relleno y los bordes establecidos a un elemento determinan la anchura y altura final del elemento. En el siguiente ejemplo se muestran los estilos CSS de un elemento:



```
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 píxel indicados en la propiedad `width`, sino que también se añaden todos sus márgenes, rellenos y bordes:

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:

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

Así, la anchura/altura establecida con CSS siempre hace referencia a la anchura/altura del contenido. La anchura/altura total del elemento debe tener en cuenta además los valores del resto de partes que componen la caja del *box model*.

5. Fondos

El último elemento que forma el *box model* es el fondo de la caja del elemento.

El fondo puede ser un **color** simple o una **imagen**.

El fondo solamente se visualiza en el área ocupada por el contenido y su relleno, ya que el color de los bordes se controla directamente desde los bordes y las zonas de los márgenes siempre son transparentes.

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

CSS define cinco propiedades para establecer el fondo de cada elemento.

5. Fondos

Propiedad **background-color**

Valores	color transparent
Valor inicial	transparent
Descripción	Establece un color de fondo para los elementos

```
body {  
  background-color: #F5F5F5;  
}
```

Propiedad **background-image**

Valores	url none
Valor inicial	none
Descripción	Establece una imagen como fondo para los elementos

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

5. Fondos

Propiedad **background-repeat**

Valores repeat | repeat-x | repeat-y | no-repeat

Valor inicial repeat

Descripción Controla la forma en la que se repiten las imágenes de fondo

https://www.w3schools.com/cssref/playit.asp?filename=playcss_background-repeat

Propiedad **background-position**

Valores (x | <num%> | left | center | right) (y | <num%> | top | center | bottom)

Valor inicial 0% 0%

Descripción Controla la posición en la que se muestra la imagen en el fondo del

https://www.w3schools.com/cssref/playit.asp?filename=playcss_background-position

Propiedad **background-attachment**

Valores scroll | fixed

Valor inicial scroll

Descripción Controla la forma en la que se visualiza la imagen de fondo: fija o se desplaza al hacer scroll

6. Otros

Propiedad **display**

Valores	inline block flex none ...
Valor inicial	Depende de la etiqueta
Descripción	Permite cambiar en que forma se visualiza un elemento HTML. <ul style="list-style-type: none">- inline: en linea (como <a>)- block: en bloque (como <p>)- flex: contenedor flex (para flexbox)- none: el elemento no se visualiza y no ocupa espacio

Propiedad **visibility**

Valores	visible hidden
Valor inicial	visible
Descripción	Especifica si un elemento es visible o no. El elemento sigue ocupando su espacio aunque no sea visible.