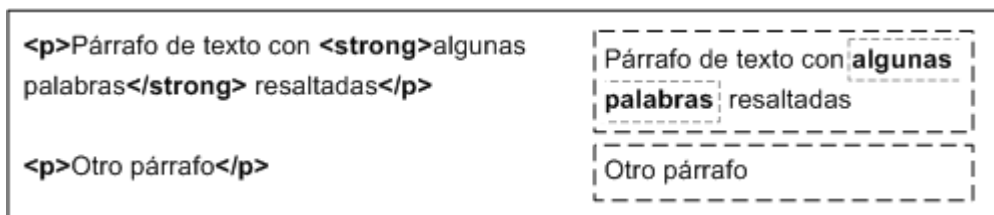


## Capítulo 4. Modelo de cajas

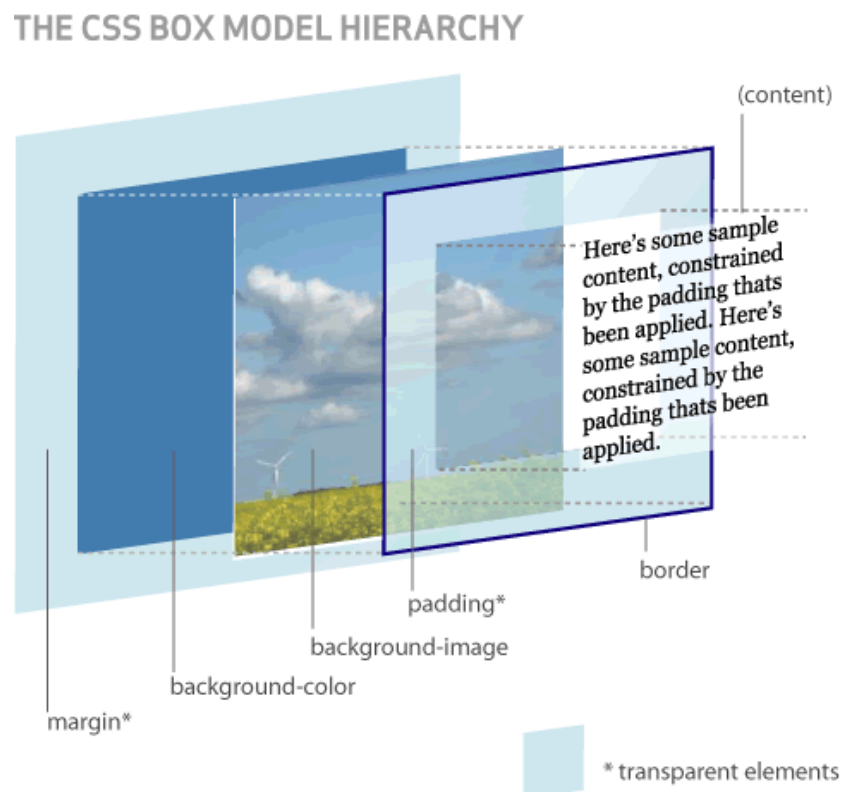
El **modelo de cajas o "box model"** es seguramente la característica más importante del lenguaje de hojas de estilos CSS, ya que condiciona el diseño de todas las páginas web. El modelo de cajas es el *comportamiento de CSS que hace que **todos los elementos de las páginas se representen mediante cajas 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 3 cajas rectangulares que crean las 3 etiquetas HTML:



**Figura 4.1.** 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:



**Figura 4.2.** Representación tridimensional del box model de CSS

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.

*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. Combinando imágenes transparentes y colores de fondo se pueden lograr efectos gráficos muy interesantes.*

## 4.1. Anchura y altura

### 4.1.1. Anchura

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

<b>Width</b>	<b>Anchura</b>
<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>	<i>Establece la anchura de un elemento</i>

La **propiedad** `width` no admite valores negativos y los valores en porcentaje se calculan a partir de la anchura de su elemento padre. El **valor** `inherit` indica que la anchura del elemento se hereda de su elemento padre. El **valor** `auto`, que es el que se utiliza si no se establece de forma explícita un valor a esta propiedad, indica que el navegador debe calcular automáticamente la anchura del elemento, teniendo en cuenta sus contenidos y el sitio disponible en la página.

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

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

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

CSS define otras dos propiedades relacionadas con la anchura de los elementos: `min-width` y `max-width`, que se verán más adelante.

### 4.1.2. Altura

La propiedad CSS que controla la altura de los elementos se denomina `height`.

<b>Height</b>	Altura
<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 columnas de tabla y los grupos de columnas de tabla
<b>Valor inicial</b>	Auto
<b>Descripción</b>	<i>Establece la altura de un elemento</i>

Al igual que sucede con `width`, la propiedad `height` no admite valores negativos. Si se indica un porcentaje, se toma como referencia la altura del elemento padre. Si el elemento padre no tiene una altura definida explícitamente, se asigna el **valor** `auto` a la altura.

El **valor** `inherit` indica que la altura del elemento se hereda de su elemento padre. El valor `auto`, que es el que se utiliza si no se establece de forma explícita un valor a esta propiedad, indica que el navegador debe calcular automáticamente la altura del elemento, teniendo en cuenta sus contenidos y el sitio disponible en la página.

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

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

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

```
...
```

```
</div>
```

CSS define otras dos propiedades relacionadas con la altura de los elementos: **min-height** y **max-height**, que se verán más adelante.

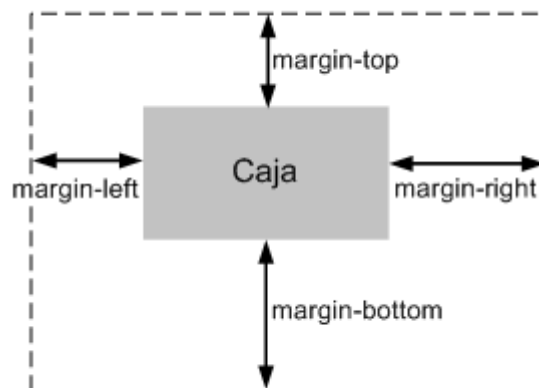
## 4.2. Margen y relleno

### 4.2.1. Margen

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

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

Cada una de las propiedades establece la *separación entre el borde lateral de la caja y el resto de cajas adyacentes*:



**Figura 4.3.** Las cuatro propiedades relacionadas con los márgenes

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***:

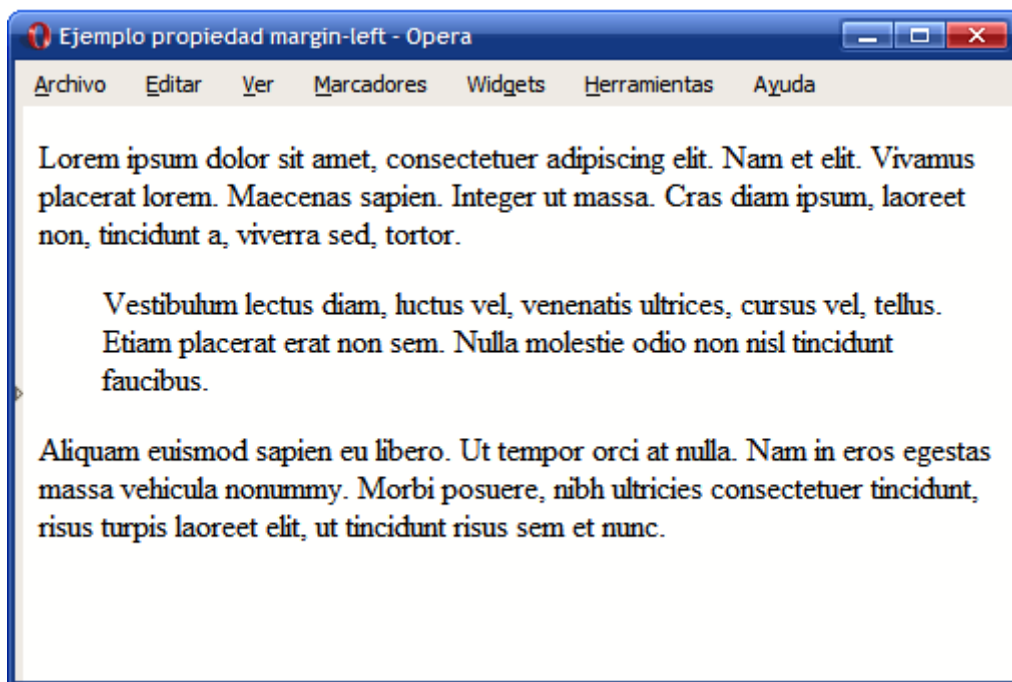
```
.destacado {  
  margin-left: 2em;  
}
```

**<p>**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.**</p>**

**<p class="destacado">**Vestibulum lectus diam, luctus vel, venenatis ultrices, cursus vel, tellus. Etiam placerat erat non sem. Nulla molestie odio non nisl tincidunt faucibus.**</p>**

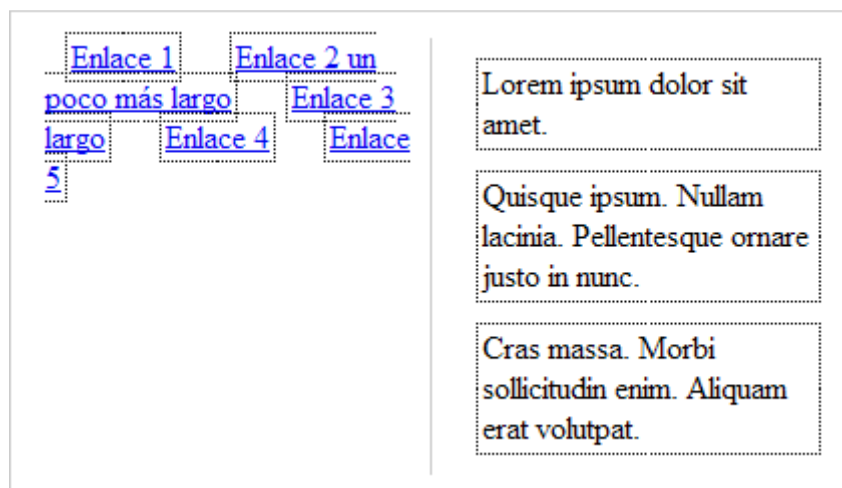
**<p>**Aliquam euismod sapien eu libero. Ut tempor orci at nulla. Nam in eros egestas massa vehicula nonummy. Morbi posuere, nibh ultricies consectetur tincidunt, risus turpis laoreet elit, ut tincidunt risus sem et nunc.**</p>**

A continuación se muestra el aspecto del ejemplo anterior en cualquier navegador:



**Figura 4.4.** Ejemplo de propiedad margin-left

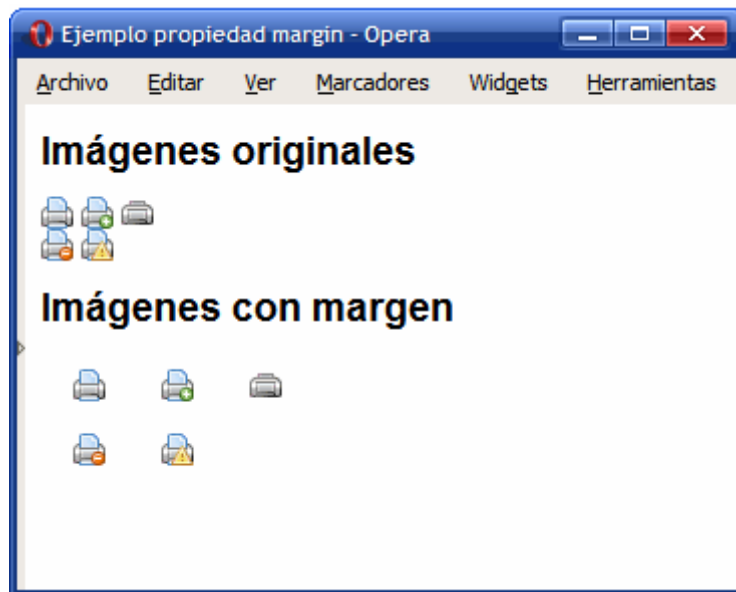
Los **márgenes verticales** (*margin-top* y *margin-bottom*) sólo se pueden aplicar a los elementos de bloque y las imágenes, mientras que los **márgenes laterales** (*margin-left* y *margin-right*) se pueden aplicar a cualquier elemento, tal y como muestra la siguiente imagen:



**Figura 4.5.** Los márgenes verticales sólo se aplican a los elementos de bloque e imágenes

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.

El siguiente **ejemplo** muestra imágenes con márgenes para facilitar su identificación y mejorar el diseño general de la página:



**Figura 4.6.** Ejemplo de propiedad margin

El código CSS del ejemplo anterior se muestra a continuación:

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

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. Este tipo de propiedades resumidas se denominan **propiedades de tipo "shorthand"** y CSS define varias propiedades de este tipo.

La propiedad que permite definir de forma simultánea los cuatro márgenes se denomina **margin**.

<b>Margin</b>	<i>Margin</i>
<b>Valores</b>	( <medida>   <porcentaje>   auto ) {1, 4}   inherit
<b>Se aplica a</b>	Todos los elementos salvo algunos casos especiales de elementos mostrados como tablas
<b>Valor inicial</b>	-
<b>Descripción</b>	<i>Establece de forma directa todos los márgenes de un elemento</i>

La notación **{1, 4}** de la definición anterior significa que la **propiedad margin** admite entre uno y cuatro valores, con el siguiente significado:

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

El ejemplo anterior de márgenes se puede reescribir utilizando la propiedad **margin**:

**Código CSS original:**

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

**Alternativa directa:**

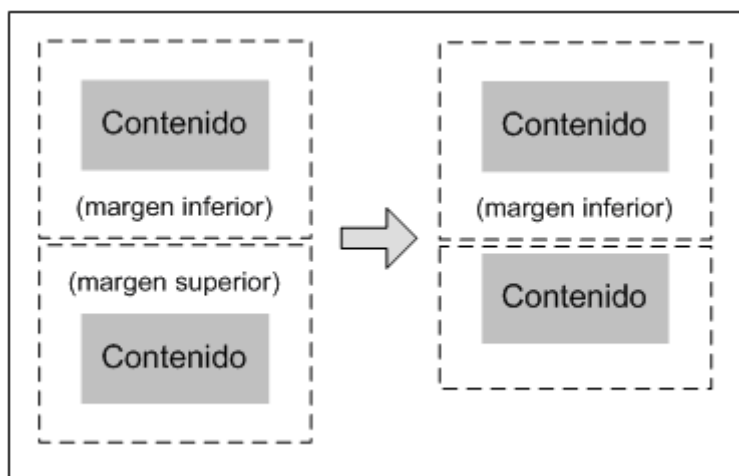
```
div img {  
  margin: .5em .5em .5em 1em;  
}
```



**Otra alternativa:**

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

El comportamiento de los **márgenes verticales** es más complejo de lo que se puede imaginar. 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.**



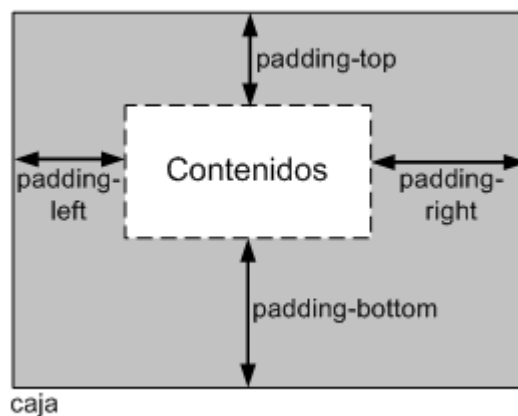
**Figura 4.7.** Fusión automática de los márgenes verticales

### 4.2.2. Relleno

CSS define cuatro propiedades para controlar cada uno de los espacios de relleno (espacio entre el contenido y el borde) horizontales y verticales de un elemento.

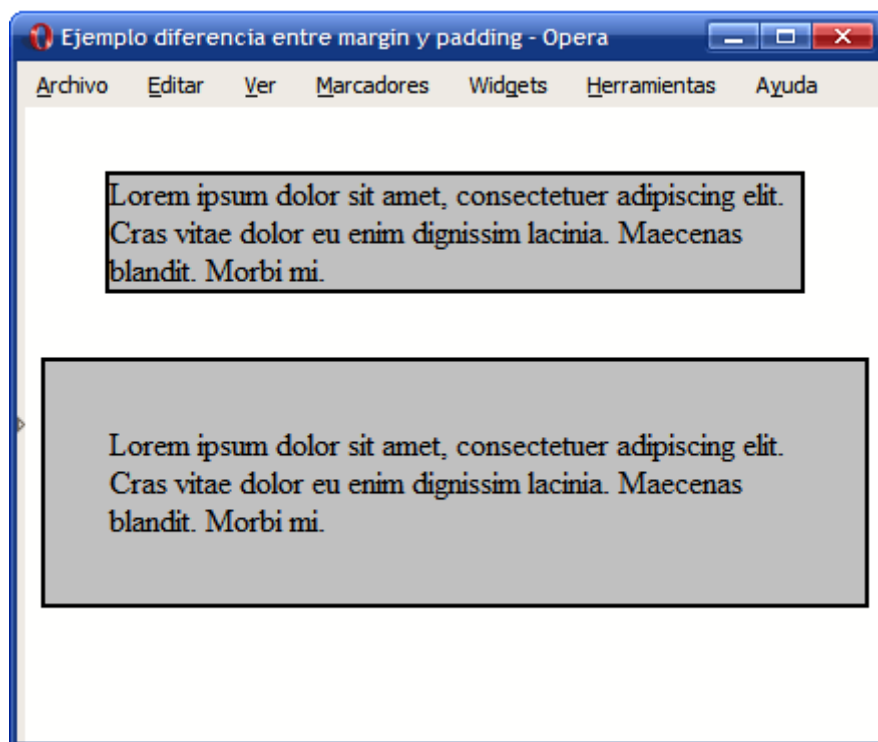
<b>padding-top</b>	<b><i>Relleno superior</i></b>
<b>padding-right</b>	<b><i>Relleno derecho</i></b>
<b>padding-bottom</b>	<b><i>Relleno inferior</i></b>
<b>padding-left</b>	<b><i>Relleno izquierdo</i></b>
<b>Valores</b>	<medida>   <porcentaje>   inherit
<b>Se aplica a</b>	Todos los elementos excepto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

Cada una de las propiedades establece la ***separación entre el lateral de los contenidos y el borde lateral de la caja***:



**Figura 4.8.** Las cuatro propiedades relacionadas con los rellenos

La siguiente imagen muestra la **diferencia entre el margen y el relleno**:



**Figura 4.9.** Diferencia entre relleno (padding) y margen (margin)

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

```
.margen {  
  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="margen">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>
```

Como sucede con la propiedad `margin`, CSS también define una **propiedad de tipo "shorthand"** para establecer los cuatro rellenos de un elemento de forma directa. La propiedad que permite definir de forma simultánea los cuatro márgenes se denomina **`padding`**.

<b>Padding</b>	<b>Relleno</b>
<b>Valores</b>	( <medida>   <porcentaje> ) {1, 4}   inherit
<b>Se aplica a</b>	Todos los elementos excepto algunos elementos de tablas como grupos de cabeceras y grupos de pies de tabla
<b>Valor inicial</b>	-
<b>Descripción</b>	<i><b>Establece de forma directa todos los rellenos de los elementos</b></i>

La notación {1, 4} de la definición anterior significa que la **propiedad padding** admite entre uno y cuatro valores, con el mismo significado que el de la propiedad *margin*.  
Ejemplo:

```
body {padding: 2em}
    /* Todos los rellenos valen 2em */
body {padding: 1em 2em}
    /* Superior e inferior = 1em, Izquierdo y derecho = 2em */
body {padding: 1em 2em 3m}
    /* Superior = 1em, derecho = 2m, Inferior = 3m, izquierdo = 2em */
body {padding: 1em 2em 3em 4em}
    /* Superior = 1em, derecho = 2em, inferior = 3em, izquierdo = 4em */
```

---

### ***Ejercicio 3***

---

## 4.3. 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 **anchura, su color y su estilo**.

### 4.3.1. Anchura

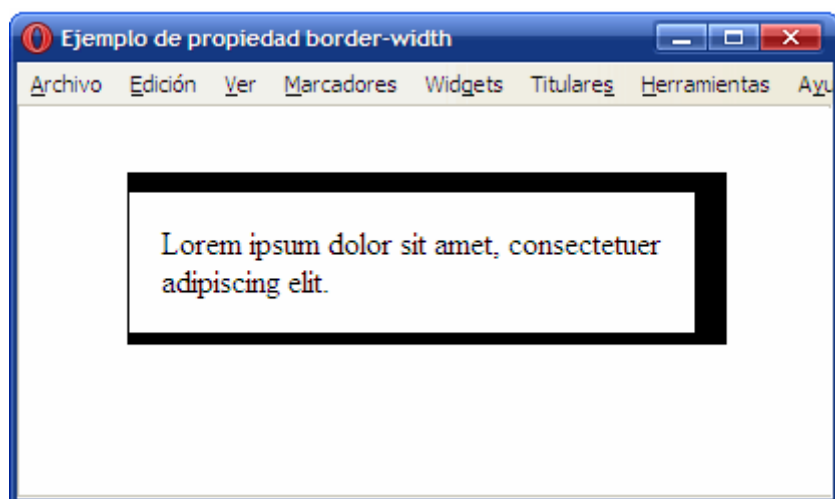
La anchura de los bordes se controla con las cuatro propiedades siguientes:

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

La anchura de los bordes se puede indicar mediante una medida (absoluta o relativa y en cualquier unidad de medida de las definidas) o mediante las palabras clave **thin** (borde delgado), **medium** (borde normal) y **thick** (borde ancho).

**La medida más habitual para indicar la anchura de los bordes es el píxel**, ya que permite un control preciso del grosor. Las **palabras clave apenas se utilizan**, ya que impiden mostrar bordes iguales en diferentes navegadores. El motivo es que el estándar CSS no indica explícitamente el grosor al que equivale cada palabra clave. Así por ejemplo, el grosor **medium** equivale a **4px** en Internet Explorer y **3px** en el resto de navegadores.

El siguiente ejemplo muestra un elemento con cuatro anchuras diferentes de borde:



**Figura 4.10.** Ejemplo de propiedad border-width

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;
}
```

Si se quiere establecer la misma anchura a todos los bordes, CSS permite la utilización de un atajo mediante una **propiedad de tipo "shorthand"**, que permiten indicar varias propiedades de forma resumida:

<b>border-width</b>	<i><b>Anchura del borde</b></i>
<b>Valores</b>	( <medida>   thin   medium   thick ) {1, 4}   inherit
<b>Se aplica a</b>	Todos los elementos
<b>Valor inicial</b>	Medium
<b>Descripción</b>	<i><b>Establece la anchura de todos los bordes del elemento</b></i>

La propiedad **border-width** permite indicar entre uno y cuatro valores. El significado de cada caso es el habitual de las propiedades "shorthand":

```
p { border-width: thin }           /* thin thin thin thin */
p { border-width: thin thick }     /* thin thick thin thick */
p { border-width: thin thick medium } /* thin thick medium thick */
p { border-width: thin thick medium thin } /* thin thick medium thin */
```

Si se indica un solo valor, se aplica a los cuatro bordes. Si se indican dos valores, el primero se aplica al borde superior e inferior y el segundo se aplica al borde izquierdo y derecho.

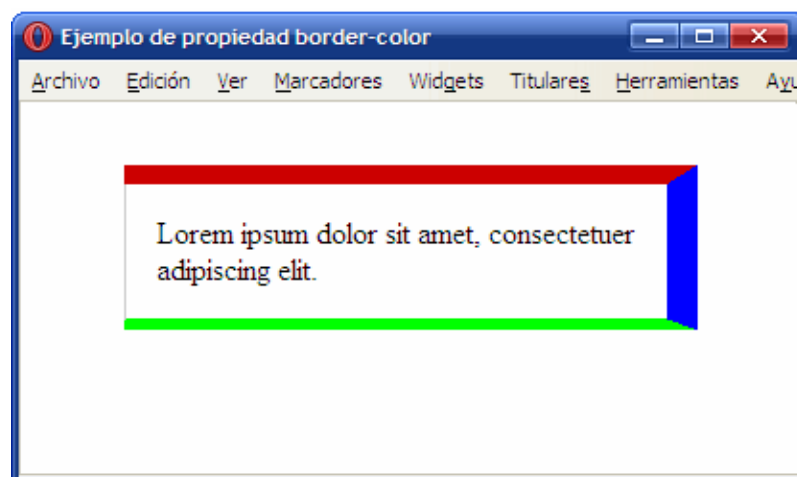
Si se indican tres valores, el primero se aplica al borde superior, el segundo se aplica al borde izquierdo y derecho y el tercer valor se aplica al borde inferior. Si se indican los cuatro valores, el orden de aplicación es superior, derecho, inferior e izquierdo.

### 4.3.2. Color

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

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

El ejemplo anterior se puede modificar para mostrar cada uno de los bordes de un color diferente:



**Figura 4.11.** Ejemplo de propiedad border-color

Las reglas CSS necesarias para mostrar los colores anteriores son las siguientes:

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

Si se quiere establecer el mismo color para todos los bordes, CSS permite la utilización de un atajo mediante una **propiedad de tipo "shorthand"**, que permiten indicar varias propiedades de forma resumida:

<b>border-color</b>	<i>Color del borde</i>
<b>Valores</b>	( <color>   transparent ) {1, 4}   inherit
<b>Se aplica a</b>	Todos los elementos
<b>Valor inicial</b>	-
<b>Descripción</b>	Establece el color de todos los bordes del elemento

En este caso, al igual que sucede con la propiedad `border-width`, es posible indicar de uno a cuatro valores y las reglas de aplicación son idénticas a la propiedad `border-width`.



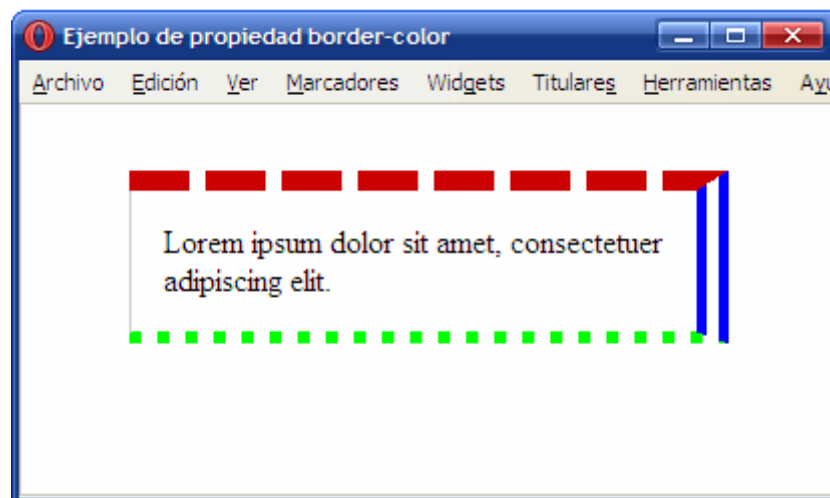
### 4.3.3. Estilo

Por último, CSS permite **establecer el estilo** de cada uno de los bordes mediante las siguientes propiedades:

<b>border-top-style</b>	<i>Estilo del borde superior</i>
<b>border-right-style</b>	<i>Estilo del borde derecho</i>
<b>border-bottom-style</b>	<i>Estilo del borde inferior</i>
<b>border-left-style</b>	<i>Estilo del borde izquierdo</i>
<b>Valores</b>	none   hidden   dotted   dashed   solid   double   groove   ridge   inset   outset   inherit
<b>Se aplica a</b>	Todos los elementos
<b>Valor inicial</b>	None (borde invisible)
<b>Descripción</b>	<i>Establece el estilo de cada uno de los cuatro bordes de los elementos</i>

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

Siguiendo el ejemplo anterior, se puede modificar el estilo de cada uno de los bordes:

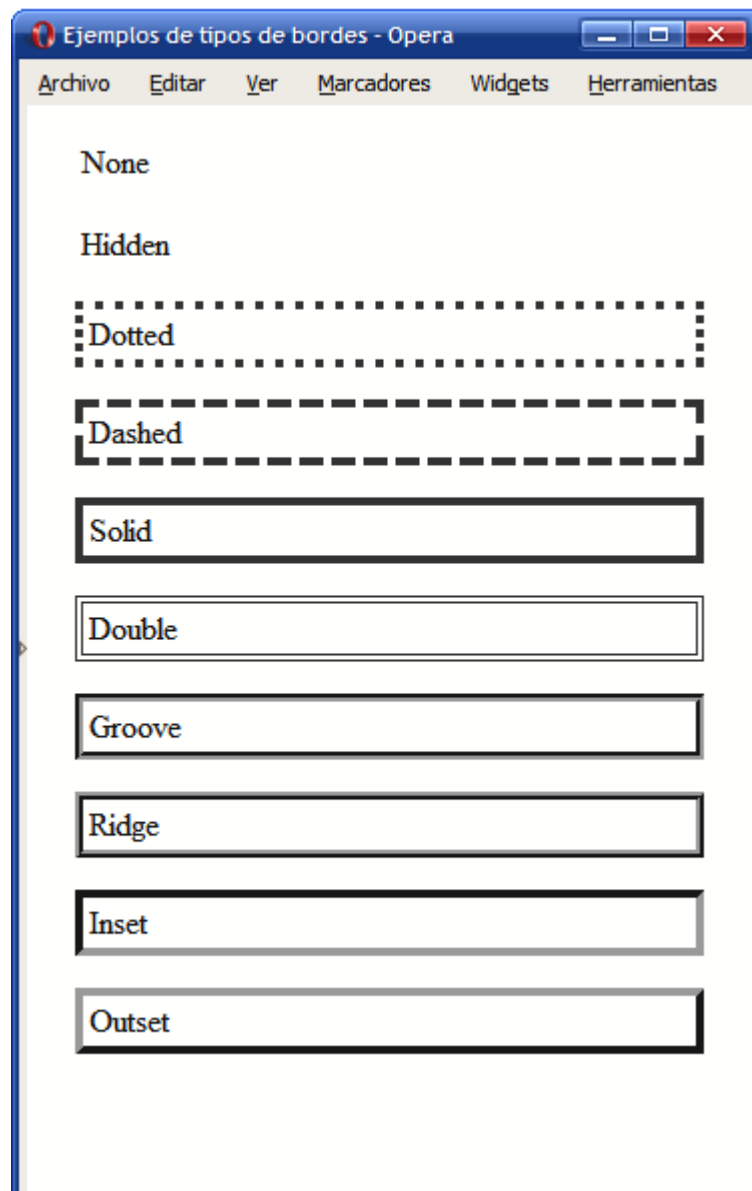


**Figura 4.12.** Ejemplo de propiedad border-style

Las reglas CSS necesarias para mostrar los estilos anteriores son las siguientes:

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

El aspecto con el que los navegadores muestran los diferentes tipos de borde se muestra a continuación:



**Figura 4.13.** Tipos de bordes definidos por CSS

Los bordes **más utilizados** en los diseños habituales son **solid** y **dashed**, seguidos de **double** y **dotted**. 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.

Si se quiere establecer el mismo estilo para todos los bordes, CSS define una **propiedad de tipo "shorthand"**:

<b>border-style</b>	<i>Estilo del borde</i>
<b>Valores</b>	(none   hidden   dotted   dashed   solid   double   groove   ridge   inset   outset ) {1, 4}   inherit
<b>Se aplica a</b>	Todos los elementos
<b>Valor inicial</b>	-
<b>Descripción</b>	<i>Establece el estilo de todos los bordes del elemento</i>

Como es habitual, la propiedad permite indicar de uno a cuatro valores diferentes y las reglas de aplicación son las habituales de las propiedades "shorthand".

#### 4.3.4. Propiedades shorthand

Como sucede con los márgenes y los rellenos, CSS define una serie de propiedades de tipo "shorthand" que permiten establecer todos los atributos de los bordes de forma directa. CSS ha definido una **propiedad "shorthand" para cada uno de los cuatro bordes** y una **propiedad "shorthand" global**.

<b>border-top</b>	<i>Estilo completo del borde superior</i>
<b>border-right</b>	<i>Estilo completo del borde derecho</i>
<b>border-bottom</b>	<i>Estilo completo del borde inferior</i>
<b>border-left</b>	<i>Estilo completo del borde izquierdo</i>
<b>Valores</b>	( <medida_borde>    <color_borde>    <estilo_borde> )   inherit
<b>Se aplica a</b>	Todos los elementos
<b>Valor inicial</b>	-
<b>Descripción</b>	<i>Establece el estilo completo de cada uno de los cuatro bordes de los elementos</i>

Las propiedades **"shorthand"** *permiten establecer alguno o todos los atributos de cada borde*. El siguiente ejemplo establece el color y el tipo del borde inferior, pero no su anchura:

```
h1 {  
  border-bottom: solid red;  
}
```

En el ejemplo anterior, la anchura del borde será la correspondiente al valor por defecto (**medium**). Este otro ejemplo muestra la forma habitual utilizada para establecer el estilo de cada borde:

```
div {  
  border-top: 1px solid #369;  
  border-bottom: #369 3px double;  
}
```

Por ultimo, CSS define una propiedad de tipo **"shorthand" global** para establecer el **valor de todos los atributos de todos los bordes** de forma directa:

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

Las siguientes reglas CSS son equivalentes:

```
div {  
  border-top: 1px solid red;  
  border-right: 1px solid red;  
  border-bottom: 1px solid red;  
  border-left: 1px solid red;  
}  
  
div { border: 1px solid red; }
```

Como el valor por defecto de la propiedad **border-style** es **none**, si una propiedad shorthand no establece explícitamente el estilo de un borde, el elemento no muestra ese borde!!

```
/* Sólo se establece el color, por lo que el estilo es "none" y el borde no se muestra */  
div { border: red; }
```

```
/* Se establece el grosor y el color del borde, pero no su estilo, por lo que es "none" y el  
borde no se muestra */  
div { border-bottom: 5px blue; }
```

Cuando los cuatro bordes no son idénticos pero sí muy parecidos, se puede utilizar la **propiedad border** para establecer de forma directa los atributos comunes de todos los bordes y **posteriormente especificar** para cada uno de los cuatro bordes sus **propiedades particulares**:

```
h1 {  
  border: solid #000;  
  border-top-width: 6px;  
  border-left-width: 8px;  
}
```

---

## Ejercicio 4

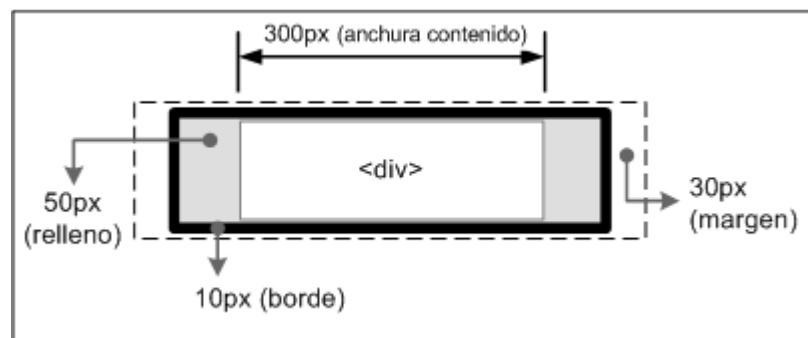
---

## 4.4. 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 se tienen en cuenta todos sus márgenes, rellenos y bordes:



**Figura 4.14.** La anchura total de un elemento tiene en cuenta los 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:

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

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*.

## 4.5. 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:

- **background-color.**
- **background-image.**
- **background-repeat.**
- **background-attachment.**
- **background-position.**

Y otra propiedad de tipo "*shorthand*": **background**.

<b>background-color</b>	<i>Color de fondo</i>
<b>Valores</b>	<code>&lt;color&gt;</code>   transparent   inherit
<b>Se aplica a</b>	Todos los elementos
<b>Valor inicial</b>	Transparent
<b>Descripción</b>	<i>Establece un color de fondo para los elementos</i>

El siguiente ejemplo muestra una página web con un color gris claro de fondo:

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

En ocasiones, es necesario crear un fondo más complejo que un simple color. CSS permite mostrar una **imagen como fondo de cualquier elemento**:

<b>background-image</b>	<b><i>Imagen de fondo</i></b>
<b>Valores</b>	<code>&lt;url&gt;</code>   none   inherit
<b>Se aplica a</b>	Todos los elementos
<b>Valor inicial</b>	None
<b>Descripción</b>	<b><i>Establece una imagen como fondo para los elementos</i></b>

CSS permite establecer de forma simultánea un color y una imagen de fondo. En este caso, la ***imagen se muestra delante del color***, por lo que solamente ***si la imagen contiene zonas transparentes es posible ver el color de fondo***.

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

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

Las imágenes de fondo se indican a través de su URL, que puede ser absoluta o relativa. Suele ser recomendable crear una ***carpeta de imágenes*** que se encuentre en el mismo directorio que los archivos CSS y que almacene todas las imágenes utilizadas en el diseño de las páginas.

Así, las imágenes correspondientes al diseño de la página se mantienen separadas del resto de imágenes del sitio y el código CSS es más sencillo (por **utilizar URL relativas**) y más fácil de mantener (por no tener que actualizar URL absolutas en caso de que se cambie la estructura del sitio web).

Por otra parte, 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, CSS la *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:**

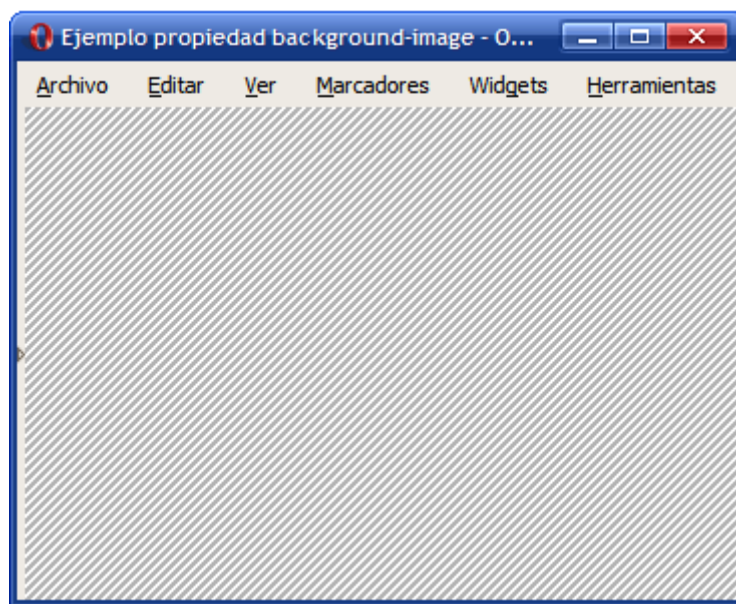


**Figura 4.15.** Imagen original utilizada para el fondo de la página

**Reglas CSS:**

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

**Resultado:**



**Figura 4.16.** Página con una imagen de fondo

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

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.**

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

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.

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`.

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

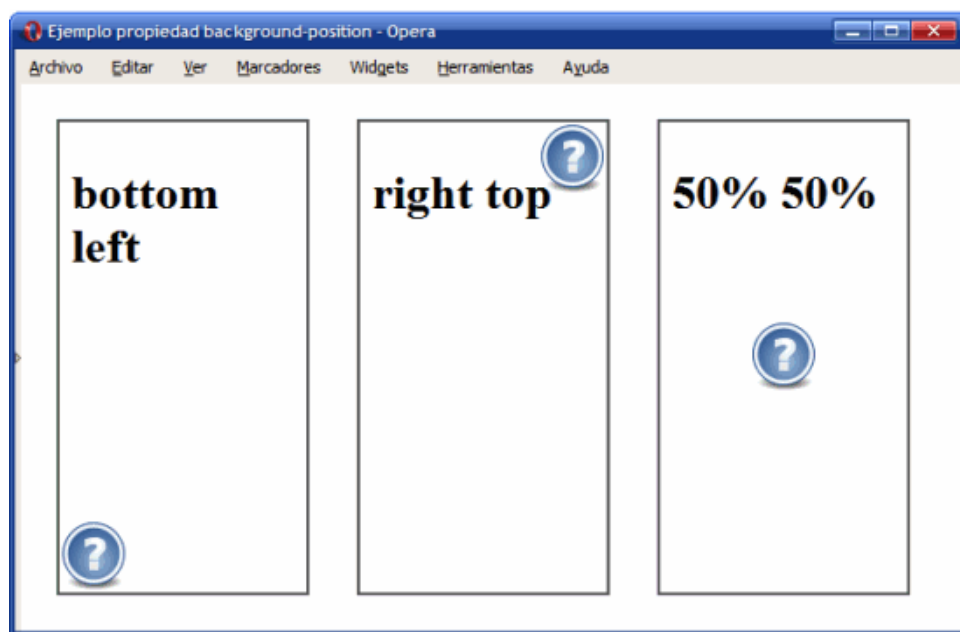
La propiedad `background-position` permite indicar **la distancia que se desplaza la imagen de fondo respecto de su posición original situada en la esquina superior izquierda.**

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). Si solamente se indica un porcentaje o una medida, se considera que es el desplazamiento horizontal y al desplazamiento vertical se le asigna automáticamente el valor de 50%.

Las **palabras clave** permitidas son equivalentes a algunos porcentajes significativos:

<b>top</b> = 0%	<b>left</b> = 0%	<b>center</b> = 50%	<b>bottom</b> = 100%	<b>right</b> = 100%.
-----------------	------------------	---------------------	----------------------	----------------------

El siguiente ejemplo muestra una misma imagen de fondo posicionada de tres formas diferentes:



**Figura 4.17.** Ejemplo de propiedad background-position

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%;
}
...
...
<div id="caja1"><h1>bottom left</h1></div>
<div id="caja2"><h1>right top</h1></div>
<div id="caja3"><h1>50% 50%</h1></div>
```

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**.

<b>background-attachment</b>	<b><i>Comportamiento de la imagen de fondo</i></b>
<b>Valores</b>	<b>scroll</b>   <b>fixed</b>   <b>inherit</b>
<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

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** y es la que generalmente se utiliza para establecer las propiedades del fondo de los elementos.

<b>background</b>	<b>Fondo de un elemento</b>
<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>	<b><i>Establece todas las propiedades del fondo de un elemento</i></b>

El **orden** en el que se indican las propiedades es **indiferente**, aunque **en general se sigue** el formato indicado de color, url de imagen, repetición y posición.

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("./imagenes/colorstrip.gif") repeat-x 0 0; }
```

```
/* La propiedad shorthand anterior es equivalente a las siguientes propiedades */
body {
  background-color: #222d2d;
  background-image: url("./imagenes/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:

```
background: url("./imagenes/colorstrip.gif") repeat-y;
```

```
background: url("./imagenes/colorstrip.gif") no-repeat bottom left;
```

```
background: transparent url("./imagenes/colorstrip.gif") no-repeat 0 -27px;
```

```
background: none;
```

---

## **Ejercicio 5**

---