

# Diseño de Interfaces Web

---

Bloque 2: Hojas de estilo CSS

**Tema 4: Flotar y posicionar**



# T4. FLOTAR Y POSICIONAR

---

1. Flotar
2. Posicionamiento
3. Estrategias para el layout de las páginas
4. Características avanzadas de CSS

## T4. FLOTAR Y POSICIONAR

---

CSS utiliza el **flotado** y el **posicionamiento** para tener el máximo **control** sobre el **lugar** que **ocupa cada elemento** en una página web, sus condiciones de visibilidad y "flotabilidad", así como controlar el manejo de capas.

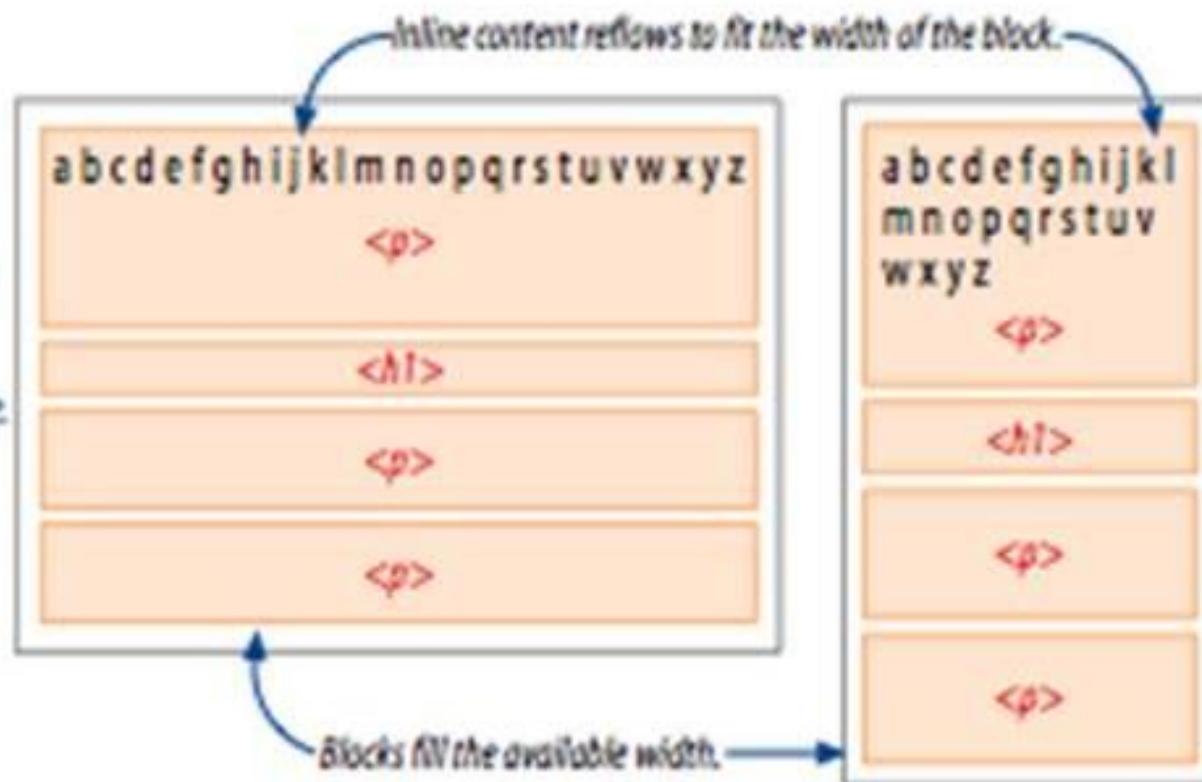
Algunas de las **propiedades** de CSS que se utilizan para **controlar el posicionamiento** de los elementos son: **float, clear, position, bottom, top, left, right, over-flow, clip, visibility, y z-index**.

Cuando hablamos de que los objetos de una página siguen el **flujo normal** del documento, queremos indicar que la **forma en la que se disponen** en la **ventana del navegador** **coincide** con el **lugar** que **ocupan** en el **documento** escrito.

## T4. FLOTAR Y POSICIONAR

Blocks are layed out in the order in which they appear in the source.

Each block starts on a new line.



## T4. FLOTAR Y POSICIONAR

### 1. Flotar

---

Flotar sirve para **mover** una **caja** a la **izquierda** o a la **derecha** hasta que su borde exterior toque el borde de la caja que lo contiene o toque otra caja flotante.

Para que un **elemento** pueda **flotar** debe tener **definido** implícita o explícitamente su **tamaño**.

Las **cajas flotantes no** se encuentran en el "flujo normal" del documento por lo que las cajas que sí siguen el flujo normal se **comportan como si las flotantes no estuviesen ahí**.

# T4. FLOTAR Y POSICIONAR

## 1. Flotar

---

**float:** Sirve para **crear diseños multicolumna**, barras de **navegación de listas no numeradas**, poner **contenido** en forma **tabular** pero **sin emplear tablas**.

Valores que puede tener la propiedad float:

- **none** hará que el objeto no sea flotante.
- **left** hace que el elemento flote a la izquierda.
- **right** hace que el elemento flote a la derecha.
- **inherit** hará que el elemento tome el valor de esta propiedad de su elemento padre.

# T4. FLOTAR Y POSICIONAR

## 1. Flotar

### float: imágenes

```
<p>They went down, down,...</p>
```

inline image in the normal flow



space next to the image is held clear

They went down, down, down, till at last they came to a passage with a door at one end, which was only fastened with a latch. The eldest Princess opened it, and they found themselves immediately in a lovely little wood, where the leaves were spangled with drops of silver which shone in the brilliant light of the moon. They next crossed another wood where the leaves were sprinkled with gold, and after that another still, where the leaves glistened with diamonds.

inline image floated to the right.

They went down, down, down, till at last they came to a passage with a door at one end, which was only fastened with a latch. The eldest Princess opened it, and they found themselves immediately in a lovely little wood, where the leaves were spangled with drops of silver which shone in the brilliant light of the moon. They next crossed another wood where the leaves were sprinkled with gold, and after that another still, where the leaves glistened with diamonds.

Image moves over and text wraps around it

```
img {  
    float: right;  
}
```

```
p {  
    padding: 15px;  
    background-color: #FFF799;  
    border: 2px solid #6C4788;  
}
```

# T4. FLOTAR Y POSICIONAR

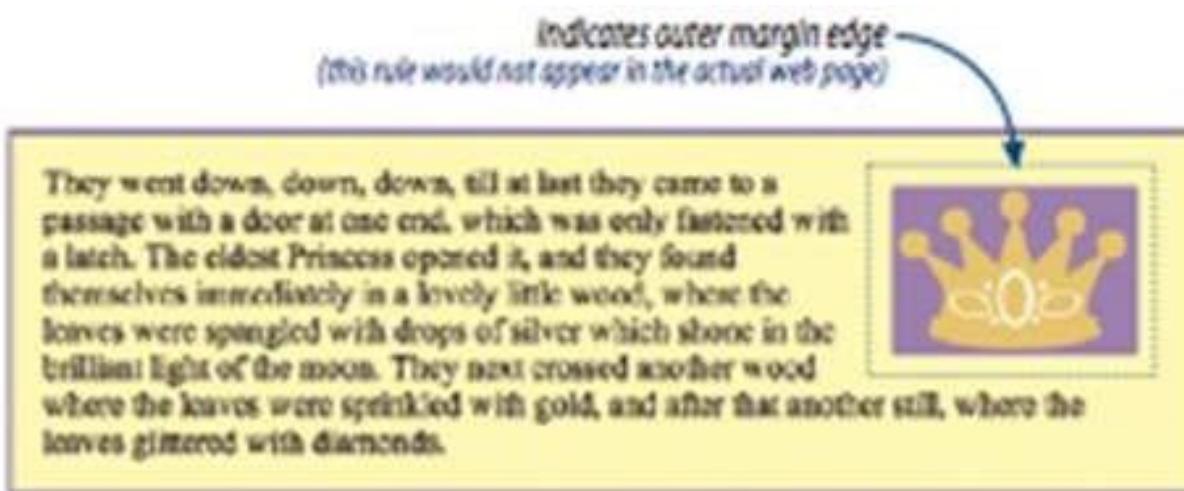
## 1. Flotar

### float: imágenes

```
<p>They went down, down, ...</p>
```

Indicates outer margin edge  
(this rule would not appear in the actual web page)

They went down, down, down, till at last they came to a passage with a door at one end, which was only fastened with a latch. The eldest Princess opened it, and they found themselves immediately in a lively little wood, where the leaves were spangled with drops of silver which shone in the brilliant light of the moon. They next crossed another wood where the leaves were sprinkled with gold, and after that another still, where the leaves glistened with diamonds.



```
img {  
    float: right;  
    margin: 10px;  
}
```

# T4. FLOTAR Y POSICIONAR

## 1. Flotar

---

### float: imágenes

Algunos **comportamientos** clave de **elementos flotantes** evidentes en las figuras anteriores:

- Un **elemento flotante** es **como una isla**.
- En primer lugar, se puede ver que la imagen es a la vez removida de su posición en el flujo normal, sin embargo, sigue influyendo en el contenido de los alrededores.
  - El texto del párrafo hace el espacio para el elemento img.
- Los **elementos flotantes permanecen** en el **área de contenido** del **elemento contenedor**.
- Es importante tener en cuenta que la **imagen flotada se coloca dentro** del **área de contenido** (los bordes interiores) del párrafo que lo contiene. No se extiende a la zona de relleno del párrafo.

# T4. FLOTAR Y POSICIONAR

## 1. Flotar

### float: elementos de texto inline

```
<p><span class="disclaimer">Disclaimer: The existence of silver, gold,  
and diamond trees is not confirmed.</span>They went down, down, down,  
till at last they came to a passage... </p>
```

They went down, down, down, till at last they came to a passage with a door at one end, which was only fastened with a latch. The eldest Princess opened it, and they found themselves immediately in a lovely little wood, where the leaves were spangled with drops of silver which shone in the brilliant light of the moon. They next crossed another wood where the leaves were sprinkled with gold, and after that another still, where the leaves glinted with diamonds. At last the Sun Gazer perceived a large lake, and on the shores of the lake twelve little boats with awnings, in which were seated twelve princes, who, grasping their oars, awaited the princesses.

Disclaimer: The existence of  
silver, gold, and diamond trees  
is not confirmed.

```
span.disclaimer {  
    float: right;  
    margin: 10px;  
    width: 200px;  
    color: #FFF;  
    background-color: #9D0800;  
    padding: 4px;  
}  
  
p {  
    padding: 15px;  
    background-color: #FFF799;  
    border: 2px solid #6C4788;  
}
```

# T4. FLOTAR Y POSICIONAR

## 1. Flotar

---

### float: elementos de texto inline

Algunos **comportamientos** clave de **elementos de texto inline que son flotantes** evidentes en la figura anterior:

- Siempre hay que proporcionar una anchura (width)
  - Si no pusiéramos una anchura, la caja del área del contenido se expandiría a la máxima anchura.
- Los **elementos flotantes se comportan como elementos de bloque.**
  - Una vez que flotamos un elemento inline, entonces sigue las reglas de los elementos de bloque.
- Es importante tener en cuenta que la **imagen flotada se coloca dentro del área de contenido** (los bordes interiores) del párrafo que lo contiene. No se extiende a la zona de relleno del párrafo.

# T4. FLOTAR Y POSICIONAR

## 1. Flotar

### float: elementos de bloque

```
<p>ONCE upon a time....</p>
<p id="float">As he had a white skin, blue eyes,...</p>
<p>The fact was he thought then very ugly...</p>
```



```
pmfloat {
    float: left;
    width: 200px;
    margin-top: 0px;
    background: #A5D3OE;
}

p {
    border: 1px solid red;
}
```

# T4. FLOTAR Y POSICIONAR

## 1. Flotar

---

### **float: elementos de bloque**

Algunos **comportamientos** clave de **elementos de bloque flotantes** evidentes en la figura anterior:

- El párrafo flotado se mueve a la izquierda y el contenido siguiente lo envuelve, aunque el comportamiento normal en los bloques sería que se apilaran uno encima del otro.
- Hay que proporcionar una anchura (width), si no completarán toda la anchura disponible.

# T4. FLOTAR Y POSICIONAR

## 1. Flotar

---

**clear:** Sirve para mantener **limpia** el **área** que está **al lado** del **elemento flotante** y que el **siguiente elemento comience** en su **posición normal** dentro del bloque que lo contiene.

Valores que puede tener la propiedad clear:

- **left** indica que el elemento se colocará debajo de cualquier otro elemento del bloque al que pertenece que estuviese flotando a la izquierda.
- **right** funciona como el left pero en este caso el elemento deberá estar flotando a la derecha.
- **both** mueve hacia abajo el elemento hasta que esté limpio de elementos flotantes a ambos lados.
- **none** permite elementos flotantes a ambos lados. Es el valor por defecto.
- **inherit** indica que heredará el valor de la propiedad clear de su elemento padre.

# T4. FLOTAR Y POSICIONAR

## 1. Flotar

clear:



```
img {  
    float: left;  
    margin-right: 10px;  
}  
  
h2 {  
    clear: left;  
    margin-top: 2em;  
}
```

# T4. FLOTAR Y POSICIONAR

## 1. Flotar

---

```
<html>
<head>
<link rel="stylesheet" type="text/css" href="estilos.css" />
<title>Flotar</title>
</head>
<body>
<div id="contenido">
<div id="caja1"> Caja 1 </div>
<div id="caja2"> Caja 2 </div>
<div id="caja3"> Caja 3 </div>
</div>
</body>
</html>
```

# T4. FLOTAR Y POSICIONAR

## 1. Flotar

---

```
body { font-weight: bold; color: white; }

#contenido {
    /* Ancho de 700px y centrado en el navegador */
    width: 700px; margin: auto;
    /* Borde sólido y fondo gris claro */
    border: solid 1px black; background-color: #CCCCCC; }

#caja1 {
    width: 100px; height: 100px; background-color:#FF0000;
    border: solid 1px black; margin: 10px; }

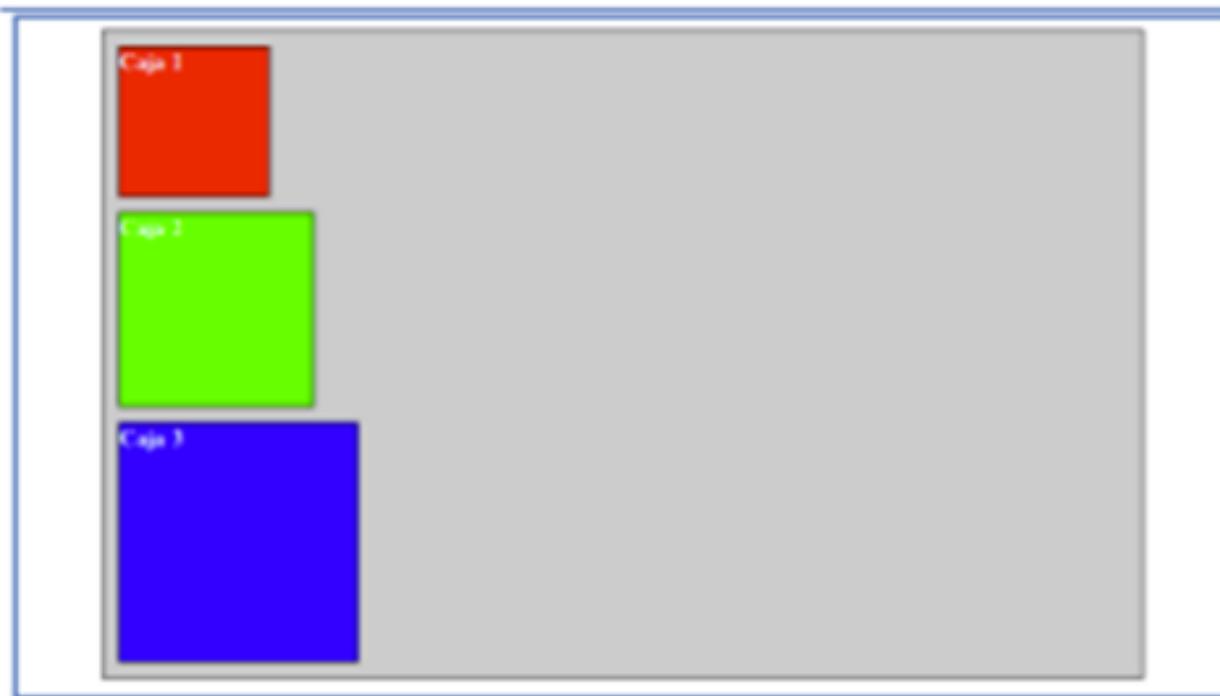
#caja2 {
    width: 130px; height: 130px; background-color:#00FF00;
    border: solid 1px black; margin: 10px; }

#caja3 {
    width: 160px; height: 160px; background-color:#0000FF;
    border: solid 1px black; margin: 10px; }
```

# T4. FLOTAR Y POSICIONAR

## 1. Flotar

---



# T4. FLOTAR Y POSICIONAR

## 1. Flotar

Flotar la Caja1 a la derecha

```
#caja1{  
    width: 100px; height: 100px;  
    background-color: #FF0000;  
    border: solid 1px black;  
    margin: 10px;  
    /* Flotar a la derecha */  
    float: right;  
}
```

La caja1 ya no se encuentra en el flujo del documento, no ocupa espacio.



# T4. FLOTAR Y POSICIONAR

## 1. Flotar

Flotar la Caja1 a la izquierda

```
#caja1{  
    width: 100px; height: 100px;  
    background-color: #FF0000;  
    border: solid 1px black;  
    margin: 10px;  
    /* Flotar a la izquierda */  
    float: left;  
}
```

La caja1 ya no se encuentra en el flujo del documento, no ocupa espacio y se sitúa sobre la caja2, ocultando así parte de ella.



# T4. FLOTAR Y POSICIONAR

## 1. Flotar

Flotar la Caja2 a la izquierda

```
#caja2{  
    width: 130px; height: 130px;  
    background-color: #00FF00;  
    border: solid 1px black;  
    margin: 10px;  
    /* Flotar a la izquierda */  
    float: left;  
}
```

La Caja3, junto con el Contenedor, es la única que queda en el flujo normal del documento. Debemos fijarnos que el tamaño del Contenedor se adapta al de los elementos que siguen en el flujo normal.



# T4. FLOTAR Y POSICIONAR

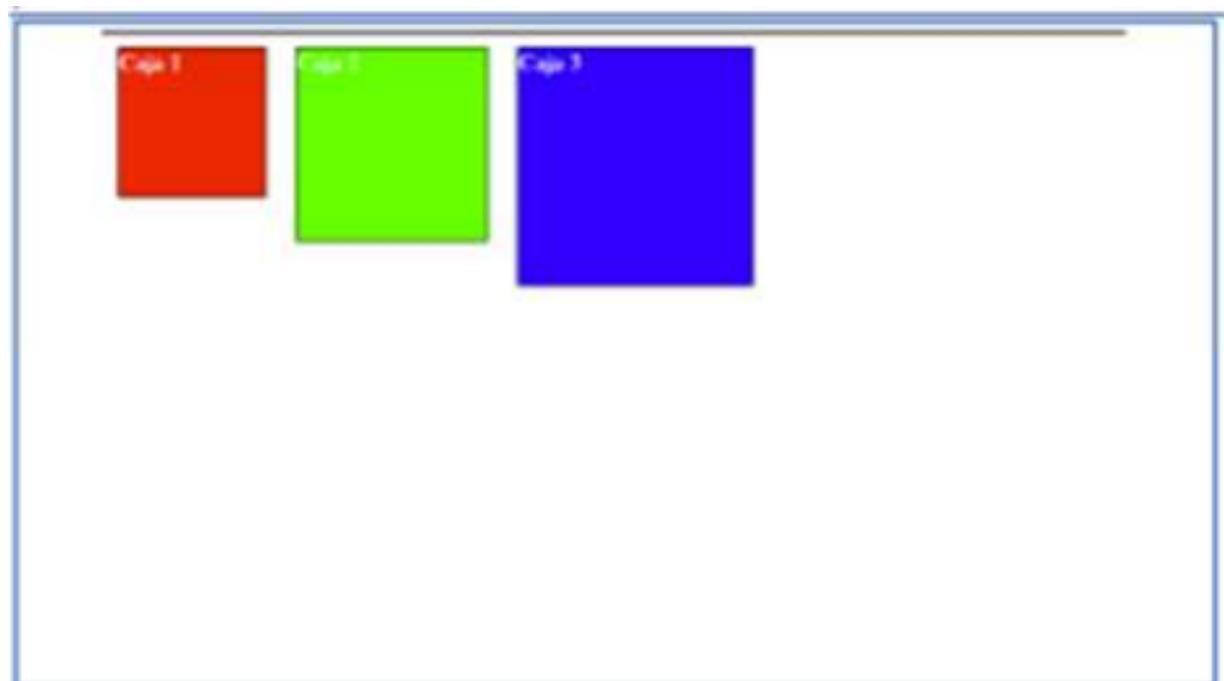
## 1. Flotar

Flotar la Caja3 a la izquierda

```
#caja3{  
    width: 160px; height: 160px;  
    background-color: #0000FF;  
    border: solid 1px black;  
    margin: 10px;  
    /* Flotar a la izquierda */  
    float: left;  
}
```

Los div de las cajas de colores siguen estando en el interior del div contenido y eso se nota en que no salen del ancho marcado por éste y que su alineamiento es con respecto a él.

El div **Contenido** tiene ahora una altura de **0px**, ya que **todo lo que tiene en su interior está fuera del flujo del documento**. Como no tiene nada dentro, no necesita tener una altura determinada.



# T4. FLOTAR Y POSICIONAR

## 1. Flotar

---

**Alternativas para volver a ver el Contenido:**

- a) aplicar una altura configurando en el selector #contenido la propiedad height con una altura mayor que el alto de la caja más grande.
- b) Se puede **flotar a la izquierda** la caja con id #contenido. De esta forma volvería a aparecer pero flotando a la izquierda de su contenedor (body).
- c) Añadir, **dentro del div Contenido** y a continuación de las tres cajas flotadas, un **div totalmente vacío**. Este div lo configuramos estableciendo la propiedad **clear con el valor both** (en este caso también serviría left, pues todas las cajas flotan a la izquierda). Esta es la solución más utilizada.

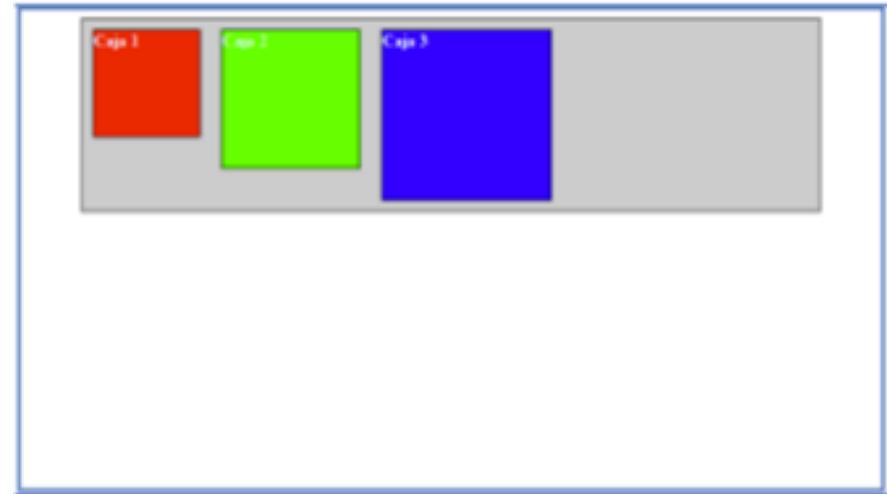
# T4. FLOTAR Y POSICIONAR

## 1. Flotar

Añadir un div dentro del div contenido:

fichero .html

```
<body>
<div id="contenido">
<div id="caja1"> Caja 1 </div>
<div id="caja2"> Caja 2 </div>
<div id="caja3"> Caja 3 </div>
<!-- Caja vacía para limpiar flotados --&gt;
&lt;div class="clearboth"&gt;&lt;/div&gt;
&lt;/div&gt;
&lt;/body&gt;</pre>
```



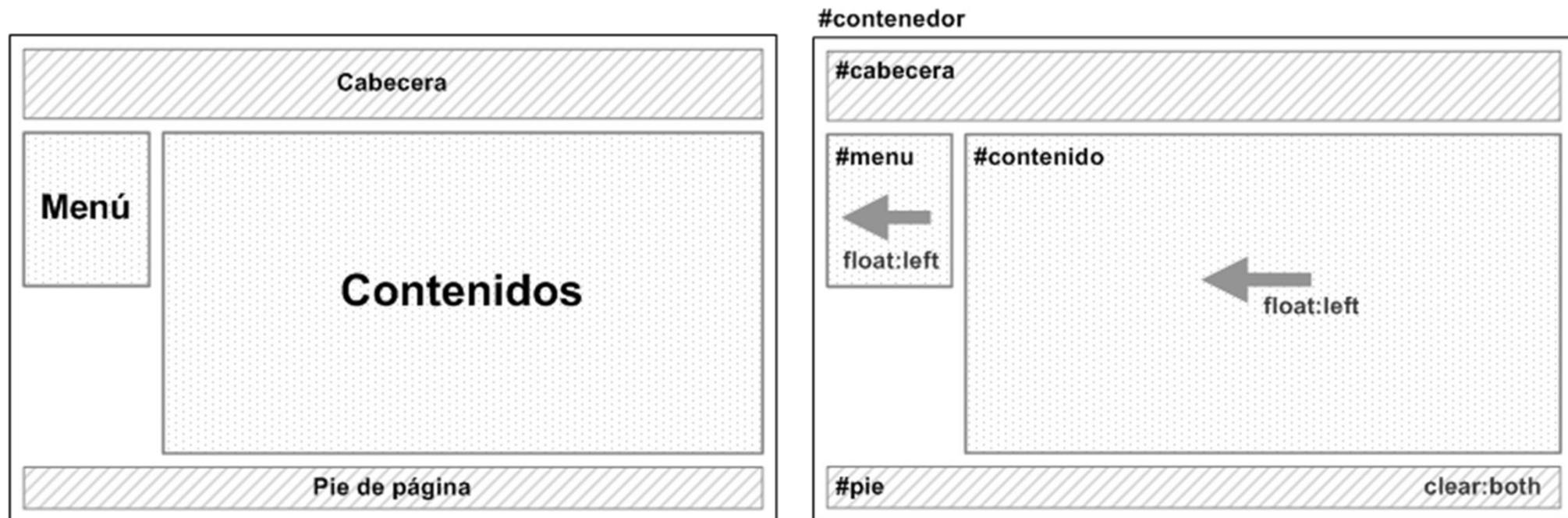
fichero .css

```
/* Limpiar flotados a izquierda y derecha */
.clearboth{clear: both;}
```

# T4. FLOTAR Y POSICIONAR

## 1. Flotar

Diseño a 2 columnas con cabecera y pie de página:



# T4. FLOTAR Y POSICIONAR

## 1. Flotar

---

Diseño a 2 columnas con cabecera y pie de página:

```
<body>
<div id="contenedor">
  <div id="cabecera">
    </div>

  <div id="menu">
    </div>

  <div id="contenido">
    </div>

  <div id="pie">
    </div>
</div>
</body>
```

```
#contenedor {
  width: 700px;
}

#cabecera {}

#menu {
  float: left;
  width: 150px;
}

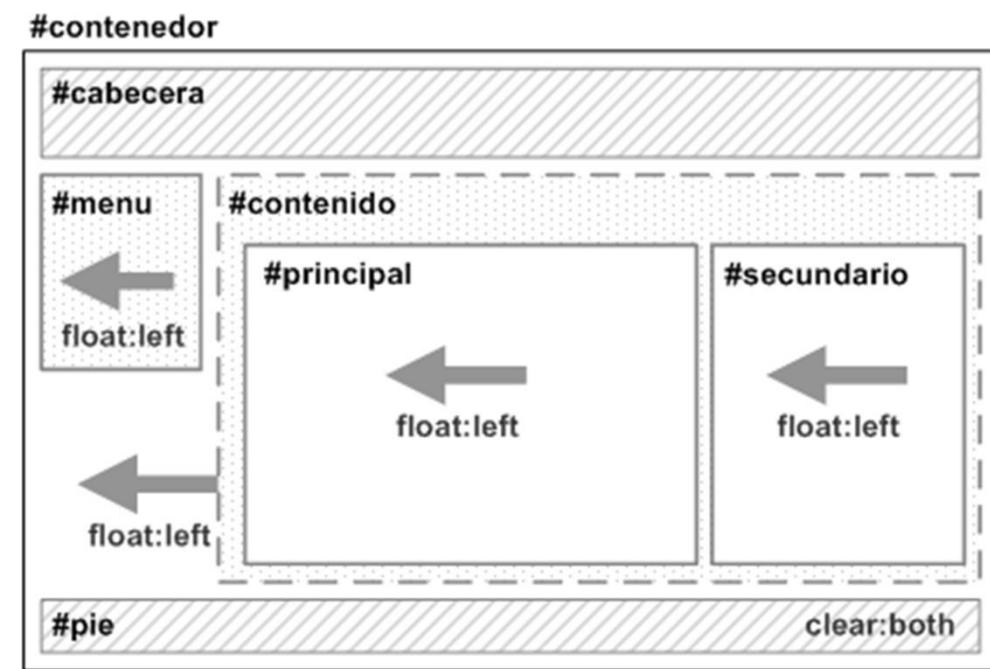
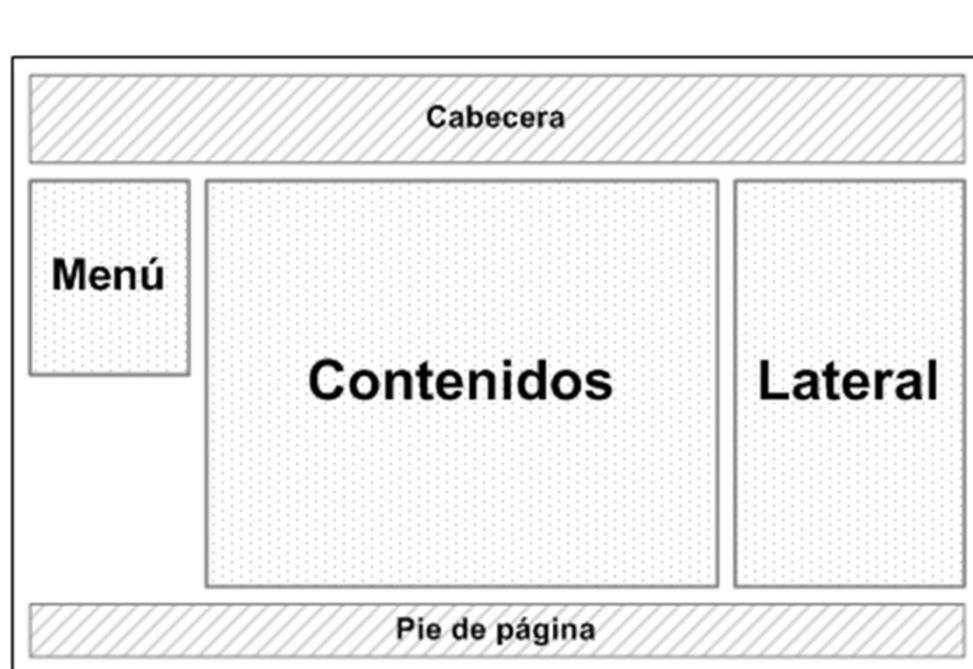
#contenido {
  float: left;
  width: 550px;
}

#pie {
  clear: both;
}
```

# T4. FLOTAR Y POSICIONAR

## 1. Flotar

Diseño a 3 columnas con cabecera y pie de página:



# T4. FLOTAR Y POSICIONAR

## 1. Flotar

---

Diseño a 3 columnas con cabecera y pie de página:

```
<body>
<div id="contenedor">
  <div id="cabecera">
    </div>
  <div id="menu">
    </div>
  <div id="contenido">
    <div id="principal">
      </div>
    <div id="secundario">
      </div>
    </div>
    <div id="pie">
      </div>
    </div>
  </div>
</body>
```

```
#contenedor {}
#cabecera {}
#menu {
  float: left;
  width: 15%;}
#contenido {
  float: left;
  width: 85%;}
#contenido #principal {
  float: left;
  width: 80%;}
#contenido #secundario {
  float: left;
  width: 20%;}
#pie {
  clear: both;}
```

# T4. FLOTAR Y POSICIONAR

## 2. Posicionamiento

---

**display:** propiedad que permite al documento interpretar de otra forma los elementos de tipo bloque y los elementos de tipo línea.

Valores que puede tomar la propiedad display:

- **block:** si quieres que un elemento "en línea" se comporte como un elemento de tipo bloque (enlaces que forman el menú de navegación).
- **inline:** si quieres que un elemento de tipo bloque se comporte como un elemento en linea (listas que se quieren mostrar horizontalmente).
- **none:** si quieres que un elemento de bloque no genere caja, no muestre su contenido y no ocupe espacio en la página.

# T4. FLOTAR Y POSICIONAR

## 2. Posicionamiento

DIV normal

DIV con display:inline

Enlace normal

Enlace con display:block

```
<div>DIV normal</div>
```

```
<div style="display:inline">DIV con display:inline</div>
```

```
<a href="#">Enlace normal</a>
```

```
<a href="#" style="display:block">Enlace con display:block</a>
```

# T4. FLOTAR Y POSICIONAR

## 2. Posicionamiento

---

**position:** propiedad que permite posicionar los elementos en un documento.

Valores que puede tomar la propiedad position:

- **static**
- **relative**
- **absolute**
- **fixed**

# T4. FLOTAR Y POSICIONAR

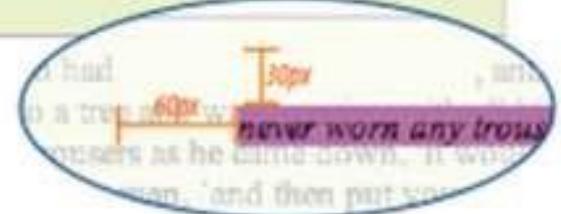
## 2. Posicionamiento

### position:

- **static**: permite colocar a los elementos según el flujo normal. Es el valor que asumirá por defecto en todos los elementos HTML.
- **relative**: consiste en posicionar una caja según el posicionamiento normal y después desplazarla respecto de su posición original estableciendo para ello una distancia vertical y/o horizontal.

```
en {  
    position: relative;  
    top: 30px;  
    left: 60px;  
    background-color: fuchsia;  
}
```

Along the road he came upon a man who had ~~never worn any trousers~~, and who was trying to put on a pair. So he had fastened them to a tree and was ~~never wear any trousers~~ right up in the air so that he should hit the two legs of the trousers as he came down. It would be much better if you held them in your hands,' said the young man, 'and then put your legs one after the other in each hole.'



# T4. FLOTAR Y POSICIONAR

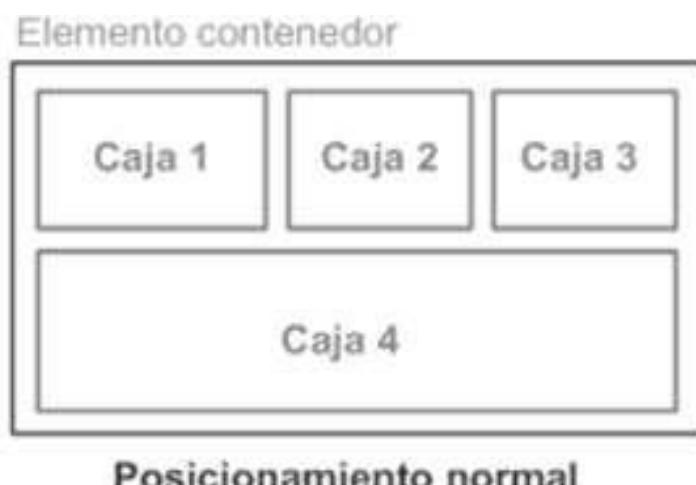
## 2. Posicionamiento

### position:

- **relative:**

El desplazamiento relativo de una caja no afecta al resto de cajas adyacentes, que se muestran en la misma posición que si la caja desplazada no se hubiera movido de su posición original.

```
#caja2 { width: 130px; height: 130px;  
background-color:#00FF00; border: solid 1px black; margin: 10px;  
/* Posicionamiento relativo */  
position: relative;  
left: 50px; top: 50px; }
```

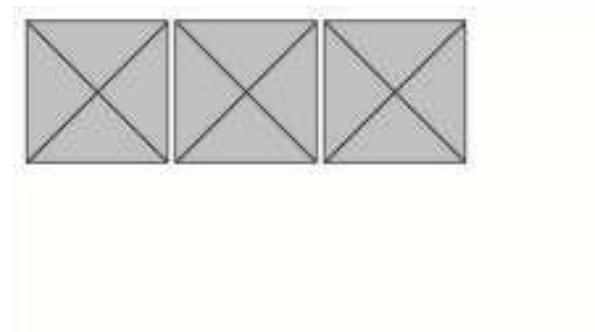


# T4. FLOTAR Y POSICIONAR

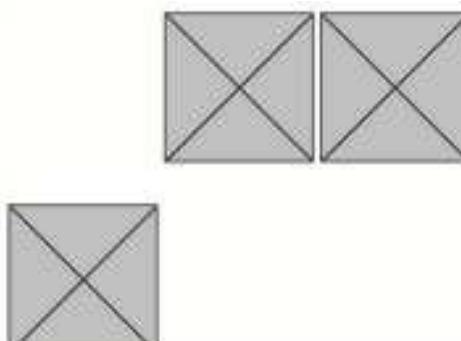
## 2. Posicionamiento

**position:**

- **relative:**



```
img.desplazada {  
    position: relative;  
    top: 8em;  
}
```



```
  
  

```

# T4. FLOTAR Y POSICIONAR

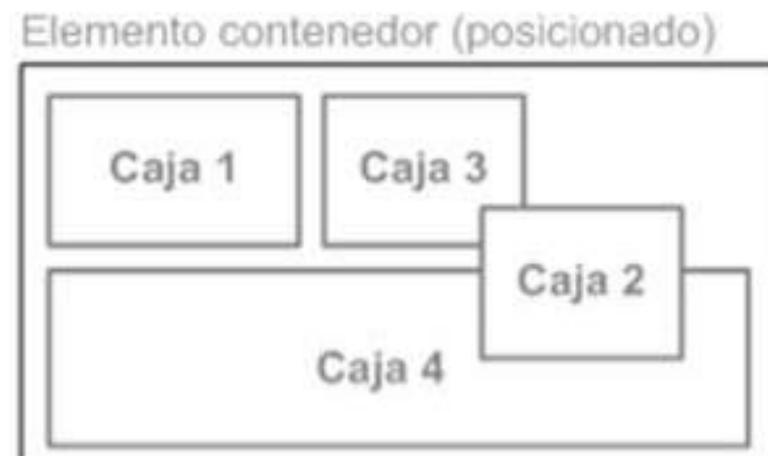
## 2. Posicionamiento

### position:

- **absolute** la posición de una caja se establece de forma absoluta respecto de su elemento contenedor y el resto de elementos de la página ignoran la nueva posición del elemento.



Posicionamiento normal



Posicionamiento absoluto de la caja 2

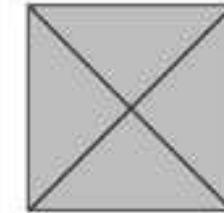
# T4. FLOTAR Y POSICIONAR

## **2. Posicionamiento**

# position:

- **absolute**

```
div {  
    border: 2px solid #CCC;  
    padding: 1em;  
    margin: 1em 0 1em 4em;  
    width: 300px;  
}
```



*Etiam* non possumus. *Nam* etiam si quis  
adipisci posset, non potest. *Quod* non  
potest, non potest.

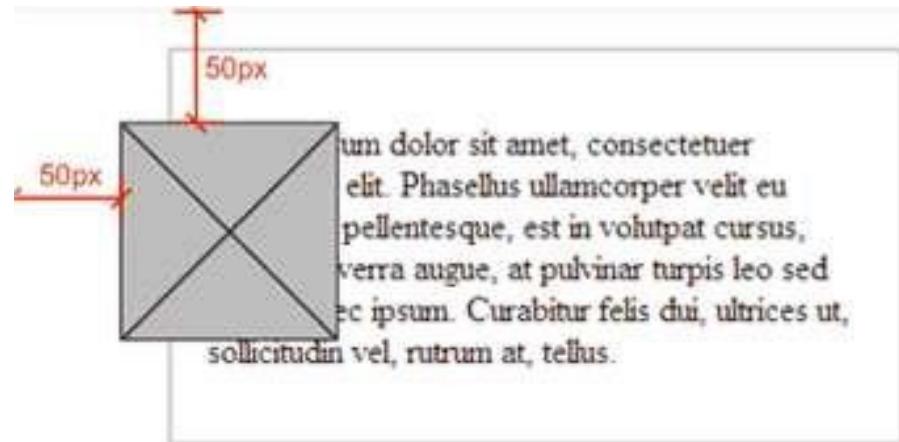
# T4. FLOTAR Y POSICIONAR

## 2. Posicionamiento

### position:

- **absolute**

```
div img {  
    position: absolute;  
    top: 50px;  
    left: 50px;  
}
```



La imagen posicionada de forma absoluta no toma como referencia su elemento contenedor <div>, sino la ventana del navegador.

Como ningún elemento contenedor está posicionado, la referencia es la ventana del navegador.

Como la imagen se posiciona de forma absoluta, el resto de elementos de la página se mueven para ocupar el lugar libre dejado por la imagen.

# T4. FLOTAR Y POSICIONAR

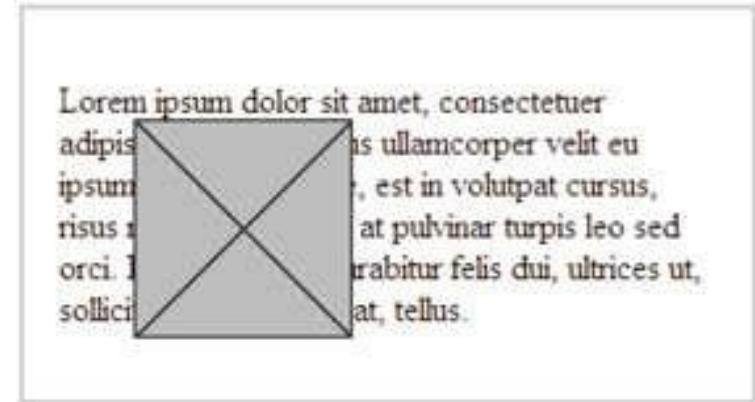
## 2. Posicionamiento

### position:

- **absolute**

```
div {  
    border: 2px solid #CCC;  
    padding: 1em;  
    margin: 1em 0 1em 4em;  
    width: 300px;  
    position: relative;  
}
```

```
div img {  
    position: absolute;  
    top: 50px;  
    left: 50px;  
}
```



La única propiedad añadida al <div> es position: relative por lo que el elemento contenedor se posiciona pero no se desplaza respecto de su posición original.

En este caso, como el elemento contenedor de la imagen está posicionado, se convierte en la referencia para el posicionamiento absoluto.

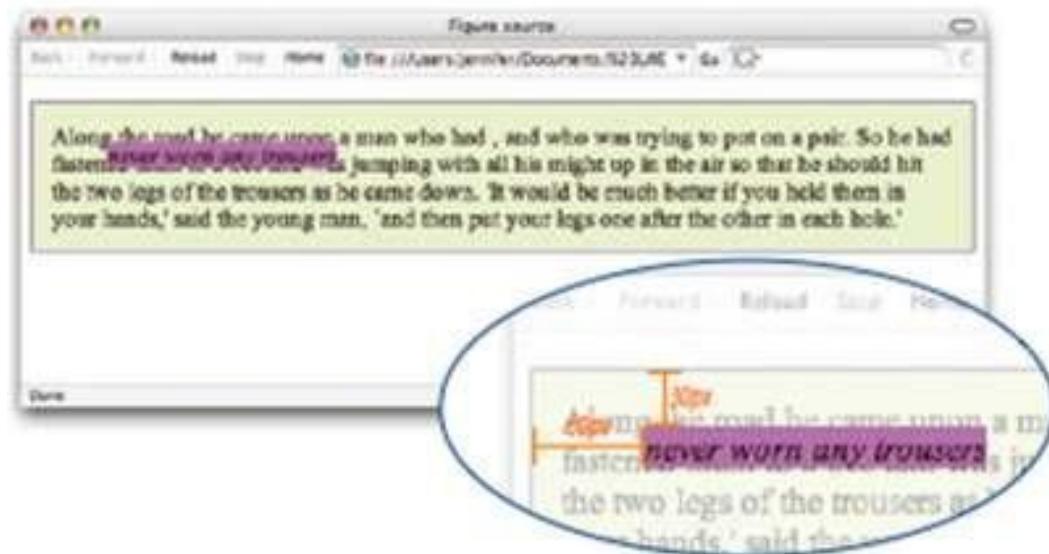
# T4. FLOTAR Y POSICIONAR

## 2. Posicionamiento

position:

- absolute

```
p {  
    position: relative;  
    padding: 15px;  
    background-color: #DBF0BA;  
    border: 2px solid #6C4788;  
}
```



En este caso, como el elemento contenedor de la imagen está posicionado, se convierte en la referencia para el posicionamiento.

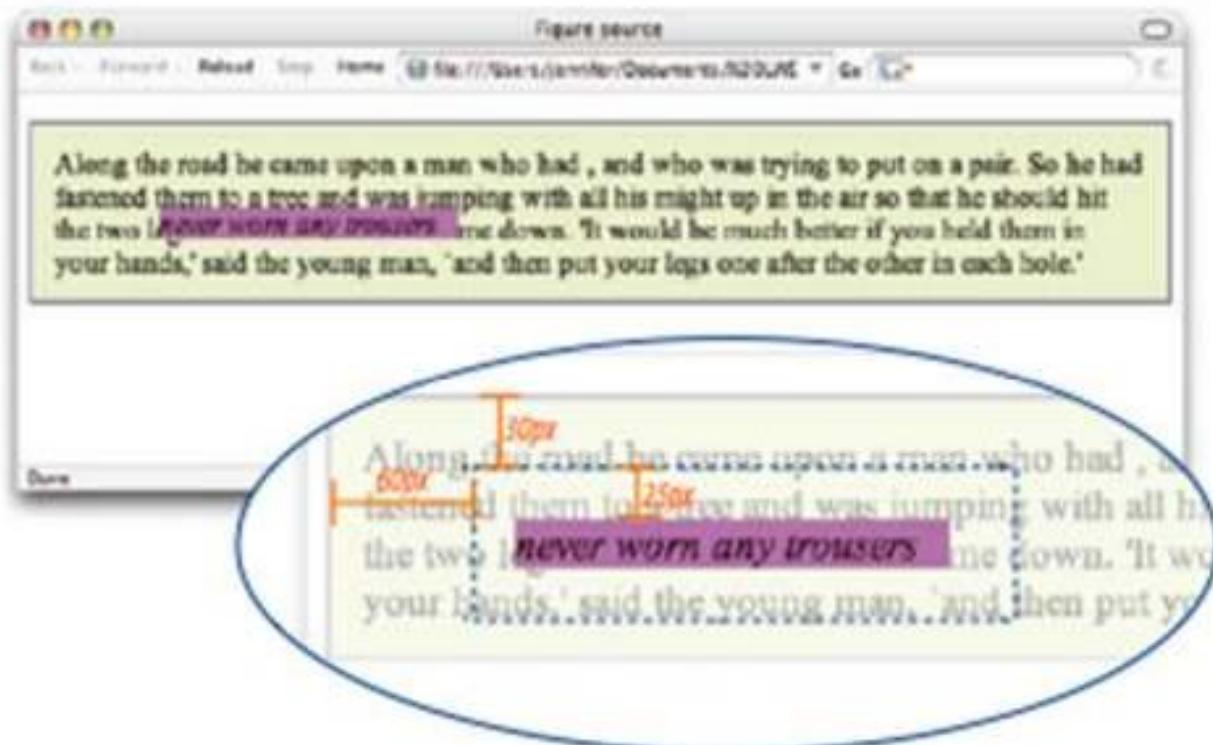
# T4. FLOTAR Y POSICIONAR

## 2. Posicionamiento

position:

- absolute

```
en {  
    width: 200px;  
    margin: 25px;  
    position: absolute;  
    top: 30px;  
    left: 60px;  
    background-color: fuchsia;  
}
```



En este caso, el valor del desplazamiento se aplica a partir del margen de la caja del elemento.

# T4. FLOTAR Y POSICIONAR

## 2. Posicionamiento

---

### position:

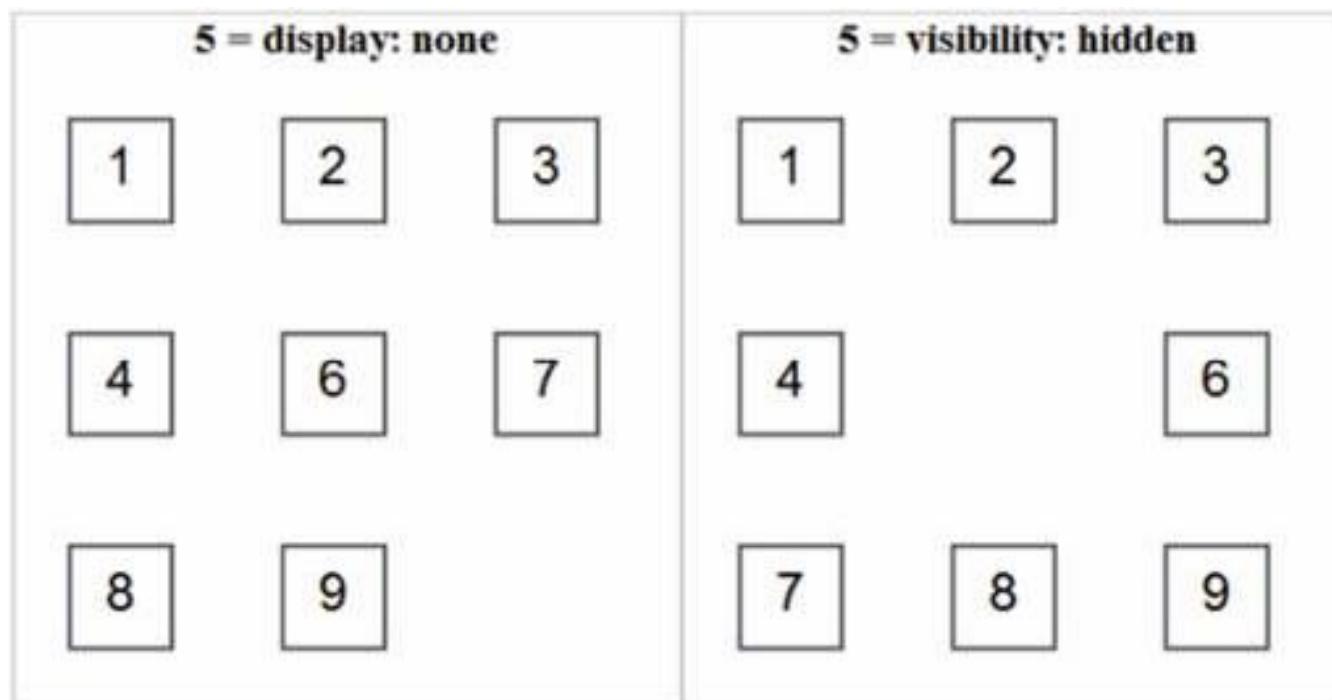
- **fixed:** es parecido al posicionamiento absoluto pero posiciona con respecto a la **ventana del navegador** apareciendo en la misma posición aunque el usuario se desplace por la página con las barras de desplazamiento. Esta característica puede ser útil para crear encabezados o pies de página en páginas HTML preparadas para imprimir.

# T4. FLOTAR Y POSICIONAR

## 2. Posicionamiento

**visibility:** propiedad que controla si el elemento será visualizado según le asignes el valor **visible** o **hidden**.

Aunque un elemento no sea visible, éste continúa ocupando su espacio en el flujo normal del documento al contrario de lo que ocurría con la propiedad display cuando se le asignaba el valor none.



# T4. FLOTAR Y POSICIONAR

## 2. Posicionamiento

---

**overflow:** controlar la forma en la que se visualizan los contenidos que sobresalen de sus elementos.

Los valores de la propiedad overflow:

- **visible:** el contenido no se corta y se muestra sobresaliendo la zona reservada para visualizar el elemento. Este es el comportamiento por defecto.
- **hidden:** el contenido sobrante se oculta y sólo se visualiza la parte del contenido que cabe dentro de la zona reservada para el elemento.
- **scroll:** solamente se visualiza el contenido que cabe dentro de la zona reservada para el elemento, pero también se muestran barras de scroll que permiten visualizar el resto del contenido.
- **auto:** el comportamiento depende del navegador, aunque normalmente es el mismo que la propiedad scroll.

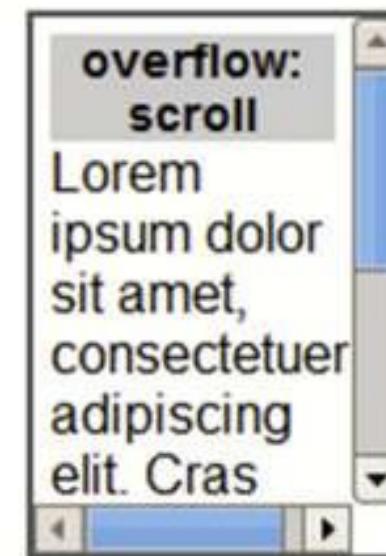
# T4. FLOTAR Y POSICIONAR

## 2. Posicionamiento

**overflow:**



lacinia.  
Maecenas  
blandit. Morbi  
mi.



# T4. FLOTAR Y POSICIONAR

## 2. Posicionamiento

---

**z-index:** Permite controlar el **orden** en el que se **presentan los elementos** que quedan **solapados** por efecto de otras propiedades.

z-index permite especificar el orden **en el eje z** de los elementos, estos es, el **orden de apilamiento en capas** del documento.

Los elementos se apilan en el orden en que aparecen: el elemento situado más abajo en el flujo normal del documento quedará encima.

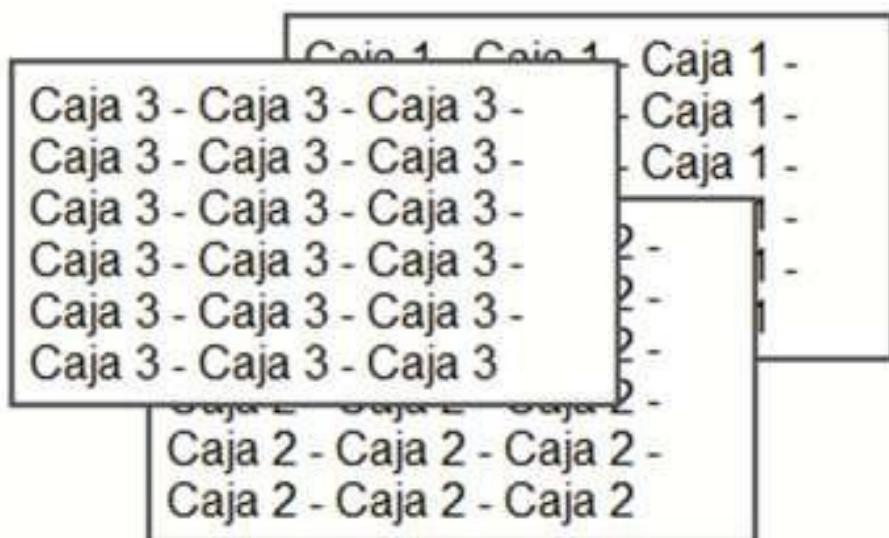
Si quieres que esta posición no sea tenida en cuenta, debes saber que los elementos con un **valor mayor de la propiedad z-index** son **colocados encima** de los que tienen un valor **menor z-index**, quedando estos últimos tapados por los primeros.

Esta **propiedad sólo se aplica a elementos** que tengan la **propiedad position** con **valores absolute o relative**.

# T4. FLOTAR Y POSICIONAR

## 2. Posicionamiento

### z-index:



```
div { position: absolute; }
#caja1 { z-index: 5; top: 1em; left: 8em; }
#caja2 { z-index: 15; top: 5em; left: 5em; }
#caja3 { z-index: 25; top: 2em; left: 2em; }
```

# T4. FLOTAR Y POSICIONAR

## float

Values: left | right | none | inherit

Default: none

Applies to: all elements

Inherits: no

## clear

Values: left | right | both | none | inherit

Default: none

Applies to: block-level elements only

Inherits: no

## position

Values: static | relative | absolute | fixed | inherit

Default: static

Applies to: all elements

Inherits: no

## top, right, bottom, left

Values: length measurement | percentage | auto | inherit

Default: auto

Applies to: Positioned elements (where position value is relative, absolute, or fixed)

Inherits: no

## z-index

Values: (number) | auto | inherit

Default: auto

Applies to: positioned elements

Inherits: no

## T4. FLOTAR Y POSICIONAR

### 3. Estrategias para el layout de las páginas

---

**Liquid pages:** ajustan su tamaño según la ventana del navegador.

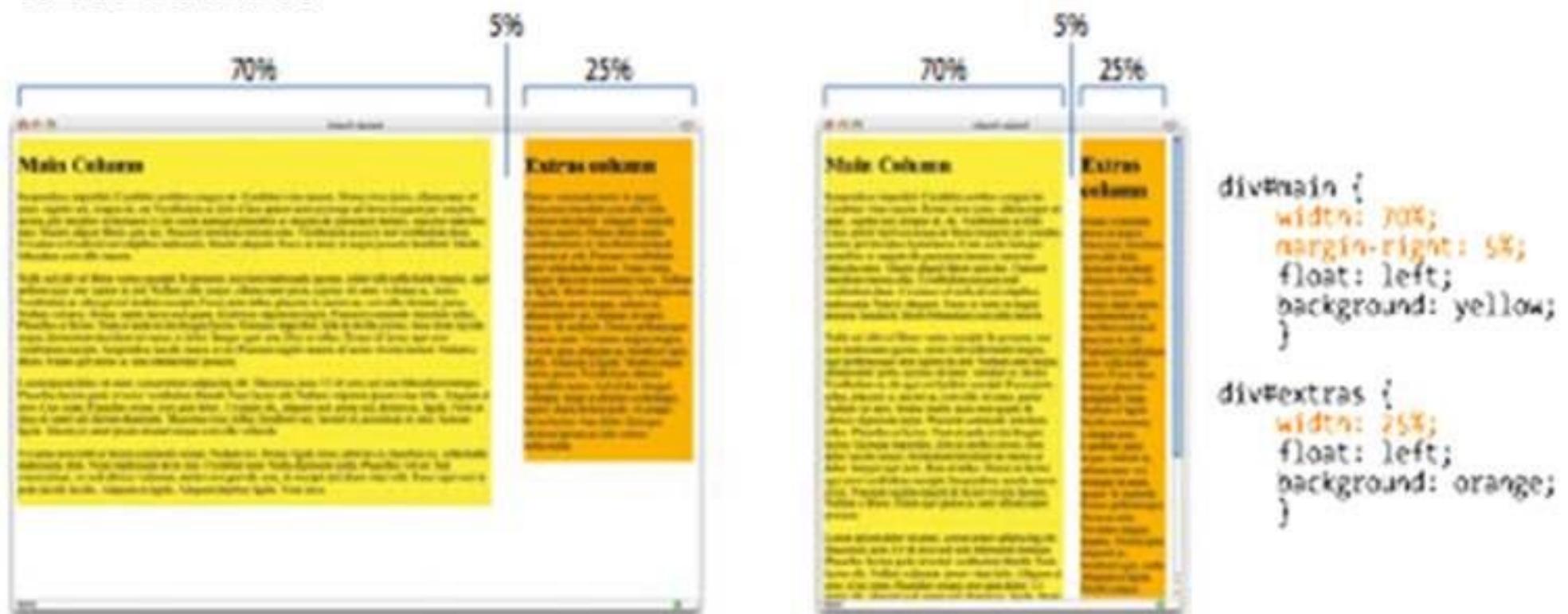
**Fixed pages:** ponen el contenido en un área de la página específica, independientemente de las dimensiones de la ventada del navegador.

**Elastic pages:** tienen áreas que se hacen más grande o más pequeño cuando se cambia el tamaño del texto.

# T4. FLOTAR Y POSICIONAR

## 3. Estrategias para el layout de las páginas

**Liquid pages:** ajustan su tamaño según la ventana del navegador.



En este diseño de dos columnas, la anchura de cada div se ha especificado como un porcentaje del ancho de página disponible. La columna principal siempre será 70% de la anchura de la ventana, y la columna derecha llena 25% (el 5% restante se utiliza para el margen entre las columnas), independientemente del tamaño de la ventana .

# T4. FLOTAR Y POSICIONAR

## 3. Estrategias para el layout de las páginas

**Liquid pages:** ajustan su tamaño según la ventana del navegador.



La columna secundaria de la izquierda se establece en un ancho de píxel específico, y de la principal área de contenido se establece en automático y llena el espacio restante en la ventana (que también podría haber dejado sin especificar para el mismo resultado). Aunque esta disposición utiliza un ancho fijo para una columna, todavía se considera líquido debido a que el ancho de la página se basa en la anchura de la ventana del navegador.

## T4. FLOTAR Y POSICIONAR

### 3. Estrategias para el layout de las páginas

---

**Fixed pages:** ponen el contenido en un área de la página específica, independientemente de las dimensiones de la ventada del navegador.

Este enfoque se basa en los **principios tradicionales de diseño gráfico**: una cuadrícula constante y la relación de los elementos de la página.

Al configurar su página a un ancho específico, tienes que decidir un par de cosas:

- un **ancho de página**, por lo general basado en resoluciones de monitor común.

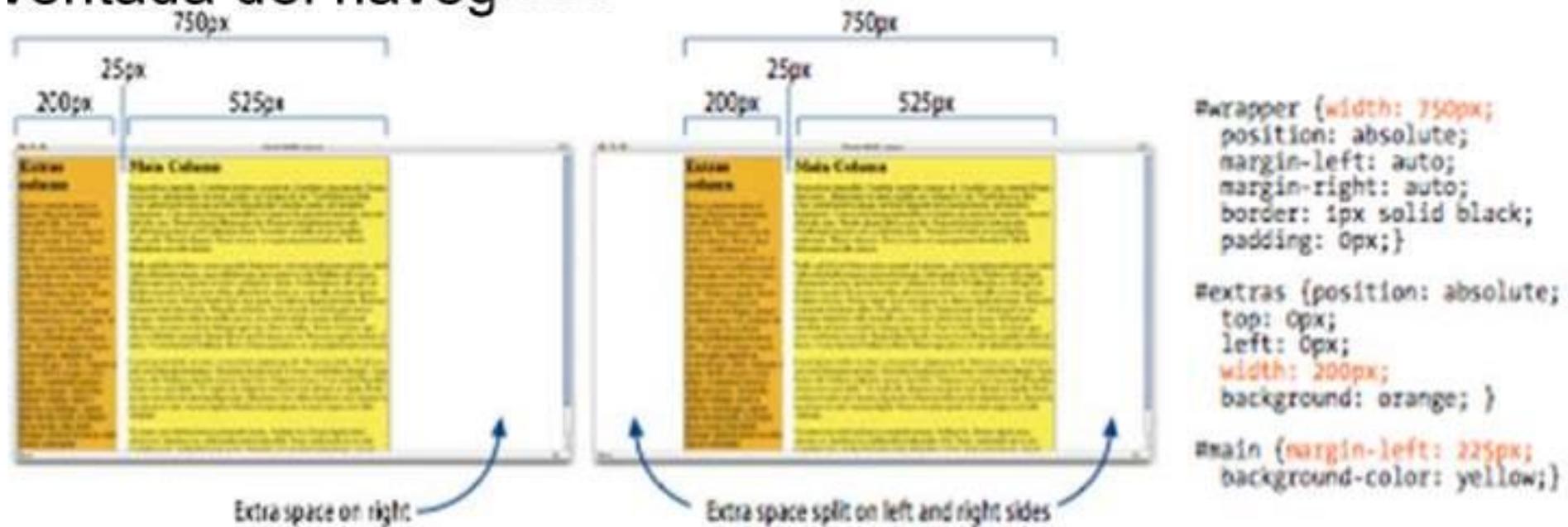
- La mayoría de las páginas web de ancho fijo como de este escrito están diseñados para encajar en una ventana del explorador de píxeles  $800 \times 600$ .

- dónde se debe colocar en la ventana del navegador.** Por defecto, se queda en el borde izquierdo del navegador, con el espacio adicional a la derecha de la misma. Algunos diseñadores optan por centrar la página, dividiendo el espacio extra sobre los márgenes izquierdo y derecho, lo que puede hacer la mirada la página como si se llena mejor la ventana del

# T4. FLOTAR Y POSICIONAR

## 3. Estrategias para el layout de las páginas

**Fixed pages:** ponen el contenido en un área de la página específica, independientemente de las dimensiones de la ventada del navegador.



Estos son dos diseños de ancho fijo. Ambos utilizan páginas de ancho fijo, pero la posición del contenido es diferente en la ventana del navegador .

## T4. FLOTAR Y POSICIONAR

### 3. Estrategias para el layout de las páginas

---

#### Fixed pages:

Una de las principales preocupaciones con el uso de diseños de ancho fijo es si la ventana del navegador del usuario no es tan amplia como la página, el contenido en el borde derecho de la página no será visible. Aunque es posible que desplazarse horizontalmente, no siempre puede ser evidente que hay más contenido que hay en el primer lugar.

Diseños de ancho fijo se crean mediante la especificación de valores de anchura en unidades de píxel.

Por lo general, el contenido de la página entera se pone en un **div** (a menudo llamado " contenido ", "container ", " envoltorio " o " página") que puede **ser ajustado a un ancho de píxel específico**. Este div también puede estar centrado en la ventana del navegador. Los anchos de elementos de **columna**, y los **márgenes de pares y relleno**, también se especifican en **píxeles**.

## T4. FLOTAR Y POSICIONAR

### 3. Estrategias para el layout de las páginas

---

**Elastic pages:** tienen áreas que se hacen más grande o más pequeño cuando se cambia el tamaño del texto.

Si el usuario hace el texto más grande, la caja que lo contiene se expande proporcionalmente.

Si el usuario hace el tamaño de texto muy pequeño , la caja que contiene encoge para caber.

El resultado es que las longitudes de línea (en términos de palabras o caracteres por línea) permanecen igual independientemente del tamaño del texto.

## T4. FLOTAR Y POSICIONAR

### 3. Estrategias para el layout de las páginas

**Elastic pages:** tienen áreas que se hacen más grande o más pequeño cuando se cambia el tamaño del texto.



La clave para los diseños elásticos es el em, una unidad de medida que se basa en el tamaño del texto.

# T4. FLOTAR Y POSICIONAR

## 4. Características avanzadas de CSS

---

CSS3 incorpora mayor control sobre el estilo de los elementos de nuestra página web. La principal desventaja es que actualmente solo algunos navegadores dan soporte. Algunas novedades:

- **Transiciones**
  
- **Transformaciones**
  - Las rotaciones
  - Escalar un elemento
  - Trasladar un elemento

# T4. FLOTAR Y POSICIONAR

## 4. Características avanzadas de CSS

---

### Transiciones

Las transiciones permiten realizar cambios en los valores CSS de forma progresiva, aunque la transición se puede indicar a cada una de las propiedades y se puede detallar, nosotros utilizaremos la forma abreviada, la aplicaremos a todas las propiedades hay una lista de elementos sobre los que se puede crear la transición, para ello basta con indicar

# T4. FLOTAR Y POSICIONAR

## 4. Características avanzadas de CSS

### LAS TRANSICIONES DE CSS3

Para activar una transición es necesario que se detecte un evento. Por ejemplo con :hover, :active o :focus. Si combinamos las transiciones con las transformaciones de CSS3, daremos un efecto de movimiento.

Veamos un ejemplo:

```
01 #nav a{  
02   background-color:red;  
03 }  
04  
05 #nav a:hover,  
06 #nav a:focus{  
07   background-color:blue;  
08  
09   -webkit-transition-property:background-color;  
10   -moz-transition-property:background-color;  
11   -o-transition-property:background-color;  
12   -ms-transition-property:background-color;  
13   transition-property:background-color;  
14  
15   -webkit-transition-duration:2s;  
16   -moz-transition-duration:2s;  
17   -o-transition-duration:2s;  
18   -ms-transition-duration:2s;  
19   transition-duration:2s;  
20  
21 }
```

En este ejemplo estamos indicando que la propiedad de transición a variar es el color del background y que se tome 2 segundos en hacer progresivamente el cambio de color. Las transiciones de CSS3 pueden tomar estos valores:  
transition-property  
transition-duration  
transition-delay  
transition-timing-function

# T4. FLOTAR Y POSICIONAR

## 4. Características avanzadas de CSS

### Transiciones

- background-color
- background-image
- background-position
- border-bottom-color
- border-bottom-width
- border-color
- border-left-color
- border-left-width
- border-right-color
- border-right-width
- border-spacing
- border-top-color
- border-top-width
- border-width
- bottom
- color
- crop
- font-size
- font-weight

### LISTA DE VALORES QUE PUEDE TOMAR TRANSITION-PROPERTY

- grid-\*
- height
- left
- letter-spacing
- line-height
- margin-bottom
- margin-left
- margin-right
- margin-top
- max-height
- max-width
- min-height
- min-width
- opacity
- outline-color
- outline-offset
- outline-width
- padding-bottom
- padding-left
- outline-width
- padding-bottom
- padding-left
- padding-right
- padding-top
- right
- text-indent
- text-shadow
- top
- vertical-align
- visibility
- width
- word-spacing
- z-index
- zoom

# T4. FLOTAR Y POSICIONAR

## 4. Características avanzadas de CSS

### Transiciones

#### Duración de la transición

Con `transition-duration` especificamos el tiempo que va a tardar la transición. Este tiempo se puede expresar tanto en segundos como en milisegundos. Así, 2.4s y 2400ms tendrán una misma duración.

Imaginemos que tenemos un elemento con el id `#desplazamiento` que queremos mover en la pantalla con las transiciones de CSS3. El modo en que lo haríamos sería el siguiente:

```
01 #desplazamiento a {  
02   position: absolute;  
03   left: 0;  
04  
05   -webkit-transition-property: left;  
06   -moz-transition-property: left;  
07   -o-transition-property: left;  
08   -ms-transition-property: left;  
09   transition-property: left;  
10  
11  -webkit-transition-duration: 2s;  
12  -moz-transition-duration: 2s;  
13  -o-transition-duration: 2s;  
14  -ms-transition-duration: 2s;  
15  transition-duration: 2s;  
16 }  
17  
18 #desplazamiento a:hover {  
19   left:15px;  
20 }
```

Aquí lo que sucede es que el enlace que contenga el ID `desplazamiento` se desplazará 15px cuando el usuario pase el ratón por encima. Fíjate que en donde especificamos el valor para el evento, en este caso `:hover`, sólo decimos el cambio que se hará y el resto de propiedades se especifican en la etiqueta normal. Date cuenta que hemos de especificar dos cosas, el `transition-property` o la propiedad de transformación y la duración de la misma con `transition-duration`.

# T4. FLOTAR Y POSICIONAR

## 4. Características avanzadas de CSS

---

### Transiciones

#### RETRASO EN LA TRANSICIÓN DE CSS3

Con la propiedad transition-delay podemos indicar un tiempo de espera antes de empezar a realizar la transición. Esta propiedad puede tomar tanto valores positivos como negativos. En el caso que le demos un valor negativo, lo que sucede es que la transición tiene lugar en el momento de accionar el evento, lo que empieza en el punto que habría comenzado en caso de haberlo hecho en el tiempo indicado. Así, si indico un retraso de -3s y tengo una animación de 10s, en el momento de empezar no lo hará en el segundo 0, sino en el segundo 3.

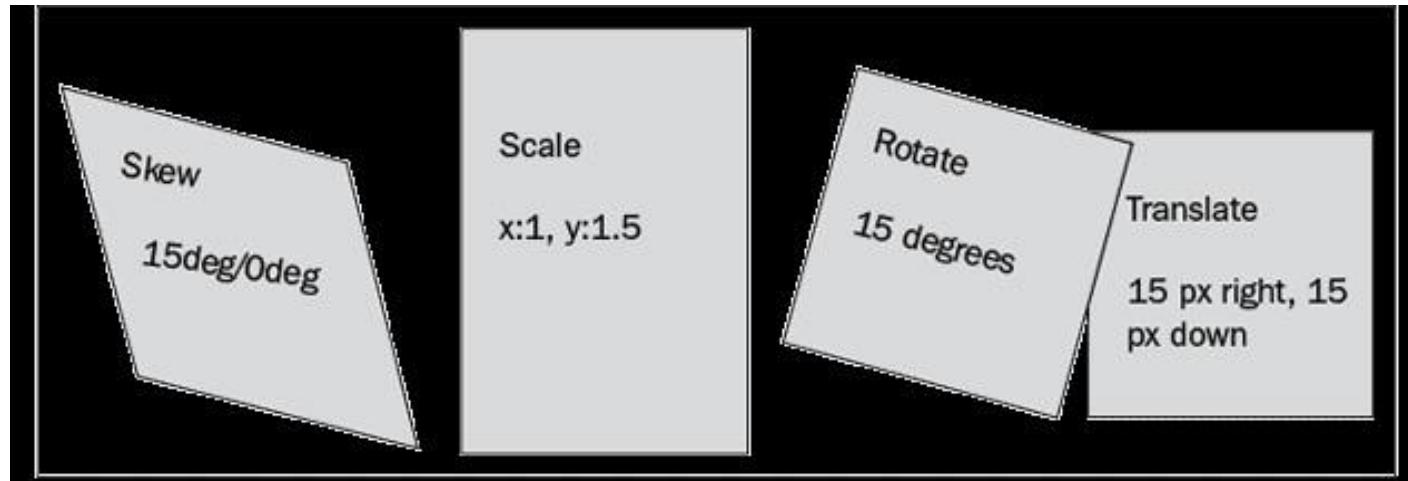
En el caso de IE y Opera no aceptan valores comprendidos entre los -10ms y los 10ms.

# T4. FLOTAR Y POSICIONAR

## 4. Características avanzadas de CSS

### Transformaciones

Nos permiten girar, desplazar, aumentar e inclinar.



De las transformaciones de CSS3 en 2D, las más usadas son:

- Rotate.** Rotate te permite rotar un elemento dándole un ángulo de giro en grados.
- Scale.** Scale te permite escalar un elemento, toma valores positivos y negativos y se le pueden poner decimales.
- Translate.** Translate nos permite trasladar un elemento a la vez en el eje de las X y de las Y, dándole las coordenadas iniciales y finales.

Si no queremos que se aplique ninguna transformación, la propiedad será de none

```
.ejemplo { transform:none; }
```

```
}
```

# T4. FLOTAR Y POSICIONAR

## 4. Características avanzadas de CSS

### Transformaciones

#### LAS ROTACIONES DE CSS3

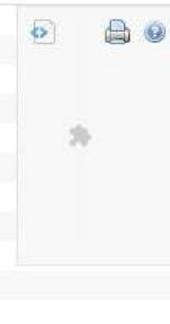
Como hemos visto, la propiedad de transformación de CSS3 tiene muchas aplicaciones, una de ellas la de rotar un elemento. Se puede aplicar tanto a elemento inline como a elementos de bloque.

```
1 .ejemplo {  
2   -webkit-transform: rotate (45deg);  
3   -moz-transform: rotate (45deg);  
4   -o-transform: rotate (45deg);  
5   -ms-transform: rotate (45deg);  
6   transform: rotate (45deg);  
7 }
```

Fíjate que los grados se marcan en positivo si el elemento se rota en el sentido de las agujas del reloj y en negativo si es al revés.

Por defecto, el punto de referencia que toma como eje para hacer la rotación es el centro, pero también se puede especificar otro punto. Aquí un ejemplo en que se traslada el eje de rotación a la parte superior izquierda del elemento:

```
01 -webkit-transform-origin: left top;  
02 -moz-transform-origin: left top;  
03 -o-transform-origin: left top;  
04 -ms-transform-origin: left top;  
05 transform-origin: left top;  
06  
07 -webkit-transform: rotate (-20deg);  
08 -moz-transform: rotate (-20deg);  
09 -o-transform: rotate (-20deg);  
10 -ms-transform: rotate (-20deg);  
11 transform: rotate (-20deg);
```

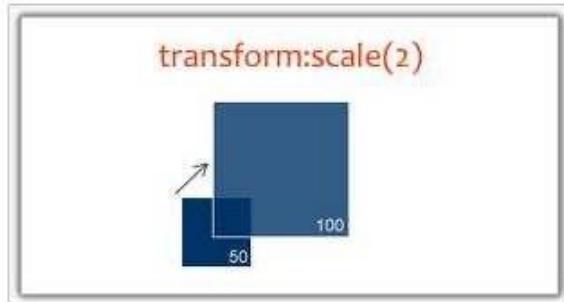


# T4. FLOTAR Y POSICIONAR

## 4. Características avanzadas de CSS

### Transformaciones

#### Escalar un elemento



Si lo deseamos, podemos aplicar el cambio de escala sólo en una dirección. Utilizamos `scaleX` para sólo escalar en el eje de las X y `scaleY` para sólo escalar en el eje de las Y:

```

1 -webkit-transform: scaleY(3);
2 -moz-transform: scaleY(3);
3 -o-transform: scaleY(3);
4 -ms-transform: scaleY(3);
5 transform: scaleY(3);

```

Si el valor es positivo, el elemento se hace más grande, y si es negativo se hace más pequeño.

En el siguiente ejemplo, se escalará el elemento multiplicando por dos:

```

1 -webkit-transform: scale(2);
2 -moz-transform: scale(2);
3 -o-transform: scale(2);
4 -ms-transform: scale(2);
5 transform: scale(2);

```

También es posible que queramos dar dos valores diferentes según las X y las Y, con lo que creamos un efecto de escala no proporcional que nos da un efecto visual de distorsión.

```

1 -webkit-transform: scale (.5, 2);
2 -moz-transform: scale (.5, 2);
3 -o-transform: scale (.5, 2);
4 -ms-transform: scale (.5, 2);
5 transform: scale (.5, 2);

```



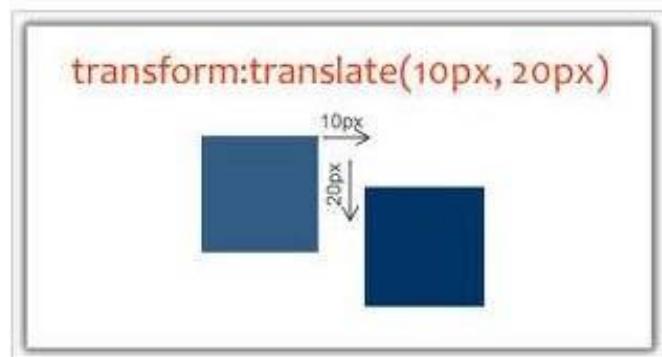
# T4. FLOTAR Y POSICIONAR

## 4. Características avanzadas de CSS

### Transformaciones

#### TRASLADAR ELEMENTOS CON CSS3

Este nombre puede llevar a confusión, ya que esta propiedad de CSS3 no hace propiamente una transición, sino que hace que un elemento pase de estar en una posición a estar en otra. Con las transiciones y las animaciones de CSS3 vamos a poder darle un efecto visual de movimiento.



```
1 -webkit-transform: translate(10px, 20px);
2 -moz-transform: translate(10px, 20px);
3 -o-transform: translate(10px, 20px);
4 -ms-transform: translate(10px, 20px);
5 transform: translate(10px, 20px);
```

# Diseño de Interfaces Web

---

Bloque 2: Hojas de estilo CSS

**Tema 4: Flotar y posicionar**

