

HOJA DE ESTILOS: CSS

Capítulo 5

Lenguajes de Marcas y Sistemas de Gestión de Información

Curso 2019/2020



Capítulo 5: POSICIONAMIENTO Y VISUALIZACIÓN

1. Introducción
2. Tipos de Elementos
3. Posicionamiento
4. Posicionamiento Normal
5. Posicionamiento Relativo
6. Posicionamiento Absoluto
7. Posicionamiento Fijo
8. Posicionamiento Flotante
9. Visualización



1. INTRODUCCIÓN

Para cumplir con el modelo de cajas presentado en el capítulo anterior, los navegadores crean una caja para representar a cada elemento de la página HTML. Los factores que se tienen en cuenta para generar cada caja son:

- Las **propiedades** `width` y `height` de la caja (ya establecidas).
- El **tipo** de elemento HTML (elemento de bloque o elemento en línea).
- **Posicionamiento** de la caja (normal, relativo, absoluto, fijo o flotante).
- Las **relaciones** entre elementos (dónde se encuentra cada elemento, elementos descendientes, etc.)
- Otro tipo de información, como por ejemplo el tamaño de las **imágenes** y el tamaño de la **ventana** del navegador.



2. TIPOS DE ELEMENTOS

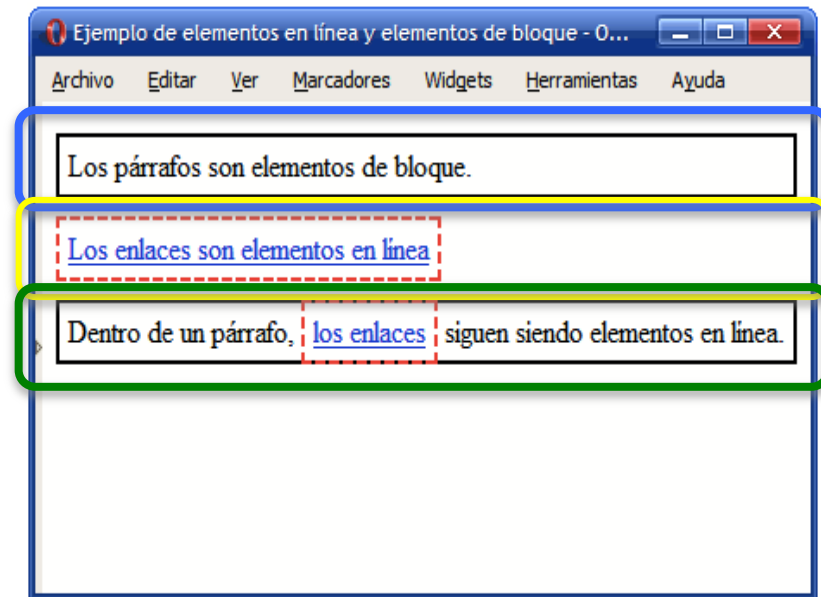
El estándar HTML clasifica a todos sus elementos en dos grandes grupos:

- **Elementos de Bloque** (*block elements*): Siempre empiezan en una nueva línea y ocupan todo el espacio disponible hasta el final de la línea.
- **Elementos en Línea** (*inline elements*): No empiezan necesariamente en nueva línea y sólo ocupan el espacio necesario para mostrar sus contenidos.



2. TIPOS DE ELEMENTOS

- El **primer elemento** es un párrafo. Los párrafos son elementos de bloque y su caja empieza en una nueva línea y llega hasta el final de esa misma línea. El navegador reserva todo el espacio disponible en la primera línea.
- El **segundo elemento** es un enlace. Los enlaces son elementos en línea, por lo que su caja sólo ocupa el espacio necesario para mostrar sus contenidos.
- El **tercer elemento** es un párrafo que se comporta de la misma forma que el primer párrafo. En su interior se encuentra un enlace que también se comporta de la misma forma que el enlace anterior. Así, el segundo párrafo ocupa toda una línea y el segundo enlace sólo ocupa el espacio necesario para mostrar sus contenidos.





2. TIPOS DE ELEMENTOS

Elementos en línea:

a, abbr, acronym, b, basefont, bdo, big, br, cite, code, dfn, em, font, i, img, input, kbd, label, q, s, samp, select, small, span, strike, strong, sub, sup, textarea, tt, u, var.

Elementos de bloque:

address, blockquote, center, dir, div, dl, fieldset, form, h1, h2, h3, h4, h5, h6, hr, isindex, menu, noframes, noscript, ol, p, pre, table, ul.

Otros elementos que se consideran que son de bloque:

dd, dt, frameset, li, tbody, td, tfoot, th, thead, tr.

Los siguientes elementos pueden ser en línea y de bloque según las circunstancias:

button, del, iframe, ins, map, object, script.

+

3. POSICIONAMIENTO

Los navegadores **crean y posicionan** de forma automática todas las cajas que forman cada página HTML. CSS permite al diseñador **modificar la posición** en la que se muestra cada caja. El estándar de CSS define cinco modelos diferentes para posicionar una caja:

- **Posicionamiento Normal o Estático.**
- **Posicionamiento Relativo.**
- **Posicionamiento Absoluto.**
- **Posicionamiento Fijo.**
- **Posicionamiento Flotante.**

+ 3. POSICIONAMIENTO

El posicionamiento de una caja se establece mediante la propiedad `position`:

| | |
|----------------------|--|
| Propiedad | <i>position</i> |
| Valores | static relative absolute fixed inherit |
| Se aplica a | Todos los elementos |
| Valor Inicial | static |
| Descripción | Selecciona el posicionamiento con el que se mostrará el elemento |

3. POSICIONAMIENTO

- **Static:** posicionamiento normal o estático. Si se utiliza este valor, se ignoran los valores de las propiedades `top`, `right`, `bottom` y `left` que se verán a continuación.
- **Relative:** posicionamiento relativo. El desplazamiento de la caja se controla con las propiedades `top`, `right`, `bottom` y `left`.
- **Absolute:** posicionamiento absoluto. El desplazamiento de la caja se controla con mismas las propiedades, pero su interpretación es más compleja; el origen de coordenadas del desplazamiento depende del posicionamiento de su elemento contenedor.
- **Fixed:** posicionamiento fijo. El desplazamiento se establece de la misma forma que en el posicionamiento absoluto, pero en este caso el elemento permanece inamovible en la pantalla.

3. POSICIONAMIENTO

Cuando se posiciona una caja también es necesario **desplazarla** respecto de su posición original o respecto de otro origen de coordenadas. CSS define cuatro propiedades llamadas `top`, `right`, `bottom` y `left` para controlar el desplazamiento de las cajas posicionadas:

| | |
|----------------------|---|
| Propiedad | <i>top, right, bottom, left</i> |
| Valores | unidad de medida porcentaje auto inherit |
| Se aplica a | Todos los elementos posicionados |
| Valor Inicial | auto |
| Descripción | Indican el desplazamiento horizontal y vertical del elemento respecto de su posición original |



4. POSICIONAMIENTO NORMAL

Es el modelo que utilizan por defecto los navegadores para mostrar los elementos de las páginas. Sólo se tiene en cuenta si el **elemento es de bloque o en línea**, sus propiedades `width` y `height` y su contenido.

Los elementos de bloque forman “*contextos de formato de bloque*”: las cajas se muestran **una debajo de otra** comenzando desde el principio del elemento contenedor.

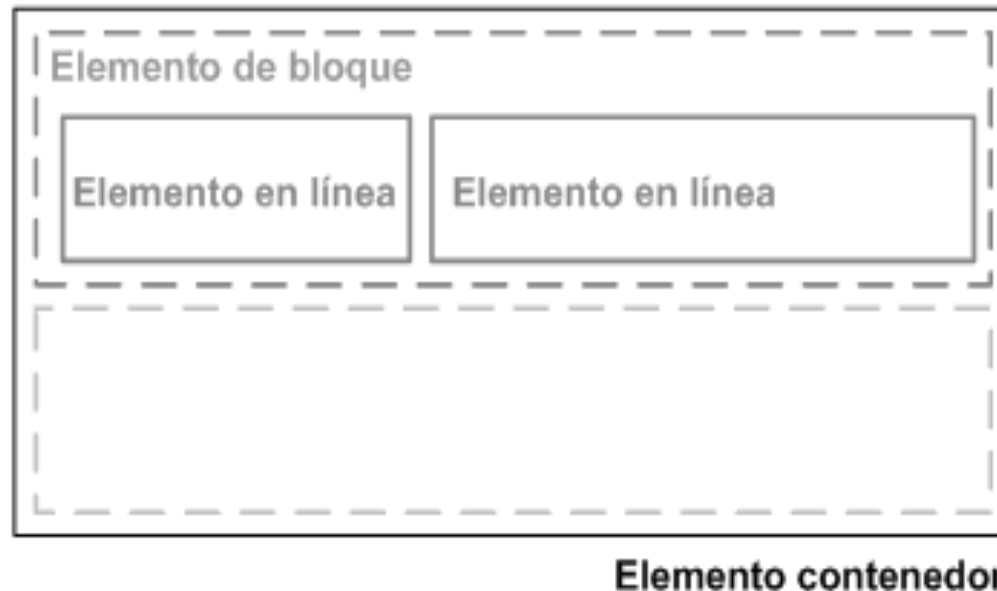


Si un elemento se encuentra dentro de otro, el elemento padre se llama “*elemento contenedor*” y determina tanto la posición como el tamaño de todas sus cajas interiores.



4. POSICIONAMIENTO NORMAL

Los elementos en línea forman los “*contextos de formato en línea*”: las cajas se muestran una detrás de otra **de forma horizontal** comenzando desde la posición más a la izquierda de su elemento contenedor.



5. POSICIONAMIENTO RELATIVO

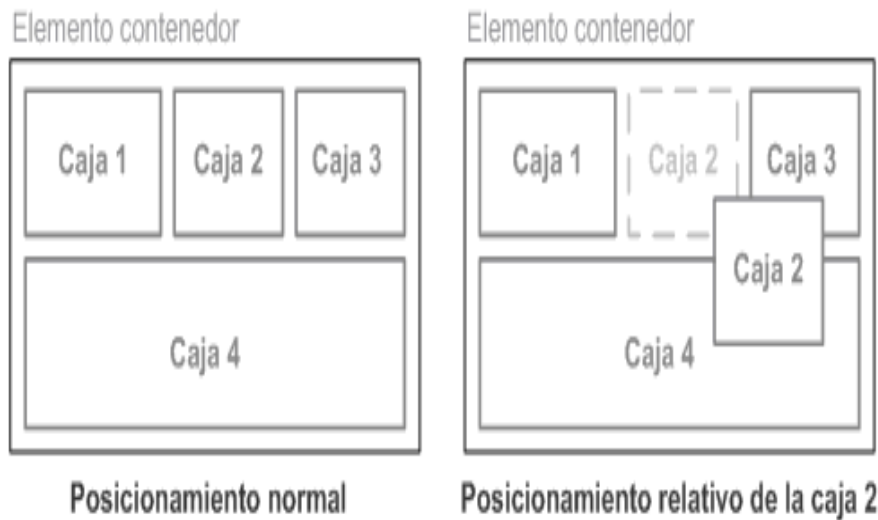
CSS considera que el posicionamiento relativo es un caso particular del posicionamiento normal, aunque en la práctica presenta muchas diferencias.

El posicionamiento relativo desplaza una caja **respecto de su posición original** establecida mediante el posicionamiento normal. El desplazamiento de la caja se controla con las propiedades `top`, `right`, `bottom` y `left`.

`Top` se emplea para mover las cajas de forma descendente, `bottom` mueve las cajas de forma ascendente, `left` para desplazar las cajas hacia la derecha y `right` mueve las cajas hacia la izquierda.

5. POSICIONAMIENTO RELATIVO

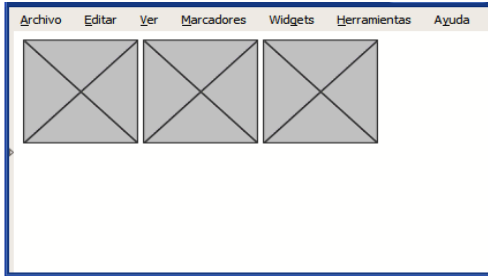
El desplazamiento relativo de una caja **no afecta** al resto de cajas adyacentes, que se muestran en la misma posición que si la caja desplazada no se hubiera movido de su posición original.



La **Caja 2** se ha desplazado lateralmente hacia la derecha y verticalmente de forma descendente. El resto de cajas de la página no modifican su posición, se producen solapamientos entre los contenidos de las cajas.



5. POSICIONAMIENTO RELATIVO



Aplicando el posicionamiento relativo, se desplaza la primera imagen de forma descendente:

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

```

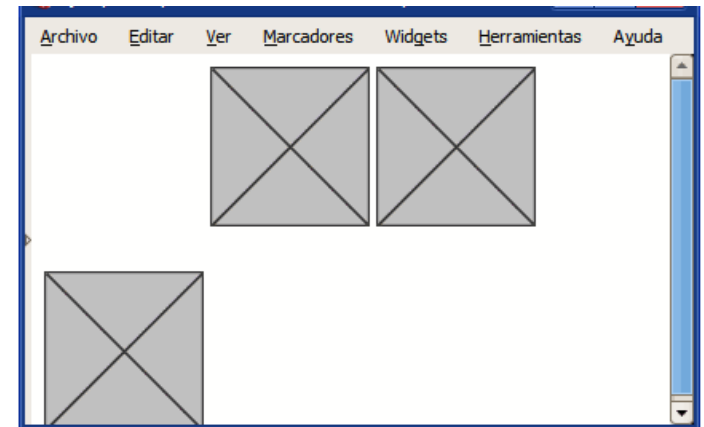
```

```

```

```

```

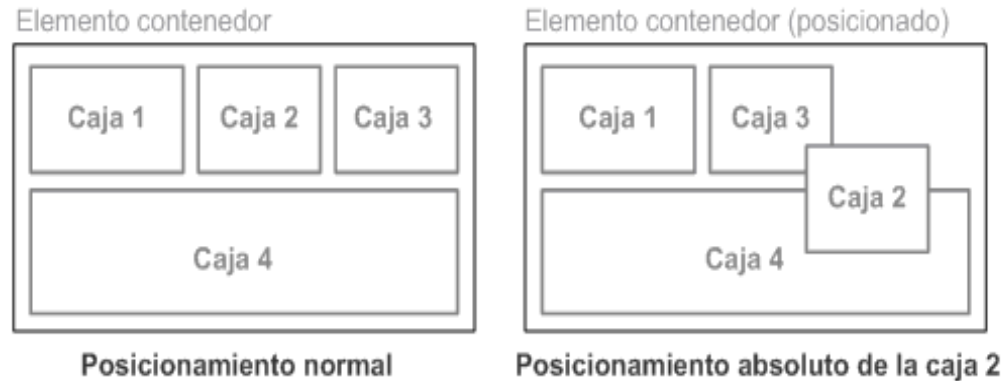


6. POSICIONAMIENTO ABSOLUTO

Se emplea para establecer de **forma exacta** la posición en la que se muestra la caja de un elemento. La nueva posición de la caja se indica mediante las propiedades `top`, `right`, `bottom` y `left`. Los valores de estas propiedades dependen del posicionamiento del elemento contenedor.

Cuando una caja se posiciona de forma absoluta, el resto de elementos de la página **se ven afectados y modifican su posición**. Al igual que en el posicionamiento relativo, cuando se posiciona de forma absoluta una caja es probable que se produzcan **solapamientos** con otras cajas.

6. POSICIONAMIENTO ABSOLUTO



La Caja 2 está posicionada de forma absoluta, lo que provoca que el resto de elementos de la página **modifiquen** su posición → La Caja 3 deja su lugar original y pasa a ocupar el hueco dejado por la Caja 2.

El estándar de CSS 2.1 indica que las cajas posicionadas de forma absoluta *“salen del flujo normal de la página”*, lo que provoca que el resto de elementos de la página se muevan y en ocasiones, ocupen la posición original en la que se encontraba la caja.

6. POSICIONAMIENTO ABSOLUTO

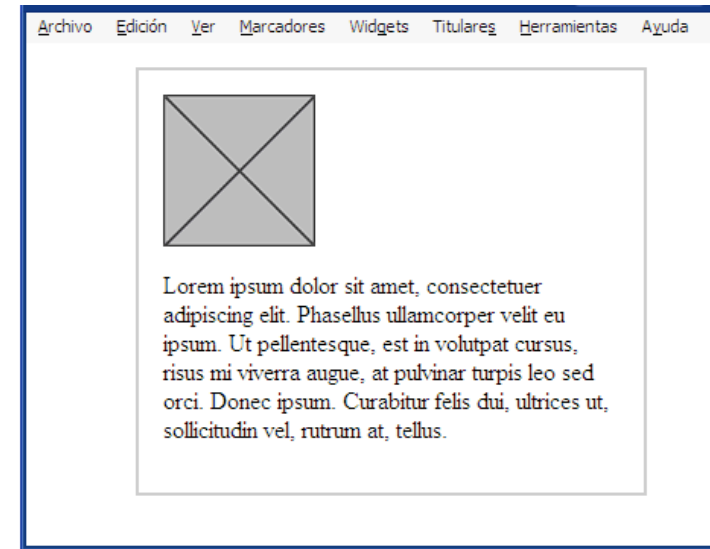
Para interpretar los valores de `top`, `right`, `bottom` y `left` de una caja posicionada de forma absoluta es un proceso complejo:

1. Se buscan todos los elementos contenedores de la caja hasta llegar al elemento `<body>` de la página.
2. Se recorren todos los elementos contenedores empezando por el más cercano a la caja y llegando hasta el `<body>`
3. El primer elemento contenedor que esté posicionado de cualquier forma diferente a `position: static` se convierte en la referencia que determina la posición de la caja posicionada de forma absoluta.
4. Si ningún elemento contenedor está posicionado, la referencia es la ventana del navegador, que no debe confundirse con el elemento `<body>` de la página.

6. POSICIONAMIENTO ABSOLUTO

A continuación, se muestra el código HTML y CSS:

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



<div>

```

```

```
<p>Lorem ipsum dolor sit amet ... vel, rutrum at, tellus.</p>
```

</div>

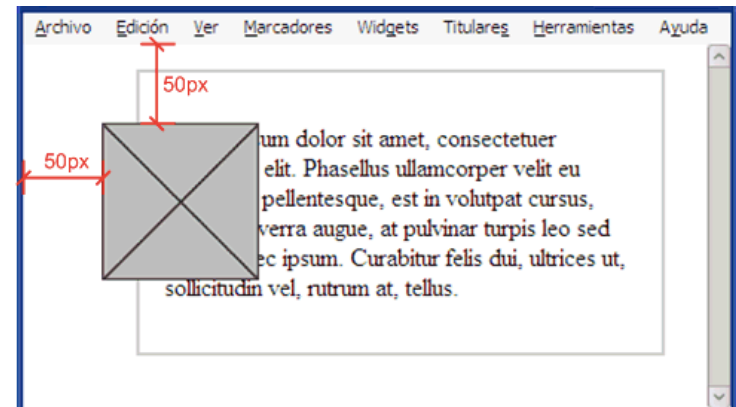
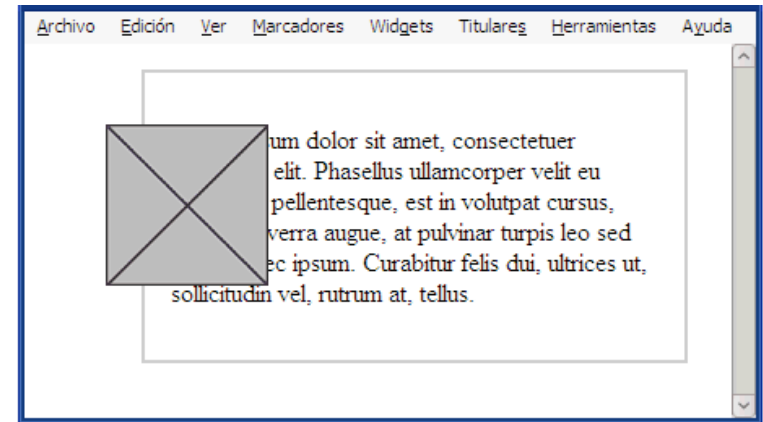


6. POSICIONAMIENTO ABSOLUTO

Se posiciona de forma absoluta la imagen mediante la propiedad `position` y se indica su nueva posición mediante las propiedades `top` y `left`:

```
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:





7. POSICIONAMIENTO FIJO

Es un **caso particular** del posicionamiento absoluto, ya que sólo se diferencian en el comportamiento de las cajas posicionadas.

La forma de obtener el origen de coordenadas para interpretar su desplazamiento es idéntica al posicionamiento absoluto.

La principal característica de una caja posicionada de forma fija es que su posición es **inamovible** dentro de la ventana del navegador. El posicionamiento fijo hace que las cajas no modifiquen su posición ni aunque el usuario suba o baje la página en la ventana de su navegador.

8. POSICIONAMIENTO FLOTANTE

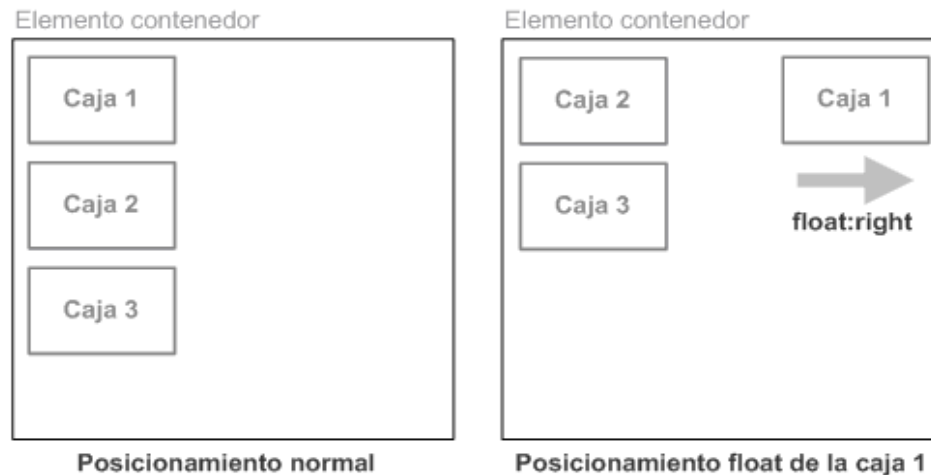
Es el más difícil de comprender pero al mismo tiempo es el más utilizado. La mayoría de estructuras de las páginas web complejas están diseñadas con el posicionamiento flotante.

Cuando una caja se posiciona con el modelo de posicionamiento flotante, automáticamente se convierte en una **caja flotante**, lo que significa que se **desplaza** hasta la zona más a la izquierda o más a la derecha de la posición en la que originalmente se encontraba.



8. POSICIONAMIENTO FLOTANTE

La imagen muestra el resultado de posicionar de forma flotante hacia la derecha la Caja 1:

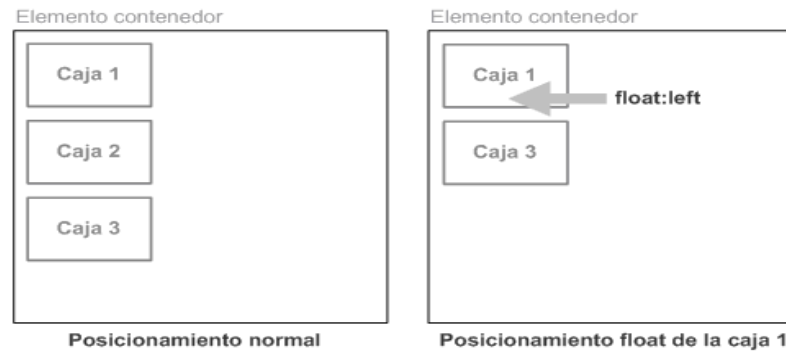


Cuando se posiciona una caja de forma flotante: ***La caja deja de pertenecer al flujo normal de la página, lo que significa que el resto de cajas ocupan el lugar dejado por la caja flotante.*** La caja flotante se posiciona lo más a la izquierda o lo más a la derecha posible de la posición en la que se encontraba originalmente.



8. POSICIONAMIENTO FLOTANTE

Si la Caja 1 se posiciona de forma flotante hacia la izquierda, el resultado es el que muestra la siguiente imagen:



La Caja 1 es de **tipo flotante**, por lo que desaparece del flujo normal de la página y el resto de cajas ocupan su lugar. El resultado es que la Caja 2 ahora se muestra donde estaba la Caja 1 y la Caja 3 se muestra donde estaba la Caja 2.

Al mismo tiempo, la Caja 1 se desplaza todo lo posible hacia la izquierda de la posición en la que se encontraba. El resultado es que la Caja 1 se muestra encima de la nueva posición de la Caja 2 y tapa todos sus contenidos.



8. POSICIONAMIENTO FLOTANTE

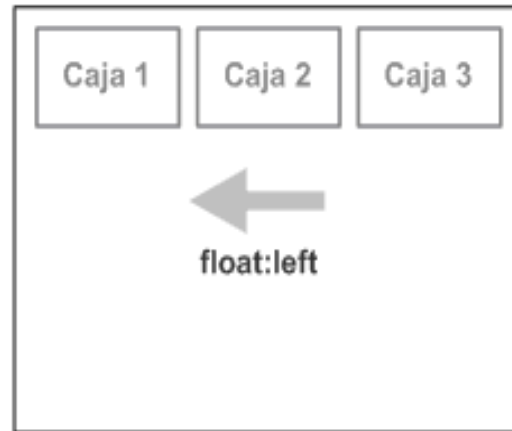
Si existen **otras cajas flotantes**, al posicionar de forma flotante otra caja, se tiene en cuenta el sitio disponible. En el siguiente ejemplo se posicionan de forma flotante hacia la izquierda las tres cajas:

Elemento contenedor



Posicionamiento normal

Elemento contenedor



Posicionamiento float de las 3 cajas

8. POSICIONAMIENTO FLOTANTE

La propiedad CSS que permite posicionar de forma flotante una caja se denomina `float`:

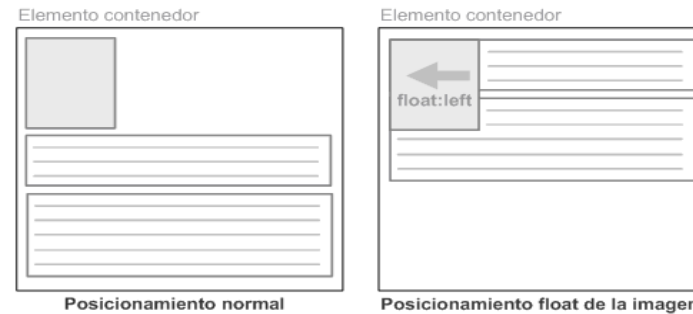
| | |
|----------------------|--|
| Propiedad | <i>float</i> |
| Valores | left right none inherit |
| Se aplica a | Todos los elementos |
| Valor Inicial | none |
| Descripción | Establece el tipo de posicionamiento flotante del elemento |



8. POSICIONAMIENTO FLOTANTE

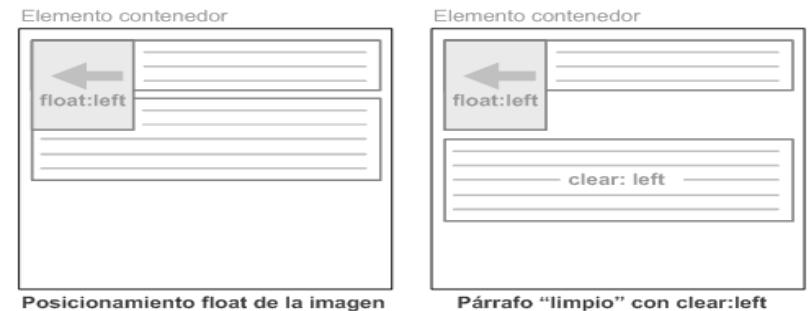
Los elementos que se encuentran alrededor de una caja flotante **adaptan** sus contenidos para que **fluyan** alrededor del elemento posicionado:

```
img {  
    float: left;  
}
```



En muchas ocasiones es admisible que algunos contenidos fluyan alrededor de una imagen, pero el resto de contenidos **deben mostrarse en su totalidad** sin fluir alrededor de la imagen:

```
<p style="clear: left;">...</p>
```





8. POSICIONAMIENTO FLOTANTE

Propiedad `clear`:

| | |
|----------------------|--|
| Propiedad | <i>clear</i> |
| Valores | none left right both inherit |
| Se aplica a | Todos los elementos de bloque |
| Valor Inicial | none |
| Descripción | Indica lado del elemento que no debe ser adyacente a ninguna caja flotante |

9. VISUALIZACIÓN

Además de las propiedades que controlan el posicionamiento de los elementos, CSS define otras cuatro propiedades para controlar su **visualización**:

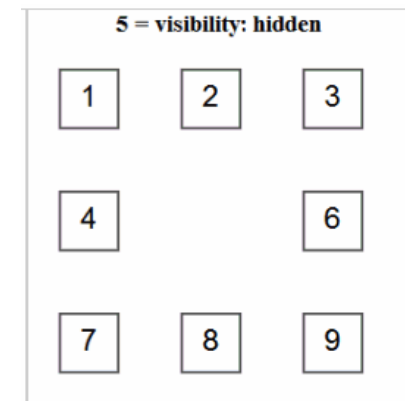
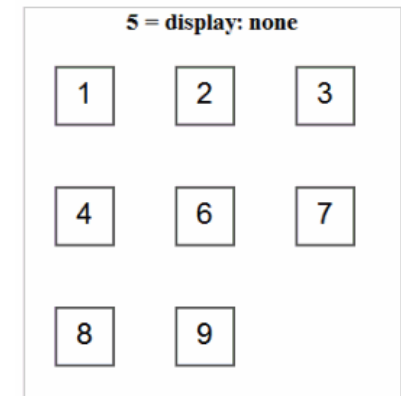
- `display`
- `visibility`
- `overflow`
- `z-index`

Utilizando algunas de estas propiedades es posible ocultar y/o hacer invisibles las cajas de los elementos, por lo que son imprescindibles para realizar **efectos avanzados y animaciones**.



9.1. PROPIEDADES *DISPLAY* Y *VISIBILITY*

- `display` permite ocultar completamente un elemento haciendo que desaparezca de la página. Como el elemento oculto no se muestra, el resto de elementos de la página se mueven para ocupar su lugar.
- `visibility` permite hacer invisible un elemento, lo que significa que el navegador crea la caja del elemento pero no la muestra. El resto de elementos de la página no modifican su posición, ya que aunque la caja no se ve, sigue ocupando sitio.



9.1. PROPIEDADES *DISPLAY* Y *VISIBILITY*

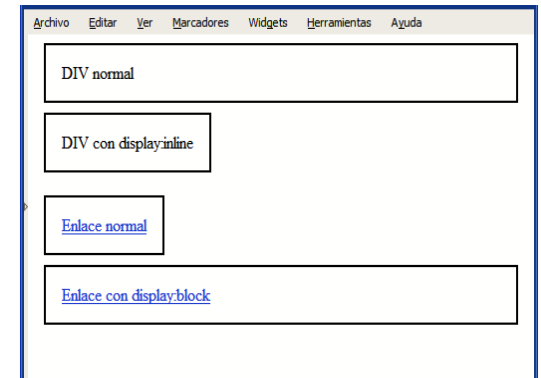
Propiedad `display`:

| Propiedad | <i>display</i> |
|---------------|---|
| Valores | inline block none list-item run-in inline-block table inline-table table-row-group table-header-group table-footer-group table-row table-column-group table-column table-cell table-caption inherit |
| Se aplica a | Todos los elementos |
| Valor Inicial | inline |
| Descripción | Permite controlar la forma de visualizar un elemento e incluso ocultarlo |



9.1. PROPIEDADES *DISPLAY* Y *VISIBILITY*

El siguiente ejemplo muestra el uso de la propiedad `display` para mostrar un elemento de bloque como si fuera un elemento en línea y para mostrar un elemento en línea como si fuera un elemento de bloque:



```
<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>
```




9.1. PROPIEDADES *DISPLAY* Y *VISIBILITY*

Propiedad `visibility`:

| | |
|----------------------|---|
| Propiedad | <i>visibility</i> |
| Valores | visible hidden collapse inherit |
| Se aplica a | Todos los elementos |
| Valor Inicial | visible |
| Descripción | Permite hacer visibles e invisibles a los elementos |



9.2. RELACIÓN ENTRE *DISPLAY*, *FLOAT* Y *POSITION*

Cuando se establecen las propiedades `display`, `float` y `position` sobre una misma caja, su interpretación es la siguiente:

1. Si `display : none`, se ignoran las propiedades `float` y `position` y la caja **no se muestra** en la página.
2. Si `position : absolute` o `fixed`, la caja se posiciona de forma absoluta, se considera que `float : none` y la propiedad `display : block` tanto para los elementos en línea como para los elementos de bloque.
3. En cualquier otro caso, si `float` tiene un valor distinto de `none`, la caja se posiciona de forma flotante y la propiedad `display : block` tanto para los elementos en línea como para los elementos de bloque.



9.3. PROPIEDADES *OVERFLOW*

Propiedad `overflow`:

| | |
|----------------------|---|
| Propiedad | <i>overflow</i> |
| Valores | visible hidden scroll auto inherit |
| Se aplica a | Elementos de bloque y celdas de tablas |
| Valor Inicial | visible |
| Descripción | Permite controlar los contenidos sobrantes de un elemento |

9.3. PROPIEDADES *OVERFLOW*

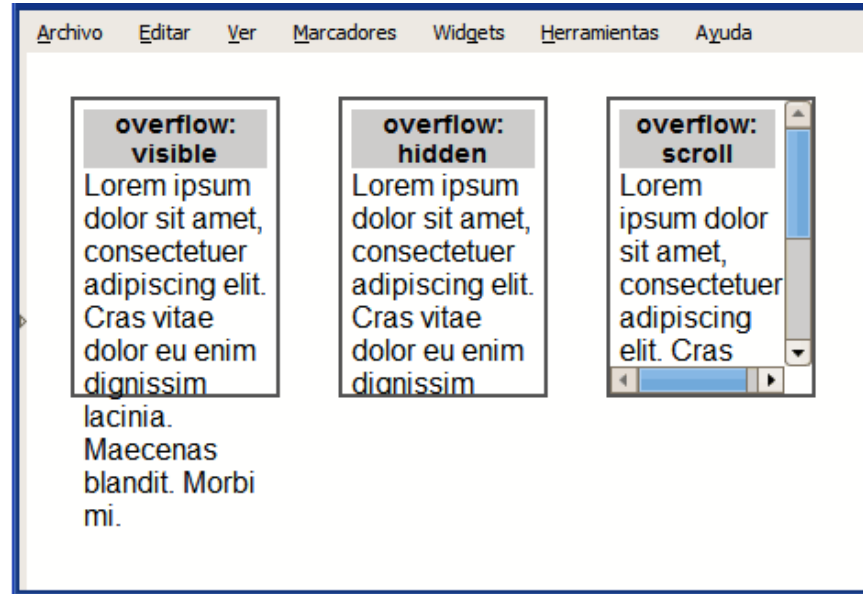
Los valores de la propiedad `overflow` tienen el siguiente significado:

- **visible**: el contenido no se corta y se muestra **sobresaliendo** la zona reservada para visualizar el elemento.
- **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`.



9.3. PROPIEDADES *OVERFLOW*

```
div {
  display: inline;
  float: left;
  margin: 1em;
  padding: .3em;
  border: 2px solid #555;
  width: 100px;
  height: 150px;
  font: 1em Arial;
}
```



```
<div><h1>overflow:  visible</h1> Lorem ipsum dolor sit amet, consectetur ...
Morbi mi.</div>
```

```
<div  style="overflow:hidden"><h1>overflow:  hidden</h1> Lorem ipsum dolor ...
Maecenas blandit. Morbi mi.</div>
```

```
<div  style="overflow:scroll"><h1>overflow:  scroll</h1> Lorem ipsum dolor sit
amet, ... Morbi mi.</div>
```



9.4. PROPIEDADES *Z-INDEX*

Además de posicionar una caja de forma horizontal y vertical, CSS permite controlar la **posición tridimensional** de las cajas posicionadas. De esta forma, es posible indicar las cajas que se muestran delante o detrás de otras cajas cuando se producen solapamientos.

| Propiedad | <i>z-index</i> |
|---------------|--|
| Valores | auto numero inherit |
| Se aplica a | Elementos que han sido posicionados explícitamente |
| Valor Inicial | auto |
| Descripción | Establece el nivel tridimensional en el que se muestra el elemento |



9.4. PROPIEDADES *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;}

<div id="caja1">Caja 1 - Caja 1 - Caja 1 ...
Caja 1 - Caja 1 - Caja 1 - Caja 1</div>

<div id="caja2">Caja 2 - Caja 2 - Caja 2 ...
Caja 2 - Caja 2 - Caja 2 - Caja 2</div>

<div id="caja3">Caja 3 - Caja 3 - Caja 3 ...
Caja 3 - Caja 3 - Caja 3 - Caja 3</div>
```

