

# ¿Qué diferencias hay entre display:block, inline e inline-block en CSS?

Por José Manuel Alarcón. Publicado el 6 de octubre de 2015 a las 09:00

['Programación profesional con JavaScript y ECMAScript': Muchos creen que saben JavaScript, pero no es fácil dominarlo »](#)

Una duda típica a la hora de trabajar con HTML y CSS es tener claras las diferencias que existen entre elementos de bloque y elementos en línea, así que vamos a intentar arrojar un poco de luz sobre este tema.

## Elementos en línea

Para explicarlo de manera sencilla, y simplificando, ciertas etiquetas HTML **no afectan en absoluto al flujo de los demás elementos**, limitándose a marcar ciertos fragmentos con una determinada semántica y **dejando que el texto y otros elementos contiguos sigan fluyendo** en la misma línea, colocándose a ambos lados de dicho elemento.

Ejemplos de estas etiquetas son: `<a>`, `<em>`, `<del>`... y quizá la más útil de todas: la etiqueta `<span>`, que se usa para envolver elementos en-línea para darles estilo sin cambiar su comportamiento. A estos elementos HTML se les llama **elementos de línea** o, incluso más a menudo, por su denominación en inglés: **elementos "inline"**.

Por ejemplo, si tenemos el siguiente estilo CSS:

```
.caja {  
    background-color: yellow;  
    border: 1px solid black;  
    width: 100px;  
    height: 100px;  
    padding: 5px;  
    margin: 50px 10px;  
}
```

Y un código HTML tan sencillo como este:

```
<p>Lorem ipsum dolor...</p>  
<span class="caja">CAJA 1</span>
```

```
<span class="caja">CAJA 2</span>
<span class="caja">CAJA 3</span>
<p>Lorem ipsum dolor...</p>
```

Lo que obtenemos en el navegador con este código es esta representación:

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

CAJA 1

CAJA 2

CAJA 3

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Como vemos las cajas tienen un **estilo in-line** porque son *spans*, se alinean unas a continuación de otras, pero **no se respetan los anchos, los altos ni los márgenes verticales de los elementos**.

## Elementos de bloque

Por el contrario, **ciertas etiquetas se renderizan en el navegador en líneas independientes**, no mezcladas con el resto del texto. Ejemplos de estas etiquetas son los encabezados (<h1> hasta <h6>), las citas en bloque (<blockquote>), por supuesto los párrafos (<p>), y quizá la más conocida de todas que es la etiqueta **<div>** usada normalmente para envolver a otros elementos. A estos elementos se les denomina **elementos de bloque**.

Siguiendo con el mismo ejemplo anterior, si simplemente sustituimos los <span> por elementos <div>, ahora las cajas son elementos de bloque, y se renderizarán de la siguiente manera:

Lorem ipsum dolor sit amet, cons

CAJA 1

CAJA 2

CAJA 3

Lorem ipsum dolor sit amet, cons

Es decir, ahora los elementos no fluyen como si fueran texto, sino que por defecto se renderizan en líneas independientes. Además, **los elementos de bloque respetan los anchos y altos y los márgenes verticales**, tal y como se observa en la figura anterior (no se ha cambiado nada más que el tipo de etiqueta: los estilos CSS son exactamente los mismos).

## Cambio de tipo de visualización y un tercer tipo intermedio

Bien, aparte de este comportamiento por defecto, es posible **cambiar el modo en el que se visualiza/renderiza cualquier elemento HTML** de una página usando la **propiedad CSS** llamada **display**.

Esta propiedad puede tomar [muchos valores diferentes](#), pero **los más comunes son estos tres**:

**block**: hace que el comportamiento del elemento sea como un bloque.

**inline**: el elemento se renderizará en línea con otros elementos.

**inline-block**: el elemento tendrá un comportamiento mezcla entre los dos anteriores, que ahora voy a describir.

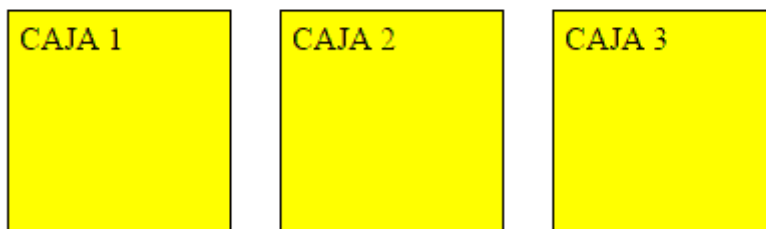
Así por ejemplo, aunque tengamos una etiqueta `span` que se renderiza en línea por defecto, si le aplicamos la propiedad `display: block`; se renderizará como un bloque, es decir, sería exactamente igual que si hubiésemos puesto

un `<div>`.

El tercer valor de la lista anterior es muy interesante ya que permite obtener un comportamiento intermedio entre los otros dos. **Los elementos inline-block fluyen con el texto y demás elementos** como si fueran elementos en-línea **y además respetan el ancho, el alto y los márgenes verticales**. Son lo mejor de los dos mundos.

Así, si por ejemplo redefinimos la etiqueta `<span>` de esta manera y volvemos a refrescar la página del principio, lo que veríamos por pantalla es esto:

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Temporibus, p



Lorem ipsum dolor sit amet, consectetur adipisicing elit. Temporibus, p

Fíjate en que ahora se respetan todas las propiedades (ancho, alto y margen vertical), pero los elementos siguen estando en la misma línea.

## Cambio automático de modo

Una cuestión importante y que muchas veces se nos pasa por alto es que **un elemento en línea puede cambiar de manera automática y pasar a ser un elemento de bloque** en ciertas circunstancias, sin que nos demos cuenta. En concreto si lo flotamos con la propiedad CSS llamada **float**, aunque el elemento sea *inline* (incluso aunque lo indiquemos explícitamente con un `display:inline !important;`) el elemento adquirirá automáticamente el estilo de bloque para poder aplicar la flotación, que tiene preferencia. Esto es **lo que indica el estándar** específicamente.

¡Espero que lo encuentres útil!