

Display

`display` es la propiedad más importante de CSS para controlar el **comportamiento** de una etiqueta o un conjunto de etiquetas.

- `display: block` Por así decirlo es el más básico de todas las opciones que tiene `display`, lo que hace esta propiedad es ocupar todo el ancho (a menos que se le indique un `width`).

A partir de HTML5

Actualmente existen "alternativas" a esto para no tener que usar siempre una etiqueta `<div>` para esto, son las siguientes: `<header>`, `<footer>` y `<section>`.
`<header>` -> Se usa para los encabezados/títulos de la página HTML5. `<footer>` -> Igual que el anterior pero para el pie de página o parte inferior de la página HTML5. `<section>` -> Igual que con las anteriores pero se suele usar para indicar el cuerpo de la página HTML5.

Si dividimos en tres la página usando las etiquetas mencionadas anteriormente, lo podemos simplificar en TÍTULO > CUERPO > FIN.

- `display: inline` Suele acompañar a etiquetas las cuales afectan a una sola palabra como podría ser la etiqueta ``. Al usar la propiedad `display: inline` lo podemos ver como que la etiqueta a la que afecta no va a crear un salto de línea por así decirlo, esta va a forzar el que la siguiente etiqueta empiece como si no hubiera nada a su izquierda ocupando espacio.

~ ~ ~ El siguiente ejemplo que se va a mostrar es una mezcla de `inline` y `block` (-> `inline-block`), a su vez el siguiente ejemplo funcionará de forma muy similar tanto si se indica `display: inline-block;` o `display: inline;`. Para que lo entendamos rápidamente `display: inline-block;` es una evolución de `display: inline;`

Un ejemplo de esto podría ser los menús de selección que aparecen en horizontal en la parte superior de muchas páginas. Para que se entienda mejor en este caso (y en la mayoría de casos que vamos a hacer/encontrar) se usa una etiqueta `` (por defecto esta etiqueta se coloca por debajo de la anterior haciendo que como si de una lista se tratara).

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <style>
    #inline li {
      display: inline-block;
    }
  </style>
</head>
<body>
  <div id="inline">
    <ul>
      <li>WIP</li>
      <li>WIP</li>
      <li>ACERCA DE</li>
      <li>WIP</li>
      <li>WIP</li>
    </ul>
  </div>
</body>
</html>
```

En este ejemplo modificaremos las etiquetas `` dentro de la etiqueta con `id=inline`, como vemos el CSS de esta tiene la propiedad `display: inline-block` en vez de `display: inline`, para el resultado que buscamos la diferencia es mínima, al usar `display: inline-block` nos permite el poder modificar más parámetros en cuanto ancho, altura, márgenes y rellenos. El fin es el mismo y la diferencia mínima, yo recomendaría usar `display: inline-block` siempre para evitar este tipo de "problemas" de ajuste de ancho, altura, márgenes y rellenos.

- `display: none;` Básicamente lo podremos ver como ocultar en todos los aspectos el bloque al que afecta la propiedad. Si lo queremos ver gráficamente, es como si directamente no se hubiera escrito las líneas de código correspondientes para verlo en la página HTML.

Se diferencia con la visibilidad del bloque en: `display: none;` -> no afecta para nada al resto `visibility: hidden;` -> el bloque/etiqueta existe y no se ve, pero ahí está

Aunque no se vea una utilidad a esta propiedad, la puede tener en un futuro al usar JavaScript. Básicamente lo podemos usar para cambiar su propiedad `display:none` (y/o otras propiedades) al pulsar sobre un botón el cual ejecuta una función JavaScript, sin afectar al resto del código de la página HTML.