Propiedad display

Modo de renderización de las cajas

Propiedad display

- Las etiquetas html se clasifican, de acuerdo a su naturaleza, en dos grupos: en línea o de bloque.
- La propiedad display puede modificar esta situación ya que define como se dibuja una caja en pantalla.
- Puede convertir una caja que naturalmente es de bloque, en una en línea, pasando a tener todas las características de ese tipo de etiquetas.
- O convertir una caja naturalmente en línea, en una de bloque, con todas sus características.

Primeros valores

- En principio los valores que cambian el comportamiento de la etiqueta al contrario son:
- display: inline; Si se la aplicamos a una etiqueta de bloque pasa a trabajar en línea. Obviamente si se la aplicamos a una naturalmente en línea no le produce ningún cambio.
- display: block; Si se la aplicamos a una etiqueta en línea pasa a trabajar de bloque. Obviamente si se la aplicamos a una naturalmente de bloque no le produce ningún cambio.
- De esta forma, "forzamos" a la etiqueta a trabajar inline o block.

display: inline

- Se ubican una al lado de la otra, y en línea con el texto.
- Aplican margin, padding y border en el eje horizontal ocupando espacio y empujando a los elementos a su lado.
- En el eje vertical: no aplican margin (top y bottom) y el padding y el border se aplican pero “no empujan”, se superpone con los elementos de la línea de arriba y la de abajo, si el tamaño excede el alto de línea donde se encuentra ubicada.
- Tampoco aplican valores de width y height desde CSS (su ancho y alto dependen exclusivamente del contenido).

display: block

- Se ubican una debajo de la otra.
- Aplican margin, padding y border en los cuatro costados de la caja haciéndola ocupar un mayor espacio (empujan a los demás elementos).
- También aplican valores de width y height desde CSS.
- Fusionan los márgenes del eje vertical entre cajas consecutivas (top de la inferior y bottom de la superior).
- En esa fusión se expresa SOLO el que tenga el mayor valor de ambos.
- Fusionan los márgenes verticales (top o bottom) entre cajas anidadas (primer y último hijo de un contenedor).
- En esa fusión se expresa SOLO el mayor de ambos al contenedor.
- Se puede cortar esta fusión aplicando padding o border al contenedor.

display: inline-block

- Es una mezcla de las dos naturalezas. No existe ninguna etiqueta que tenga éste comportamiento por defecto. Solo se puede forzar desde CSS y aplica "lo mejor" de cada tipo.
- Las cajas funcionan inline, una al lado de la otra, ocupando el ancho y el alto que necesiten de acuerdo a su contenido.
- Pero conservan las mejores características de las cajas de bloque: aplican width, height, padding, margin y border en los 4 costados, empujando a los elementos circundantes.
- No fusionan márgenes verticales entre ellas (cuando no entran en línea por tamaño y una baja a la línea inferior)
- "Estiran" verticalmente la línea en la que se ubican de acuerdo a su alto (por contenido o porque le aplicamos un height desde CSS).
- Por trabajar en línea se centran con text-align center (al padre).

Alineación vertical

- Si tenemos dos etiquetas inline-block consecutivas (una al lado de la otra) con distinto tamaño de alto, porque le aplicamos height distintos o por cantidad de contenido, se pueden alinear verticalmente con la propiedad vertical-align.
- Los valores de esa propiedad son top, bottom, middle o baseline.
- baseline: valor por defecto, el último elemento de cada caja se apoya sobre la línea, si tienen distinto tamaño de padding la parte inferior de las cajas queda desfazada.
- bottom: alínea las aristas inferiores de las cajas.
- top: alínea las aristas superiores de las cajas.
- middle: alínea las cajas al centro dentro del alto de línea.
- Los tres primeros valores alcanza con aplicárselos solo a la menos alta, el valor middle debe aplicarse a todas las cajas.

Para tener en cuenta...

- Esta es una gran herramienta de maquetación ya que nos permite hacer columnas que crezcan verticalmente de acuerdo a su contenido (en general) o ajustándoles sus altos (y anchos, obvio).
- Pero… las cajas inline-block , cuando se ubican una al lado de la otra, dejan, por defecto, un espacio entre ellas, que puede complicarnos a la hora de maquetar.
- Esto se debe a que este tipo de elementos son tratados como si fueran “palabras” y para que dos palabras no “se peguen” pareciendo que hubiera una sola, les agregamos un espacio con la barra espaciadora.
- Por lo que el browser les agrega, por defecto, ese espacio con esa misma intención (no sabe si son dos elementos como palabras que funcionan en línea o si son dos columnas de contenido).

Resolviendo...

- Si queremos eliminar ese espacio, la solución depende de cómo escribimos el html.
- **¡SI!** a diferencia de lo dicho anteriormente que no importa cómo escribimos el código, las etiquetas se ubican una al lado de la otra o por debajo dependiendo de su naturaleza, acá si importa.
- Si escribimos **pegadas** las etiquetas de cierre de la primera caja con la de apertura de su consecutiva (ambas inline-block) el espacio desaparece: `<article>…</article><article>…</article>`
- O si las unimos por medio de un comentario html

```
<article>…</article><!-

-->
<article>…</article>
```

Resolviendo...

- Si queremos eliminar e… …cómo escribimos el html.
- **¡SI!** a diferencia de lo … …cómo escribimos el código, la… …la otra o por debajo dependiendo de su naturaleza, aca si importa.
- Si escribimos **pegadas** las etiquetas de cierre de la primera caja con la de apertura de su consecutiva (ambas inline-block) el espacio desaparece: `<article>…</article><article>…</article>`
- O si las unimos por medio de un comentario html

```
<article>…</article><!-

--><article>…</article>
```

Si las separamos aunque sea con un solo espacio producido por la barra espaciadora, las cajas vuelven a separarse entre ellas

Resolviendo...

- Si queren… epende de cómo escribimo…
- **¡SI!** a dif… no importa cómo escribimo… a al lado de la otra o por deb… á si importa.
- Si escribimos **pegadas** las etiquetas de cierre de la primera caja con la de apertura de su consecutiva (ambas inline-block) el espacio desaparece: `<article>…</article><article>…</article>`
- O si las unimos por medio de un comentario html

```
<article>…</article><!-

--><article>…</article>
```

El comentario puede tener texto o quedar vacío, incluso podemos aplicar un salto de línea en el código con el enter que las etiquetas seguirán unidas por él.

display: none

- Todas las etiquetas que escribimos en el código (estén bien o mal escritas) el navegador las va a dibujar en pantalla.
- Pero con el valor none se pueden quitar elementos del layout.
- No es que el elemento se haga transparente (lo que haría que siga ocupando el lugar que le corresponde).
- Sino que no ocupa lugar, no se dibuja y los elementos maquetados a continuación, ocupan el lugar vacío que queda.
- Es como si nunca lo hubiésemos escrito en el código, normalmente esta característica es muy útil para hacer efectos o que "aparezcan" elementos con alguna acción del usuario (por ejemplo, con la pseudoclase hover).