



Bloque y en línea

Curso de Fundamentos

Introducción

En la lección anterior, descubrimos que los distintos tipos de visualización tienen modelos de cuadro únicos y podemos modificar el cálculo del cuadro cambiando la `box-sizing` propiedad. CSS tiene dos tipos de cuadro: `block` y `inline` cuadros, que determinan el comportamiento y la interacción de los elementos. La `display` propiedad controla cómo aparecen los elementos HTML en la página web. Exploraremos sus distintas opciones con más detalle en esta lección.

Resumen de la lección

Esta sección contiene una descripción general de los temas que aprenderá en esta lección.

- Aprenderá sobre el "flujo normal".
- Aprenderá la diferencia entre elementos `block` y `inline`.
- Aprenderá qué elementos tienen como valor predeterminado `block` y cuáles tienen como valor predeterminado `inline`.
- Aprenderá qué son `div`s y `span`s.

Bloque vs en línea

La mayoría de los elementos que has aprendido hasta ahora son elementos de bloque. En otras palabras, su estilo predeterminado es `display: block`. De forma

predeterminada, los elementos de bloque aparecerán en la página apilados uno sobre otro y cada elemento nuevo comenzará en una nueva línea.

Sin embargo, los elementos en línea no comienzan en una nueva línea. Aparecen en línea con los elementos a su lado. Un claro ejemplo de un elemento en línea es un enlace o `<a>` etiqueta. Si colocas uno de estos en medio de un párrafo de texto, [el enlace se comportará como una parte del párrafo](#). Además, el relleno y el margen se comportan de manera diferente en los elementos en línea. En general, no debes intentar colocar relleno o margen adicional en los elementos en línea.

El bloque en línea de término medio

Los elementos de bloque en línea se comportan como elementos en línea, pero con relleno y margen de estilo bloque. `display: inline-block` es una herramienta útil que conviene conocer, pero en la práctica, probablemente termines recurriendo a flexbox con más frecuencia si estás intentando alinear un montón de cuadros. Flexbox se tratará en profundidad en la próxima lección.

Divisiones y tramos

No podemos hablar de elementos en bloque y en línea sin hablar de los divs y los spans. Todos los demás elementos HTML que hemos encontrado hasta ahora dan significado a su contenido. Por ejemplo, los elementos de párrafo le indican al navegador que muestre el texto que contiene como un párrafo. Los elementos fuertes le indican al navegador qué textos son importantes, etc. Sin embargo, los divs y los spans no le dan un significado particular a su contenido. Son simplemente cajas genéricas que pueden contener cualquier cosa.

Disponer de elementos como estos es mucho más útil de lo que parece a primera vista. A menudo necesitaremos elementos que no tengan otra finalidad que la de ser elementos "gancho". Podemos darles un id o una clase para que se les aplique estilo con CSS. Otro caso de uso al que nos enfrentaremos con regularidad es el de agrupar elementos relacionados bajo un elemento padre para posicionarlos correctamente en la página. Los divs y los spans nos proporcionan la capacidad de hacer esto.

Div es un elemento de nivel de bloque por defecto. Se utiliza habitualmente como elemento contenedor para agrupar otros elementos. Los divs nos permiten *dividir* la página en diferentes bloques y aplicar estilos a esos bloques.

HTML

CSS

Result

EDIT ON
CODEPEN

```
<div class="introduction">
  <h2>Introduction</h2>
</div>

<div class="main-content">
  <h2>Main Content</h2>
</div>

<div class="contact-us">
  <h2>Contact Us</h2>
</div>
```

Run Pen 

Resources1x0.5x0.25xRerun

Span es un elemento de nivel en línea de forma predeterminada. Se puede utilizar para agrupar contenido de texto y elementos HTML en línea para aplicarles estilo y solo se debe utilizar cuando ningún otro elemento HTML semántico sea apropiado.


HTML

CSS

Result

EDIT ON
CODEPEN

```
<p>
  Lorem ipsum dolor sit amet,
  consectetur adipiscing elit, sed do
  eiusmod tempor incididunt ut
  labore et dolore magna aliqua. Ut
  enim ad
  minim veniam, <span
  class="highlight">quis nostrud <a
  href="https://www.dictionary.com/browse
  ullaamco laboris</span> nisi ut
  aliquip ex ea commodo consequat.
```

Run Pen 

Resources1x0.5x0.25xRerun

Asignación

1. El concepto de “flujo normal” está implícito en los recursos del modelo de caja, pero no se explica de forma muy específica. Lee [“Flujo normal” de MDN](#) para asegurarte de que entiendes cómo se disponen los elementos de forma predeterminada.
2. [“Bloques HTML y elementos en línea”](#) de las escuelas W3 tiene una descripción y una lista de todos los bloques HTML y elementos en línea predeterminados.

3. El tutorial de Digital Ocean [“Visualización en línea vs. visualización en bloque en línea en CSS”](#) tiene un par de excelentes ejemplos que aclaran la diferencia entre `inline` y `inline-block`.
4. Realiza los ejercicios en [el directorio de nuestro repositorio de ejercicios CSS foundations/block-and-inline](#) (recuerda que las instrucciones están en el README) en el orden:
 - `01-margin-and-padding-1`
 - `02-margin-and-padding-2`

Nota: Las soluciones de estos ejercicios se pueden encontrar en la `solution` carpeta de cada ejercicio.

5. Aplica lo que aprendiste sobre el modelo de caja para mejorar el aspecto de la `index.html` página de inicio de tu página de Recetas. Actualmente, es solo una lista simple, así que sé creativo con los diseños, colores y estilos para que tu página sea única y atractiva.

Comprobación de conocimientos

Las siguientes preguntas son una oportunidad para reflexionar sobre temas clave de esta lección. Si no puede responder una pregunta, haga clic en ella para revisar el material, pero tenga en cuenta que no se espera que memorice o domine este conocimiento.

- [¿Cuál es la diferencia entre un elemento de bloque y un elemento en línea?](#)
- [¿Cuál es la diferencia entre un elemento en línea y un elemento de bloque en línea?](#)
- [¿Es un `h1` bloque o en línea?](#)
- [¿Es `button` en bloque o en línea?](#)
- [¿Es `div` en bloque o en línea?](#)
- [¿Es `span` en bloque o en línea?](#)

Recursos adicionales

Esta sección contiene enlaces útiles a contenido relacionado. No es obligatoria, por lo que se la puede considerar complementaria.

- [Learn CSS Layout](#) es un tutorial un poco anticuado, pero sus ejemplos son claros. Las primeras 6 diapositivas cubren el material que hemos visto hasta ahora.
- Mire este breve video sobre [lo que significa el término "Flujo normal"](#) en CSS.
- Para obtener una explicación y un ejemplo más interactivos, pruebe este [Scrim en bloque y visualización en línea](#).

[!\[\]\(e2376d476d06eb31946dc01a69a4403a_img.jpg\) Mejorar en GitHub](#)[!\[\]\(74d4806277d7e73349d8e8c0897931e9_img.jpg\) Informar un problema](#)[Ver registro de cambios de la lección](#)[Ver curso](#)[Marcar como completado](#)[Próxima lección](#)

¡Apóyanos!

El Proyecto Odin está financiado por la comunidad. ¡Únase a nosotros para ayudar a estudiantes de todo el mundo apoyando el Proyecto Odin!

[Más información](#)[Dona ahora](#)

THE ODIN PROJECT

Educación en codificación de alta calidad mantenida por una comunidad de código abierto.



Sobre nosotros

[Acerca de](#)

[Equipo](#)

[Blog](#)

[Casos de éxito](#)

Apoyo

[Preguntas frecuentes](#)

[Contribuir](#)

[Contáctenos](#)

Guías

[Guías de la comunidad](#)

[Guías de instalación](#)

Legal

[Términos](#)

[Privacidad](#)

© 2025 El Proyecto Odin. Todos los derechos reservados.