



IT Academy
by KIBERNUM



El Modelo de Cajas (Box Model)

¿Qué aprendiste en la clase anterior sobre Sass y el patrón 7-1 que te ayudará a organizar mejor los estilos en tus futuros proyectos?





Propósito de la clase

En esta sesión, exploraremos el modelo de cajas en CSS y cómo afecta la organización y presentación de los elementos en una página web. Además, aprenderemos sobre los distintos tipos de posicionamiento y cuál es su impacto en la disposición del contenido. Finalmente, analizaremos los layouts más utilizados en el diseño web y sus ventajas y desventajas.

Al finalizar esta clase, serás capaz de:

- Comprender el modelo de cajas en CSS, identificando sus propiedades principales y cómo influyen en la estructura de los elementos.
- Diferenciar los tipos de cajas en CSS, reconociendo las diferencias entre elementos de bloque y línea y su impacto en el diseño.
- Inspeccionar y analizar elementos en el navegador, utilizando herramientas de desarrollo para visualizar márgenes, rellenos y bordes.
- Aplicar correctamente propiedades de posicionamiento, incluyendo estático, relativo, absoluto, fijo y flotante, para controlar la disposición de los elementos en una página.
- Definir qué es un layout y sus tipos (fluido, fijo, elástico y absoluto), comprendiendo sus ventajas y desventajas en distintos escenarios de diseño.

📌 Preguntas de Reflexión Iniciales

Antes de entrar en materia, piensa en estas preguntas:

- ❓ ¿Has intentado modificar el diseño de un sitio web y te ha costado entender cómo están organizados los elementos?
- ❓ ¿Sabías que cada elemento en una página web es una caja y que su tamaño y posición dependen de propiedades específicas en CSS?
- ❓ ¿Alguna vez has cambiado la posición de un elemento y el resto de la página se desordenó? ¿Sabes por qué sucede esto?
- 📌 Reflexiona sobre estos puntos, ya que en esta clase aprenderemos a comprender el modelo de cajas en CSS, identificar cómo se estructuran los elementos en la página y aplicar distintas técnicas de posicionamiento y diseño para organizarlos de manera eficiente. 🚀



Inspector de Elementos



🔍 Inspector de Elementos en Chrome y el Modelo de Caja en CSS

El Inspector de Elementos de Google Chrome es una herramienta poderosa que nos permite analizar y modificar el código HTML y CSS de cualquier página web en tiempo real.

Con él, podemos inspeccionar cómo los navegadores interpretan el modelo de cajas en CSS y ajustar estilos sobre la marcha.



A screenshot of the Google Chrome DevTools Elements tab. The main view shows the Google homepage with the search bar and logo. A tooltip is open over the Google logo, displaying accessibility information: "img.InXdpd 272 x 92", "ACCESSIBILITY", "Name: Google", "Role: image", and "Keyboard-focusable: false". The DevTools sidebar on the right shows the DOM tree and the CSS styles applied to the elements. The "Styles" panel is visible, showing the current styles for the selected element. The "Computed" panel shows the final computed styles after all declarations and overrides. The "Layout" panel shows the element's position and dimensions within the page. The "Event Listeners" panel shows any event listeners attached to the element. The "Memory" panel shows memory usage details. The "Network" panel shows network requests. The "Performance" panel shows performance metrics. The "Sources" panel shows the source code of the page. The "Network" panel shows network requests. The "Console" panel shows developer console logs. The "What's new" panel shows highlights from the latest update. The "Highlights from the Chrome 130 update" panel shows recommendations for new live metrics in Performance. The "Recommendations for the new live metrics in Performance" panel shows information about new live metrics in the Performance panel. The "The new live metrics in the Performance panel now provide" panel shows a brief description of the new live metrics.



Accediendo al Inspector de Elementos en Chrome

1 Método Rápido:

Haz clic derecho sobre cualquier elemento de una página web y selecciona "Inspeccionar".

También puedes usar el atajo de teclado:

- Windows/Linux: Ctrl + Shift + I
- Mac: Cmd + Option + I

2 Apertura desde Herramientas de Desarrollo:

- Presiona F12 para abrir las herramientas de desarrollo y selecciona la pestaña "Elements".

The screenshot shows a browser window displaying the KIBERNUM website. The DevTools Elements tab is open, with the cursor hovering over the main title 'It Academy'. A tooltip shows the element path: 'h1.elementor-heading-title.elementor-size-default'. The right panel displays the DOM structure of the page, with the title element highlighted. The CSS panel on the right shows the styles applied to the title, including 'font-family: var(--e-global-typography-vamta)', 'font-size: var(--e-global-typography-vamta)', and 'font-weight: var(--e-global-typography-vamta)'. The bottom status bar indicates 'Highlights from the Chrome 130 update'.

Explorando el Modelo de Caja con el Inspector

Una vez abierto el Inspector de Elementos, sigue estos pasos para analizar el modelo de caja (Box Model):

- ◆ Selecciona un elemento en la pestaña "Elements".
- ◆ Ve a la sección "Styles" (Estilos) en el panel lateral derecho.
- ◆ Desplázate hasta la sección "Box Model".



The screenshot shows the Chrome DevTools interface with the 'Elements' tab selected. A specific element in the page's DOM tree is highlighted, and its detailed properties are shown in the 'Styles' panel on the right. The 'Box Model' section displays the element's bounding box with its dimensions: width (930px), height (278.844px), padding (15px), margin (32px), and border (0px). The page content visible includes sections like 'CONOCE NUESTROS PILARES' and 'NUESTROS BOOTCAMPS'.



¿Qué muestra el Modelo de Caja en el Inspector?

El Box Model se representa en un diagrama visual con los siguientes elementos:

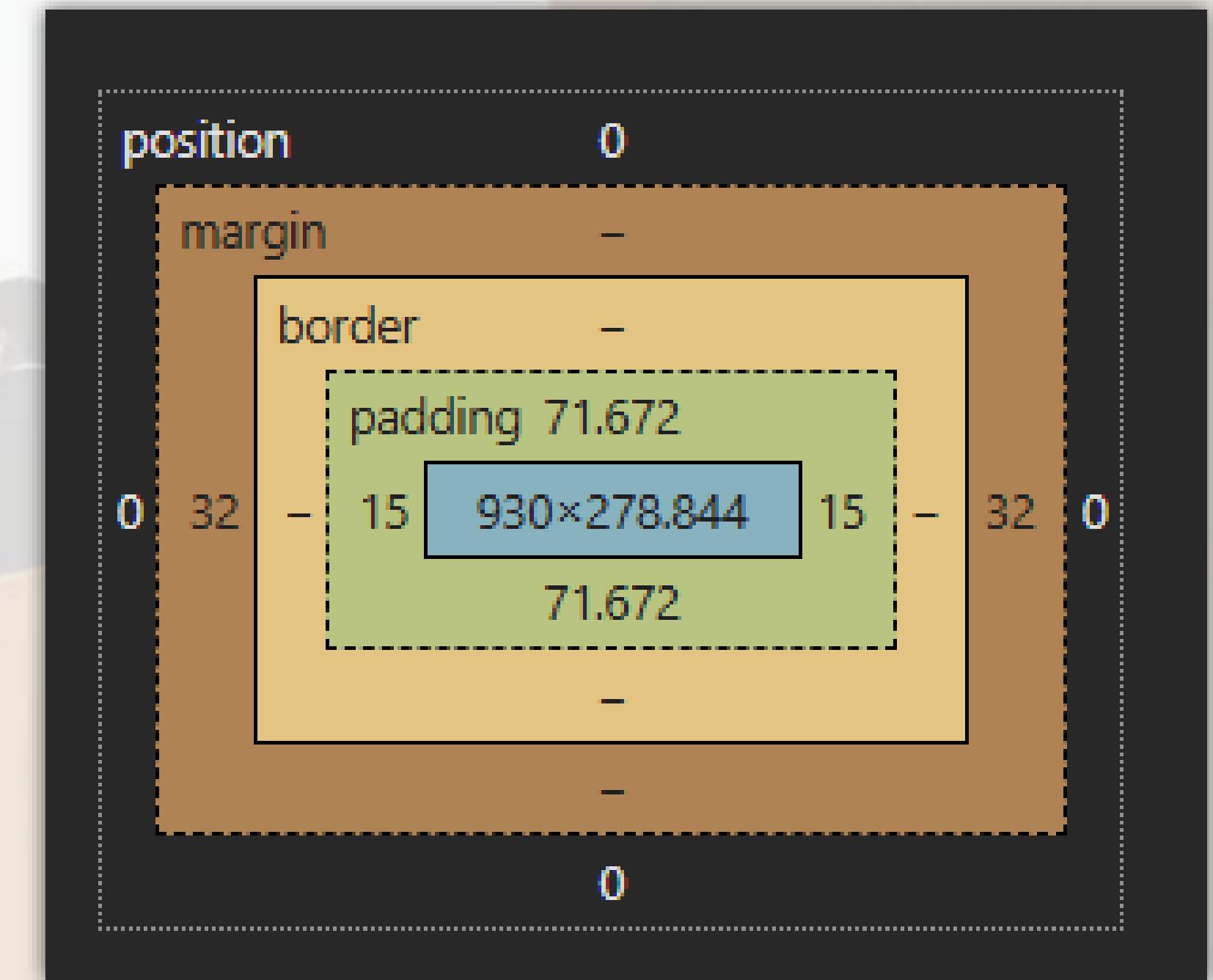
1 Contenido (Content): Representa el espacio donde se muestra el texto o las imágenes dentro del elemento.

2 Padding: Es el espacio entre el contenido y el borde del elemento.

3 Border: Representa el borde del elemento.

4 Margin: Es el espacio externo que separa el elemento de otros elementos de la página.

💡 **Pro tip:** Puedes modificar directamente los valores de margin, border y padding en el panel del Box Model para ver cambios en tiempo real.



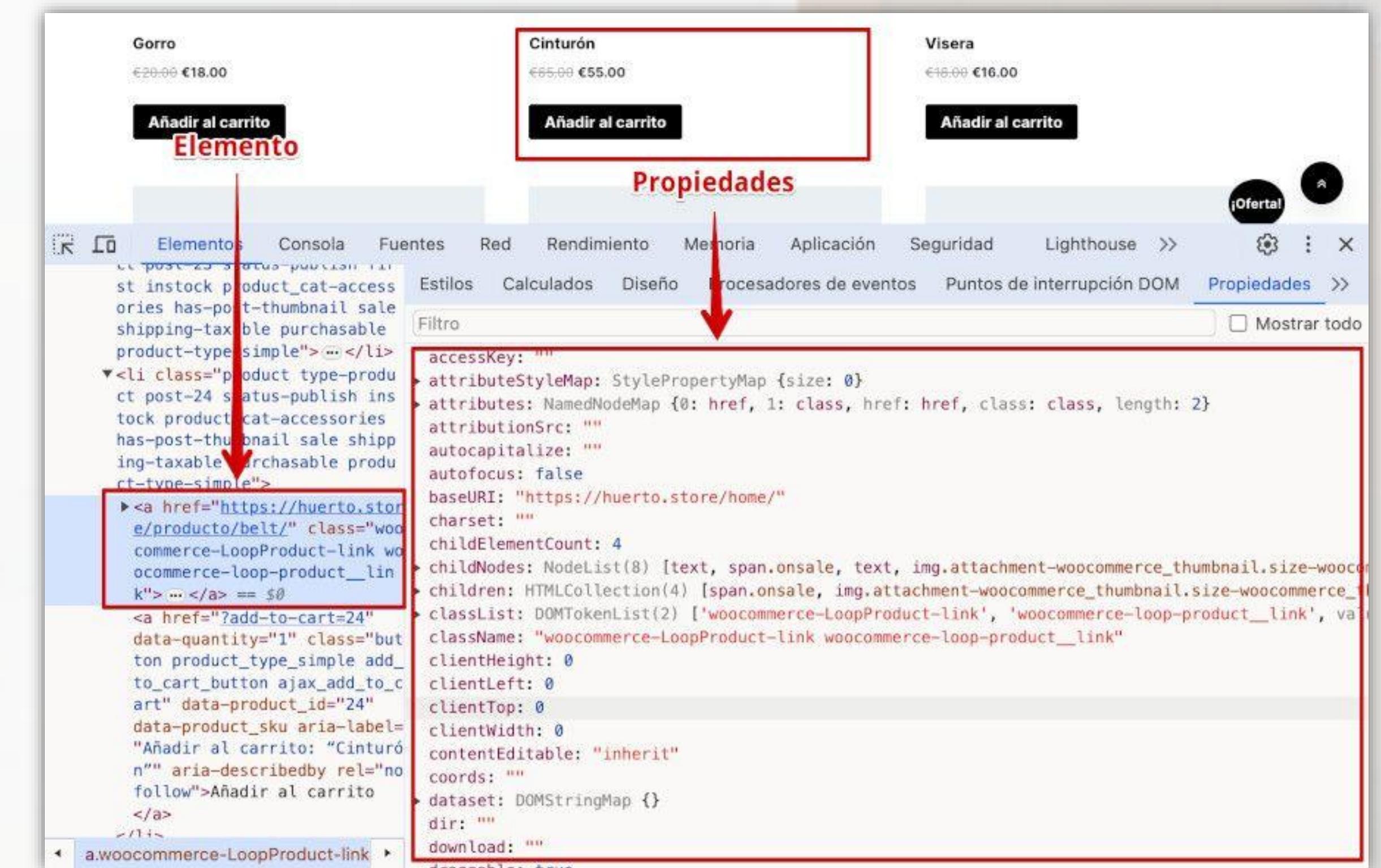


Ejemplo Práctico: Inspeccionemos un Elemento

- 1 Abre cualquier página web (puedes usar <https://www.google.com>).
- 2 Selecciona un elemento (como el cuadro de búsqueda) con el Inspector.
- 3 Explora su Box Model en la pestaña "Styles".
- 4 Cambia los valores de padding o margin y observa el resultado en la página.

¿Por qué es útil el Inspector de Elementos?

- Nos ayuda a visualizar y entender cómo el navegador interpreta el modelo de caja.
- Nos permite hacer ajustes en tiempo real antes de modificar el código.
- Facilita el debugging de estilos y el diseño responsivo.
- Es una herramienta fundamental para cualquier desarrollador frontend.



El Modelo de Cajas y el Layout en CSS

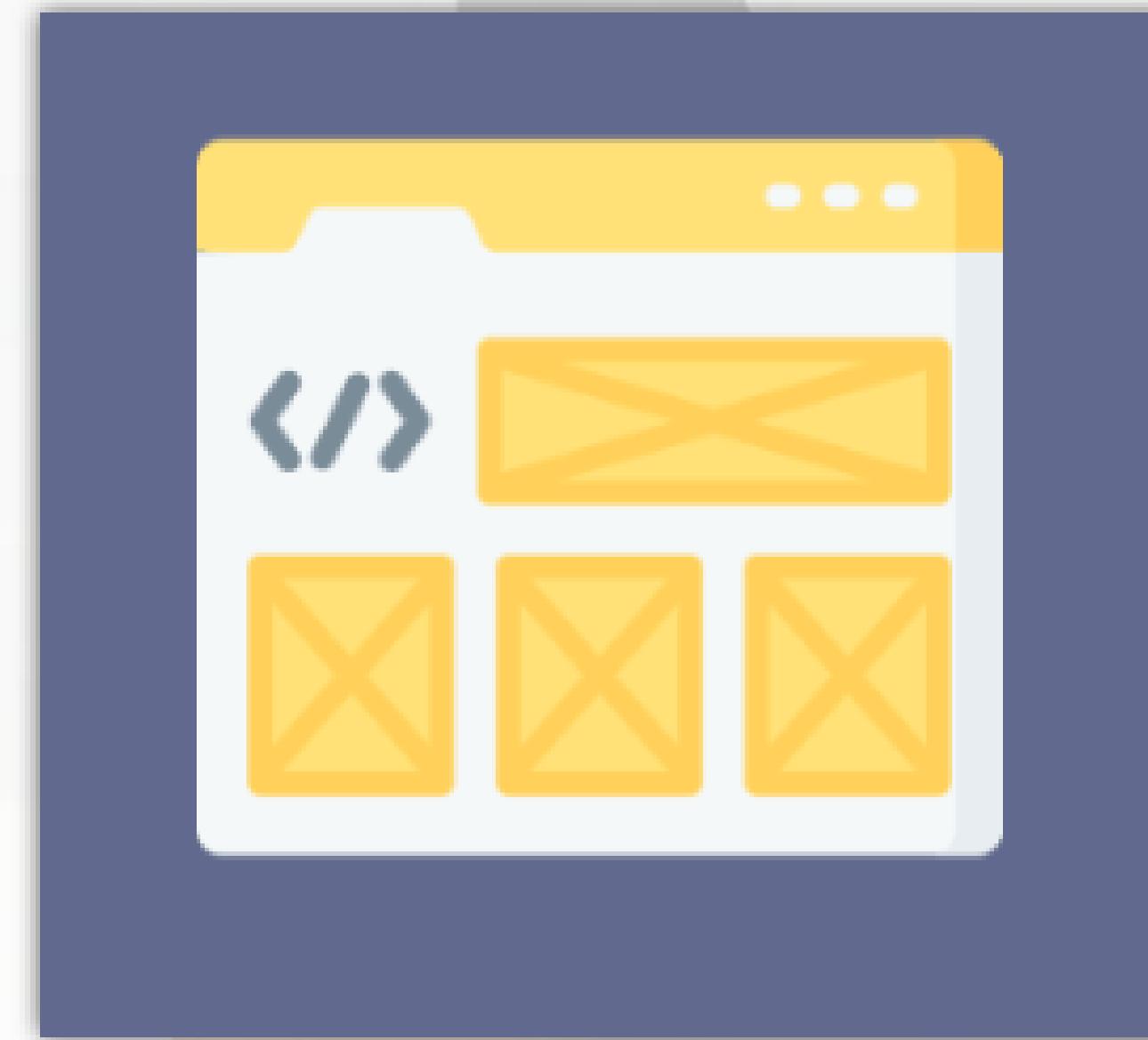




¿Qué es el Modelo de Cajas?

CSS organiza todos los elementos de una página web como si fueran cajas. Cada imagen, párrafo, botón o encabezado está dentro de una caja invisible, y esta caja tiene propiedades que controlan su tamaño, espacio y comportamiento.

🔊 Imagina que cada elemento de tu página es una caja dentro de otra caja, como muñecas rusas.



The screenshot shows a code editor interface with two tabs: 'index.html' and 'styles.css'. The 'styles.css' tab contains the following CSS code:

```
.box{border: 10px solid; padding: 50px 20px 80px 30px; margin: 30px;}
```

Below the code editor is a preview window titled 'Modelo de caja' (Box Model). It displays two identical boxes, each containing placeholder text: 'Lorem ipsum dolor sit, amet consectetur adipisicing elit. Dolor delectus omnis quibusdam quis aut natus deserunt minima placeat, doloremque nobis nesciunt incidunt maxime. Cupiditate natus quo quia optio nam repellendus.' The top box has a thick black border, while the bottom box has a thin black border. A purple highlighter is used to emphasize the 'border' property in the CSS code and the corresponding border around the top box in the preview.



Fuente: <https://www.youtube.com/shorts/8DhSL0YxWAA>



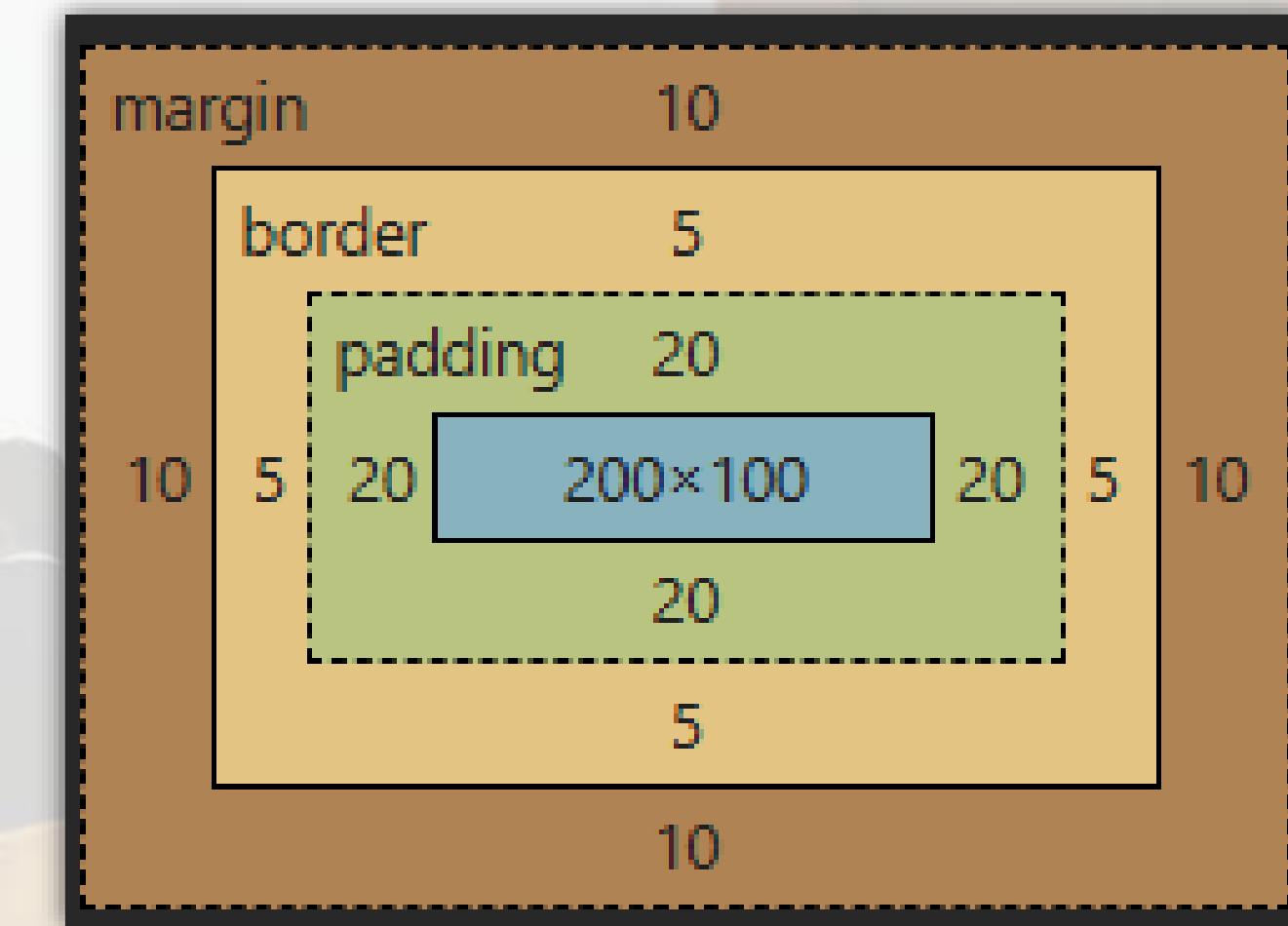
¿Cómo está compuesta una caja en CSS?

📦 Cada caja en CSS tiene cuatro partes principales:

- 1 **Contenido (Content)**: Es lo que realmente ves (texto, imagen, botón, etc.).
- 2 **Padding (Relleno)**: Espacio entre el contenido y el borde de la caja.
- 3 **Border (Borde)**: Línea que rodea la caja.
- 4 **Margin (Margen)**: Espacio entre esta caja y las cajas cercanas.

💡 Ejemplo visual de una caja en CSS:

```
.caja {  
    width: 200px;          /* Ancho del contenido */  
    height: 100px;         /* Alto del contenido */  
    padding: 20px;         /* Espacio entre el contenido y el borde */  
    border: 5px solid black; /* Borde negro */  
    margin: 10px;          /* Espacio con otros elementos */  
    background-color: lightblue; /* color de fondo */  
}
```



- ✓ El padding empuja el contenido hacia adentro.
- ✓ El margen empuja la caja lejos de otras cajas.



¿Existen otros modelos de cajas?

1 Modelo de Caja Estándar (Default Box Model)

- El tamaño de la caja es **contenido + padding + borde**.
- Si defines **width: 200px**, el padding y el borde **se suman** al tamaño total.

2 Modelo **border-box** (Muy utilizado actualmente)

- El tamaño total **incluye el padding y el borde**.
- Si defines **width: 200px**, el padding y el borde **NO afectan** el tamaño **total**.

```
.caja {  
    width: 200px;  
    padding: 20px;  
    border: 5px solid black;  
    box-sizing: border-box; /* No suma padding y borde al ancho total */  
}
```

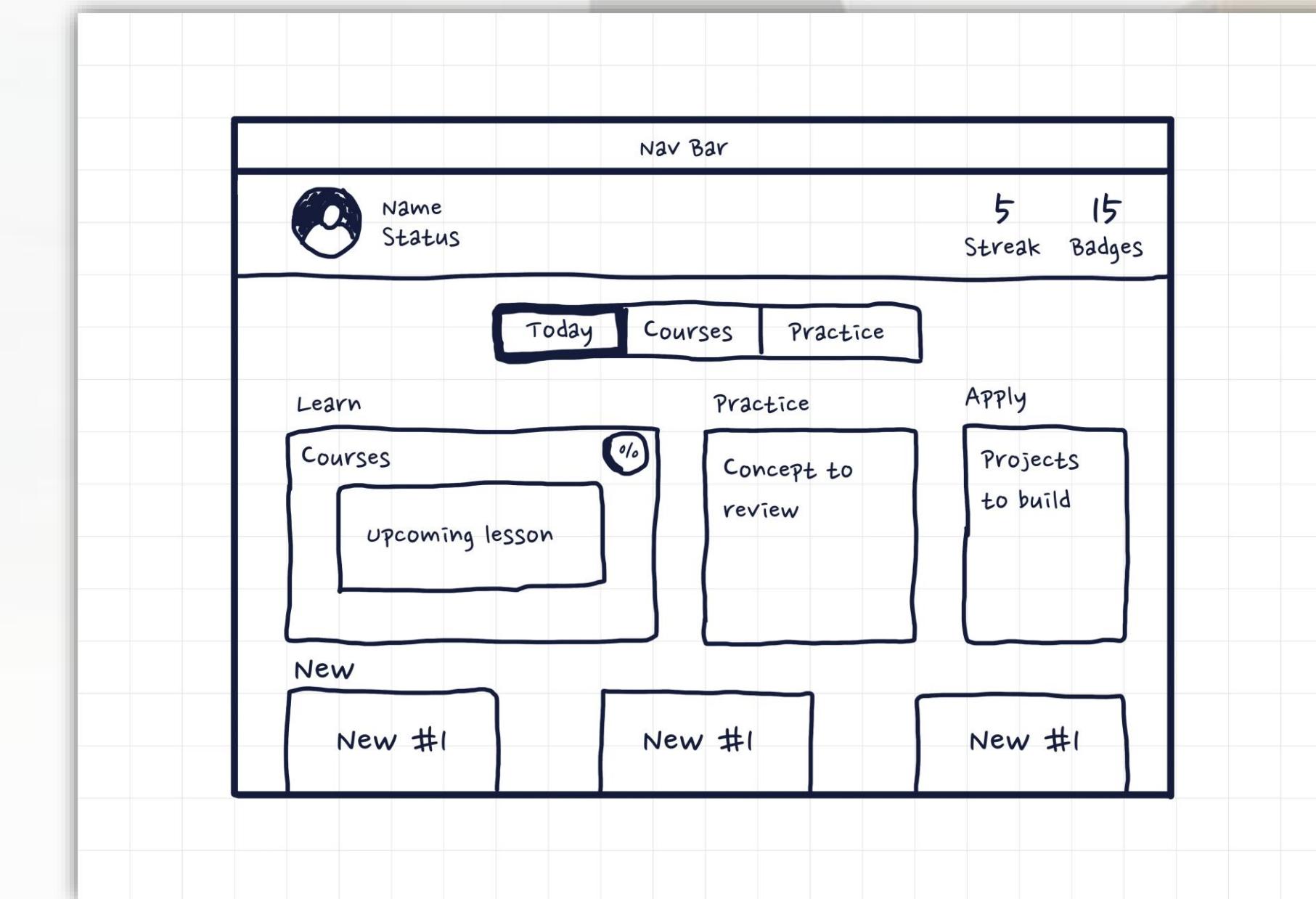


Ejemplo de border-box



¿Por qué es mejor usar box-sizing: border-box; en CSS?

Cuando diseñamos interfaces web, una de las mayores frustraciones es cómo el tamaño de los elementos cambia al agregar padding y bordes. Aquí es donde box-sizing: border-box; se vuelve una opción más predecible y manejable. Vamos a analizarlo en detalle.



Fuente: <https://www.codecademy.com/article/learn-wireframing>

🛠 ¿Cómo funciona box-sizing?

El box-sizing define cómo se calculan las dimensiones de un elemento en CSS. Hay dos valores principales:

content-box (Valor por defecto)

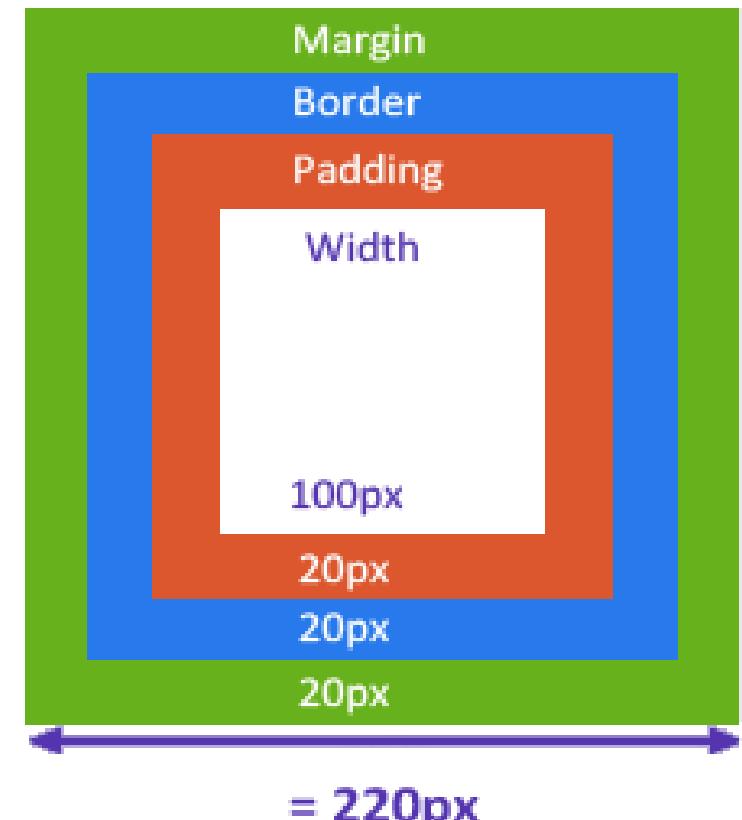
- 📏 El ancho y alto solo incluyen el contenido.
- 🧩 El padding y el borde se suman aparte, lo que puede hacer que el elemento ocupe más espacio del esperado.

border-box (Más predecible y recomendado)

- 📏 El ancho y alto incluyen contenido, padding y borde.
- 🛠 Facilita el diseño, ya que el tamaño del elemento no cambia cuando agregamos padding o bordes.

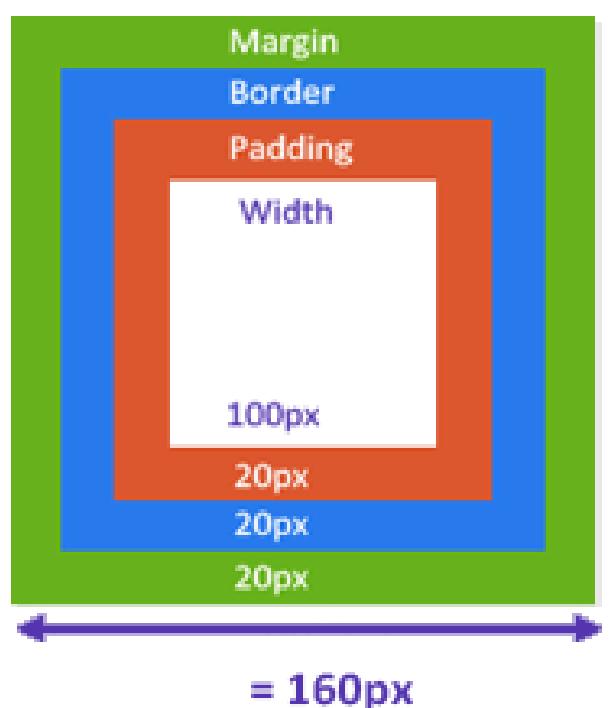
box-sizing : content-box;

Content-box model exclude the border, padding and margin



box-sizing : border-box;

In border-box model all included inside with (margin, padding, border)



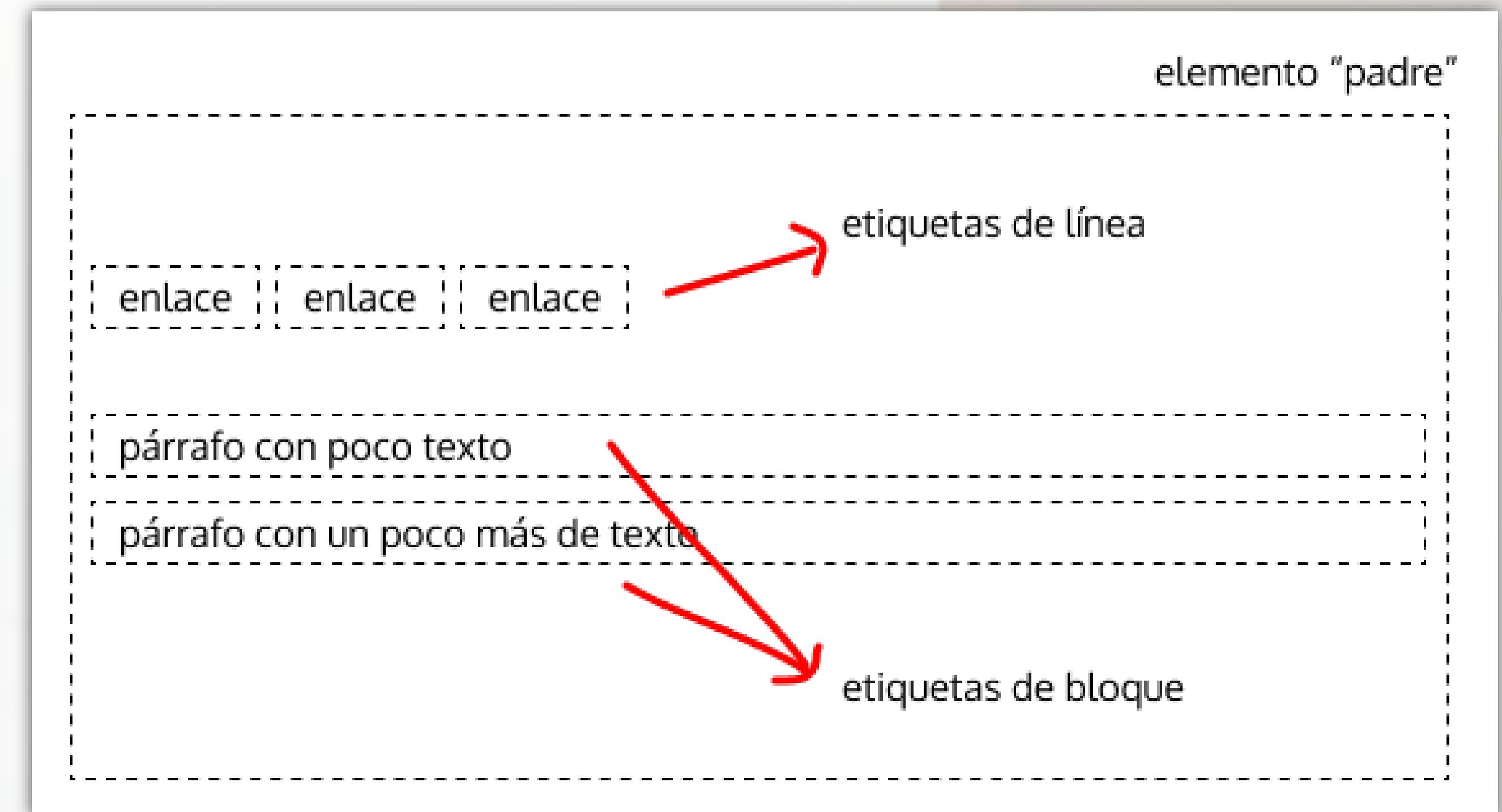
📌 Tipos de Cajas en CSS: Bloque vs. Línea

◆ Elementos de Bloque 📏

- Ocupan **todo el ancho disponible**.
- Se apilan unos sobre otros.
- Ejemplos: <div>, <p>, <h1>, <section>.

◆ Elementos de Línea 📄

- Solo ocupan el **espacio necesario**.
- Se colocan **uno al lado del otro** si hay espacio.
- Ejemplos: , <a>, .

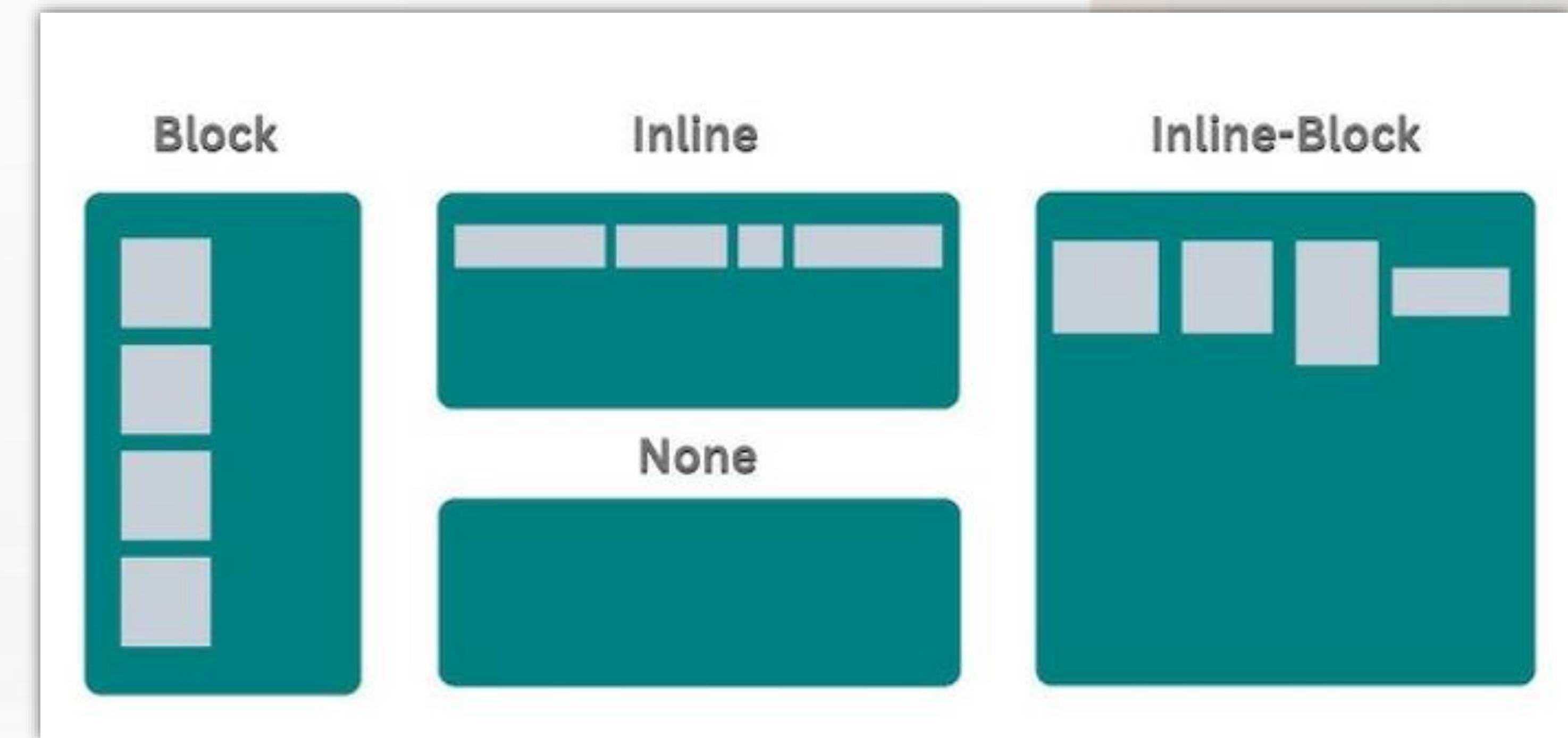


Fuente: <https://laurachuburu.com/elementos-de-linea-y-de-bloque/>

📌 Diferencia entre Elementos de Bloque, Línea e Inline-Block

En HTML y CSS, los elementos tienen una propiedad predeterminada llamada display, que define cómo se comportan en la página. Los tres tipos principales son:

- 1 Elementos de Bloque (block)
- 2 Elementos de Línea (inline)
- 3 Elementos en Línea-Bloque (inline-block)



◆ Elementos de Bloque (display: block)

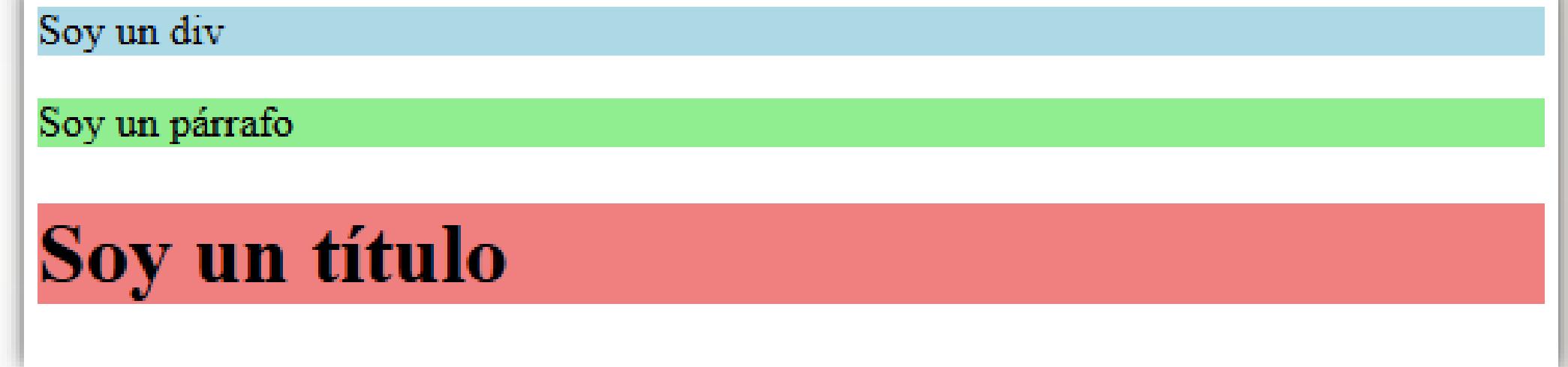
Los elementos de bloque ocupan todo el ancho disponible de su contenedor y siempre comienzan en una nueva línea.

📌 Características:

- Ocupa todo el ancho disponible de su contenedor.
- Comienza en una nueva línea y empuja a los siguientes elementos hacia abajo.
- Se pueden definir width, height, margin y padding sin restricciones.
- Puede contener otros elementos de bloque o en línea.

📋 Ejemplo de Elementos de Bloque:

```
<div style="background: lightblue;">Soy un div</div>
<p style="background: lightgreen;">Soy un párrafo</p>
<h1 style="background: lightcoral;">Soy un título</h1>
```



🔍 Observación: Cada uno de estos elementos ocupa toda la línea, empujando el siguiente hacia abajo.

Ejemplo con CSS

```
<div style="background: lightblue;">Soy un div</div>
<p style="background: lightgreen;">Soy un párrafo</p>
<h1 style="background: lightcoral;">Soy un título</h1>
```

```
div {
    display: block;
    width: 300px;
    height: 100px;
    background-color: lightblue;
}
```

Soy un div

Soy un párrafo

Soy un título

☞ El div tomará 300px de ancho y 100px de alto porque los elementos de bloque permiten modificar width y height.

◆ Elementos de Línea (display: inline)

Los elementos en línea solo ocupan el espacio necesario para su contenido y NO comienzan en una nueva línea.

📌 Características:

- Solo ocupa el ancho del contenido, no toda la línea.
- No inicia una nueva línea, se coloca junto a otros elementos en línea.
- No se pueden definir width ni height, ya que su tamaño se ajusta al contenido.
- Solo acepta margin y padding en los lados izquierdo y derecho.

📋 Ejemplo de Elementos en Línea:

```
<span style="background: yellow;">Soy un span</span>
<a href="#" style="background: pink;">Soy un enlace</a>
<strong style="background: lightgray;">Soy un texto en negrita</strong>
```

```
span {
    display: inline;
    width: 200px; /* No tendrá efecto */
    height: 100px; /* No tendrá efecto */
    background-color: yellow;
}
```

📌 Aunque definimos width y height, el span ignorará esos valores porque es un elemento en línea.

Soy un span [Soy un enlace](#) **Soy un texto en negrita**

🔍 Observación: Todos los elementos aparecen en la misma línea y solo ocupan el ancho de su contenido.

◆ Elementos en Línea-Bloque (display: inline-block)

El modo **inline-block** combina lo mejor de inline y block:

Se comporta como un elemento en línea, es decir, se coloca junto a otros elementos en la misma línea. Permite definir width y height, igual que los elementos de bloque.

📌 Características:

- Se coloca en línea con otros elementos (como inline).
- Permite definir width y height (como block).
- Se usa para crear botones, tarjetas y elementos responsivos sin romper la línea.



IT Academy

by KIBERNUM