

# 31 CSS: Adición de Márgenes Horizontales

A diferencia de los márgenes verticales ( `margin-top` , `margin-bottom` ) que pueden colapsar entre sí bajo ciertas condiciones, los márgenes horizontales ( `margin-left` , `margin-right` ) tienen un comportamiento mucho más directo y predecible: **siempre se suman**.

## ¿Cómo Funcionan los Márgenes Horizontales?

- `margin-left` : Añade espacio en blanco al lado izquierdo de un elemento, fuera de su borde.
- `margin-right` : Añade espacio en blanco al lado derecho de un elemento, fuera de su borde.

Estos márgenes empujan a los elementos adyacentes (o al borde del contenedor padre) horizontalmente.

### La Clave: ¡No Hay Colapso!

Cuando dos elementos están uno al lado del otro en el eje horizontal (por ejemplo, dos elementos `display: inline-block` , o elementos flotantes, o ítems en un contenedor Flex/Grid), sus márgenes horizontales adyacentes **se suman** para determinar el espacio total entre ellos.

- **Ejemplo:**

- **HTML:**

```
<div class="contenedor">
  <span class="caja caja1">Caja 1</span><!-- Sin espacio entre spans para evitar espacio extra →
  <span class="caja caja2">Caja 2</span>
</div>
```

- **CSS:**

```
.contenedor {
  border: 1px solid grey;
  padding: 10px;
}

.caja {
  display: inline-block; /* Para que estén en línea */
  width: 100px;
  height: 50px;
  background: lightgrey;
  border: 1px solid black;
}

.caja1 {
  margin-right: 30px; /* Margen derecho de 30px */
}

.caja2 {
```

```
margin-left: 20px; /* Margen izquierdo de 20px */
}
```

- **Resultado:** El espacio total entre la Caja 1 y la Caja 2 será la **suma** de `margin-right` de la Caja 1 y `margin-left` de la Caja 2. Es decir,  $30\text{px} + 20\text{px} = 50\text{px}$ . Este es el comportamiento opuesto al colapso de márgenes verticales.

### Impacto en el Layout y Ancho Total:

Los márgenes horizontales contribuyen al espacio total que un elemento intenta ocupar en la línea o dentro de su contenedor. Recuerda que el espacio horizontal total ocupado por un elemento (considerando el modelo `box-sizing: content-box` por defecto) es:

`margin-left` + `border-left` + `padding-left` + `width` + `padding-right` + `border-right` + `margin-right`

Si usas `box-sizing: border-box`, el cálculo del espacio que ocupa la caja visible (`width`) ya incluye `padding` y `border`, pero los márgenes (`margin-left`, `margin-right`) siguen estando fuera y se suman a ese `width` para determinar el espacio total ocupado.

### Uso del Valor `auto` :

El valor `auto` tiene un significado especial para los márgenes horizontales en elementos de bloque:

- Cuando se aplica a `margin-left` o `margin-right` (o ambos) en un **elemento de nivel de bloque con un `width` definido**, el navegador calcula automáticamente el margen para ocupar el espacio horizontal disponible en el contenedor.
- **Caso Común (Centrado):** Si estableces `width` a un elemento de bloque y luego pones `margin-left: auto;` y `margin-right: auto;`, el navegador distribuirá el espacio sobrante equitativamente entre ambos márgenes, resultando en el centrado horizontal del elemento dentro de su contenedor. (Este tema se verá en detalle en "Centrado de Cajas").

```
.elemento-centrado {
width: 80%; /* O un ancho fijo como 500px */
margin-left: auto;
margin-right: auto;
/* ... otros estilos ... */
}
```

### Márgenes Negativos Horizontales:

Al igual que los verticales, los márgenes horizontales pueden tener valores negativos. Un `margin-left` negativo tirará del elemento hacia la izquierda (potencialmente superponiéndose con elementos anteriores o saliéndose del contenedor), y un `margin-right` negativo tirará de los elementos *siguientes* hacia la izquierda, acercándolos o haciendo que se superpongan con el elemento actual. (Los márgenes negativos se tratarán en un punto posterior).