

32 CSS: Centrado de Cajas (Centering Elements)

Centrar elementos, tanto horizontal como verticalmente, es una tarea muy común en el diseño web. CSS ofrece diferentes métodos para lograr el centrado, y la técnica adecuada dependerá del tipo de elemento que quieres centrar (bloque, en línea) y de la dirección (horizontal, vertical).

1. Centrado Horizontal de Elementos de Bloque

Este es uno de los casos de uso más frecuentes: centrar un contenedor principal, una tarjeta, o cualquier elemento de nivel de bloque dentro de su elemento padre.

- **Método Clásico:** `margin: auto;`
 - **Cómo funciona:** Cuando aplicas `margin-left: auto;` y `margin-right: auto;` a un elemento de bloque, le dices al navegador que calcule automáticamente los márgenes izquierdo y derecho. Si el elemento tiene un ancho definido (`width` o `max-width`) que es menor que el ancho de su contenedor padre, el navegador distribuirá el espacio horizontal restante a partes iguales entre los márgenes izquierdo y derecho, centrando así el elemento.
 - **Condiciones:**
 1. El elemento debe ser de **nivel de bloque** (`display: block;` o un elemento que se comporte como tal por defecto, como `<div>` , `<p>` , `<section>` , `<article>` , etc.).
 2. El elemento debe tener un **ancho explícito** (`width` o `max-width`) menor que el de su contenedor. Si no se especifica un ancho, un elemento de bloque ocupará por defecto todo el ancho disponible, y no habrá espacio para distribuir con `margin: auto;` .
 - **Sintaxis (Forma corta):**

```
.contenedor-centrado {  
  width: 80%; /* O un valor fijo como 600px */  
  max-width: 900px; /* Buena práctica para limitar el ancho en pantallas grandes */  
  margin-left: auto;  
  margin-right: auto;  
  /* Forma corta y común: */  
  /* margin: 0 auto; */ /* (0 para top/bottom, auto para left/right) */  
  /* O si ya tienes márgenes verticales: */  
  /* margin-top: 20px; */  
  /* margin-bottom: 20px; */  
  /* margin-left: auto; */  
  /* margin-right: auto; */  
  
  background-color: lightblue; /* Para visualizar */  
  padding: 20px;  
  border: 1px solid blue;  
}
```

- **Ejemplo HTML:**

```
<body style="border: 1px dashed grey;"> <!-- Contenedor padre →
<div class="contenedor-centrado">
  Este div está centrado horizontalmente.
</div>
</body>
```

- **Resultado:** El `div.contenedor-centrado` aparecerá centrado horizontalmente dentro del `<body>` (o cualquier otro bloque contenedor en el que se encuentre).

2. Centrado Horizontal de Contenido (Texto, Inline, Inline-Block)

Si lo que quieres es centrar el contenido *dentro* de un elemento de bloque (como texto, imágenes, botones `inline-block`), el método es diferente.

- **Método:** `text-align: center;`

- **Cómo funciona:** Esta propiedad se aplica al **contenedor padre** y afecta a todos los elementos **en línea** (`inline`) e **inline-block** (`inline-block`) que contiene, así como al texto.
- **Sintaxis:**

```
.contenedor-con-contenido-centrado {
  text-align: center; /* Aplicado al padre */
  border: 1px solid green;
  padding: 15px;
  width: 300px; /* Ancho del contenedor padre */
  margin: 20px auto; /* Centramos el propio contenedor también */
}

.boton-ejemplo {
  display: inline-block; /* Para que text-align le afecte */
  padding: 10px 20px;
  background-color: darkcyan;
  color: white;
  text-decoration: none;
  border-radius: 5px;
  margin-top: 10px; /* Espacio superior */
}
```

- **Ejemplo HTML:**

```
<div class="contenedor-con-contenido-centrado">
  <p>Este texto estará centrado.</p>
   <br>
  <a href="#" class="boton-ejemplo">Botón Centrado</a>
</div>
```

- **Resultado:** El texto, la imagen (que es inline por defecto) y el botón (`inline-block`) dentro del `div` estarán centrados horizontalmente respecto a los bordes del `div`.

3. Centrado Vertical

El centrado vertical ha sido históricamente más complejo que el horizontal usando técnicas tradicionales.

- **Técnicas Tradicionales (Menos comunes hoy en día):**

- **Padding:** Si conoces la altura del contenedor, puedes usar `padding-top` y `padding-bottom` iguales. Funciona bien para contenido de una sola línea o altura fija.
- **Line-height:** Para **una sola línea de texto**, puedes centrarla verticalmente estableciendo la propiedad `line-height` del contenedor igual a su `height`.

```
.caja-texto-vertical-centrado {  
  height: 50px;  
  line-height: 50px; /* Igual a height */  
  text-align: center; /* Centrado horizontal también */  
  border: 1px solid orange;  
  white-space: nowrap; /* Evita que el texto salte de línea */  
  overflow: hidden; /* Oculta si el texto es muy largo */  
  text-overflow: ellipsis; /* Muestra '...' si se corta */  
}
```

- **Display Table:** Usar `display: table` en el padre y `display: table-cell` con `vertical-align: middle;` en el hijo. Funciona bien pero cambia el comportamiento de los elementos a uno similar al de las tablas.
- **Posicionamiento Absoluto y Transformación:** Una técnica potente pero que requiere entender el posicionamiento (`position: absolute; top: 50%; left: 50%; transform: translate(-50%, -50%);`). La veremos más adelante cuando hablemos de posicionamiento.

- **Métodos Modernos: Flexbox y Grid (Preferidos)**

Aunque probablemente tengas secciones dedicadas a Flexbox y Grid más adelante, es importante mencionar aquí que son las **soluciones modernas y más eficientes** para el centrado, especialmente el vertical.

- **Con Flexbox:**

```
.contenedor-flex-padre {  
  display: flex;  
  justify-content: center; /* Centrado Horizontal */  
  align-items: center; /* Centrado Vertical */  
  height: 200px; /* Necesita una altura para centrar verticalmente */  
  border: 1px solid purple;  
}  
.hijo-flex {  
  background-color: lightpink;  
  padding: 20px;  
}
```

HTML: `<div class="contenedor-flex-padre"><div class="hijo-flex">Centrado!</div></div>`

- **Con CSS Grid:**

```
.contenedor-grid-padre {  
  display: grid;  
  place-items: center; /* Centra horizontal y verticalmente a la vez */  
  height: 200px; /* Necesita una altura */  
  border: 1px solid teal;  
}  
.hijo-grid {  
  background-color: lightyellow;  
  padding: 20px;  
}
```

HTML: `<div class="contenedor-grid-padre"><div class="hijo-grid">Centrado!</div></div>`

Conclusión:

- Para **centrar horizontalmente elementos de bloque**: usa `width` (o `max-width`) y `margin: auto;`.
 - Para **centrar horizontalmente contenido en línea/inline-block**: usa `text-align: center;` en el elemento padre.
 - Para **centrar verticalmente** (y también horizontalmente de forma robusta): las herramientas modernas **Flexbox** (`display: flex; justify-content: center; align-items: center;`) y **CSS Grid** (`display: grid; place-items: center;`) son las más recomendadas y flexibles. Las técnicas tradicionales existen pero suelen ser más limitadas o complejas.
-