

Centrado en CSS: una guía completa



Chris Coyier en 2 de septiembre de 2014 (Actualizado en 2 de agosto de 2021)

Centrar las cosas en CSS es el ejemplo de las quejas de CSS. *¿Por qué tiene que ser tan difícil?* Ellos se burlan. Creo que el problema no es que sea difícil de hacer, sino que hay tantas formas diferentes de hacerlo, dependiendo de la situación, es difícil saber cuál alcanzar.

Así que hagamos un árbol de decisiones y, con suerte, lo haremos más fácil.

necesito centrarme...

▼ (#aa-horizontally) Horizontalmente

▼ ¿Son elementos en línea o en línea-* (como texto o enlaces)?

Puede centrar elementos en línea horizontalmente, dentro de un elemento principal a nivel de bloque, con solo:

```
.center {  
  text-align: center;  
}
```

CSS

Embedded Pen Here

Esto funcionará para inline, inline-block, inline-table, inline-flex, etc.

▼ ¿Es un elemento de nivel de bloque?

Puede centrar un elemento a nivel de bloque dándole `margin-left` `margin-right` `auto` (y tiene un `set width`, de lo contrario, sería de ancho completo y no necesitaría centrado). Eso a menudo se hace con taquigrafía como esta:

```
.center-me {  
  margin: 0 auto;  
}
```

CSS

Embedded Pen Here

Esto funcionará sin importar el ancho del elemento de nivel de bloque que esté centrando o el padre.

Tenga en cuenta que no puede `float` un elemento al centro. Aunque hay un truco. (<https://css-tricks.com/float-center/>)

▼ ¿Hay más de un elemento de nivel de bloque?

Si tiene dos o más elementos a nivel de bloque que deben estar centrados horizontalmente *en una fila*, es probable que le resulte mejor si los convierte en un `display` tipo diferente. Aquí hay un ejemplo de cómo hacerlos `inline-block` y un ejemplo de `flexbox`:

Embedded Pen Here

A menos que quiera decir que tiene varios elementos de nivel de bloque apilados uno encima del otro, en cuyo caso la técnica de margen automático todavía está bien:

Embedded Pen Here

▼ (#aa-vertically) Verticalmente

El centrado vertical es un poco más complicado en CSS.

▼ ¿Son elementos en línea o en línea-* (como texto o enlaces)?

▼ ¿Es una sola línea?

A veces, los elementos en línea/texto pueden aparecer centrados verticalmente, solo porque hay un relleno igual encima y debajo de ellos.

```
.link {  
  padding-top: 30px;  
  padding-bottom: 30px;  
}
```

CSS

Embedded Pen Here

Si el relleno no es una opción por alguna razón, y está tratando de centrar un texto que sabe que no se ajustará, hay un truco para hacer `line-height` que el texto sea igual a la altura `center`.

```
.center-text-trick {  
  height: 100px;
```

CSS

```
line-height: 100px;  
white-space: nowrap;  
}
```

Embedded Pen Here

▼ ¿Son varias líneas?

El relleno igual en la parte superior e inferior también puede dar el efecto centrado para varias líneas de texto, pero si eso no funciona, tal vez el elemento en el que se encuentra el texto puede ser una celda de tabla, ya sea literalmente o hecho para comportarse como uno con CSS. La [vertical-align](https://css-tricks.com/almanac/properties/v/vertical-align/) (<https://css-tricks.com/almanac/properties/v/vertical-align/>) propiedad maneja esto, en este caso, a diferencia de lo que normalmente hace, que es manejar la alineación de los elementos alineados en una fila. ([Más sobre eso.](https://css-tricks.com/what-is-vertical-align/) (<https://css-tricks.com/what-is-vertical-align/>))

Embedded Pen Here

Si algo parecido a una mesa está descartado, ¿quizás podrías usar flexbox? Se puede hacer que un solo hijo flexible se centre en un padre flexible con bastante facilidad.

```
.flex-center-vertically {  
  display: flex;  
  justify-content: center;  
  flex-direction: column;  
  height: 400px;  
}
```

CSS

Embedded Pen Here

Recuerde que solo es realmente relevante si el contenedor principal tiene una altura fija (px, %, etc.), razón por la cual el contenedor aquí tiene una altura.

Si ambas técnicas están descartadas, puede emplear la técnica del "elemento fantasma", en la que se coloca un pseudoelemento de altura completa dentro del contenedor y el texto se alinea verticalmente con él.

```
.ghost-center {  
  position: relative;  
}  
.ghost-center::before {  
  content: " ";  
  display: inline-block;  
  height: 100%;  
  width: 1%;  
  vertical-align: middle;  
}  
.ghost-center p {  
  display: inline-block;  
  vertical-align: middle;  
}
```

CSS

Embedded Pen Here

▼ ¿Es un elemento a nivel de bloque?

▼ ¿Conoces la altura del elemento?

Es bastante común *no* saber la altura en el diseño de la página web, por muchas razones: si el ancho cambia, el reflujo del texto puede cambiar la altura. La variación en el estilo del texto puede cambiar la altura. La variación en la cantidad de texto puede cambiar la altura. Los elementos con una relación de aspecto fija, como las imágenes, pueden cambiar de altura cuando se les cambia el tamaño. Etc.

Pero si conoce la altura, puede centrar verticalmente como:

```
.parent {  
  position: relative;  
}  
.child {  
  position: absolute;  
  top: 50%;  
  height: 100px;  
  margin-top: -50px; /* account for padding and border if not using box-sizing: border-box; */  
}
```

Embedded Pen Here

▼ ¿Es el elemento de altura desconocida?

Todavía es posible centrarlo empujándolo hacia arriba a la mitad de su altura después de golpearlo hasta la mitad:

```
.parent {  
  position: relative;  
}  
.child {  
  position: absolute;  
  top: 50%;  
  transform: translateY(-50%);  
}
```

Embedded Pen Here

▼ ¿Te importa si el elemento estira la altura del contenedor?

Si no es así, solo necesita que el contenido esté centrado verticalmente; usar tablas o una pantalla CSS para convertir elementos en tablas puede ser la solución.

Embedded Pen Here

▼ ¿Puedes usar flexbox?

No es una gran sorpresa, esto es mucho más fácil en flexbox (<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>).

```
.parent {  
  display: flex;  
  flex-direction: column;  
  justify-content: center;  
}
```

CSS

Embedded Pen Here

También puede centrarse en flexbox usando `margin: auto;` el elemento secundario.

▼ (#aa-both-horizontally-vertically) Tanto horizontal como verticalmente

Puede combinar las técnicas anteriores de cualquier manera para obtener elementos perfectamente centrados. Pero encuentro que esto generalmente cae en tres campos:

▼ ¿El elemento es de ancho y alto fijos?

El uso de márgenes negativos equivalentes a la mitad de ese ancho y alto, después de haberlo posicionado absolutamente al 50 %/50 %, lo centrará con una gran compatibilidad entre navegadores:

```
.parent {  
  position: relative;  
}  
  
.child {  
  width: 300px;  
  height: 100px;  
  padding: 20px;  
  
  position: absolute;  
  top: 50%;  
  left: 50%;  
  
  margin: -70px 0 0 -170px;  
}
```

CSS

Embedded Pen Here

▼ ¿Es el elemento de anchura y altura desconocidas?

Si no conoce el ancho o el alto, puede usar la propiedad de transformación y una traducción negativa del 50% en ambas direcciones (se basa en el ancho/alto actual del elemento) para centrar:

```
.parent {  
  position: relative;  
}  
.child {  
  position: absolute;  
  top: 50%;  
  left: 50%;  
  transform: translate(-50%, -50%);  
}
```

CSS

Embedded Pen Here

▼ ¿Puedes usar flexbox?

Para centrar en ambas direcciones con flexbox, debe usar dos propiedades de centrado:

```
.parent {  
  display: flex;  
  justify-content: center;
```

CSS

```
align-items: center;
}
```

Embedded Pen Here

▼ ¿Puedes usar la cuadrícula?

Este es solo un pequeño truco (enviado por Lance Janssen) que funcionará bastante bien para un elemento:

```
body, html {
  height: 100%;
  display: grid;
}
span { /* thing to center */
  margin: auto;
}
```

CSS

Embedded Pen Here

▼ (#aa-conclusion) Conclusión

Puedes centrar totalmente las cosas en CSS.