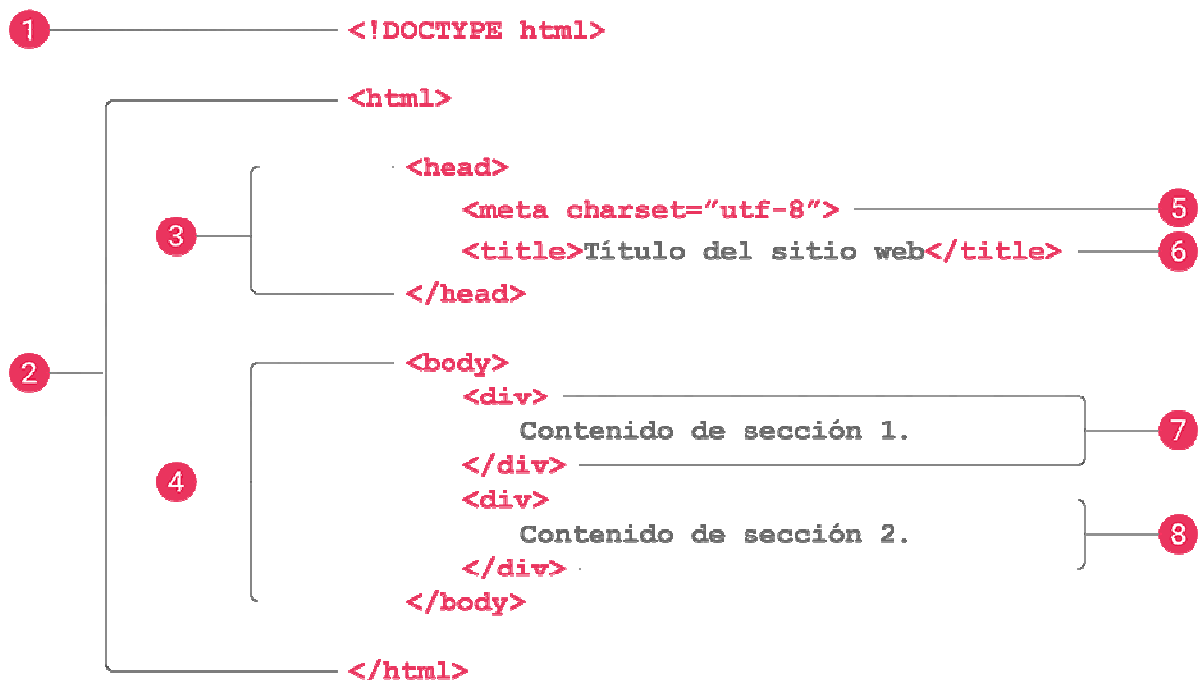


Diferencia entre PX, EM, REM, %, VW y VH

Estructura básica de un documento HTML

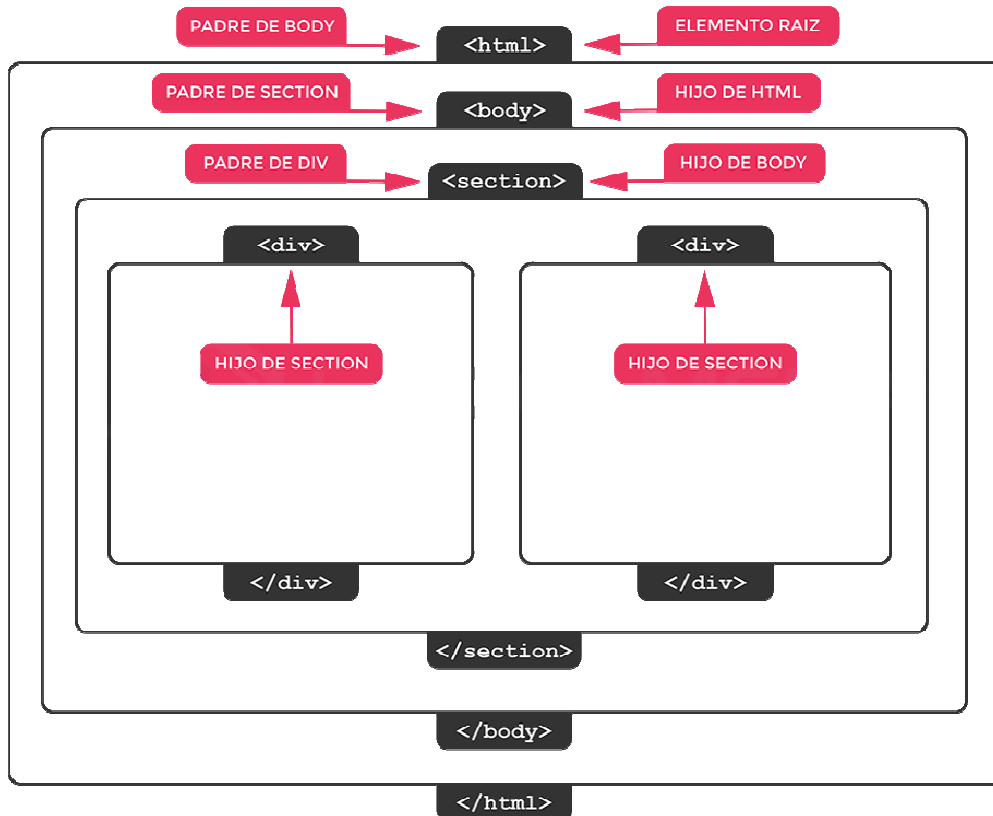
Tal como decía Carl Sagan, *«Si se quiere hacer una tarta de manzana desde cero, hay que inventar primero el universo»*, y sí, para poder entender unidades relativas de Elementor, qué es un elemento padre o el contenedor raíz, etc., primero tenemos que saber como se estructura de forma básica una página web en lenguaje HTML.

Aquí vemos la estructura mínima de un documento HTML (Lenguaje de Marcas de Hipertexto), el lenguaje HTML se compone de etiquetas que se encuentran delimitadas por los signos menor que < y mayor que >, ejemplo <html>, salvo excepciones como como la primera etiqueta <!DOCTYPE html> que es absoluta, todas las etiquetas tienen apertura y cierre, ejemplo <h1>Encabezado</h1>, el cierre se distingue porque la apertura de la etiqueta incluye </ además del “menor que” la barra diagonal también llamada **slash** /.



1. La primera línea del ejemplo es un elemento absoluto (no tiene etiqueta de cierre); es una **declaración de tipo de documento** (también llamada **declaración DOCTYPE**) que identifica este documento como un documento HTML5 y permite a los navegadores modernos saber que deben interpretar el documento como está escrito de acuerdo con la especificación HTML5.
2. Todo el documento está contenido dentro de un **html**. El **html** se denomina **elemento raíz** porque contiene todos los elementos del documento y no puede estar incluido en ningún otro elemento.
3. Dentro del **html**, el documento se divide en un encabezado (**head**) y un cuerpo (**body**). El **head** contiene información descriptiva sobre el documento en sí, como su título, las hojas de estilo que utiliza, los scripts y otros tipos de información "meta".
4. El **body** es el cuerpo de la página, es el elemento contiene todo lo que queremos que aparezca en la ventana del navegador.
5. Los **meta** elementos dentro de la etiqueta **head** proporcionan información sobre el documento en sí. Un meta elemento puede usarse para proporcionar todo tipo de información, pero en el caso de este ejemplo especifica la codificación de caracteres (la colección estandarizada de letras, números y símbolos) que se usa en el documento.
6. Dentro del **head** también está la etiqueta **title** obligatorio. Según la especificación HTML, cada documento debe contener un título descriptivo. Es este título el que se muestran en la pestaña del navegador y el que mostrará Google en el listado de búsqueda como título de esta página.

7. Dentro del bloque body podemos crear tantas secciones como queramos, la etiqueta **div**, esta etiqueta permite agrupar varios elementos de bloque (párrafos, encabezados, listas, etc). Estas secciones son contenedores.
8. Es otra sección igual que la anterior.



Como puedes ver en la *Figura*, todas estas etiquetas son como una especie de contenedores, la raíz o la principal es la etiqueta **html** que contiene a todas las demás, dentro de **html** tenemos el **body**, y a su vez dentro del **body** tenemos otros **div**. Así que podemos decir que el **elemento raíz** es **html**, **body** es hijo de **html** y los **div** son hijos de **body**.

Unidades de medida de Elementor

Al diseñar con Elementor, puedes ver que algunos elementos tienen opciones de tamaño, lo que te permite elegir **PX**, **EM**, **REM**, **%**, **VW** o **VH**. Pero, ¿qué significan realmente esas unidades de medida y cuándo deberías usar una u otra?

Empecemos con lo básico, en CSS, puedes especificar tamaños o longitudes de elementos utilizando varias unidades de medida. Las unidades de medida que encontrarás en algunas opciones de los widgets de Elementor incluyen *PX*, *EM*, *REM*, *%*, *VW* y *VH*, aunque hay varias más disponibles en CSS. No todos los elementos de Elementor ofrecerán todas estas unidades. Elementor solo presenta las opciones que tienen más sentido para el elemento dado.

El aspecto más importante para aprender sobre las diferentes unidades es que algunas unidades, como PX, son **unidades absolutas** y otras son **unidades relativas**.

Unidades absolutas

PX: los píxeles (px) se consideran unidades absolutas, aunque son relativas al DPI y la resolución del dispositivo de visualización. Pero en el dispositivo en sí, la unidad PX es fija y no cambia en función de ningún otro elemento. El uso de PX puede ser problemático para los sitios web responsivos, pero son útiles para mantener un tamaño consistente para algunos elementos. Si tienes elementos a los que no se les debe cambiar el tamaño, entonces usar PX es una buena opción.

Unidades relativas

- **EM**: relativo al elemento padre
- **REM**: relativo al elemento raíz (etiqueta HTML y en su defecto el tamaño que tenga configurado el navegador)
- **%**: relativo al elemento padre
- **VW**: relativo al ancho de la ventana gráfica
- **VH**: relativo a la altura de la ventana gráfica

A diferencia de PX, las unidades relativas como %, EM y REM se adaptan mejor al diseño responsivo y también ayudan a **cumplir con los estándares de accesibilidad**. Las unidades relativas escalan mejor en diferentes dispositivos porque pueden escalar hacia arriba y hacia abajo según el tamaño de otro elemento. Veamos un ejemplo sencillo, en la mayoría de los navegadores, el tamaño de fuente predeterminado es **16px**. Las unidades relativas calculan el tamaño a partir de esta base. Si cambia esa base estableciendo un tamaño de base para la etiqueta HTML a través de CSS, eso se convierte en la base para calcular las

unidades relativas en el resto de la página. Del mismo modo, si un usuario ajusta su tamaño de fuente, eso se convierte en la base para calcular las unidades relativas.

Entonces, ¿qué significan estas unidades cuando se trata del valor predeterminado de 16px?

El número que especifique multiplicará ese número por el tamaño predeterminado.

$1\text{em} = 16\text{px} (1 * 16)$

$2\text{em} = 32\text{px} (2 * 16)$

$0.5\text{em} = 8\text{px} (.5 * 16)$

$1\text{rem} = 16\text{px}$

$2\text{rem} = 32\text{px}$

$0.5\text{rem} = 8\text{px}$

$100\% = 16\text{px}$

$200\% = 32\text{px}$

$50\% = 8\text{px}$

Bien, genial, pero ¿qué pasa si el usuario cambia el tamaño predeterminado en su navegador? Debido a que se trata de unidades relativas, los valores de tamaño final se basarán en el nuevo tamaño base. Aunque el valor predeterminado del navegador suele ser 16px, si el usuario lo cambiara a 14px, los tamaños calculados terminarían siendo:

$1\text{em} = 14\text{px} (1 * 14)$

$2\text{em} = 28\text{px} (2 * 14)$

$0.5\text{em} = 7\text{px} (.5 * 14)$

$1\text{rem} = 14\text{px}$

2rem = 28px

0.5rem = 7px

100% = 14 px

200% = 28 px

50% = 7px

Esto le da al usuario la libertad de ajustar el tamaño de fuente predeterminado de su navegador, mientras mantiene la escala relativa de cada elemento que ha especificado.

¿Cuál es la diferencia entre EM y REM?

Mirando el gráfico de arriba, muestra EM y REM con exactamente el mismo aspecto. Entonces, ¿en qué se diferencian?

En pocas palabras, difieren según la herencia. Como se mencionó, **REM** se basa en el **elemento raíz** (HTML) y en su defecto si no está establecido el que tenga el navegador del usuario, que **por defecto está a 16px**. Cada elemento hijo que usa **REM** utilizará el tamaño de la raíz HTML como punto de cálculo, independientemente de si un elemento padre tiene o no especificados tamaños diferentes.

EM, por otro lado, se basa en el tamaño de fuente del elemento principal. Si un elemento principal tiene un tamaño diferente al del elemento raíz, el cálculo de **EM** se basará en el elemento principal y no en el elemento raíz. Esto significa que los elementos anidados que usan **EM** para dimensionar a veces pueden terminar teniendo un tamaño que no anticipaba. Por otro lado, te brinda un control más detallado si lo necesitas para especificar el tamaño de un área particular de una página.

Entonces, ¿qué pasa con %, VW y VH? ¿De qué se tratan?

Mientras que **PX**, **EM** y **REM** se utilizan principalmente para el tamaño de fuente, **%**, **VW** y **VH** se utilizan principalmente para márgenes, relleno, espaciado y anchos / alturas. Para reiterar, **VH** significa “viewport height” (altura de la ventana gráfica), que es la altura de la pantalla visible. **100VH** representaría el 100% de la altura de la ventana gráfica o la altura completa de la pantalla. Y, por supuesto, **VW** significa “viewport width” (ancho de la ventana gráfica), que es el ancho de la pantalla visible. **100VW** representaría el 100% del ancho de la ventana gráfica o el ancho completo de la pantalla. **%** refleja un porcentaje del tamaño del **elemento principal**, independientemente del tamaño de la ventana gráfica.

Veamos algunos ejemplos de dónde Elementor ofrece opciones de **%**, **VW** y **VH**.

- **Anchos de columna** : si editas el diseño de una columna de Elementor, notará que solo hay una unidad de tamaño de ancho disponible: **%**. Los anchos de columna solo funcionan bien y de manera receptiva cuando se usan porcentajes, por lo que no se ofrece ninguna otra opción.
- **Márgenes**: los márgenes de una sección se pueden especificar en **PX** o **%**. Por lo general, es preferible usar **%** para garantizar que los márgenes no sean más grandes que el contenido al reducir la escala para un dispositivo móvil, por ejemplo. Al utilizar un porcentaje del ancho del dispositivo, sus márgenes se mantendrán en relación con el tamaño del contenido, lo que casi siempre es preferible.
- **Relleno**: el relleno de una sección se puede especificar en **PX**, **EM** o **%**. Al igual que con los márgenes, a menudo es preferible utilizar **EM** o **%** para que el relleno permanezca relativo a medida que aumenta el tamaño de la página.
- **Tamaño de fuente**: si editas la tipografía de un elemento, como un encabezado, verás cuatro opciones: **PX**, **EM**, **REM** y **VH**.

¿Alguna vez ha creado un encabezado grande y has admirado lo bien que se veía en el escritorio, solo para darse cuenta de que era demasiado grande en el móvil?

La clave para resolver esto de manera elegante es usar **EM**, **REM** o **VW** en lugar de **PX**. Lo que elijas depende de tu situación particular. Por lo general, elijo **EM** porque quiero que el tamaño sea relativo al padre del encabezado. Pero si prefiere que el

tamaño sea relativo al tamaño de la raíz (HTML), elije REM en su lugar. O bien, puede configurarlo para que sea relativo al ancho de la ventana gráfica (VW) si eso funciona mejor para tu caso.

Ten en cuenta que también puedes establecer valores PX de tamaño de fuente específicos por dispositivo utilizando los iconos de dispositivo para especificar un tamaño para escritorio, tableta y dispositivo móvil. Pero eso todavía impone límites a la capacidad de respuesta y la accesibilidad, así que tenlo en cuenta si eliges PX.

Más sobre VW y VH

Las unidades de la ventana gráfica representan un porcentaje de la ventana gráfica del navegador actual (tamaño actual del navegador). Si bien es similar a las unidades de porcentaje, hay una diferencia. Las unidades de la ventana gráfica se calculan como un porcentaje del tamaño actual de la ventana gráfica del navegador. Las unidades de porcentaje, por otro lado, se calculan como un porcentaje del **elemento principal**, que puede ser diferente del tamaño de la ventana gráfica.

Consideremos un ejemplo de una ventana de visualización de pantalla móvil de 480px x 800px.

1 VW = 1% del ancho de la ventana gráfica (o 4,8 px)

50 VW = 50% del ancho de la ventana gráfica (o 240 px)

1 VH = 1% de la altura de la ventana gráfica (o 8px)

50 VH = 50% de la altura de la ventana gráfica (o 400 px)

Si el tamaño de la ventana gráfica cambia, el tamaño del elemento cambia respectivamente.

¿Cuándo debería usar una unidad sobre otra?

En última instancia, no existe una respuesta perfecta para esta pregunta. En general, a menudo es mejor elegir una de las unidades relativas en lugar de PX para que tu página web tenga la mejor oportunidad de representar un diseño responsivo. Sin embargo, elije **PX** si necesitas asegurarte de que un elemento nunca cambie de tamaño en ningún punto de interrupción y permanezca igual independientemente de si un usuario ha elegido un tamaño predeterminado diferente. Las unidades PX garantizan resultados consistentes incluso si eso no es ideal.

- **EM** es relativo al tamaño de fuente del elemento padre, por lo que si deseas escalar el tamaño del elemento en función del tamaño de su padre utiliza EM.
- **REM** es relativo al tamaño de fuente de la raíz (HTML) y si este no está establecido se usará el que tenga configurado el navegador, por lo que si deseas escalar el tamaño del elemento en función del tamaño de la raíz, sin importar cuál sea el tamaño principal, usa REM. Si has utilizado EM y estás encontrando problemas de tamaño debido a muchos elementos anidados, REM probablemente sea la mejor opción.
- **VW** es útil para crear elementos de ancho completo (100%) que llenan todo el ancho de la ventana gráfica. Por supuesto, puedes usar cualquier porcentaje del ancho de la ventana gráfica para lograr otros objetivos, como el 50% para la mitad del ancho, etc.
- **VH** es útil para crear elementos de altura completa (100%) que llenan toda la altura de la ventana gráfica. Por supuesto, puedes usar cualquier porcentaje de la altura de la ventana gráfica para lograr otros objetivos, como el 50% para la mitad de la altura, etc.
- **%** es similar a VW y VH, pero no es una longitud relativa al ancho o alto de la ventana gráfica. En cambio, es un porcentaje del ancho o alto del elemento padre. Las unidades de porcentaje suelen ser útiles para establecer el ancho de los márgenes, por ejemplo.
- Elementor facilita la elección de la opción que mejor se adapte a tu diseño. En última instancia, es tu elección.