

# 45 CSS: Unidades de medida

En CSS, las unidades de medida son esenciales para definir tamaños, distancias y proporciones en el diseño web. Estas pueden clasificarse en dos grandes grupos: **unidades absolutas** y **unidades relativas**. A continuación, se detallan las principales unidades de medida y cuándo es recomendable usar cada una.

---

## 1. Píxeles (px)

- **Definición:** El píxel es una unidad de medida absoluta que representa una pequeña porción de la pantalla. Un píxel equivale a **1/96 de una pulgada física**.
- **Uso común:** Se utiliza para dimensiones fijas, como anchos, alturas, márgenes y bordes.
- **Ejemplo:**

```
width: 100px;  
height: 50px;
```

- **Consideraciones:**
    - Es una unidad precisa y fácil de entender.
    - No es flexible ni adaptable; puede causar problemas en diseños responsivos si se abusa de ella.
- 

## 2. EMs

- **Definición:** Los `em` son unidades relativas basadas en el tamaño de fuente ( `font-size` ) del elemento al que se aplican o de su contenedor padre.
- **Uso común:** Útil para crear diseños proporcionales y jerárquicos, especialmente en tipografía y espaciado interno.
- **Ejemplo:**

```
font-size: 1.5em; /* 1.5 veces el tamaño de fuente del padre */  
padding: 1em; /* Igual al tamaño de fuente actual */
```

- **Consideraciones:**
    - Puede complicarse debido a la herencia. Si un elemento no tiene un tamaño de fuente definido, buscará el valor en sus ancestros.
    - Ideal para componentes escalables, como botones o tarjetas.
- 

## 3. REMs

- **Definición:** Los `rem` son similares a los `em`, pero están basados en el tamaño de fuente del elemento raíz ( `html` ), que por defecto es **16px** en la mayoría de navegadores.
- **Uso común:** Perfecto para mantener consistencia en todo el diseño, especialmente en tipografía y márgenes.

- **Ejemplo:**

```
font-size: 1.2rem; /* 1.2 veces el tamaño de fuente del html */  
margin: 2rem;
```

- **Consideraciones:**

- Más predecible que `em` porque no depende de la herencia.
- Recomendado para la mayoría de los casos, especialmente en diseño responsivo.

## 4. EXs

- **Definición:** Los `ex` están basados en la altura de la letra "x" de la fuente del elemento. Son poco utilizados en la práctica.

- **Uso común:** Se emplean en casos específicos donde se necesita alinear elementos con precisión vertical.

- **Ejemplo:**

```
line-height: 2ex;
```

- **Consideraciones:**

- Poco soporte y uso limitado en proyectos modernos.
- Generalmente se prefiere usar `em` o `rem`.

## 5. CHs

- **Definición:** Los `ch` están basados en la anchura del carácter "0" de la fuente del elemento.

- **Uso común:** Útil para controlar dimensiones de texto, como asegurar que un botón tenga suficiente espacio para un número específico de caracteres.

- **Ejemplo:**

```
width: 20ch; /* Ancho suficiente para 20 ceros */
```

- **Consideraciones:**

- Muy útil en diseño de formularios y componentes de interfaz.
- Proporciona un control preciso sobre el contenido textual.

## 6. Porcentajes (%)

- **Definición:** Los porcentajes son unidades relativas que toman como referencia el tamaño del contenedor padre.

- **Uso común:** Ampliamente utilizado en diseño responsivo para ajustar anchos, alturas y márgenes.

- **Ejemplo:**

```
width: 50%; /* La mitad del ancho del contenedor padre */
margin: 10%;
```

- **Consideraciones:**
  - Depende directamente del tamaño del contenedor padre.
  - Ideal para layouts fluidos y adaptativos.

## 7. Unidades del Viewport

- **Definición:** Las unidades del viewport están basadas en las dimensiones de la ventana del navegador. Incluyen:
  - **vw** : 1% del ancho del viewport.
  - **vh** : 1% de la altura del viewport.
  - **vmin** : El menor entre **vw** y **vh** .
  - **vmax** : El mayor entre **vw** y **vh** .
- **Uso común:** Diseños full-screen o elementos que deben adaptarse al tamaño de la pantalla.
- **Ejemplo:**

```
width: 100vw; /* Ocupa todo el ancho de la pantalla */
height: 50vh; /* Ocupa la mitad de la altura de la pantalla */
```

- **Consideraciones:**
  - Muy útiles para hero images, sliders y layouts responsivos.
  - Pueden causar problemas si no se manejan correctamente en dispositivos móviles.

## ¿Cuándo usar cada unidad de medida?

### 1. Píxeles (px):

- Para dimensiones fijas, como bordes, iconos o imágenes.
- No recomendado para diseño responsivo.

### 2. EMs:

- Para diseño jerárquico y escalable dentro de un componente.
- Ideal cuando necesitas que los elementos crezcan o disminuyan en relación con su contexto.

### 3. REMs:

- Para tipografía y espaciado global.
- Recomendado en la mayoría de los casos debido a su consistencia y facilidad de uso.

### 4. EXs:

- En casos muy específicos donde la altura de la fuente es crítica.

- Rara vez utilizado en proyectos modernos.

#### 5. CHs:

- Para controlar el ancho de elementos basados en caracteres, como botones o campos de texto.
- Muy útil en interfaces de usuario.

#### 6. Porcentajes (%):

- Para layouts fluidos y responsivos.
- Ideal cuando el tamaño depende del contenedor padre.

#### 7. Unidades del Viewport:

- Para elementos que deben adaptarse al tamaño de la pantalla.
- Perfecto para hero sections, sliders y layouts full-screen.

---

Cada unidad de medida tiene su propósito y escenario ideal. Para diseño responsivo, las unidades relativas como `rem`, `%`, y las unidades del viewport son las más adecuadas. Los `em` son útiles para componentes escalables, mientras que los `ch` y `ex` tienen aplicaciones específicas pero limitadas. Finalmente, los píxeles siguen siendo útiles para detalles precisos, aunque deben usarse con moderación en diseño moderno.