

INTRODUCCIÓN A CSS

2. Unidades de medida y colores

CONTENIDO

1. Introducción	1
2. Unidades de medida	1
2.1. Unidades relativas	1
2.2. Unidades absolutas	3
3. Colores	4
3.1. Palabras clave	4
3.2. RGB	4
3.3. Colores del sistema	5

1. Introducción

Muchas declaraciones de CSS necesitan que se especifique una medida y/o colores para aplicar un estilo. A continuación veremos de qué modos es posible especificar distintas unidades de medida y colores en las declaraciones.

2. Unidades de medida

Las medidas en CSS se emplean para definir parámetros de aspecto tales como el alto, ancho o los márgenes de un elemento del documento. Todas las medidas se indican mediante un valor numérico (entero o decimal) seguido de una unidad de medida (sin dejar espacios en blanco entre el número y la unidad). Si no se especifica la unidad de medida, la propiedad es ignorada y no se aplica.

Existen dos tipos de unidades de medida:

- **Unidades de medida relativas:** las medidas relativas definen su valor en relación con otra medida, por lo que para obtener su valor real se debe realizar alguna operación con el valor indicado.
- **Unidades de medida absolutas:** establecen de forma completa el valor de una medida por lo que su valor real es directamente el valor indicado.

2.1. Unidades relativas

Las unidades relativas son más flexibles que las absolutas ya que permiten adaptarse fácilmente a los distintos medios. A continuación se muestra la lista de unidades de medida relativas y la referencia que toma para determinar su valor real:

- ***em***, es una unidad relativa al tamaño de letra empleado. 1em representa el ancho de la letra 'M' en la fuente utilizada. Para el caso de fuentes TrueType, 1em equivale al tamaño de la fuente utilizada.
- ***ex***, es una unidad relativa a la altura de la letra 'x' en la fuente que se esté utilizando.
- ***px***, es una unidad relativa respecto a la pantalla del usuario. 1px equivale a un píxel en la pantalla del usuario.
- **Porcentajes (%)**, es una unidad relativa formada por un valor numérico seguido del símbolo '%'.
Ejemplo: `width: 50%;`

Por ejemplo, para indicar que el tamaño de la fuente utilizada en el documento sea un 90% del tamaño de la fuente predeterminada se utiliza la siguiente regla:

```
body { font-size: 0.9em; }
```

Asumiendo que el tamaño de la fuente por defecto en la mayoría de navegadores es de 12 puntos, el valor de 0.9em sería igual a $0.9 \times 12 = 10.8$ puntos.

De manera equivalente, podría haberse expresado la regla anterior mediante porcentajes:

```
body { font-size: 90%; }
```

El funcionamiento de la unidad **ex** es idéntico a **em**, excepto que en este caso, la referencia es la altura de la letra 'x' minúscula.

Las medidas indicadas en píxeles también se consideran relativas, ya que dependen de la resolución del dispositivo. Un pixel que se percibe aislado en una pantalla a baja resolución es reescalado a un conjunto de píxeles en una pantalla a alta definición o en una impresora.

Las distintas unidades se pueden mezclar entre los diferentes elementos de una página, como en el siguiente ejemplo:

```
body { font-size: 10px; }  
h1   { font-size: 2.5em; }
```

En este ejemplo se establece un tamaño de letra base de 10 píxeles para toda la página. A continuación, se asigna un tamaño de 2.5em al elemento <h1>, por lo que el tamaño de fuente real será de $2.5em \times 10px = 25px$.

Recuerda que los estilos son heredados por las etiquetas hijas, de manera que si se establece el tamaño de fuente para la etiqueta <body>, todas las etiquetas que contenga la etiqueta <body> tendrán el mismo tamaño de fuente, salvo que se especifique lo contrario.

Sin embargo, las medidas relativas no se heredan directamente, sino que **se heredan sus valores reales una vez calculados**. El siguiente ejemplo muestra este comportamiento:

```
body { font-size: 12px; }  
p    { font-size: 2.5em; }  
p div { font-size: 2em; }
```

En este caso, se establece un tamaño base de fuente de 12px. Para aplicar la siguiente regla, que selecciona todas la etiquetas <p>, se calcula el valor real de 2.5em sobre el tamaño de fuente base: $12 \times 2.5 = 30$ px. Por tanto, 30 px es el valor que se hereda para la última regla, que selecciona todas las etiquetas <div> contenidas dentro de una etiqueta <p>. En esta última regla, el tamaño de fuente será $30 \times 2 = 60$ px.

La ventaja que introduce la utilización de unidades relativas es que mantienen la proporción de tamaños con independencia de la resolución o el tamaño de fuente predeterminada en los distintos navegadores. En general **se recomienda el uso de unidades relativas siempre que sea posible**, ya que mejora la accesibilidad de la página y permite que los documentos se adapten fácilmente a cualquier medio y dispositivo.

2.2. Unidades absolutas

Las unidades absolutas definen las medidas de forma completa ya que sus valores no se calculan a partir de otro valor de referencia, sino que el navegador muestra directamente el valor indicado. Las unidades absolutas definidas en CSS son las siguientes:

- **in**, es una unidad que indica el número de pulgadas (inches). Una pulgada equivale a 2.54 cm.
- **cm**, centímetros.
- **mm**, milímetros.
- **pt**, puntos. 1 punto equivale a 1/72 pulgadas, es decir, unos 0.35 milímetros. Suele ser la medida absoluta más utilizada ya que es la empleada para expresar tamaños absolutos de tipografía.
- **pc**, picas. 1 pica equivale a 12 puntos, es decir, unos 4.23 milímetros).

Su utilización es idéntica a la de las unidades relativas, con la diferencia de que los valores indicados con unidades absolutas son directamente los valores mostrados por el navegador, sin necesidad de calcular los valores reales en función de otras referencias.

3. Colores

Los colores en CSS se pueden indicar de cinco formas diferentes:

- **Palabras clave**
- **RGB decimal**
- **RGB porcentual**
- **RGB hexadecimal**
- **Colores del sistema**

3.1. Palabras clave

CSS define 17 palabras clave para referirse a los colores básicos. Las palabras se corresponden con el nombre en inglés de cada color: aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, orange, purple, red, silver, teal, white, yellow. Aunque es una forma muy sencilla de referirse a los colores básicos, este método prácticamente no se utiliza en las hojas de estilos de los sitios web, ya que se trata de una gama de colores muy limitada. Además de esta lista básica de colores, la mayoría de navegadores soportan muchos otros nombres de colores. Una lista completa puede encontrarse en http://en.wikipedia.org/wiki/Web_safe.

Por ejemplo:

```
p { color: green; }
```

3.2. RGB

El modelo RGB consiste en definir un color indicando la cantidad de color rojo, verde y azul que se debe *mezclar* para obtener la mezcla colorimétrica esperada. Cuando la cantidad de rojo, verde y azul especificada es cero, el color creado es el negro. Por el contrario, si todas las componentes tienen el máximo valor, el color obtenido es el blanco.

En el modelo RGB decimal, cada componente puede tomar valores entre 0 y 255 con la siguiente sintaxis:

```
<selector>
{
  color: rgb(<color_rojo>, <color_verde>, <color_azul>);
}
```

Por ejemplo:

```
p { color: rgb(71, 98, 176); }
```

En el modelo RGB porcentual se utiliza la misma sintaxis pero posponiendo el carácter ‘%’ a cada componente. En este caso los valores de cada componente van del 0% al 100%:

```
p { color: rgb(71%, 98%, 24%); }
```

En el modelo RGB hexadecimal, la sintaxis utilizada es ligeramente diferente: es necesario anteponer el carácter ‘#’ y, a continuación, tres valores hexadecimales sin separación entre ellos. Cada uno de los valores hexadecimales está formado por dos dígitos que van del 00 (valor decimal 0) al FF (valor decimal 255). Como en los anteriores modelos RGB, el primer valor define la componente de color rojo, el segundo valor es la componente verde y el último corresponde al azul:

```
p { color: #1022FF; }
```

Cuando todas las componentes de una especificación son iguales dos a dos, puede utilizarse una notación comprimida, como en los siguientes ejemplos:

```
#AAA = #AAAAAA  
#FFF = #FFFFFF  
#A0F = #AA00FF  
#369 = #336699
```

Las letras que forman parte de cada dígito hexadecimal pueden escribirse en mayúsculas o minúsculas indistintamente.

maroon #800000	red #ff0000	orange #ffa500	yellow #ffff00	olive #808000
purple #800080	fuchsia #ff00ff	white #ffffff	lime #00ff00	green #008000
navy #000080	blue #0000ff	aqua #00ffff	teal #008080	
black #000000	silver #c0c0c0	gray #808080		

3.3. Colores del sistema

Los colores del sistema son similares a los colores indicados mediante su nombre, pero en este caso hacen referencia al color que muestran algunos elementos del sistema operativo del usuario. Debido a que cada sistema operativo utiliza colores completamente distintos, éste es un sistema muy poco utilizado. La lista completa de colores está recogida en <http://www.w3.org/TR/CSS21/ui.html#system-colors>.