

# COLORES Y UNIDADES CSS

Jara Rodríguez Carlos Eduardo  
Cód.: 102509 CJ102509@ingenieria.sanmartin.edu.co  
FUNDACIÓN UNIVERSITARIA SAN MARTIN

*Este artículo tiene como objetivo informar sobre la variedad de colores que existen en las hojas de estilo para la presentación de los documentos en HTML, también para informar sobre las diferentes formas de identificar los colores en CSS.*

*Por otro lado se quiere mostrar las unidades de medida para las hojas de estilo CSS.*

## **I. INTRODUCCIÓN**

Este documento consiste en mostrar las diferentes formas de identificar los colores en una hoja de estilos CSS, también busca exponer las unidades de medidas existentes para las hojas de estilo.

## **II. II.DIFERENTES FORMAS DE IDENTIFICAR LOS COLORES EN UNA HOJA DE ESTILOS.**

### **COLORES RGB HEXADECIMAL**

Se trata del método mas utilizado en los sitios web, se dice que es el método más difícil para indicar los colores.

Para entender este método se necesita implementar el sistema numérico Hexadecimal.

El sistema hexadecimal, utiliza 16 símbolos para representar sus números.

Como sólo conocemos 10 símbolos numéricos, el sistema hexadecimal utiliza también seis letras (de la A a la F) para representar los números. De esta forma, en el sistema hexadecimal, después del 9 no va el 10, sino la A. La letra B equivale al número 11, la C al 12, la D al 13, la E al 14 y la F al número 15.

Definir un color en CSS con el método RGB hexadecimal requiere realizar los siguientes pasos:

- Determinar las componentes RGB decimales del color original, por ejemplo: R = 71, G = 98, B = 176
- Transformar el valor decimal de cada componente al sistema numérico hexadecimal. Se trata de una operación exclusivamente matemática, por lo que puedes utilizar una calculadora. En el ejemplo anterior, el valor hexadecimal de cada componente es:

R = 47, G = 62, B = B0 - Para obtener el color completo en formato RGB hexadecimal, se concatenan los valores hexadecimales de las componentes RGB en ese orden y se les añade el prefijo #, de esta forma, el color del ejemplo anterior es #4762B0 en formato RGB hexadecimal.

Siguiendo el mismo ejemplo de las secciones anteriores, el color del párrafo se indica de la siguiente forma utilizando el formato RGB hexadecimal:

```
p { color: #4762B0; }
```

Aunque es el método más complicado para definir un color, se trata del método que utilizan la inmensa mayoría de sitios web, por lo que es imprescindible dominarlo.

El formato RGB hexadecimal es la forma más compacta de indicar un color, ya que incluso es posible comprimir sus valores cuando todas sus componentes son iguales dos a dos:

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

## 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.

Existen varios colores definidos, ejemplo `ActiveBorder`, que hace referencia al color del borde de las ventanas activas.

Aunque es posible definir los colores en CSS utilizando estos nombres, se trata de un método que nunca se utiliza, por lo que se puede considerar prácticamente como una rareza de CSS.

## RGB DECIMAL

El modelo RGB consiste en definir un color indicando la cantidad de color rojo, verde y azul que se debe mezclar para obtener ese color.

Técnicamente, el modelo RGB es un modelo de tipo

"aditivo", ya que los colores se obtienen sumando sus componentes.

Por lo tanto, un color se define indicando sus tres componentes R (rojo), G (verde) y B (azul). Cada una de las componentes puede tomar un valor entre cero y un valor máximo. De esta forma, el color rojo puro en RGB se crea mediante el máximo valor de la componente R y un valor de 0 para las componentes G y B.

Si todas las componentes valen 0, el color creado es el negro y si todas las componentes toman su valor máximo, el color obtenido es el blanco. En CSS, las componentes de los colores definidos mediante RGB decimal pueden tomar valores entre 0 y 255. El siguiente ejemplo establece el color del texto de un párrafo:









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

La sintaxis que se utiliza para indicar los colores es `rgb()` y entre paréntesis se indican las tres componentes RGB, en ese mismo orden y separadas por comas. El color del ejemplo anterior se obtendría mezclando las componentes R=71, G=98, B=176, que se corresponde con un color azul claro.

Si se indica un valor menor que 0 para una componente, automáticamente se transforma su valor en 0. Igualmente, si se indica un valor mayor que 255, se transforma automáticamente su valor a 255.

Se puede observar un ejemplo o ir a:

<http://www.javascripter.net/faq/rgbtohex.htm>

rgb (255, 255, 255)	# FFFFFF		rgb (255, 0, 255)	# FF00FF
rgb (245, 255, 250)	# F5FFFA		rgb (238, 130, 238)	# EE82EE
rgb (240, 255, 255)	# F0FFFF		rgb (218, 112, 214)	# DA70D6
rgb (240, 255, 240)	# F0FFF0		rgb (186, 85, 211)	# BA55D3
rgb (224, 255, 255)	# E0FFFF		rgb (147, 112, 219)	# 9370DB
rgb (240, 248, 255)	# F0F8FF		rgb (138, 43, 226)	# 8A2BE2
rgb (245, 245, 245)	# F5f5f5		rgb (148, 0, 211)	# 9400D3
rgb (248, 248, 255)	# F8F8FF		rgb (153, 50, 204)	# 9932CC

## RGB PORCETUAL

Las componentes RGB de un color también se pueden indicar mediante un porcentaje. El funcionamiento y la sintaxis de este método es el mismo que el del RGB decimal. La única diferencia es que en este caso el valor de las componentes RGB puede tomar valores entre 0% y 100%. Por tanto, para transformar un valor RGB decimal en un valor RGB porcentual, es preciso realizar una regla de tres considerando que 0 es igual a 0% y 255 es igual a 100%.

El mismo color del ejemplo anterior se puede representar de forma porcentual:

```
p { color: rgb(27%, 38%, 69%); }
```

Al igual que sucede con el RGB decimal, si se indica un valor inferior a 0%, se transforma automáticamente en 0% y si se indica un valor superior a 100%, se trunca su valor a 100%.

## COLORES WEB SAFE

Como cada componente RGB de los colores puede tomar un valor entre 0 y 255, el número total de colores que se pueden representar con este formato es de  $256 \times 256 \times 256 = 16.777.216$  colores. Sin

embargo, en la década de los 90 los monitores de los usuarios no eran capaces de mostrar más de 256 colores diferentes.

A partir de todos los colores disponibles, se eligieron 216 colores que formaron la paleta de colores "web safe". Esta paleta de colores podía ser utilizada por los diseñadores con la seguridad de que se verían correctamente en cualquier navegador de cualquier sistema operativo de cualquier usuario.

Hoy en día, su importancia ha descendido notablemente, ya que prácticamente todos los usuarios utilizan dispositivos con una profundidad de color de 16 y 32 bits. No obstante, el auge en el uso de los dispositivos móviles hace que siga siendo un tema a considerar, ya que las pantallas de muchos móviles sólo pueden representar un número reducido de colores.

## III. UNIDADES DE MEDIDA

Las medidas en CSS se emplean, para definir la altura, anchura y márgenes de los elementos y para establecer el tamaño de letra del texto. Todas las medidas se indican como un valor numérico entero o decimal seguido de una unidad de medida (sin ningún espacio en blanco entre el número y la unidad de medida).

CSS divide las unidades de medida en dos grupos: absolutas y 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. Las unidades absolutas establecen de forma completa el valor de una medida, por lo que su valor real es directamente el valor indicado.

Si el valor es 0, la unidad de medida es opcional. Si el valor es distinto a 0 y no se indica ninguna unidad, la medida se ignora completamente, lo que suele ser uno de los errores más habituales de los diseñadores que empiezan con CSS. Algunas propiedades permiten indicar medidas negativas, aunque habitualmente sus valores son positivos.

Si el valor decimal de una medida es inferior a 1, se puede omitir el 0 de la izquierda (0.5em es equivalente a .5em).

### UNIDADES ABSOLUTAS

Una medida indicada mediante unidades absolutas está completamente definida, ya que su valor no depende de otro valor de referencia.

Lista completa de unidades absolutas definidas por CSS y su significado:

in, pulgadas ("*inches*", en inglés). Una pulgada equivale a 2.54 centímetros.

cm, centímetros.

mm, milímetros.

pt, puntos. Un punto equivale a 1 pulgada/72, es decir, unos 0.35 milímetros.

pc, picas. Una pica equivale a 12 puntos, es decir, unos 4.23 milímetros.

Ejemplo:

```
/* El cuerpo de la página debe mostrar un margen de media pulgada */
```

```
body { margin: 0.5in; }
```

La principal ventaja de las unidades absolutas es que su valor es directamente el valor que se debe

utilizar, sin necesidad de realizar cálculos intermedios.

Su desventaja es que son muy poco flexibles y no se adaptan fácilmente a los diferentes medios.

De todas las unidades absolutas, la única que suele utilizarse es el punto (pt). Se trata de la unidad de medida preferida para establecer el tamaño del texto en los documentos que se van a imprimir, es decir, para el medio print de CSS.

### UNIDADES RELATIVAS

Las unidades relativas, a diferencia de las absolutas, no están completamente definidas, ya que su valor siempre está referenciado respecto a otro valor.

A pesar de su aparente dificultad, son las más utilizadas en el diseño web por la flexibilidad con la que se adaptan a los diferentes medios.

las tres unidades de medida relativas definidas por CSS y la referencia que toma cada una para determinar su valor real:

- em, (no confundir con la etiqueta <em> de HTML) relativa respecto del tamaño de letra del elemento.
- ex, relativa respecto de la altura de la letra x ("*equis minúscula*") del tipo y tamaño de letra del elemento.

- px, (píxel) relativa respecto de la resolución de la pantalla del dispositivo en el que se visualiza la página HTML.

Las unidades em y ex no han sido creadas por CSS, sino que llevan décadas utilizándose en el campo de la tipografía. Aunque no es una definición exacta, la unidad 1em equivale a la anchura de la letra M ("*eme mayúscula*") del tipo y tamaño de letra del elemento.

La unidad em hace referencia al tamaño en puntos de la letra que se está utilizando. Si se utiliza una tipografía de 12 puntos, 1em equivale a 12 puntos. El valor de 1ex se puede aproximar por 0.5 em.

Si se considera el siguiente ejemplo:

```
p { margin: 1em; }
```

La regla CSS anterior indica que los párrafos deben mostrar un margen de anchura igual a 1em. Como se trata de una unidad de medida relativa, es necesario realizar un cálculo matemático para determinar la anchura real de ese margen.

La unidad de medida em siempre hace referencia al tamaño de letra del elemento. Por otra parte, todos los navegadores muestran por defecto el texto de los párrafos con un tamaño de letra de 16 píxel. Por tanto, en este caso el margen de 1em equivale a un margen de anchura 16px.

A continuación se modifica el ejemplo anterior para cambiar el tamaño de letra de los párrafos:

```
p { font-size: 32px; margin: 1em; }
```

El valor del margen sigue siendo el mismo en unidades relativas (1em) pero su valor real ha variado porque el tamaño de letra de los párrafos ha variado. En este caso, el margen tendrá una anchura de 32px, ya que 1em siempre equivale al tamaño de letra del elemento.

Si se quiere reducir la anchura del margen a 16px pero manteniendo el tamaño de letra de los párrafos en 32px, se debe utilizar la siguiente regla CSS:

```
p { font-size: 32px; margin: 0.5em; }
```

El valor 0.5em se interpreta como "la mitad del tamaño de letra del elemento", ya que se debe multiplicar por 0.5 su tamaño de letra ( $32\text{px} \times 0.5 = 16\text{px}$ ). De la misma forma, si se quiere mostrar un margen de 8px de anchura, se debería utilizar el valor 0.25em, ya que  $32\text{px} \times 0.25 = 8\text{px}$ .

La gran ventaja de las unidades relativas es que siempre mantienen las proporciones del diseño de la página. Establecer el margen de un elemento con el valor 1em equivale a indicar que "el margen del elemento debe ser del mismo tamaño que su letra y debe cambiar proporcionalmente".

En efecto, si el tamaño de letra de un elemento aumenta hasta un valor enorme, su margen de 1em también será enorme. Si su tamaño de letra se reduce hasta un valor diminuto, el margen de 1em

también será diminuto. El uso de unidades relativas permite mantener las proporciones del diseño cuando se modifica el tamaño de letra de la página.

El funcionamiento de la unidad ex es idéntico a em, salvo que en este caso, la referencia es la altura de la letra x minúscula, por lo que su valor es aproximadamente la mitad que el de la unidad em.

Las medidas indicadas en píxel también se consideran relativas, ya que el aspecto de los elementos dependerá de la resolución del dispositivo en el que se visualiza la página HTML. Si un elemento tiene una anchura de 400px, ocupará la mitad de una pantalla con una resolución de 800x600, pero ocupará menos de la tercera parte en una pantalla con resolución de 1440x900.

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

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

En primer lugar, se establece un tamaño de letra base de 10 píxel para toda la página.

Se asigna un tamaño de 2.5em al elemento <h1>, por lo que su tamaño de letra real será de 2.5 x 10px = 25px.

El valor de la mayoría de propiedades CSS se hereda de padres a hijos. Así se establece el tamaño de letra al elemento <body>, todos los elementos de

la página tendrán el mismo tamaño de letra, salvo que indiquen otro valor.

Sin embargo, el valor de las medidas relativas no se hereda directamente, sino que se hereda su valor real una vez calculado. El siguiente ejemplo muestra este comportamiento:

```
body {  
  font-size: 12px;  
  text-indent: 3em;  
}  
h1 { font-size: 15px }
```

## REFERENCIAS

[HTTP://LIBROSWEB.ES/CSS/CAPITULO\\_3/UNIDADES\\_DE\\_MEDIDA.HTML](http://LIBROSWEB.ES/CSS/CAPITULO_3/UNIDADES_DE_MEDIDA.HTML)

[HTTP://LIBROSWEB.ES/CSS/CAPITULO\\_3/COLORES.HTML](http://LIBROSWEB.ES/CSS/CAPITULO_3/COLORES.HTML)

[HTTP://WWW.JAVASCRIPTER.NET/FAQ/RGBTOHEX.HTML](http://WWW.JAVASCRIPTER.NET/FAQ/RGBTOHEX.HTML)

[HTTP://WWW.W3SCHOOLS.COM/CSSREF/CSS\\_COLORS.ASP](http://WWW.W3SCHOOLS.COM/CSSREF/CSS_COLORS.ASP)

## Autor

Carlos Eduardo Jara Rodríguez  
Código 102509  
Estudiante Ingeniería de sistemas VIII semestre.  
21 de febrero de 2014