

Estilos más utilizados

A veces es recomendable declarar los estilos en un archivo independiente y luego cargar ese archivo desde un documento que lo necesite, pero obliga al desarrollador a implementar diferentes mecanismos para establecer la relación entre las reglas CSS y los elementos dentro del documento que se verán afectados por éstas.

Existen varios métodos para seleccionar los elementos que serán afectados por una regla CSS. Ya se ha visto que es posible usar el nombre del elemento, pero también se pueden usar los valores de los atributos id y class para referenciar un solo elemento o un grupo de elementos, e incluso combinarlos para construir selectores más específicos.

<pre>span { font-size: 20px; }</pre>	Afecta a todos los elementos que estén contenidos dentro de la una etiqueta .
<pre>p, span { font-size: 20px; }</pre>	Asigna los mismos estilos a elementos con nombres diferentes; la regla afecta a todos los elementos <p> y encontrados en el documento.
<pre>main p { font-size: 20px; }</pre>	Afecta solo a los elementos <p> que se encuentran dentro de un elemento <main>, ya sea como contenido directo o insertados en otros elementos.
<pre>section > p { font-size: 20px; }</pre>	Usa el carácter > para referenciar un elemento que es hijo directo de otro. Indica que el elemento afectado es el elemento de la derecha cuando tiene como padre al elemento de la izquierda.
<pre>h1 + p { font-size: 20px; }</pre>	Este selector hace referencia a un elemento que está precedido por otro elemento; ambos deben compartir el mismo elemento padre. Afecta a todos los elementos <p> que se ubican inmediatamente después de un elemento <h1>.
<pre>p - p { font-size: 20px; }</pre>	Afecta a todos los elementos <p> que preceden a otro elemento <p>.
<pre>#mitexto { font-size: 20px; }</pre>	Se aplica al elemento identificado por el atributo id y el valor "mitexto". La ventaja de este procedimiento es que cada vez que se crea una referencia usando el identificador mitexto en un archivo CSS, solo se modifica el

	elemento con esa identificación, pero el resto de los elementos no se ven afectados. Esta es una forma muy específica de referenciar a un elemento y se usa comúnmente con elementos estructurales, como <section> o <div>.
<code>.mitexto { font-size: 20px; }</code>	Este atributo es más flexible y se puede asignar a varios elementos dentro del mismo documento. Para referenciar un elemento usando su atributo class, el selector debe incluir el valor del atributo precedido por un punto.

Las propiedades son la pieza central de un documento CSS. Todos los estilos que es posible aplicar a un elemento se definen a través de ellas; hay cientos de propiedades disponibles, y se pueden clasificar en dos tipos: propiedades de formato y propiedades de diseño. Las primeras se encargan de dar forma a los elementos y su

contenido, mientras que las de diseño están enfocadas a determinar el tamaño y la posición de los elementos en la pantalla.

Las propiedades de formato se pueden clasificar según el tipo de modificación que producen. Por ejemplo, algunas propiedades cambian el tipo de letra que se usa para mostrar el texto, otras generan un borde alrededor del elemento, asignan un color de fondo, etc. A continuación, se indicarán algunas propiedades fundamentales de los elementos, clasificadas según su tipo.

Texto

<code>font-family</code>	Declara el tipo de letra que se usa para mostrar en un texto. Se pueden declarar múltiples valores separados por coma para ofrecer al navegador varias alternativas en caso de que algunos tipos de letra no se encuentren disponibles en el equipo del usuario. Valores estándar son Georgia, "Times New Roman", Arial, Helvetica, "Arial Black", Gadget, Tahoma, Geneva, Helvetica, Verdana, Geneva, Impact y sans-serif.
<code>font-size</code>	Determina el tamaño de la letra; puede estar indicado en pixeles, puntos, porcentaje o cualquiera de las opciones que entrega el lenguaje.

<code>font-weight</code>	Indica si el texto se mostrará en negrita o no. Los valores posibles son normal y bold, pero también se puede indicar en múltiplos de 100 para grosores de letra.
<code>font-style</code>	Permite establecer el estilo de la letra. Valores posibles: normal, italic, y oblique.
<code>font</code>	Permite declarar varios atributos al mismo tiempo, separándolos por un espacio y en un orden adecuado. El estilo y el grosor se deben declarar antes que el tamaño, y el tipo de letra al final (por ejemplo, <code>font: bold 24px Arial,sans-serif</code>).
<code>text-align</code>	Alinea el texto de un elemento. Valores posibles: left, right, center, y justify.
<code>text-align-last</code>	Alinea la última línea de un párrafo. Valores posibles: left, right, center, y justify.
<code>text-indent</code>	Establecer el tamaño de sangría en un párrafo, que es el espacio vacío al comienzo de la línea.
<code>vertical-align</code>	Alinea elementos de forma vertical. Se usa para alinear texto con imágenes, y la propiedad se aplica a la imagen. Valores posibles: baseline, sub, super, text-top, text-bottom, middle, top, y bottom.
<code>line-height</code>	Define espacio entre líneas. Valor se declara en pixeles, porcentaje, puntos, o en cualquiera de las medidas establecidas en CSS.

Colores

<code>rgb (rojo, verde, azul)</code>	Define un color por medio de los valores especificados por los atributos (desde 0 a 255): nivel de rojo, nivel de verde y nivel de azul, en ese mismo orden.
<code>rgba (rojo, verde, azul, alfa)</code>	Similar a <code>rgb()</code> , pero con un parámetro adicional que define la opacidad; este valor va entre 0 (totalmente transparente) y 1 (totalmente opaco).

hsl (matiz, saturación, luminosidad)	Define un color desde valores especificados por los atributos. Los valores se declaran en números decimales y porcentajes.
hsla (matiz, saturación, luminosidad, alfa)	Cumple una función similar a hsl(), pero considera un parámetro adicional: alfa u opacidad.
color	Declara el color del contenido del elemento.

A continuación, se indica un ejemplo de cómo aplicar estilo de fuente a un elemento y un color.

```
#titulo { font: bold 26px Verdana, sans-serif; color: #CCCCCC;
}
```

El color indicado anteriormente, #CCCCCC, es otra forma de representar un valor. Sigue la misma idea que el color RGB, y cada par de valores considera un color, y cada letra puede ir entre 0 - 9 y A - F. A este tipo de color se le denomina “color hexadecimal”.

Tamaño

width	Esta propiedad declara el ancho de un elemento. Se puede expresar en pixeles, porcentaje, puntos, etc.
height	Declara la altura de un elemento; se representa de igual forma en pixeles, puntos, porcentaje, centímetros, etc.
margin	Declara el margen de un elemento, y corresponde al espacio que hay alrededor de la caja. Puede recibir cuatro valores que representan el margen superior, derecho, inferior, e izquierdo, en ese orden y separados por un espacio (por ejemplo, margin: 10px 30px 10px 30px;).
padding	Declara el relleno de un elemento. Este último es el espacio entre el contenido del elemento y los límites de su caja. Los valores se declaran de la misma forma que lo hacemos para la propiedad margin.

overflow	Indica cómo se mostrará contenido que sale el elemento. Valores disponibles: visible (por defecto), hidden (esconde lo que no entra dentro de la caja), scroll (muestra barras para desplazar el contenido), auto (navegador decide).
----------	---

Fondo

background-color	Asigna un color de fondo a un elemento.
background-image	Asigna una imagen de fondo a un elemento, referenciando la imagen con el método url().
background-position	Declara el inicio de una imagen de fondo; los valores se pueden indicar en porcentaje, pixeles, o con las palabras clave center, left, right, top, y bottom.
background-size	Define el tamaño de la imagen de fondo; puede ser en pixeles o porcentaje, o bien usando las palabras clave cover o contain.
background-repeat	Determina si la imagen de fondo se repetirá, y en qué direcciones. Usa los valores repeat, repeat-x, repeat-y y no-repeat.

Bordes

border-width	Ancho del borde; acepta hasta cuatro valores separados por un espacio para especificar el ancho de cada lado del borde.
border-style	Indica estilos de bordes, cuatro valores separados por espacio. Valores posibles: none, hidden, dotted, dashed, solid, double, groove, ridge, inset, y outset.
border-color	Color del borde, cuatro valores separados.
border	Permite declarar todos los atributos del borde al mismo tiempo.
border-radius	Define el radio del círculo virtual que el navegador utilizará para dibujar las esquinas redondeadas.



Estas son solo algunas de las propiedades que se pueden definir a través de CSS. Desde la llegada de CSS3, se han agregado muchas opciones diversas para generar sitios web con diseños adecuados a las demandas actuales. Existen otras funciones más que es necesario al menos mencionar:

- **Sombras:** genera sombras para contenedores o textos; opciones posibles: box-shadow, text-shadow.
- **Gradientes:** se forma mediante una serie de colores que varían continuamente con una transición suave de un color a otro. Opciones posibles: linear-gradient, radial-gradient.
- **Filtros:** agregan efectos a un elemento y su contenido. CSS incluye la propiedad filter para asignar un filtro a un elemento y las siguientes funciones para crearlo. Opciones posibles: blur(), grayscale(), drop-shadow(), sepia(), brightness(), contrast(), hue-rotate(), invert(), saturate(), opacity().
- **Transformaciones:** Esta propiedad realiza cuatro transformaciones básicas a un elemento: escalado, rotación, inclinación y traslación. Opciones posibles: scale(), rotate(), skew(), translate(); también existen en versión 3D.
- **Transiciones:** Con la seudoclase :hover se puede realizar transformaciones dinámicas. Sin embargo, una animación real requiere una transición entre los dos pasos del proceso. Para este propósito, CSS ofrece las siguientes propiedades: transition-property, transition-duration, transition-timing-function, transition-delay, transition.
- **Animaciones:** La propiedad transition crea una animación básica, pero solo se con dos estados en el proceso: el estado determinado por los valores actuales de las propiedades, y el estado determinado por los nuevos valores. CSS incluye las siguientes propiedades para componer animaciones más complejas: animation-name, animation-duration, animation-timing-function, animation-delay, animation-iteration-count, animation-direction, animation-fill-mode, animation.