



# LENGUAJE DE ESTILOS EN CASCADA

## ¿Qué es el CSS?

Las siglas CSS (Cascading Style Sheets) significan «Hojas de estilo en cascada» y parten de un concepto simple pero muy potente: aplicar estilos (colores, formas, márgenes, etc.) a uno o varios documentos (generalmente documentos HTML, páginas webs) de forma automática y masiva.

Se le denomina estilos en cascada porque se lee, procesa y aplica el código desde arriba hacia abajo (siguiendo un patrón denominado herencia) y en el caso de existir ambigüedad (código que se contradice), se siguen una serie de normas para resolver dicha ambigüedad.

## ¿Para qué sirve el CSS?

Como hemos mencionado, el CSS es un lenguaje informático que especifica cómo se los documentos a los usuarios: cómo se diseñan, compaginan, etc. Un documento suele ser un archivo de texto estructurado con un lenguaje de marcado: HTML es el más común, pero también existen otros como SVG o XML. Presentar un documento a un usuario significa convertirlo en un formulario que el público pueda utilizar. Los navegadores, como por ejemplo Firefox, Chrome o Edge (en-US), están diseñados para presentar documentos visualmente en una pantalla de ordenador, un proyector o una impresora.

El CSS se puede usar para estilos de texto muy básicos como, por ejemplo, cambiar el color y el tamaño de los encabezados y los enlaces. Se puede utilizar para crear un diseño, como podría ser convertir una columna de texto en una composición (en-US) con un área de contenido principal y una barra lateral para información relacionada. Incluso se puede usar para crear efectos de animación. Echa un vistazo a los enlaces de este párrafo para ver ejemplos específicos.

## ¿Por qué se usa CSS?

La idea de CSS es la de utilizar el concepto de separación de presentación y contenido. Este concepto se basa en que, como programadores, lo ideal es separar claramente el código que escribimos. ¿Por qué? Porque con el tiempo, esto hará que el código sea más fácil de modificar y mantener.

La idea es la siguiente:

**Los documentos HTML (contenido)** incluirán sólo información y datos, todo lo relativo a la información a transmitir.

**Los documentos CSS (presentación)** incluirán sólo los aspectos relacionados con el estilo (diseño, colores, formas, etc...).

## Sintaxis del CSS

CSS tiene una sintaxis simple y usa un conjunto de palabras clave en inglés para especificar los nombres de varias propiedades de estilo. Una hoja de estilos consiste en una serie de reglas. Cada regla, o conjunto de reglas consisten en uno o más selectores, y un bloque de declaración.

### **.Selectores {**

Declaran qué etiquetas se le aplican a los estilos que coincidan con la etiqueta o atributo señalados en la regla.

Los selectores pueden aplicarse a:

Todos los elementos de un tipo, como los párrafos <p>

Elementos seguidos de un atributo, en particular:

**id:** identificador, un identificador único para la etiqueta.

**class:** clase, un identificador para anotar múltiples elementos.

```
h1 {  
    color: red;  
    font-size: 5em;  
}
```

### **.Bloque de declaraciones {**

Un bloque de declaraciones consiste en una lista de declaraciones unidas. Cada declaración consiste en una propiedad, dos puntos (:), y un valor. Si hay muchas declaraciones en un bloque, un punto y coma (;) es insertado para separar cada declaración. Las propiedades son insertadas en el estándar CSS. Cada propiedad tiene un conjunto de posibles valores. Algunas propiedades afectan a cualquier elemento, otras solo a un grupo particular de elementos. Los valores pueden ser palabras clave, como "center" o "inherit", o valores numéricos, como 200px (200 píxeles) o 80% (80 por ciento del ancho de la ventana). Los valores de colores son especificados por medio de una palabra clave (ej. "red"), de valores hexadecimales (ej. #Ff0000, pudiéndose abreviar como #F00), valores RGB en una escala del 0 al 255 (ej. rgb(255, 0, 0)), valores RGBA igual que los valores RGB pero con soporte para el canal alfa de transparencias (ej. rgba(255, 0, 0, 0.8)), y valores HSL o HSLA (ej. hsl(000, 100%, 50%), hsla(000, 100%, 50%, 80%)).

### **.Clase {**

Esto funciona siempre que se desee que todos los elementos de ese tipo tengan el mismo aspecto en el documento. La mayoría de las veces no es el caso, por lo que deberás encontrar una manera de seleccionar un subconjunto de los elementos sin que cambien los demás. La forma más común de hacer esto es añadir una clase al elemento HTML y determinarla.

### **.Id {**

Esto funciona siempre que se desee que todos los elementos de ese tipo tengan el mismo aspecto en el documento. La mayoría de las veces no es el caso, por lo que deberás encontrar una manera de seleccionar un subconjunto de los elementos sin que cambien los demás. La forma más común de hacer esto es añadir una clase al elemento HTML y determinarla.