

CSS en 5 minutos...

(o el pinta y colorea para Geeks)

V de Ruido

March 14, 2020

Versión 1.0

Antes de empezar

Este libro o tutorial es una continuación directa del libro *HTML en 5 minutos*. Es bastante recomendable haberlo leído antes de comenzar con este... a no ser que ya tengas conocimientos de HTML. En ese caso, puedes empezar sin miedo.

Una vez más, tengo que agradecer a Alfonso Gil París su trabajo revisando este documento y aportando su granito de arena con recomendaciones y correcciones.

Espero que lo disfrutéis y os sirva.

V de ruido
<https://vderuido.github.io>

Contents

1	¿Qué es CSS?	4
2	¿Qué se necesita?	4
3	Cómo escribir algo en CSS	4
3.1	Elementos HTML	5
3.2	Clases e identificadores	5
3.3	Acciones	6
3.4	Combinando clases y objetos	6
4	Atributos	6
4.1	Lista de atributos más habituales	6
5	¿El orden importa?	7
6	Se acabó la teoría... otra vez	7

1 ¿Qué es CSS?

CSS son las siglas de *Cascade Style Sheet*. No es un lenguaje de programación al uso. No puedes cacharrear cosas monstruosas, pero tiene una utilidad muy clara: hacer que el HTML sea bonito. Tan simple como suena.

En el anterior libro decíamos que HTML era el esqueleto de una página web. Si continuamos con la metáfora, CSS sería el relleno de dicho esqueleto: si la web es más alta, más baja, rubia, morena, con pelo... todas estas cosas podemos decidir las con CSS.

Es muy probable que tengáis dudas sobre su utilidad y que os estéis planteando dejar de leer. Para convenceros, os muestro el siguiente ejemplo de una web con y sin CSS:

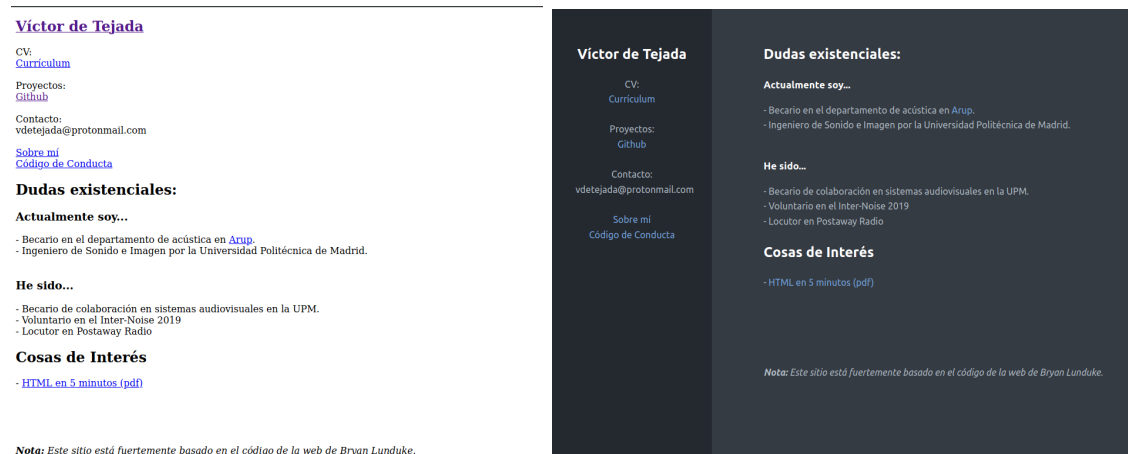


Figure 1: Capturas de la misma web. A la izquierda sin CSS y a la derecha con.

Creo que la diferencia es, cuanto menos, evidente. Así que vamos a empaparnos un poco sobre cómo darle un poco de color a nuestra vida online.

2 ¿Qué se necesita?

Al igual que con HTML, no necesitas más que un editor de texto normal y corriente. *Notepad++*, *Gedit*, *VIM*... usa el que más te guste. Incluso el bloc de notas sirve (aunque agradecerás el los colorines que las otras opciones te dan). Tampoco nos olvidamos de nuestro navegador favorito para visualizar los cambios que hagamos en la web.

3 Cómo escribir algo en CSS

Como hemos dicho antes, CSS es un lenguaje peculiar. Por este motivo, la estructura es mucho más libre, pero antes de comenzar una anarquía voy a darte unas pautas que te recomiendo que sigas:

CSS aplica normas a cosas de HTML. A priori, pensarías en los párrafos, enlaces, imágenes, títulos... y no te equivocarías; pero también puedes aplicárselo a identificadores, clases e incluso a acciones. Cada uno de ellos tiene sus particularidades y su forma de escribirlas. ¡Vamos a ello!

3.1 Elementos HTML

Los elementos HTML son los huesos de los que hablamos en el material de *HTML en 5 minutos*. Podemos hacer que nuestra hoja de estilo aplique las mismas normas para todos los huesos del mismo tipo. Por ejemplo: podemos hacer que todos los párrafos estén justificados, con tamaño de letra 11 y que el texto sea en blanco. Es importante remarcar que estas normas se aplicarán a **TODOS** los elementos de este tipo; por lo que el control queda algo limitado. Para solucionar esto, tendremos que utilizar las clases y categorías (que explicaremos más adelante).

No obstante, aplicar estilos a esta forma es realmente útil cuando queremos que la apariencia general de nuestra web sea lo más consistente posible.

Y bueno, ahora te preguntará cómo demonios se escribe esto. Pues bien, se escribe así:

```
elemento1 {atributo1: ValorAtributo; atributo2: ValorAtributo}
elemento2 {atributo3: ValorAtributo}
```

Si, por ejemplo, quisieras modificar todos los párrafos, sólo habría que sustituir “elemento1” por “p”. También podrías poner “h1” si quisiéramos modificar los títulos, o “a” para los enlaces. Las opciones son tantas como necesites.

También puede pasar que quieras aplicar las mismas normas a varios elementos. Podrías hacerlo uno a uno, pero es más rápido poniendo los elementos entre “,” en la misma línea:

```
elemento1, elemento2, elemento3 {atributo1: ValorAtributo; atributo2: ValorAtributo}
```

De esta forma, ahorramos espacio y líneas y luego será más fácil de modificar si quieres cambiar algo.

3.2 Clases e identificadores

Las clases e identificadores son dos atributos que podían añadirse a un objeto HTML (“class” y “id” respectivamente). Ambos sirven para diferenciar estos elementos de otros y poderles aplicar normas de CSS específicas. La diferencia entre ambas es que “id” se puede usar para un único objeto; mientras que “class” puede utilizarse para tantos como quieras (ya sean objetos iguales o distintos).

Como podrás suponer, cada uno se escribe en CSS de forma distinta: las clases se escriben con un punto (.) delante del nombre que tengan, mientras que los identificadores se usan con la almohadilla delante (#). Aquí te dejo un ejemplo de cada uno.

```
.clase1 {atributo1: ValorAtributo; atributo2: ValorAtributo}
.clase2 {atributo1: ValorAtributo}
#id1 {atributo1: ValorAtributo; atributo2: ValorAtributo}
#id2 {atributo1: ValorAtributo}
```

Como parece lógico pensar, se aplicarán las normas que fijemos a todos aquellos objetos que sean de dicha clase o que tengan ese identificador.

Existe un identificador que es un poco especial. Se le llama “universal” y lo que hace es que sirve para todos los elementos de la página. Se escribe con el asterisco (*):

```
* {atributo1: ValorAtributo; atributo2: ValorAtributo}
```

3.3 Acciones

Una parte interesante de CSS es que permite aplicar reglas a acciones como pasar el ratón por encima o clicar un enlace. Se escribe añadiendo “:” y el nombre de la acción.

```
a:hover{ atributo1: ValorAtributo; atributo2: ValorAtributo}
a:active{ atributo1: ValorAtributo; }
```

En el ejemplo puesto arriba nuestro como se escribiría para modificar el aspecto de un enlace (*a*) cuando se pasea el ratón por encima (*hover*) y cuando se clicka (*active*).

Siendo sinceros, son las únicas acciones que conozco y que he utilizado en estos años. Si hubiera más las añadiré en futuras ediciones.

3.4 Combinando clases y objetos

Los casos expuestos anteriormente cubren la gran mayoría de situaciones, pero.. ¿qué pasa si quiero cambiar sólomente los enlaces dentro de la clase “Pepito”? Pues puedes hacerlo y de una forma, a priori, bastante sencilla.

```
.pepito a {atributo:ValorAtributo }
```

Como podéis ver, sólo hay que poner uno detrás de otro. Partiendo por las clases o id’s y luego el tipo de elemento. Este orden es importante, porque de lo contrario es posible que no funcione como es debido.

4 Atributos

Llegamos de nuevo a la parte más caótica de mis textos. La parte en la que comento rápidamente qué atributos se suelen utilizar y para qué sirven. Al final, quien sabe qué atributo usar en cada momento es uno mismo y hay muchísimas opciones. Te recomiendo que si echas en falta alguno mirés la web de w3school (<https://www.w3schools.com/>) donde podrás buscar todos y bien explicados (en inglés eso sí).

4.1 Lista de atributos más habituales

- color: determina el color del texto. Puede ponerse en inglés o con código RGB.
- background: determina el color de fondo. Muy útil en *div* y otras cajas. Puede ponerse en inglés o en código RGB.
- background-image: te permite añadir una imagen como fondo. Simplemente hay que poner su dirección (local o de internet). Se puede hacer directamente con *background*, pero a mí me parece más rápido así.
- font-size: Determina el tamaño del texto. Puede ser en pixels, puntos, etc.
- font-family: Determina el tipo de letra a usar: Verdana, New Romance, etc.
- text-align: Determina la justificación del texto.

- padding: Determina el espacio entre el contenido y el borde de un objeto. Se puede poner en puntos, píxeles o porcentaje. ¡Es distinto de los márgenes! Va *Padding-tro...* (lo siento).
- margin: Determina los márgenes de un objeto. Se puede fijar en puntos, píxeles o porcentaje.
- border-radius: Determina cuán redondito es un borde (so cute).

Entre muchos y muchos otros...

5 ¿El orden importa?

Ahora que ya conocemos la teoría, toca ponerse a escribir. Pero al igual que en otras ocasiones, deja que te dé un consejo gratis:

¡Sé ordenado!

Esto puede parecer una banalidad, o lo típico que diría tu madre o tu padre, pero nada más lejos. Es algo muy importante.

Cuando comienzas con tu primera hoja de estilo, es muy fácil saber localizar las cosas por si tienes que cambiarlas. Por desgracia, cuando vas cogiendo algo de soltura, puedes encontrarte con una archivo CSS con más de 100 líneas... y ahí las cosas se complican.

Para hacerte la vida un poco más fácil te recomiendo que sigas siempre el mismo orden; de lo más general a lo más concreto, y siempre de bloque en bloque. De esta forma te será más fácil de revisar si algo no sale como esperas.

Un ejemplo de esto podría ser el siguiente en el que he incluido, además, todas las situaciones que he comentado antes (qué majo soy):

```
body{margin:0; font-family: ubuntu; background: #353b43; color: #afbac4}
    h1,h2,h3,h4,h5,h6 {color: white}
    h3 {font-size: 17px}
    a {color: red; text-decoration: none}
    a:hover {color: blue; transition:color .8s}
.app-header {padding: 2.5em; background: #242930; text-align: center}
    .app-header-social {font-size: 2em; color: #fff}
    .app-header-social a {margin: 0.1em}
```

6 Se acabó la teoría... otra vez

¡Pues ahora toca practicar! Sé que al principio puede dar cosita, pero con un poco de práctica harás cosas realmente chulas. Recuerda que tienes la web de <https://w3schools.com> para consultar tus dudas.

También puedes inspirarte en páginas web que ya existan. Existen numerosos sitios donde descargar plantillas como: <https://www.free-css.com/free-css-templates> o puedes ver el CSS de una página en concreto usando las herramientas de “Desarrollo Web” de tu navegador (en las opciones de Chrome y Firefox entre otros).

¡Mucha suerte dando color a tu página web! :D