

Guía práctica CSS

Del inglés *Cascading Style Sheets*, CSS son unos scripts o códigos usados para definir y mejorar la presentación de una página HTML. Un referencia completa de CSS se encuentra en www.w3schools.com/css

Los códigos CSS se pueden implementar utilizando (3) métodos:

1. **Internamente en la página HTML.** Los códigos CSS se incorporan dentro de la etiqueta `<style></style>` que va inmersa dentro del bloque de la etiqueta `<head>` de la página HTML. Este método NO es el recomendado para incorporar estilo a la página HTML.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <style type="text/css">
5       h1 {
6         color: #FF7643;
7         font-family: Arial;
8       }
9       p {
10        color: red;
11        font-size: 1.5em;
12      }
13    </style>
14  </head>
15  <body>
16    <h1>Titulo 1</h1>
17    <p>La hoja de estilo incrustada de una página a otra.</p>
18  </body>
19 </html>
```

2. **En línea con las etiquetas HTML.** Los códigos CSS se pueden incorporar directamente sobre la etiqueta HTML. Este método solo debe ser usado en *casos especiales* u *ocasionalmente*.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Ejemplo CSS</title>
5   </head>
6   <body>
7     <h1 style="color: #FF7643; font-family: Arial">Titulo 1</h1>
8     <p style="color: red;">La hoja de estilo incrustada de una página a otra.</p>
9   </body>
10  </html>
```

3. **Archivo externo .css** Este es el método que siempre debe utilizado para incorporar los estilos a una página HTML. Se trata de un archivo de texto externo con extensión `.css` que puede ser llamado desde la página HTML. El llamado del archivo externo se realiza dentro del bloque de la etiqueta `<head>` Existen (2) formas para llamar el archivo CSS desde la página HTML:

a.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Ejemplo CSS</title>
5     <link rel="stylesheet" type="text/css" href="style.css" />
6   </head>
7   <body>
8     <h1>Titulo 1</h1>
9     <p>La hoja de estilo incrustada de una página a otra.</p>
10  </body>
11 </html>
```

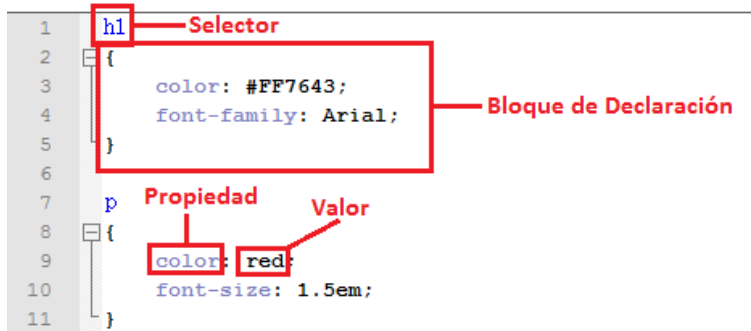
b.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Ejemplo CSS</title>
5     <style type="text/css">
6       @import url(style.css);
7     </style>
8   </head>
9   <body>
10    <h1>Titulo 1</h1>
11    <p>La hoja de estilo incrustada de una página a otra.</p>
12  </body>
13 </html>
```

NOTA: Tener presente la ruta del archivo CSS. En los ejemplos anteriores el archivo **style.css** se encuentra en la misma carpeta del archivo HTML.

Anatomía de un estilo CSS

Un estilo (o regla) está compuesta de dos partes principales: un **Selector**, el cual le dice al navegador web qué va a formatear, y un **Bloque de Declaración**, el cual lista las instrucciones de formato que los navegadores usan para dar estilo al selector.



El bloque de Declaración comienza con una llave de apertura (`{`) y termina con una llave de cierre (`}`). Cada declaración está compuesta de una **Propiedad** y de un **Valor**. Entre la propiedad y el valor se coloca dos puntos (`:`) y al final de la declaración se cierra con punto y coma (`;`).

Tipos de selectores

Existen (3) tipos de selectores a saber: Selectores de Etiqueta, Selectores de Clase y Selectores ID.

Selectores de Etiqueta: Estilo a toda la página

Esta es la forma de aplicar un estilo de forma general a una de las etiquetas propias del HTML. Con ellos, usted puede hacer cambios radicales en el diseño de una página con muy poco esfuerzo.

Los Selectores de etiquetas son fáciles de detectar en una regla de CSS, ya que llevan el nombre exacto la etiqueta HTML, por ejemplo `body`, `p`, `h1`, `table`, `img`, y demás etiquetas HTML.

Selectores de Clase: El control preciso

Los Selectores de clases le permiten localizar exactamente un elemento en una página, independientemente de su etiqueta. Todos los nombres de selectores de clase deben comenzar con un **punto**. Así es como los navegadores web detectar un selector de clase en la hoja de estilo.

```
1 .esp
2 {
3     color:#FF0000;
4     font-family: "Monotype Corsiva";
5 }
```

Para adicionar este tipo de estilo a una etiqueta HTML, se debe adicionar un atributo **class** a la etiqueta. Por ejemplo, si se aplicara el estilo anterior a un párrafo:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Ejemplo CSS</title>
5     <style type="text/css">
6       @import url(scripts/style.css);
7     </style>
8   </head>
9   <body>
10    <h1>Titulo 1</h1>
11    <p class="esp">La hoja de estilo incrustada de una página a otra.</p>
12  </body>
13 </html>
```

Selectores ID: Elementos específicos de página

CSS reserva el selector ID para la identificación de una parte única de la página, como una banner, la barra de navegación, o el área de contenido principal. Al igual que un selector de clase, se crea un identificador dándole un nombre en el CSS, y luego lo aplica mediante la adición de un ID a una etiqueta en el código HTML. Todos los nombres de selectores ID deben comenzar con el símbolo numeral (#).

```
1 #psp
2 {
3   color: #F1CD33;
4   font-size: 14px;
5 }
```

Al decidir si se debe utilizar una clase o un ID, siga estas reglas generales:

- Para usar un estilo varias veces en una página, debe utilizar las clases.
- Utilizar ID para identificar las secciones que se presentan sólo una vez por página.
- Considere el uso de un selector de ID para eludir los conflictos en estilos, ya que los navegadores web dan a los selectores ID prioridad sobre los selectores de clase.

Para adicionar este tipo de estilo a una etiqueta HTML, se debe adicionar un atributo **id** a la etiqueta.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Ejemplo CSS</title>
5     <style type="text/css">
6       @import url(scripts/style.css);
7     </style>
8   </head>
9   <body>
10    <h1>Titulo 1</h1>
11    <p id="psp">La hoja de estilo incrustada de una página a otra.</p>
12  </body>
13 </html>
```

Agrupando Selectores

Para trabajar con los selectores como un grupo, basta con crear una lista de selectores separados por comas.

```
1 h1, h2, h3, h4, h5, h6
2 {
3     color: #F1CD33;
4 }
```

Selectores descendientes

Una etiqueta dentro de una o más etiquetas es considerada como una relación descendiente. Por lo tanto, así mismo se podría aplicar un estilo a una relación descendiente de etiquetas teniendo en cuenta el orden jerárquico en el que se encuentran anidadas las etiquetas. Por ejemplo tenemos la siguiente relación descendiente de etiquetas, donde un enlace se encuentra dentro de un párrafo:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Ejemplo CSS</title>
5     <style type="text/css">
6       @import url(style.css);
7     </style>
8   </head>
9   <body>
10    <h1>Titulo 1</h1>
11    <p>En el sitio <a href="www.aumentality.com">Aumentality</a> vea los ejemplos.</p>
12  </body>
13 </html>
```

Si se quisiera aplicar estilo exclusivamente al enlace dentro del párrafo, la forma correcta de aplicar un estilo CSS sería:

```
1 p a
2 {
3     color: #F1CD33;
4 }
```

Se debe dejar espacio por cada relación descendiente. Estas relaciones descendientes pueden combinarse también con **ID** y **clases**; por lo tanto, se puede hacer relaciones descendientes con todos los (3) tipos de selectores que existen.

Las pseudo-Clases y los pseudo-Elementos

A veces es necesario seleccionar partes de una página web que no tienen etiquetas en sí, pero son fáciles de identificar, como puede ser la primera línea de un párrafo o los estados de un enlace a medida que mueva el ratón sobre él.

Estilos para enlaces

Existen cuatro pseudo-clases para formatear los enlaces basados en los cuatro estados sobre como un visitante interactúa con los enlaces:

- **a:link** selecciona un enlace que no ha sido visitado todavía.
- **a:visited** es un enlace que un visitante a dado click.
- **a:hover** le permite cambiar el aspecto de un enlace cuando un visitante pasa el ratón por encima de él.
- **a:active** le permite determinar cómo será el aspecto de un enlace cuando se ha dado click.

Estilizando Partes de un Párrafo

CSS provee dos pseudo-elementos para los párrafos:

- El pseudo-elemento **:first-letter** crea una letra capital (la letra inicial) al principio del párrafo.
- El pseudo-elemento **:first-line** da estilo a la primera línea de un párrafo en un color diferente que mantiene los ojos de un lector en movimiento y hace que el texto aparezca atractivo y fresco.

Propiedades de uso frecuente

Existen muchas propiedades CSS para dar estilo un sitio web, sin embargo, casi todos los estilos se resumen en las mismas acciones, a menos que se quiera aplicar algo fuera de lo común. A continuación voy a mencionar esas propiedades que más se utilizan, y que casi se vuelven repetitivas para aplicar estilos a las etiquetas.

Recuerde que existen documentos técnicos para realizar consultas más extensas, nunca se quede solamente con lo dicho en el presente documento.

- **Fuente de letra especial**, utilice al principio del estilo CSS:

```
@font-face
{
  font-family: 'Zurich BT';
  src: url('../fonts/Zurich BT.ttf') format('truetype');
  font-weight: normal;
  font-style: normal;
}
```

Nombre de la fuente

Ubicación de la fuente

- **Márgenes**, pueden darse individualmente o en conjunto. En conjunto tenga en cuenta la secuencia: *Arriba Derecha Abajo Izquierda (top right bottom left)*

```
margin: 10px 20px 0px 15px;
margin-top: 10px;
margin-right: 20px;
margin-bottom: 0px;
margin-left: 15px;
```

Son válidas las siguientes propiedades para márgenes:

```
top: 10px;  
right: 20px;  
bottom: 0px;  
left: 15px;
```

- **Rellenos**, a veces pueden ser confundidos en su efecto a las márgenes. También pueden darse individualmente o en conjunto. Tiene la misma secuencia: *Arriba Derecha Abajo Izquierda (top right bottom left)*

```
padding: 10px 20px 0px 15px;  
padding-top: 10px;  
padding-right: 20px;  
padding-bottom: 0px;  
padding-left: 15px;
```

- **Gestion de Fuentes**, se utilizan para determinar las características que pueden tener las fuentes de letra.

```
font-family: Verdana, Tahoma, 'Comic Sans';  
font-size: 15px;  
font-weight: bold; /*bold, normal*/  
font-style: oblique; /*normal, italic, oblique*/  
font-variant: small-caps; /*normal, small-caps, inherit*/  
color: #FFFFCC;
```

- **Gestión de fondos**, son válidos tanto para fondo de pagina como para cualquier tipo de fondo, así se para elementos de formulario:

```
background-color: #7C7C7C;  
background-image: url('imagen.jpg');  
background-repeat: repeat-x; /*repeat-x, no-repeat*/  
background-attachment: fixed; /*scroll, fixed, inherit*/  
background-position: center; /*se puede usar combinaciones de posiciones*/
```

- **Bordes**, a toda etiqueta se le puede adicionar un borde con tipo de línea (solid, dotted y dashed) o efecto (doblé, Groove, ridge, inset, outset), ancho, color. También se puede independizar los bordes por posición y aplicar una característica deseada.

```
border: 1px solid #567CFF;  
border-width: 1px;  
border-style: solid /*solid, dotted, dashed, double*/  
border-color: #567CFF; /*adicionar groove, ridge, inset, outset*/
```

```
border-left-style: dashed;
```

```
border-collapse: collapse; /*usado en tablas para colapsar celdas*/
```

- **Gestión de capas (div).** En las capas se puede utilizar todas las anteriores propiedades, pero en especial para capas se debe tener en cuenta las el tamaño y en especial la posición o ubicación en el área de trabajo.

```
position: relative; /*absolute, relative, fixed, static*/  
float: left; /*left, right, none*/  
overflow: auto; /*auto, scroll, visible, hidden*/  
visibility: hidden; /*visible, hidden, collapse*/  
opacity: 0.75; /*0.0 full transparente, 1.0 full opaco*/  
text-align: right; /*left, center, right*/  
vertical-align: middle; /*top, middle, bottom*/
```

Las recomendaciones para trabajar con capas (div) son las siguientes:

- Las capas principales deben estar como **position: relative;**
- Las capas internas a las principales estar como **float: left;** (o right)
- Las capas más internas dentro de los float, NO se les coloca ni position ni float
- Solo establecer los anchos (width) y altos (height) de las capas , los márgenes no son necesarios si todo el bloque de capas está centrado dentro de las etiquetas **<center></center>**.

- Otras propiedades que se uso frecuente son:

```
clear: both; /* Resetear estilos en adelante */  
text-decoration: none; /* Eliminar el subrayado de los enlaces */  
list-style-type: none; /* Eliminar los puntos de las viñetas */
```

Las propiedades anteriores son los más utilizados para dar estilo a etiquetas HTML, existen muchas más y que se utilizan en situaciones especiales. Complemente información visitando el sitio www.w3schools.com/css