

## IFCD0110 CONFECCIÓN Y PUBLICACIÓN DE PÁGINAS WEB

### Módulo formativo: MF0950\_2: Construcción de páginas web

**Unidad formativa: UF1303 Elaboración hojas de estilo.**

**UNIDAD 1 Hojas de estilo CSS3 en la construcción de páginas web. Funciones y características. Selectores. Atributos de estilo de fuente, listas, tablas y enlaces.**

**FORMACIÓN PROFESIONAL PARA EL EMPLEO.**

**CIPP VIRGEN DE GRACIA**



CIPP VIRGEN DE GRACIA



Castilla-La Mancha



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE TRABAJO  
Y ECONOMÍA SOCIAL



1.1.	Funciones y características .....	5
1.2.	Descripción de estilos .....	6
1.3.	Sintaxis básica de estilos.....	7
1.4.	Utilización de estilos .....	9
	Estilos en cascada .....	9
	Errores más comunes .....	9
1.5.	Tipos de estilo: incrustados, enlazados, importados, en línea.....	10
1.6.	Selectores y reglas de estilo.....	14
	Selector de tipo: .....	14
	Selector universal: * .....	14
	Selector de clase.....	14
	Selector de ID: #.....	15
	Selector de descendientes: E F .....	16
	Selector de hijos: E > F.....	18
	Selector de consecutivos: E + F.....	18
	Selector de hermanos: E ~ F .....	19
	Selector de atributo.....	20
1.7.	Las pseudo-clases y los pseudo-elementos .....	21
	La pseudo-clase :first-child .....	22
	Las pseudo-clases dinámicas :hover, :active y :focus .....	25
	Los pseudo-elementos ::first-line y ::first-letter.....	27
	Los pseudo-elementos ::before y ::after .....	30
1.8.	Adecuación de las hojas de estilos.....	35
	Reglas distintas que se aplican al mismo elemento .....	37
1.9.	Unidades. Distancias y tamaños.....	42
	La unidad relativa respecto al tipo de letra rem.....	46
	Unidades relativas respecto al tamaño de la pantalla vw, vh, vmin y vmax .....	46
	Unidades absolutas .....	47
1.10.	Colores .....	48
	Códigos de colores RGB.....	48
	Códigos de colores RGBA.....	49
	Códigos de colores HSL.....	49
	Códigos de colores HSLA .....	50
	Nombres de colores HTML 3.2 .....	50
	Nombres de colores SVG / X11.....	50

Nombres de colores del sistema .....	52
Opacidad (transparencia): opacity .....	52
Esquemas de colores .....	53
1.11. Adaptación de los ficheros de estilo para distintas páginas web y dispositivos móviles. ....	54
Diseño adaptativo (responsive design) .....	54
Recomendaciones.....	66
Densidad de píxeles: etiqueta meta viewport.....	66
La regla arroba @viewport.....	67
Hojas de estilo específicas: etiqueta link media.....	67
Características del dispositivo: media queries .....	68
Ancho de pantalla: min-width y max-width .....	69
1.12. Guía de estilo .....	70
1.13. Validación de páginas web y hojas de estilo .....	74
Validación de páginas web .....	74
Validadores disponibles en Internet.....	74
Validación en el navegador .....	75
1.14. Listado de propiedades CSS estables.....	75
1.15. Atributos de estilo para fuente .....	78
Características tipográficas.....	86
Fuentes Web.....	90
Google Fonts / Google Fonts API.....	91
Color del texto: color .....	98
Color de fondo: background-color .....	98
Alineación horizontal: text-align.....	98
Alineación vertical: vertical-align .....	99
Sangría: text-indent.....	100
Espaciado de letras: letter-spacing .....	100
Espaciado de palabras: word-spacing.....	101
Decoración: text-decoration .....	101
Mayúsculas / minúsculas: text-transform .....	102
Sombreado: text-shadow .....	102
Espacios en blanco: white-space .....	103
Dirección: direction.....	105

Dirección: unicode-bidi .....	106
<b>1.16. Líneas horizontales .....</b>	<b>106</b>
Tamaño: width y height .....	107
Color: color, background-color y border-color .....	107
Alineación: margin-left, margin-right y width .....	108
<b>1.17. Tablas.....</b>	<b>108</b>
Posición de la leyenda: caption-side.....	109
Centrar una tabla.....	109
Disposición de la tabla: table-layout .....	110
Modos de bordes: border-collapse .....	110
Separación entre celdas: border-spacing.....	110

## 1.1. Funciones y características.

CSS son las siglas de Cascade Style Sheet que traducido significa hojas de estilo en cascada.

Las hojas de estilo es una tecnología que nos permite controlar la apariencia o formato de una página web. En un principio, los sitios web se concentraban más en su contenido que en su presentación.

HTML no pone atención en la apariencia del documento, sino en la estructura. CSS describe como los elementos dispuestos en la página son presentados al usuario. CSS es un gran avance que complementa el HTML y la Web en general.

Con CSS podemos especificar estilos como **el tamaño, fuentes, color, espaciado entre textos y recuadros así como el lugar donde disponer texto e imágenes en la página**.

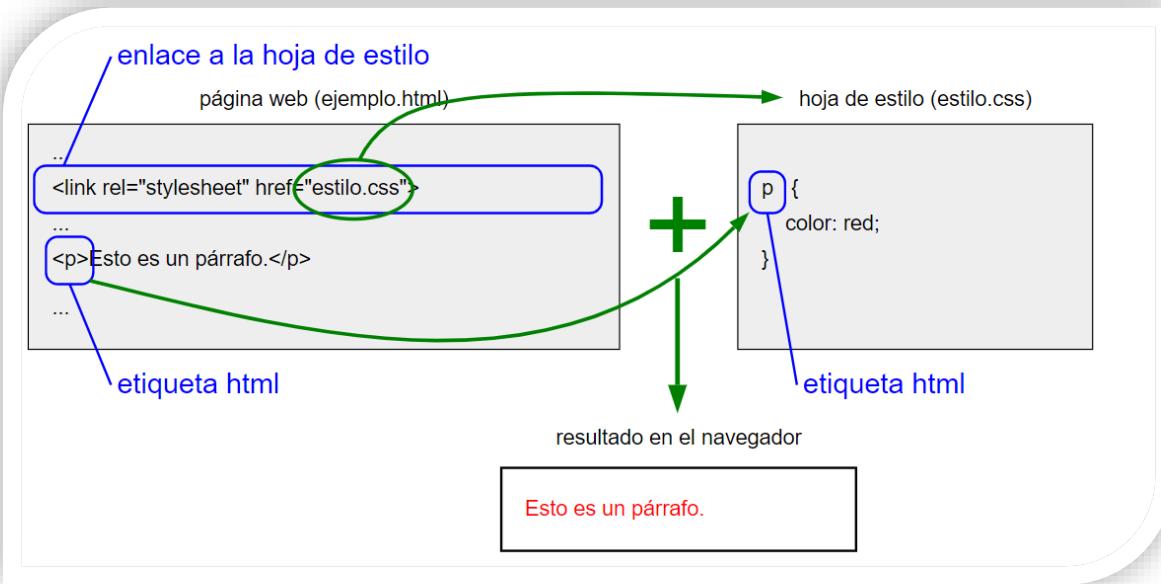
El lenguaje de las Hojas de Estilo está definido en la Especificaciones CSS1, CSS2 y CSS3 del World Wide Web Consortium (W3C), es un **estándar aceptado** por toda la industria relacionada con la Web, o por lo menos, gran parte de navegadores (es verdad que el Internet Explorer de Microsoft dio algunos dolores de cabeza en versiones antiguas). Podemos visitar [W3C](http://www.w3.org)

Veremos que podemos asociar las reglas de estilo a las marcas HTML de tres maneras: directamente a la marca, en el head de la página o agrupar las reglas de estilo en un archivo independiente con extensión \*.css

El funcionamiento de las hojas de estilo es el siguiente:

- En la página web (**archivo .html**) se escriben las **etiquetas** que definen categorías o **elementos**.
- En la hoja de estilo (**archivo .css**) se escribe cómo queremos que sea el **estilo de presentación** de las etiquetas (**color, tamaño, fuente, bordes, márgenes, posición, etc.**).
- En la página web se indica qué hoja de estilo queremos utilizar.

Las **ventajas** de utilizar hojas de estilo son muchas, sobre todo **permiten hacer un diseño consistente y fácil de modificar**. Si varias páginas web hacen referencia a la misma hoja de estilo, para cambiar la apariencia de todas las páginas es suficiente con hacer los **cambios en un único lugar, en la hoja de estilo**.



## 1.2. Descripción de estilos

Un documento CSS se compone de un conjunto de reglas y cada regla consta de las siguientes partes:

- **Selector**: Indica a qué elemento o parte de la página se aplica la regla. Normalmente los selectores se corresponden con las etiquetas del lenguaje HTML. Por ejemplo `table` es un selector.
- **Declaración**: Indica la definición del estilo a aplicar al selector indicado. La declaración a su vez puede dividirse en dos partes:
  - **Propiedad**: Indica el nombre del atributo al que va a ser aplicado.
  - **Valor**: Define el valor de la propiedad.

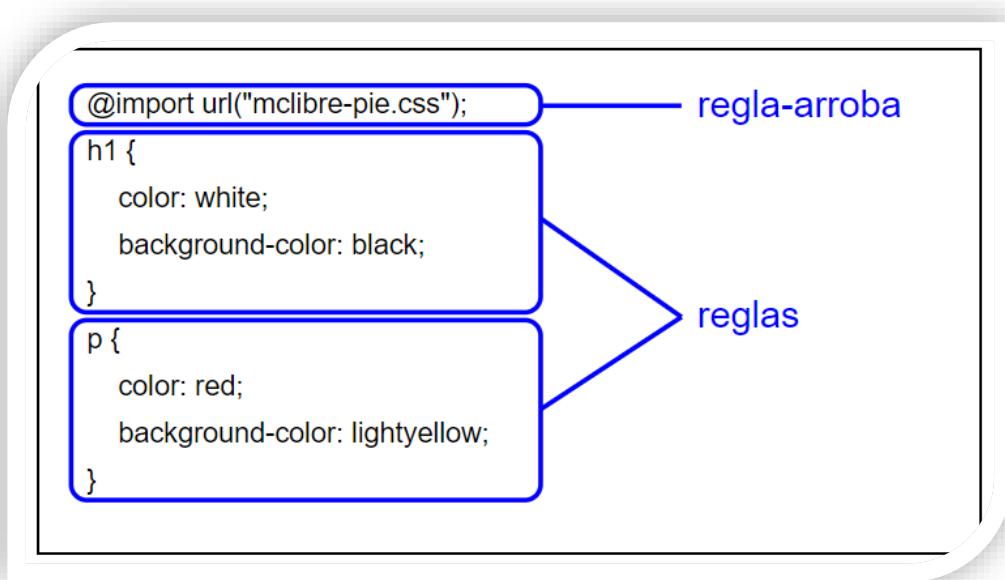
Una hoja de estilo es un fichero de texto plano (sin formato) en el que se define el aspecto de las etiquetas de una página web:

```

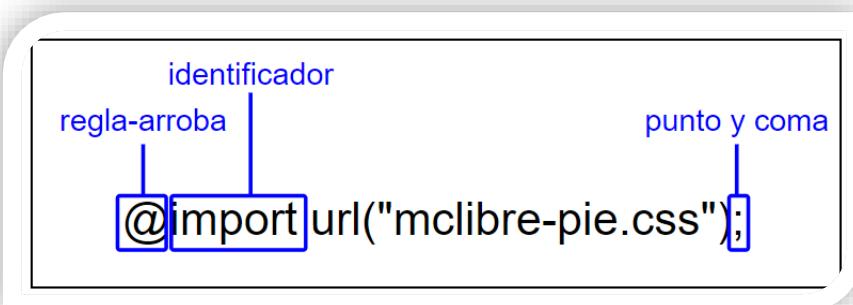
h1 {
  color: white;
  background-color: black;
}
p {
  color: red;
  background-color: lightyellow;
}

```

Una hoja de estilo está formada por una o varias **sentencias**. Existen dos tipos de sentencias: las reglas-arroba y las reglas:



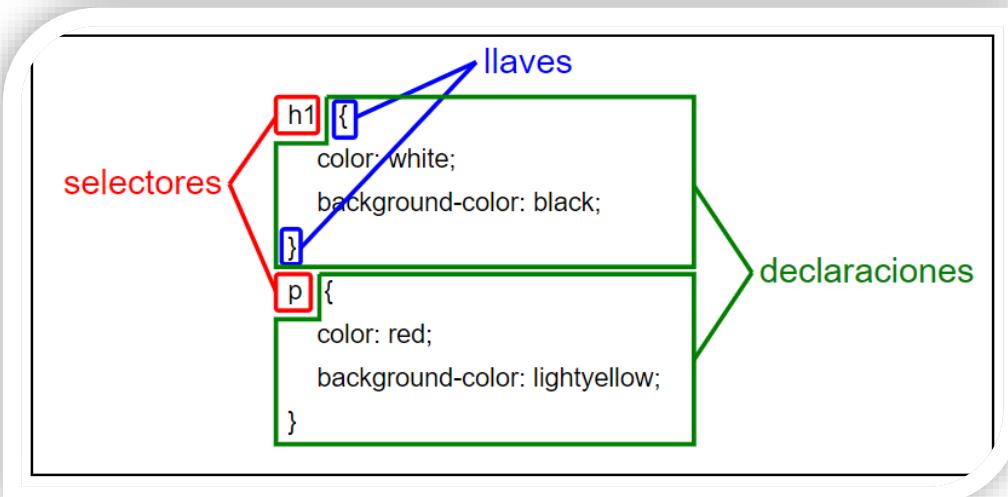
Las reglas-arroba empiezan por el símbolo de arroba (@) pegado a un **identificador** y terminan por un punto y coma (;):



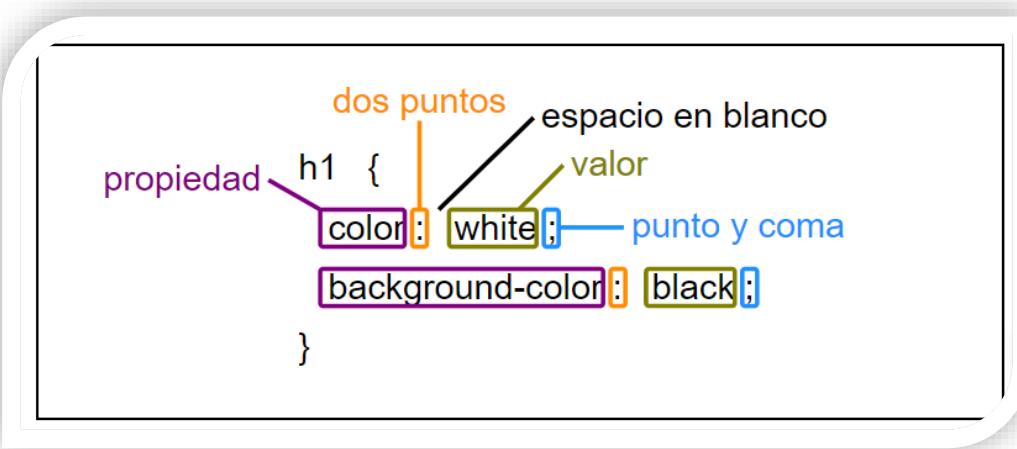
Algunas reglas-arroba (**@import** o **@namespace**) deben aparecer al principio de la hoja de estilo, pero el resto (**@font-face**, **@media**, etc.) pueden escribirse en cualquier lugar de la hoja de estilo.

### 1.3. Sintaxis básica de estilos.

Las reglas están formadas por un **selector** y un bloque de **declaración**. El bloque de declaración empieza y acaba con llaves {}.



Cada declaración está formada por una o **varias propiedades** y sus valores correspondientes. Las propiedades van **separadas entre sí por puntos y comas**. El **valor** (o valores) van separados de las propiedades por **dos puntos**. En general, si una propiedad necesita **varios valores**, los valores van separados por **espacios en blanco**. En general, si una propiedad admite varios **valores alternativos**, los valores van **separados por comas**.



### Comentarios en las hojas de estilo

Una hoja de estilo puede contener comentarios. Los delimitadores del comentario son `/* ... */`. Los comentarios pueden extenderse varias líneas, como ilustra el siguiente ejemplo:

```
/*
 Autor:
 Fecha:
 */

h1 {
    background-color: black ;      /* fondo negro */
    color: white ;                /* texto blanco */
}
```

## 1.4. Utilización de estilos.

### Estilos en cascada

El formato final de cualquier elemento depende de la combinación de todas las propiedades que afectan al elemento. La denominación de hojas de estilo **en cascada** hace referencia precisamente a este hecho.

Por ejemplo, una de esas reglas es que a un elemento le afectan todas las propiedades de los elementos en los que está incluido, como muestra el ejemplo siguiente, en el que el color de texto del párrafo <p> viene determinado por la propiedad establecida para el elemento <body>:

```
body {
    color: white;
}

p {
    background-color: lightcoral;
}
```

Esto es un párrafo.

### Errores más comunes

Un error bastante habitual es escribir los caracteres < y > en la hoja de estilo.

 <p> {
 color: red;
}

Esto es un párrafo.

### Definir varios selectores simultáneamente

Se pueden definir varios selectores simultáneamente escribiendo los selectores separados por comas, como muestran los siguientes ejemplos:

```
p, address {
    color: red
}
```

Esto es un párrafo <p>.  
Esto es una dirección <address>.

Lo que no es correcto es definir varias propiedades simultáneamente, como muestran los siguientes ejemplos:



```
p {  
    border-top, border-bottom: red 3px solid;  
}
```

Esto es un párrafo.



```
p {  
    border-top: red 3px solid;  
    border-bottom: red 3px solid;  
}
```

Esto es un párrafo.

### Repetir selectores

Aunque no es incorrecto, no se aconseja repetir selectores, ya que distribuye la información en varios lugares de la hoja de estilo y eso dificulta la lectura e interpretación de la hoja de estilo.

Si se repiten las propiedades, el navegador aplica la propiedad que aparece en último lugar, como muestra el siguiente ejemplo:



```
p {  
    color: red;  
}  
  
p {  
    color: blue;  
}
```

Esto es un párrafo.

Si no se repiten las propiedades, el navegador aplica todas las propiedades, como muestra el siguiente ejemplo:



```
p {  
    color: white;  
}  
  
p {  
    background-color: black;  
}
```

Esto es un párrafo.

## 1.5.Tipos de estilo: incrustados, enlazados, importados, en línea.

Las reglas de estilo pueden ser declaradas: incrustados, en la cabecera (head) del documento o en un documento externo con extensión CSS.

### Incrustar un estilo dentro de un documento HTML

- En línea con el elemento: **utilizando estilos** directamente sobre aquellos elementos que lo permiten a través del atributo `<style>` dentro de `<body>`. Pero este tipo de definición del estilo **pierde las ventajas** que ofrecen las hojas de estilo al mezclarse el contenido con la presentación. No deben utilizarse:

```
<h2 style="color: red; font-family: Helvetica, Geneva, Arial, sans-serif"> Título  
de la cabecera </h2>
```

- **Declaración en el head del CSS.** Utilizando el elemento `<style>`, en el interior del documento al que se le quiere dar estilo, en la sección `<head>`. De esta forma los estilos serán reconocidos antes de que la página se cargue por completo.

```
<!DOCTYPE html>
<html lang="es">
<title>hoja de estilo interna</title>
<style type="text/css">

    body {
        padding-left: 11em;
        font-family: Georgia, "Times New Roman", serif;
        color: red;
        background-color: #d8da3d;
    }

    h1 {
        font-family: Helvetica, Geneva, Arial, sans-serif;
    }

</style>
</head>
<body>
    <h1>Aquí se aplicará el estilo de letra para el Título</h1>
</body>
</html>
```

*Enlazar una hoja de estilo externa a un documento HTML.*

- Utilizando una hoja de estilo externa que estará **vinculada** a un documento a través del elemento **<link>**, el cual debe ir situado en la sección **<head>**.

```
<!DOCTYPE html>
<html lang="es">

<head>
    <title>Título</title>
    <link rel="stylesheet" type="text/css" href="css/misestilos.css">
</head>
<body>
</body>
```

Una página web puede tener enlazadas varias hojas de estilo distintas para que el usuario elija cuál se aplica.

*Definir hojas de estilo alternativas*

En una página web, la etiqueta **<link>** situada en la cabecera (**<head>**) indica la ubicación y nombre de la hoja de estilo enlazada mediante el atributo **href**:

```
<link rel="stylesheet" href="nombre_del_archivo.css">
```

Se pueden enlazar dos o más hojas de estilo, con lo que el código fuente contendrá tantas etiquetas `<link>` como hojas enlazadas:

```
<link rel="stylesheet" href="nombre_de_archivo_1.css">
<link rel="stylesheet" href="nombre_de_archivo_2.css">
```

En este caso, el navegador aplica simultáneamente las dos hojas de estilo a la página web. Si un elemento está definido en las dos hojas, el navegador hará lo que diga la última hoja de estilo de la lista.

---

Para que sean hojas de estilo alternativas, es decir, para que el usuario pueda elegir en el navegador qué hoja de estilo quiere aplicar, es suficiente con que los enlaces a las hojas de estilo tengan el atributo `title`:

```
<link rel="stylesheet" href="nombre_de_archivo_1.css" title="Hoja 1">
<link rel="stylesheet" href="nombre_de_archivo_2.css" title="Hoja 2">
```

En este caso la primera hoja de estilo enlazada es la que se aplica de forma predeterminada y el resto son hojas de estilo alternativas.

---

Si se quiere indicar explícitamente la hoja de estilo predeterminada, también se puede establecer en el enlace a la hoja de estilo predeterminada el atributo `rel` con el valor `"stylesheet"` y establecer en el resto de enlaces el atributo `rel` con el valor `"alternate stylesheet"`. En este caso, el orden de los enlaces no es importante.

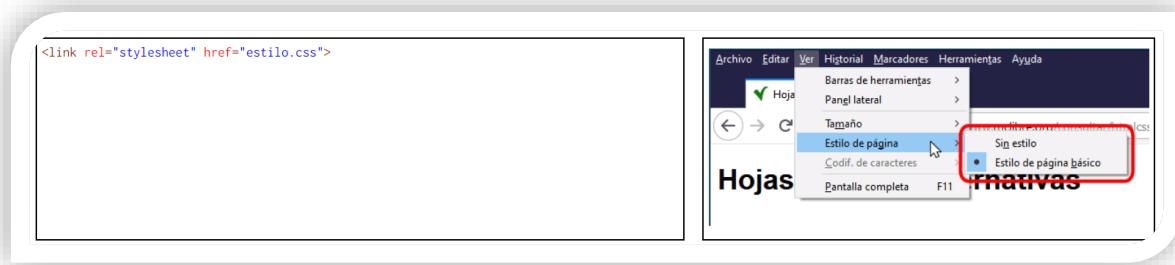
```
<link rel="stylesheet" href="nombre_de_archivo_1.css" title="Hoja 1">
<link rel="alternate stylesheet" href="nombre_de_archivo_2.css" title="Hoja 2">
```

Hojas de estilo alternativas en los navegadores

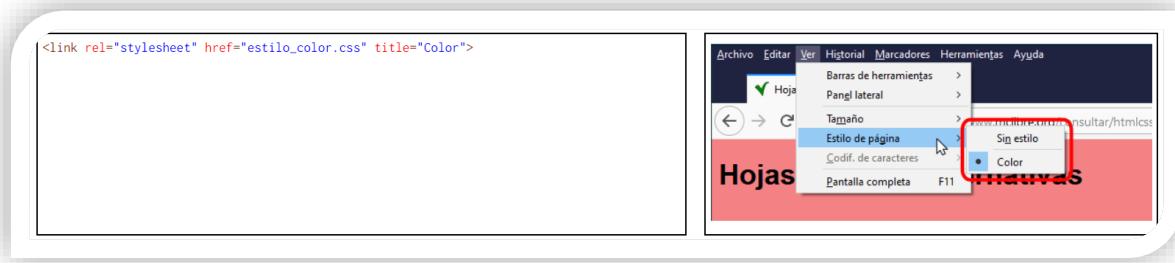
### Hojas de estilo alternativas en Firefox

La selección de hojas de estilo se realiza en Firefox mediante el menú Ver > Estilo de página. El menú se muestra pulsando la tecla Alt. Las opciones de este menú dependen de los atributos de las hojas de estilo enlazadas.

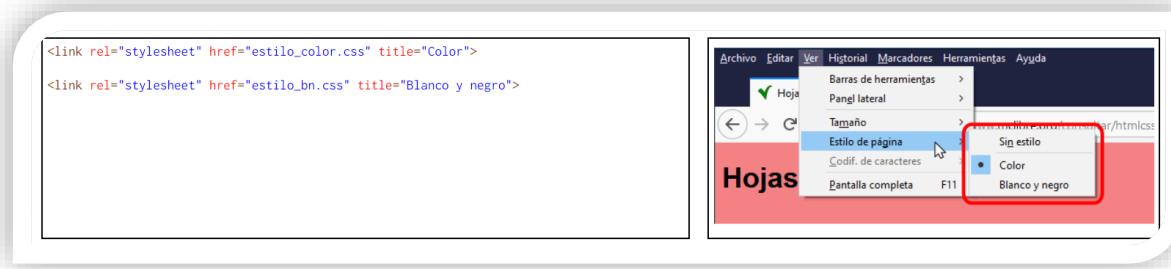
- si la hoja (u hojas) de estilo enlazada tiene el atributo `rel="stylesheet"`, pero no tiene el atributo `title`, el menú muestra las opciones: "Sin estilo" y "Estilo de página básico". Si se elige "Sin estilo", Firefox muestra la página sin aplicar la hoja de estilo. Si se quiere volver a aplicar la hoja de estilo, hay que elegir "Estilo de página básico".



- si hay una sola hoja de estilo enlazada y tiene el atributo `title`, el menú muestra las opciones: "Sin estilo" y el título de la hoja de estilo. Si se elige "Sin estilo", Firefox muestra la página sin aplicar la hoja de estilo. Si se quiere volver a aplicar la hoja de estilo, hay que elegir la otra opción.



- si hay varias hojas de estilo enlazadas y todas tienen el atributo `title`, el menú muestra las opciones: "Sin estilo" y todos los títulos de las hojas de estilo. Si se elige "Sin estilo", Firefox muestra la página sin aplicar la hoja de estilo. Si se quiere aplicar cualquier hoja de estilo, hay que elegir la opción correspondiente.



### Hojas de estilo alternativas en Google Chrome

Google Chrome no permite elegir hojas de estilo alternativas, aunque se pueden instalar extensiones como [AltCSS](#) para añadir esta funcionalidad.

## 1.6. Selectores y reglas de estilo.

En las hojas de estilo CSS, los selectores son la parte de las reglas que indican al navegador a qué elementos se van a aplicar las propiedades incluidas en las declaraciones.

La recomendación [Selectors Level 3 \(2º edición\)](#), publicada en noviembre de 2018, define la sintaxis de los selectores, las pseudo-clases y los pseudo-elementos.

Se comentan únicamente los selectores simples y sus combinaciones. Las pseudo-clases y los pseudo-elementos se estudiarán más adelante.

### Selector de tipo:

Si se escribe una etiqueta, las propiedades afectan a todos los elementos con dicha etiqueta. En el ejemplo siguiente, todos los párrafos `<p>` se ven de color rojo.

```
p {  
    color: red;  
}
```

Esto es un párrafo <p>.  
Esto es otro párrafo <p>.

### Selector universal: \*

Si se escribe un asterisco (\*), las propiedades afectan a todos los elementos de la página. En el ejemplo siguiente, tanto el párrafo `<p>` como la dirección `<address>` se ven de color rojo.

```
* {  
    color: red;  
}
```

Esto es un párrafo <p>.  
Esto es una dirección <address>.

### Selector de clase

Si se escribe una etiqueta seguida de un punto y un nombre de clase, las propiedades afectan a todos los elementos con la etiqueta, cuyo atributo `class` tenga el valor e. En el ejemplo siguiente, sólo el párrafo `<p>` de clase "aviso" se ven de color rojo.

```
<body>
  <p>Esto es un párrafo &lt;p&gt;.</p>

  <p class="aviso">Esto es un párrafo &lt;p&gt; de clase "aviso".</p>
</body>
</html>
```

```
p.aviso {
  color: red;
}
```

Esto es un párrafo <p>.  
Esto es un párrafo <p> de clase "aviso".

Para definir una clase que se pueda aplicar a cualquier etiqueta, se puede escribir como selector un punto y el nombre de la clase (.nombre-clase) o el selector universal \*.nombre-clase. En los dos ejemplos siguientes, cualquier elemento de clase "aviso" se ve de color rojo.

```
<body>
  <p>Esto es un párrafo sin clase <span class="aviso">pero</span> con un &lt;span&gt; de clase "aviso" (en la palabra "pero").</p>
  <p class="aviso">Esto es un párrafo &lt;p&gt; de clase "aviso".</p>
  <address class="aviso">Esto es una dirección &lt;address&gt; de clase "aviso".</address>
</body>
```

```
.aviso {
  color: red;
}
```

Esto es un párrafo sin clase **pero** con un <span> de clase "aviso" (en la palabra "pero").  
Esto es un párrafo <p> de clase "aviso".  
*Esto es una dirección <address> de clase "aviso".*

```
<body>
  <p>Esto es un párrafo sin clase <span class="aviso">pero</span> con un &lt;span&gt; de clase "aviso" (en la palabra "pero").</p>
  <p class="aviso">Esto es un párrafo &lt;p&gt; de clase "aviso".</p>
  <address class="aviso">Esto es una dirección &lt;address&gt; de clase "aviso".</address>
</body>
```

```
*.aviso {
  color: red;
}
```

Esto es un párrafo sin clase **pero** con un <span> de clase "aviso" (en la palabra "pero").  
Esto es un párrafo <p> de clase "aviso".  
*Esto es una dirección <address> de clase "aviso".*

### Selector de ID: #

**Nota:** A menudo se desaconseja el uso de selectores de ID y se recomienda utilizar en su lugar un selector de clase, por eso en la se recomienda configurar el editor de manera que se señale como aviso el uso de selectores de ID.

Si se escribe una etiqueta seguida de una almohadilla y un nombre de **id**, las propiedades afectan al elemento con etiqueta cuyo atributo **id** tenga el valor. En el ejemplo siguiente, sólo el párrafo **<p>** de **id** **aviso** se ve de color rojo.

```
<body>
  <p class="aviso">Esto es un párrafo &lt;p&gt; de clase "aviso".</p>

  <p id="aviso">Esto es un párrafo &lt;p&gt; con id "aviso".</p>
</body>
```

```
p#aviso {
  color: red;
}
```

Esto es un párrafo **<p>** de clase "aviso".  
Esto es un párrafo **<p>** con id "aviso".

Hay que tener en cuenta que el atributo **id** no se puede repetir, es decir, no puede haber dos elementos con el mismo valor del atributo **id** (independientemente de que los elementos tengan etiquetas iguales o distintas).

Para definir un **id** que se pueda aplicar a cualquier etiqueta, se puede escribir como selector una almohadilla y el nombre del **id** (#nombre-id) o el selector universal \*#nombre-id. En los dos ejemplos siguientes, cualquier elemento con **id** "aviso" se ve de color rojo.

Aunque en una página web no puede haber dos elementos con el mismo valor del atributo **id**, en páginas distintas sí que puede haber dos elementos con el mismo valor del atributo **id**. Si la misma hoja de estilo se enlaza desde varias páginas web, el selector universal afectará en cada página a los elementos con el **id** correspondiente.

```
<body>
  <p class="aviso">Esto es un párrafo &lt;p&gt; de clase "aviso".</p>
  <p id="aviso">Esto es un párrafo &lt;p&gt; con id "aviso".</p>
</body>
```

```
<body>
  <address class="aviso">Esto es una dirección &lt;address&gt; de clase "aviso".</address>
  <address id="aviso">Esto es una dirección &lt;address&gt; con id "aviso".</address>
</body>
```

```
#aviso {
  color: red;
}
```

Esto es un párrafo **<p>** de clase "aviso".  
Esto es un párrafo **<p>** con id "aviso".

```
*#aviso {
  color: red;
}
```

Esto es una dirección **<address>** de clase "aviso".  
Esto es una dirección **<address>** con id "aviso".

### Selector de descendientes: E F

Si se escriben dos etiquetas seguidas (E F), las propiedades afectan a los elementos con la segunda etiqueta (F) contenidos dentro de la primera etiqueta (E), aunque haya etiquetas intermedias. En el ejemplo siguiente, todos los párrafos **<p>** dentro de una división **<div>** se ven de color rojo.

```
<body>
  <p>Esto es un párrafo &lt;p&gt;.</p>

  <div>
    <p>Esto es un párrafo &lt;p&gt;.</p>

    <ul>
      <li>Esto es una lista
        <p>Esto es un párrafo &lt;p&gt; dentro de una lista</p>
      </li>
    </ul>

    <p>Esto es otro párrafo &lt;p&gt;.</p>

    <address>Esto no es un párrafo &lt;p&gt;, sino una dirección &lt;address&gt;.</address>
  </div>

  <p>Esto es otro párrafo &lt;p&gt;.</p>
</body>
```

```
div {
  border: red 3px solid;
  margin: 10px;
  padding: 10px;
}

div p {
  color: red;
}
```

Esto es un párrafo <p>.  
Esto es un párrafo <p>.  
• Esto es una lista  
 Esto es un párrafo <p> dentro de una lista  
Esto es otro párrafo <p>.  
Esto no es un párrafo <p>, sino una dirección <address>.  
Esto es otro párrafo <p>.

En general, si se escriben varias etiquetas seguidas (E F G ...), las propiedades afectan a los elementos con la última etiqueta contenidos dentro de la etiqueta anterior, contenidos a su vez dentro de etiqueta anterior y así sucesivamente, aunque haya etiquetas intermedias. En el ejemplo siguiente, únicamente los párrafos **<p>** dentro de un elemento de lista **<li>** dentro de una división **<div>** se ven de color rojo.

```
<body>
  <p>Esto es un párrafo &lt;p&gt;.</p>

  <div>
    <p>Esto es un párrafo &lt;p&gt;.</p>

    <ul>
      <li>Esto es una lista
        <p>Esto es un párrafo &lt;p&gt; dentro de una lista</p>
      </li>
    </ul>

    <p>Esto es otro párrafo &lt;p&gt;.</p>

    <address>Esto no es un párrafo &lt;p&gt;, sino una dirección &lt;address&gt;.</address>
  </div>

  <p>Esto es otro párrafo &lt;p&gt;.</p>
</body>
```

```
div {  
    border: red 3px solid;  
    margin: 10px;  
    padding: 10px;  
}  
  
div li p {  
    color: red;  
}
```

Esto es un párrafo <p>.  
Esto es un párrafo <p>.  
• Esto es una lista  
 Esto es un párrafo <p> dentro de una lista  
Esto es otro párrafo <p>.  
Esto no es un párrafo <p>, sino una dirección <address>.  
Esto es otro párrafo <p>.

### Selector de hijos: E > F

Si se escriben dos etiquetas seguidas separadas por un signo "mayor que" (E > F), las propiedades afectan a los elementos con la segunda etiqueta (F) contenidos dentro de la primera etiqueta (E), pero no afecta si hay etiquetas intermedias entre ellos. En el ejemplo siguiente, únicamente los párrafos <p> dentro de una división <div> sin elementos intermedios se ven de color rojo.

```
<body>  
    <p>Esto es un párrafo &lt;p&gt;.</p>  
  
    <div>  
        <p>Esto es un párrafo &lt;p&gt;.</p>  
  
        <ul>  
            <li>Esto es una lista  
                <p>Esto es un párrafo &lt;p&gt; dentro de una lista</p>  
            </li>  
        </ul>  
  
        <p>Esto es otro párrafo &lt;p&gt;.</p>  
  
        <address>Esto no es un párrafo &lt;p&gt;, sino una dirección &lt;address&gt;.</address>  
    </div>  
  
    <p>Esto es otro párrafo &lt;p&gt;.</p>  
</body>
```

```
div {  
    border: red 3px solid;  
    margin: 10px;  
    padding: 10px;  
}  
  
div > p {  
    color: red;  
}
```

Esto es un párrafo <p>.  
Esto es un párrafo <p>.  
• Esto es una lista  
 Esto es un párrafo <p> dentro de una lista  
Esto es otro párrafo <p>.  
Esto no es un párrafo <p>, sino una dirección <address>.  
Esto es otro párrafo <p>.

### Selector de consecutivos: E + F

Si se escriben dos etiquetas seguidas separadas por un signo "más" (E + F), las propiedades afectan únicamente a los elementos con la segunda etiqueta (F) que van justo después de un elemento con la primera etiqueta (E). En el ejemplo siguiente, únicamente el párrafo <p> inmediatamente posterior a la división <div> se ven de color rojo.

```
<body>
  <p>Esto es un párrafo &lt;p&gt;.</p>

  <div>
    <p>Esto es un párrafo &lt;p&gt;.</p>

    <ul>
      <li>Esto es una lista
        <p>Esto es un párrafo &lt;p&gt; dentro de una lista</p>
      </li>
    </ul>

    <p>Esto es otro párrafo &lt;p&gt;.</p>

    <address>Esto no es un párrafo &lt;p&gt;, sino una dirección &lt;address&gt;.</address>
  </div>

  <p>Esto es otro párrafo &lt;p&gt;.</p>

  <p>Esto es otro párrafo &lt;p&gt;.</p>
</body>
```

```
div {
  border: red 3px solid;
  margin: 10px;
  padding: 10px;
}

div + p {
  color: red;
}
```

Esto es un párrafo <p>.

Esto es un párrafo <p>.

- Esto es una lista

Esto es un párrafo <p> dentro de una lista

Esto es otro párrafo <p>.

*Esto no es un párrafo <p>, sino una dirección <address>.*

Esto es otro párrafo <p>.

Esto es otro párrafo <p>.

### Selector de hermanos: E ~ F

Si se escriben dos etiquetas seguidas separadas por un signo "tilde" (E ~ F), las propiedades afectan únicamente a los elementos con la segunda etiqueta (F) que van después de un elemento con la primera etiqueta (E), siendo ambos elementos hermanos, es decir, hijos del mismo parente. En el ejemplo siguiente, se ven de color rojo dos de los párrafos **<p>**. Dentro de la división, el segundo párrafo se muestra de color rojo aunque haya una lista entre ellos. Fuera de la división, el segundo párrafo se muestra de color rojo, mientras que el primero no porque los párrafos de dentro de la división no son sus hermanos.

```
<body>
<div>
    <p>Esto es un párrafo &lt;p&gt;.</p>

    <ul>
        <li>Esto es una lista
            <p>Esto es un párrafo &lt;p&gt; dentro de una lista</p>
        </li>
    </ul>

    <p>Esto es otro párrafo &lt;p&gt;.</p>

    <address>Esto no es un párrafo &lt;p&gt;, sino una dirección &lt;address&gt;.</address>
</div>

<p>Esto es otro párrafo &lt;p&gt;.</p>

<p>Esto es otro párrafo &lt;p&gt;.</p>
</body>
```

```
div {
    border: red 3px solid;
    margin: 10px;
    padding: 10px;
}

div ~ p {
    color: red;
}
```

Esto es un párrafo <p>.  
 • Esto es una lista  
 Esto es un párrafo <p> dentro de una lista  
 Esto es otro párrafo <p>.  
 Esto no es un párrafo <p>, sino una dirección <address>.

Esto es otro párrafo <p>.  
 Esto es otro párrafo <p>.

### Selector de atributo

Hay cuatro formas de seleccionar elementos con determinados atributos: E[atributo], E[atributo="valor"], E[atributo~="valor"], E[atributo|= "valor"].

Si se escribe una etiqueta seguida del nombre de un atributo entre corchetes, E[atributo], las propiedades afectan a todos los elementos que tengan establecidos ese atributo. En el ejemplo siguiente, los párrafos con clase, independientemente de la clase asignada, se muestran en mayúsculas.

```
<body>
<p class="p1">Esto es un párrafo &lt;p&gt; de clase p1.</p>

<p class="p2">Esto es un párrafo &lt;p&gt; de clase p2.</p>

<p>Esto es un párrafo &lt;p&gt; sin clase.</p>
</body>
```

```
p.p1 {
    color: red;
}

p.p2 {
    color: blue;
}

p[class] {
    text-transform: uppercase;
}
```

ESTO ES UN PÁRRAFO <p> DE CLASE P1.  
 ESTO ES UN PÁRRAFO <p> DE CLASE P2.  
 Esto es un párrafo <p> sin clase.

Si se escribe una etiqueta seguida del nombre de un atributo igual a un valor entre corchetes, E[atributo="valor"], las propiedades afectan a todos los elementos que tengan establecidos ese atributo exactamente con ese valor. En el ejemplo siguiente, el párrafo con dos clases (asignadas en el mismo orden en el que aparecen entre corchetes en la hoja de estilo), se muestran en mayúsculas. Además, los párrafos con dos clases se ven de color azul porque las clases no se aplican en el orden en el que aparecen en el atributo clase, sino en el orden en que aparecen en la hoja de estilo.

```
<body>
  <p class="p1">Esto es un párrafo &lt;p&gt; de clase p1.</p>
  <p class="p2">Esto es un párrafo &lt;p&gt; de clase p2.</p>
  <p class="p1 p2">Esto es un párrafo &lt;p&gt; con clase p1 y p2 (en ese orden).</p>
  <p class="p2 p1">Esto es un párrafo &lt;p&gt; con clase p2 y p1 (en ese orden).</p>
</body>
```

```
p.p1 {
  color: red;
}

p.p2 {
  color: blue;
}

p[class="p1 p2"] {
  text-transform: uppercase;
}
```

Esto es un párrafo <p> de clase p1.  
Esto es un párrafo <p> de clase p2.  
ESTO ES UN PÁRRAFO <p> CON CLASE P1 Y P2 (EN ESE ORDEN).  
Esto es un párrafo <p> con clase p2 y p1 (en ese orden).

Si se escribe una etiqueta seguida del nombre de un atributo tilde-igual a un valor entre corchetes, E[atributo~="valor"], las propiedades afectan a todos los elementos que tengan establecidos ese atributo con ese valor (entre otros).

Si se escribe una etiqueta seguida del nombre de un atributo |-igual a un valor entre corchetes, E[atributo|= "valor"], las propiedades afectan a todos los elementos que tengan establecidos ese atributo con un valor que comience por ese valor seguido de un guion -. Este selector está pensado para seleccionar elementos con diferentes variantes de un mismo idioma.

### 1.7.Las pseudo-clases y los pseudo-elementos

Las hojas de estilo asocian características de estilo a los elementos basándose en las etiquetas de los elementos y en su posición relativa (en el árbol del documento). Las pseudo-clases y los pseudo-elementos permiten hacer referencia a determinados elementos sin basarse en la información contenida en el árbol del documento.

La diferencia entre pseudo-clases y pseudo-elementos es sutil. En general, los pseudo-elementos suelen hacer referencia a determinadas partes de un elemento, mientras que las pseudo-clases suelen hacer referencia al estado del elemento.

Otra diferencia importante es que un pseudo-elemento sólo puede aparecer al final de un selector, mientras que una pseudo-clase puede aparecer en cualquier elemento del selector.

En CSS 2, la sintaxis de las pseudo-clases y los pseudo-elementos era la misma, etiqueta:pseudo-elemento-o-pseudo-clase (es decir, el nombre de la etiqueta seguido de dos puntos y del nombre del pseudo-elemento o de la pseudo-clase). En 2011, en la recomendación CSS 3 Selectores (cuya segunda versión [CSS 3 Selectores](#) se publicó en 2018), se modificó la notación de los pseudo-elementos a etiqueta::pseudo-elemento (es decir, el nombre de la etiqueta seguido dos veces de dos puntos y del nombre del pseudo-elemento) para distinguir unos de otros, pero permitiendo utilizar también la antigua notación.

### La pseudo-clase :first-child

La pseudo-clase **:first-child** hace referencia al primer elemento de un tipo contenido dentro de otro. El ejemplo siguiente muestra cómo identificar al primer párrafo dentro de una división sin necesidad de asignar ninguna clase al párrafo.

```
<body>
  <div>
    <p>Este es el primer párrafo &lt;p&gt; en una división que contiene tres párrafos.</p>
    <p>Este es el segundo párrafo &lt;p&gt; en una división que contiene tres párrafos.</p>
    <p>Este es el tercer párrafo &lt;p&gt; en una división que contiene tres párrafos.</p>
  </div>
</body>
```

```
div {
  border: black 5px solid;
  margin: 10px;
  padding: 10px;
}

div p:first-child {
  color: red;
}
```

Este es el primer párrafo <p> en una división que contiene tres párrafos.

Este es el segundo párrafo <p> en una división que contiene tres párrafos.

Este es el tercer párrafo <p> en una división que contiene tres párrafos.

Es necesario que el tipo del primer elemento hijo en la página web sea el indicado en la hoja de estilo. En el ejemplo siguiente, el primer elemento dentro de la división no es un **<p>**, sino un **<pre>**, así que no se le aplica el estilo indicado en la hoja de estilo.

```
<body>
  <div>
    <pre>El primer hijo no es un párrafo &lt;p&gt;, sino un &lt;pre&gt;.</pre>
    <p>Este es el primer párrafo &lt;p&gt; en una división que contiene tres párrafos.</p>
    <p>Este es el segundo párrafo &lt;p&gt; en una división que contiene tres párrafos.</p>
    <p>Este es el tercer párrafo &lt;p&gt; en una división que contiene tres párrafos.</p>
  </div>
</body>
```

```
div {  
    border: black 5px solid;  
    margin: 10px;  
    padding: 10px;  
}  
  
div p:first-child {  
    color: red;  
}
```

El primer hijo no es un párrafo <p>, sino un <pre>. Este es el primer párrafo <p> en una división que contiene tres párrafos. Este es el segundo párrafo <p> en una división que contiene tres párrafos. Este es el tercer párrafo <p> en una división que contiene tres párrafos.

En el ejemplo siguiente, la pseudo-clase **:first-child** se aplica al primer elemento dentro de la división, sea cual sea el elemento, ya que en la hoja de estilo no se especifica el elemento.

```
<body>  
    <div>  
        <pre>El primer hijo no es un párrafo &lt;p&gt;, sino un &lt;pre&gt;.</pre>  
        <p>Este es el primer párrafo &lt;p&gt; en una división que contiene tres párrafos.</p>  
        <p>Este es el segundo párrafo &lt;p&gt; en una división que contiene tres párrafos.</p>  
        <p>Este es el tercer párrafo &lt;p&gt; en una división que contiene tres párrafos.</p>  
    </div>  
</body>  
...
```

```
div {  
    border: black 5px solid;  
    margin: 10px;  
    padding: 10px;  
}  
  
div :first-child {  
    color: red;  
}
```

El primer hijo no es un párrafo <p>, sino un <pre>. Este es el primer párrafo <p> en una división que contiene tres párrafos. Este es el segundo párrafo <p> en una división que contiene tres párrafos. Este es el tercer párrafo <p> en una división que contiene tres párrafos.

### Las pseudo-clases de enlace **:link** y **:visited**

La pseudo clase **:link** (o **a:link**) permite especificar el aspecto de los enlaces que todavía no han sido visitados.

La pseudo clase **:visited** (o **a:visited**) permite especificar el aspecto de los enlaces que sí han sido visitados.

La pseudo-clase **:link** establece el aspecto de los enlaces no visitados, pero si sólo se incluye en la hoja de estilo la pseudo-clase **:link**, el aspecto no cambiará al hacer clic en los enlaces:

```
<body>  
    <p>Este párrafo contiene <a href="#destino">un enlace</a> al párrafo siguiente.</p>  
    <p id="destino">Este párrafo es el destino del enlace anterior.</p>  
</body>
```

```
a:link {  
    background-color: lightblue;  
}
```

Este párrafo contiene **un enlace** al párrafo siguiente.  
Este párrafo es el destino del enlace anterior.

Es necesario incluir las dos pseudo-clases **:link** y **:link** para que el aspecto del enlace cambie al hacer clic en él.

```
<body>
  <p>Este párrafo contiene <a href="#destino">un enlace</a> al párrafo siguiente.</p>
  <p id="destino">Este párrafo es el destino del enlace anterior.</p>
</body>
```

```
a:link {
  background-color: lightblue;
}

a:visited {
  background-color: coral;
}
```

Este párrafo contiene un enlace al párrafo siguiente.  
Este párrafo es el destino del enlace anterior.

Los navegadores "recuerdan" los enlaces visitados hasta que se borra el historial en el navegador.

```
<body>
  <p id="destino-plv1">Este párrafo tiene un <a href="#destino-plv1">enlace a sí mismo</a>. El enlace no aparece como suele ser habitual, porque se muestra en verde.</p>
</body>
```

```
a:link {
  color: green;
}
```

Este párrafo tiene un enlace a sí mismo. El enlace no aparece como suele ser habitual, porque se muestra en verde.

```
<body>
  <p id="destino-plv2">Este párrafo tiene un <a href="#destino-plv2">enlace a sí mismo</a>. Si el navegador reconoce que el enlace se ha visitado, en vez de en morado, se muestra en rojo.</p>
</body>
```

```
a:visited {
  color: red;
}
```

Este párrafo tiene un enlace a sí mismo. Si el navegador reconoce que el enlace se ha visitado, en vez de en morado, se muestra en rojo.

En la recomendación CSS 2 se podía utilizar la pseudo clase **:visited** para modificar cualquier propiedad, pero como una página web maliciosa podía utilizar esta propiedad para saber las páginas que el usuario había visitado, los navegadores han reducido el número de propiedades que se pueden aplicar con esta propiedad. La recomendación CSS 2.1 alerta de ese peligro y permite que los navegadores no permitan modificar ninguna propiedad o sólo admitan algunas propiedades.

Enlaces relacionados: [Privacy and the :visited selector - Plugging the CSS History Leak](#)

Teóricamente, las propiedades que podrían modificarse sin riesgo son las relacionadas con colores. El siguiente ejemplo prueba esas cuatro propiedades. En el ejemplo se establecen antes las propiedades en la etiqueta **<a>** porque si no Firefox y Chrome no la modifican.

```
<body>
  <p id="destino">Este párrafo tiene un <a href="#destino">enlace a sí mismo</a>. Si el navegador reconoce que el enlace se ha visitado, se muestra en rojo y con el fondo amarillo (y con un borde negro y un contorno rojo).</p>
</body>
```

```
a {
  background-color: white;
  color: blue;
  border: orange 3px solid;
  outline: green 3px solid;
}

a:visited {
  background-color: yellow;
  color: red;
  border-color: black;
  outline-color: red;
}
```

Este párrafo tiene un **enlace a sí mismo**. Si el navegador reconoce que el enlace se ha visitado, se muestra en rojo y con el fondo amarillo (y con un borde negro y un contorno rojo).

Si queremos cambiar el aspecto de un enlace, pero sin distinguir que el enlace se haya visitado o no, tenemos entonces tres opciones:

- Definir simultáneamente las dos pseudo-clases:

```
<body>
  <p>Este párrafo contiene <a href="#destino">un enlace</a> al párrafo siguiente.</p>

  <p id="destino">Este párrafo es el destino del enlace anterior.</p>
</body>
```

```
a:link, a:visited {
  background-color: lightblue;
}
```

Este párrafo contiene **un enlace** al párrafo siguiente.  
Este párrafo es el destino del enlace anterior.

- Definir únicamente los enlaces no visitados

```
a:link {
  background-color: lightblue;
}
```

Este párrafo contiene **un enlace** al párrafo siguiente.  
Este párrafo es el destino del enlace anterior.

- Definir los enlaces en general

```
a {
  background-color: lightblue;
}
```

Este párrafo contiene **un enlace** al párrafo siguiente.  
Este párrafo es el destino del enlace anterior.

### Las pseudo-clases dinámicas :hover, :active y :focus

#### *La pseudo-clase dinámica :hover*

La pseudo-clase **:hover** permite especificar el aspecto del elemento sobre el que se encuentra el ratón.

```
<body>
  <p id="destino">Este párrafo está escrito en negro sobre blanco, pero cuando se hace clic en <a href="#destino">este enlace,</a> el enlace se muestra de color blanco sobre fondo negro.</p>
</body>
```

```
p {
  background-color: white;
  color: black;
}

a:hover {
  background-color: black;
  color: white;
}
```

Este párrafo está escrito en negro sobre blanco, pero cuando se hace clic en [este enlace](#), el enlace se muestra de color blanco sobre fondo negro.

```
<body>
  <p>Este párrafo está escrito en negro sobre blanco, pero cuando se pasa el ratón por encima del párrafo, se muestra de color blanco sobre fondo negro.</p>
</body>
```

```
p {
  background-color: white;
  color: black;
}

p:hover {
  background-color: black;
  color: white;
}
```

Este párrafo está escrito en negro sobre blanco, pero cuando se pasa el ratón por encima del párrafo, se muestra de color blanco sobre fondo negro.

### La pseudo-clase dinámica :active

La pseudo-clase **:active** permite especificar el aspecto de un elemento cuando se hace clic sobre él (y mientras se mantiene el botón del ratón apretado).

```
<body>
  <p id="destino">Este párrafo está escrito en negro sobre blanco, pero cuando se hace clic en <a href="#destino">este enlace,</a> el enlace se muestra de color blanco sobre fondo negro.</p>
</body>
```

```
p {
  background-color: white;
  color: black;
}

a:active {
  background-color: black;
  color: white;
}
```

Este párrafo está escrito en negro sobre blanco, pero cuando se hace clic en [este enlace](#), el enlace se muestra de color blanco sobre fondo negro.

```
<body>
  <p>Este párrafo está escrito en negro sobre blanco, pero cuando se hace clic en él, se muestra de color blanco sobre fondo negro.</p>
</body>
```

```
p {
  background-color: white;
  color: black;
}

p:active {
  background-color: black;
  color: white;
}
```

Este párrafo está escrito en negro sobre blanco, pero cuando se hace clic en él, se muestra de color blanco sobre fondo negro.

### La pseudo-clase dinámica :focus

La pseudo-clase **:focus** permite especificar el aspecto de un elemento cuando este tiene el foco. Los elementos que admiten el foco en una página web son aquellos que reaccionan a entrada por teclado (por ejemplo, los elementos de los formularios o los enlaces). En el ejemplo siguiente, al hacer clic en un campo de texto (o cambiar de campo con el tabulador), el campo seleccionado se resalta con un borde rojo.

```
<body>
  <p>
    <input type="text" value="Texto 1" size="8" tabindex="1" class="ifocus">
    <input type="text" value="Texto 2" size="8" tabindex="1" class="ifocus">
    <input type="text" value="Texto 3" size="8" tabindex="1" class="ifocus">
    <input type="text" value="Texto 4" size="8" tabindex="1" class="ifocus">
  </p>
</body>
```

```
input:focus {
  border: red 2px dotted;
  padding: 2px;
}
```

Texto 1    Texto 2    Texto 3    Texto 4



Cuando se utiliza a la vez :link :visited y :hover los navegadores antiguos hacían cada uno una cosa distinta, pero es posible que los modernos no tengan ese problema.

### La pseudo-clase de idioma :lang()

La pseudo-clase de idioma **:lang()** permite especificar el aspecto de los elementos de un idioma determinado. En el ejemplo siguiente se cambia las comillas utilizadas en una cita <q> según el idioma indicado en el atributo **lang**.

```
<p>Ambrose Bierce dijo (más o menos) que <q lang="es">una cita es una manera de repetir erróneamente las palabras de otros</q>.</p>
```

```
<p>Ambrose Bierce dijo (más o menos) que <q lang="en">a quotation is the act of repeating erroneously the words of another</q>.</p>
```

```
q:lang(es) {
  quotes: "<" ">";
}

q:lang(en) {
  quotes: "“” “”";
}
```

Ambrose Bierce dijo (más o menos) que «una cita es una manera de repetir erróneamente las palabras de otros».

Ambrose Bierce dijo (más o menos) que "a quotation is the act of repeating erroneously the words of another".

### Los pseudo-elementos ::first-line y ::first-letter

#### El pseudo-elemento ::first-line

El pseudo-elemento **::first-line** permite especificar el aspecto de la primera línea de texto.

```
<body>
  <p>La primera línea de este párrafo se tiene que ver en mayúsculas. Modifique el tamaño de la fuente o el tamaño de la ventana para ver cómo se modifica el aspecto del párrafo.</p>
</body>
```



```
p::first-line {  
    text-transform: uppercase;  
}
```

LA PRIMERA LINEA DE ESTE PÁRRAFO SE TIENE QUE VER EN MAYÚSCULAS.  
Modifique el tamaño de la fuente o el tamaño de la ventana para ver cómo se modifica el aspecto del párrafo.

En el ejemplo anterior, al hacer zoom o variar el ancho de la ventana se observa únicamente la primera línea de texto está en mayúsculas.

**Nota:** Aunque Chrome muestra la primera línea de texto en mayúsculas, no lo hace correctamente.

*El pseudo-elemento ::first-letter*

El pseudo-elemento **::first-letter** permite especificar el aspecto de la primera letra de texto.

```
<body>  
    <p>La primera letra de este párrafo se tiene que ver en rojo.</p>  
  
    <p>Y la primera de éste también.</p>  
  
    <p>1 Si es número, también se ve en rojo.</p>  
</body>
```

```
p::first-letter {  
    color: red;  
}
```

La primera letra de este párrafo se tiene que ver en rojo.  
Y la primera de éste también.  
1 Si es número, también se ve en rojo.

```
<body>  
    <p>La primera letra de este párrafo es una letra capital, es decir, una letra más grande que ocupa varias líneas de texto. Para ello, en la hoja de estilo tienes que hacer la letra flotante y aumentar su tamaño. En este caso también he cambiado el color y el tipo de letra.</p>  
</body>
```

```
p::first-letter {  
    background-color: pink;  
    color: red;  
    float: left;  
    font-family: monospace;  
    font-size: 400%;  
}
```

L a primera letra de este párrafo es una letra capital, es decir, una letra más grande que ocupa varias líneas de texto. Para ello, en la hoja de estilo tienes que hacer la letra flotante y aumentar su tamaño. En este caso también he cambiado el color y el tipo de letra.

Si hay una imagen al principio del elemento, el aspecto de la primera letra de texto no se modifica.

```
<body>  
    <p> La primera letra de este párrafo no se ve en rojo porque hay una imagen.</p>  
    <p>Pero la primera de éste sí.</p>  
</body>
```

```
p::first-letter {  
    color: red;  
}
```

⚠ La primera letra de este párrafo no se ve en rojo porque hay una imagen.  
Pero la primera de éste sí.

Si la imagen situada al principio del elemento es una imagen flotante, Firefox no modifica el aspecto de la primera letra de texto, aunque Chrome sí lo hace.

```
! p::first-letter {  
    color: red;  
}  
  
! img {  
    float: right;  
}
```

⚠ La primera letra de este párrafo se ve rojo en algunos navegadores y en otros no porque hay una imagen flotante (la imagen se vea a la derecha porque tiene float: right).

Pero la primera de éste sí.

```
! p::first-letter {  
    color: red;  
}  
  
! img {  
    float: left;  
}
```

⚠ La primera letra de este párrafo se ve rojo en algunos navegadores y en otros no porque hay una imagen flotante (la imagen se vea a la izquierda porque tiene float: left).

Pero la primera de éste sí.

Si el primer carácter no es un carácter alfanumérico, el pseudo-elemento se aplica hasta el primer carácter alfanumérico:

```
udy>  
<p>¿Cuántas letras se ven en rojo?</p>  
  
<p>¡En Firefox 1.0.X se ven varias!</p>  
  
<p>¿¿Cuántas letras se ven en rojo??</p>  
  
<p>¿¿¿¿Cuántas letras se ven en rojo????</p>  
  
<p>{Llaves}</p>  
  
<p>{{Llaves}}</p>  
  
<p>#Almohadilla</p>  
  
<p>[Corchetes]</p>  
  
<p>@arroba</p>  
  
<p>@[{}]@</p>  
</body>
```

```
p::first-letter {  
    color: red;  
}
```

¿Cuántas letras se ven en rojo?  
¡En Firefox 1.0.X se ven varias!  
¿¿Cuántas letras se ven en rojo??  
¿¿¿¿Cuántas letras se ven en rojo????  
{Llaves}  
{{Llaves}}  
#Almohadilla  
[Corchetes]  
@arroba  
@[{}]@

### Los pseudo-elementos ::before y ::after

Contenido generado: **content**

Los pseudo-elementos ::before y ::after permiten añadir contenido a un elemento desde la hoja de estilo, al principio o al final del elemento.

El contenido generado no puede seleccionarse con el ratón (para copiarlo y pegarlo en otro documento, por ejemplo).

El contenido generado mediante la propiedad **content** puede incluir texto:

```
<body>
  <p>Este párrafo es un párrafo &lt;p&gt; sin clase.</p>

  <p class="cuidado">Este párrafo es un párrafo &lt;p&gt; con clase "cuidado".</p>

  <p>Este párrafo es un párrafo &lt;p&gt; sin clase.</p>
</body>
```

```
p.cuidado::before {
  content: "Aviso: ";
  font-weight: bold;
  text-decoration: underline;
}
```

Este párrafo es un párrafo <p> sin clase.  
**Aviso:** Este párrafo es un párrafo <p> con clase "cuidado".  
Este párrafo es un párrafo <p> sin clase.

```
<body>
  <p>Este párrafo es un párrafo &lt;p&gt; sin clase.</p>

  <p class="autor-barto">Este párrafo es un párrafo &lt;p&gt; con clase "autor-barto".</p>

  <p>Este párrafo es un párrafo &lt;p&gt; sin clase.</p>
</body>
```

```
p.autor-barto::after {
  content: " (escrito por Barto).";
}
```

Este párrafo es un párrafo <p> sin clase.  
Este párrafo es un párrafo <p> con clase "autor-barto". (escrito por Barto).  
Este párrafo es un párrafo <p> sin clase.

El contenido generado mediante la propiedad **content** puede incluir una imagen indicando su URI:

```
<body>
  <p>Este párrafo es un párrafo &lt;p&gt; sin clase.</p>

  <p class="ff">Este párrafo es un párrafo &lt;p&gt; con clase "ff".</p>

  <p>Este párrafo es un párrafo &lt;p&gt; sin clase.</p>
</body>
```

```
p.ff::before {
  content: url("../icono-ff.svg");
}
```

Este párrafo es un párrafo <p> sin clase.



Este párrafo es un párrafo <p> con clase "ff".

Este párrafo es un párrafo <p> sin clase.

No está permitido incluir etiquetas html en la propiedad **content**. Los navegadores muestran las etiquetas como texto, no como código html.

```
<body>
  <p>Este párrafo es un párrafo &lt;p&gt; sin clase.</p>

  <p class="cuidado">Este párrafo es un párrafo &lt;p&gt; con clase "cuidado2".</p>

  <p>Este párrafo es un párrafo &lt;p&gt; sin clase.</p>
</body>
```

```
p.cuidado::before {
  content: "<em>Aviso:</em> ";
}
```

Este párrafo es un párrafo <p> sin clase.

<em>Aviso:</em> Este párrafo es un párrafo <p> con clase "cuidado2".

Este párrafo es un párrafo <p> sin clase.

*Generar contadores: **counter**, **counter-increment** y **counter-reset***

Se pueden generar contadores en los pseudo-elementos **::after** y **::before** mediante la propiedad **content** y el valor **counter(nombre\_de\_contador)**. El contador debe ponerse a cero con la propiedad **counter-reset** y aumentarse con la propiedad **counter-increment**.

En los ejemplos siguientes se ha definido un contador que se llama cuenta-párrafos. Este contador se genera al principio de cada párrafo, se pone a cero con el elemento **<pre>** y se incrementa en cada párrafo.

```
<body>
  <pre>Ejemplo de contadores (este párrafo es un &lt;pre&gt;)</pre>

  <p>Este párrafo es un párrafo &lt;p&gt; sin clase.</p>

  <p>Este párrafo es un párrafo &lt;p&gt; sin clase.</p>

  <p>Este párrafo es un párrafo &lt;p&gt; sin clase.</p>

  <p>Este párrafo es un párrafo &lt;p&gt; sin clase.</p>

</body>
```

```
pre {
  counter-reset: cuenta-parrafos;
}

p::before {
  content: counter(cuenta-parrafos);
  counter-increment: cuenta-parrafos;
}
```

Ejemplo de contadores (este párrafo es un <pre>)  
1Este párrafo es un párrafo <p> sin clase.  
2Este párrafo es un párrafo <p> sin clase.  
3Este párrafo es un párrafo <p> sin clase.  
4Este párrafo es un párrafo <p> sin clase.  
5Este párrafo es un párrafo s<p> in clase.

```
<body>
  <pre>Ejemplo de contadores (este párrafo es un &lt;pre&gt;)</pre>

  <p>Este párrafo es un párrafo &lt;p&gt; sin clase.</p>

  <p>Este párrafo es un párrafo &lt;p&gt; sin clase.</p>

  <p>Este párrafo es un párrafo &lt;p&gt; sin clase.</p>

  <pre>Ejemplo de contadores (este párrafo es un &lt;pre&gt;)</pre>

  <p>Este párrafo es un párrafo &lt;p&gt; sin clase.</p>

  <p>Este párrafo es un párrafo &lt;p&gt; sin clase.</p>

</body>
```

```
pre {
  counter-reset: cuenta-parrafos;
}

p::before {
  content: counter(cuenta-parrafos);
  counter-increment: cuenta-parrafos;
}
```

Ejemplo de contadores (este párrafo es un <pre>)  
1Este párrafo es un párrafo <p> sin clase.  
2Este párrafo es un párrafo <p> sin clase.  
3Este párrafo es un párrafo <p> sin clase.  
Ejemplo de contadores (este párrafo es un <pre>)  
1Este párrafo es un párrafo <p> sin clase.  
2Este párrafo es un párrafo <p> sin clase.

El contador puede incrementarse en cualquier cantidad entera, como muestra el ejemplo siguiente:

```
.body>
<pre>Ejemplo de contadores (este párrafo es un <pre>)</pre>
<p>Este párrafo es un párrafo <p> sin clase.</p>
<p>Este párrafo es un párrafo <p> in clase.</p>
</body>
```

```
pre {
  counter-reset: cuenta-parrafos;
}

p::before {
  content: counter(cuenta-parrafos);
  counter-increment: cuenta-parrafos 10;
}
```

Ejemplo de contadores (este párrafo es un <pre>  
10Este párrafo es un párrafo <p> sin clase.  
20Este párrafo es un párrafo <p> sin clase.  
30Este párrafo es un párrafo <p> sin clase.  
40Este párrafo es un párrafo <p> sin clase.  
50Este párrafo es un párrafo <p> in clase.

Junto con el contador se puede generar texto, como muestra el ejemplo siguiente:

```
<body>
<pre>Ejemplo de contadores (este párrafo es un <pre>)</pre>
<p>Este párrafo es un párrafo <p> normal y corriente.</p>
<p>Este párrafo es un párrafo <p> normal y corriente.</p>
<p>Este párrafo es un párrafo <p> normal y corriente.</p>
<pre>Ejemplo de contadores (este párrafo es un <pre>)</pre>
<p>Este párrafo es un párrafo <p> normal y corriente.</p>
<p>Este párrafo es un párrafo <p> normal y corriente.</p>
</body>
```

```
pre {  
    counter-reset: cuenta-parrafos;  
}  
  
p::before {  
    content: counter(cuenta-parrafos)", ";  
    counter-increment: cuenta-parrafos;  
}
```

Ejemplo de contadores (este párrafo es un <pre>)

1. Este párrafo es un párrafo <p> normal y corriente.
  2. Este párrafo es un párrafo <p> normal y corriente.
  3. Este párrafo es un párrafo <p> normal y corriente.
- Ejemplo de contadores (este párrafo es un <pre>)
1. Este párrafo es un párrafo <p> normal y corriente.
  2. Este párrafo es un párrafo <p> normal y corriente.

El contador predeterminado muestra números enteros, pero se puede utilizar cualquiera de los [estilos de listas](#) como estilo de contador. El estilo se indica en la propiedad **content** mediante el valor **counter(nombre\_de\_contador, estilo\_de\_lista)**, como muestra el ejemplo siguiente:

```
<body>  
    <pre>Ejemplo de contadores (este párrafo es un &lt;pre&gt;)</pre>  
  
    <p>Este párrafo es un párrafo &lt;p&gt; normal y corriente.</p>  
  
    <p>Este párrafo es un párrafo &lt;p&gt; normal y corriente.</p>  
  
    <p>Este párrafo es un párrafo &lt;p&gt; normal y corriente.</p>  
  
    <pre>Ejemplo de contadores (este párrafo es un &lt;pre&gt;)</pre>  
  
    <p>Este párrafo es un párrafo &lt;p&gt; normal y corriente.</p>  
  
    <p>Este párrafo es un párrafo &lt;p&gt; normal y corriente.</p>  
</body>
```

```
pre {  
    counter-reset: cuenta-parrafos;  
}  
  
p::before {  
    content: counter(cuenta-parrafos, upper-roman)" . ";  
    counter-increment: cuenta-parrafos;  
}
```

Ejemplo de contadores (este párrafo es un <pre>)

- I. Este párrafo es un párrafo <p> normal y corriente.
  - II. Este párrafo es un párrafo <p> normal y corriente.
  - III. Este párrafo es un párrafo <p> normal y corriente.
- Ejemplo de contadores (este párrafo es un <pre>)
- I. Este párrafo es un párrafo <p> normal y corriente.
  - II. Este párrafo es un párrafo <p> normal y corriente.

Se pueden utilizar varios contadores simultáneamente, como muestra el ejemplo siguiente:

```

<body>
  <h1>Ejemplo de contadores</h1>

  <h2>Apartado 1</h2>

  <p>Este párrafo es un párrafo &lt;p&gt; normal y corriente.</p>
  <p>Este párrafo es un párrafo &lt;p&gt; normal y corriente.</p>
  <p>Este párrafo es un párrafo &lt;p&gt; normal y corriente.</p>

  <h2>Apartado 2</h2>

  <p>Este párrafo es un párrafo &lt;p&gt; normal y corriente.</p>
  <p>Este párrafo es un párrafo &lt;p&gt; normal y corriente.</p>
</body>

```

```

h1 {
  counter-reset: cuenta-parrafos;
  counter-reset: cuenta-apartados;
}

h2 {
  counter-reset: cuenta-parrafos;
}

h2::before {
  content: counter(cuenta-apartados, upper-alpha) ". ";
  counter-increment: cuenta-apartados;
}

p::before {
  content: counter(cuenta-apartados, upper-alpha)"-"counter(cuenta-parrafos)"";
  counter-increment: cuenta-parrafos;
}

```

## Ejemplo de contadores

### A. Apartado 1

- A-1. Este párrafo es un párrafo <p> normal y corriente.
- A-2. Este párrafo es un párrafo <p> normal y corriente.
- A-3. Este párrafo es un párrafo <p> normal y corriente.

### B. Apartado 2

- B-1. Este párrafo es un párrafo <p> normal y corriente.
- B-2. Este párrafo es un párrafo <p> normal y corriente.

## 1.8.Adecuación de las hojas de estilos.

Las hojas de estilo se llaman hojas de estilo "en cascada" porque:

- Las propiedades de estilo pueden estar escritas en varios sitios (en varios lugares de la página web o de la hoja de estilo) y dependiendo del sitio, afectan a más o menos elementos.
- Cuando un elemento está contenido en otro (por ejemplo, un párrafo <p> dentro de una división <div>), al elemento de dentro se le aplican también las propiedades definidas para el elemento de fuera (al párrafo <p> se le aplicarían las propiedades definidas para la división <div>).
- Dos reglas distintas pueden ser de aplicación a un mismo elemento (por ejemplo, a un párrafo <p> con clase **nombre**, le es de aplicación tanto el selector **.nombre** como el selector **p**).

Si las propiedades (escritas en diferentes sitios o para diferentes elementos) no entran en conflicto, el navegador aplica todas las propiedades. Por ejemplo, si el color de fondo de un elemento está definido en un sitio y el tamaño de letra en otro sitio, el navegador aplicará ambas propiedades al elemento.

Pero si las propiedades entran en conflicto (por ejemplo, el color del fondo del elemento está definido en varios sitios con colores distintos), existen reglas para decidir qué propiedad tiene preferencia.

### Elementos dentro de otros

Cuando un elemento está contenido en otro (por ejemplo, un párrafo `<p>` dentro de una división `<div>`), al elemento de dentro se le aplican también las propiedades definidas para el elemento de fuera.

En el ejemplo siguiente, el párrafo incluido dentro de la división se muestra de color rojo porque el párrafo está incluido dentro de la división.

```
<div>
  <p>Esto es un párrafo dentro de una división.</p>
</div>
```

Esto es un párrafo dentro de una división.

```
div {
  border: black 3px solid; color: red;
}
```

Si una misma propiedad está definida para el elemento inferior y para el superior, se aplica el valor establecido para el elemento inferior.

En el ejemplo siguiente, el párrafo se muestra de color negro, independientemente del orden en que se escriban las reglas.

```
<body>
  <div>
    <p>Esto es un párrafo dentro de una división.</p>
  </div>
</body>
```

```
div {
  border: black 3px solid;
  color: red;
}

p {
  color: black;
}
```

Esto es un párrafo dentro de una división.

```
<body>
  <div>
    <p>Esto es un párrafo dentro de una división.</p>
  </div>
</body>
```

```
p {
  color: black;
}

div {
  border: black 3px solid;
  color: red;
}
```

Esto es un párrafo dentro de una división.

### Reglas distintas que se aplican al mismo elemento

Dos reglas distintas se aplican a un mismo elemento cuando el elemento coincide con los selectores de ambas reglas. La regla que se aplica es la del selector de mayor especificidad. La especificidad de un selector se calcula atendiendo a los siguientes criterios:

1. Número de atributos **id** en el selector
2. Número de otros atributos y pseudo-clases en el selector (los pseudo-elementos se ignoran)
3. Número de elementos en el selector
4. Posición en la hoja de estilo

Estos criterios se aplican en orden, es decir, primero se comparan el número de atributos **id** de cada selector. Si un selector tiene más que el otro, se aplica esa regla, si el número es el mismo, entonces se calcula el segundo criterio (número de otros atributos y pseudo-clases). Y así sucesivamente.

A continuación, se muestran varios ejemplos de aplicación de estas reglas.

#### Número de atributos **id** en el selector

El ejemplo siguiente muestra dos selectores con un número de atributos **id** diferente (1 o 0). Puede comprobarse que se aplica siempre el selector con atributo **id**, independientemente del orden en que aparezcan en la hoja de estilo.

```
<p id="nuevo">Esto es un párrafo con atributo id "nuevo"</p>
<p>Esto es un párrafo sin atributo id.</p>
```

```
p#nuevo {
    color: red;
}

p {
    color: black;
}
```

Esto es un párrafo con atributo id "nuevo"  
Esto es un párrafo sin atributo id.

```
p {
    color: black;
}

p#nuevo {
    color: red;
}
```

Esto es un párrafo con atributo id "nuevo"  
Esto es un párrafo sin atributo id.

El ejemplo siguiente muestra también dos selectores con un número de atributos **id** diferente (2 o 1). Puede comprobarse que se aplica siempre el selector con más atributos **id**, independientemente del orden en que aparezcan en la hoja de estilo.

```
<body>
    <div id="viejo">
        <p id="nuevo">Esto es un párrafo con atributo id "nuevo" dentro de una división con atributo id "viejo"</p>
    </div>

    <p>Esto es un párrafo sin atributo id.</p>
</body>
```

```
div#viejo {
    border: black 3px solid;
    margin: 2px;
    padding: 2px;
}

div p#nuevo {
    color: red;
}

div#viejo p#nuevo {
    color: black;
}
```

Esto es un párrafo con atributo id "nuevo" dentro de una división con atributo id "viejo"

Esto es un párrafo sin atributo id.

```
div#viejo {
    border: black 3px solid;
    margin: 2px;
    padding: 2px;
}

div#viejo p#nuevo {
    color: black;
}

div p#nuevo {
    color: red;
}
```

Esto es un párrafo con atributo id "nuevo" dentro de una división con atributo id "viejo"

Esto es un párrafo sin atributo id.

### Número de otros atributos y pseudo-clases en el selector

El ejemplo siguiente muestra dos selectores con el mismo número de atributos **id** (0), pero con un número diferente de atributos **class** (1 o 0). Puede comprobarse que se aplica siempre el selector con más atributos **class**, independientemente del orden en que aparezcan en la hoja de estilo.

```
<body>
    <div class="viejo">
        <p>Esto es un párrafo dentro de una división con atributo class "viejo".</p>
    </div>

    <p>Esto es un párrafo que no está dentro de ninguna división.</p>
</body>
```

```
div.viejo {
    border: black 3px solid;
    margin: 2px;
    padding: 2px;
}

div.viejo p {
    color: red;
}

div p {
    color: black;
}
```

Esto es un párrafo dentro de una división con atributo class "viejo".

Esto es un párrafo que no está dentro de ninguna división.

```
div.viejo {
    border: black 3px solid;
    margin: 2px;
    padding: 2px;
}

div p {
    color: black;
}

div.viejo p {
    color: red;
}
```

Esto es un párrafo dentro de una división con atributo class "viejo".

Esto es un párrafo que no está dentro de ninguna división.

### Número de elementos en el selector

El ejemplo siguiente muestra dos selectores con un número diferente de elementos (2 o 1). Puede comprobarse que se aplica siempre el selector con más elementos, independientemente del orden en que aparezcan en la hoja de estilo.

```
<body>
  <div>
    <p>Esto es un párrafo dentro de una división.</p>
  </div>

  <p>Esto es un párrafo que no está dentro de ninguna división.</p>
</body>
```

```
div {
  border: black 3px solid;
  margin: 2px;
  padding: 2px;
}

div p {
  color: red;
}

p {
  color: black;
}
```

Esto es un párrafo dentro de una división.

Esto es un párrafo que no está dentro de ninguna división.

```
div {
  border: black 3px solid;
  margin: 2px;
  padding: 2px;
}

p {
  color: black;
}

div p {
  color: red;
}
```

Esto es un párrafo dentro de una división.

Esto es un párrafo que no está dentro de ninguna división.

El ejemplo siguiente también muestra dos selectores con un número diferente de elementos (2 o 1). Puede comprobarse que se aplica siempre el selector con más elementos, independientemente del orden en que aparezcan en la hoja de estilo.

```
<body>
  <ul>
    <li>Esto es un elemento de lista</li>
  </ul>
</body>
```

```
li {  
    color: black;  
}  
  
ul li {  
    color: red;  
}
```

• Esto es un elemento de lista

```
ul li {  
    color: red;  
}  
  
li {  
    color: black;  
}
```

• Esto es un elemento de lista

El ejemplo siguiente también muestra dos selectores con un número diferente de elementos (1 o 0). Puede comprobarse que se aplica siempre el selector con más elementos, independientemente del orden en que aparezcan en la hoja de estilo.

```
<body>  
    <p class="nuevo">Esto es un párrafo de clase "nuevo".</p>  
  
    <p>Esto es un párrafo sin clase.</p>  
</body>
```

```
p.nuevo {  
    color: black;  
}  
  
.nuevo {  
    color: red;  
}
```

Esto es un párrafo de clase "nuevo".

Esto es un párrafo sin clase.

### Posición en la hoja de estilo

Si a un elemento le afectan dos selectores con la misma especificidad, el navegador aplica la propiedad que aparece después en la hoja de estilo.

El ejemplo siguiente muestra dos selectores idénticos. Puede comprobarse que se aplica siempre el selector que aparece después en la hoja de estilo. En este caso, la penúltima regla es superflua y puede eliminarse sin afectar al resultado.

```
<body>  
    <div>  
        <p>Esto es un párrafo dentro de una división.</p>  
    </div>  
</body>
```

```
div {  
    border: black 3px solid;  
}  
  
div p {  
    color: red;  
}  
  
div p {  
    color: black;  
}
```

Esto es un párrafo dentro de una división.

```
div {  
    border: black 3px solid;  
}  
  
div p {  
    color: black;  
}  
  
div p {  
    color: red;  
}
```

Esto es un párrafo dentro de una división.

### Posición en la hoja de estilo

Si a un elemento le afectan dos selectores con la misma especificidad, el navegador aplica la propiedad que aparece después en la hoja de estilo.

El ejemplo siguiente muestra dos selectores idénticos. Puede comprobarse que se aplica siempre el selector que aparece después en la hoja de estilo. En este caso, la penúltima regla es superflua y puede eliminarse sin afectar al resultado.

```
<body>  
    <div>  
        <p>Esto es un párrafo dentro de una división.</p>  
    </div>  
</body>
```

```
div {  
    border: black 3px solid;  
}  
  
div p {  
    color: red;  
}  
  
div p {  
    color: black;  
}
```

Esto es un párrafo dentro de una división.

```
div {  
    border: black 3px solid;  
}  
  
div p {  
    color: black;  
}  
  
div p {  
    color: red;  
}
```

Esto es un párrafo dentro de una división.

El ejemplo siguiente muestra una situación diferente y más habitual. Se trata de un párrafo con dos clases. Puede comprobarse que se aplica siempre el selector que aparece después en la hoja de estilo.

```
<body>
  <p class="viejo nuevo">Esto es un párrafo con atributo class "viejo" y "nuevo"</p>
</body>
```

```
p.viejo {
  color: red;
}

p.nuevo {
  color: black;
}
```

Esto es un párrafo con atributo class "viejo" y "nuevo"

```
p.nuevo {
  color: black;
}

p.viejo {
  color: red;
}
```

Esto es un párrafo con atributo class "viejo" y "nuevo"

### 1.9.Unidades. Distancias y tamaños

En una página web o en una hoja de estilo se pueden definir las distancias o tamaños como porcentajes o como valores numéricos (absolutos o relativos).

- los porcentajes se escriben con el símbolo de porcentaje pegado al valor numérico (sin espacio entre ellos). Por ejemplo, 100%, 50%, etc.
- los valores numéricos pueden ser positivos o negativos, enteros o decimales (utilizando el punto como separador de las partes entera y decimal). Las unidades deben ir pegadas a los valores numéricos (sin espacio entre ellos). Por ejemplo, 10px, 0.5cm, -10pt, etc.
- cuando el valor es cero, se aconseja no escribir la unidad (por ejemplo, escribir 0 en vez de 0px)

Las unidades relativas definidas respecto al tipo de letra son:

- **em**: el tamaño de fuente del elemento contenedor
- **ex**: la altura de la letra x de la fuente del elemento contenedor
- **ch**: la altura de la cifra 0 (cero) de la fuente del elemento contenedor
- **rem**: el tamaño de la fuente del elemento raíz de la página

En la práctica, las unidades relativas más utilizadas son **em** y **rem**.

Las unidades relativas definidas respecto a la pantalla son:

- **vw**: el 1% del ancho de la ventana (sin contar bordes, pestanas barras de desplazamiento, etc.)
- **vh**: el 1% del alto de ventana (sin contar bordes, pestanas barras de desplazamiento, etc.)
- **vmin**: el menor de los tamaños **vw** y **vh**
- **vmax**: el mayor de los tamaños **vw** y **vh**

Las unidades absolutas disponibles son:

- **px**: píxeles, la 96ava parte de una pulgada (96 px = 1in = 2.54cm)
- **cm**: centímetros
- **mm**: milímetros
- **Q**: cuartos de milímetros

- **in:** pulgadas ( $1\text{in} = 2.54\text{cm} = 25.4\text{mm}$ )
- **pc:** picas, la sexta parte de una pulgada ( $1\text{in} = 6\text{pc}$ )
- **pt:** puntos, la doceava parte de una pica, es decir, la 72ava parte de una pulgada ( $1\text{in} = 6\text{pc} = 72\text{pt}$ )

En la práctica, las unidades absolutas más utilizadas son **px** y **pt**.

### Unidades relativas

Entre las unidades relativas, es necesario distinguir la unidad **rem** del resto de unidades relativas o de los porcentajes.

### Porcentajes %

Cuando se emplean porcentajes para expresar el tamaño del tipo de letra, los valores superiores a 100% significan una ampliación y los valores inferiores a 100% significan reducción

```
<body>
  <p>Este párrafo está aumentado al 200% (el doble del tamaño normal).</p>
</body>
```

```
p {
  font-size: 200%;
}
```

Este párrafo está aumentado al 200% (el doble del tamaño normal).

```
p {
  font-size: 75%;
}
```

Este párrafo está reducido al 75% del tamaño normal.

Cuando se emplean porcentajes para expresar distancias o tamaños de elementos, los porcentajes se interpretan con respecto al elemento contenedor (cuando se refieren a distancias horizontales), o respecto al elemento contenedor o el tamaño de la ventana (cuando se refieren a distancias verticales).

```
hr {
  width: 50%;
}
```

Este separador ocupa el 50% del espacio horizontal disponible:

Debido a que las propiedades se aplican en cascada, al cambiar el tamaño de un elemento, los elementos contenidos en él también modifican su tamaño. En el ejemplo siguiente, el párrafo situado en la división **<div>** aumenta su tamaño al 150%

```
<p>Párrafo fuera de una división.</p>
<div>
  <p>Párrafo dentro de una división.</p>
</div>
```

```
div {
  border: black 2px solid;
  font-size: 150%;
}
```

Párrafo fuera de una división.

Párrafo dentro de una división.

Si se cambia el tamaño en dos elementos anidados, los porcentajes se multiplican entre sí. A este fenómeno se le denomina "composición", es decir, el tamaño final es la composición de tamaños de cada elemento.

En el ejemplo siguiente, los párrafos están aumentados al 150%, por lo que el párrafo situado en la división (cuyo contenido está a su vez aumentado al 150%), está aumentado el 225% (el 150% del 150%,  $150\% \times 150\% = 225\%$ ).

```
<p>Este párrafo está aumentado al 150%.</p>
<div>
  <p>Este párrafo está aumentado al 150%.</p>
</div>

div {
  border: black 1px solid;
  font-size: 150%;
}

p {
  font-size: 150%;
}
```

Este párrafo está aumentado al 150%.

Este párrafo está aumentado al 150%.

Esta composición de tamaños no siempre es conveniente y puede provocar cambios de tamaños indeseados debido a anidamientos no previstos. Para evitarlo, se puede utilizar la unidad relativa **rem**, introducida en CSS 3.

En el ejemplo siguiente, las líneas horizontales ocupan el 50% del espacio disponible, por lo que la línea situada en la división (que a su vez ocupa el 50% del espacio disponible), ocupa en realidad el 25% del total (el 50% del 50%,  $50\% \times 50\% = 25\%$ ).

```
<body>
  <p>Este separador ocupa el 50% del espacio horizontal disponible:</p>
  <hr>

  <div>
    <p>Esta división ocupa el 50% del espacio horizontal disponible.</p>
    <p>Este separador ocupa el 50% del espacio horizontal disponible:</p>
    <hr>
  </div>
</body>
```

```
div {
  width: 50%;
  border: black 1px solid;
}

hr {
  width: 50%;
```

Este separador ocupa el 50% del espacio horizontal disponible:

Esta división ocupa el 50% del espacio horizontal disponible.

Este separador ocupa el 50% del espacio horizontal disponible:

### Unidades relativas respecto al tipo de letra **em**, **ex** y **ch**

Las unidades relativas **em**, **ex** y **ch** permiten expresar el tamaño en función de la altura de un carácter:

- **em** corresponde a la altura total del tipo de letra

- **ex** corresponde a la altura del carácter "x"
- **ch** corresponde a la altura del carácter "0"

En el ejemplo siguiente, los párrafos tienen un borde con un grosor expresado en **em**, **ex** y **ch**.

```
<body>
  <p class="borde-em">Este párrafo tiene un borde de 1em de grosor (la altura total del tipo de letra).</p>
  <p class="borde-ex">Este párrafo tiene un borde de 1ex de grosor (la altura de la letra "x").</p>
  <p class="borde-ch">Este párrafo tiene un borde de 1ch de grosor (la altura de la cifra "0").</p>
</body>
```

```
p.borde-em {
  border: red 1em solid;
}

p.borde-ex {
  border: green 1ex solid;
}

p.borde-ch {
  border: blue 1ch solid;
}
```

Este párrafo tiene un borde de 1em de grosor (la altura total del tipo de letra).

Este párrafo tiene un borde de 1ex de grosor (la altura de la letra "x").

Este párrafo tiene un borde de 1ch de grosor (la altura de la cifra "0").

En estas unidades, el tamaño depende del tipo de letra que se esté utilizando. En los ejemplos siguientes, los párrafos tienen un borde con un grosor **1em**. Como cada fuente tiene una altura total diferente, el grosor del borde es distinto en cada caso.

```
<body>
  <p class="borde-em-serif">Este párrafo tiene un borde de 1em de grosor (la altura total del tipo de letra).</p>
  <p class="borde-em-sans-serif">Este párrafo tiene un borde de 1em de grosor (la altura total del tipo de letra).</p>
  <p class="borde-em-monospace">Este párrafo tiene un borde de 1em de grosor (la altura total del tipo de letra).</p>
</body>
```

```
p.borde-em-serif {
  border: red 1em solid;
  font-family: serif;
}

p.borde-em-sans-serif {
  border: green 1em solid;
  font-family: sans-serif;
}

p.borde-em-monospace {
  border: blue 1em solid;
  font-family: monospace;
}
```

Este párrafo tiene un borde de 1em de grosor (la altura total del tipo de letra).

Este párrafo tiene un borde de 1em de grosor (la altura total del tipo de letra).

Este párrafo tiene un borde de 1em de grosor (la altura total del tipo de letra).

La unidad relativa **em** se puede relacionar directamente con los porcentajes, concretamente **1em = 100%**.

Con las unidades relativas **em**, **ex** y **ch** se produce también el fenómeno de composición comentado en el apartado anterior sobre porcentajes, es decir que si se cambia el tamaño en dos elementos anidados, los porcentajes se multiplican entre sí

```
<p>Este párrafo está aumentado a 1.5em.</p>
<div>
  <p>Este párrafo está aumentado a 1.5em.</p>
</div>
```

```
div {
  border: black 2px solid;
  font-size: 1.5em;
}

p {
  font-size: 1.5em;
}
```

Este párrafo está aumentado a 1.5em.

Este párrafo está aumentado a 1.5em.

Esta composición de tamaños no siempre es conveniente y puede provocar cambios de tamaños indeseados debido a anidamientos no previstos. Para evitarlo, se puede utilizar la unidad relativa **rem**, introducida en CSS 3.

### La unidad relativa respecto al tipo de letra rem

La unidad relativa **rem** corresponde a la altura total del tipo de letra, como **em**, pero con la diferencia que la altura se interpreta siempre con respecto al elemento raíz de la página.

Por tanto, con la unidad relativa **rem** no se produce el fenómeno de composición de tamaños en elementos anidados, como muestra el ejemplo siguiente, en el que los párrafos dentro de la división se ven del mismo tamaño que los párrafos de fuera de la división:

```
<p>Este párrafo está aumentado a 1.5rem.</p>
<div>
  <p>Este párrafo está aumentado a 1.5rem.</p>
</div>
```

```
div {
  border: black 2px solid;
  font-size: 1.5rem;
}

p {
  font-size: 1.5rem;
}
```

Este párrafo está aumentado a 1.5rem.

Este párrafo está aumentado a 1.5rem.

### Unidades relativas respecto al tamaño de la pantalla vw, vh, vmin y vmax

Las unidades relativas **vw** y **vh** corresponden, respectivamente, al 1% del ancho y alto total disponibles para el documento.

```
<p>Este párrafo ocupa la cuarta parte de la pantalla disponible.</p>
<p>Este párrafo ocupa la cuarta parte de la pantalla disponible.</p>
```

```
p {
  background-color: lightgray;
  width: 50vw;
  height: 50vh;
}
```

Este párrafo ocupa la cuarta parte de la pantalla disponible.

Este párrafo ocupa la cuarta parte de la pantalla disponible.

Las unidades relativas **vmin** y **vmax** corresponden, respectivamente, al menor y mayor de los valores **vw** y **vh**.

```
<p class="min">Este párrafo cuadrado ocupa la mitad del lado más corto.</p>
<p class="max">Este párrafo cuadrado ocupa la mitad del lado más largo.</p>
```

```
p.min {
  background-color: lightgray;
  width: 50vmin;
  height: 50vmin;
}

p.max {
  background-color: lightgray;
  width: 50vmax;
  height: 50vmax;
}
```

Este párrafo cuadrado ocupa la mitad del lado más corto.

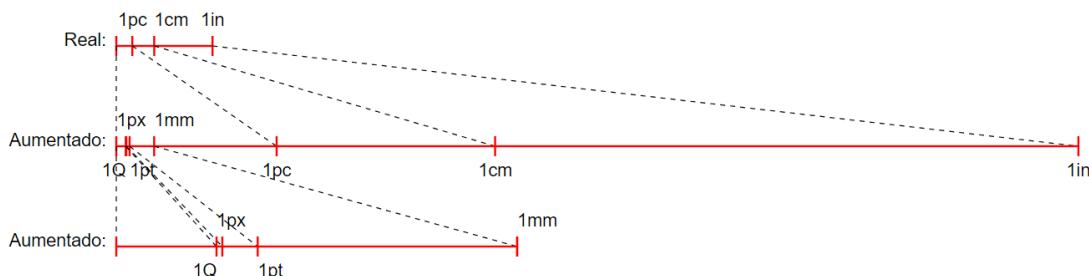
Este párrafo cuadrado ocupa la mitad del lado más largo.

### Unidades absolutas

Todas las unidades absolutas están relacionadas, puesto que se trata de unidades de distancia:

- **px**: píxeles, la 96ava parte de una pulgada ( $96 \text{ px} = 1\text{in} = 2.54\text{cm}$ )
- **cm**: centímetros
- **mm**: milímetros ( $10\text{mm} = 1\text{cm}$ )
- **Q**: cuartos de milímetros ( $4\text{Q} = 1\text{mm}$ )
- **in**: pulgadas ( $1\text{in} = 2.54\text{cm} = 25.4\text{mm}$ )
- **pc**: picas, la sexta parte de una pulgada ( $6\text{pc} = 1\text{in}$ )
- **pt**: puntos, la doceava parte de una pica, es decir, la 72ava parte de una pulgada ( $72\text{pt} = 6\text{pc} = 1\text{in}$ )

La imagen siguiente muestra estas distancias a tamaño real y aumentado:



### Redefinición de la unidad **px** en CSS 3

En CSS 2 el píxel (**px**) era una unidad relativa, ya que se refería al tamaño de un píxel físico de una pantalla. Cuando se redactó CSS 2, en los años 90, los píxeles de las pantallas tenían más o menos el mismo tamaño en todas las pantallas, por lo que al definir elementos en **px**, el resultado era más o menos el mismo en todas las pantallas.

El problema es que desde hace unos años existen pantallas de alta densidad (sobre todo en los móviles) en los que los píxeles físicos son mucho más pequeños que en las pantallas de los años 90. En estas pantallas, los elementos definidos en **px** se ven mucho más pequeños, lo que convierte en ilegibles las páginas.

Este problema se ha resuelto en CSS 3 redefiniendo la unidad **px**, convirtiéndola en una distancia absoluta, independiente del tamaño del píxel de la pantalla. Para mantener cierta compatibilidad con la definición antigua, se ha tenido en cuenta que las pantallas de los años 90 tenían una densidad de píxeles de 96dpi (96dpi = 96 dots per inch = 96 puntos por pulgada), definiendo el nuevo **px** de manera que 96 píxeles miden 1 pulgada ( $96\text{px} = 1\text{in}$ ).

### Unidades recomendadas en estos apuntes

La elección de unidades es algo muy personal, pero se suelen utilizar mayoritariamente las siguientes unidades:

- **px** para distancias pequeñas (anchos de bordes, márgenes, etc.).
- **rem** para tamaños de fuente, salvo que se quiera aprovechar en el diseño la composición de tamaños, en cuyo caso se usarán porcentajes **%**.
- **vw, vh** y porcentajes **%** para tamaños de elementos grandes y distribución de elementos por la pantalla.

## 1.10. Colores

La recomendación [CSS 3 Color \(2º ed\)](#) (junio de 2018) describe los sistemas de definición de colores (nombres o códigos numéricicos) que se pueden utilizar en las hojas de estilo, así como la forma de definir la opacidad de los elementos.

Los colores se pueden expresar mediante varios sistemas de unidades: RGB, RGBA, HSL y HSLA, así como mediante nombre de colores (básicos o extendidos)

Se explicará únicamente cómo se hace referencia a un color en una hoja de estilo. Para encontrar combinaciones de colores visualmente atractivas, se puede recurrir a sitios como [Paletton](#) o [Coolors](#).

### Códigos de colores RGB

Los códigos de colores RGB ya formaban parte de la recomendación [CSS 1](#) (diciembre de 1996).

Las pantallas de ordenador consiguen los colores mezclando tres colores básicos (**rojo**, **verde** y **azul**). Cada color admite 256 niveles de intensidad, lo que hace un total de  $256 \times 256 \times 256 = 16.777.216$  colores distintos. Para hacer referencia a un color concreto, basta con indicar las intensidades de cada uno de los tres colores básicos.

Los códigos RGB se pueden expresar de distintas formas:

- **rgb(rojo, verde, azul)**, donde rojo, verde y azul son números enteros desde 0 a 255.
- **rgb(rojo, verde, azul)**, donde rojo, verde y azul son porcentajes desde 0% hasta 100%.
- **#RRGGBB**, donde RR, GG y BB son números hexadecimales desde 00 hasta FF.
- **#RGB**, donde R, G y B son números hexadecimales desde 0 hasta F (el navegador transforma esos números en números de dos cifras repitiendo el valor, es decir, F se convierte en FF, 6 en 66, etc.)

Las más utilizadas son la tercera y la primera.

```
<body>
  <p>Si algo puede ir mal, irá mal.</p>
</body>
```

```
p {  
background-color: #14B76E;  
}
```

Si algo pude ir mal, irá mal.

El color negro se consigue con la ausencia de cualquier color, así que su código RGB es `rgb(0, 0, 0)`, `rgb(0%, 0%, 0%)`, `#000000` o `#000`.

El color blanco se consigue mezclando los tres colores con la máxima intensidad, así que su código RGB es `rgb(255, 255, 255)`, `rgb(100%, 100%, 100%)`, `#FFFFFF` o `FFF`.

En las siguientes páginas encontrarás información sobre cómo obtener códigos de color expresados en diferentes formatos:

- <https://htmlcolorcodes.com/es/>
- <https://www.shopify.es/blog/62615749-14-generadores-de-paletas-de-color-online-para-diseno-web>

### Códigos de colores RGBA

Los códigos de colores RGBA se incorporaron a CSS en la primera edición (junio de 2011) de la recomendación [CSS 3 Color \(2º ed\)](#) (junio de 2018).

Estos códigos incorporan la transparencia alpha a los [códigos RGB](#) mediante un cuarto valor que indica la transparencia del elemento. La transparencia se expresa como un número decimal entre 0 y 1, en el que el 0 significa completamente transparente y el 1 completamente opaco.

Los códigos RGBA se pueden expresar de distintas formas:

- **rgba(rojo, verde, azul, transparencia)**, donde rojo, verde y azul son números enteros desde 0 a 255 y transparencia un número decimal entre 0 y 1.
- **rgba(rojo, verde, azul, transparencia)**, donde rojo, verde y azul son porcentajes desde 0% hasta 100% y transparencia un número decimal entre 0 y 1.

En la siguiente página puedes encontrar un generador de códigos RGBA:

- <https://www.css3maker.com/css-3-rgba.html>

### Códigos de colores HSL

Los códigos de colores HSL se incorporaron a CSS en la primera edición (junio de 2011) de la recomendación [CSS 3 Color \(2º ed\)](#) (junio de 2018).

En el [modelo de color HSL](#) el color se define mediante tres valores:

- Hue (Matiz) es un entero entre 0 y 360 y recorre todos los colores



- Saturation (Saturación) es un porcentaje que define la intensidad del color.
- Lightness (Luminosidad) es un porcentaje que indica la claridad u oscuridad del color.

El modelo HSL es más intuitivo que el modelo RGB y su uso probablemente se generalice en el futuro.

En la siguiente página puedes encontrar varios generadores de color entre ellos HSL:

- <http://colorizer.org/>

### Códigos de colores HSLA

Los códigos de colores HSLA se incorporaron a CSS en la primera edición (junio de 2011) de la recomendación [CSS 3 Color \(2º ed\)](#) (junio de 2018).

Estos códigos incorporan la transparencia alpha a los [códigos HSL](#) mediante un cuarto valor que indica la transparencia del elemento. La transparencia se expresa como en el caso de los códigos RGBA con un número decimal entre 0 y 1, en el que el 0 significa completamente transparente y el 1 completamente opaco.

En la siguiente página puedes encontrar un generador de códigos HSLA:

- <https://hslpicker.com/#8f0,0.45>

### Nombres de colores HTML 3.2

En la recomendación [HTML 3.2](#) (aprobada el 14 de enero de 1997) se incluyeron dieciséis nombres de colores: aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white y yellow. Estos colores son los colores de la paleta VGA de Windows. La tabla e ilustración siguientes muestran estos 16 colores, sus nombres y códigos RGB.

Color	Nombre	Código RGB	Color	Nombre	Código RGB
	blue	#0000FF		navy	#000080
	fuchsia	#FF00FF		purple	#800080
	red	#FF0000		maroon	#800000
	yellow	#FFFF00		olive	#808000
	lime	#00FF00		green	#008000
	aqua	#00FFFF		teal	#008080
	black	#000000		gray	#808080
	white	#FFFFFF		silver	#C0C0C0

### Nombres de colores SVG / X11

Los códigos de colores SVG (o X11) se incorporaron a CSS en la primera edición (junio de 2011) de la recomendación [CSS 3 Color \(2º ed\)](#) (junio de 2018), aunque los navegadores los aceptaban desde hacía tiempo. Se muestran a continuación los 147 colores, sus nombres y códigos RGB y HSL.

### Notas:

- Los colores Fuchsia y Magenta son el mismo color y los colores Aqua y Cyan son también el mismo color, por lo que la tabla contiene en realidad 138 colores distintos.
- También están duplicados los nombres de los colores grises, que pueden incluir los términos "gray" o "grey" (DarkGrey/DarkGray, etc). En X11 sólo era válida una de las dos variantes, pero en la recomendación SVG 1.0 ya se admitían ambas variantes.

AliceBlue #F0F8FF rgb(240, 248, 255) hsl(208, 100%, 97.1%)	AntiqueWhite #FAEBD7 rgb(250, 235, 215) hsl(34, 77.8%, 91.2%)	Aqua #00FFFF rgb(0, 255, 255) hsl(180, 100%, 50%)	Aquamarine #7FFFDD rgb(127, 255, 212) hsl(160, 100%, 74.9%)	Azure #F0FFFF rgb(240, 255, 255) hsl(180, 100%, 97.1%)
Beige #F5F5DC rgb(245, 245, 220) hsl(60, 55.6%, 91.2%)	Bisque #FFE4C4 rgb(255, 228, 196) hsl(33, 100%, 88.4%)	Black #000000 rgb(0, 0, 0) hsl(0, 0%, 0%)	BlanchedAlmond #FFEB3B rgb(255, 235, 205) hsl(36, 100%, 90.2%)	Blue #0000FF rgb(0, 0, 255) hsl(240, 100%, 50%)
BlueViolet #8A2BE2 rgb(138, 43, 226) hsl(271, 75.9%, 52.7%)	Brown #A52A2A rgb(165, 42, 42) hsl(30, 59.4%, 40.6%)	BurlyWood #DEB887 rgb(222, 184, 135) hsl(34, 56.9%, 70%)	CadetBlue #5F9EA0 rgb(95, 158, 160) hsl(182, 25.5%, 50%)	Chartreuse #7FFF00 rgb(127, 255, 0) hsl(90, 100%, 50%)
Chocolate #D691E rgb(210, 105, 30) hsl(25, 75%, 47.1%)	Coral #FF7F50 rgb(255, 127, 80) hsl(16, 100%, 65.7%)	CornflowerBlue #6498ED rgb(100, 149, 237) hsl(219, 79.2%, 66.1%)	Cornsilk #FFF8DC rgb(255, 248, 220) hsl(48, 100%, 93.1%)	Crimson #DC143C rgb(220, 20, 60) hsl(348, 83.3%, 47.1%)
Cyan #00FFFF rgb(0, 255, 255) hsl(180, 100%, 50%)	DarkBlue #00008B rgb(0, 0, 139) hsl(240, 100%, 27.3%)	DarkCyan #00888B rgb(0, 139, 139) hsl(180, 100%, 27.3%)	DarkGoldenrod #B8860B rgb(184, 134, 111) hsl(43, 88.7%, 38.2%)	DarkGray #A9A9A9 rgb(169, 169, 169) hsl(0, 0%, 66.3%)
DarkGreen #006400 rgb(0, 100, 0) hsl(120, 100%, 19.6%)	DarkGrey #A9A9A9 rgb(169, 169, 169) hsl(0, 0%, 66.3%)	DarkKhaki #BDB76B rgb(189, 183, 107) hsl(56, 38.3%, 58%)	DarkMagenta #80008B rgb(139, 0, 139) hsl(300, 100%, 27.3%)	DarkOliveGreen #556B2F rgb(85, 107, 47) hsl(182, 39%, 30.2%)
DarkOrange #FF8C00 rgb(255, 140, 0) hsl(33, 100%, 50%)	DarkOrchid #9932CC rgb(153, 50, 204) hsl(280, 60.6%, 49.8%)	DarkSlateBlue #483D8B rgb(72, 61, 139) hsl(248, 39%, 39.2%)	DarkSlateGray #2F4F4F rgb(47, 79, 79) hsl(180, 25.4%, 24.7%)	DarkSalmon #E9967A rgb(233, 150, 122) hsl(15, 71.6%, 69.6%)
DarkSlateBlue #483D8B rgb(72, 61, 139) hsl(248, 39%, 39.2%)	DarkSlateGray #2F4F4F rgb(47, 79, 79) hsl(180, 25.4%, 24.7%)	DarkSlateGray #2F4F4F rgb(47, 79, 79) hsl(180, 25.4%, 24.7%)	DarkTurquoise #00CED1 rgb(0, 206, 209) hsl(181, 100%, 41%)	DarkViolet #9400D3 rgb(148, 0, 211) hsl(282, 100%, 41.4%)
DeepPink #FF1493 rgb(255, 20, 147) hsl(328, 100%, 53.9%)	DeepSkyBlue #00BFBB rgb(0, 191, 255) hsl(195, 100%, 50%)	DimGray #696969 rgb(105, 105, 105) hsl(0, 0%, 41.2%)	DimGrey #696969 rgb(105, 105, 105) hsl(0, 0%, 41.2%)	DodgerBlue #1E90FF rgb(30, 144, 255) hsl(210, 100%, 55.9%)
FireBrick #B22222 rgb(178, 34, 34) hsl(0, 67.9%, 41.6%)	FloralWhite #FFFFFF rgb(255, 250, 240) hsl(40, 100%, 97.1%)	ForestGreen #228B22 rgb(34, 139, 34) hsl(120, 60.7%, 33.9%)	Fuchsia #FF00FF rgb(255, 0, 255) hsl(300, 100%, 50%)	Gainsboro #DCDCDC rgb(220, 220, 220) hsl(0, 0%, 86.3%)
GhostWhite #F8F8FF rgb(248, 248, 255) hsl(240, 100%, 98.6%)	Gold #FFD700 rgb(255, 215, 0) hsl(51, 100%, 50%)	Goldenrod #DAA520 rgb(218, 165, 32) hsl(43, 74.4%, 49%)	Gray #808080 rgb(128, 128, 128) hsl(0, 0%, 50.2%)	Green #008000 rgb(0, 128, 0) hsl(120, 100%, 25.1%)
GreenYellow #ADFF2F rgb(173, 255, 47) hsl(84, 100%, 59.2%)	Grey #808080 rgb(128, 128, 128) hsl(0, 0%, 50.2%)	Honeydew #F0FFF0 rgb(240, 255, 240) hsl(120, 100%, 97.1%)	HotPink #FF69B4 rgb(255, 105, 180) hsl(330, 100%, 70.6%)	IndianRed #CD5C5C rgb(205, 92, 92) hsl(0, 53.1%, 58.2%)
Indigo #4A0082 rgb(75, 0, 130) hsl(275, 100%, 25.5%)	Ivory #FFFFFF rgb(255, 255, 240) hsl(60, 100%, 97.1%)	Khaki #F0E68C rgb(240, 230, 140) hsl(54, 76.9%, 74.5%)	Lavender #E6E6FA rgb(230, 230, 250) hsl(240, 66.7%, 94.1%)	LavenderBlush #FFF0F5 rgb(255, 240, 245) hsl(340, 100%, 97.1%)
LawnGreen #7CF00 rgb(124, 252, 0) hsl(90, 100%, 49.4%)	LemonChiffon #FFFACD rgb(255, 250, 205) hsl(54, 100%, 90.2%)	LightBlue #ADD8E6 rgb(173, 216, 230) hsl(195, 53.3%, 79%)	LightCoral #F08080 rgb(240, 128, 128) hsl(15, 78.9%, 72.2%)	LightCyan #00FFFF rgb(224, 255, 255) hsl(180, 100%, 93.9%)
LightGoldenrodYellow #FAFAD2 rgb(250, 250, 210) hsl(60, 80%, 90.2%)	LightGray #D3D3D3 rgb(211, 211, 211) hsl(0, 0%, 82.7%)	LightGreen #90EE90 rgb(144, 238, 144) hsl(120, 73.4%, 74.9%)	LightGrey #D3D3D3 rgb(211, 211, 211) hsl(0, 0%, 82.7%)	LightPink #FFB6C1 rgb(255, 182, 193) hsl(351, 100%, 85.7%)
LightSalmon #FFA07A rgb(255, 160, 122) hsl(17, 100%, 73.9%)	LightSeaGreen #20B2AA rgb(32, 178, 170) hsl(177, 69.5%, 41.2%)	LightSkyBlue #87CEFA rgb(135, 206, 250) hsl(203, 92%, 75.5%)	LightStateGray #778899 rgb(119, 136, 153) hsl(210, 14.3%, 53.3%)	LightSlateGrey #778899 rgb(119, 136, 153) hsl(210, 14.3%, 53.3%)
LightSteelBlue #B0C4DE rgb(176, 196, 222) hsl(214, 41.1%, 78%)	LightYellow #FFFFE0 rgb(255, 255, 224) hsl(60, 100%, 93.9%)	Lime #00FF00 rgb(0, 255, 0) hsl(120, 100%, 50%)	LimeGreen #32CD32 rgb(50, 205, 50) hsl(120, 60.8%, 50%)	Linen #FAF0E6 rgb(250, 240, 230) hsl(30, 66.7%, 94.1%)

Magenta #F00FF rgb(255, 0, 255) hsl(300, 100%, 50%)	Maroon #800000 rgb(128, 0, 0) hsl(0, 100%, 25.1%)	MediumAquamarine #66CDAA rgb(102, 205, 170) hsl(160, 50.7%, 60.2%)	MediumBlue #0000CD rgb(0, 0, 205) hsl(240, 100%, 40.2%)	MediumOrchid #BA55D3 rgb(186, 85, 211) hsl(288, 58.9%, 58%)
MediumPurple #800080 rgb(128, 0, 128) hsl(260, 59.8%, 64.9%)	MediumSeaGreen #3CB371 rgb(60, 179, 113) hsl(147, 49.8%, 46.9%)	MediumSlateBlue #7B68EE rgb(123, 104, 238) hsl(249, 79.8%, 67.1%)	MediumSpringGreen #00FA9A rgb(0, 250, 154) hsl(157, 100%, 49%)	MediumTurquoise #48D1CC rgb(72, 209, 204) hsl(178, 59.8%, 55.1%)
MediumVioletRed #C71585 rgb(199, 21, 133) hsl(322, 80.9%, 43.1%)	MidnightBlue #191970 rgb(25, 25, 112) hsl(240, 63.5%, 26.9%)	MintCream #F5FFF4 rgb(255, 255, 250) hsl(150, 100%, 98%)	MistyRose #FFE4E1 rgb(255, 228, 225) hsl(6, 100%, 94.1%)	Moccasin #F0E68C rgb(255, 228, 181) hsl(38, 100%, 85.5%)
NavajoWhite #FFDEAD rgb(255, 222, 173) hsl(36, 100%, 83.9%)	Navy #000080 rgb(0, 0, 128) hsl(240, 100%, 25.1%)	OldLace #FDF5E6 rgb(253, 245, 230) hsl(39, 85.2%, 94.7%)	Olive #808000 rgb(128, 128, 0) hsl(60, 100%, 25.1%)	OliveDrab #6B8E23 rgb(107, 142, 35) hsl(80, 60.5%, 34.7%)
Orange #FFA500 rgb(255, 165, 0) hsl(39, 100%, 50%)	OrangeRed #FF4500 rgb(255, 69, 0) hsl(16, 100%, 50%)	Orchid #DA70D6 rgb(218, 112, 214) hsl(302, 58.9%, 64.7%)	PaleGoldenrod #EEE8AA rgb(238, 232, 170) hsl(55, 66.7%, 80%)	PaleGreen #98FB98 rgb(152, 251, 152) hsl(120, 92.5%, 79%)
PaleTurquoise #AFEEEE rgb(175, 238, 238) hsl(180, 64.9%, 81%)	PaleVioletRed #DB7093 rgb(219, 112, 147) hsl(340, 59.8%, 64.9%)	PapayaWhip #FFFFD5 rgb(255, 239, 213) hsl(37, 100%, 91.8%)	PeachPuff #FFDAB9 rgb(255, 218, 185) hsl(28, 100%, 86.3%)	Peru #CD853F rgb(205, 133, 63) hsl(30, 58.7%, 52.5%)
Pink #FFC0CB rgb(255, 192, 203) hsl(350, 100%, 87.6%)	Plum #DDA0DD rgb(221, 160, 221) hsl(187, 51.9%, 79.6%)	PowderBlue #B0E0E6 rgb(176, 224, 230) hsl(187, 50.3%, 79.6%)	Purple #800080 rgb(128, 0, 128) hsl(300, 100%, 25.1%)	Red #FF0000 rgb(255, 0, 0) hsl(0, 100%, 50%)
RosyBrown #BC8F8F rgb(188, 143, 143) hsl(0, 25.1%, 64.9%)	RoyalBlue #4169E1 rgb(65, 105, 225) hsl(225, 72.7%, 56.9%)	SaddleBrown #8B4513 rgb(139, 69, 19) hsl(25, 75.9%, 31%)	Salmon #FA8072 rgb(250, 128, 114) hsl(6, 93.2%, 71.4%)	SandyBrown #F4A460 rgb(244, 164, 96) hsl(28, 87.1%, 66.7%)
SeaGreen #2E8B57 rgb(46, 139, 87) hsl(146, 50.3%, 36.3%)	Seashell #FFF5EE rgb(255, 245, 238) hsl(25, 100%, 96.7%)	Sienna #A0522D rgb(160, 82, 45) hsl(19, 56.1%, 40.2%)	Silver #C0C0C0 rgb(192, 192, 192) hsl(0, 0%, 75.3%)	SkyBlue #87CEEB rgb(135, 206, 235) hsl(197, 71.4%, 72.5%)
SlateBlue #6A5ACD rgb(106, 90, 205) hsl(248, 53.5%, 57.8%)	SlateGray #708090 rgb(112, 128, 144) hsl(210, 12.6%, 50.2%)	SlateGrey #708090 rgb(112, 128, 144) hsl(210, 12.6%, 50.2%)	Snow #FFFFFF rgb(255, 250, 250) hsl(0, 100%, 99%)	SpringGreen #00FF7F rgb(0, 255, 127) hsl(150, 100%, 50%)
SteelBlue #4682B4 rgb(70, 130, 180) hsl(207, 44%, 49%)	Tan #D2B48C rgb(210, 180, 140) hsl(34, 43.8%, 68.6%)	Teal #008080 rgb(0, 128, 128) hsl(180, 100%, 25.1%)	Thistle #D8bfd8 rgb(216, 191, 216) hsl(300, 24.3%, 79.8%)	Tomato #F06347 rgb(255, 99, 71) hsl(9, 100%, 63.9%)
Turquoise #40E0D0 rgb(64, 224, 208) hsl(174, 72.1%, 56.5%)	Violet #EE82EE rgb(238, 130, 238) hsl(300, 76.1%, 72.2%)	Wheat #F5DEB3 rgb(245, 222, 179) hsl(39, 76.7%, 83.1%)	White #FFFFFF rgb(255, 255, 255) hsl(0, 0%, 100%)	WhiteSmoke #F5F5F5 rgb(245, 245, 245) hsl(0, 0%, 96.1%)
Yellow #FFFF00 rgb(255, 255, 0) hsl(60, 100%, 50%)	YellowGreen #9ACD32 rgb(154, 205, 50) hsl(80, 60.8%, 50%)			

### Nombres de colores del sistema

Los colores del sistema se introdujeron en la recomendación [CSS 2](#) y hacen referencia a los colores utilizados por el interfaz del sistema operativo.

Los navegadores interpretaban e interpretan correctamente estos nombres de colores, pero en estos apuntes estos colores ya no se comentan en esta lección, sino en la [lección de elementos obsoletos](#), porque en la primera edición (junio de 2011) de la recomendación [CSS 3 Color \(2º ed\)](#) (junio de 2018) estos nombres pasaron a considerarse obsoletos, en favor de la propiedad [appearance](#).

El problema es que la propiedad [appearance](#) no está incluida en ninguna recomendación ya aprobada. Esta propiedad se encuentra definida únicamente para elementos de formularios en la futura recomendación [CSS 4: Interface de usuario básico](#) (octubre de 2019) en elaboración, pero parece que esta propiedad tiene serios problemas de incompatibilidad entre navegadores.

Mientras los navegadores los admitan, los nombres de colores de sistema se pueden utilizar, pero nada garantizada que los navegadores dejen de aceptarlos en un futuro.

### Opacidad (transparencia): [opacity](#)

La propiedad [opacity](#) permite hacer que un elemento sea parcialmente transparente. El valor de esta propiedad debe ser un valor decimal entre 0 y 1 (el valor 0 hace que el elemento sea completamente transparente y el valor 1 hace que el elemento sea completamente opaco).

Si un elemento parcialmente transparente se superpone a otro elemento de otro color, el navegador mezcla los colores de los elementos, como muestra el siguiente ejemplo:

```
p.p1 {  
    background-color: yellow;  
}  
  
p.p2 {  
    background-color: red;  
    color: white;  
    opacity: 0.5;  
}  
  
/* En el ejemplo también se utilizan propiedades de posicionamiento que no aparecen aquí. */
```

## Esquemas de colores

Los esquemas de colores (también llamadas paletas de colores) son combinaciones de varios colores. Los esquemas de colores pueden generarse manualmente o matemáticamente a partir de un color básico y de un círculo cromático.

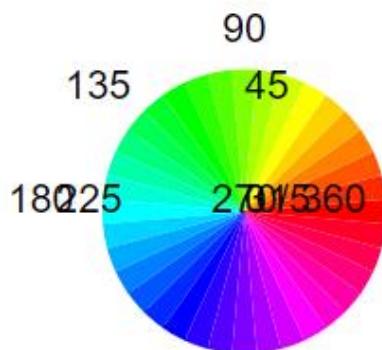
## *Círculos cromáticos*

Los círculos cromáticos son círculos en los que cada sector circular tiene un color distinto. Existen distintos círculos cromáticos, en función de la distribución de los colores a lo largo del círculo.

## Círculo cromático RGB o HSL

El círculo cromático RGB es el círculo en el que los colores rojo, verde y azul se encuentran uniformemente distribuidos. Este círculo corresponde también a la [representación HSL](#), en el que el valor del matiz (desde 0 hasta 360) coincide con el ángulo (rojo =  $0^\circ$ , verde =  $120^\circ$  y azul =  $240^\circ$ ).

Una manera de elegir combinaciones de colores es elegir colores que se encuentren distribuidos de una forma regular en el círculo cromático a partir de un color base. Algunos ejemplos de combinaciones son las siguientes:



- Complementarios: el color base y el situado al lado opuesto del círculo
  - Tríada de colores: el color base y los dos que se encuentran situados en los vértices de un triángulo equilátero
  - Análogos: el color base y los dos que se encuentran a 30 grados del color base

- Complementarios partidos: el color base y los dos que se encuentran a 30 grados del complementario
- Dobles complementarios: el color base, el que se encuentra a 30 grados del color base y sus dos complementarios



Complementarios



Tríada



Análogos



Compl. partidos

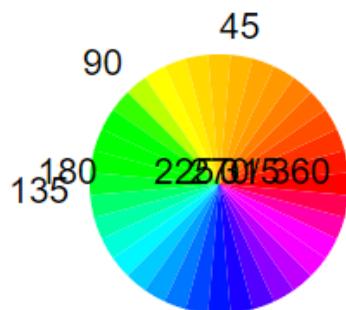


Dobles complem.

#### Círculo cromático Adobe

El generador de esquemas de colores de Adobe, [Adobe color wheel](#), utiliza un círculo cromático ligeramente distinto al RGB, en el que los colores se encuentran distribuidos en el mismo orden pero en el que los tonos entre el rojo y el amarillo ocupan más espacio.

Al crear las combinaciones de colores mediante su distribución en el círculo cromático, las combinaciones de colores que se obtienen a partir de este círculo no son las mismas que las que se obtienen a partir del círculo RGB:



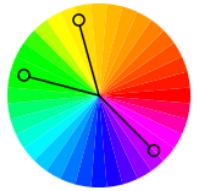
Complementarios



Tríada



Análogos



Compl. partidos



Dobles complem.

En la siguiente página puedes encontrar diferentes generadores de paletas:

- <https://paletadecolores.online/>

## 1.11. Adaptación de los ficheros de estilo para distintas páginas web y dispositivos móviles.

Las consultas de medios permiten detectar características del dispositivo que está utilizando el usuario y optimizar la hoja de estilo.

#### Diseño adaptativo (responsive design)

Al diseñar una página web, se debe tener en cuenta la diversidad de dispositivos en los que se va a ver esa página web:

- la mayoría de páginas web se ven en pantallas, pero también hay que pensar en las impresoras, en los lectores de pantallas, etc.

- la diversidad en las resoluciones de las pantallas va en aumento: pantallas de escritorio cada vez más grandes, pantallas de móviles mucho más pequeñas que las de escritorio, pantallas de tabletas de tamaños intermedios entre unas y otras
- las pantallas de escritorio están siempre en la misma posición, pero los móviles y tabletas pueden cambiar fácilmente de orientación
- las pantallas de escritorio suelen tener una densidad de píxeles de 96 píxeles por pulgada ([DPI](#)), pero los dispositivos móviles pueden tener densidades mucho más altas (en 2016 ya había móviles con pantallas de más de 800 DPI)

En los últimos años se ha acuñado el término *responsive web design* para referirse a las técnicas de diseño que se preocupan por tratar esta diversidad. En español *responsive web design* se suele traducir por **diseño web adaptativo**.

Las consultas de medios (media queries) de las hojas de estilo que se comentan en esta lección son una herramienta básica para un diseño adaptativo.

---

### Herramientas:

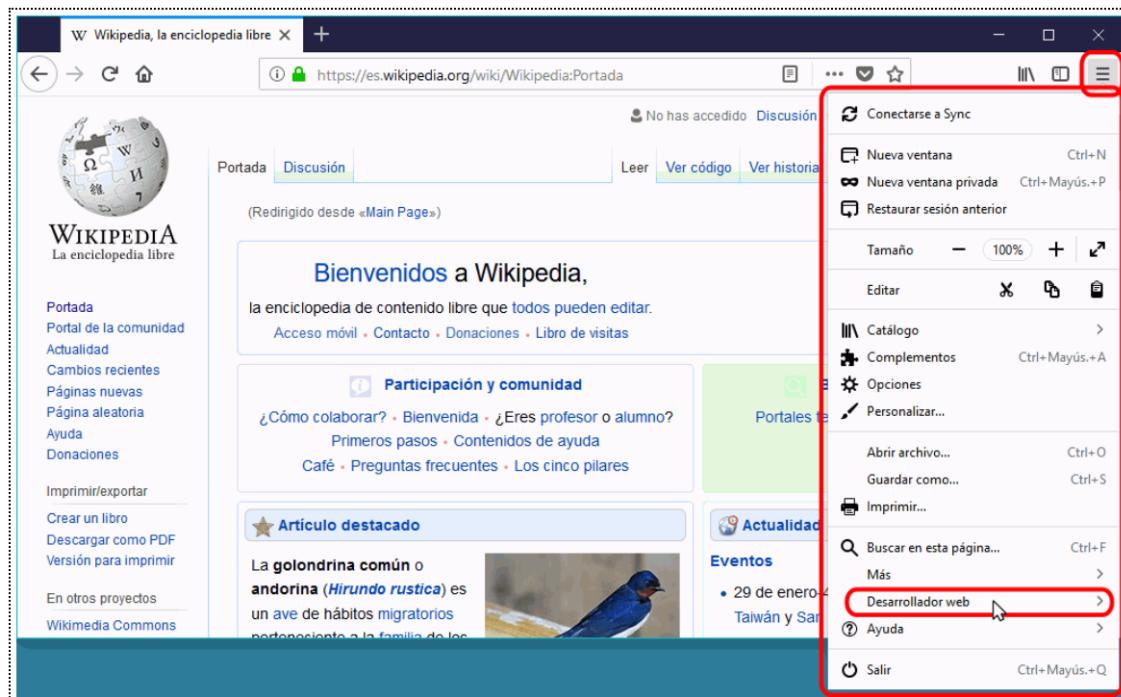
- Chrome > Más herramientas > Herramientas para desarrolladores > Device mode (Ctrl+Shift+M, o ícono móvil)
- Extensión Web Developer de Firefox > Resize > View Responsive Layouts
- [Prueba de optimización para móviles de Google](#)
- [Responsinator](#)
- Firefox > Herramientas > Desarrollador web > Vista de diseño adaptable (Ctrl+Shift+M). Ampliar con

Se comentan las herramientas de desarrollador web incluidas en la versión estándar de Firefox. Existe también una versión especial de Firefox llamada [Developer Edition](#), que está basada en Firefox Beta y que tiene además una configuración optimizada para las herramientas de desarrolladores (y alguna herramienta experimental todavía no disponible en la versión estándar de Firefox).

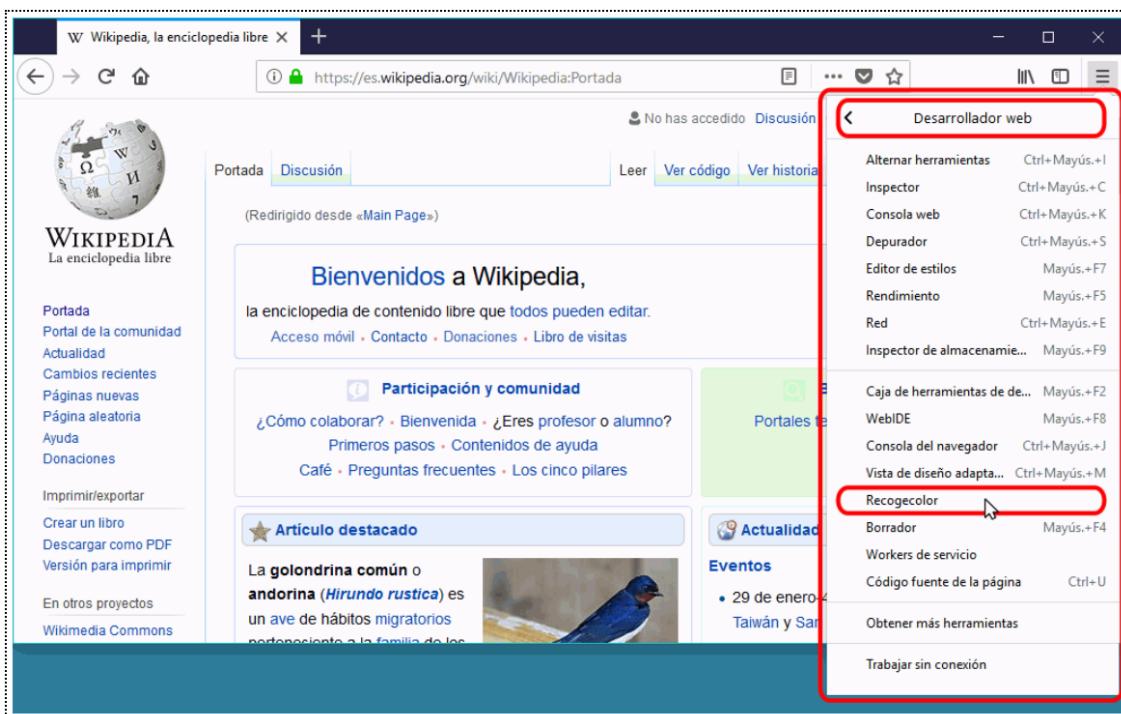
#### *Selector de color*

Un selector de color permite obtener el código RGB de los píxeles de la pantalla. En Firefox 31 (publicado en julio de 2014) se añadió un selector de color a las herramientas para desarrollador web.

Para abrir el selector de color, abra el menú de herramientas y seleccione la opción Desarrollador web ...



... en la lista de herramientas, seleccione la opción Recogecolor:



El cursor se convierte en una lupa que muestra el color del píxel central



Al hacer clic, el valor del color en formato RGB se copia en el portapapeles y se cierra el selector de color.



### Notas:

- Más información sobre el selector de color en la página dedicada al [Selector de color en Mozilla Developer Network](#).
- La extensión ColorZilla incluye también un [selector de color](#) más sofisticado.
- El selector de color (eyedropper) no tiene un atajo de teclado asociado: [bug 1177108](#).
- El selector de color no permite copiar los códigos de color en otros formatos que no sea RGB.

### Herramientas para Desarrollador Web

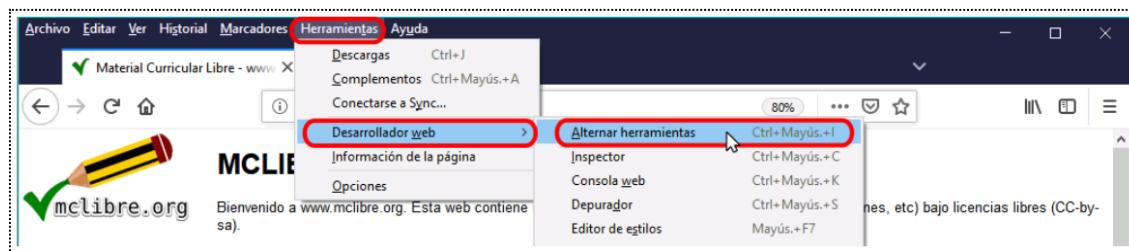
Desde Firefox 16 (publicado en octubre de 2012), Firefox incluye una ventana de **Herramientas para Desarrollador Web** que facilita el análisis técnico de las páginas web. Estas herramientas son similares a las que ofrecen Chrome y Edge. Estas herramientas son herederas de una extensión de Firefox llamada [Firebug](#), que desde 2017 ya no está en desarrollo.

En el blog Mozilla Hacks se pueden consultar las [novedades sobre estas herramientas](#), que se van incorporando en cada nueva versión de Firefox.

### Abrir la ventana de Herramientas para desarrollador web

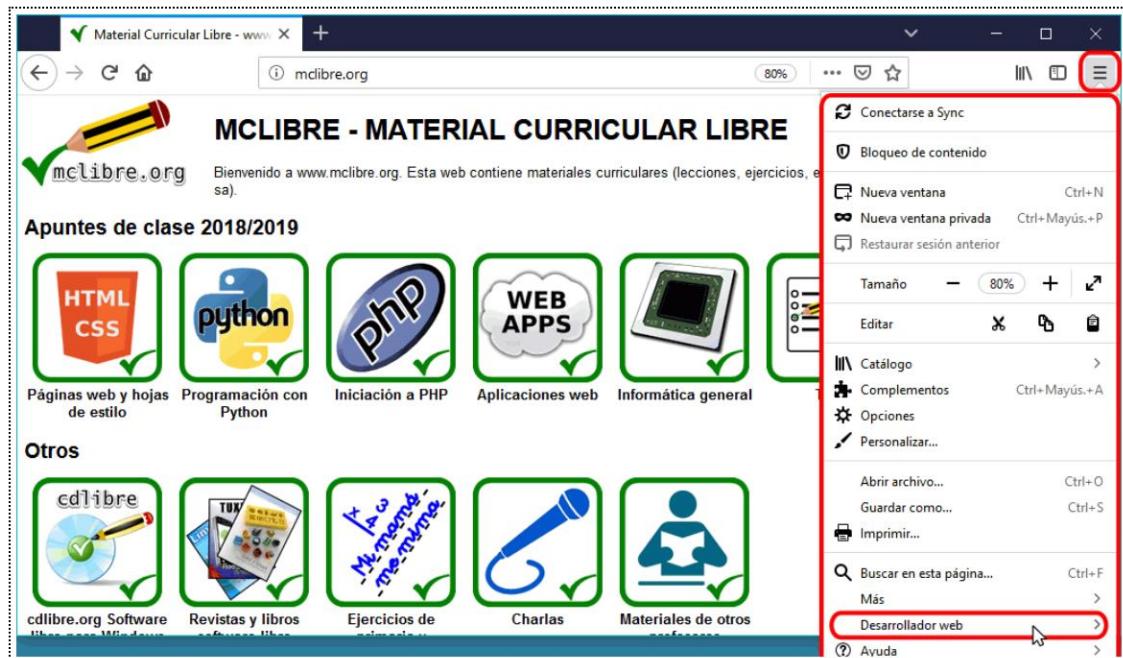
La ventana de Herramientas para desarrollador web se puede abrir de varias maneras:

- mediante el atajo de teclado **Ctrl+May+I** o la tecla **F12**
- mediante el menú **Herramientas > Desarrollador web > Alternar herramientas** (para mostrar el menú, pulse la tecla **Alt**) podemos abrir tanto la ventana de herramientas como herramientas individuales:



mediante el botón "Abrir menú" de la barra de dirección y eligiendo la opción "Desarrollador web"

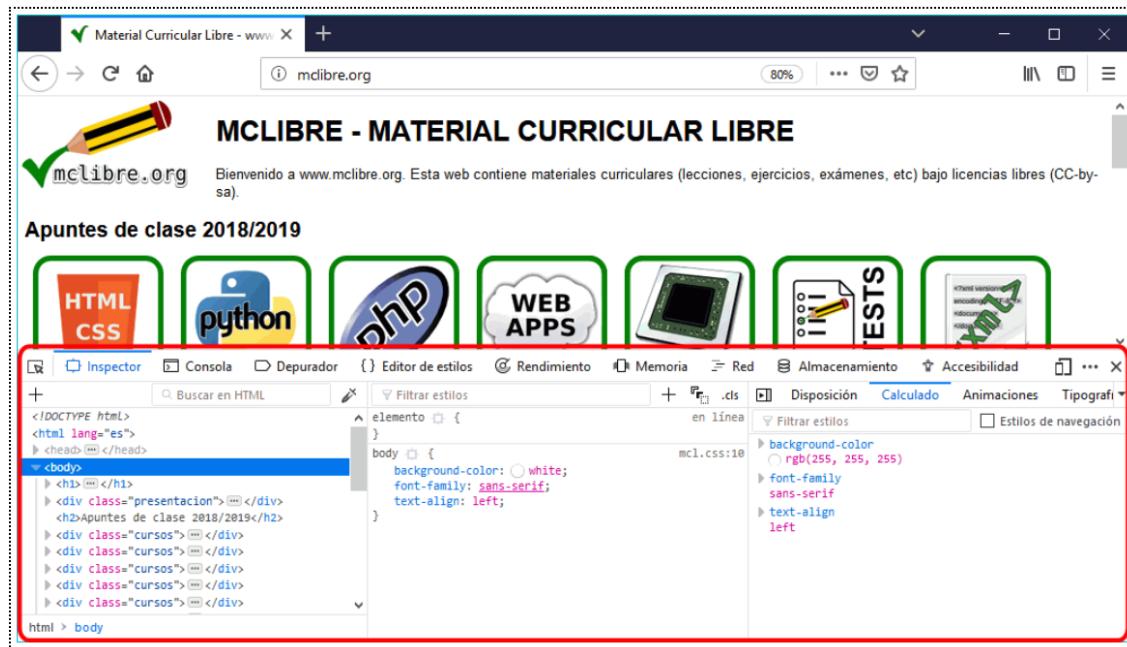
...



se abrirá un menú de opciones similar al menú **Herramientas > Desarrollador web** en el que debe elegir la opción "Alternar herramientas":



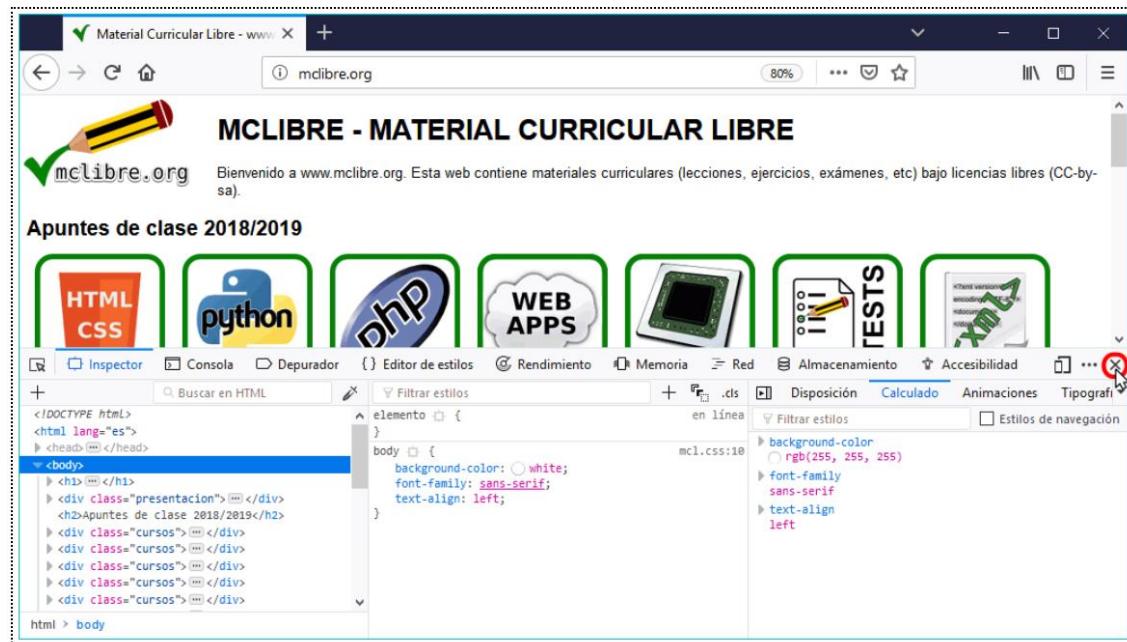
La ventana de Herramientas para desarrollador web se abre en la parte inferior de Firefox. Esta ventana personalizable da acceso directo a casi todas las herramientas disponibles (y se pueden incluir el resto en caso necesario).



### Cerrar la ventana de Herramientas para desarrollador web

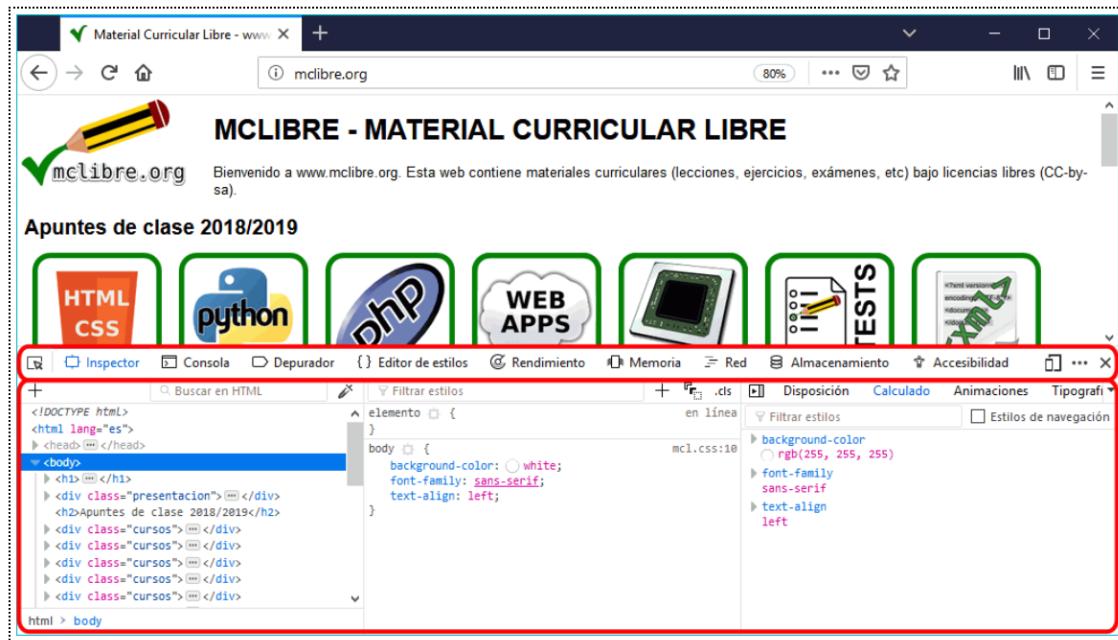
La ventana de herramientas para Desarrollador Web se puede cerrar de varias maneras:

- mediante el atajo de teclado Ctrl+May+I o la tecla F12
- haciendo clic en el icono Cerrar situado en la parte superior derecha de la ventana de Herramientas de desarrollador web.



### Elementos de la ventana de Herramientas para desarrollador web

La ventana de Herramientas para desarrollador web contiene su propio menú, que da acceso a cada una de las herramientas.

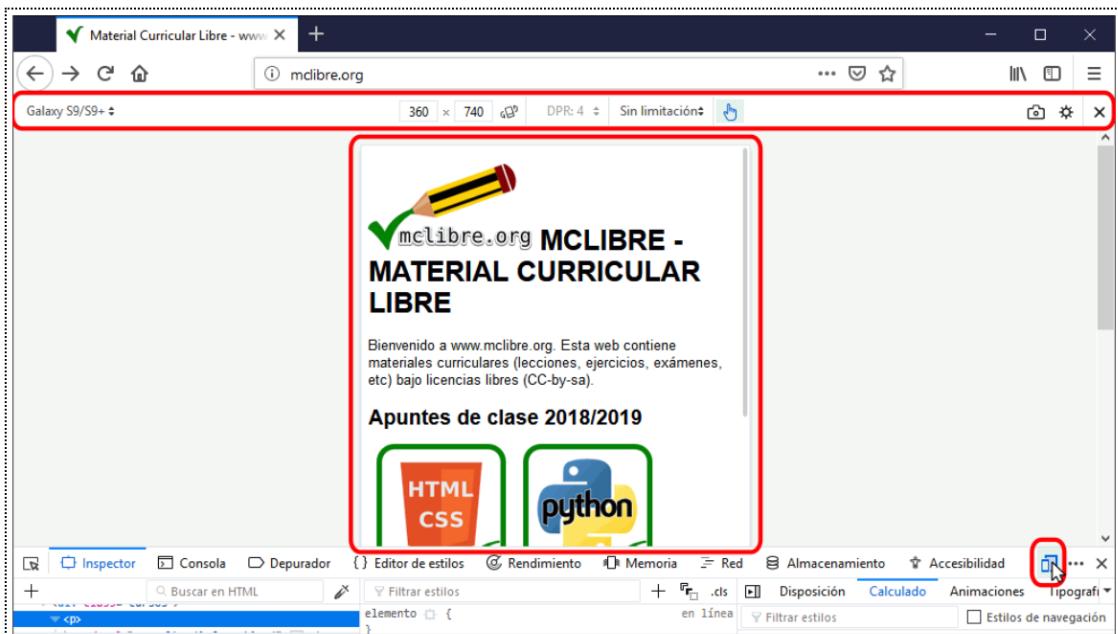


### Vista de diseño adaptable (responsive mode)

La vista de diseño adaptable (en inglés, *responsive mode*) permite probar en Firefox cómo se visualizaría la página web en diferentes dispositivos móviles.

Puede consultar información detallada en [MDN: Responsive Design Mode](#).

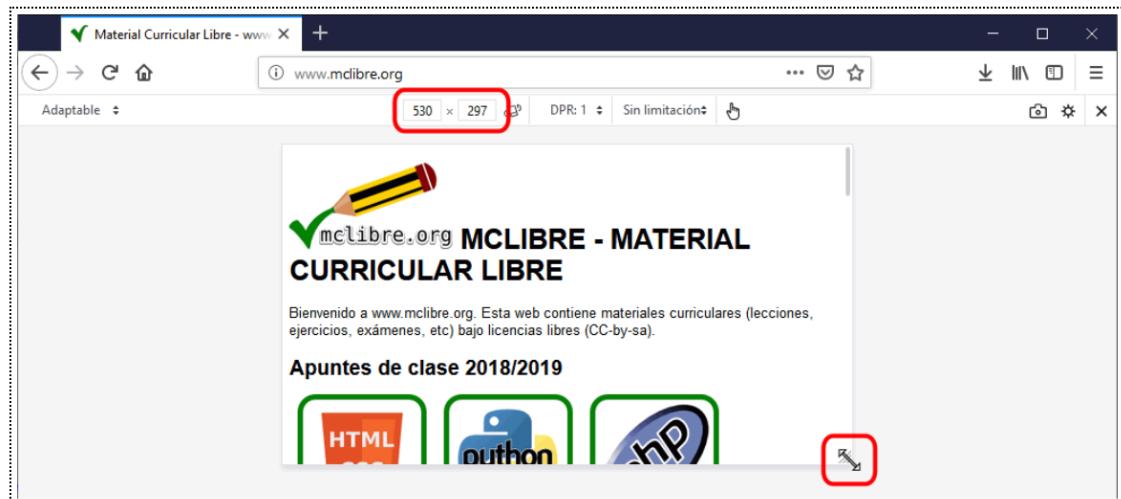
Para abrir la vista de diseño adaptable, haga clic en el ícono situado en la parte derecha del menú de las herramientas de desarrollador (el ícono representa un móvil y un tablet):



Para cerrar la vista de diseño adaptable, puede hacer clic en el ícono que abre la vista o hacer clic en el ícono de aspa del menú:



Puede modificar manualmente el tamaño del área de visualización del dispositivo emulado:

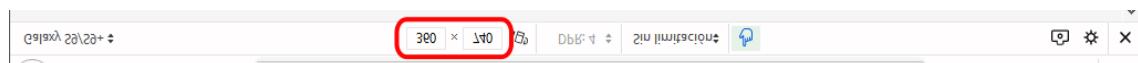


El menú de la vista de diseño adaptable permite elegir las características del dispositivo emulado:

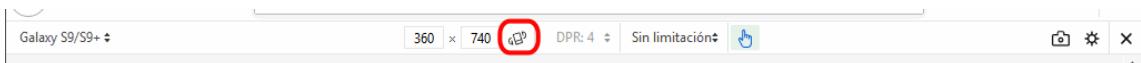
- Puede elegir un dispositivo específico de una lista (en la que se pueden añadir más dispositivos):



- Puede especificar un tamaño concreto:



- Puede cambiar la orientación del dispositivo (en vertical o apaisado):



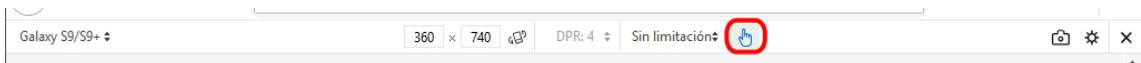
- Puede cambiar la densidad de píxeles del dispositivo:



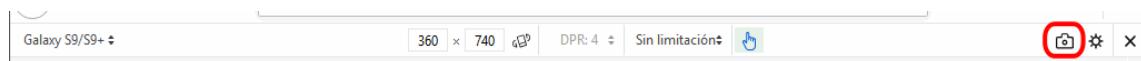
- Puede simular diferentes condiciones de red:



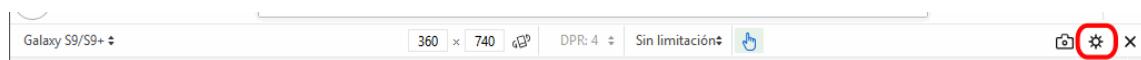
- Puede simular una pantalla táctil:



- Puede hacer capturas de la pantalla emulada (la captura se guarda automáticamente en la carpeta de descargas):



- Puede elegir algunas opciones adicionales de la vista de diseño adaptable:



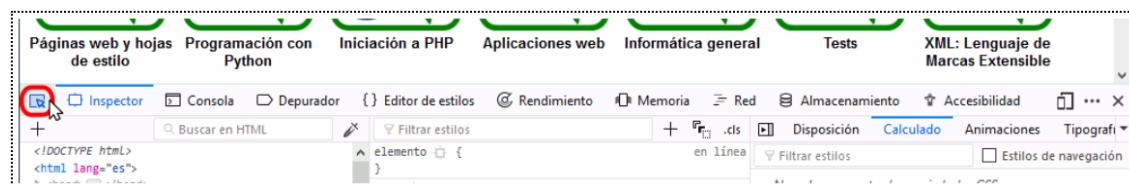
### Inspector de elementos

El inspector de elementos permite seleccionar visualmente un elemento de una página y ver el código html del elemento elegido y las propiedades CSS que se le aplican.

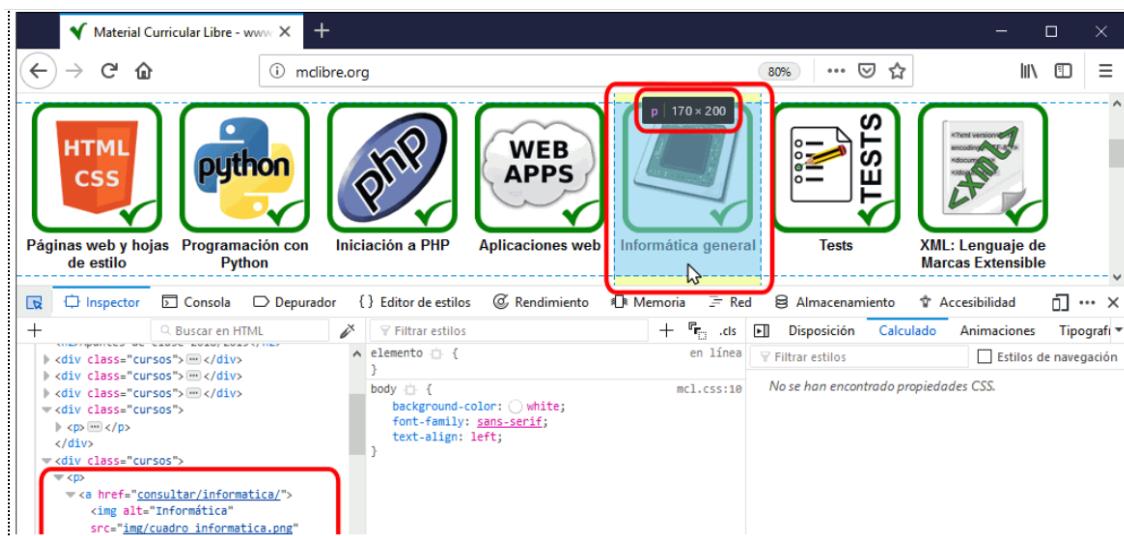
Para seleccionar elementos, haga clic en el ícono situado en el extremo izquierdo de la barra de menú:



El ícono cambia a color azul

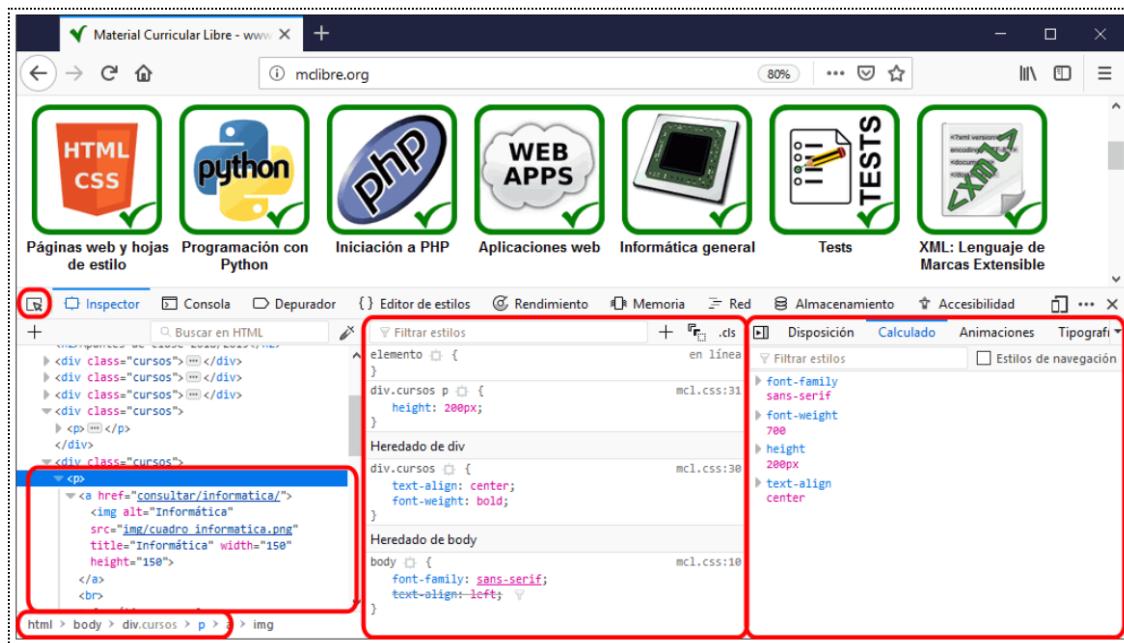


A partir de ese momento, al pasar el cursor sobre los elementos de la página, se van resaltando sus límites y mostrando la etiqueta correspondiente (en un bocadillo negro). Además, en la parte inferior se resalta con un fondo azul claro el código html correspondiente al elemento.



Al hacer clic en un elemento, el selector se desactiva y en la parte inferior se muestra:

- en la parte izquierda, el código html y la jerarquía de etiquetas
- en la parte central, las reglas de estilo que se aplican al elemento
- en la parte derecha, información sobre el elemento



En la parte del css, la información se puede ver de diferentes formas.

- En la parte central, se muestran los fragmentos de la hoja de estilo que define las propiedades que se aplican al elemento.

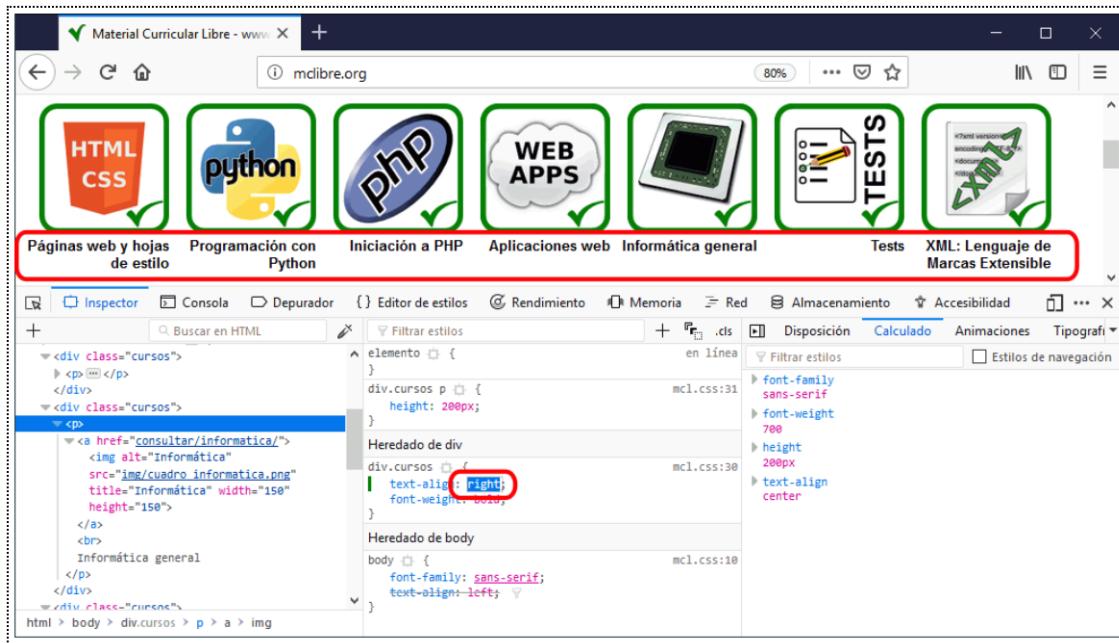
En el ejemplo siguiente, en el que se ha seleccionado un párrafo `<p>`, se muestran todas las propiedades que afecta al párrafo ordenadas por reglas. Para cada regla se muestra el selector, el nombre de la hoja de estilo y el número de línea en la que se encuentra y las propiedades.

Si una misma propiedad se establece en diferentes reglas, las propiedades que no se aplican por tener la regla menos prioridad se muestran tachadas.

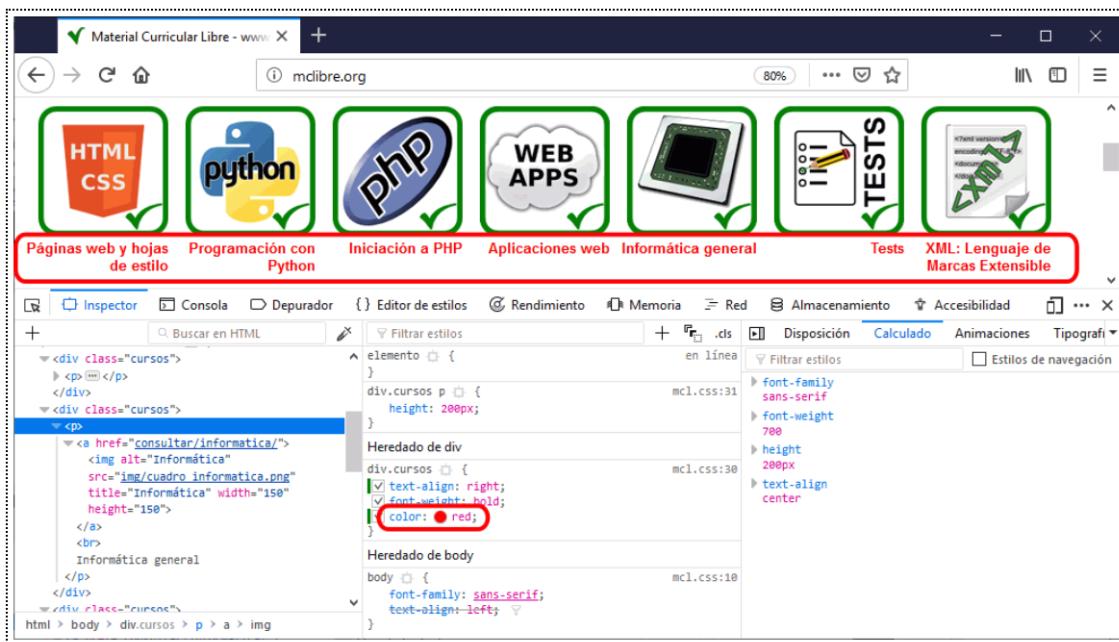
En la pestaña **Calculado** de la parte derecha, se muestra la misma información, pero ordenada por propiedades. Haciendo clic en los triángulos laterales, se detalla la hoja de estilo y regla que ha establecido la propiedad.

En la parte central, se pueden modificar las hojas de estilo y ver el resultado inmediatamente. Estas modificaciones son "virtuales" y al actualizar la página se vuelve a la vista original.

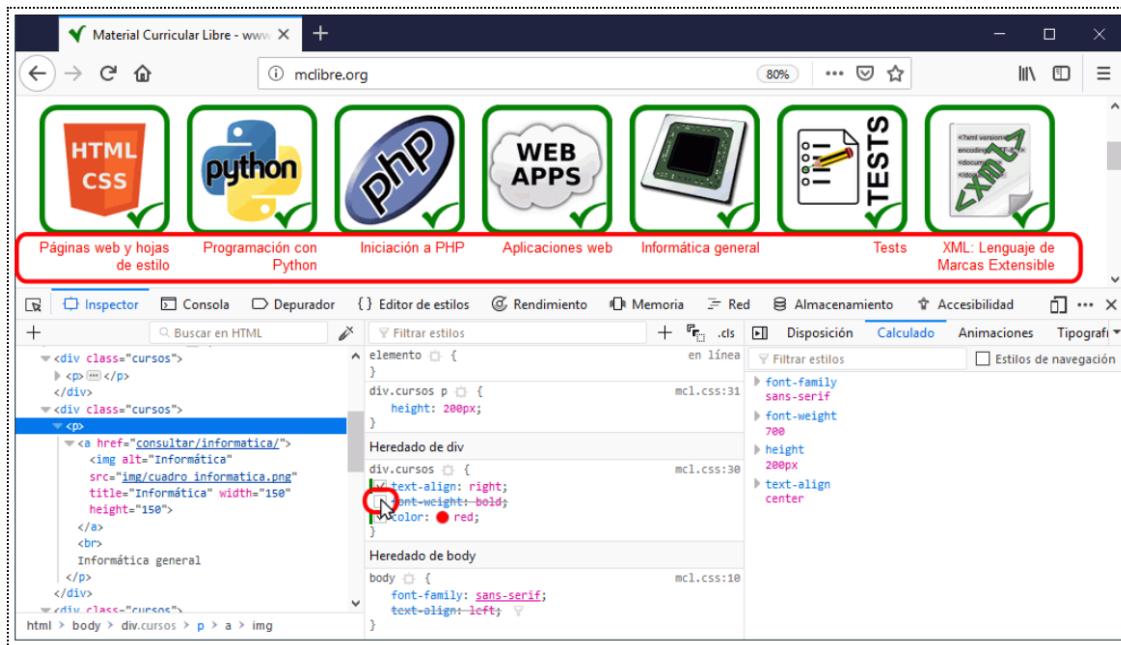
- Puede modificar el valor de una propiedad haciendo doble clic en el valor y escribiendo el nuevo valor:



- Puede añadir nuevas propiedades:



- Puede desactivar propiedades haciendo clic en las casillas situadas a la izquierda de la propiedad



### Recomendaciones

Las recomendaciones HTML 4.01 y CSS 2 ya incluían etiquetas y propiedades que permitían adaptar la página web a los tipos de medios, pero de manera muy básica.

La recomendación [CSS 3: Media queries](#), editada por Florian Rivoal, se aprobó el 19 de junio de 2012. Esta recomendación ampliaba las posibilidades de adaptación al permitir obtener detalles sobre las características de los dispositivos.

En octubre de 2019, el W3C se elaboraron dos recomendaciones relacionadas:

- la recomendación [Media Queries Level 4](#), que desarrolla la recomendación CSS 3: media queries.
- la recomendación [CSS Device Adaptation Modules Level 1](#), dedicada específicamente a las características de las pantallas.

### Densidad de píxeles: etiqueta meta viewport

El atributo **viewport** de la etiqueta **meta** no forma parte de ninguna recomendación ya aprobada, pero formará parte de la futura recomendación [CSS Device Adaptation Module Level 1](#) y es admitido por los navegadores.

La densidad de píxeles de una pantalla es el número de píxeles por unidad de distancia (aunque no sea lo mismo, también se utiliza con el mismo significado la expresión de puntos por unidad de distancia. Las unidades más utilizadas son [PPI](#) (píxeles por pulgada) y [DPI](#) (puntos por pulgada).

Los monitores de escritorio suelen tener más o menos 96 DPI, pero los dispositivos móviles incorporan pantallas de alta densidad, de 300 DPI o más.

Eso complica bastante el diseño, pues no sólo se tienen que tener en cuenta los anchos de las pantallas, sino también la densidad de píxeles.

Una forma de simplificar el diseño es incluir en la página web la etiqueta **meta viewport**:

```
<head>
...
<meta name="viewport" content="width=device-width, initial-scale=1.0">
...
</head>
```

Esta etiqueta hace que en dispositivos con pantallas de alta densidad el navegador aplique automáticamente un zoom que facilite la lectura.

Esta etiqueta fue creada por Apple en 2007, aunque rápidamente fue adoptada por todos los navegadores de los móviles. Por ello, se recomienda incluir esta etiqueta en todas las páginas web.

### La regla arroba @viewport

La futura recomendación [CSS Device Adaptation Module Level 1](#) define la regla arroba `@viewport`, que está inspirada por la etiqueta `meta viewport` pero está pensada tanto para navegadores de móviles como de escritorio.

El borrador declara la etiqueta `meta viewport` equivalente a un caso particular de la regla arroba `@viewport`. Si la regla arroba `@viewport` llega a implementarse en los navegadores, se podrá sustituir la etiqueta `meta viewport` de todas las páginas por la regla arroba `@viewport` en la hoja de estilo, pero parece que los navegadores se niegan a implementarla por diversas razones.

Referencias:

- MDN: [meta viewport](#)
- A tale of two viewports: [Part 1](#) - [Part 2](#)
- MDN: [Usando la etiqueta meta viewport en móviles](#)
- Safari Developer Library: [Configuring the viewport](#)
- La referencia más antigua a la etiqueta `meta viewport` que he encontrado ([One line of code](#)) es de julio de 2007.

### Hojas de estilo específicas: etiqueta link media

HTML 4 y CSS 2 ya permitían enlazar hojas de estilo específicas para los diferentes tipos de dispositivos. Al añadir el atributo `media` al enlace a la hoja de estilo, el navegador aplica esa hoja de estilo únicamente cuando el documento se muestra en ese dispositivo.

```
<head>
...
<link rel="stylesheet" media="screen" href="pantalla.css">
<link rel="stylesheet" media="print" href="impresora.css">
...
</head>
```

Los tipos de dispositivos definidos en HTML 4.01 eran: `screen` (pantallas), `tty` (pantallas de muy baja resolución y en blanco y negro), `tv` (pantallas de baja resolución y sin barras de desplazamiento, aunque con color), `projection` (proyectores), `handheld` (pantallas pequeñas y en blanco y negro), `print` (papel), `braille` (pantallas braille táctiles), `aural` (sintetizadores de voz) y `all` (todos). En CSS 2.1

se declaraba `aural` obsoleto y se añadían `embossed` (impresoras braille) y `speech` (sintetizadores de voz), además de definir grupos de medios.

Probar el siguiente ejemplo en una pestaña aparte y en la vista previa de impresión comprobar que se cambia el tipo de letra y el estilo del título.

```
<head>
...
<link rel="stylesheet" media="screen" href="pantalla.css">
<link rel="stylesheet" media="print" href="impresora.css">
...
</head>
```

```
/* pantalla.css */
body { font-family: sans-serif; }

h1 { text-align: center; text-transform: uppercase; }
```

```
/* impresora.css */
body { font-family: serif; }

h1 { text-decoration: underline; }
```

La idea de definir tipos de dispositivos no ha dado buen resultado, ya que las características de los dispositivos del mismo tipo a veces son muy diferentes y distintos tipos de dispositivos tienen características comunes. Por eso, en las recomendaciones posteriores se ha reducido el número de dispositivos y creado nuevas propiedades relacionadas con las características de los dispositivos (tamaños, resoluciones, número de colores, etc.).

La recomendación [CSS 3: Media queries](#), publicada en 2012, no definía los tipos de dispositivos, ya que estaban definidos en la recomendación HTML 4.01. Las recomendaciones HTML 5 y posteriores ya no incluían una lista de tipos de dispositivos, que se encuentra ahora en la recomendación [Media Queries Level 4](#). En ella, ya sólo se consideran los tipos `all` (pantallas), `print`, `screen` (pantallas) y `speech`.

### Características del dispositivo: media queries

La recomendación [CSS 3: Media queries](#) define toda una serie de **características de medios** (*media features*) que corresponden a características del dispositivo. Las más importantes son:

- **width**: anchura de la pantalla
- **height**: altura de la pantalla
- **orientation**: orientación de la pantalla `portrait` (vertical) o `landscape` (horizontal)
- **aspect-ratio**: proporción entre la anchura y la altura de la pantalla
- **resolution**: resolución (densidad de píxeles)

Estas características no permiten modificar las características físicas del dispositivo, lógicamente. Lo que hacen es tomar como valor las características del dispositivo, de una forma dinámica.

Algunas características tienen variantes con prefijos **min-** y **max-** para expresar comparaciones sin tener que utilizar los caracteres menor o mayor (< y >), que en HTML son caracteres especiales.

Esas propiedades se combinan con la regla arroba `@media` para condicionar la aplicación de propiedades a características del dispositivo.

Se pueden combinar varias características mediante el operador and y la coma (que representa el operador lógico or) para crear expresiones más complejas.

### Notas:

- En CSS 3: Media queries se definen también las características de medios **color, color-index, monochrome, scan** y **grid**, que se mantienen en Media Queries Level 4.
- En CSS 3: Media queries se definen también las características de medios **device-width, device-height** y **device-aspect-ratio**, que en Media Queries Level 4 se consideran obsoletas.
- En Media Queries Level 4 se definen también las características de medios **update, overflow-block, overflow-inline, color-gamut, pointer, hover, any-pointer** y **any-hover**.

### Ancho de pantalla: **min-width** y **max-width**

La característica de medios **min-width** se refiere al ancho mínimo de pantalla, es decir, que las reglas que contenga se aplicarán cuando la pantalla tenga una anchura mayor de la indicada.

La característica de medios **max-width** se refiere al ancho máximo de pantalla, es decir, que las reglas que contenga se aplicarán cuando la pantalla tenga una anchura menor o igual a la indicada.

El ejemplo siguiente muestra una hoja de estilo que modifica el color de fondo de la página dependiendo de si la pantalla tiene menos o más de 500px de ancho.

```
<body>
<p>El color de fondo de esta página cambia según el ancho de la pantalla:</p>
<ul>
    <li>si es menor de 500px, el color es azul</li>
    <li>si es mayor o igual a 500px, el color es rojo</li>
</ul>

<p>El ancho actual es <strong><span id="size"></span>px</strong></p>

<script>
    var ancho = window.innerWidth;
    var texto = document.getElementById("size");
    texto.textContent = ancho;
    window.onresize = function () {
        var ancho = window.innerWidth;
        var texto = document.getElementById("size");
        texto.textContent = ancho;
    }
</script>
</body>
```

```
@media screen and (max-width: 500px) {
    body {
        background-color: lightblue;
    }
}
@media screen and (min-width: 500px) {
    body {
        background-color: lightcoral;
    }
}
```

El color de fondo de esta página cambia según el ancho de la pantalla:  
• si es menor de 500px, el color es azul  
• si es mayor o igual a 500px, el color es rojo

El ancho actual es 787px

Los valores de ancho de pantalla en los que cambian las propiedades se denominan en inglés *breakpoints*. Cuantos más incluyamos, mejor podremos adaptar nuestras páginas web a diferentes dispositivos.

### Referencias:

- [Material Design recommended breakpoints](#)

### Orientación de pantalla: *orientation*

La característica de medios *orientation* toma dos posibles valores:

- **landscape**: orientación horizontal, es decir, que el ancho es mayor o igual que el alto
- **portrait**: orientación vertical, es decir, que el ancho es menor que el alto

El ejemplo siguiente muestra una hoja de estilo que modifica el color de fondo de la página dependiendo de si el ancho es mayor que el alto.

```
@media (orientation: portrait) {  
    html {  
        background-color: lightblue;  
    }  
}  
  
@media (orientation: landscape) {  
    html {  
        background-color: lightcoral;  
    }  
}
```

El color de fondo de esta página cambia según la orientación de la pantalla:

- Si el ancho es menor que el alto el color es azul
- Si el ancho es mayor o igual que el alto el color es rojo

El ancho actual es **681px**.

El alto actual es **288px**.

## 1.12. Guía de estilo

Una hoja de estilo se puede escribir de muchas maneras distintas y dar el mismo resultado.

Hay muchos aspectos que se pueden hacer de varias maneras (los espacios en blanco utilizados, los saltos de línea, el orden en que se escriben las propiedades, el orden en que se escriben las sentencias, etc.) y no se puede decir de forma objetiva que una manera esté bien y otra esté mal. Por ejemplo, los dos siguientes ejemplos son completamente equivalentes para los navegadores:

```
h1{color:white;background-color:black}  
h1  
{  
    color : white ;  
    background-color : black ;  
}
```

Lo importante al escribir una hoja de estilo es seguir siempre los mismos criterios, para que sea más fácil detectar errores o reutilizar las hojas de estilo en nuevos proyectos. Pero la diversidad de casos posibles hace que cada regla tenga excepciones.

Se deben seguir una serie de criterios que se detallan a continuación. En la mayoría de casos se trata de criterios bastante extendidos, por lo que se recomienda al alumno seguirlos al realizar los ejercicios.

### Juego de caracteres

Se recomienda que las hojas de estilo se guarden en el juego de caracteres UTF-8. De todas formas, se recomienda utilizar únicamente caracteres "ingleses" en las hojas de estilo. Es decir, se recomienda no utilizar acentos, ñ, ç, etc. en los nombres de las clases o ids. La única excepción admisible sería el uso de caracteres "no ingleses" en los comentarios

### Notas:

- Hasta julio de 2019 el borrador de la recomendación CSS Syntax Module Level 3 recomendaba que las hojas de estilo comenzaran con la regla-arroba

```
@charset "utf-8";
```

Pero desde julio de 2019 este borrador desaconseja su uso. Firefox no muestra correctamente los archivos CSS guardados en el juego de caracteres UTF-8,

### Tabuladores

Se recomienda no usar tabuladores. Cuando se quiera tabular una línea, se recomienda usar un par de espacios en blanco.

### Saltos de línea y espacios en blanco

Se recomienda escribir las sentencias de la siguiente manera:

- Escribir el selector y la llave de apertura en la misma línea, sin sangrar.
- Dejar un espacio en blanco entre el selector y la llave.
- Escribir cada propiedad en una línea, sangrando cada línea con dos espacios en blanco y completando la línea con un punto y coma.
- No escribir espacios antes de los dos puntos que separan la propiedad y su valor. Escribir un espacio tras los dos puntos.
- No escribir espacios en blanco antes del punto y coma final de cada línea de propiedad
- Escribir la llave de cierre en una línea, sin sangrar. En esa línea sólo estará la llave de cierre.
- Dejar una línea en blanco entre sentencia y sentencia.

```
h1 {  
    background-color: black;  
    color: white;  
}  
  
p {  
    text-indent: 3em;  
}
```

### Orden de las sentencias en la hoja de estilo

Se pueden utilizar dos criterios:

- I. Escribir las sentencias en el orden en que aparecen los elementos en la página web. Este criterio se suele utilizar cuando una hoja de estilo se va a aplicar a una sola página.

```
html {  
    /* ... */  
}  
  
body {  
    /* ... */  
}  
  
h1 {  
    /* ... */  
}
```

- II. Agrupar las sentencias por categorías. Por ejemplo, escribir primero las relativas a los bloques y después las relativas a elementos en línea. Dentro de cada categoría se pueden escribir las sentencias por orden alfabético o por el orden en que suelen aparecer en la página. Este criterio se suele utilizar cuando una hoja de estilo se va a aplicar a muchas páginas distintas, lo que suele ser habitual en sitios web.

```
h1 {  
  /* ... */  
}  
  
p {  
  /* ... */  
}  
  
address {  
  /* ... */  
}  
  
em {  
  /* ... */  
}  
  
strong {  
  /* ... */  
}
```

### *Orden de las propiedades en cada sentencia*

Las propiedades se escribirán en el siguiente orden:

- propiedades relacionadas con el posicionamiento (**position, float, clear, width, height, top, right, bottom, left, overflow**, etc.)
- propiedades relacionadas con el modelo de caja (**border, margin, padding**, etc.)
- propiedades relacionadas con el color (**background-color, color**)
- resto de propiedades (en orden alfabético)

Cuando hay varias propiedades que hacen referencia a los lados (por ejemplo, **margin-top, margin-right, margin-bottom, margin-left**) se puede escribir en el orden **left, right, top, bottom**. Así mismo, los valores de la propiedad **border** en el orden color, grosor, estilo.

### *Nombres de las clases*

Se intentará que los nombres de las clases hagan referencia a categorías más que a aspectos visuales. Por ejemplo, conviene evitar llamar a las clases con nombres de colores (azul, blanco, etc.) porque en una página web real los colores es uno de los elementos que más se suele cambiar con el paso del tiempo.

Lógicamente, esta recomendación es importante en las webs reales ya que en ellas se modifica a menudo el aspecto visual.

Por otro lado, esta recomendación es una de las más ambiguas y por tanto difíciles de seguir. Por ejemplo, aunque se desaconseje utilizar nombres de colores, sería más admisible definir clases "izquierda" y "derecha" para imágenes flotantes, que también hacen referencia a un aspecto visual. El motivo es que es menos probable que decidamos cambiar a la derecha todas las imágenes flotantes que salían a la izquierda y viceversa, que decidamos cambiar el color de un elemento o cambiar la gama de colores de un sitio web.

En los casos en los que no sea fácil nombrar las categorías, se recomienda utilizar nombres genéricos, por ejemplo "color-1", "color-2", "color-3", sería preferible a "azul", "rojo", etc.

Cuando el nombre de la clase contiene varias palabras o números, separar las palabras con guiones (-): por ejemplo "estilo-1", "fecha-edicion", "autor-libro", etc.

### *Orden de las clases en la hoja de estilo*

Si se definen clases, se agruparán todas las sentencias en la primera aparición de la etiqueta (aunque en la página web esas clases aparezcan más tarde).

```
h1 {  
  /* ... */  
}  
  
p {  
  /* ... */  
}  
  
p.importante {  
  /* ... */  
}  
  
p.errata {  
  /* ... */  
}
```

### Clases o id

Se utilizarán preferentemente clases en vez de id, aunque las clases se vayan a asignar a un único elemento.

Únicamente se aconseja el uso de id cuando asignar la misma clase a varios elementos provoque problemas en la página (por ejemplo, en el caso de posicionamiento absoluto de elementos).

### Número de clases

Cuando un elemento se muestra en una página de N formas distintas, se puede hacer como mínimo de dos maneras:

- definir N clases distintas (una para cada de las formas distintas en que aparece el elemento)
- definir N-1 clases distintas (una para cada una -menos una- de las formas distintas en que aparece el elemento) y definir la etiqueta sin clase como la forma distinta restante.

Se aconseja utilizar el primer método, es decir, definir tantas clases como tipos de elementos, y reservar la sentencia sin clase para los aspectos comunes a todas las clases.

Por ejemplo, supongamos que en una página hay tres tipos distintos de párrafos <p> (con tipos de letra o colores distintos, etc.). Podemos hacerlo de dos maneras (se aconseja utilizar la primera):

- definir 3 clases distintas de párrafo <p> (una para cada de los tres tipos que hay en la página)

```
p.clase-1 {  
  font-family: sans-serif;  
}  
  
p.clase-2 {  
  font-family: serif;  
}  
  
p.clase-3 {  
  font-family: monospace;  
}
```

- definir sólo 2 clases distintas de párrafo <p> (para dos de los tipos que hay en la página) y definir la etiqueta sin clase para el tercer tipo

```
p {  
  font-family: sans-serif;  
}  
  
p.clase-1 {  
  font-family: serif;  
}  
  
p.clase-2 {  
  font-family: monospace;  
}
```

## 1.13. Validación de páginas web y hojas de estilo

### Validación de páginas web

Una de las principales diferencias entre los ordenadores y los seres humanos es que los humanos estamos preparados para afrontar situaciones con información incompleta o errónea. Los ordenadores, sin embargo, necesitan exactitud y precisión para poder seguir adelante. Así, en el caso de los lenguajes los humanos podemos entendernos, aunque no hablemos con corrección y sabemos reconocer la ironía, el cinismo, los juegos de palabras, las paradojas o las contradicciones. Sin embargo, los lenguajes de programación de ordenadores exigen siempre que los programas se ajusten exactamente a las reglas del lenguaje, sin ambigüedades.

El HTML no es un lenguaje de programación, pero tiene su propio vocabulario (las marcas, atributos y valores de atributos que podemos utilizar) y su propia sintaxis (las reglas de utilización de sus elementos). Las páginas web son documentos creados por humanos e interpretados por los ordenadores (por los navegadores), por lo que podemos entender el HTML como un lenguaje con el que los humanos nos comunicamos con los ordenadores, aunque sólo sea para decirles cómo tienen que mostrar el contenido de las páginas web a otros humanos.

Quizás uno de los motivos del éxito de la web (es decir, del HTML) fue que los navegadores siempre han sido capaces de procesar documentos con errores y de conseguir en muchos casos mostrarlos como deseaba su autor. Por supuesto, esa capacidad se debe al trabajo y esfuerzo de los programadores que crean los navegadores, que han incluido en los navegadores numerosas reglas para reconocer los errores más comunes e intentar corregirlos sobre la marcha.

En el año 2000, el W3C intentó con el XHTML obligar a que las páginas web dejaran de tener errores de sintaxis. Ese intento fue un fracaso, y en el HTML 5 se ha tomado el camino contrario, detallando cómo actuar ante violaciones de la propia sintaxis y consiguiendo un comportamiento similar en todos los navegadores. Por cierto, aunque en el HTML 5 se mantiene una variante con sintaxis estricta, el XHTML 5, su uso real es residual.

De todas formas, en un curso de creación de páginas web es lógico insistir en la necesidad de escribir páginas sin errores de sintaxis. Es verdad que podemos cometer errores y es bueno que los navegadores aun así consigan mostrar las páginas correctamente, pero si nos acostumbramos a escribir páginas plagadas de errores, en el momento en que los navegadores ya no puedan entender nuestras páginas será más difícil encontrar el origen del error porque habrá demasiados. Merece la pena acostumbrarse desde el principio a no cometer errores y crear páginas más fáciles de entender, modificar y reutilizar en el futuro.

Para ayudarnos a detectar errores debemos utilizar validadores. En esta sección se comenta la instalación y uso de validadores, tanto en el editor como en el navegador.

### Validadores disponibles en Internet

#### *W3C Markup Validation Service*

El W3C ofrece desde hace muchos años varios validadores que permiten validar todo tipo de documentos (HTML, XHTML, CSS, SVG, MathML, XML, etc.). Los distintos validadores son:

- [Validación de lenguajes de marcas](#) (HTML, XHTML, SVG, MathML, SMIL)
- [Validación de hojas de estilo CSS](#)
- [Comprobador de enlaces](#)

- [Validación total](#) (HTML, CSS, internacionalización, RSS)

La validación se puede hacer de tres maneras:

- dando la dirección web del recurso (si ya está disponible en Internet),
- subiendo el archivo mediante un formulario de subida de archivos
- copiando el código fuente en un formulario.

Para la validación de HTML 5, el validador del W3C utiliza el validador de [validator.nu](#). Este validador fue iniciado en 2005 por [Henri Sivonen](#), que desde 2007 trabaja en la fundación Mozilla.

### Validación en el navegador

En Firefox o Chrome se pueden validar las páginas web y hojas de estilo accediendo a las páginas web de los validadores disponibles en Internet.

Como esta validación es un poco farragosa, en Firefox podremos recurrir a extensiones como Web Developer, cuya instalación y uso se comenta en la lección [Complementos de Firefox](#).

### 1.14. Listado de propiedades CSS estables

La tabla siguiente resume las propiedades definidas en las recomendaciones de Hojas de Estilo en Cascada (CSS) y admitidas correctamente por los navegadores más utilizados:

- Se incluyen las propiedades CSS 2, excepto las propiedades relacionadas con las hojas de estilo auditivas (aural).
- Las propiedades que han sido eliminadas en la recomendación [CSS 2.1](#) se listan al final.
- Los valores de las propiedades añadidos o eliminados en la recomendación [CSS 2.1](#) se indican con los símbolos <sup>(+)</sup> o <sup>(-)</sup>.
- Las propiedades o los valores de las propiedades definidas en recomendaciones CSS 3 en elaboración que los navegadores actuales admiten correctamente se indican con el símbolo <sup>(3)</sup>.

Tipos	Propiedad	Descripción	Valores
	contador	Contador	counter(nombre, estilo)
familia-genérica		Familia genérica de fuente	<i>cursive</i>   <i>fantasy</i>   <i>monospace</i>   <i>serif</i>   <i>sans-serif</i>
forma		Forma	rect(top, right, bottom, left)
tamaño-absoluto		Tamaño absoluto de fuente	<i>xx-small</i>   <i>x-small</i>   <i>medium</i>   <i>large</i>   <i>x-large</i>   <i>xx-large</i>
tamaño-relativo		Tamaño relativo de fuente	<i>larger</i>   <i>smaller</i>
uri (para <a href="#">Imagen de fondo</a> o <a href="#">fuentes web</a> )		Dirección absoluta o relativa	url("ruta_y_nombre_de_archivo")

Reglas arroba	Propiedad	Descripción	Descriptores / Características
@import		Importar recurso	url(...)
@font-face		Fuente web	font-family, src, font-style, font-weight, font-stretch, unicode-range, font-variant, font-feature-settings
@media		Características del dispositivo	width   height   orientation   aspect-ratio   resolution   etc.

## UF1303 Elaboración hojas de estilo

### Tipo de letra (fuente)

Propiedad	Descripción	Valores
<code>font</code>	propiedad compuesta	[ [ font-style    font-variant    font-weight ]? font-size [ / line-height ]? font-family ]   caption   icon   menu   message-box   small-caption   status-bar
<code>font-family</code>	tipo de letra (fuente)	[ nombre-fuente   familia-genérica ] [, nombre-fuente   familia-genérica ]*
<code>font-size</code>	tamaño	tamaño-absoluto   tamaño-relativo   distancia   porcentaje
<code>font-weight</code>	grosor del trazo (negrita)	normal   bold   bolder   lighter   100   200   300   400   500   600   700   800   900
<code>font-style</code>	inclinación (cursiva)	normal   italic   oblique
<code>font-variant<sup>(3)</sup></code>	variantes tipográficas	normal   small-caps
<code>font-variant-caps<sup>(3)</sup></code>	versalitas	normal   small-caps
<code>font-kerning<sup>(3)</sup></code>	kerning	auto   normal   none
<code>font-variant-ligatures<sup>(3)</sup></code>	ligaduras	normal   none   common-ligatures   no-common-ligatures   discretionary-ligatures   no-discretionary-ligatures   historical-ligatures   no-historical-ligatures   contextual   no-contextual
<code>font-variant-numeric<sup>(3)</sup></code>	versalitas	normal   lining-nums   oldstyle-nums   proportional-nums   tabular-nums   diagonal-fractions   stacked-fractions   ordinal   slashed-zeros
<code>font-variant-position<sup>(3)</sup></code>	posición	normal   sub   super
<code>font-variant-east-asian<sup>(3)</sup></code>	lenguas asiáticas	normal   jis78   jis83   jis90   jis04   simplified   traditional   full-width   proportional-width   ruby
<code>font-feature-settings<sup>(3)</sup></code>	características OpenType	normal   características OpenType
<code>font-synthesis<sup>(3)</sup></code>	síntesis	none   weight   style
<code>font-size-adjust<sup>(3)</sup></code>	ajuste del tamaño	número
<code>font-stretch<sup>(3)</sup></code>	anchura	ultra-condensed, extra-condensed, condensed, semi-condensed, normal, semi-expanded, expanded, extra-expanded, ultra-expanded

### Texto

Propiedad	Descripción	Valores
<code>color</code>	color del texto	color
<code>direction</code>	dirección del texto	ltr   rtl
<code>letter-spacing</code>	espacio entre caracteres	normal   distancia
<code>line-height</code>	espaciado entre líneas	normal   número   distancia   porcentaje
<code>text-align</code>	alineación del texto	center   justify   left   right
<code>text-decoration</code>	decoración del texto	none   blink   line-through   overline   underline
<code>text-indent</code>	sangrado de la primera línea	distancia   porcentaje
<code>text-transform</code>	mayúsculas / minúsculas	none   capitalize   lowercase   uppercase
<code>text-shadow<sup>(3)</sup></code>	sombreado	desplazamiento horizontal, desplazamiento vertical, tamaño desenfoque, color
<code>unicode-bidi</code>	dirección del texto	normal   embed   bidi-override
<code>vertical-align</code>	alineación vertical	baseline   bottom   middle   sub   super   text-bottom   text-top   top   distancia   porcentaje
<code>white-space</code>	espacios en blanco, saltos de línea y wrap	normal   nowrap   pre   pre-line <sup>(+)</sup>   pre-wrap <sup>(+)</sup>
<code>word-spacing</code>	espacio entre palabras	normal   distancia

### Bordes

Propiedad	Descripción	Valores
<code>border</code>	cuatro bordes simultáneamente	border-color    border-width    border-style
<code>border-top</code>	borde superior	border-color    border-width    border-style
<code>border-right</code>	borde derecho	border-color    border-width    border-style
<code>border-bottom</code>	borde inferior	border-color    border-width    border-style
<code>border-left</code>	borde izquierdo	border-color    border-width    border-style
<code>border-radius<sup>(3)</sup></code>	esquinas redondeadas	[ distancia   porcentaje ] {1, 4}
<code>border-top-right-radius<sup>(3)</sup></code>	esquina superior derecha redondeada	distancia   porcentaje
<code>border-bottom-right-radius<sup>(3)</sup></code>	esquina inferior derecha redondeada	distancia   porcentaje
<code>border-bottom-left-radius<sup>(3)</sup></code>	esquina inferior izquierda redondeada	distancia   porcentaje
<code>border-top-left-radius<sup>(3)</sup></code>	esquina superior izquierda redondeada	distancia   porcentaje
<code>border-color</code>	color de los bordes	[ color   transparent ] {1, 4}
<code>border-width</code>	grosor de los bordes	[ medium   thick   thin   distancia ] {1, 4}
<code>border-style</code>	estilos de los bordes	[ none   hidden   dashed   dotted   double   groove   inset   outset   ridge   solid ] {1, 4}
<code>border-top-color</code>	color del borde superior	[ color   transparent ] {1, 4}
<code>border-top-width</code>	grosor del borde superior	[ medium   thick   thin   distancia ] {1, 4}
<code>border-top-style</code>	estilo del borde superior	[ none   hidden   dashed   dotted   double   groove   inset   outset   ridge   solid ] {1, 4}
<code>border-right-color</code>	color del borde derecho	[ color   transparent ] {1, 4}
<code>border-right-width</code>	grosor del borde derecho	[ medium   thick   thin   distancia ] {1, 4}
<code>border-right-style</code>	estilo del borde derecho	[ none   hidden   dashed   dotted   double   groove   inset   outset   ridge   solid ] {1, 4}
<code>border-bottom-color</code>	color del borde inferior	[ color   transparent ] {1, 4}
<code>border-bottom-width</code>	grosor del borde inferior	[ medium   thick   thin   distancia ] {1, 4}
<code>border-bottom-style</code>	estilo del borde inferior	[ none   hidden   dashed   dotted   double   groove   inset   outset   ridge   solid ] {1, 4}
<code>border-left-color</code>	color del borde izquierdo	[ color   transparent ] {1, 4}
<code>border-left-width</code>	grosor del borde izquierdo	[ medium   thick   thin   distancia ] {1, 4}
<code>border-left-style</code>	estilo del borde izquierdo	[ none   hidden   dashed   dotted   double   groove   inset   outset   ridge   solid ] {1, 4}

## UF1303 Elaboración hojas de estilo

### Margen exterior

Propiedad	Descripción	Valores
<code>margin</code>	cuatro márgenes exterior simultáneamente	[ <code>auto</code>   <code>distancia</code>   <code>porcentaje</code> ] {1, 4}
<code>margin-top</code>	márgen exterior superior	<code>auto</code>   <code>distancia</code>   <code>porcentaje</code>
<code>margin-right</code>	márgen exterior derecho	<code>auto</code>   <code>distancia</code>   <code>porcentaje</code>
<code>margin-bottom</code>	márgen exterior inferior	<code>auto</code>   <code>distancia</code>   <code>porcentaje</code>
<code>margin-left</code>	márgen exterior izquierdo	<code>auto</code>   <code>distancia</code>   <code>porcentaje</code>

### Margen interior

Propiedad	Descripción	Valores
<code>padding</code>	cuatro márgenes interiores simultáneamente	[ <code>distancia</code>   <code>porcentaje</code> ] {1, 4}
<code>padding-top</code>	márgen interior superior	<code>distancia</code>   <code>porcentaje</code>
<code>padding-right</code>	márgen interior derecho	<code>distancia</code>   <code>porcentaje</code>
<code>padding-bottom</code>	márgen interior inferior	<code>distancia</code>   <code>porcentaje</code>
<code>padding-left</code>	márgen interior izquierdo	<code>distancia</code>   <code>porcentaje</code>

### Fondos

Propiedad	Descripción	Valores
<code>background</code>	propiedad compuesta	<code>background-attachment</code>    <code>background-color</code>    <code>background-image</code>    <code>background-position</code>    <code>background-repeat</code>
<code>background<sup>(3)</sup></code>	fondos múltiples	separados por comas
<code>background-attachment</code>	ligadura de la imagen	<code>fixed</code>   <code>scroll</code>
<code>background-color</code>	color de fondo	<code>transparent</code>   color
<code>background-image</code>	imagen de fondo	<code>none</code>   <code>uri</code>
<code>background-position</code>	posición de la imagen de fondo	[ [ <code>left</code>   <code>center</code>   <code>right</code>   <code>distancia</code>   <code>porcentaje</code> ] [ <code>top</code>   <code>center</code>   <code>bottom</code>   <code>distancia</code>   <code>porcentaje</code> ] ]?   [ [ <code>left</code>   <code>center</code>   <code>right</code> ]    [ <code>top</code>   <code>center</code>   <code>bottom</code> ] ]
<code>background-repeat</code>	repetición de la imagen de fondo	<code>no-repeat</code>   <code>repeat</code>   <code>repeat-x</code>   <code>repeat-y</code>   <code>space<sup>(3)</sup></code>   <code>round<sup>(3)</sup></code>
<code>background-clip</code> <sup>(3)</sup>	límite de la imagen de fondo	<code>border-box</code>   <code>padding-box</code>   <code>content-box</code>
<code>background-size<sup>(3)</sup></code>	tamaño de la imagen de fondo	<code>auto</code>   [ <code>distancia</code>   <code>porcentaje</code> ] {2}   <code>contain</code>   <code>cover</code>
<code>background-origin<sup>(3)</sup></code>	origen de la imagen de fondo	<code>border-box</code>   <code>padding-box</code>   <code>content-box</code>

### Listas

Propiedad	Descripción	Valores
<code>list-style</code>	propiedad compuesta	<code>list-style-image</code>    <code>list-style-position</code>    <code>list-style-type</code>
<code>list-style-image</code>	imagen del marcador	<code>none</code>   <code>uri</code>
<code>list-style-position</code>	posición del marcador	<code>inside</code>   <code>outside</code>
<code>list-style-type</code>	tipo de marcador	<code>none</code>   <code>circle</code>   <code>disc</code>   <code>square</code>   <code>decimal</code>   <code>decimal-leading-zero</code>   <code>lower-alpha</code>   <code>upper-alpha</code>   <code>lower-greek</code>   <code>lower-latin</code>   <code>upper-latin</code>   <code>lower-roman</code>   <code>upper-roman</code>   <code>armenian</code>   <code>georgian</code>   <code>hebrew<sup>(1)</sup></code>   <code>cjk-ideographic<sup>(2)</sup></code>   <code>hiragana<sup>(3)</sup></code>   <code>katakana<sup>(3)</sup></code>   <code>hiragana-iroha<sup>(3)</sup></code>   <code>katakana-iroha<sup>(3)</sup></code>

### Tablas

Propiedad	Descripción	Valores
<code>border-collapse</code>	modo de bordes	<code>collapse</code>   <code>separate</code>
<code>border-spacing</code>	separación entre celdas	<code>distancia</code> <code>distancia?</code>
<code>caption-side</code>	posición de la leyenda	<code>top</code>   <code>bottom</code>   <code>left<sup>(1)</sup></code>   <code>right<sup>(1)</sup></code>
<code>empty-cells</code>	borde de casillas vacías	<code>hide</code>   <code>show</code>
<code>table-layout</code>	algoritmo ancho de tabla	<code>auto</code>   <code>fixed</code>

### Modelo de caja / Tamaño

Propiedad	Descripción	Valores
<code>display</code>	tipo de caja	<code>none</code>   <code>block</code>   <code>compact<sup>(1)</sup></code>   <code>inline</code>   <code>inline-block<sup>(2)</sup></code>   <code>inline-table</code>   <code>list-item</code>   <code>marker<sup>(1)</sup></code>   <code>run-in</code>   <code>table</code>   <code>table-caption</code>   <code>table-cell</code>   <code>table-column</code>   <code>table-column-group</code>   <code>table-footer-group</code>   <code>table-header-group</code>   <code>table-row</code>   <code>table-row-group</code>   <code>flex<sup>(3)</sup></code>   <code>inline-flex<sup>(3)</sup></code>
<code>box-sizing<sup>(3)</sup></code>	posición del borde y márgenes	<code>border-box</code>   <code>content-box</code>
<code>width</code>	anchura	<code>auto</code>   <code>distancia</code>   <code>porcentaje</code>
<code>min-width</code>	anchura mínima	<code>distancia</code>   <code>porcentaje</code>
<code>max-width</code>	anchura máxima	<code>distancia</code>   <code>porcentaje</code>
<code>height</code>	altura	<code>auto</code>   <code>distancia</code>   <code>porcentaje</code>
<code>min-height</code>	altura mínima	<code>distancia</code>   <code>porcentaje</code>
<code>max-height</code>	anchura máxima	<code>distancia</code>   <code>porcentaje</code>
<code>overflow</code>	si el contenido desborda al elemento	<code>auto</code>   <code>hidden</code>   <code>scroll</code>   <code>visible</code>

### Posicionamiento

Propiedad	Descripción	Valores
<code>float</code>	modo de posicionamiento flotante	<code>none</code>   <code>left</code>   <code>right</code>
<code>clear</code>	lado en el que no puede haber elementos flotantes	<code>none</code>   <code>both</code>   <code>left</code>   <code>right</code>
<code>position</code>	modo de posicionamiento	<code>absolute</code>   <code>fixed</code>   <code>relative</code>   <code>static</code>
<code>top</code>	posición del borde superior del elemento	<code>auto</code>   <code>distancia</code>   <code>porcentaje</code>
<code>right</code>	posición del borde derecho del elemento	<code>auto</code>   <code>distancia</code>   <code>porcentaje</code>
<code>bottom</code>	posición del borde inferior del elemento	<code>auto</code>   <code>distancia</code>   <code>porcentaje</code>
<code>left</code>	posición del borde izquierdo del elemento	<code>auto</code>   <code>distancia</code>   <code>porcentaje</code>
<code>clip</code>	recorta el elemento	<code>auto</code>   <code>forma</code>
<code>visibility</code>	visibilidad	<code>hidden</code>   <code>collapse</code>   <code>visible</code>
<code>z-index</code>	apilamiento	<code>auto</code>   <code>número-entero</code>

### Cajas flexibles

Propiedad	Descripción	Valores
<code>display</code>	tipo de caja	<code>flex<sup>(3)</sup></code>    <code>inline-flex<sup>(3)</sup></code>
<code>flex-flow<sup>(3)</sup></code>	propiedad compuesta	<code>flex-direction</code>    <code>flex-wrap</code>
<code>flex-direction<sup>(3)</sup></code>	dirección	<code>row</code>    <code>row-reverse</code>    <code>column</code>    <code>column-reverse</code>
<code>flex-wrap<sup>(3)</sup></code>	ajuste de línea	<code>wrap</code>    <code>no-wrap</code>    <code>wrap-reverse</code>
<code>order<sup>(3)</sup></code>	número de orden	número
<code>flex<sup>(3)</sup></code>	propiedad compuesta	[ <code>flex-grow</code>    <code>flex-shrink</code>    <code>flex-basis</code> ]    <code>initial</code>    <code>auto</code>    <code>none</code>    número
<code>flex-grow<sup>(3)</sup></code>	factor de expansión	número
<code>flex-shrink<sup>(3)</sup></code>	factor de compresión	número
<code>flex-basis<sup>(3)</sup></code>	tamaño inicial	distancia    auto    content    width
<code>justify-content<sup>(3)</sup></code>	alineación en la dirección principal	<code>flex-start</code>    <code>flex-end</code>    <code>center</code>    <code>space-between</code>    <code>space-around</code>
<code>align-items<sup>(3)</sup></code>	alineación en la dirección secundaria (una línea)	<code>flex-start</code>    <code>flex-end</code>    <code>center</code>    <code>baseline</code>    <code>stretch</code>
<code>align-self<sup>(3)</sup></code>	alineación individual en la dirección secundaria	<code>auto</code>    <code>flex-start</code>    <code>flex-end</code>    <code>center</code>    <code>baseline</code>    <code>stretch</code>
<code>align-content<sup>(3)</sup></code>	alineación en la dirección secundaria (varias líneas)	<code>flex-start</code>    <code>flex-end</code>    <code>center</code>    <code>space-between</code>    <code>space-around</code>    <code>stretch</code>

### Impresora

Propiedad	Descripción	Valores
<code>page-break-after</code>	salto de página después de	<code>auto</code>   <code>always</code>   <code>avoid</code>   <code>left</code>   <code>right</code>
<code>page-break-before</code>	salto de página antes de	<code>auto</code>   <code>always</code>   <code>avoid</code>   <code>left</code>   <code>right</code>
<code>page-break-inside</code>	salto de página dentro de	<code>auto</code>   <code>avoid</code>
<code>orphans</code>	número de líneas al final de página	número-entero
<code>widows</code>	número de líneas al principio de la página	número-entero

### Interface de usuario

Propiedad	Descripción	Valores
<code>cursor</code>	tipo de cursor	[ <code>uri,</code> *]    <code>auto</code>   <code>crosshair</code>   <code>default</code>   <code>help</code>   <code>move</code>   <code>pointer</code>   <code>progress</code>   <code>n-resize</code>   <code>ne-resize</code>   <code>e-resize</code>   <code>se-resize</code>   <code>s-resize</code>   <code>sw-resize</code>   <code>w-resize</code>   <code>text</code>   <code>wait</code>
<code>outline</code>	cuatro bordes simultáneamente	<code>outline-color</code>    <code>outline-width</code>    <code>outline-style</code>
<code>outline-color</code>	color de los bordes	color
<code>outline-width</code>	grosor de los bordes	<code>border-width</code>
<code>outline-style</code>	estilos de los bordes	<code>border-style</code>

### Pseudo-elementos

Nombre	Descripción	Valores
<code>::after</code>	inserta contenido después del elemento	<code>content: contenido</code>
<code>::before</code>	inserta contenido antes del elemento	<code>content: contenido</code>
<code>::first-letter</code>	primera letra	
<code>::first-line</code>	primera línea de texto	
<code>content</code>	contenido generado	<code>normal<sup>(*)</sup></code>   <code>none<sup>(+)</sup></code>   [ <code>texto</code>   <code>uri</code>   <code>contador</code>   <code>attr(atributo)</code>   <code>open-quote</code>   <code>close-quote</code>   <code>no-open-quote</code>   <code>no-close-quote</code> ]+
<code>counter-increment</code>	incremento de contador	<code>none</code>   [ <code>identificador</code> <code>número-entero?</code> ]+
<code>counter-reset</code>	puesto a cero de contador	<code>none</code>   [ <code>identificador</code> <code>número-entero?</code> ]+
<code>quotes</code>	comillas	<code>none</code>   [ <code>texto-apertura</code> <code>texto-cierre</code> ]+

### Pseudo-clases

Nombre	Descripción	Valores
<code>:active</code>	cuando se hace clic sobre el elemento	
<code>:first-child</code>	primer elemento hijo	
<code>:focus</code>	cuando el elemento tiene el foco	
<code>:hover</code>	cuando el ratón pasa sobre el elemento	
<code>:lang</code>	idioma	<code>:lang(en)</code>   <code>:lang(es)</code>   <code>:lang(fr)</code> etc.
<code>:link</code>	enlaces no visitados	
<code>:visited</code>	enlaces ya visitados	

### Propiedades no incluidas en CSS 2.1 ni en CSS 3

Propiedad	Descripción	Valores
<code>marker-offset</code>	distancia entre marcador y elemento	distancia
<code>marks</code>	marcas de corte de papel	<code>crop</code>   <code>cross</code>

## 1.15. Atributos de estilo para fuente

Estas propiedades permiten al autor de una página web elegir con todo detalle el tipo de letra que se mostrará en el navegador del usuario.

El autor de la página web puede entregar la fuente al navegador mediante el mecanismo de las fuentes web o esperar que el usuario tenga instalada la fuente deseada en su ordenador. Si la fuente solicitada no está instalada, en algunos de esos casos los navegadores hacen una transformación por software que imita el resultado esperado y en otros no.

Se explican a continuación las propiedades que permiten describir el tipo de letra (la fuente) en una página web:

- Interlineado: **line-height** (no es propiamente una característica del tipo de letra, pero se comenta en esta lección)
- La propiedad compuesta **font**
- Tipo de letra: **font-family**
- Tamaño del tipo de letra: **font-size**
- Subfamilias tipográficas
  - Grosor del trazo (negrita): **font-weight**
  - Inclinación (italica): **font-style**
  - Variantes: **font-variant**
    - Versalitas: **font-variant-caps**
    - *Kerning*: **font-kerning**
    - Ligaduras: **font-variant-ligatures**
    - Formato de números: **font-variant-numeric**
    - Posición: **font-variant-posititon**
    - Tamaño y sustitución de caracteres asiáticos: **font-variant-east-asian**
    - Características de fuentes OpenType: **font-feature-settings**
- Síntesis: **font-synthesis**
- Ajuste del tamaño: **font-size-adjust**
- Anchura: **font-stretch**

### Interlineado: **line-height**

La propiedad **line-height** permite establecer el interlineado de un bloque de texto. El valor se puede expresar en cualquier unidad de longitud, aunque lo más lógico es utilizar porcentajes o em. Si se expresa en porcentajes, el valor 100% corresponde al interlineado normal.

```
p {  
    line-height: 100%;  
}
```

Este párrafo tiene un interlineado del **100%**, es decir un interlineado sencillo. Este es el mismo interlineado que se aplica si no se utiliza esta propiedad.

```
p {  
    line-height: 200%;  
}
```

Este párrafo tiene un interlineado del **200%**, es decir un interlineado doble. Las líneas de texto están separadas el doble de lo normal.

```
p {  
    line-height: 150%;  
}
```

Este párrafo tiene un interlineado del **150%**, es decir un interlineado de 1,5 líneas. Las líneas de texto están separadas un 50% más de lo normal.

```
p {  
    line-height: 75%;  
}
```

Este párrafo tiene un interlineado del **75%**, es decir un interlineado de 3/4 líneas. Las líneas de texto están más juntas de lo normal, lo que puede dificultar su lectura.

```
p {  
    line-height: 1.5em;  
}
```

Este párrafo tiene un interlineado del **1.5em**, es decir un interlineado de 1,5 líneas. Las líneas de texto están separadas un 50% más de lo normal.

### La propiedad compuesta **font**

La propiedad compuesta **font** permite definir simultáneamente las propiedades relacionadas con el tipo de letra: **font-style**, **font-variant** (solamente los valores definidos en CSS 2.1 **normal** y **small-caps**), **font-weight**, **font-stretch**, **font-size**, **line-height** y **font-family** (explicadas más adelante).

Al escribir la propiedad compuesta, el orden en que deben aparecer los valores es el orden en que aparecen las propiedades simples en el párrafo anterior. Las únicas propiedades que deben aparecer obligatoriamente son **font-size** y **font-family**. Si se escribe la propiedad **line-height** debe aparecer separada de **font-size** por una barra (/).

```
p {  
    font: 150% sans-serif;  
}
```

Esto es un párrafo con fuente **sans-serif** (en Windows suele ser Arial) al 150%.

```
p {  
    font: bold 90% monospace;  
}
```

Esto es un párrafo con fuente **monospace** (en Windows suele ser Courier) al 90% y en negrita.

```
p {  
    font: oblique bold 120% cursive;  
}
```

Esto es un párrafo con fuente **cursive** (en Windows suele ser Comic Sans) al 120%, en negrita y en cursiva.

```
p {  
    font: small-caps 100%/200% serif;  
}
```

ESTO ES UN PÁRRAFO CON FUENTE **SERIF** (EN WINDOWS SUELE SER TIMES NEW ROMAN) AL 100%, CON UN INTERLINEADO DOBLE, Y EN VERSALITAS.

Si las propiedades no están en el orden correcto, o falta alguna de las propiedades obligatorias, los navegadores no aplican ninguna propiedad.

✗ 

```
p {  
    font: monospace 150%;  
    /* el orden no es correcto */  
}
```

Esto intentaba ser un párrafo con fuente **monospace** (en Windows suele ser Courier) al 150%, pero la propiedad no está escrita correctamente.

✗ 

```
p {  
    font: bold 90%;  
    /* falta font-type */  
}
```

Esto intentaba ser un párrafo al 90% y en negrita, pero la propiedad no está escrita correctamente.

✗ 

```
p {  
    font: oblique 120% sans-serif bold;  
    /* bold debería estar al principio */  
}
```

Esto intentaba ser un párrafo con fuente **sans-serif** (en Windows suele ser Arial) en cursiva, al 120%, y en negrita, pero la propiedad no está escrita correctamente.

Si los navegadores no entienden alguna de las propiedades incluidas en la propiedad compuesta, aplican el resto de propiedades.

En el ejemplo siguiente, los navegadores no aplican el valor **condensed**, aunque sí los valores **120%** y **sans-serif**. El motivo por el que no aplican la propiedad **font-stretch** es porque la fuente no incluye esa variante (en el apartado dedicado a la propiedad font-stretch se comenta cómo conseguir que la apliquen, utilizando la regla-arroba **@font-face**).

```
p {  
    font: condensed 120% sans-serif;  
}
```

Esto es un párrafo con fuente **sans-serif** (en Windows suele ser Arial) al 120% y condensada.

Además, la propiedad **font** admite los valores **caption**, **icon**, **menu**, **message-box**, **small-caption** y **status-bar**, que hacen referencia a los tipos de letra que utiliza el sistema operativo para esos elementos.

<pre>p {     font: caption; }</pre>	Este párrafo se muestra como un <b>caption</b> .
<pre>p {     font: icon; }</pre>	Este párrafo se muestra como un <b>icon</b> .
<pre>p {     font: menu; }</pre>	Este párrafo se muestra como un <b>menu</b> .
<pre>p {     font: message-box; }</pre>	Este párrafo se muestra como un <b>message-box</b> .
<pre>p {     font: small-caption; }</pre>	Este párrafo se muestra como un <b>small-caption</b> .
<pre>p {     font: status-bar; }</pre>	Este párrafo se muestra como un <b>status-bar</b> .

### Tipo de letra: **font-family**

La propiedad **font-family** permite establecer el tipo de letra (fuente) del elemento.

La propiedad **font-family** puede tomar como valor el nombre concreto de una fuente. El nombre de la fuente debe escribirse entre comillas si contiene espacios (y puede escribirse entre comillas si no contiene espacios). Si el ordenador no tiene instalada la fuente indicada, el navegador utilizará una fuente de la familia genérica **serif** (que se explica a continuación). En los ejemplos siguientes si no está instalada alguna de las fuentes Arial, Segoe UI u Homer Simpson, el ejemplo se verá en el mismo tipo de letra que el cuarto ejemplo.

<pre>p {     font-family: Arial; }</pre>	Esto es un párrafo con fuente <b>Arial</b> .
<pre>p {     font-family: "Segoe UI"; }</pre>	Esto es un párrafo con fuente <b>Segoe UI</b> .
<pre>p {     font-family: "Homer Simpson UI"; }</pre>	Esto es un párrafo con fuente <b>Homer Simpson</b> .
<pre>p {     font-family: serif; }</pre>	Esto es un párrafo con fuente <b>serif</b> .

Para evitar este problema se puede utilizar uno de los nombres de familia de fuentes genéricos **serif**, **sans-serif**, **monospace**, **cursive** y **fantasy**. Los navegadores tienen asociados a cada nombre genérico una fuente que sí que está instalada en el ordenador (salvo en el caso de la familia genérica **fantasy** que puede no tener asociada ninguna fuente especial).

```
p {  
    font-family: serif;  
}
```

Esto es un párrafo con fuente **serif** (en Windows suele ser Times New Roman).

```
p {  
    font-family: sans-serif;  
}
```

Esto es un párrafo con fuente **sans-serif** (en Windows suele ser Arial).

```
p {  
    font-family: monospace;  
}
```

Esto es un párrafo con fuente **monospace** (en Windows suele ser Courier New).

```
p {  
    font-family: cursive;  
}
```

Esto es un párrafo con fuente **cursive** (en Windows suele ser Comic Sans).

```
p {  
    font-family: fantasy;  
}
```

Esto es un párrafo con fuente **fantasy** (en Windows cada navegador aplica una fuente distinta).

serif

afim10

sans-serif

afim10

monospace

afim10

cursive

afim10

Si el valor de la propiedad **font-family** son varios nombres de fuentes separados por comas, el navegador aplicará la primera fuente de la lista que esté instalada en el ordenador. Por ello es conveniente incluir en último lugar uno de los nombres genéricos, por si acaso el resto de fuentes no estuvieran disponibles. Hay que tener en cuenta también que, si una fuente no contiene un carácter determinado, el navegador busca ese carácter en las siguientes fuentes de la lista.

```
p {  
    font-family: "Constantia", serif;  
}
```

Esto es un párrafo con fuente "Constantia". Si no está instalada en el ordenador se verá como **serif**.

Esto es un párrafo con fuente **serif** (en Windows suele ser Times New Roman).

```
p {  
    font-family: "Segoe UI", sans-serif;  
}
```

Esto es un párrafo con fuente "Segoe UI". Si no está instalada en el ordenador se verá como **sans-serif**.

Esto es un párrafo con fuente **sans-serif** (en Windows suele ser Arial).

```
p {  
    font-family: "Consolas", monospace;  
}
```

Esto es un párrafo con fuente "Consolas". Si no está instalada en el ordenador se verá como **monospace**.

Esto es un párrafo con fuente **monospace** (en Windows suele ser Courier New).

Los nombres de fuentes se pueden escribir en mayúsculas o minúsculas. En los ejemplos siguientes se escribe la fuente Arial de cuatro maneras distintas. La última forma es incorrecta (porque el nombre de la fuente está escrito con espacios), por lo que el ejemplo se ve con una fuente **serif**.

**Nota:** En una enumeración de fuentes, lo lógico es poner al final la familia genérica a la que pertenecen el resto de fuentes. Así, en los ejemplos siguientes lo lógico sería haber escrito la familia genérica **sans-serif**. Pero estos ejemplos no se han hecho así para que se pueda identificar fácilmente cuándo el navegador no puede aplicar la fuente (concretamente, en el último ejemplo).

```
p {  
    font-family: "arial", serif;  
}
```

Esto es un párrafo con fuente "arial". Si no está instalada, se verá como **serif** (en Windows suele ser Times New Roman).

```
p {  
    font-family: "ARIAL", serif;  
}
```

Esto es un párrafo con fuente "ARIAL". Si no está instalada, se verá como **serif** (en Windows suele ser Times New Roman).

```
p {  
    font-family: "AriaL", serif;  
}
```

Esto es un párrafo con fuente "AriaL". Si no está instalada, se verá como **serif** (en Windows suele ser Times New Roman).

```
p {  
    font-family: "a r i a l", serif;  
}
```

Esto es un párrafo con fuente "a r i a l". Si no está instalada, se verá como **serif** (en Windows suele ser Times New Roman).

### Tamaño del tipo de letra: **font-size**

La propiedad **font-size** permite establecer el tamaño del tipo de letra (fuente) del elemento. Se puede expresar el tamaño de varias maneras: tamaño absoluto, tamaño relativo, distancia o porcentaje, aunque generalmente se aconseja utilizar unidades relativas (% o em).

#### Tamaño absoluto

El navegador tiene definidos una serie de tamaños que se llaman **xx-small**, **x-small**, **small**, **medium**, **large**, **x-large** y **xx-large**. Estos tamaños el navegador los elige en función de la resolución de la pantalla, el tamaño de la fuente, etc.

A continuación, se encuentra cada uno de los nombres de tamaños absolutos a su propio tamaño: **xx-small**, **x-small**, **small**, **medium**, **large**, **x-large** y **xx-large**.

```
p {  
    font-size: x-large;  
}
```

Esto es un párrafo de tamaño x-large.

#### Tamaño relativo

Se pueden utilizar los valores **larger** o **smaller**. Estos valores se interpretan con respecto al elemento padre y se toman el valor siguiente o anterior de la lista de tamaños absolutos.

En el ejemplo siguiente, la hoja de estilo establece que una lista tiene que tener un tamaño menor que el elemento padre. Si se escriben listas anidadas, cada sublista es cada vez más pequeña. Cuando se llega al límite inferior, el navegador puede seguir reduciendo la letra o dejarla en **xx-small**.

```
ul {  
    font-size: smaller;  
}
```

- Lista 1
  - Lista 2
    - Lista 3
      - Lista 4

#### Valor numérico con unidades absolutas

Se puede definir un tamaño concreto, por ejemplo, 14pt, 3cm, 20px, etc.

**Nota:** Hace años (antes de 2010) se desaconsejaba utilizar estas unidades de forma indiscriminada porque los navegadores no permitían reducir ni ampliar los elementos definidos mediante unidades absolutas, pero actualmente no existe esa limitación.

```
p {  
    font-size: 16pt;  
}
```

Esto es un párrafo de tamaño 16pt.

### Valor numérico con unidades relativas

Se puede definir el tamaño de cada elemento mediante porcentajes (o **em** o **rem**, teniendo en cuenta que  $1\text{em} = 100\%$  y que  $1\text{rem} = 100\%$ ), que se interpretan con respecto al tamaño base.

```
p {  
    font-size: 150%;  
}
```

Esto es un párrafo de tamaño 150%.

```
p {  
    font-size: 120%;  
}
```

Esto es un párrafo de tamaño 120%.

```
p {  
    font-size: 80%;  
}
```

Esto es un párrafo de tamaño 80%.

### Subfamilias tipográficas

En Tipografía existen lo que se llaman subfamilias tipográficas, es decir, fuentes que son variaciones de una fuente determinada. Algunas de las posibles variaciones son:

- el grosor del trazo
- la inclinación
- Otras características

Las propiedades **font-weight**, **font-style** y **font-variant** le indican al navegador qué variación es la que debería mostrar. El navegador es el que tiene que decidir qué fuente (de las instaladas en el ordenador en el que se están viendo las páginas) debe utilizar o, si es capaz de ello, simular la propiedad.

#### Grosor del trazo (negrita): **font-weight**

La propiedad **font-weight** permite elegir el grosor del trazo. Existen nueve valores numéricos (100, 200, 300, 400, 500, 600, 700, 800, 900), del más fino al más grueso. Además, existen los valores **normal** y **bold** que deben coincidir respectivamente, con los valores numéricos 400 y 700. Y, por último, existen los valores **lighter** y **bolder** que significan, respectivamente, un valor inferior y superior en la lista de valores numéricos y se interpretan respecto del elemento padre.

Al definir el grosor con valores numéricos, los navegadores muestran tres grosores distintos (de 100 a 500 como **normal**, 600 y 700 como **bold** y 800 y 900 más grueso que **bold**). Esos grosores no se aprecian de la misma manera en cada tipo de letra. En la siguiente lista, cada valor está en el grosor indicado por el propio valor:

- Fuente **sans-serif**
  - 100, 200, 300, 400, 500, **600, 700, 800, 900**
  - **normal** y **bold**
  - **lighter** y **bolder**
- Fuente **serif**
  - 100, 200, 300, 400, 500, **600, 700, 800, 900**
  - **normal** y **bold**
  - **lighter** y **bolder**
- Fuente **monospace**
  - 100, 200, 300, 400, 500, **600, 700, 800, 900**
  - **normal** y **bold**

- lighter y **bolder**
- Fuente **cursive**
  - 100, 200, 300, 400, 500, **600, 700, 800, 900**
  - normal y **bold**
  - lighter y **bolder**

```
p { font-family: sans-serif; }

p.g400 { font-weight: 400; }

p.n { font-weight: normal; }

p.g700 { font-weight: 700; }

p.b { font-weight: bold; }

p.g900 { font-weight: 900; }
```

Este párrafo está en negrita (400).  
Este párrafo está en negrita (normal).  
Este párrafo está en negrita (700).  
Este párrafo está en negrita (bold).  
**Este párrafo está en negrita (900).**

El valor **bolder** hace que el elemento se muestre, si es posible, con un grosor mayor que el del elemento en el que está incluido. En el siguiente ejemplo, cada lista se muestra con un grosor mayor que la anterior:

```
p {
  font-weight: normal;
}

ul {
  font-family: sans-serif;
  font-weight: bolder;
}
```

Este es un párrafo normal.

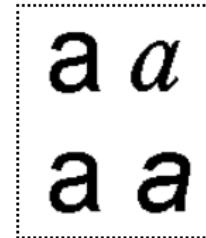
- Lista con trazo más grueso
  - **Lista con trazo todavía más grueso**
    - **Lista con trazo todavía más grueso**
      - **Lista con trazo todavía más grueso**

### Inclinación (italica): **font-style**

Una forma de obtener una variante de un tipo de letra es inclinar las letras. En Tipografía se distinguen dos formas de inclinación:

- oblicua: la forma de la letra no cambia, simplemente se inclina
- itálicas: además de inclinarse, la forma de algunas letras cambia

Para distinguir si una fuente es oblicua o itálica, es necesario fijarse en letras como la "a". La imagen siguiente muestra un ejemplo: la primera línea corresponde a la fuente Aldine (normal e itálica) y la segunda línea corresponde a la fuente Arial (normal y oblicua).



La propiedad **font-style** permite elegir la inclinación: **normal**, **oblique** o **italic**. Para que los navegadores puedan mostrar las variantes oblicua e itálica es necesario que en el ordenador esté instaladas fuentes de ambas variantes, lo que no suele ser habitual.

En general, si no hay instaladas fuentes específicas, los navegadores simplemente inclinan el tipo de letra normal, tanto para **oblique** como **italic**. Como esto correspondería más bien a la variante oblicua, se aconseja dar el valor **oblique** en vez de **italic**.

```
p {
  font-family: sans-serif;
  font-size: 150200%;
  font-style: normal;
}
```

Este párrafo está en estilo **normal**.

```
p {
  font-family: sans-serif;
  font-size: 150%;
  font-style: italic;
}
```

Este párrafo está en estilo **italic**.

```
p {
  font-family: sans-serif;
  font-size: 150%;
  font-style: oblique;
}
```

Este párrafo está en estilo **oblique**.

Para probar si realmente los navegadores distinguen entre **italic** y **oblique**, se debe instalar una fuente que tenga ambas variantes. Por ejemplo, la fuente libre [Latin Modern Roman](#). El ejemplo siguiente debería poder verse correctamente una vez instalada esa fuente (realmente son 11 fuentes distintas para cubrir las variantes más habituales).



```
p.italic {  
    font-family: "Latin Modern Roman 10";  
    font-style: italic;  
}  
  
p.oblique {  
    font-family: "Latin Modern Roman 10";  
    font-style: oblique;  
}
```

Esto es un párrafo con Latin Modern Roman.

*Esto es un párrafo con Latin Modern Roman italic.*

*Esto es un párrafo con Latin Modern Roman oblique.*

**Nota:** Firefox aplica esta propiedad correctamente, mientras que Chrome no lo hace.

### Características tipográficas

Las fuentes en formato [OpenType](#) pueden incluir muchas características tipográficas ([lista de características en Wikipedia](#)). En CSS 3 se han añadido propiedades para poder acceder a esas características. Por desgracia, las fuentes incluidas en los sistemas operativos no disponen de esas características, así que es imposible aprovecharlas.

En Internet se pueden encontrar fuentes que sí incluyen algunas de esas características y podemos utilizarlas mediante el mecanismo de las [fuentes web](#). En los ejemplos de este apartado se utilizan algunas de estas fuentes.

#### *La propiedad compuesta **font-variant***

En CSS 2, la propiedad **font-variant** únicamente admitía los valores **normal** y **small-caps**, que permite mostrar un tipo de letra en versalitas.

Las versalitas son tipos en los que todos los caracteres están en mayúsculas, aunque con el tamaño de las minúsculas (en principio, con la altura de la letra x del tipo, aunque no siempre es así).

```
p {  
    font-variant: normal;  
}
```

Este párrafo está en el tipo **normal**: aAbBcCdDeEff.

```
p {  
    font-variant: small-caps;  
}
```

ESTE PÁRRAFO ESTÁ EN VERSALITAS: AABBCDDEEFF.

En CSS 3, **font-variant** se ha convertido en una propiedad compuesta, que además de las versalitas permite seleccionar varias características tipográficas disponibles en las fuentes OTF (y en algunos casos, TTF).

#### *Versalitas: **font-variant-caps***

La propiedad **font-variant-caps** permite mostrar un tipo de letra en versalitas (tipo de letra en el que las letras minúsculas tienen la forma pero no el tamaño de las letras mayúsculas). Admite los valores:

- **normal**: minúsculas y mayúsculas tienen formas distintas
- **small-caps**: minúsculas y mayúsculas tiene la misma forma pero tamaños distintos
- **all-small-caps**: minúsculas y mayúsculas tiene la misma forma y tamaño
- **petite-caps**: minúsculas y mayúsculas tiene la misma forma pero tamaños distintos (inferior a **small-caps**)
- **all-petite-caps**: minúsculas y mayúsculas tiene la misma forma y tamaño (inferior a **small-caps**)
- **unicase**: sólo las mayúsculas se muestran en versalitas

- **titling-caps**: variante de versalitas en las que las mayúsculas son más finas que las mayúsculas en versalitas estándar

	<pre>p {     font-variant-caps: normal; }</pre>	Este párrafo está en el tipo <b>normal</b> : aAbBcCdDeEfF.
	<pre>p {     font-variant: small-caps; }</pre>	ESTE PÁRRAFO ESTÁ EN VERSALITAS: AABBCdDeEfF.
	<pre>p {     font-variant: all-small-caps; }</pre>	ESTE PÁRRAFO ESTÁ EN VERSALITAS: AABCCDDEEFF.
	<pre>p {     font-variant: petite-caps; }</pre>	ESTE PÁRRAFO ESTÁ EN VERSALITAS: AABBCdDeEfF.
	<pre>p {     font-variant: all-petite-caps; }</pre>	ESTE PÁRRAFO ESTÁ EN VERSALITAS: AABCCDDEEFF.
	<pre>p {     font-variant: unicase; }</pre>	Este párrafo está en <b>versalitas</b> : aAbBcCdDeEfF.
	<pre>p {     font-variant: titling-caps; }</pre>	Este párrafo está en <b>versalitas</b> : aAbBcCdDeEfF.

### Kerning: *font-kerning*

La propiedad **font-kerning** permite aplicar *kerning* (interletraje) al tipo de letra. El *kerning* es la posibilidad de acercar más o menos dos letras consecutivas aprovechando la forma de las letras. Admite los valores **auto**, **normal** y **none**.

<pre>p { }</pre>	VA V. WA ft
<pre>p {     font-kerning: auto; }</pre>	VA V. WA ft
<pre>p {     font-kerning: normal; }</pre>	VA V. WA ft
<pre>p {     font-kerning: none; }</pre>	VA V. WA ft

### Ligaduras: *font-variant-ligatures*

La propiedad **font-variant-ligatures** permite aplicar ligaduras en el tipo de letra. Las *ligaduras* es la posibilidad de unir dos caracteres consecutivos. Admite los valores **normal**, **none**, **common-ligatures**, **no-common-ligatures**, **discretionary-ligatures**, **no-discretionary-ligatures**, **historical-ligatures**, **no-historical-ligatures**, **contextual** y **no-contextual**.

<pre>p {     font-variant-ligatures: normal; }</pre>	
<pre>p {     font-variant-ligatures: none; }</pre>	

Se suele dar como ejemplo de ligadura la unión de letras "f" consecutivas, como muestra el siguiente ejemplo en el que las letras "f" consecutivas se muestran sin espacio entre ellas:

```
p {  
}
```

f ff ftft flfl fff

Pero en realidad no es una ligadura sino un efecto de *kerning*, como demuestra el siguiente ejemplo en el que al desactivar el *kerning* las letras "f" consecutivas se muestran entonces con espacio entre ellas:

```
p {  
    font-kerning: none;  
}
```

f ff ftft flfl fff

### Formato de números: *font-variant-numeric*

La propiedad **font-variant-numeric** permite mostrar números con formatos especiales. Admite los valores **normal**, **lining-nums**, **oldstyle-nums**, **proportional-nums**, **tabular-nums**, **diagonal-fractions**, **stacked-fractions**, **ordinal** y **slashed-zeros**.

```
p {  
    font-variant-numeric: proportional-nums;  
}
```

0123456789  
oooooooooooo  
1111111111

```
p {  
    font-variant-numeric: tabular-nums;  
}
```

0123456789  
0000000000  
1111111111

### Posición: *font-variant-position*

La propiedad **font-variant-position** permite elegir la posición del carácter. Admite los valores **normal**, **sub** y **super**.

### Ajuste del tamaño del tipo de letra: *font-size-adjust*

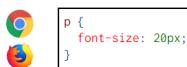
Cada tipo de letra tiene una proporción distinta entre el tamaño del tipo y el tamaño de la letra x de ese mismo tipo (para entendernos, una proporción distinta entre los tamaños de las letras mayúsculas y minúsculas). Es decir, que para un tamaño determinado, unos tipos de letras tienen las letras minúsculas más grandes que otros (y por tanto, son más legibles a tamaños pequeños).

Esto puede provocar problemas cuando el navegador no pueden mostrar el tipo de letra indicado por el autor y debe sustituirlo por otro. Si el tipo de letra que elige el navegador tiene unas minúsculas más pequeñas que las del tipo de letra elegido por el autor de la página, puede que el texto sea ilegible, ya que la propiedad **font-size** únicamente indica el tamaño del tipo.

La propiedad **font-size-adjust** permite establecer la proporción entre el tamaño del tipo y el de la letra x, indicándolo como número decimal.

**Nota:** La propiedad **font-size-adjust** se introdujo en la recomendación [CSS 2](#), pero no se incluyó en la recomendación [CSS 2.1](#), por lo que en CSS 2 se consideraba obsoleta. Sin embargo, esta propiedad se recuperó en la recomendación [CSS 3: Fuentes](#).

Firefox interpreta parcialmente esta propiedad, mientras que Google Chrome no interpreta esta propiedad.



```
p {  
    font-size: 20px;  
}
```

Esta letra x tiene 20 px de altura.



```
p {  
    font-size: 20px;  
    font-size-adjust: 1.5;  
}
```

Esta letra x debería tener  $20 \times 1.5 = 30$  px de altura.



```
p {  
    font-size: 20px;  
    font-size-adjust: 0.5;  
}
```

Esta letra x debería tener  $20 \times 0.5 = 10$  px de altura.

### Anchura: **font-stretch**

La propiedad **font-stretch** permite elegir que un tipo de letra más o menos condensado o expandido. Los valores permitidos son los siguientes: **ultra-condensed**, **extra-condensed**, **condensed**, **semi-condensed**, **normal**, **semi-expanded**, **expanded**, **extra-expanded** y **ultra-expanded**.

**Nota:** La propiedad **font-stretch** se introdujo en la recomendación [CSS 2](#), pero no se incluyó en la recomendación [CSS 2.1](#), por lo que en CSS 2 se consideraba obsoleta. Sin embargo, esta propiedad se recuperó en la recomendación [CSS 3: Fuentes](#).

Los tipos de letra no suelen incluir todas estas versiones, y como los navegadores no la emulan por software, esta propiedad no tiene efecto en la mayoría de tipos de letra.

```
p {  
    font-stretch: ultra-expanded;  
}
```

Este párrafo debería verse en el tipo ultra-extendido.

El problema es que las fuentes expandidas y condensadas se suelen distribuir como fuentes distintas. Una forma de poder utilizar esta propiedad consiste en asociar las variantes a la propiedad **font-stretch** mediante una regla-arroba **@font-face**.

```
@font-face {  
    font-family: "Encode Sans";  
    font-stretch: normal;  
    src: url("encode-sans-v2-latin-regular.woff") format("woff");  
}  
  
@font-face {  
    font-family: "Encode Sans";  
    font-stretch: condensed;  
    src: url("encode-sans-condensed-v2-latin-regular.woff") format("woff");  
}  
  
@font-face {  
    font-family: "Encode Sans";  
    font-stretch: expanded;  
    src: url("encode-sans-expanded-v2-latin-regular.woff") format("woff");  
}  
  
.condensed {  
    font-stretch: condensed;  
}  
  
.expanded {  
    font-stretch: expanded;  
}  
  
p {  
    font-family: "Encode Sans";  
}
```

Esto es un párrafo en el tipo Encode Sans condensed.

Esto es un párrafo en el tipo Encode Sans normal.

Esto es un párrafo en el tipo Encode Sans expanded.

**Nota:** Este ejemplo utiliza fuentes en formato woff para que pueda verse correctamente en Internet Explorer 11 (IE 11 no admite fuentes woff2).

Las fuentes web son las fuentes que pueden mostrarse en el navegador sin necesidad de que el usuario las tenga instaladas en su ordenador ya que se descargan automáticamente.

## Fuentes Web

### *Fuentes WOFF (Web Open Font Format)*

En 2009 se desarrolló un formato de fuentes llamado WOFF (Web Open Font Format) pensado para la web. En realidad, no se trata de un nuevo formato, puesto que una fuente WOFF no es más que una fuente TrueType, OpenType, Open Font o SVG comprimida y con metadatos para indicar detalles como el origen de la fuente o la licencia.

En abril de 2010, Mozilla, Microsoft y Opera presentaron este formato al W3C, que aprobó la recomendación [WOFF File Format 1.0](#) en diciembre de 2012.

En marzo de 2018 se aprobó la recomendación [WOFF File Format 2.0](#). Gracias al nuevo algoritmo de compresión [Brotli](#), esta recomendación consigue reducir el tamaño de las fuentes a la cuarta parte, aumentar la velocidad de descompresión y reducir los requisitos de memoria, como se explica en el [WOFF 2.0 Evaluation Report](#) publicado por el W3C en marzo de 2016 o en este [artículo en lwn.net sobre WOFF2 \(13/11/2013\)](#).

El formato WOFF 2 es mucho mejor que el formato WOFF 1 y servicios como Google Fonts ya sólo ofrecen WOFF 2, pero Internet Explorer 11 no admite WOFF 2, por lo que todavía puede ser necesario seguir usando WOFF 1.

Para crear fuentes WOFF se puede utilizar webs como [Font Squirrel](#) o [Web Font Generator](#) en la que subiendo una fuente TTF u OTF podemos descargar la misma fuente en los formatos WOFF2, WOFF, EOT y SVG.

### *La regla arroba @font-face*

La recomendación [CSS 3: Fuentes](#), publicada en septiembre de 2018, permite la utilización de fuentes web mediante el uso de la regla-arroba `@font-face`, como muestra el ejemplo siguiente:

```
@font-face {  
    font-family: "Mystery Quest";  
    src: url("mystery-quest.woff2");  
}  
  
p {  
    font-family: "Mystery Quest", cursive;  
    font-size: 150%;  
}
```

Esto es un párrafo con fuente web Mystery Quest al 150%.

En la regla `@font-face`:

- La propiedad `font-family` establece cómo tenemos que llamar a la fuente en la hoja de estilo. Si el nombre contiene espacios en blanco, es necesario escribir el nombre entre comillas.
- La propiedad `src` indica el camino (absoluto o relativo) del fichero de la fuente. En el caso de indicar un camino relativo, el camino relativo es el camino relativo de la fuente con respecto a la hoja de estilo, no respecto a la página web.

Una vez definido el nombre de la fuente, se puede hacer referencia a ella en las propiedades `font-family`.

La recomendación no impone ningún formato específico para los tipos de letra, pero nombra algunos como posibilidad: woff (WOFF Web Open Font Format, .woff), truetype (TrueType, .ttf), opentype (OpenType, .ttf, .otf), embedded-opentype (Embedded OpenType, .eot), svg (SVG Font, .svg, .svgz).

Entre ellos, se recomienda utilizar el formato WOFF 2 (Web Open Font Format, .woff2), definido en la recomendación [WOFF File Format 2.0](#) de marzo de 2018, o en su defecto, el formato WOFF (.woff) definido en la recomendación [WOFF File Format 1.0](#) de diciembre de 2012.

**Nota:** La regla-arroba `@font-face` estaba incluida en la recomendación [CSS 2](#), pero no se incluyó en la recomendación [CSS 2.1](#), por lo que en CSS 2 se consideraba obsoleta. Sin embargo, esta regla-arroba se recuperó en la recomendación [CSS 3: Fuentes](#)

### Google Fonts / Google Fonts API

Google ofrece un servicio de alojamiento de fuentes libres, [Google Fonts](#).

Google Fonts permite descargar las fuentes en formato TTF a nuestro ordenador para utilizarlas por ejemplo en un procesador de textos. Para ello, hay añadir las fuentes deseadas a una colección y descargar la colección completa.

Lo más interesante de Google Fonts es que las fuentes pueden utilizarse en nuestras páginas web sin necesidad de alojarlas en nuestro propio servidor. Eso tiene la ventaja de que nos ahorramos el tráfico que genera la fuente, pero tiene el inconveniente de que en cualquier momento pueden no estar disponibles. Para intentar evitar este problema conviene indicar varias fuentes, poniendo en último lugar una de las familias genéricas.

Si queremos alojar nosotros las fuentes, Google Fonts ofrece la posibilidad de descargarla en formato TTF, pero a continuación podemos o bien convertir la fuente TTF al formato woff mediante alguna de las páginas web citadas en el apartado [Fuentes WOFF](#), o bien descargarla de Google Fonts utilizando el procedimiento comentado en el apartado [Descargar las fuentes de Google Fonts en formato WOFF2](#).

Las fuentes incluidas en Google Fonts también se encuentran disponibles como repositorio de GitHub: <https://github.com/google/fonts/>.

### Enlazar las fuentes de Google Fonts

Para utilizar Google Fonts, la documentación oficial propone dos métodos equivalentes, aunque se recomienda el primero ya que toda la información se encuentra en la hoja de estilo.

- Incluir al principio de la hoja de estilo una regla-arroba **@import** y la fuente se puede utilizar en la hoja de estilo como una familia más, como muestra el ejemplo siguiente:

```
@import url("https://fonts.googleapis.com/css?family=Mystery+Quest");  
p {  
  font-family: "Mystery Quest", cursive;  
  font-size: 150%;  
}
```

Este es un párrafo con fuente web Mystery Quest al 150% alojada en Google Fonts.

Las reglas **@import** deben escribirse obligatoriamente al principio de la hoja de estilo.

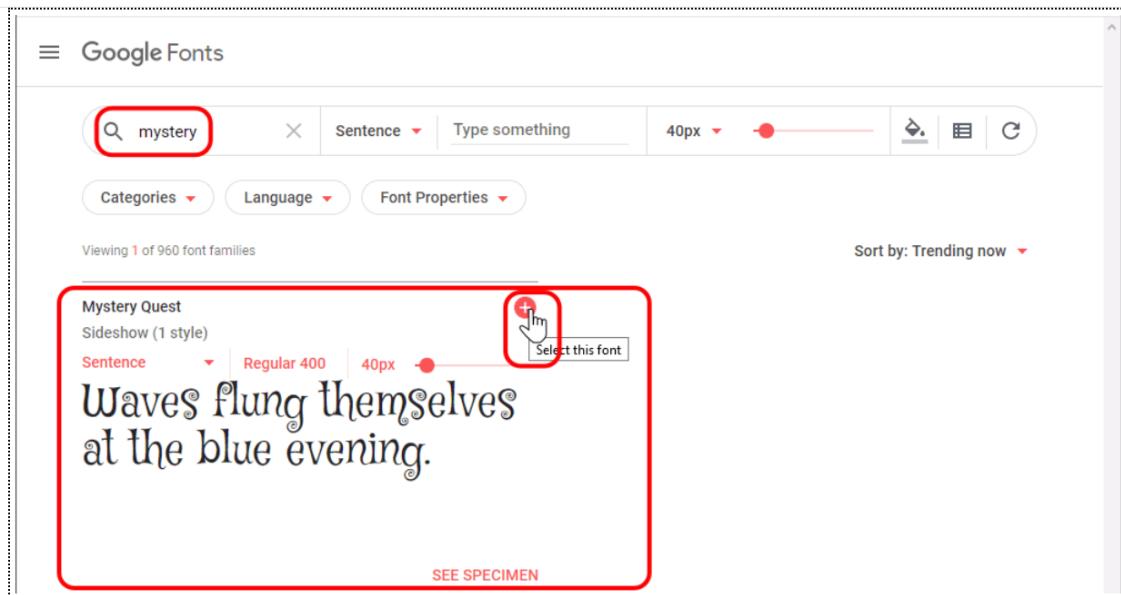
- Incluir en el documento html un enlace como si fuera un enlace a una hoja de estilo y la fuente se puede utilizar en la hoja de estilo como una familia más, como muestra el ejemplo siguiente:

```
<head>  
  ...  
  <link rel="stylesheet" href="https://fonts.googleapis.com/css?  
family=Mystery+Quest">  
  ...  
</head>  
  
p {  
  font-family: "Mystery Quest", cursive;  
  font-size: 150%;  
}
```

Este es un párrafo con fuente web Mystery Quest al 150% alojada en Google Fonts.

Se muestra a continuación cómo encontrar en Google Fonts el código de los ejemplos anteriores:

- Seleccione la fuente haciendo clic en el símbolo 



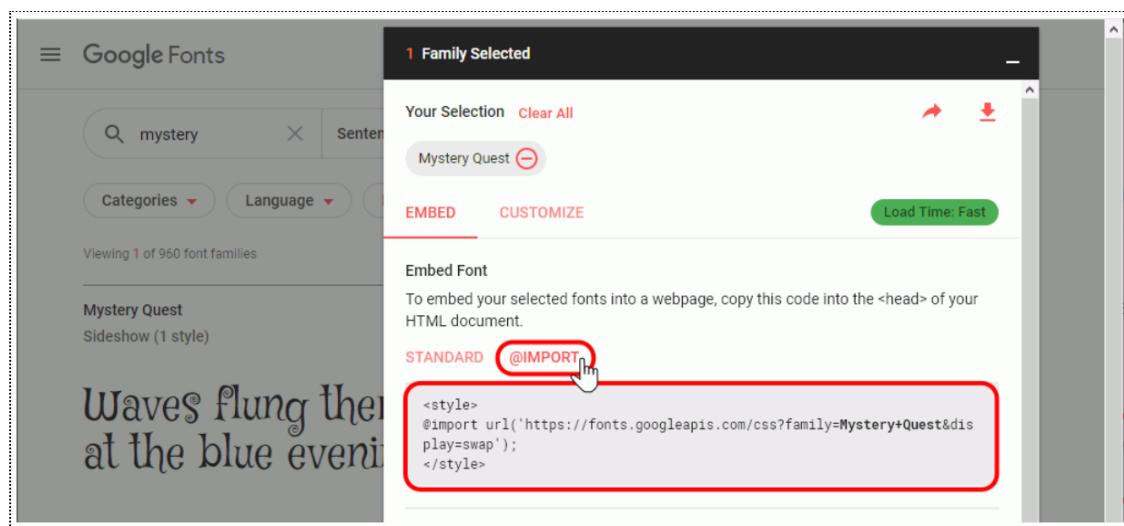
- Haga clic en la barra negra que se mostrará en la parte inferior de la ventana:

The screenshot shows the Google Fonts interface. A search bar at the top contains the text 'mystery'. Below it, a dropdown menu shows 'Sentence' selected. A text input field says 'Type something' with a '40px' dropdown next to it. To the right are icons for font preview, grid view, and refresh. Below these are buttons for 'Categories', 'Language', and 'Font Properties'. A status message 'Viewing 1 of 960 font families' is on the left, and 'Sort by: Trending now' is on the right. A red circle highlights the 'Mystery Quest' font family listed under 'Sideshow (1 style)'. A sample sentence 'Waves flung themselves at the blue evening.' is displayed in the font. At the bottom, a toolbar shows '1 Family Selected' with a hand cursor icon, and a close button.

Se mostrará el código a insertar como enlace a hoja de estilo. Se recomienda hacerlo como regla-arroba **@import**, que se muestra en el punto siguiente, porque de esa manera se escribe todo en la hoja de estilo:

This screenshot shows the same Google Fonts interface as above, but with a modal window open over it. The modal title is '1 Family Selected'. It displays the 'Your Selection' section with 'Mystery Quest' selected. Below are 'EMBED' and 'CUSTOMIZE' buttons, and a note about fast load time. The 'EMBED' section contains an 'STANDARD' button and an '@IMPORT' link. A red box highlights the '@IMPORT' link, which is followed by a red box containing the generated CSS code: `<link href="https://fonts.googleapis.com/css?family=Mystery+Quest&display=swap" rel="stylesheet">`

- Haga clic en **@import** y se mostrará el código a insertar como regla-arroba **@import** (recomendado):



### Nota:

- Por motivos de seguridad, Google Chrome y Firefox (a partir de la versión 26), obligan a que el contenido mixto activo se obtenga mediante https en vez de por http ([explicación en Mozilla Developer Network](#)). Es decir, que si una página que se sirve por https (conexión segura) incluye elementos enlazados por http (conexión insegura), estos elementos no se descargan.

Se recomienda por tanto que el enlace a Google Fonts se haga mediante https en vez de http, por si en el futuro nuestra página, que enlaza a Google Fonts, se sirve mediante https.

### *Descargar las fuentes de Google Fonts en formato WOFF2*

Cuando se descarga una fuente de [Google Fonts](#), tan sólo se incluye la fuente en formato TTF. Para obtenerla en formato WOFF2 puede utilizar alguno de estos procedimientos:

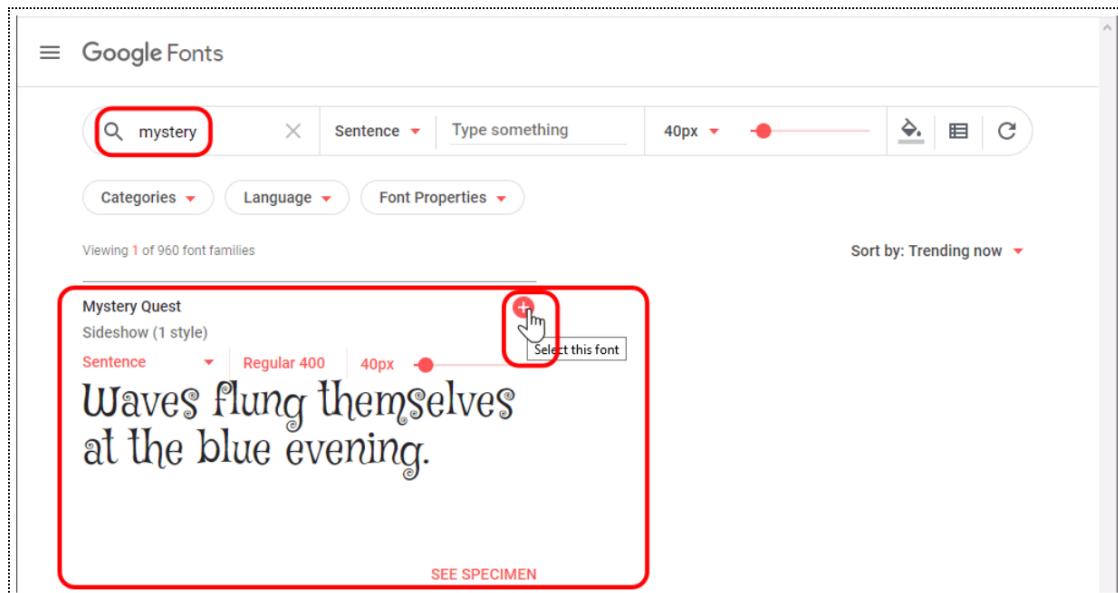
Una de las formas más sencillas es utilizar páginas como <https://canonicalized.com/local-font/>, que permiten elegir una fuente y descargar un fichero zip con la fuente en varios formatos (WOFF2, WOFF, TTF, EOT, SVG).



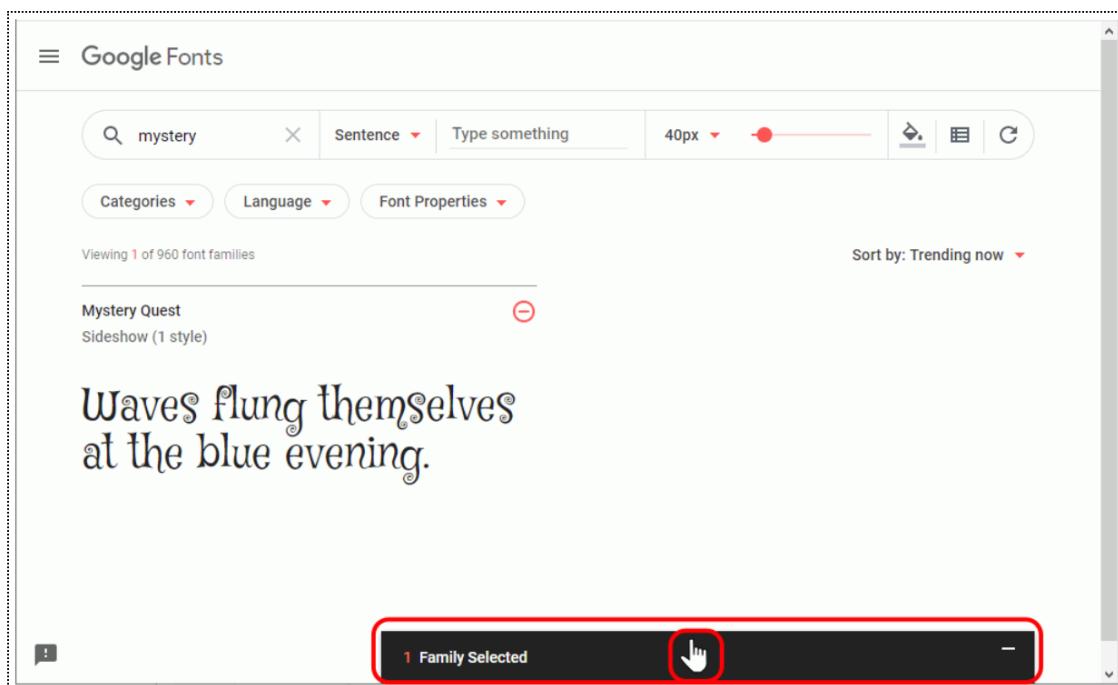
Si la página anterior no está disponible, puede descargar de Google Fonts la fuente en formato TTF y convertirlas utilizando páginas como [Font Squirrel](#) o [Web Font Generator](#).

Por último, también es posible descargar de Google Fonts una fuente en formato WOFF2, aunque el procedimiento es un poco laborioso:

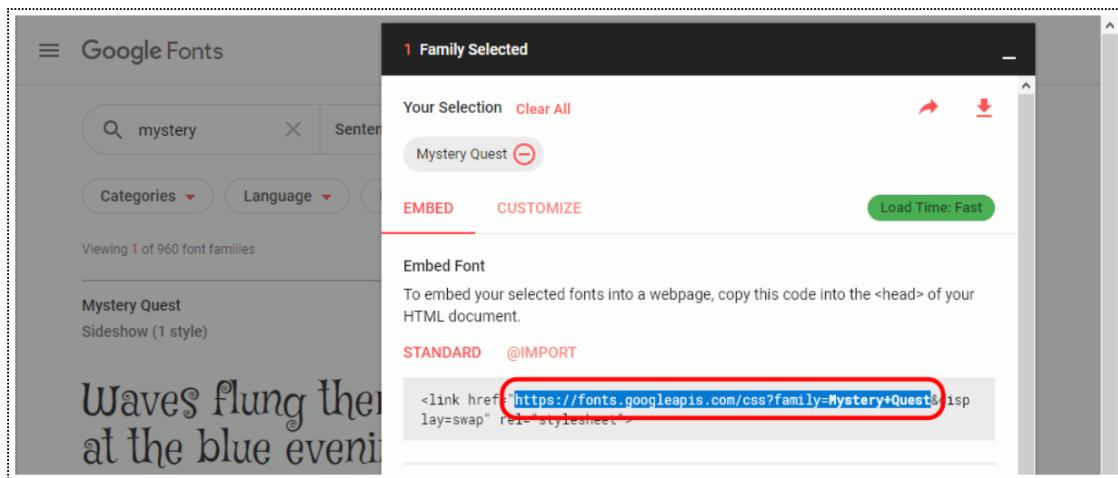
- Seleccione la fuente haciendo clic en el símbolo +:



- Haga clic en la barra negra que se mostrará en la parte inferior de la ventana:



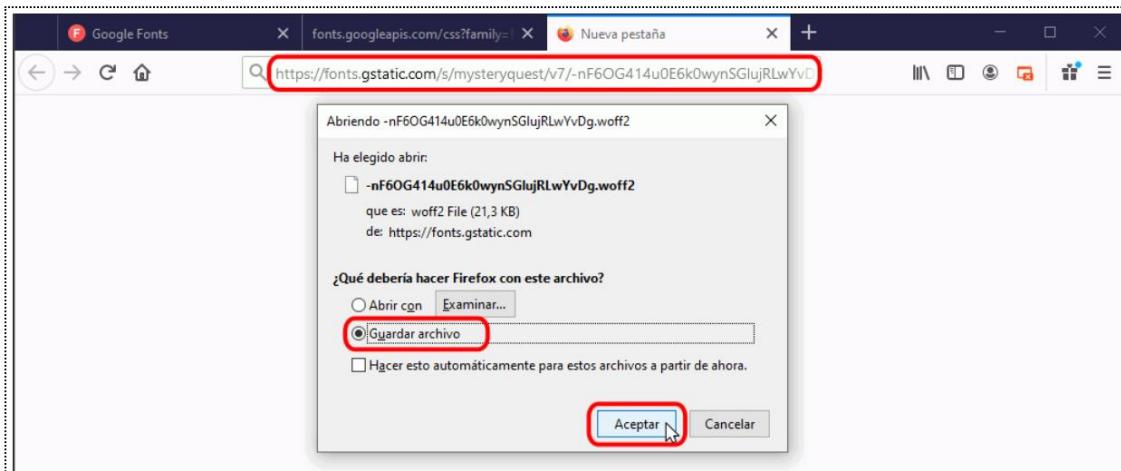
- Se mostrará el código a insertar como enlace a hoja de estilo. En la pestaña **standard** o la pestaña **@import**, seleccione y copie la dirección web <http://fonts.googleapis.com/css?family=...>:



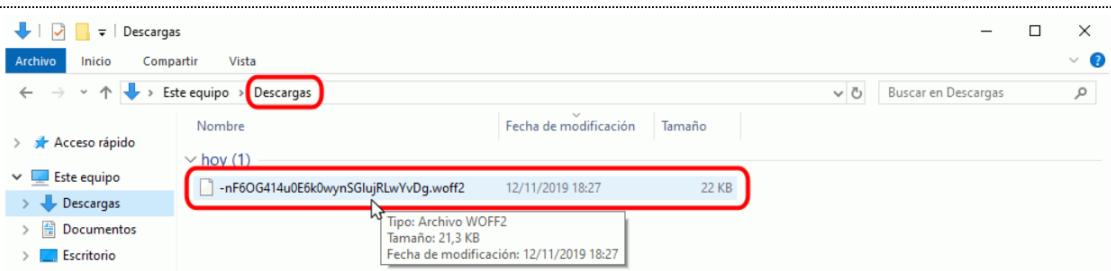
- Pegue esa dirección en la barra de dirección del navegador y pulse **Intro**. Se mostrará un documento de texto en el que se puede encontrar la dirección web de un archivo woff2. Puede haber varias versiones de la misma fuente (en la captura el documento contiene dos enlaces a diferentes versiones de la fuente). Normalmente, la versión a descargar es la versión **latin**, pero si esa versión no contiene todos los caracteres necesarios (acentos, etc.), a veces es necesario descargar la versión **latin-extended** u otras. Copie la dirección web de la versión elegida.



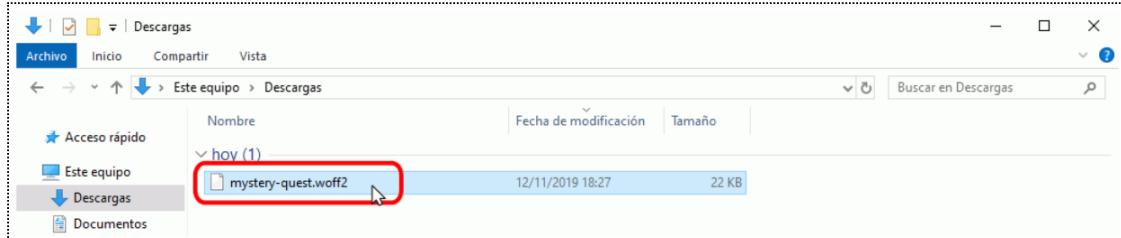
- Pegue esa dirección en la barra de dirección del navegador y pulse **Intro**. Automáticamente, el navegador le dará la opción de guardar la fuente. Haga clic en Guardar archivo y en Aceptar:



- En la carpeta de descargas se mostrará el archivo woff2 descargado:



- Cambie el nombre del fichero al nombre de la fuente para facilitar su uso posterior:



### Posición de `@import` en la hoja de estilo

La regla `@import` debe escribirse al principio de la hoja de estilo, sin otras reglas delante, ni siquiera reglas `@font-face`, como muestran los siguientes ejemplos:

✓ `@import url("https://fonts.googleapis.com/css?family=Indie+Flower");  
@font-face {  
 font-family: "Mystery Quest";  
 src: url("mystery-quest.woff2");  
}`

Esto es un párrafo con fuente web Indie Flower al 150% alojada en Google Fonts.

Esto es un párrafo con fuente web Mystery Quest al 150% alojada localmente.

✗ `@font-face {  
 font-family: "Mystery Quest";  
 src: url("mystery-quest.woff2");  
}  
  
@import url("https://fonts.googleapis.com/css?family=Indie+Flower");`

Esto es un párrafo con fuente web Indie Flower al 150% alojada en Google Fonts.

Esto es un párrafo con fuente web Mystery Quest al 150% alojada localmente.

### Propiedades a aplicar sobre texto

Se comentan a continuación las propiedades usuales que se pueden aplicar al texto de una página web:

## Color del texto: color

La propiedad **color** permite establecer el color del texto.

```
<body>
  <h1>Esto es un encabezado h1</h1>

  <p>Esto es un párrafo &lt;p&gt; normal y corriente.</p>

  <p>Esto es otro párrafo &lt;p&gt;.</p>
</body>
```

```
h1 {
  color: magenta;
}

p {
  color: blue;
}
```

**Esto es un encabezado h1**

Esto es un párrafo <p> normal y corriente.

Esto es otro párrafo <p>.

## Color de fondo: background-color

La propiedad **background-color** permite establecer el color de fondo del elemento.

```
body {
  background-color: aqua;
}

h1 {
  background-color: gold;
}

p {
  background-color: lightblue;
}
```

**Esto es un encabezado h1**

Esto es un párrafo <p> normal y corriente.

Esto es otro párrafo <p>.

El color de fondo de la página se establece en la etiqueta **<body>** (o en **<html>**)

El color de fondo de una etiqueta no se aplica solamente al texto, sino a la caja que contiene el texto (por eso el color de fondo se extiende hasta el extremo derecho de la línea)

## Alineación horizontal: text-align

La propiedad **text-align** permite establecer la alineación horizontal de un bloque de texto. Los posibles valores de **text-align** son **left** (izquierda), **right** (derecha), **center** (centro) y **justify** (justificado a derecha e izquierda).

```
<body>
  <h1>Esto es un encabezado h1</h1>
  <h2>Esto es un encabezado h2</h2>
  <p>Esto es un párrafo &lt;p&gt; normal y corriente que se muestra justificado a derecha e izquierda porque se ha establecido la propiedad en <span class="html-eti">&lt;body&gt;</span> y, por tanto, afecta a todos los elementos menos a los que tienen la propiedad explicitamente establecida con otro valor</p>
</body>
```

```
body {
    text-align: justify;
}

h1 {
    text-align: center;
}

h2 {
    text-align: right;
}
```

### Esto es un encabezado h1

### Esto es un encabezado h2

Esto es un párrafo `p` normal y corriente que se muestra justificado a derecha e izquierda porque se ha establecido la propiedad en `<body>` y, por tanto, afecta a todos los elementos menos a los que tienen la propiedad explicitamente establecida con otro valor

Para centrar una imagen, el elemento que contiene la imagen tiene que tener su contenido centrado. Por ejemplo, si la imagen está en un párrafo, se puede definir la clase "p.centrado" con el atributo **text-align** establecido con el valor **center** y después dar al párrafo que contiene la imagen la clase "centrado". Lógicamente, si se añade texto al párrafo, el texto también se centrará.

```
<p class="centrado">
    
</p>
```

```
p.centrado {
    text-align: center;
}
```



### Alineación vertical: **vertical-align**

La propiedad **vertical-align** permite establecer la alineación vertical de un elemento respecto al elemento que lo contiene. Los posibles valores de **vertical-align** son **baseline**, **sub**, **super**, **top**, **text-top**, **middle**, **bottom** y **text-bottom**.

```
<body>
    <pre style="font-weight: bold">Alineación: sub, bottom, middle, text-bottom, baseline, text-top, top, super.
    <span style="text-decoration:underline overline">Ejemplo: <span style="font-size: 300%"><span style="vertical-align: sub">bq</span> <span style="vertical-align: middle">bq</span> <span style="vertical-align: text-bottom">bq</span> <span style="vertical-align: baseline">bq</span> <span style="vertical-align: text-top">bq</span> <span style="vertical-align: top">bq</span>
    </pre>
```

Alineación: sub, bottom, middle, text-bottom, baseline, text-top, top, super.  
Ejemplo: bq bq bq bq bq bq bq bq

```
<body>
    <pre>Alineación: sub, bottom, middle, text-bottom, baseline, text-top, top, super.
    <span style="text-decoration:underline overline">Ejemplo:         </span></pre>
```

Alineación: sub, bottom, middle, text-bottom, baseline, text-top, top, super.  
Ejemplo: = = = = = = =

**Nota:** La posición relativa de los elementos es la misma en Chrome y Firefox, aunque las líneas horizontales que se han añadido a los ejemplos para facilitar la comparación no se muestran de la misma manera en estos navegadores

## Sangría: **text-indent**

La propiedad **text-indent** permite establecer la sangría de un bloque de texto. Su valor se puede establecer como cualquier otra longitud.

```
p {
    text-indent: 25px;
}
```

Esto es un párrafo <p> normal y corriente cuya primera línea tiene sangría, es decir, empieza más a la derecha que el resto de líneas del párrafo. La sangría es habitual en España (y no sólo para beberla).

```
p {
    text-indent: -25px;
}
```

Esto es un párrafo <p> normal y corriente cuya primera línea tiene sangría negativa (sangría francesa), por ello la primera linea empieza más a la izquierda que el resto de líneas del párrafo. Como se sale de la celda, se ve mal, pero poniendo margen al párrafo se corrige este problema (ver ejemplo siguiente).

```
p {
    margin-left: 25px;
    text-indent: -25px;
}
```

Esto es un párrafo <p> normal y corriente con margen izquierdo y cuya primera línea tiene sangría negativa del mismo valor que el margen (sangría francesa), por ello la primera linea empieza más a la izquierda que el resto de líneas del párrafo.

## Espaciado de letras: **letter-spacing**

La propiedad **letter-spacing** permite establecer un espaciado entre cada una de las letras de un bloque de texto. Estos valores, positivos o negativos, se establecen como cualquier otra longitud y se añaden al espaciado normal.

```
p {
    letter-spacing: 5px;
}
```

Esto es un párrafo con un espaciado entre letras mayor de lo normal.

```
p {
    letter-spacing: 10px;
}
```

Esto es un párrafo con un espaciado entre letras mayor de lo normal.

```
p {
    letter-spacing: -2px;
}
```

Esto es un párrafo con un espaciado entre letras negativo que apenas se puede leer.

La propiedad **letter-spacing** se aplica a todas las letras, incluida la última, como muestran los ejemplos siguientes en los que se añade un espacio final:

```
span {
    letter-spacing: 10px;
    text-decoration: underline;
}
```

L e t r a s   s e p a r a d a s

```
span {
    background-color: lightblue;
    letter-spacing: 10px;
}
```

L e t r a s   s e p a r a d a s

Este espacio final puede producir efectos no deseados, como en los ejemplos siguientes en que el texto parece no estar centrado (al poner color de fondo al texto se ve que el texto sí que está centrado, pero el espacio final hace que parezca que no).

```
p {
    letter-spacing: 50px;
    text-align: center;
}
```

X   X   X   X

```
p {
    letter-spacing: 50px;
    text-align: center;
}
```

X   X   X   X

```
span {
    background-color: lightblue;
}
```

## Espaciado de palabras: word-spacing

La propiedad **word-spacing** permite establecer un espaciado entre cada una de las palabras de un bloque de texto. Estos valores, positivos o negativos, se establecen como cualquier otra longitud y se añaden al espaciado normal.

```
p {
  word-spacing: 10px;
}
```

Esto es un párrafo con un espaciado entre palabras mayor de lo normal.

```
p {
  word-spacing: 20px;
}
```

Esto es un párrafo con un espaciado entre palabras mayor de lo normal.

```
p {
  word-spacing: -5px;
}
```

Esto es un párrafo sin espaciado entre palabras.

La propiedad **word-spacing** no se aplica a la última palabra.

```
p {
  text-align: center;
  word-spacing: 50px;
}
```

Palabras separadas

```
p {
  text-align: center;
  word-spacing: 50px;
}
span {
  background-color: lightblue;
}
```

Palabras separadas

## Decoración: text-decoration

La propiedad **text-decoration** permite añadir a un bloque de texto distintos tipos de rayado.

Los posibles valores de **text-decoration** son **none** (ninguno), **underline** (subrayado), **overline** (sobrerrayado) y **line-through** (tachado).

```
p {
  text-decoration: none;
}
```

Esto es un párrafo <p> normal y corriente.

```
p {
  text-decoration: underline;
}
```

Esto es un párrafo <p> subrayado.

```
p {
  text-decoration: overline;
}
```

Esto es un párrafo <p> sobrerrayado.

```
p {
  text-decoration: line-through;
}
```

Esto es un párrafo <p> tachado.

**Nota:** Los navegadores no muestran el valor **overline** de la misma manera.

El color de la línea es el mismo que el del texto.

```
p {
  color: red;
  text-decoration: line-through;
}
```

Esto es un párrafo <p> subrayado.

## Mayúsculas / minúsculas: **text-transform**

La propiedad **text-transform** permite cambiar el texto a minúsculas o mayúsculas. Los posibles valores de **text-transform** son **none** (ninguno), **capitalize** (la primera letra de cada palabra en mayúsculas), **uppercase** (todas las letras en mayúsculas) y **lowercase** (todas las letras en minúsculas).

```
p {
    text-transform: capitalize;
}
```

En Este Párrafo Todas Las Palabras Empiezan Por Mayúscula.

```
p {
    text-transform: uppercase;
}
```

EN ESTE PÁRRAFO TODAS LAS LETRAS SON MAYÚSCULAS.

```
p {
    text-transform: lowercase;
}
```

en este párrafo todas las letras son minúsculas.

El valor **capitalize** únicamente transforma la primera letra de cada palabra, el resto de letras se mantienen como estén (minúsculas o mayúsculas).

```
p {
    text-transform: capitalize;
}
```

En Este PÁRRAFO Una Palabra Estaba En Mayúsculas.

## Sombreado: **text-shadow**

La propiedad **text-shadow** permite añadir una o varias sombras al texto. Para definir cada sombra hay que indicar el color de la sombra (si se omite, se utiliza el color del texto), la distancia horizontal (positiva o negativa), la distancia vertical (positiva o negativa) y el diámetro de desenfoque.

Esta propiedad estaba incluida en la recomendación [CSS 2](#) y no está incluida en la recomendación [CSS 2.1](#), pero está incluida en la futura recomendación [CSS 3: Text decoration](#).

Si se escriben dos distancias, la primera distancia es la distancia horizontal y la segunda la distancia vertical de la sombra con respecto al texto:

```
p {
    font-size: 200%;
    font-weight: bold;
    letter-spacing: 10px;
    text-align: center;
    text-shadow: gray 6px 0;
    text-transform: uppercase;
}
```

**TEXTO SOMBREADO**

```
p {
    font-size: 200%;
    font-weight: bold;
    letter-spacing: 10px;
    text-align: center;
    text-shadow: gray -6px 0;
    text-transform: uppercase;
}
```

**TEXTO SOMBREADO**

```
p {
    font-size: 200%;
    font-weight: bold;
    letter-spacing: 10px;
    text-align: center;
    text-shadow: gray 0 6px;
    text-transform: uppercase;
}
```

**TEXTO SOMBREADO**

```
p {  
    font-size: 200%;  
    font-weight: bold;  
    letter-spacing: 10px;  
    text-align: center;  
    text-shadow: gray 0 -6px;  
    text-transform: uppercase;  
}
```

TEXTO SOMBREADO

```
p {  
    font-size: 200%;  
    font-weight: bold;  
    letter-spacing: 10px;  
    text-align: center;  
    text-shadow: gray 6px 0px;  
    text-transform: uppercase;  
}
```

TEXTO SOMBREADO

Si se escriben tres distancias, la tercera distancia indica el tamaño del difuminado de la sombra.

```
p {  
    font-size: 200%;  
    font-weight: bold;  
    letter-spacing: 10px;  
    text-align: center;  
    text-shadow: black 6px 6px 2px;  
    text-transform: uppercase;  
}
```

TEXTO SOMBREADO

```
p {  
    font-size: 200%;  
    font-weight: bold;  
    letter-spacing: 10px;  
    text-align: center;  
    text-shadow: black 6px 6px 4px;  
    text-transform: uppercase;  
}
```

TEXTO SOMBREADO

```
p {  
    font-size: 200%;  
    font-weight: bold;  
    letter-spacing: 10px;  
    text-align: center;  
    text-shadow: black 6px 6px 6px;  
    text-transform: uppercase;  
}
```

TEXTO SOMBREADO

```
p {  
    font-size: 200%;  
    font-weight: bold;  
    letter-spacing: 10px;  
    text-align: center;  
    text-shadow: black 6px 6px 8px;  
    text-transform: uppercase;  
}
```

TEXTO SOMBREADO

Se pueden definir varias sombras simultáneamente, escribiéndolas separadas por comas (,).

```
p {  
    font-size: 200%;  
    font-weight: bold;  
    letter-spacing: 10px;  
    text-align: center;  
    text-shadow: gray 8px 8px, gray -8px -8px;  
    text-transform: uppercase;  
}
```

TEXTO SOMBREADO

### Espacios en blanco: white-space

La propiedad **white-space** permite establecer el comportamiento del navegador con respecto a los espacios en blanco repetidos o con los saltos de línea del código fuente. Los posibles valores de **white-space** son **normal**, **pre**, **nowrap**, **pre-wrap** y **pre-line**.

El comportamiento habitual de los navegadores (salvo en el caso de la etiqueta **<pre>**, en la que hace todo lo contrario) es:

- colapsar los espacios en blanco, es decir, mostrar un único espacio en blanco cuando en el código fuente hay varios espacios en blanco seguidos
- colapsar los saltos de línea, es decir, no hacer caso de los saltos de línea que haya en el código fuente
- hacer ajuste de líneas (*wrap*), es decir, pasar el texto a la línea siguiente si el texto no cabe en una sola línea en el espacio disponible.

La tabla siguiente resume el comportamiento de cada uno de los valores de `white-space`

	<code>normal</code>	<code>pre</code>	<code>nowrap</code>	<code>pre-wrap</code>	<code>pre-line</code>
colapsar espacios en blanco	✓	X	✓	X	✓
colapsar saltos de línea	✓	X	✓	X	X
hacer ajuste de línea	✓	X	X	✓	✓

Los ejemplos siguientes muestran la diferencia entre los cinco valores. Los ejemplos se aplican al siguiente elemento `<pre>`:

```
<pre>Primera linea.  
Segunda linea, seguida de una linea en blanco.  
  
Cuarto linea, con cinco espacios en blanco entre parentesis: (     ).  
Quinta linea. Esta linea es muy larga para ver si el navegador mantiene todo en la misma linea o si lo muestra en varias lineas, de acuerdo con el espacio disponible.  


```

- El valor `normal` hace que el navegador colapse los espacios en blanco, colapse los saltos de línea y haga ajuste de línea (nota: para ver que el navegador hace ajuste de línea, puede hacer falta estrechar la ventana del navegador):

```
pre {  
    white-space: normal;  
}
```

```
Primera linea. Segunda linea, seguida de una linea en blanco. Cuarta linea, con cinco espacios en blanco entre parentesis: (     ). Quinta linea. Esta linea es muy larga para ver si el navegador mantiene todo en la misma linea o si lo muestra en varias lineas, de acuerdo con el espacio disponible.
```

- El valor `pre` hace que el navegador no colapse los espacios en blanco, no colapse los saltos de línea y no haga ajuste de línea.

```
pre {  
    white-space: pre;  
}
```

```
Primera linea.  
Segunda linea, seguida de una linea en blanco.  
  
Cuarto linea, con cinco espacios en blanco entre parentesis: (     ).  
Quinta linea. Esta linea es muy larga para ver si el navegador mantiene todo en la mi
```

- El valor `nowrap` hace que el navegador colapse los espacios en blanco, colapse los saltos de línea y no haga ajuste de línea:

```
pre {  
    white-space: nowrap;  
}
```

```
Primera linea. Segunda linea, seguida de una linea en blanco. Cuarta linea, con cinco
```

- El valor `pre-wrap` hace que el navegador no colapse los espacios en blanco, no colapse los saltos de línea y haga ajuste de línea (nota: para ver que el navegador hace ajuste de línea, puede hacer falta estrechar la ventana del navegador):

```
pre {
  white-space: pre-wrap;
}
```

Primera linea.  
Segunda linea, seguida de una linea en blanco.  
Cuarta linea, con cinco espacios en blanco entre paréntesis: (   ).  
Quinta linea. Esta linea es muy larga para ver si el navegador mantiene todo en la misma linea o si lo muestra en varias lineas, de acuerdo con el espacio disponible.

- El valor **pre-line** hace que el navegador colapse los espacios en blanco, no colapse los saltos de linea y haga ajuste de linea (nota: para ver que el navegador hace ajuste de linea, puede hacer falta estrechar la ventana del navegador):

```
pre {
  white-space: pre-line;
}
```

Primera linea.  
Segunda linea, seguida de una linea en blanco.  
Cuarta linea, con cinco espacios en blanco entre paréntesis: ( ).  
Quinta linea. Esta linea es muy larga para ver si el navegador mantiene todo en la misma linea o si lo muestra en varias lineas, de acuerdo con el espacio disponible.

## Dirección: **direction**

La propiedad **direction** permite establecer la dirección de escritura. Los posibles valores de **direction** son **ltr** (de izquierda a derecha) y **rtl** (de derecha a izquierda).

```
p {
  direction: ltr;
}
```

Esto es un párrafo con dirección de izquierda a derecha.

```
p {
  direction: rtl;
}
```

Esto es un párrafo con dirección de derecha a izquierda

Para que la propiedad **direction** se imponga al algoritmo de dirección de escritura del navegador, es necesario utilizar también la propiedad **unicode-bidi**.

```
p {
  direction: rtl;
  unicode-bidi: bidi-override;
}
```

.adreiuzzi a alcered ed nōiccerid noc ofarráp nu se otsE

Cuando se aplica a una tabla, la propiedad **direction** establece el orden de las columnas de la tabla.

```
<body>
  <table>
    <caption></caption>
    <tbody>
      <tr>
        <td>Columna 1 - Fila 1</td>
        <td>Columna 2 - Fila 1</td>
      </tr>
      <tr>
        <td>Columna 1 - Fila 2</td>
        <td>Columna 2 - Fila 2</td>
      </tr>
    </tbody>
  </table>
</body>
```

```
table {
  border: black 1px solid;
  direction: ltr;
}
td {
  border: black 1px solid;
}
```

Columna 1 - Fila 1	Columna 2 - Fila 1
Columna 1 - Fila 2	Columna 2 - Fila 2

```
table {
  border: black 1px solid;
  direction: rtl;
}
td {
  border: black 1px solid;
}
```

Columna 2 - Fila 1	Columna 1 - Fila 1
Columna 2 - Fila 2	Columna 1 - Fila 2

### Dirección: unicode-bidi

La propiedad **unicode-bidi** permite establecer el algoritmo de establecimiento de direccionalidad. Los posibles valores de **unicode-bidi** son **normal** (no modifica el algoritmo utilizado por el navegador), **embed** (para elementos en-línea, la dirección se puede redefinir en ellos) y **bidi-override** (para elementos en-línea y bloques, la dirección se puede redefinir en cualquier elemento y sub-elemento del bloque).

**Nota:** No está muy clara la diferencia entre **embed** y **bidi-override**.

```
p {
  unicode-bidi: bidi-override;
  direction: rtl;
}

strong {
  direction: ltr;
}
```

a adreiuqzi ed nōiccerid noc arbalap anu noc adreiuqzi a ahcered ed **ofarráp** nu se otsE .ahcered

```
p {
  unicode-bidi: bidi-override;
  direction: rtl;
}

strong {
  unicode-bidi: bidi-override;
  direction: ltr;
}
```

a adreiuqzi ed nōiccerid noc arbalap anu noc adreiuqzi a ahcered ed **párrafo** nu se otsE .ahcered

```
p {
  unicode-bidi: bidi-override;
  direction: rtl;
}

strong {
  unicode-bidi: embed;
  direction: ltr;
}
```

a adreiuqzi ed nōiccerid noc arbalap anu noc adreiuqzi a ahcered ed **párrafo** nu se otsE .ahcered

```
p {
  unicode-bidi:embed;
  direction: rtl;
}

strong {
  direction: ltr;
}
```

Esto es un **párrafo** de derecha a izquierda con una palabra con dirección de izquierda a derecha

```
p {
  unicode-bidi:embed;
  direction: rtl;
}

strong {
  unicode-bidi: bidi-override;
  direction: ltr;
}
```

Esto es un **párrafo** de derecha a izquierda con una palabra con dirección de izquierda a derecha

```
p {
  unicode-bidi:embed;
  direction: rtl;
}

strong {
  unicode-bidi: embed;
  direction: ltr;
}
```

Esto es un **párrafo** de derecha a izquierda con una palabra con dirección de izquierda a derecha

## 1.16. Líneas horizontales

Se aplican a una línea horizontal **<hr>** de una página web:

## Tamaño: width y height

La propiedad **height** establece la altura de la línea.

```
hr {
  height: 10px;
  background-color: black;
}
```



La propiedad **width** establece la anchura de la línea. Se suele expresar como porcentaje.

```
hr {
  height: 10px;
  width: 50%;
  background-color: black;
}
```



## Color: color, background-color y border-color

La propiedad **background-color** establece el color de la línea, aunque en Firefox la línea debe tener más de 2px de altura para ser visible y en Chrome basta con tener establecida la propiedad **height**.

```
hr {
  background-color: red;
}
```



```
hr {
  height: 1px;
  background-color: red;
}
```



```
hr {
  height: 2px;
  background-color: red;
}
```



```
hr {
  height: 3px;
  background-color: red;
}
```



```
hr {
  height: 10px;
  background-color: red;
}
```



En Firefox la propiedad **color** establece el color del borde de la línea, pero en Chrome no lo hace.

```
hr {
  color: red;
}
```



```
hr {
  height: 1px;
  color: red;
}
```



```
hr {
  height: 10px;
  color: red;
}
```



La propiedad **border-color** establece el color del borde de la línea.

```
hr {
  border-color: red;
}
```



```
hr {
  height: 1px;
  border-color: red;
}
```



```
hr {
  height: 10px;
  border-color: red;
}
```



## Alineación: margin-left, margin-right y width

Si no se indican los márgenes, las líneas se alinean al centro.

```
hr {  
    height: 10px;  
    width: 50%;  
    background-color: red;  
}
```



Para situar la línea en una posición determinada, se deben utilizar las propiedades **margin-left** y **margin-right**, sin utilizar la propiedad **width**. Los valores de estas propiedades se pueden expresar como distancias o como porcentajes.

```
hr {  
    height: 10px;  
    margin-left: 0%;  
    margin-right: 50%;  
    background-color: red;  
}
```



```
hr {  
    height: 10px;  
    margin-left: 50%;  
    margin-right: 0%;  
    background-color: red;  
}
```



```
hr {  
    height: 10px;  
    margin-left: 50%;  
    margin-right: 25%;  
    background-color: red;  
}
```



Cuando se aplican las propiedades **margin-left**, **margin-right** y **width** a una línea, la línea se dibuja a la izquierda del espacio definido por los márgenes si hay **margin-left** o a la derecha si sólo hay **margin-right**.

En el ejemplo siguiente, la línea se sitúa a la izquierda del todo:

```
hr {  
    background-color: red;  
    height: 10px;  
    margin-left: 0%;  
    width: 50%;  
}
```



En el ejemplo siguiente, la línea se sitúa a la derecha del todo:

```
hr {  
    background-color: red;  
    height: 10px;  
    margin-right: 0%;  
    width: 50%;  
}
```



En el ejemplo siguiente, la línea se sitúa a partir del centro:

```
hr {  
    background-color: red;  
    height: 10px;  
    margin-left: 50%;  
    width: 25%;  
}
```



## 1.17. Tablas

**Nota:** En las hojas de estilo de los ejemplos que se muestran en esta página se ha añadido bordes a las celdas y a las tablas mediante las siguientes reglas:

```
table {
  border: #b2b2b2 1px solid;
}
td {
  border: black 1px solid;
}
```

### Posición de la leyenda: **caption-side**

La propiedad **caption-side** permite elegir la posición de la leyenda (**<caption>**) con respecto a la tabla. Los dos valores permitidos son **top** (arriba) y **bottom** (abajo). Si no se establece la posición de la leyenda, los navegadores aplican el valor **top**.

```
caption {
```

Esto es la leyenda

Celda 1	Celda 2
Celda 3	Celda 4

```
caption {
  caption-side: top;
```

Esto es la leyenda

Celda 1	Celda 2
Celda 3	Celda 4

```
caption {
  caption-side: bottom;
```

Celda 1 Celda 2  
Celda 3 Celda 4

Esto es la leyenda

En la recomendación CSS 2 la propiedad **caption-side** podía tomar también los valores **right** (derecha) y **left** (izquierda), pero en la recomendación [CSS 2.1](#) se eliminaron esos valores, que tampoco aparecen en la futura recomendación [CSS 3: Tablas](#), actualmente (octubre de 2019) en elaboración.

### Centrar una tabla

Para centrar una tabla horizontalmente en la página, se deben establecer las dos propiedades **margin-left** y **margin-right** con el valor **auto**.

```
table {
  margin-left: auto;
  margin-right: auto;
}
```

Esto es la leyenda

Celda 1, con más texto	Celda 2
Celda 3	Celda 4, con más texto

La alineación del contenido de la leyenda se interpreta dentro del espacio que ocupa la leyenda (que coincide con el ancho de la tabla).

```
table {
  margin-left: auto;
  margin-right: auto;
}

caption {
  background-color: lightblue;
  text-align: right;
}
```

Esto es la leyenda

Celda 1, con más texto	Celda 2
Celda 3	Celda 4, con más texto

## Disposición de la tabla: **table-layout**

La propiedad **table-layout** permite elegir el modo en el que el navegador calcula el tamaño total de la tabla y el de cada fila o columna. Los dos valores permitidos son **fixed** (fijo) y **auto** (automático).

En el modo fijo (**fixed**), el tamaño de las tablas y de las celdas depende de las propiedades **width** y **height**.

En el modo fijo (**auto**), el tamaño de las tablas y de las celdas depende de su contenido (y si es posible, de las propiedades **width** y **height**).

## Modos de bordes: **border-collapse**

La propiedad **border-collapse** permite elegir el modo de presentación de los bordes de las celdas y de la tabla. Los dos valores permitidos son **separate** (separado) y **collapse** (colapsado). En el modo separado existe un hueco entre los bordes de cada celda, mientras que en el modo colapsado los bordes se superponen. Si no se establece el modo de presentación, los navegadores aplican el valor **separate**.

```
table {  
}
```

Esto es la leyenda  

Celda 1	Celda 2
Celda 3	Celda 4

```
table {  
    border-collapse: separate;  
}
```

Esto es la leyenda  

Celda 1	Celda 2
Celda 3	Celda 4

```
table {  
    border-collapse: collapse;  
}
```

Esto es la leyenda  

Celda 1	Celda 2
Celda 3	Celda 4

## Separación entre celdas: **border-spacing**

La propiedad **border-spacing** permite establecer una separación entre las celdas cuando se utiliza el modo de bordes separado. Se puede escribir un único valor, que se aplicaría a los cuatro lados, o dos valores, que se aplicarían en horizontal (a derecha e izquierda) y en vertical (arriba y abajo), respectivamente. Los valores no se pueden expresar en porcentajes, pero sí en cualquier otra unidad (**px**, **em**, etc).

```
table {  
    border-spacing: 20px;  
}
```

Esto es la leyenda  

Celda 1	Celda 2
Celda 3	Celda 4

```
table {  
    border-spacing: 30px 10px;  
}
```

Esto es la leyenda  

Celda 1	Celda 2
Celda 3	Celda 4

[Más información sobre definición de propiedades sobre bordes, márgenes y fondos en tablas](#)

---

*REFERENCIAS:*

---

Los contenidos y ejemplos incluidos en esta unidad se basan principalmente en:

- Curso Páginas web HTML y hojas de estilo CSS por Bartolomé Sintes Marco que se distribuye bajo una Licencia Creative Commons Reconocimiento-CompartirIgual 4.0 Internacional (CC BY-SA 4.0).  
<https://www.mclibre.org/consultar/htmlcss/index.html>
- [https://oscarmaestre.github.io/lenguajes\\_marcas/](https://oscarmaestre.github.io/lenguajes_marcas/)
- <http://jorgesanchez.net/lmsgi>
- <https://uniwebsidad.com/>
- <http://desarrolloweb.dlsi.ua.es/>