

UT2.-CSS

LMSGI 22-23
DAM - IES Alfonso X

Índice

1. Las limitaciones de HTML
2. Versiones de CSS
3. Sintaxis básica de CSS
4. Inserción de código CSS
5. Selectores
6. Formas de indicar valores de propiedades CSS
7. Formato de fuente
8. Formato de texto
9. Fondo

10. Listas
11. Configuración de bordes y espacios alrededor de los elementos.
12. Tablas
13. Posicionamiento
14. Propiedad display
15. Flexbox - display:flex

1. Las limitaciones de HTML

1.1 Introducción

- HTML es muy limitado para dar formato a las páginas web.
- Razones para no usar HTML para dar formato:
 - Con HTML no es posible reutilizar el formato de una forma fácil (si lo es con CSS).
 - Es preferible usar HTML únicamente como lenguaje semántico (dejando todo el formato en las páginas de estilos CSS).
 - Si sólo se utilizara HTML habría infinidad de etiquetas ya que la necesidad de estilizar obligaría a HTML a crecer de forma descontrolada.
 - Conviene que el formato y el contenido sean independientes.

1. Las limitaciones de HTML

1.2 La ayuda de CSS

- CSS es la abreviatura de Cascade Style Sheets (Hojas de Estilo en Cascada).
- Es un lenguaje de texto que se incrusta en las páginas web para modificar su formato.
- Es capaz de actuar sobre todas las etiquetas de un mismo tipo o sobre unas concretas, de forma que se pueda controlar desde las páginas de estilo qué estilo tendrán todos y cada uno de los componentes de la páginas del sitio web. Permite reutilizar estilos muy fácilmente.
- Facilita la homogeneidad y el mantenimiento de las páginas.

2. Versiones de CSS

CSS se idea a mediados de los años 90 y se ha ido estandarizando desde entonces:

- **CSS1.** Versión original estandarizada en 1996 por la W3C. Incluye formatos de texto, párrafo, márgenes, listas, tamaños de imágenes, ...
- **CSS2.** Estándar desde 1998. Amplía la versión anterior incluyendo posicionamiento (manejo de capas, tipos de medios (pantallas, impresoras, ...)).
- **CSS3.** Se ha ido desarrollando desde 1998 y es el estándar actual. Evoluciona CSS2 incluyendo nuevos módulos, como por ejemplo, para el manejo de contenido, sombreados, rellenos avanzados, transparencias, transiciones, nuevos selectores.

3. Sintaxis básica de CSS

- CSS es un lenguaje muy diferente sintácticamente a HTML y muy sencillo.
- Su dificultad radica en que es muy extenso.
- CSS sigue esta sintaxis:

```
selector {  
    propiedad1:valor1;  
    propiedad2:valor2;  
    ...  
}
```

- El selector es la parte de CSS que nos permite indicar a qué elemento (o elementos) de la página web se va a aplicar el formato CSS.
- Para indicar dicho formato se añaden tantas propiedades como se requiera al elemento.

3. Sintaxis básica de CSS

- Ejemplo:

```
p{  
  font-size:14pt;  
  color:red;  
}
```

- Este código CSS provocaría que los elementos tipo <p> salgan con tamaño de letra de 14 puntos y color rojo.

3.1 Comentarios

Dentro del código CSS se pueden colocar comentarios. Para ello el texto del comentario se encierra entre los símbolos /* y */. Ejemplo:

```
p{  
  line-height:10pt;  
  /*El siguiente código marcará el texto subrayado*/  
  text-decoration:underline;  
  text-align:center;  
}
```

4. Inserción de código CSS

4.1 En una etiqueta concreta de HTML

Todos los elementos de HTML (**p**, **strong**, **abbr**, **h1**,...) pueden utilizar un atributo llamado **style**, dentro de este atributo podemos añadir código CSS. Ejemplo:

```
<!doctype html>
<html lang="es">
  <head>
    <meta charset="UTF-8">
    <title></title>
  </head>
  <body>
    <p>Este texto sale de color negro (normal)</p>
    <p style="color:red;">Este texto sale de color rojo </p>
  </body>
</html>
```

En este caso el código CSS no lleva selector alguno, porque el estilo se aplica al elemento en el que se incrustó el código CSS. El resto del documento no queda afectado por el código CSS.

4.2 En el elemento de cabecera style

- En este caso el código CSS se inserta debajo del elemento HTML **style** que se colocará en la zona de cabecera (**head**). Este código afectará a toda la página web. Ejemplo:
- Ahora todos los elementos de **tipo p** se mostrarán en color rojo ya que es ese selector sobre el que se ha indicado el formato.
- El elemento **style** tendrá el atributo **type** que siempre contendrá el valor **text/css**.
- Atributo **media** (all, screen, print, tty, ...)

```
<!doctype html>
<html lang="es">
  <head>
    <meta charset="UTF-8">
    <title></title>
```

```
    <style type="text/css">
      p{
        color:red;
      }
    </style>
```

```
  </head>
  <body>
    <p>Este texto sale de color rojo</p>
    <p>Este texto sale de color rojo también </p>
  </body>
</html>
```

4.3 En un archivo externo

- Es el caso más habitual. Se trata de crear un archivo de texto con extensión CSS donde se insertará el código CSS.
- El mismo archivo nos sirve para aplicar estilos a diferentes páginas.
- Para colocar el código CSS existente en una página web HTML, hay que usar el elemento link (dentro del elemento head).
- Por ejemplo, si creamos el archivo de texto **estilos.css** con este contenido:

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

- Lo enlazaremos a nuestra página web de la siguiente manera (en este ejemplo, archivos css y html en el mismo directorio:

```
<!doctype html>
```

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

```
<head>
```

```
<meta charset="UTF-8">
```

```
<title></title>
```

```
<link rel="stylesheet" href="estilo1.css" type="text/css">
```

```
</head>
```

```
<body>
```

4.3 En un archivo externo

La etiqueta **link** tiene los atributos:

- **href**. Con el que se indica la ruta de la hoja de estilos que se está incluyendo.
- **rel**. Que siempre contiene el texto **stylesheet** (hoja de estilos).
- **type**. Siempre “**text/css**”.

4.4 Orden de aplicación de estilos

Es posible que algunas páginas web utilicen varios estilos referidos al mismo elemento. Ejemplo:

Pregunta: ¿De qué color saldrá el texto Hola (del <p> de abajo en el ejemplo)?

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>Document</title>
    <style type="text/css" media="screen">
      p{
        color:blue;
      }
    </style>
  </head>
  <body>
    <p style="color:red;">Hola</p>
  </body>
</html>
```

4.4 Orden de aplicación de estilos

La respuesta es de color rojo, porque tiene prioridad la definición más restrictiva.

En orden de preferencia, se ejecuta:

- Primero los estilos del **navegador**.
- Después se aplican los estilos externos (archivos css incorporados por la etiqueta **link**).
- A continuación los estilos de la **etiqueta style**.
- Por último se aplican los estilos definidos internamente en el elemento (mediante **atributo style**).
- En caso de dos estilos referidos al mismo elemento y definidos en el mismo ámbito → preferencia último.

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>Document</title>
    <style type="text/css" media="screen">
      p{
        color:blue;
      }
    </style>
  </head>
  <body>
    <p style="color:red;">Hola</p>
  </body>
</html>
```

4.5 Herencias

- Hay que tener en cuenta que hay etiquetas que son padre de otras. Es decir etiquetas que contienen a otras.
- La etiqueta **p** es padre de la etiqueta **em** (**em** está dentro de **p**). Esto hace que **em** herede todo el estilo que posea **p** y además añada el suyo propio.
- En el ejemplo anterior, **los años veinte** tendrán color azul y tamaño 14.

```
<p>Arturo Herrero: <em>Los años veinte</em></p>
```

```
p{  
  color:blue;  
  font-size:12pt  
}  
em{  
  font-size:14pt;  
}
```

5. Selectores

5.1 Introducción

- Los estilos CSS se aplican hacia el elemento HTML que indiquemos.
- Este elemento puede ser una etiqueta HTML, pero también podemos hacer selecciones más elaboradas.
- Hacer buenas selecciones nos permitirá conseguir hojas de estilos muy sofisticadas y coherentes.

5.2 Selección de elementos HTML

- Aplicar un estilo a un elemento concreto.
- En el ejemplo, todos los elementos de tipo <p> saldrán de color azul.
- Podemos incluso aplicar el estilo a varias etiquetas a la vez (h1, h2 o h3 saldrán en color azul).

```
p{  
  color:blue;  
  font-size:12pt  
}
```

```
h1,h2,h3{  
  color:blue;  
}
```

5.3 Selección de clases

- Una de las formas que permite diferenciar elementos del mismo tipo (por ejemplo un párrafo de otro) son las clases.
- Una clase es un identificador que asignamos a una serie de propiedades y valores CSS.

```
selector.nombreclase{  
    propiedad1:valor1;  
    propiedad2:valor2;  
    ...  
}
```

- Por ejemplo:

Un párrafo de la “**clase1**” adoptará este estilo →

```
p.clase1{  
    color: #339999;  
    background-color: #D6D6D6;  
}
```

```
<p class="clase1">
```

Este texto sale con el formato indicado por la clase `clase1`

```
</p>
```

5.3 Selección de clases

- Definir la clase sin indicar a qué selector pertenece:

```
.clase1{  
  color: #339999;  
  background-color: #D6D6D6;  
}
```

- En este caso cualquier elemento HTML (no sólo los <p>, como en el ejemplo anterior) podrán utilizar esa clase.

```
<h1 class="clase1">  
  Título con estilo clase1  
</h1>
```


5.4 Selección de identificadores

- El nombre que indiquemos se referirá al atributo identificador de un elemento, el cual se marca con el conocido atributo **id de HTML** (presente en todas las etiquetas).
- Los identificadores no se pueden repetir en el mismo documento, por lo que el formato indicado sólo se aplicará a un elemento de la página.
- Ejemplo:

```
#parrafo1{  
  color: #339999;  
  background-color: #D6D6D6;  
}
```

```
<p id="parrafo1">  
  texto del párrafo coloreado  
</p>
```

5.5 Selectores de limitación

- Permite indicar que el estilo definido se aplica a una determinada etiqueta pero cuando sea hija de otra que especificada. Ejemplo →
- En el ejemplo, se aplica a los elementos de tipo **p** cuando están dentro de elementos de celda (**td**).

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

5.6 Selector universal

- Existe un selector que permite aplicar un estilo a todas las etiquetas. Es el asterisco. Ejemplo →
- En el ejemplo de abajo, colorea en rojo el texto de los párrafos cuando esta etiqueta se use en tablas (es decir, dentro de elementos table); no importará si entre table y p hay otras etiquetas (sean del tipo que sean).

```
*{  
    color:black;  
}
```

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

5.7 Selectores jerárquicos

- Podemos entender HTML como un documento formado de manera jerárquica, donde hay elementos que contienen otros elementos formando una estructura de árbol.

```
<!DOCTYPE html>
<html lang="es-ES">
  <head>
    <meta charset="UTF-8">
    <title></title>
  </head>
  <body>
    <h1>Monumentos de Palencia</h1>
    <p>Palencia dispone de numerosos monumentos interesantes. Debido a su
    importante actividad medieval posee una gran cantidad de edificios
    religiosos entre los que destaca la <strong>Catedral de Palencia</strong>
    así como la iglesia de San Miguel con su peculiar torre de defensa y el
    monumento más conocido de la ciudad: El Cristo del Otero</p>
    <p>A finales del siglo <em>XIX</em> y principios del <em>XX</em>
    aparecieron edificios suntuosos y civiles que han embellecido una buena
    parte de la ciudad, en especial la transitada Calle Mayor.</p>
    <h2>Edificios religiosos</h2>
    <ul>
      <li>Cristo del Otero</li>
      <li>Catedral Mayor</li>
      <li>Iglesia de San Miguel</li>
      <li>Iglesia de San Lázaro</li>
      <li>Convento de San Pablo</li>
      <li>Iglesia de la Compañía</li>
    </ul>
  </body>
</html>
```

Podemos entender que sus elementos forman este diagrama:

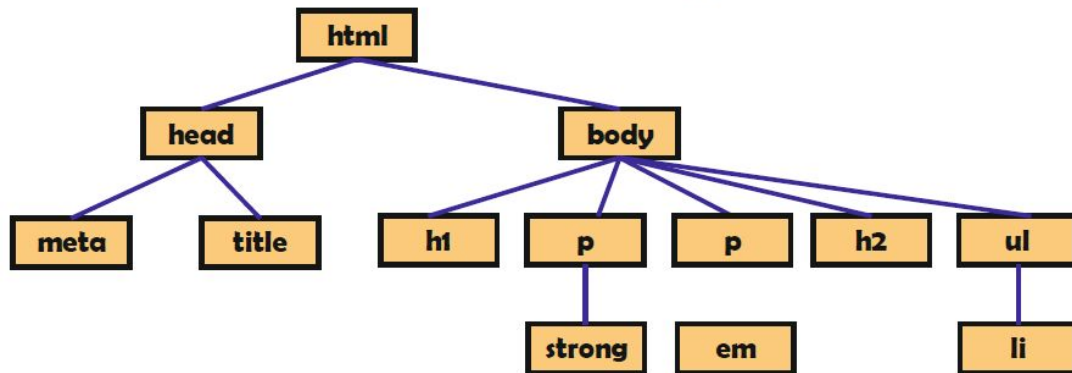


Ilustración 1, Estructura jerárquica de una página HTML

5.7 Selectores jerárquicos

- En él se observa como los elementos body y head son hijos de html. Mientras que meta y title son hijos de head (luego nietos de html). Los nodos al mismo nivel forman hermanos (h1, h2, p y ul en este esquema son hermanos).

Para especificar una relación padre/hijo, se puede emplear el signo > que indica una relación jerárquica de padre a hijo.

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

Podemos entender que sus elementos forman este diagrama:

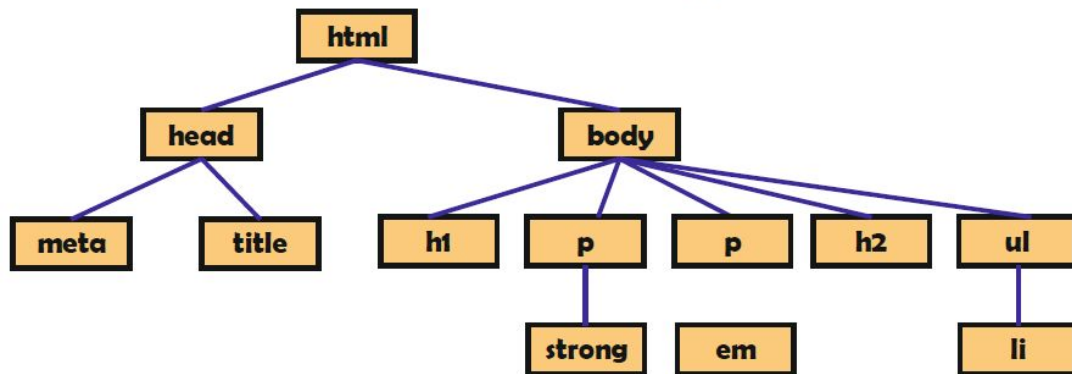


Ilustración 1, Estructura jerárquica de una página HTML

5.7 Selectores jerárquicos

- Otras expresiones posibles son:

| sintaxis | significado |
|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| elemento1 + elemento2 | El estilo se aplica al <i>elemento2</i> cuando es hermano del <i>elemento1</i> y además el <i>elemento1</i> precede inmediatamente al <i>elemento2</i> . |
| elemento1 ~ elemento2 | Se aplica al <i>elemento2</i> cuando es hermano del <i>elemento1</i> y éste le precede, aunque no sea inmediatamente. |

- Incluso se pueden hacer combinaciones. Por ejemplo:

```
ul > li + li {  
  color:green;  
}
```

Se aplica a los elementos li que estén dentro de elementos ul y además estén inmediatamente precedidos por otro elemento li.

5.8 Pseudoclasses

- Las pseudoclasses permiten asociar estilos a un selector cuando le ocurre una determinada circunstancia. Las pseudoclasses más famosas son las que se aplican a los enlaces (elemento a). Las cuales son:
 - **a:link**. Se aplica para los enlaces no visitados
 - **a:visited**. Enlaces visitados
 - **a:active**. Enlaces activos (aquellos sobre los que hacemos clic)
 - **a:hover**. Se aplica cuando el ratón pasa por encima del enlace
- Estas pseudoclasses permiten un cierto dinamismo en la página HTML. Actualmente además es posible utilizarlas en otros elementos distintos de la etiqueta a (como p, div, etc.) lo que da enormes posibilidades.

5.8 Pseudoclases

- Otras pseudoclases son:

| pseudoclase | significado |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------|
| :focus | Cuando el elemento obtiene el foco. Muy útil en formularios. |
| :lang(código) | Se aplica cuando el elemento esté marcado con el lenguaje indicado por su código (<i>es</i> para español, <i>en</i> para inglés,...) |
| :enabled | Cuando está habilitado (útil en formularios) |
| :disabled | Cuando está deshabilitado (útil en formularios) |
| :checked | Para controles de formulario de tipo radio o checkbox cuando estén activados |
| :before | Para indicar contenido anterior al párrafo. Siempre se suele usar con la propiedad content para añadir contenido al elemento. |
| :after | Para indicar contenido después del elemento. |
| :first-line | Aplica estilo a la primera línea del elemento. |
| :first-letter | Aplica el estilo a la primera letra del elemento. |

5.8 Pseudoclasas

Las pseudoclasas se pueden combinar, de modo que se pueden construir cosas como:

```
input:focus:hover {  
    background-color: yellow;  
}
```

Se pondrá color de fondo amarillo, cuando el cuadro de tipo input tenga el foco y el ratón pase por encima de él. O cosas más complejas como:

```
input.clase1:focus:hover[type="password"] {  
    background-color: yellow;  
}
```

En este caso se aplica el fondo amarillo cuando el cursor esté encima de un control de texto de tipo contraseña que además tenga el foco y sea de *clase1*.

6. Formas de indicar valores de propiedades CSS

Numerosas propiedades CSS tienen la necesidad de indicar **valores**. Por ejemplo el tamaño de la letra requiere un número. La cuestión es que CSS permite que dicho número se puede indicar con diferentes medidas (píxeles, pulgadas, centímetros, etc.)

6.1 Unidades de medida numéricas

- **cm.** Centímetro
- **pt.** Puntos. Suele utilizarse para tamaños de fuentes.
- **ex y em.** Tamaños relativos respecto a las letras x minúscula y M mayúscula respectivamente. Por ejemplo **2x** el doble del tamaño de altura de una x.
- **px.** Píxeles. Medida relativa respecto al dispositivo de salida (ya que en cada dispositivo el tamaño del píxel varía). Se suele usar en elementos grandes o contenedores como capas (divs) o tablas.
- **%.** Porcentaje. Tamaño relativo respecto al tamaño del elemento padre del elemento al que le ponemos la medida.
- **fr.** Fraction. Facilita reparto de espacio en filas o columnas
(<https://css-tricks.com/introduction-fr-css-unit/>)

https://www.w3schools.com/cssref/css_units.asp

6.2 Indicación de color

Hay numerosas etiquetas con capacidad de mostrar colores. Para indicar un color, CSS dispone de estas posibilidades:

- **Notación hexadecimal.** Cifra hexadecimal, precedida de #. Las dos primeras cifras indican nivel de rojo, las dos siguientes el nivel de verde y las dos últimas de azul.
- **Mediante función RGB.** Consiste en usar el formato **rgb(r,g,b)** donde **r** es el nivel de **rojo** (entre 0 y 255, o con %, es decir, entre 0% y 100%), **g** el nivel de **verde** y **b** el de **azul**.
- **Mediante función RGBA (con transparencia.** Igual que **rgb** pero con un cuarto parámetro **alpha** para indicar la transparencia. La función **rgba(r,g,b,a)** indica los valores **r,g,b** al igual que el anterior, pero la **a (alpha)** indica un valor entre **0 (transparencia total)** y **1 (opacidad total)**.
- **Por el nombre.** Permite indicar el color por su nombre estándar.
- **HSL:** Hue + Saturation + Lightness
(https://www.w3schools.com/colors/colors_hsl.asp)

https://www.w3schools.com/cssref/css_colors_legal.asp


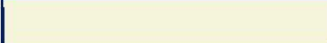
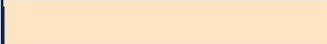







6.2 Indicación de color (ejemplos)

```
#p1 {background-color: rgb(255, 0, 0);} /* red */
#p2 {background-color: rgb(0, 255, 0);} /* green */
#p3 {background-color: rgb(0, 0, 255);} /* blue */
```

```
#p1 {background-color: #ff0000;} /* red */
#p2 {background-color: #00ff00;} /* green */
#p3 {background-color: #0000ff;} /* blue */
```

```
#p1 {background-color: rgba(255, 0, 0, 0.3);} /* red with opacity */
#p2 {background-color: rgba(0, 255, 0, 0.3);} /* green with opacity */
#p3 {background-color: rgba(0, 0, 255, 0.3);} /* blue with opacity */
```

```
#p1 {background-color: blue;}
#p2 {background-color: red;}
#p3 {background-color: coral;}
#p4 {background-color: brown;}
```

| Color | Name | Hexadecimal | Modo RGB | Modo HSL |
|-------------------------------------------------------------------------------------|-----------------|-------------|--------------------|-------------------|
|  | Azure | #F0FFFF | RGB(94%,100%,100%) | HSL(180,100%,97%) |
|  | Beige | #F5F5DC | RGB(96%,96%,86%) | HSL(60,56%,91%) |
|  | Bisque | #FFE4C4 | RGB(100%,89%,77%) | HSL(33,100%,88%) |
|  | Black | #000000 | RGB(0%,0%,0%) | HSL(0,0%,0%) |
|  | Blanched Almond | #FFEBCD | RGB(100%,92%,80%) | HSL(36,100%,90%) |
|  | Blue | #0000FF | RGB(0%,0%,100%) | HSL(240,100%,50%) |
|  | Blue Violet | #8A2BE2 | RGB(54%,17%,89%) | HSL(271,76%,53%) |
|  | Brown | #A52A2A | RGB(65%,16%,16%) | HSL(0,59%,41%) |
|  | Burlywood | #DEB887 | RGB(87%,72%,53%) | HSL(34,57%,70%) |
|  | Cadet Blue | #5F9EA0 | RGB(37%,62%,63%) | HSL(182,26%,50%) |

6.3 Indicación de URL

Muchas propiedades de CSS necesitan poder indicar direcciones URL a recursos necesarios para las páginas (direcciones de enlaces, imágenes, etc.). Para ello se utiliza una función llamada URL a la que (entre paréntesis) se le indica la dirección URL. Ejemplos (cambio de la imagen de fondo):

```
body{  
    background-image: url(imágenes/fondo1.jpg);  
}
```

```
body{  
    background-imagen:url(www.libreria.com/imagenes/img1.jpg);  
}
```

7. Formato de fuente. Propiedades

- **font-size:** Tamaño de la fuente en pantalla. Tres maneras:
 - **absoluto:** xx-small, x-small, small, medium, large, x-large, xx-large
 - **relativo** (al tamaño de letra del elemento padre): smaller, larger.
 - **exacto:** 12px, 2pt, 1.2em, ...
- **font-family:** Indica el tipo de letra (la fuente). No todas las fuentes están disponibles en todos los sistemas, por lo que suelen indicar varias opciones separadas por comas.
- **font-weight:** Peso de la fuente (grosor). Valores posibles:
 - **normal**
 - **bold** (negrita)
 - **número** (puede que no todos los navegadores soporten todos los pesos (valores que pueden ser 100, 200, .. 900)).
- **font-style:** Estilo de letra. Puede ser **normal**, **italic** (cursiva) u **oblique** (normalmente igual que la anterior).
- **font-variant:** Permite hacer uso de letra versalita (**small-caps**) o **normal**.
- **line-height:** Permite calibrar el interlineado (la distancia entre cada línea). Dos formas:
 - Un **número**. Múltiplo de la distancia normal. Por ejemplo 2, el interlineado será doble.
 - Una **distancia exacta con unidad de medida**. Por ejemplo, con 16px estamos indicando la distancia exacta entre cada línea (que será de 16 píxeles).

7. Formato de fuente. Propiedades

- **font:** Permite desde una sola propiedad cambiar en un solo golpe todas las

font: font-style font-variant font-weight font-size/line-height font-family;

El orden tiene que ser estrictamente ese, pero algunas propiedades se pueden dejar sin utilizar. Ejemplos: https://www.w3schools.com/css/tryit.asp?filename=trycss_font

```
/*Letra cursiva y negrita de tipo Comic con opciones a Arial y Helvetica*/
```

```
font: italic bold 16pt "Comic Sans MS", Arial, Helvetica, sans-serif;
```

```
/* Letra cursiva con versalitas de tamaño 18 puntos y 24 de puntos de distancia entre cada línea */
```

```
font: italic small-caps 18pt/24pt;
```

```
/* Letra cursiva */
```

```
font: italic;
```

- **color:** Color de la fuente mediante cualquier forma de indicar color comentados anteriormente.

7. Formato de fuente. Propiedades

- **@font-face**: CSS3 ha introducido el uso de esta directiva en el código CSS para precargar el tipo de letra necesario (por si la fuente no estuviera en el dispositivo de la persona que está viendo la página HTML). Para ello deberemos utilizar el archivo que contiene el tipo de letra en sí. Dicho archivo puede ser ttf (letras de tipo True Type) u otf (Open Type).

```
@font-face{  
    font-family:artistik;  
    src: url('Artistik.ttf'), url('Artistik.eot');  
}  
  
p {  
    font-family: artistik;  
}
```

En la parte @font-face hay que indicar al menos dos propiedades:

- **font-family**. Para dar nombre de familia a la letra que estamos importando. Ese nombre es el que luego utilizarán los selectores para elegir la letra que hemos cargado.
- **src**. En la que se indica una o más URL (separadas por comas) que apuntan a archivos de letras.

8. Formato del texto

A continuación vamos a ver propiedades que afectan al texto, fundamentalmente cambian el formato de los párrafos y aspectos del texto que no se refieren a su tipografía.

8.1 word-spacing (https://www.w3schools.com/cssref/tryit.asp?filename=trycss_text_word-spacing)

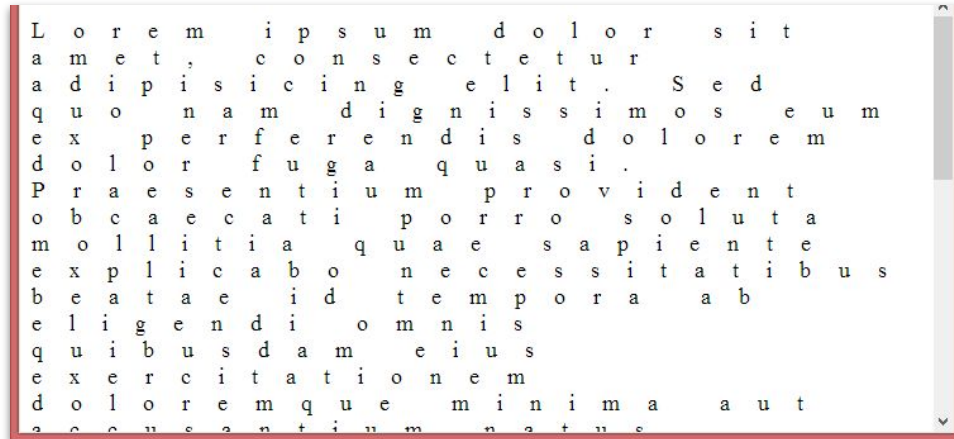
Indica la distancia entre las palabras del texto. Usa las mismas medidas que la propiedad font-size. Ejemplo de texto con word-spacing:20px;

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed quo
nam dignissimos eum ex perferendis dolorem dolor fuga quasi.
Praesentium provident obcaecati porro soluta mollitia quae sapiente
explicabo necessitatibus beatae id tempora ab eligendi omnis quibusdam
eius exercitationem doloremque minima aut accusantium natus temporibus
repellendus deleniti repellat dolorum possimus quasi ipsam magni nisi
voluptatibus voluptas illum numquam placeat inventore veritatis voluptate
est ratione. Sint veritatis suscipit maiores voluptatem minus at
quibusdam possimus voluptate rem reiciendis explicabo perspiciatis quo
tempora laudantium aspernatur illo molestias aliquid assumenda harum
similique enim officia vitae dolores ducimus ipsum culpa dicta corrupti
vel numquam sapiente repudiandae.

8.2 letter-spacing

Indica la distancia entre las letras del texto. Es similar a la anterior, pero ahora referida a la distancia horizontal entre caracteres. Ejemplo: **letter-spacing:20px;**

https://www.w3schools.com/cssref/tryit.asp?filename=trycss_letter-spacing



8.3 text-decoration

Se indican posibles efectos en el texto. Valores:

- **overline.** Línea por encima del texto.
- **line-through.** Tachado, línea que atraviesa el texto.
- **underline.** Subrayado (línea por debajo del texto)
- **none.** Quita todas las decoraciones (útil por ejemplo <a>)

This is heading 1

~~This is heading 2~~

This is heading 3

This is heading 4

[Link](#)

Permite combinar los anteriores, por ejemplo, **text-decoration: underline overline;**
https://www.w3schools.com/cssref/tryit.asp?filename=trycss_text-decoration

8.4 vertical-align

Posición vertical del texto (o imagen) respecto a su contenedor. Es muy versátil porque permite tanto alinear en vertical un texto respecto, por ejemplo, a la celda de la tabla en la que se encuentre; como indicar superíndices y subíndices. Posibilidades:

- **baseline.** En la línea base inferior del texto
- **sub.** Subíndice
- **super.** Superíndice
- **top.** Arriba respecto al elemento más alto de la línea
- **text-top.** En la línea superior del texto
- **middle.** Medio respecto a la altura del texto o contenedor en el que estemos
- **bottom.** Abajo respecto al elemento más alto de la línea
- **text-bottom.** En la línea inferior del texto
- **Porcentaje.** Porcentaje respecto al texto (por ejemplo 120%)

https://www.w3schools.com/cssref/tryit.asp?filename=trycss_vertical-align

8.5 text-align (https://www.w3schools.com/cssref/tryit.asp?filename=trycss_text-align)

Alineación horizontal del texto. Puede ser: **left** (izquierda), **right** (derecha), **center** (centrada) o **justify** (justificada a derecha e izquierda).

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Sed quo nam dignissimos eum ex perferendis dolorem dolor fuga quasi. Praesentium provident obcaecati porro soluta mollitia quae sapiente explicabo necessitatibus beatae id tempora ab eligendi omnis quibusdam eius exercitationem doloremque minima

Alineación izquierda: **text-align:left;**

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Sed quo nam dignissimos eum ex perferendis dolorem dolor fuga quasi. Praesentium provident obcaecati porro soluta mollitia quae sapiente explicabo necessitatibus beatae id tempora ab eligendi omnis quibusdam eius exercitationem doloremque minima

Alineación izquierda: **text-align:center;**

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Sed quo nam dignissimos eum ex perferendis dolorem dolor fuga quasi. Praesentium provident obcaecati porro soluta mollitia quae sapiente explicabo necessitatibus beatae id tempora ab eligendi omnis quibusdam eius exercitationem doloremque minima

Alineación izquierda: **text-align:right;**

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Sed quo nam dignissimos eum ex perferendis dolorem dolor fuga quasi. Praesentium provident obcaecati porro soluta mollitia quae sapiente explicabo necessitatibus beatae id tempora ab eligendi omnis quibusdam eius exercitationem doloremque minima

Alineación izquierda: **text-align:justify;**

8.6 text-indent (https://www.w3schools.com/cssref/tryit.asp?filename=trycss_text-indent)

Sangría de la primera línea del párrafo. Distancia extra que se deja a la primera línea respecto al resto de líneas. Por ejemplo si indicamos, **text-indent:50px;** el resultado sería (para un párrafo al que se aplique ese código):

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Sed quo nam dignissimos eum ex
perferendis dolore dolor fuga quasi. Praesentium provident obcaecati porro soluta mollitia quae
sapiente explicabo necessitatibus beatae id tempora ab eligendi omnis quibusdam eius exercitationem
doloremque minima aut accusantium natus temporibus repellendus deleniti repellat dolorum
possimus quasi ipsam magni nisi voluptatibus voluptas illum numquam placeat inventore veritatis
voluptate est ratione. Sint veritatis suscipit maiores voluptatem minus at quibusdam possimus
voluptate rem reiciendis explicabo perspiciatis quo tempora laudantium aspernatur illo molestias
aliquid assumenda harum similique enim officia vitae dolores ducimus ipsum culpa dicta corrupti vel
numquam sapiente repudiandae.

8.7 text-transform

Permite modificar el texto para que se muestre en mayúsculas o minúsculas.

- **capitalize.** La primera letra en Mayúsculas
- **uppercase.** Mayúsculas
- **lowercase.** Minúsculas
- **none.** No hace ninguna transformación

This Is Some Text.

THIS IS SOME TEXT.

this is some text.

8.8 direction

Procede de CSS2, especifica la dirección en la que se escribe el texto. Posibilidades:

- **ltr.** [Left to right](#), de izquierda a derecha.
- **rtl.** [Right to left](#), de derecha a izquierda.

8.9 text-overflow (https://www.w3schools.com/cssref/tryit.asp?filename=trycss3_text-overflow)

Parte de CSS3. Indica qué hacer con el texto cuando está dentro de un contenedor (como una capa) y no tiene el tamaño suficiente para mostrar todo el texto. Posibilidades:

- **clip.** El texto sale recortado. Sólo se ve el texto que cabe en la capa, el resto no se muestra
- **ellipsis.** Como la anterior, pero se ponen ... al final del texto recortado.

text-overflow: clip (default):

Hello w

text-overflow: ellipsis:

Hell...

8.10 text-shadow

Se trata de una propiedad CSS3. Permite colocar una sombra al texto para darle efecto de volumen. Tiene esta sintaxis:

```
text-shadow: h-shadow v-shadow blur-radius color
```

- **h-shadow.** Es el desplazamiento horizontal que tendrá la sombra (puede ser positivo o negativo)
- **v-shadow.** Es el desplazamiento vertical que tendrá la sombra (puede ser positivo o negativo)
- **blur-radius.** Es opcional e indica cuánto se va a desenfocar la sombra. Se indica una cantidad que cuanto mayor sea, más desenfocherà el fondo.
- **color.** Es el color de la sombra (opcional)

https://www.w3schools.com/cssref/tryit.asp?filename=trycss3_text-shadow

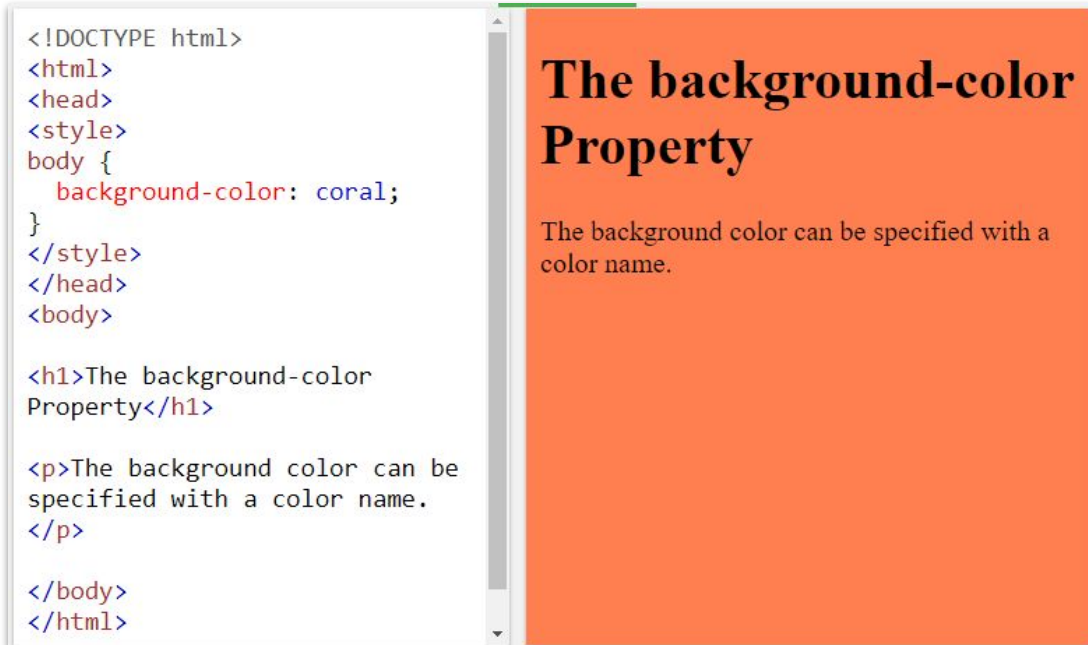
```
h1 {  
  text-shadow: 2px 2px 8px #FF0000;  
}
```

Text-shadow with blur effect

9. Fondo

9.1 Color de fondo. background-color

La propiedad **background-color** permite establecer un color de fondo al elemento al que se aplique la propiedad. Si se aplica al elemento **body**, toda la página tendrá ese color de fondo.



9.2 Imagen de fondo. background-image

La propiedad **background-image** permite establecer una imagen de fondo. Esta imagen se superpone al color de fondo, de modo que si la imagen no se puede cargar (porque la ruta a ella no se ha indicado), entonces aparece el color de fondo. La imagen de fondo se repite las veces necesarias (creando un mosaico), hasta que se rellena el elemento.

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
  background-image:
url("paper.gif");
  background-color: #cccccc;
}
</style>
</head>
<body>

<h1>The background-image
Property</h1>

<p>Hello World!</p>

</body>
</html>
```

The background-image Property

Hello World!

9.3 Repetición del fondo. background-repeat

En principio, el fondo se repite en todas las direcciones hasta rellenar el elemento al que se aplica la imagen de fondo. Posibles valores de la propiedad:

- **repeat.** Es el valor por defecto. la imagen se repite en todas las direcciones (efecto mosaico), hasta rellenar completamente el elemento.
- **repeat-x.** La repetición de la imagen se hace sólo en horizontal.
- **repeat-y.** La repetición de la imagen se hace sólo en vertical.
- **no-repeat.** La imagen no se repite aparecerá sólo una vez

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
  background-image:
  url("paper.gif");
  background-repeat: repeat-y;
}
</style>
</head>
<body>

<h1>The background-repeat
Property</h1>
<p>Here, the background image is
repeated only vertically.</p>

</body>
</html>
```

The background-repeat Property

Here, the background image is repeated only vertically.

9.4 Posición de la imagen. background-position

Por defecto la imagen se coloca desde la esquina superior izquierda de la página (posición 0,0) y desde ahí se repite (si se ha indicado repetición de la imagen con la propiedad anterior) o no. La posición desde la que la imagen se coloca inicialmente se puede modificar. La forma de hacerlo es así:

background-position: *posicionHorizontal posicionVertical*

- **posicionHorizontal.** Se puede especificar las palabras: left (izquierda), right (derecha) o middle (centro). También se puede indicar una coordenada horizontal concreta (por ejemplo 12px) o relativa (5%).
- **posicionVertical.** Se puede especificar las palabras: top (arriba), bottom (abajo) o center (centro). También se puede indicar una coordenada vertical concreta (por ejemplo 20px) o relativa (15%).

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
  background-image: url('w3css.gif');
  background-repeat: no-repeat;
  background-attachment: fixed;
  background-position: right bottom;
}
</style>
</head>
<body>

<h1>The background-position Property</h1>
<p>Here, the background image will be
positioned in the center of the element,
using percent.</p>

</body>
</html>
```

The background-position Property

Here, the background image will be positioned in the center of the element, using percent.



9.5 Desplazamiento del fondo. background-attachment

Cuando se desplaza el contenido de una página web, el fondo se mueve con el resto de la página. Mediante esta propiedad podremos hacer que el fondo quede fijo mientras que sólo el resto de elementos se mueven.

- **scroll.** Valor por defecto. El fondo y el texto se mueven juntos cuando el usuario se desplaza por la página
- **fixed.** El fondo es fijo y sólo se mueve el texto.

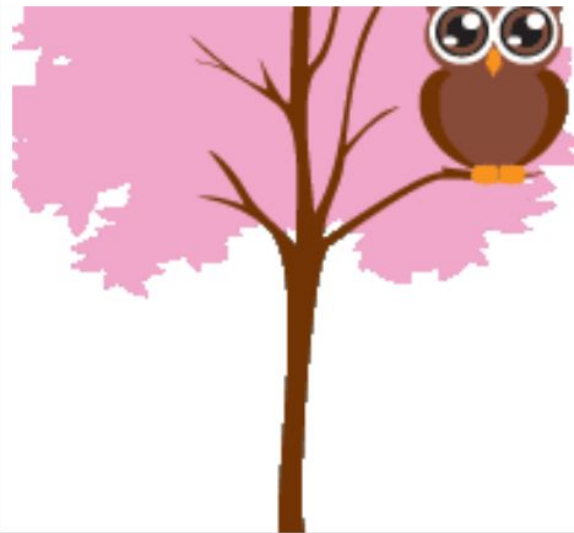
```
<head>
<style>
.fixed-bg {
  background-image: url("img_tree.gif");
  min-height: 500px;
  background-attachment: fixed;
  background-position: center;
  background-repeat: no-repeat;
  background-size: cover;
}
</style>
</head>
<body>

<p>In this example, we have created a fixed
background image that will disappear slowly
on scroll. Scroll the page to see the
effect. <strong>Note:</strong> Try to remove
the background-attachment property to really
understand this example.</p>

<div class="fixed-bg"></div>

<div style="height:800px;background-
color:yellow;">This div is only here to
enable scrolling (height = 800 pixels).
</div>
```

In this example, we have created a fixed background image that will disappear slowly on scroll. Scroll the page to see the effect. **Note:** Try to remove the background-attachment property to really understand this example.



9.6 Fondo (todas las propiedades juntas). background

Fija en una sola propiedad todas las propiedades de fondo. Sintaxis:

```
background: background-color background-image background-repeat  
background-attachment background-position
```

Ejemplo:

```
background: maroon url('fondo1.gif') no-repeat fixed left bottom;
```

Coloca color marrón de fondo, la imagen *fondo1.gif*, que sólo aparece una vez, en la parte inferior izquierda y que se queda fijo en el fondo cuando el usuario se desplaza por la pantalla.

https://www.w3schools.com/cssref/tryit.asp?filename=trycss_background

10. Listas

CSS permite varios formatos pensados para listas. En general para los elementos que contienen las etiquetas **ul** y **ol**.

10.1 list-style-type

Permite especificar el tipo de elemento de numeración de la lista. Estos son los más comunes:

- **circle**. La lista quedará encabezada por círculos.
- **decimal**. Números decimales.
- **decimal-leading-zero**. Números decimales empezando por cero.
- **disc**. La lista quedará encabezada por círculos sin rellenar.
- **lower-alpha**. Letras minúsculas del alfabeto actual.
- **lower-greek**. Letras griegas minúsculas.
- **lower-latin**. Letras minúsculas latinas.
- **lower-roman**. Números romanos en minúsculas.
- **none**. Sin números.
- **square**. Cuadrados rellenos.
- **upper-alpha**. Letras mayúsculas del alfabeto actual.
- **upper-latin**. Letras mayúsculas latinas.
- **upper-roman**. Números romanos en mayúsculas.

10.1 list-style-type

https://www.w3schools.com/cssref/tryit.asp?filename=trycss_list-style-type_ex

```
<!DOCTYPE html>
<html>
<head>
<style>
ul.a {list-style-type: circle;}
ul.b {list-style-type: square;}
ol.c {list-style-type: upper-roman;}
ol.d {list-style-type: lower-alpha;}
</style>
</head>
<body>

<h1>The list-style-type Property</h1>

<p>Example of unordered lists:</p>

<ul class="a">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Coca Cola</li>
</ul>

<ul class="b">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Coca Cola</li>
</ul>

<p>Example of ordered lists:</p>

<ol class="c">
  <li>Coffee</li>
  <li>Tea</li>
```

The list-style-type Property

Example of unordered lists:

- Coffee
- Tea
- Coca Cola

- Coffee
- Tea
- Coca Cola

Example of ordered lists:

- I. Coffee
- II. Tea
- III. Coca Cola

- a. Coffee
- b. Tea
- c. Coca Cola

10.2 list-style-image

En lugar de utilizar uno de los símbolos anteriores, se puede indicar una imagen con la que se rellenará la lista. La imagen puede tener cualquiera de los formatos habituales en las páginas web (gif, jpg, png). Lógicamente el tamaño debe de ser apropiado; si es muy grande la lista quedará totalmente descuadrada. Ejemplo:

list-style-image: url('cuadrado.gif');

```
<!DOCTYPE html>
<html>
<head>
<style>
ul {
  list-style-image: url('sqpurple.gif');
}
</style>
</head>
<body>

<h1>The list-style-image Property</h1>

<p>The list-style-image property replaces the list-
item marker with an image:</p>

<ul>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Coca Cola</li>
</ul>

</body>
</html>
```

The list-style-image Property

The list-style-image property replaces the list-item marker with an image:

- Coffee
- Tea
- Coca Cola

https://www.w3schools.com/cssref/tryit.asp?filename=trycss_list-style-image

10.3 list-style-position

Sólo tiene dos posibles valores referidos a la posición del texto respecto de la imagen.

- **inside.** El símbolo de numeración es interior a los márgenes del elemento en el que se coloca. Es la opción por defecto.
- **outside.** El símbolo de numeración es exterior.

```
<style>
ul.a {
  list-style-position: outside;
}

ul.b {
  list-style-position: inside;
}
</style>
</head>
<body>
```

```
<h1>The list-style-position Property</h1>
```

```
<h2>list-style-position: outside (default):</h2>
```

```
<ul class="a">
  <li>Coffee - A brewed drink prepared from
roasted coffee beans</li>
  <li>Tea - An aromatic beverage commonly prepared
by pouring hot or ...</li>
  <li>Coca Cola - A carbonated soft drink produced
by The Coca-Cola Company.</li>
</ul>
```

```
<h2>list-style-position: inside:</h2>
```

```
<ul class="b">
```

The list-style-position Property

list-style-position: outside (default):

- Coffee - A brewed drink prepared from roasted coffee beans
- Tea - An aromatic beverage commonly prepared by pouring hot or ...
- Coca Cola - A carbonated soft drink produced by The Coca-Cola Company.

list-style-position: inside:

- Coffee - A brewed drink prepared from roasted coffee beans
- Tea - An aromatic beverage commonly prepared by pouring hot ...
- Coca Cola - A carbonated soft drink produced by The Coca-Cola Company.

https://www.w3schools.com/cssref/tryit.asp?filename=trycss_list-style-position

10.4 opciones display referidas a listas

CSS dispone de una propiedad muy potente llamada **display**. Esta propiedad modifica el comportamiento de los elementos y permite indicar otras posibilidades. En el caso de las listas hay dos valores de display que son interesantes:

- **display:inline**. El valor **inline** aplicado a una lista hace que la lista no se muestre separada y acompañada de los símbolos de lista, sino que se muestra en la misma línea separados con un espacio cada elemento. Ejemplo:

Resultado:

Lista de la compra

Azucar Sal Alubias Refresco

```
<!doctype html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    li{
      display:inline;
    }
  </style>
</head>
<body>
  <ol>
    <li>Azúcar</li>
    <li>Sal</li>
    <li>Alubias</li>
    <li>Refresco</li>
  </ol>
</body>
</html>
```

10.4 opciones display referidas a listas

- **display:list-item.** Es justo el contrario del anterior permite a otros elementos usarse como parte de una lista. Ejemplo:

El resultado es:

Lista de la compra

1. Azucar
2. Sal
3. Alubias
4. Refresco

La etiqueta `p` se comporta en el ejemplo como si fuera una etiqueta `li`, sin embargo respetará el tipo de letra, los espacios,.. es decir todas las características restantes de los elementos `p`. Se observa también que cuando `p` no está dentro de una etiqueta de lista (como `ul` u `ol`) funciona de manera habitual (como se observa en el párrafo Lista de la compra).

```
<!doctype html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    p{
      display:list-item;
    }
  </style>
</head>
<body>
  <p>Lista de la compra</p>
  <ol>
    <p>Azúcar</p>
    <p>Sal</p>
    <p>Alubias</p>
    <p>Refresco</p>
  </ol>
</body>
</html>
```

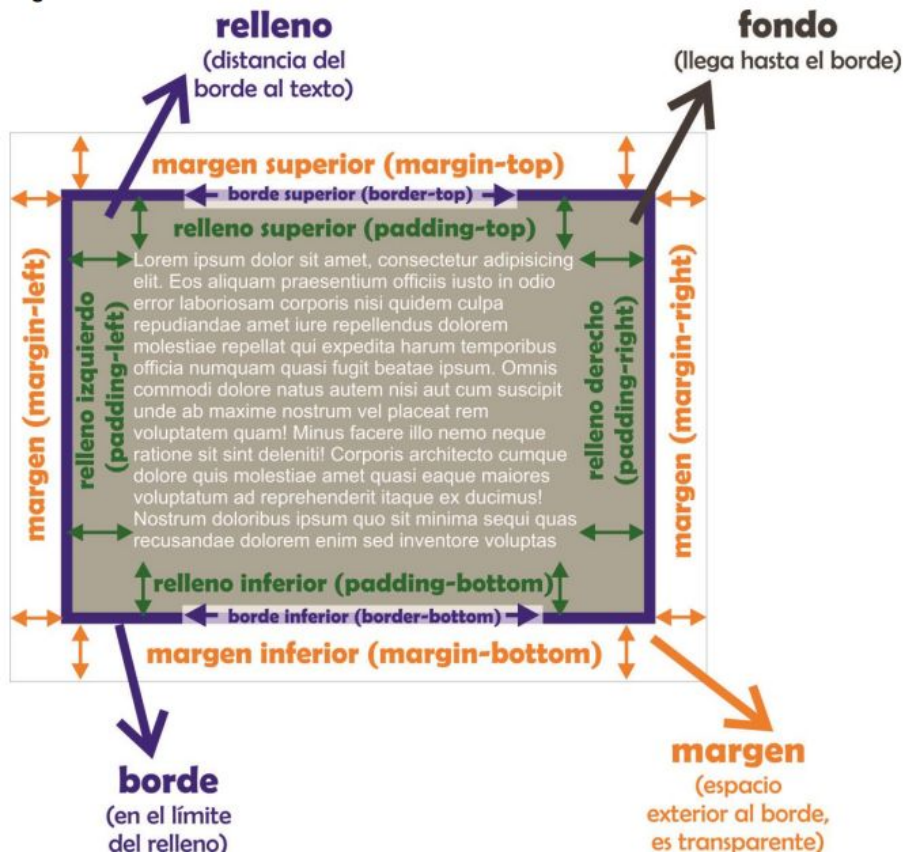
11. Bordos y espacios alrededor de elementos

11.1 Elementos del formato de caja

Parte del CSS que modela el rectángulo imaginario que envuelve a cualquier elemento HTML:

Box Model:

https://www.w3schools.com/css/css_boxmodel.



11.2 Configuración del borde

Configuración del borde (CSS1 y CSS2)

- Tres propiedades fundamentales: color, estilo y grosor.
- Es posible configurar independientemente cada borde (izquierdo, derecho, superior e inferior).
- Propiedades:
 - **border-width**: anchura del borde (ej: 12px, thin, medium, thick).
 - **border-style**: estilo del borde (ej: solid, dashed, dotted, double, groove, ridge, inset, outset, none).
 - **border-(top/left/bottom/right)-(width/color/style)**
 - **border**: permite de una vez ponerlo todo (ej: 1px solid red)

Bordes redondeados (CSS3)

- En CSS3 aparece la posibilidad de bordes redondeados, haciendo que las esquinas tomen forma circular con un radio que podemos especificar mediante:
 - **border-radius**: indica el radio del borde redondeado, si varios, uno para cada esquina.
 - **border-(bottom/top)-(left/right)-radius**

12. Tablas

Las propiedades CSS específicas para tablas son:

- **border-collapse:** permite indicar cómo manejar los bordes adyacentes entre elementos.

Posibles valores:

- **collapse:** se unen los bordes adyacentes para formar un único borde.
- **separate:** los bordes adyacentes se mantienen separados. Es el valor por defecto.

Ejemplo:

```
table{  
  width:100%;  
}  
td{  
  border:2px solid black;  
}
```

| | | | |
|--|--|--|--|
| | | | |
| | | | |
| | | | |

```
table{  
  width:100%;  
  border-collapse:collapse;  
}  
td{  
  border:2px solid black;  
}
```

Resultado:

| | | | |
|--|--|--|--|
| | | | |
| | | | |
| | | | |

12. Tablas

- **border-spacing**: indica el espacio entre bordes (dos medidas: distancia horizontal y vertical, si sólo una, se aplica a ambas).

```
table{  
  width:100%;  
  border-spacing: 10px 25px;  
  border:2px solid black;  
}  
td{  
  border:2px solid black;  
}
```

| | | | |
|--|--|--|--|
| | | | |
| | | | |
| | | | |

13. Posicionamiento

13.1 Introducción

- Necesidad imperiosa para maquetar → poder controlar la posición concreta en la que aparecen los componentes.
- CSS2 se centró en esta labor de maquetación, entre otras cosas, introduciendo **div**.
- Posteriormente HTML5 introduce las etiquetas semánticas (ej: header, nav, section, article, aside, footer, ...)

13.2 flotación

- La propiedad float es muy útil para contenidos multimedia como imágenes o tablas.
- Permite indicar cómo se comportan los elementos siguientes sobre aquel al que le establecemos la propiedad.
- Normalmente los elementos no flotan.

13.2 Flotación

- Es decir cuando colocamos, por ejemplo, una imagen, el contenido que va a continuación aparece en su línea base, dejando un hueco poco estético.

`<p>`In this example, the image will float to the right in the text, and the text in the paragraph will wrap around the image.`</p>`

```

<p>
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac. In at libero sed nunc venenatis imperdiet sed ornare turpis. Donec vitae dui eget tellus accumsan venenatis. Tamen

In this example, the image will float to the right in the text, and the text in the paragraph will wrap around the image.



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa

13.2 Flotación

- Podemos hacer que ese contenido (<p> en el ejemplo) contornee la imagen por su derecha (indicando float:left a la imagen) o por su izquierda (indicando float:right a la imagen).
- Posibles valores de la propiedad float:
 - **left.** El elemento flota a la izquierda. El contenido inmediatamente siguiente le contornea a su derecha.
 - **right.** El elemento flota a la derecha. El contenido inmediatamente siguiente le contornea a su izquierda.
 - **none.** El elemento no flota (valor por defecto).
- Por lo tanto, los elementos siguientes al elemento flotante fluirán alrededor de él. Para evitar esto, debes usar la propiedad **clear** (por ejemplo, clear:both, clear:left o clear:right).

https://www.w3schools.com/cssref/tryit.asp?filename=trycss_float

```
<!DOCTYPE html>
<html>
<head>
<style>
img {
  float: left;
}

</style>
</head>
<body>
<h1>The float Property</h1>

<p>In this example, the image will float to the right in the
text, and the text in the paragraph will wrap around the
image.</p>


  <p>Uno</p>
  <p>Dos</p>
  <p>Tres</p>
  <p>Cuatro</p>
  <p>Cinco</p>
  <p>Seis</p>
</body>
```

```
1  p {
2    border: 1px solid black;
3  }
```

Uno

Dos

Tres

Cuatro

Cinco

Seis

- Sin embargo display puede cambiar la forma de funcionar de un elemento.

https://www.w3schools.com/cssref/tryit.asp?filename=trycss_display

14. La propiedad display

- Por ejemplo, podemos hacer que los párrafos pasen a comportarse como elementos de línea, poniéndole **display: inline** (el párrafo es por defecto de bloque):

```
p {  
  border: 1px solid black;  
  display: inline;  
}
```

Uno Dos Tres Cuatro Cinco Seis

- Los valores más importantes de la propiedad **display** son:
 - **none**. El elemento es eliminado (no se muestra).
 - **inline**. Muestra en la misma línea (respetando el flujo) todos los elementos y no acepta las propiedades width, height ni márgenes verticales.
 - **block**. Muestra los elementos en líneas independientes y acepta las propiedades width, height y márgenes verticales.
 - **inline-block**. Su comportamiento es una mezcla entre los dos anteriores, se muestran en la misma línea (respetando el flujo) todos los elementos y además, acepta las propiedades width, height y márgenes verticales.

14. La propiedad display

```
<!DOCTYPE html>
<html>
<head>
<style>
p {color: red;}

p.ex1 {display: none;}
p.ex2 {display: inline;}
p.ex3 {display: block;}
p.ex4 {display: inline-block;}
</style>
</head>
<body>
<h1>The display Property</h1>

<h2>display: none:</h2>
<div>
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat
pulvinar, at pulvinar felis blandit. <p class="ex1">HELLO WORLD!</p> Vestibulum volutpat
tellus diam, consequat gravida libero rhoncus ut.
</div>

<h2>display: inline:</h2>
<div>
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat
pulvinar, at pulvinar felis blandit. <p class="ex2">HELLO WORLD!</p> Vestibulum volutpat
tellus diam, consequat gravida libero rhoncus ut.
</div>

<h2>display: block:</h2>
<div>
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat
pulvinar, at pulvinar felis blandit. <p class="ex3">HELLO WORLD!</p> Vestibulum volutpat
tellus diam, consequat gravida libero rhoncus ut.
</div>
```

The display Property

display: none:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat pulvinar, at pulvinar felis blandit. Vestibulum volutpat tellus diam, consequat gravida libero rhoncus ut.

display: inline:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat pulvinar, at pulvinar felis blandit. **HELLO WORLD!** Vestibulum volutpat tellus diam, consequat gravida libero rhoncus ut.

display: block:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat pulvinar, at pulvinar felis blandit.

HELLO WORLD!

Vestibulum volutpat tellus diam, consequat gravida libero rhoncus ut.

display: inline-block:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat pulvinar, at pulvinar felis blandit.

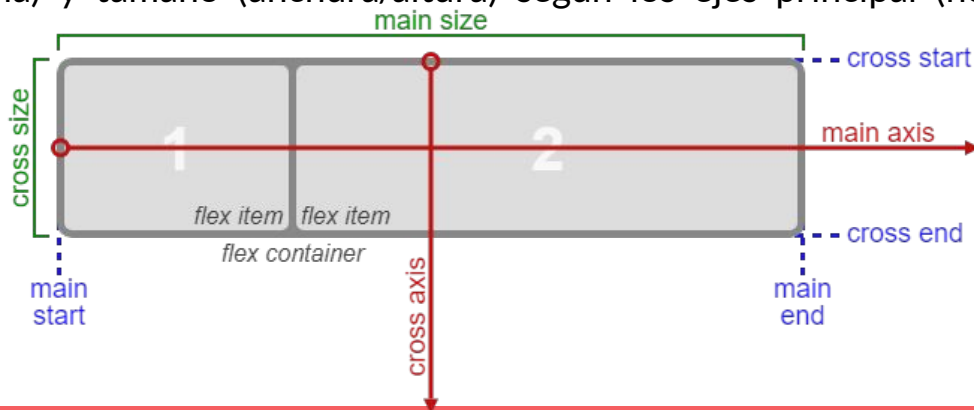
HELLO WORLD! Vestibulum volutpat tellus diam, consequat gravida libero rhoncus ut.

https://www.w3schools.com/cssref/tryit.asp?filename=trycss_display

15. Flexbox – display:flex

<https://www.yunbitsoftware.com/blog/2017/03/30/flexbox-css3-tutorial-descripcion-ejemplos-html/>

- Flexbox viene de “Flexible Box Layout”, que se puede traducir como “Diseño de caja flexible”.
- Aporta una magnífica solución para la maquetación “responsive”.
- Permite crear un conjunto de elementos flexibles que se adaptan automáticamente a su contenedor y con el que se puede controlar parámetros tales como la alineación, dirección (horizontal/vertical), ajuste de la fila según tamaños y multitud de posibilidades.
- Hasta ahora hemos utilizado las display, position y las propiedades para dimensionar y posicionar:
- El W3C define a esta estructura **Flexbox** como un “flujo flexible” de elementos en dirección (arriba/abajo/izquierda/derecha) y tamaño (anchura/altura) según los ejes principal (horizontal) y transversal (vertical).



15. Flexbox – display:flex

Ejemplos de maquetación utilizando Flexbox.

Para poder probar todas las posibilidades que ofrece Flexbox CSS, vamos a crear una capa `class="contenedor"` que hará de padre y siete capas `class="elemento"` numeradas. ¿Y por qué siete? Para los ejemplos nos viene bien que sea un número impar para ver tanto la ordenación como la adaptación y su resultado, y siete también para ver un listado de elementos no muy corto, aunque podéis probar a colocar cualquier número de elementos.

```
1 .contenedor{  
2   display: flex;  
3 }
```

Para comenzar a utilizar Flexbox añadimos al contenedor la propiedad `"display:flex"`.

Como vemos, al no haber definido aún el comportamiento de dirección y tamaño que tendrán los elementos de nuestro contenedor, los elementos se adaptan a su padre ocupando el 100% de anchura entre la suma de todos. Por defecto, tiene ese comportamiento `"flexible"` como indica su nombre. Pero Flexbox es mucho más.

```
1 <div class="contenedor">  
2   <div class="elemento">1</div>  
3   <div class="elemento">2</div>  
4   <div class="elemento">3</div>  
5   <div class="elemento">4</div>  
6   <div class="elemento">5</div>  
7   <div class="elemento">6</div>  
8   <div class="elemento">7</div>  
9 </div>
```

EJEMPLO 1:

