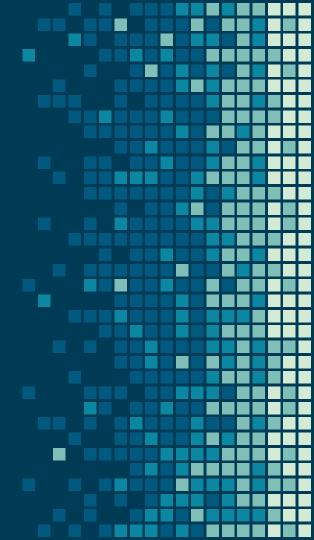
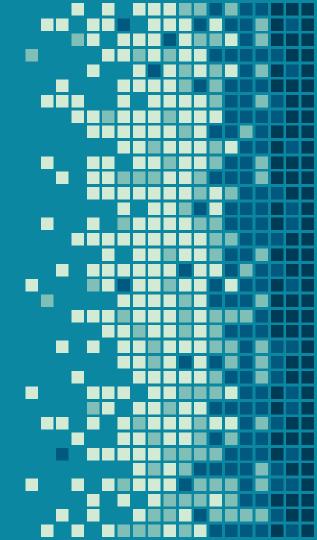
UD2 – Creación de páginas web estáticas con HTML5 y CSS3

Lenguajes de Marcas y Sistemas de Gestión de la Información



66 Índice

- 1. Introducción
- 2. Estructura
- 3. Etiquetas y atributos HTML
- 4. HTML5
- 5. Hojas de estilos CSS
- 6. CSS3
- 7. Responsive Web Design
- 8. XHTML



1. Introducción





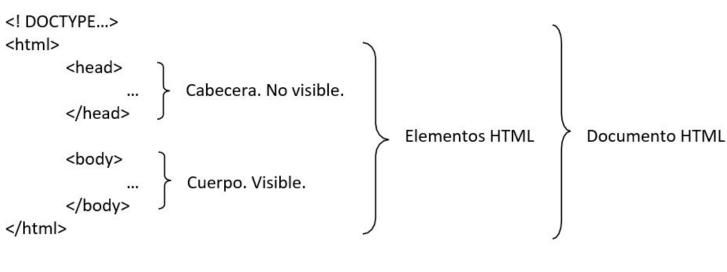
1. Introducción

Documentos HTML (hypertext markup language):

Ficheros de texto plano escritos utilizando el lenguaje HTML y cuya extensión es .html o .htm y que permiten visualizar contenido (estático) en los navegadores.



2. Estructura





2. Estructura

```
hola_mundo.html 

c!DOCTYPE html>
chead>
chead>
ctitle>Pagina hola mundo</title>
c/head>
chead>
chea
```



Ejemplo elemento HTML:

	etiqueta d	e inicio	contenido	etiqueta de cierre
<р	align-	"center">	Hola mundo	
	atributo	valor		



Declaración de documento HTML:

<! DOCTYPE html>

*La codificación por defecto de HTML5 es UTF-8



<html>

Lang: El atributo lang aunque es optativo es recomendable para indicar el idioma en que está escrita la página

<html lang="es">



<head>

 Title: Define un título en la barra de herramientas del navegador.

```
<head>
<title>My page</title>
</head>
```



<head>

Meta: Datos (información) sobre los datos de la propia página. Dicha información puede ser utilizada por los navegadores (cómo mostrar contenido o recarga de la página), los motores de búsqueda (palabras clave) u otros servicios web.

<head>

 Link: Permite invocar a un recurso externo a la página a fin de incorporar a la misma el contenido del recurso.

```
<head>
</head>
</r>
```

k>

atributo	uso	
rel	Indica la relación del recurso con la página. Posibilidades:(algunas)	
	 stylesheet. Es la que más se utiliza, indica que el recurso es una hoja de estilos (CSS). 	
	 alternate. Indica que el recurso es una versión alternativa del documento actual. Se usa bastante. 	
	 icon. Indica que el recurso es el icono de la página web. Se suele usar icon para iconos de tamaño 60x60, mientras que se usa short icon para iconos de tamaño 12x12 (normalmente se usan ambas entradas, escribiendo una etiqueta link distinta para cada tipo de icono). 	
href	Dirección URL al recurso solicitado; puede ser relativa o absoluta.	
hreflang	Propio de HTML 5. Indica el lenguaje en el que está escrito el recurso.	
type	Tipo MIME del documento relacionado. Los tipos MIME son indicaciones pertenecientes al protocolo http que usan los servidores web para indicar el tipo de información. Así por ejemplo text/css es el tipo MIME que indica que un recurso es una hoja de estilos CSS.	
media	Permite indicar el tipo de dispositivo más adecuado para mostrar el recurso.	

^{*} rel y href obligatorios

<head>

```
Style: Permite colocar código CSS en la página web.
     <head>
           <style>
                body {background-color: powderblue;}
                 h1 {color: blue;}
                    {color: red;}
           </style>
      </head>
```



<head>

Script: Permite colocar código script en la página web. Hoy en día lo habitual es utilizar esta etiqueta para incorporar código JavaScript a la misma. Los usos más comunes de JavaScript son la manipulación de imágenes, validación de formularios, y los cambios dinámicos de contenido.

Head: <script> function myFunction() { document.getElementById("demo").innerHTML = "Paragraph changed."; </script> **Body:** A Paragraph <button type="button" onclick="myFunction()">Try it</button>

<head>

- Base:
 - Atributo href: Especifica la URL por defecto (la carpeta raíz) para todos los hipervínculos que figuren en la página y que se indiquen de forma relativa.
 Si no se indica, los enlaces relativos parten del directorio que alberga a la página.

Si tenemos una ruta de nombre "logo.png", realmente será la ruta http://www.mipagina.com/conts/logo.png



<head>

- Base:
 - Atributo target: Mediante el atributo target se especifica el tipo de destino del enlace.

Value	Description
_blank	Opens the link in a new window or tab
_self	Default. Opens the link in the same frame as it was clicked
_parent Opens the link in the parent frame	
_top	Opens the link in the full body of the window
framename Opens the link in a named frame	

<body>

Encabezados: Para fijar apartados (títulos, subtítulos,...) se usan los encabezados mediante los elementos **h1** (más importante), **h2**, **h3**, **h4**, **h5** y **h6** (menos importante). Es importante utilizarlos para mostrar la estructura del documento.

El texto se visualiza en negrita y se añade un salto de línea antes y después.

```
<body>
<h1>Hola mundo</h1>
</body>
```

```
<h1>h1 tag</h1>
<h2>h2 tag</h2>
<h3>h3 tag</h3>
<h4>h4 tag</h4>
<h5>h5 tag</h5>
<h5>h5 tag</h5>
```

<body>

 Párrafo: Se trata de un nuevo párrafo. Se añade automáticamente una línea en blanco antes y después de un párrafo.

```
<body>
Esto es un párrafo
</body>
```

Espacios obligatorios: añadiendo la entidad

<body>

Salto de línea:

Línea horizontal:

<hr/>

Comentarios: Los comentarios no se muestran en el navegador,
 pero pueden ayudar a documentar su código HTML. Pueden ir también en el head u otras partes del documento.

<!--Esto es un comentario -->

<body>

<i>Este texto estará en itálica.</i>

Formateo básico del texto:

Voy a tachar losiguiente

Tag	Description			
<u></u>	Defines bold text			
	Defines emphasized text			
<u><i>></i></u>	Defines a part of text in an alterna	te voice or mood		
<small></small>	Defines smaller text	<html> <head></head></html>		
	Defines important text	<title></title>		
	Defines subscripted text	<body></body>		
	Defines superscripted text	 b>Ejemplos		
<ins></ins>	Defines inserted text	<hr/> <u>Lenguaje de marcas subrayado</u> 		
	Defines deleted text	<pre></pre>		
<mark></mark>	Defines marked/highlighted text	<i>Lenguaje de marcas cursiva</i>		

^{**} Para el formateo completo se usará CSS

<body>

- Listas:
 - Con viñetas: Puede cambiarse el estilo con CSS

```
<body>
```

</body>

Agua

Vino

Cerveza

- Agua
- Vino
- Cerveza





<body>

Listas:

Numéricas:

<body>

<0|>

Туре	Description
type="1"	The list items will be numbered with numbers (default)
type="A"	The list items will be numbered with uppercase letters
type="a"	The list items will be numbered with lowercase letters
type="I"	The list items will be numbered with uppercase roman numbers
type="i" The list items will be numbered with lowercase roman	

```
Agua
```

Vino

Cerveza

</0|>

- 1. Agua
- 2. Vino
- Cerveza

<body>

- Listas:
 - Descriptivas:

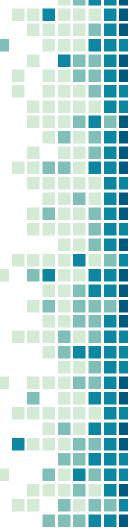
</dl>

Coffee

black hot drink

Milk

- white cold drink



25

- Listas:
 - Anidadas:

- Coffee
- Tea
 - Black tea
 - Green tea
- Milk

- Enlaces: etiqueta <a>
 Una URL está formada por:
 - Protocolo:
 - o http:// (para recursos de la web)
 - https:// (para recursos de la web contenidos en un servidor seguro)
 - o ftp:// (recursos contenidos en un servidor de ficheros)
 - o ftps:// (recursos contenidos en un servidor de ficheros seguro)
 - o mailto: (dirección de correo electrónico)
 - o file:/// (recurso dentro de nuestra propia computadora)

- Enlaces: etiqueta <a>Una URL está formada por:
 - Servidor. Nombre completo en Internet del servidor que aloja el recurso al que deseamos acceder. Por ejemplo: www.iessierradeguara.com
 - Puerto. Puerto por el que se debe conectar con el servidor para obtener el recurso. Si no se indica (que es lo habitual) se toma el puerto por defecto. Ejemplo en http se usa el 80. Si queremos usar uno en particular se indica tras el servidor poniendo dos puntos y el puerto. Por ejemplo www.qqq.com:1521

- Enlaces: etiqueta <a>Una URL está formada por:
 - Ruta: Indica el recorrido dentro del servidor que hay que hacer en sus directorios para llegar al recurso que queremos. Se pone después del servidor. Ejemplos:
 - o /index.html Accede a la página index.html situada en la raíz del servidor.
 - o /imagenes/paisajes/foto001.jpg Accede a la imagen foto001.jpg dentro del directorio paisajes dentro, a su vez, del directorio imagenes

<body>

- Enlaces: etiqueta <a>
 - Una URL está formada por:
 - Ruta: Cadena de búsqueda. Sólo aparece tras direcciones a páginas web que admitan recibir parámetros (como las páginas PHP, ASP o JSP por ejemplo). Ejemplo *?pagina=5&idioma=es*, pasará los parámetros pagina e idioma usando los valores 5 y es respectivamente.

Ejemplo de URL completa:

http://www.ejemplourl.com:9000/dirs/srv/pagina1.php?a=90&r=7

<body>

Enlaces: etiqueta <a>
 El atributo **href** permite indicar la URL del enlace

Tiberio

Tiberio (estará en el mismo directorio que el actual html)

Tiberio

Tiberio



<body>

Enlaces: etiqueta <a>

Enlace a correos:

· Envía mail



<body>

Enlaces: etiqueta <a>

Atributo id: Navegar a determinada parte de una página

```
<a id="punto_1">Aquí</a>
```

Volver arriba

Si el enlace se realiza desde otra página:

lr



<body>

- Img: imágenes, atributos obligatorios
 - src (ruta absoluta o relativa a la imagen)
 - alt (texto alternativo que se muestra cuando no se carga la imagen o en los lectores de pantallas utilizados por personas con discapacidad visual)

```
<img src="html5.gif" alt="The official HTML5 Icon" />
```


**La W3C recomienda adaptar el tamaño mediante estilos CSS



<body>

lmg

- src: se añade el nombre de la imagen.
- width: se establece el ancho de la imagen.
- *height*: establece el alto de la imagen.
- alt: se usa para dar una breve descripción de la imagen.
- align: para la alineación de la imagen y puede tomar los valores:
 - top: se sitúa en la parte superior del texto.
 - middle: se sitúa centrada respecto al texto.
 - bottom: se alineará con la parte inferior del texto.
 - left: se alinea a la izquierda.
 - right: se alinea a la derecha.
- border: se le asigna un borde con valor numérico.

<body>

Hipervínculos en imágenes:

```
<a href="mipagina.html">
    <img src="html5.gif" alt="HTML5 Icon" width="128" height="128" />
    </a>
```

<body>

Mapas de imágenes



<Click on the computer, the phone, or the cup of coffee to go to a new page and read more about the topic:</p>

```
<img src="workplace.jpg" alt="Workplace" usemap="#workmap" width="400" height="379">
```

```
<map name="workmap">
  <area shape="rect" coords="34,44,270,350" alt="Computer" href="computer.htm">
  <area shape="rect" coords="290,172,333,250" alt="Phone" href="phone.htm">
  <area shape="circle" coords="337,300,44" alt="Cup of coffee" href="coffee.htm">
  </map>
```

<body>

- Table: tablas en html
 - (define una tabla html)
 - <caption> (agrega un título a la tabla)
 - (celda de cabecera en la primera línea, lo pone en negrita y centrado)
 - (cada elemento tr define una fila, va dentro de)
 - (cada elemento td define una columna, va dentro de)

<body>

Table:

Companies

Company	Contact	Country Germany	
Alfreds Futterkiste	Maria Anders		
Centro comercial Moctezuma	Francisco Chang	Mexico	

```
<caption>Companies</caption>
 Company
  Contact
  Country
 Alfreds Futterkiste
  Maria Anders
  Germany
 (tr>
 Centro comercial Moctezuma
  Francisco Chang
  Mexico
```

<body>

Combinar celdas en una tabla, válido para y

atributo	significado
colspan	Combina la celda actual con el número de celdas a la derecha que se indique. Por ejemplo <i>colspan="3"</i> une esta celda con las dos que tiene a su derecha, formando una combinación de tres celdas en horizontal.
rowspan	Combina la celda actual con el número de celdas hacia abajo que se indique. Por ejemplo <i>rowspan="3"</i> une esta celda con las dos que tiene hacia abajo, formando una combinación de tres celdas en vertical.

```
January
100
4d>$100
4d>$50

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100

100
100
```

<body>

Hay tres elementos HTML que sirven para diferenciar las tres partes principales de una tabla, son:

- thead. Sirve para indicar las filas que forman la cabecera de la tabla
- tfoot. Indica el pie de la tabla
- tbody. Indica el cuerpo de la tabla

De esa forma se podrá más adelante dar formato diferencia a cada parte mediante CSS

<body>

Form: Los formularios son un elemento en las páginas web
 que permiten recabar información para enviarla a un servidor de
 aplicaciones y que la procese haciendo uso de un lenguaje de
 programación web (PHP, ASP, ...).

Nombre:	
Apellido:	
Email:	
Varón:	
Mujer:	
Enviar Restablecer	

<body>

Form

Tag	Description	
<form></form>	Defines an HTML form for user input	
<input/>	Defines an input control	
<textarea></td><td>Defines a multiline input control (text area)</td></tr><tr><td><label></td><td>Defines a label for an <input> element</td></tr><tr><td><fieldset></td><td>Groups related elements in a form</td></tr><tr><td><legend></td><td>Defines a caption for a <fieldset> element</td></tr><tr><td><select></td><td>Defines a drop-down list</td></tr><tr><td><optgroup></td><td>Defines a group of related options in a drop-down list</td></tr><tr><td><option></td><td>Defines an option in a drop-down list</td></tr><tr><td><button></td><td>Defines a clickable button</td></tr><tr><td><datalist></td><td colspan=2>Specifies a list of pre-defined options for input controls</td></tr><tr><td><keygen></td><td colspan=2>Defines a key-pair generator field (for forms)</td></tr><tr><td><output></td><td colspan=2>Defines the result of a calculation</td></tr></tbody></table></textarea>		

<body>

Form

Attribute	Value	Description
accept-charset	character_set	Specifies the character encodings that are to be used for the form submission
action	URL	Specifies where to send the form-data when a form is submitted
autocomplete 5	on off	Specifies whether a form should have autocomplete on or off
enctype	application/x-www-form-urlencoded multipart/form-data text/plain	Specifies how the form-data should be encoded when submitting it to the server (only for method="post")
method	get post	Specifies the HTTP method to use when sending form-data
name	text	Specifies the name of a form
novalidate 5	novalidate	Specifies that the form should not be validated when submitted
target	_blank _self _parent _top	Specifies where to display the response that is received after submitting the form

<body>

Input. Cuadro de texto:



<body>

Input. Cuadro de contraseñas:



<body>

Input. Botón de envío:



<body>

 Input. Botón de restablecer (anula los datos introducidos y los deja en su estado inicial)

<body>

Input. Botón genérico

**Define un botón que se puede pulsar (en la mayoría de los casos se utiliza con un JavaScript para activar una secuencia de comandos)

<body>

O Male O Female

Atributo booleano llamado **checked** si queremos marcar uno por defecto

<body>

```
Input. Casillas de verificación. Check Box

<form action="/action_page.php" method="post">

<input type="checkbox" name="vehicle" value="Bike">I have a bike<br/>
<input type="checkbox" name="vehicle" value="Car">I have a car

I have a bike

I have a car
```

Atributo booleano llamado **checked** si queremos marcar uno por defecto

<body>

Suba un archivo: Seleccionar archivo Ningún archi... seleccionado



<body>

- Más tipos de Input : http://www.w3schools.com/tags/att input type.asp
- Más atributos de Input : http://www.w3schools.com/tags/tag input.asp



<body>

Select. Lista desplegable

```
<form action="/action page.php">
  <label for="cars">Choose a car:</label>
  <select name="cars">
    <optgroup label="Swedish Cars">
      <option value="volvo">Volvo</option>
      <option value="saab">Saab</option>
   </optgroup>
    <optgroup label="German Cars">
      <option value="mercedes">Mercedes</option>
      <option value="audi">Audi</option>
    </optgroup>
 </select>
  <input type="submit" value="Submit">
</form>
```

Choose a car: Volvo Swedish Cars
Submit Volvo
Saab
German Cars
Mercedes

Audi

<body>

Select. Lista desplegable

Attribute		Value	Description
autofocus	5	autofocus	Specifies that the drop-down list should automatically get focus when the page loads
disabled		disabled	Specifies that a drop-down list should be disabled
<u>form</u>	5	form_id	Defines one or more forms the select field belongs to
multiple		multiple	Specifies that multiple options can be selected at once
name		name	Defines a name for the drop-down list
required	9	required	Specifies that the user is required to select a value before submitting the form
size		number	Defines the number of visible options in a drop-down list



<body>

Select. Lista desplegable

Atributo: multiple

<select name="cars" multiple>



Submit



<body>

textarea. Área de texto.

```
<form action="/action_page.php">
    Escribe un texto descriptivo: <br />
    <textarea rows="10" cols="40" name="texto"></textarea>
</form>
```



<body>

Iframe. Marco flotante, se utiliza para mostrar otro documento (página web, pdf...) dentro de la misma web.

<iframe src="./pagina.html" width="100%" height="500">

El navegador no soporta iframe

</iframe>



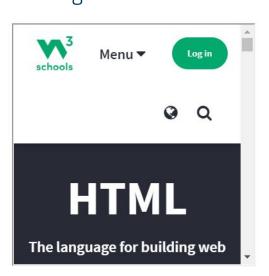
<body>

Iframe.

<iframe src="https://www.w3schools.com" height="300" width="300"</pre>

title="Iframe Example"></iframe>

Por defecto el marco se visualiza con un borde, si se desea eliminar basta con asignar valor 0 al atributo frameborder



<body>

Plugins: object
 Los plugins son programas informáticos que amplían la funcionalidad
 estándar de un navegador web, como por ejemplo applets de Java, lectores
 de PDF, Flash Players,...

Object define un objeto incrustado:

```
<object width="400" height="50" data="bookmark.swf"></object>
```

<object width="100%" height="500px" data="snippet.html"></object>

<object data="audi.jpeg"></object>

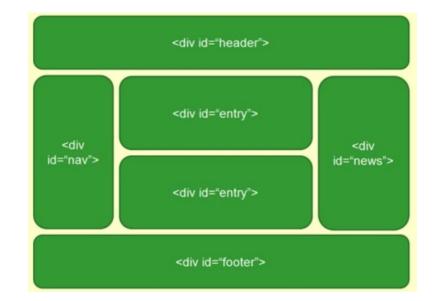
<body>

- Elementos de agrupamiento:
 - De bloque: siempre empiezan en una nueva línea y ocupan todo el espacio disponible, aunque sus contenidos no lo necesite. Pueden contener otros elementos inline o block. Elementos de bloque son, por ejemplo: , <div>, , , <form>, etc
 - **En línea:** sólo ocupan el espacio necesario para mostrar sus contenidos. No crean una nueva línea. Se colocan 'en línea' con otros elementos. Sólo pueden contener otros elementos inline. Se emplean para seleccionar un fragmento del contenido y resaltarlo, crear enlaces, etc... Elementos en línea son, por ejemplo: <a>, , , , etc

```
<body>
                                        <div id="myDiv">
                                          <h2>This is a heading in a div element</h2>
     Elementos de agrupamiento:
                                          This is some text in a div element.
                                        </div>
          <div> (de bloque) y <span> (en línea) no producen ningún efecto
          visualmente (salvo el salto de línea del div).
          Se utilizan para agrupar contenido que esté relacionado de
          alguna manera y al que con estilos CSS se pretenda maquetar de
          manera conjunta.
                                             <span id="entrada">
                                               <strong>Articulo Nuevo</strong>,
                                               <em>,12 de marzo de 2016</em>
          ¡Cuidado con el uso excesivo de divs!
```


<body>

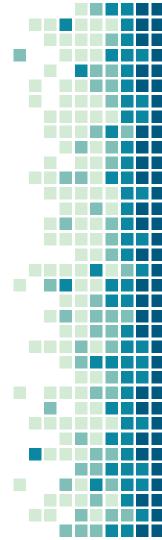
• Ejemplo de esquema de maquetación muy utilizado





<body>

- Novedades con HTML5:
 - Elementos multimedia, como <audio> y <video>
 - Elementos semánticos, como <header>, <footer>, etc
 - Nuevos atributos en los elementos de los formularios como email, number, date, time, range o calendar...
 - Elementos gráficos como <canvas>



<body>

- Novedades con HTML5:
 - Elementos multimedia, como <audio> y <video>
 - Elementos semánticos, como <header>, <footer>, etc
 - Nuevos atributos en los elementos de los formularios como email, number, date, time, range o calendar...
 - Elementos gráficos como <canvas>



<body>

Video: reproducción de un vídeo



**Es conveniente ponerlo en varios formatos, no todos los navegadores soportan todos los formatos

<body>

Video: reproducción de un vídeo

File Format	Media Type
MP4	video/mp4
WebM	video/webm
Ogg	video/ogg



<body>

Video: reproducción de un vídeo

Attribute	Value	Description
autoplay	autoplay	Specifies that the video will start playing as soon as it is ready
controls	controls	Specifies that video controls should be displayed (such as a play/pause button etc).
<u>height</u>	5 pixels	Sets the height of the video player
<u>Іоор</u>	loop	Specifies that the video will start over again, every time it is finished
muted	5 muted	Specifies that the audio output of the video should be muted
poster	URL URL	Specifies an image to be shown while the video is downloading, or until the user hits the play button
preload	auto metadata none	Specifies if and how the author thinks the video should be loaded when the page loads
src	 URL	Specifies the URL of the video file
width	5 pixels	Sets the width of the video player



```
<body>
```

Track: permite incluir subtítulos

```
<body>
```

Audio: reproducción de un audio

```
▶ 0:01 / 0:01 ------ •>
```



<body>

Audio: reproducción de un audio

File Format	Media Type
MP3	audio/mpeg
OGG	audio/ogg
WAV	audio/wav



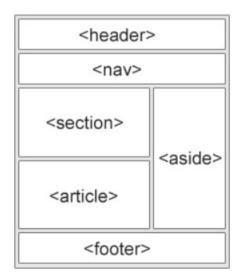
<body>

Audio: reproducción de un audio

Attribute		Value	Description
autoplay	5	autoplay	Specifies that the audio will start playing as soon as it is ready
controls	5	controls	Specifies that audio controls should be displayed (such as a play/pause button etc)
loop	9	loop	Specifies that the audio will start over again, every time it is finished
muted	9	muted	Specifies that the audio output should be muted
preload	5	auto metadata none	Specifies if and how the author thinks the audio should be loaded when the page loads
src	目	URL	Specifies the URL of the audio file

<body>

 Elementos semánticos/estructurales: no dan ningún formato al texto pero permiten agruparlo dándole un significado y además permitiendo posteriormente que ese contenido tenga un formato especial mediante CSS.



<body>

<header>: Usado para albergar el contenido del encabezado del sitio

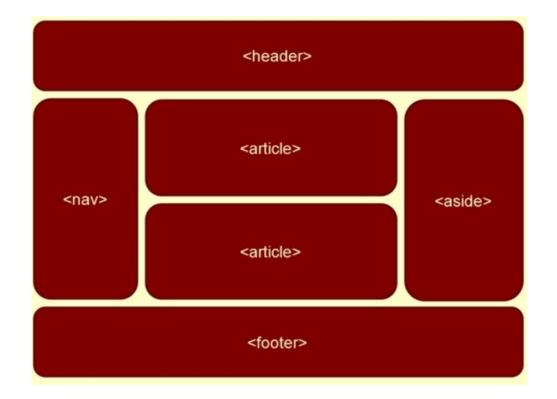
<footer>: Aloja el contenido del pie de página de un sitio.

<nav>: Contiene el menú de navegación u otra funcionalidad de navegación de la página.

<article>: Contiene una pieza independiente de contenido que tendría sentido como, por ejemplo, una noticia.

<section>: Se utiliza tanto para agrupar diferentes artículos en diferentes propósitos o temas, como para definir las distintas secciones de un único artículo

<aside>: Define un bloque de contenido que está relacionado con el contenido principal que lo engloba, pero no es fundamental para el flujo del mismo.





```
<!DOCTYPE html>
<html lang="es">
<head>
    <title>Título de la WEB</title>
    <meta charset="UTF-8">
    <meta name="title" content="Título de la WEB">
    <meta name="description" content="Descripción de la WEB">
    <link href="estilos.css" rel="stylesheet">
</head>
<body>
    <header>
        <h1>Título de la WEB</h1>
    </header>
    <nav>
        <a href="#">IR SECCIÓN 2</a>
        <a href="#">IR SECCIÓN 3</a>
    </nav>
    <section>
        <article>
            <h2>CONTENIDO PRINCIPAL</h2>
            Este es el contenido principal de mi web.
       </article>
    </section>
    <aside>
        <h3>Testimonios</h3>
        Me gusta mucho esta página.
    </aside>
    <footer>
        <h4>Redes sociales</h4>
        <a href="#">Facebook</a>
        <a href="#">Tiwtter</a>
    </footer>
</body>
</html>
```

<body>

Plugins: embed
 Los plugins son programas informáticos que amplían la funcionalidad
 estándar de un navegador web, como por ejemplo applets de Java, lectores
 de PDF, Flash Players,...

Embed define un objeto incrustado:

```
<embed type="text/html" src="snippet.html" width="500" height="200">
<embed type="image/jpg" src="pic_trulli.jpg" width="300" height="200">
<embed type="video/mp4" src="movie.mp4">
```

<body>

Embed attributes

Attribute		Value	Description
<u>height</u>	5	pixels	Specifies the height of the embedded content
src	9	URL	Specifies the address of the external file to embed
type	5	media_type	Specifies the media type of the embedded content
width	티	pixels	Specifies the width of the embedded content



<body>

 Canvas: se utiliza para dibujar gráficos a través de secuencias de comandos (normalmente JavaScript).

```
<canvas id="myCanvas" width="200" height="100"></canvas>

<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.font = "30px Arial";
ctx.strokeText("Hello World", 10, 50);
</script>

Smile!
```





<body>

• SVG (Scalable Vector Graphics): El lenguaje SVG es un lenguaje ya veterano basado en XML que sirve para dibujar gráficos en una página web.

<body>

Math:

Fórmulas matemáticas

**Debemos incluir código JavaScript para que funcione

```
<math>
 <msub><mi>x</mi><mtext>0</mtext></msub>
 <mo>=</mo>
 <mn>2</mn>
 <msup>
   <mfenced open="(" close=")">
    <mfrac>
      <msup>
        <mi>y</mi><mn>2</mn>
      </msup>
      <msup>
        <mi>z</mi><mn>3</mn>
      </msup>
    </mfrac>
   </mfenced>
                    x_0=2\Big(rac{y^2}{z^3}\Big)
   <mn>2</mn>
 </msup>
```

Más novedades de HTML5 <u>aquí</u>



- Las hojas de estilos se crearon para poder separar el contenido de un documento de su aspecto o presentación, lo que permite tener documentos mejor estructurados y limpios.
- Las hojas de estilos pueden aplicarse a un documento entero, a una parte o simplemente a una etiqueta concreta.
- Permiten modificar la presentación de un documento HTML sin tocar su código.

CSS (Cascading Style Sheets): es un lenguaje de hojas de estilo que permite detallar cómo se va a mostrar la información presente en el documento en la pantalla dependiendo del dispositivo.

CSS permite a los desarrolladores controlar el estilo y el formato de múltiples páginas al mismo tiempo. Cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento



- Formas más conocidas de dar estilo a un documento:
 - 1) Utilizando una hoja de estilo externa (que se almacena con el formato .css) 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="./ruta_al_fichero/fichero.css" />
</head>
```

. . . .

- Formas más conocidas de dar estilo a un documento:
 - **1)** Utilizando una **hoja de estilo externa** (que se almacena con el formato .css) En el fichero *fichero.css* se colocarían los estilos css, por ejemplo:

```
td {
  background-color: green;
  font-family: Arial Black;
}
```

*Es la mejor manera ya que así los estilos fijados en el fichero.css pueden ser utilizados en más de un documento HTML.

- Formas más conocidas de dar estilo a un documento:
 - **2)** Estilos aplicados en el mismo documento. Utilizando el elemento <style> en el interior del documento al que se le quiere dar estilo y que se sitúa en la sección <head>.

```
<head>
     <style type="text/css">
           td {
             background-color: green;
             font-family: Arial Black;
     </style>
</head>
```

Formas más conocidas de dar estilo a un documento:

3) Utilizando estilos directamente sobre aquellos elementos que lo permiten a través del atributo <style> dentro de <body>. (Se desaconseja ya que este tipo de definición del estilo pierde las ventajas que ofrecen las hojas de estilo al mezclarse el contenido con la presentación)

Versiones de CSS:

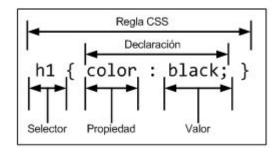
CSS 1: a finales de 1996, se publica la primera versión que permite describir formato, alineación y atributos de texto, así como las propiedades de las fuentes (tamaño, color, tipo, etc.). Actualmente, no se usa.

CSS 2: incluye casi la totalidad de CSS 1. A mediados de 1998, se publica la segunda recomendación oficial de W3C. Añade como mejora para las capas <DIV> el tipo de posicionamiento y nivel.

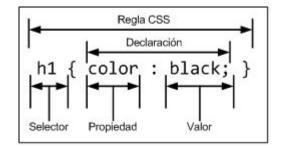
CSS 2.1: elimina errores y funcionalidades en desuso de versiones anteriores.

CSS 3: usa módulos que añaden nueva funcionalidad a CSS 2, preservando de esa manera la compatibilidad con versiones anteriores. Surge a finales del 2011.

Sintaxis: **Reglas CSS**. Cada regla está compuesta de una parte de "selectores", un símbolo de "llave de apertura" ({), otra parte denominada "declaración" y por último, un símbolo de "llave de cierre" (}).



- Sintaxis: Reglas CSS.
 - Selector: indica el elemento o elementos HTML a los que se aplica la regla CSS.
 - <u>Declaración</u>: especifica los estilos que se aplican a los elementos. Está compuesta por una o más parejas propiedad-valor:
 - Propiedad: permite especificar una característica del elemento.
 - Valor: indica el valor de la característica.



Sintaxis: Reglas CSS.

 Una hoja de estilos CSS puede contener infinitas reglas CSS, cada regla puede contener infinitos selectores y cada declaración puede estar formada por un número infinito de pares propiedad/valor.

```
p {
          line-height:10pt;
          text-decoration:underline;
          text-align:center;
}
```

Sintaxis: **Reglas CSS**.

¿Cuántos selectores hay en la siguiente hoja de estilos?

```
body{ background-color:Blue;}
h1{font-family:Verdana; color:White;}
li{ text-decoration:underline}
```



Tipos de selectores:

1) Selector de tipo:

```
p {
  text-align: center;
  color: red;
}
```

Para no repetir código, mejor agrupar tipos de elementos si tienen los mismos estilos:

```
h1, h2, p {
  text-align: center;
  color: red;
}
```

Tipos de selectores:

2) Selector universal:

```
* {
  text-align: center;
  color: blue;
}
```

Aplica a todos los elementos de la página web, no suele ser utilizado



Tipos de selectores:

3) Selector descendiente:

```
<!DOCTYPE html>
<html>
<head>
<style>
div p {
 background-color: yellow;
</style>
</head>
<body>
<div>
  Paragraph 1 in the div.
  Paragraph 2 in the div.
  <section>Paragraph 3 in the div.</section>
</div>
Paragraph 4. Not in a div.
Paragraph 5. Not in a div.
</body>
</html>
```

Tipos de selectores:

4) Selector de clase:

```
<!DOCTYPE html>
<html>
<head>
<style>
.myclass {
 text-align: center;
 color: red;
</style>
</head>
<body>
<h1 class="myclass">Red and center-aligned heading</h1>
Not in the class
</body>
</html>
```

</html>

Tipos de selectores: <!DOCTYPE html> <html> 4) Selector de clase: <head> <style> p.center { text-align: center; color: red; </style> </head> <body> <h1 class="center">This heading will not be affected</h1> This paragraph will be red and center-aligned. </body>

- Tipos de selectores:
 - **4) Selector de clase**: A un mismo elemento se le pueden asignar varias clases. Todo lo que tenemos que hacer es declarar los nombres de las clases separados por un espacio (por ejemplo, class="texto1 texto2").

Tipos de selectores:

5) Selector por id:

```
<!DOCTYPE html>
<html>
<head>
<style>
#para1 {
 text-align: center;
 color: red;
</style>
</head>
<body>
Hello World!
This paragraph is not affected by the style.
</body>
</html>
```



Tipos de selectores:

6) Selector hijo:

*solo afecta a los hijos directos

```
<!DOCTYPE html>
<html>
<head>
<style>
div > p {
 background-color: yellow;
</style>
</head>
<body>
<div>
 Paragraph 1 in the div.
 Paragraph 2 in the div.
 <section>
   <!-- not Child but Descendant -->
   Paragraph 3 in the div (inside a section element).
 </section>
 Paragraph 4 in the div.
</div>
Paragraph 5. Not in a div.
Paragraph 6. Not in a div.
</body>
</html>
```

Tipos de selectores:

7) Selector advacente:

*solo afecta a los elementos inmediatamente después

```
<!DOCTYPE html>
<html>
<head>
<style>
div + p {
 background-color: yellow;
</style>
</head>
<body>
(div)
 Paragraph 1 in the div.
 Paragraph 2 in the div.
</div>
Paragraph 3. After a div. Affected
Paragraph 4. After a div.
<div>
 Paragraph 5 in the div.
 Paragraph 6 in the div.
</div>
Paragraph 7. After a div. Affected 
Paragraph 8. After a div.
</body>
</html>
```

Tipos de selectores:

8) Selector hermano:

```
<!DOCTYPE html>
<html>
<head>
<style>
div ~ p {
 background-color: yellow;
</style>
</head>
<body>
<h2>General Sibling Selector</h2>
The general sibling selector (~) selects all
elements that are next siblings of a specified element.
Paragraph 1.
<div>
 Paragraph 2.
</div>
Paragraph 3. Affected
<code>Some code.</code>
Paragraph 4. Affected
</body>
</html>
```

Resumen(1):

Selector	Example	Example description
<u>#id</u>	#firstname	Selects the element with id="firstname"
<u>.class</u>	.intro	Selects all elements with class="intro"
element.class	p.intro	Selects only elements with class="intro"
*	*	Selects all elements
<u>element</u>	р	Selects all elements
element,element,	div, p	Selects all <div> elements and all elements</div>



Resumen(2):

Selector	Example	Example description	
element element	div p	Selects all elements inside <div> elements</div>	
element>element	div > p	Selects all elements where the parent is a <div> element</div>	
<u>element+element</u>	div + p	Selects the first $<$ p $>$ element that are placed immediately after $<$ div $>$ elements	
element1~element2	p ~ ul	Selects every element that are preceded by a element	



- Las propiedades son la pieza central de CSS. Todos los estilos que podemos aplicar a un elemento se definen por medio de propiedades. Ya hemos introducido algunas en los ejemplos anteriores, pero hay cientos de propiedades disponibles.
- Para simplificar su estudio, se pueden clasificar en dos tipos:
 - Propiedades de formato → Las propiedades de formato se encargan de dar forma a los elementos y su contenido
 - Propiedades de diseño → están enfocadas a determinar el tamaño y la posición de los elementos en la pantalla.

1) <u>Selectores</u>	2) <u>Fondos</u>	3) <u>Textos</u>
4) <u>Fuentes</u>	5) <u>Enlaces</u>	6) <u>Listas</u>
7) <u>Tablas</u>	8) <u>Modelo de cajas</u>	9) <u>Bordes</u>
10) <u>CSS Outline</u>	11) <u>Márgenes</u>	12) Rellenos
13) <u>Dimensiones</u>	14) <u>Visualización</u>	15) <u>Posicionamiento</u> <u>Otro enlace</u>



16) <u>Flotación</u>	17) <u>Alinear</u>	18) <u>Combina</u> <u>ndo</u> <u>selectore</u> <u>S</u>
19) <u>Selectores de</u> <u>pseudoclases</u>	20) <u>Selectores de</u> <u>pseudoelementos</u>	21) Barra de navegación
22) Galería de imagenes	23) <u>Opacidad</u>	



Herencia de estilos: Cuando se establece el valor de alguna propiedad en un elemento, todos sus descendientes heredan inicialmente ese mismo valor.

```
<!DOCTYPE html>
<html>
                 <head>
                                   <title>Email</title>
                                   <meta charset="utf-8" />
                                   <style>
     body {color:blue;padding:100px;margin:50px}
     /*comenta o descomenta para experimentar*/
     p{color:black}/*Esta anula la herencia de la prop.
      color definida en body. Resultado, el párrafo 2 en
      negro.*/
     p{font-size: 25px;}/*Esta no anula nada*/
    </style>
                 </head>
                 <body>
   <div style="border:1px solid;padding:inherit;">párrafo 1</div>
   párrafo 2
   <a href="#">Enlace</a>
 </body>
</html>
```

Herencia de estilos:

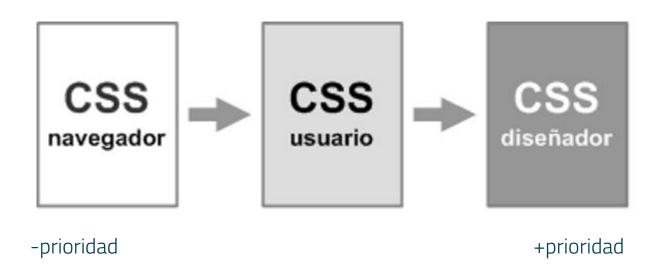
Propiedades que se heredan:	Propiedades que no se heredan:
color font (y sus propiedades relacionadas) letter-spacing line-height list-style (y sus propiedades relacionadas) text-align text-indent text-transform visibility white-space word-spacing	background (y sus propiedades relacionadas) border (y sus propiedades relacionadas) display float y clear height y width margin (y sus propiedades relacionadas) min- y max- height y -width outline overflow padding (y sus propiedades relacionadas) position (y sus propiedades relacionadas) text-decoration vertical-align z-index SI SE DESEA QUE UN HIJO HEREDE EL
	VALOR DEL PADRE, SE PUEDE INDICAR CON EL VALOR inherit .

Orden de aplicación de reglas de estilos

Además de las hojas de estilos definidas por los diseñadores, los navegadores aplican a cada página otras dos hojas de estilos:

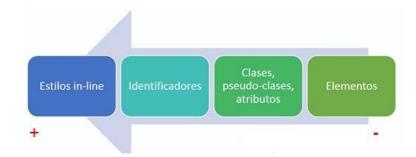
- la del propio navegador utilizado (se utiliza para establecer el estilo inicial por defecto a todos los elementos HTML: tamaños de letra iniciales, decoración del texto, márgenes entre elementos, etc.)
- **la del usuario** (cada usuario puede crear su propia hoja de estilos y aplicarla automáticamente a todas las páginas que visite con su navegador; se trata de una opción muy útil para personas discapacitadas visualmente, ya que pueden aumentar el contraste y el tamaño del texto de todas las páginas para facilitar su lectura).

Orden de aplicación de reglas de estilos



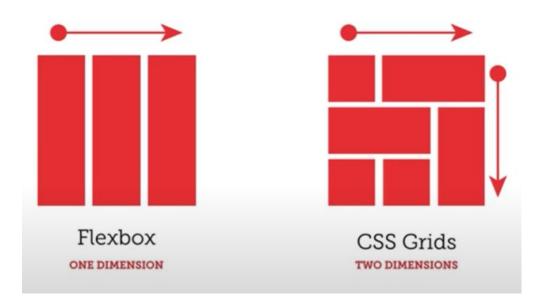


- Orden de aplicación de reglas de estilos (dentro del CSS del diseñador)
 - > Menor prioridad: estilos externos (los que se colocan con link)
 - > Después los que proceden de la etiqueta style
 - > Después se ordenan las declaraciones según lo específico que sea el selector. Cuanto más genérico es un selector, menos importancia tienen sus declaraciones



- Orden de aplicación de reglas de estilos (dentro del CSS del diseñador)
 - > A continuación aquellos que tengan la anotación !important
 - > Si después de aplicar las normas anteriores existen dos o más estilos con la misma prioridad, se aplica la que se indicó en último lugar.

Flexbox vs CSS Grids





Flexbox y derivados

Enlace

CSS GRID (cuadrículas)

Enlace



Otras novedades de CSS3:

- 1) Borders
- 2) Backgrounds
- 3) Gradients
- 4) Text Effects



Otras novedades de CSS3:

- 5) Fonts
- 6) 2D Transforms
- 7) 3D Transforms
- 8) <u>Transitions</u>



Otras novedades de CSS3:

- 9) Animations
- 10) Multiple Columns

. . .



CSS3 presenta nuevos selectores de atributos.

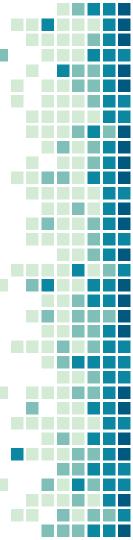
[attribute^=value] - Ejemplo: a[href^="https"] Selects every <a> element whose href attribute value begins with "https"

[attribute\$=value] - Ejemplo a[href\$=".pdf"] Selects every <a> element whose href attribute value ends with ".pdf"

[attribute*=value] - Ejemplo a[href*="w3schools"] Selects every <a> element whose href attribute value contains the substring "w3schools"



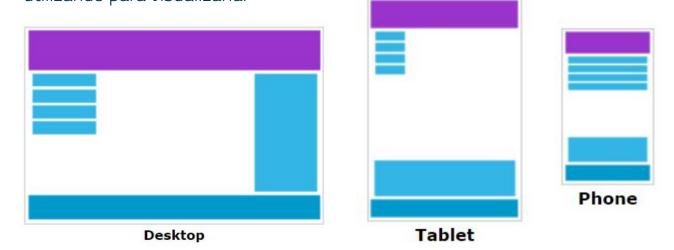
- Cuidado: algunas propiedades y elementos de CSS3, no están disponibles en todos los navegadores. Por ello en muchas ocasiones las propiedades CSS3 tienen distintas variantes según el navegador.
 - Los prefijos más utilizados actualmente son:
 - -webkit-. Para los navegadores que usen el motor conocido como Webkit (como Chrome o Safari).
 - -ms-. Para los navegadores de Microsoft (como Internet Explorer)
 - -o-. Para el navegador Opera.
 - -moz-. Para los navegadores Mozilla (como Firefox)



 Así pues, cuando una propiedad de CSS no nos funcione en un navegador o varios, podemos mirar si existe una propia que nos sirva para cada uno y añadirlas dentro de una misma clase.

```
#recuadro2 {
  -ms-transform: rotate(10deg);
  -webkit-transform: rotate(10deg);
  -moz-transform: rotate(10deg);
  -o-transform: rotate(10deg);
  transform: rotate(10deg);
}
```

El diseño web adaptable o adaptativo, conocido por las siglas RWD (Responsive Web Design) es una filosofía de diseño y desarrollo cuyo objetivo es adaptar la apariencia de las páginas web al dispositivo que se esté utilizando para visualizarla.



La fragmentación en dispositivos : un desafío para los desarrolladores web



Media queries

```
/* On screens that are 992px or less, set the background color to blue */
@media only screen and (max-width: 992px) {
 body {
     background-color: blue;
@media only screen and (max-width: 600px) {
 body {
     background-color: olive;
```

Media queries

```
@media only screen and (orientation: landscape) {//o... orientation: portrait
 body {
   background-color: lightblue;
@media only screen and (min-width: 600px) {
  div.example {
     display: none;
```



Media queries

```
@media only screen and (min-width:768px) and (max-width:1024px) {
    .example {
      background-color: blue;
    }
}
```



Media queries

Las podemos definir donde colocamos los propios estilos o podemos importar diferentes hojas de estilo según las características de la pantalla:

```
rel="stylesheet" media="screen and (min-width: 900px)"href="widescreen.css">

link rel="stylesheet" media="screen and (max-width: 600px)"href="smallscreen.css">

....
```

XHTML (eXtensible HyperText Markup Language): se considera como la adaptación de HTML en XML, pues mantiene las características de HTML e incluye normas XML. Es más estricto que HTML a nivel técnico.

XHTML hereda la rigidez de XML, con lo cual no se puede escribir documentos
 XHTML de cualquier manera, como ocurría con HTML, sino atendiendo a unas normas.

HTML

XML

XHTML

 Se puede procesar su contenido por cualquier programa informático (igual que ocurre con el XML)

Los navegadores no tienen por qué volverse locos intentando interpretar lo que el desarrollador ha querido escribir ni solucionar los posibles errores de código cometidos, como ocurría con HTML.



Sintaxis (comparada con HTML):

- Se necesita escribir una declaración DOCTYPE en el inicio del documento XHTML.
- El atributo xmlns en <html> es obligatorio
- Etiquetas y atributos deben ir en minúscula (XML es sensible a mayúsculas y minúsculas)
- Todos los atributos deben tener obligatoriamente un valor y deben ir entre comillas.
- Todas las etiquetas deben estar cerradas correctamente.
- Todas las etiquetas deben estar anidadas correctamente.
- El atributo "id" sustituye al atributo "name"

Ejemplo XHTML:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-/W3C//DTD XHTML 1.1//EN"</pre>
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd"
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Title of document</title>
</head>
<body>
Este es un ejemplo de XHTML
</body>
</html>
```

