

# UT 2. Lenguajes para la visualización de la información: HTML, XHTML y CSS

---

LENGUAJES DE MARCAS Y SISTEMAS DE GESTIÓN DE INFORMACIÓN

I.E.S. Luis Vives - Desarrollo de Aplicaciones Web

Curso 2022/2023

# Versiones de HTML

---

HTML: HyperText Markup Language

Hypertext: Enlaza a otro contenido

Markup: Trabaja con etiquetas `<p> Hola<\p>`

Lenguaje de marcado. No es un lenguaje de programación.

Las páginas creadas en HTML se localizan, normalmente en servidores web. Los clientes solicitan la página y el servidor la envía al cliente.

Refs:

- <https://www.w3schools.com> aprendizaje html
- <https://developer.mozilla.org/es/docs/Web/HTML> Estándares de la web

Las versiones de HTML están definidas por la comunidad internacional W3C (World Wide Web Consortium). HTML nace en el año 1991, desarrollado por el científico Tim Berners-Lee (conocido como el padre de la web) y era denominado como HTML Tags.

- HTML 1.0. 1991 20 etiquetas
- HTML 2.0. En 1995, el Internet Engineering Task Force (IETF) creó el primer estándar oficial, Mosaic, +19 etiquetas. DTD
- HTML 3.2. 1997. Es una recomendación. (1996 internet explorer), +tablas, mapas (imágenes con enlaces por zonas). 1ª versión de la W3C. Aparece CSS publicada por el W3C (incluyendo applets de Java, etc.).

- HTML 4.0. El 24 de abril de 1998 se publica esta versión, que presenta como novedades: tablas complejas, mejoras en los formularios, hojas de estilo, añadir script y mejora de accesibilidad.
- HTML 4.01. Recomendación. A finales de 1999, surge HTML 4.01 como actualización y revisión de HTML 4.0, aunque no aporta novedades relevantes.
- XHTML 1.0. 2000, Incorpora rigidez de HTML para validar documentos.
- HTML 5. Publicada a finales del 2014, establece nuevos elementos, algunos similares a los anteriores, pero con un significado semántico. Etiquetas semánticas, APIs (interfaces de programación adaptados a todas las etiquetas. Simplifica DOCTYPE, Link y Script. Mejorar formularios. La página web <https://html5test.com/> indica cuánto HTML5 cumple mi navegador

- CSS: Cascading Style Sheets. Hojas de estilo en cascada.
- El estilo definido en CSS se propaga en cascada a todas las páginas que afecte.
- Son ficheros .css
- Es un lenguaje de diseño gráfico para crear la presentación de un documento escrito en un lenguaje de marcado.
- Cuando un navegador cliente pide una página html a un servidor web, **el servidor**, si la página html está enlazada a una hoja de estilos (css), **envía al cliente el html y el css** y el cliente mostrará la página html adaptada a la hoja de estilos recibida
- Refs:
  - <https://www.w3schools.com> aprendizaje css
  - <https://developer.mozilla.org/es/docs/Web/CSS> Estándares de la web



## Visual studio code : Editor de texto

➤ Extensiones: pulsar icono  de la barra lateral izquierda.

- HTML CSS Support. Facilita el trabajo con html y css.
- w3c web validator. Valida el código html/css contra el estándar. También en línea en <https://validator.w3.org/>
- css formatter. Formatea el código css para facilitar legibilidad (Ctrl +Alt+ F en Windows).
- Auto close tag. Para cerrar las etiquetas cuando las abres
- Auto rename tag . Ayuda con los nombres de etiquetas.
- Prettier Code Formatter Da formato a ficheros.
- Git lens para trabajar con el repositorio
- IntelliSense for CSS. Para trabajar con css

## GitHub.

Firefox ,Edge, Chrome -> Herram. desarrolladores (En Chrome CTRL +MAY +J o botón dcho ratón ->Inspeccionar )

CheatSheets: <https://web.stanford.edu/group/csp/cs21/htmlcheatsheet.pdf>

- <https://websitesetup.org/css3-cheat-sheet>

# Estructura de un documento HTML

---



Los navegadores también permiten ir inspeccionando el código conforme aparece el resultado (pulsas botón dcho en el **navegador** sobre la web y selecciona **Inspeccionar**).

La estructura principal de una página web es la siguiente:

La etiqueta primera de un documento es `<html>`, que define el inicio del fichero, por lo que `</html>` es la última que tiene que escribirse. Después, se encuentra la etiqueta `<head>`, donde se especifica la cabecera del documento y que puede anidar dentro otras como:

- `<title>`. Título que se muestra en la barra de título del explorador, en la pestaña.
- `<meta>`. Sirve para el uso de metadatos y posicionar la web en los buscadores.
- `<style>`. Se utiliza en la definición de estilos.
- `<link>`. Enlaces a otras páginas. Vincula las hojas de estilo como recurso externo.
- `<base>` para establecer una ruta por defecto para las referencias relativas que haya en la página. Ejem `<base href=http://www.jmpl.com/>`



```
<html>
  <head>
    <title>LM</title>
  </head>
  <body>
    <!-- contenido -->
  </body>
</html>
```

La cabecera no se muestra en la página web, son metadatos sobre la página web.

Por último, se aprecia la cabecera `<body>` o cuerpo, que es donde se define el contenido de una página web visible en los navegadores.

En el ejemplo anterior (estructura principal de una página web),

Se aprecia un **comentario**, entre las etiquetas `<!--` y `-->`

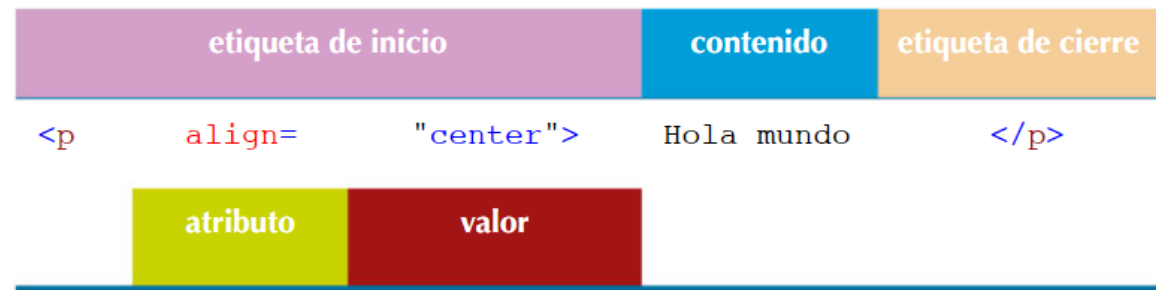
El uso de comentarios en este tipo de documentos es muy importante, ya que ayuda a organizar el código y a ocultar texto que ya no se usa.

Las páginas web son un árbol DOM (Document Object Model)

# Elementos HTML

---

La estructura básica de HTML se denomina elemento y está compuesto por dos propiedades: atributos y contenido. Por lo general, tiene una etiqueta de inicio (que puede contener un atributo con su valor) y otra de cierre. Existen algunas etiquetas sin contenidos o vacías que no necesitan de etiqueta de cierre como, por ejemplo, `<br/>`.



Algunas de las etiquetas HTML que se utilizan en una página web aparecen reflejadas en la siguiente imagen:



# Declaración de tipo de documento

---

DTD

La declaración de tipo de documento (DTD) añade reglas sintácticas que permite definir el documento con sus reglas y atributos, permitiendo de esta manera comprobar si es válido. Entre la información que puede añadirse, destaca la versión HTML. Debe declararse al principio del documento. Por ejemplo, para HTML 4.01, existen tres DTD diferentes que incluyen los siguientes campos:

1. Estricto. Solo pueden usarse etiquetas HTML 4.01 y no permite marcado de presentación.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
```

2. Transicional. Permite todos los elementos HTML, incluyendo los antiguos, desaconsejados o en desuso, salvo las etiquetas framset.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
```

3. Conjunto de marcos. Permite todos los elementos HTML, incluyendo los antiguos, desaconsejados o en desuso.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">
```

Para html5 es suficiente con poner **<!DOCTYPE html>**

Si se analiza cada uno de los campos de una declaración, se observa que consisten en:

- !DOCTYPE: se usa esta cabecera para la declaración de tipo. Es la manera que tiene el navegador de saber la versión especificada.
- HTML: informa de cuál es el elemento raíz del documento.
- PUBLIC: presenta dos posibles valores para este campo, PUBLIC o SYSTEM, donde la primera de ellas indica que la declaración de tipo de documento está disponible de manera pública.
- -//W3C//DTD HTML 4.01 Transitional//EN": es el identificador público formal, que informa de la organización y contiene información diversa separada por los caracteres de la barra doble (//).
  - -: presenta dos posibles valores ( es una organización no registrada por la ISO, mientras que + es una organización registrada por la ISO).
  - W3C: es la organización responsable.
  - DTD HTML 4.01 Transitional: indica el tipo de documento que está declarándose.



# Etiquetas y atributos HTML

---

Una etiqueta HTML es el elemento de marcado que define el comportamiento del elemento.

- Etiquetas con contenido

Son etiquetas que tiene tres partes (por este orden)

- Apertura de la etiqueta
- Contenido de la etiqueta
- Cierre de la etiqueta (mismo nombre que en apertura).

Ejem:

<etiqueta>

Contenido de la etiqueta

</etiqueta>

<b>&lt;h1&gt;</b> <b>Título</b> <b>&lt;/h1&gt;</b>	<b>&lt;p&gt;</b> <b>Texto del párrafo</b> <b>Sigue el texto</b> <b>&lt;/p&gt;</b>
--	--

- Etiquetas sin contenido

- Por ejemplo las de imágenes.

Los atributos son elementos compuestos por dos partes claramente diferenciadas (el nombre del atributo y su valor correspondiente) y se añaden en la etiqueta de inicio de un elemento HTML.

Hay atributos generales a todas las etiquetas (nombre, id,..) y otros específicos del tipo de etiqueta.

- Se representan mediante el formato clave="valor" (id= "container") con el valor entre comillas dobles.

`<a href="https://www.google.es" title="Google">Google </a>`

The diagram illustrates the structure of the HTML anchor tag `<a href="https://www.google.es" title="Google">Google </a>`. Arrows point from labels to specific parts of the tag: 'Etiqueta' points to the opening tag `<a`, 'Atributos' points to the attribute list `href="https://www.google.es" title="Google"`, and 'Cierre Etiqueta' points to the closing tag `</a>`.

- Las etiquetas y los atributos deben ir en **minúsculas**
- Debemos **cerrar** las etiquetas.
- Valor de **atributos** entre **comillas**.
- **Tabuladores** para hacer la página legible
- Utilizar comentarios con `<!-- texto del comentario -->` (2 guiones al principio y al final)
- Utilizar **rutas relativas** en lugar de rutas absolutas.

# HTML5

---

## ETIQUETAS Y ATRIBUTOS

El objetivo del consorcio W3C con la quinta versión de este lenguaje es **modernizar las páginas web** añadiendo nuevos elementos y eliminando aquellos que resultan obsoletos. La revisión cuarta de HTML no podía, entre otras cosas, validar ciertos tipos de datos en formularios sin utilizar JavaScript, tampoco permitía arrastrar objetos y soltar, no tenía etiquetas para la web semántica o web 3.0, etc.

Básicamente, las mejoras de HTML5 se pueden agrupar en cuatro grupos:

- a) Elementos multimedia, como <audio> y <video>.
- b) Elementos semánticos, como <header>, <footer>, etc.
- c) Nuevos atributos en los elementos de los formularios como email, number, date, time, range o calendar.
- d) Elementos gráficos, como <canvas>.

**Todos los navegadores actuales soportan HTML5**

## Cambios en la definición con respecto a HTML4

### ➤ DOCTYPE simplificado.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">
```

```
<!DOCTYPE html>
```

### ➤ Declaración de charset reducida.

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

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

La codificación por defecto en HTML5 es "UTF-8".

La etiqueta DOCTYPE va antes de la <html>



La etiqueta meta define características generales. Va dentro del <head>

Define el juego de caracteres UTF-8 y permite que se vean correctamente acentos, eñes, etc.: **<meta charset="UTF-8">**

Para añadir descripciones para buscadores: **<meta name="description" content="página sobre html"/>**

Para añadir palabras clave para buscadores: **<meta name="keywords" content="html, css"/>** . Ya no se suele utilizar el meta tag keywords para los buscadores.

Autor de la web: **<meta name="author" content="José Manuel Pérez"/>**

Para que no se descuadre la página y se escale a la anchura del navegador utilizado. **<meta name="viewport" content="width=device-width, initial-scale=1.0">**

Para refrescar la página cada x (30 en el ejemplo) segundos: **<meta http-equiv="refresh" content="30">** Puede no funcionar en navegadores por cuestiones de seguridad.

- `<body>` Contenido del documento

Atributos body: <https://tutorialehtml.com/es/tutoriales-html-body/>

Algunas etiquetas dentro del body:

Cabeceras desde **h1 hasta h6** con diferente tamaño `<h1>` Cabecera `</h1>`

- `<p>` texto del parrafo`</p>` Etiqueta de parrafo
- `<br/>` Forzar salto de línea
- `<hr/>` Separación de tema con una línea de separación. Atributo color. Ejemplo:  
`<hr/color=#FF0000>`

Prueba con <https://www.w3schools.com/#gsc.tab=0>

A continuación se numeran las nuevas etiquetas que añade HTML5:

- `<article>`. Permite definir un artículo (contenido independiente) dentro de un documento.
- `<aside>`. Permite definir contenido que está indirectamente relacionado con el contenido principal del documento.
- `<audio>`. Permite reproducir audio en la página web.
- `<canvas>`. Lienzo en el que se pueden dibujar gráficos mientras que se está ejecutando la página utilizando JavaScript.
- `<datalist>`. Especifica una lista de opciones predefinidas para los elementos de entrada de información.
- `<details>`. Define detalles adicionales que el usuario puede ver u ocultar.
- `<dialog>`. Define un cuadro o ventana de diálogo.
- `<embed>`. Define un contenedor para una aplicación externa no HTML.

- `<figure>`. Define contenido autocontenido.
- `<footer>`. Permite incorporar un pie a un documento (o sección).
- `<header>`. Permite incorporar una cabecera a un documento (o sección)
- `<mark>`. Permite marcar o destacar texto en el documento.
- `<meter>`. Define una medida dentro de un rango conocido.
- `<nav>`. Define enlaces de navegación.
- `<output>`. Define el resultado de un cálculo.
- `<progress>`. Permite representar el progreso de una tarea.
- `<section>`. Define una sección dentro de un documento.
- `<source>`. Define recursos para elementos multimedia, como `<video>` y `<audio>`.
- `<time>`. Define una fecha u hora.
- `<video>`. Permite visualizar un vídeo.

Lista de elementos que se pueden usar para hacer Web HTML5:

[https://developer.mozilla.org/es/docs/Web/HTML#recursos\\_clave](https://developer.mozilla.org/es/docs/Web/HTML#recursos_clave)

<https://www.w3schools.com/tags/>

<https://dev.w3.org/html5/html-author/>

# Tipos de elementos HTML

---

EN BLOQUE O EN LÍNEA

Los elementos en HTML usualmente son elementos en línea o en bloque

- "en línea". Un elemento en línea ocupa sólo el espacio delimitado por las etiquetas que definen el elemento en línea. Ocupa el espacio mínimo y se puede colocar otro elemento a su lado
- "en bloque". Un elemento en bloque ocupa todo el espacio de su elemento padre (contenedor), creando así un "bloque". Los navegadores suelen mostrar el elemento a nivel de bloque con un salto de línea antes y después del elemento. Ocupa todo el ancho disponible y no permite que otros se coloquen a su lado.

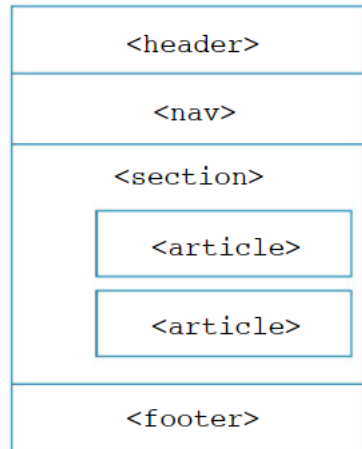


Para crear el esqueleto de una web, el elemento más importante y básico dentro del body es la etiqueta div de "division". Sirve para crear secciones o agrupar contenidos. Sus etiquetas son: <div> y </div> (ambas obligatorias). Puede contener: Texto, y/o cero o más elementos en bloque o en línea.

```
<div style="color: blue;">
  <h2> Ejemplo de div y span </h2>
  <p>
    Esto es un párrafo dentro de un div,
    <span style="color: red;"> y esto un span dentro de un párrafo.</span>
  </p>
</div>
```

## Ejemplo de div y span

Esto es un párrafo dentro de un div, y esto un span dentro de un párrafo.



Como se puede ver en la figura, se utilizarán las etiquetas `<header>` y `<footer>` para definir la cabecera y el pie de la página. La etiqueta `<nav>` permitirá crear un pequeño menú para poder acceder a cierta información y por último, se puede observar que se han incluido dos artículos (`<article>`) dentro de una sección (`<section>`) del documento. También se podría dar el caso contrario, incluir dos secciones `<section>` dentro de un elemento `<article>`.

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>LM</title>
</head>
<body>
  <header>
    <h1>Ejemplo HTML5</h1>
  </header>
  <nav>
    <a href="https://es.wikipedia.org/wiki/Superman">Wikipedia Superman </a>
    <a href="https://es.wikipedia.org/wiki/Spider-Man">Wikipedia Spiderman </a>
    <a href="https://es.wikipedia.org/wiki/Batman">Wikipedia Batman</a>
  </nav>
  <section>
    <h2>Super Héroes</h2>
    <article>
      <h3>Superman</h3>
      <p>Superman (cuyo nombre kryptoniano es Kal-El y su nombre terrestre es Clark Kent) es un superhéroe ficticio, que apareció por primera vez en los cómics estadounidenses publicados por DC Comics... Para más info consultar enlace a la wikipedia</p>
    </article>
    <article>
      <h3>Spiderman</h3>
      <p>Spider-Man (llamado Hombre Araña en muchas de las traducciones al español) es un superhéroe ficticio creado por los escritores y editores Stan Lee y Steve Ditko... Para más info consultar enlace a la wikipedia</p>
    </article>
    <article>
      <h3>Batman</h3>
      <p>Batman (conocido inicialmente como Bat-Man y en español como el Hombre Murciélago) es un personaje creado por los estadounidenses Bob Kane y Bill Finger,5 y propiedad de DC Comics... Para más info consultar enlace a la wikipedia</p>
    </article>
  </section>
  <footer>Realizado por: Manu Herrera</footer>
</body>
</html>
```

## Ejemplo HTML5

[Wikipedia Superman](https://es.wikipedia.org/wiki/Superman) | [Wikipedia Spiderman](https://es.wikipedia.org/wiki/Spider-Man) | [Wikipedia Batman](https://es.wikipedia.org/wiki/Batman)

### Super Héroes

#### Superman

Superman (cuyo nombre kryptoniano es Kal-El y su nombre terrestre es Clark Kent) es un superhéroe ficticio, que apareció por primera vez en los cómics estadounidenses publicados por DC Comics... Para más info consultar enlace a la wikipedia

#### Spiderman

Spider-Man (llamado Hombre Araña en muchas de las traducciones al español) es un superhéroe ficticio creado por los escritores y editores Stan Lee y Steve Ditko... Para más info consultar enlace a la wikipedia

#### Batman

Batman (conocido inicialmente como Bat-Man y en español como el Hombre Murciélago) es un personaje creado por los estadounidenses Bob Kane y Bill Finger,5 y propiedad de DC Comics... Para más info consultar enlace a la wikipedia

Realizado por: Manu Herrera

# Formateo de Texto

---

ETIQUETAS Y ATRIBUTOS

**Etiquetas de formato de texto.** El formato dentro de un documento es fundamental y más si se habla de ver dicho documento en un navegador web. Para ello, existe una serie de etiquetas HTML que permiten formatear el texto a gusto del usuario, dotando al documento de un diseño más atractivo y legible.

## Texto simple

Todo el contenido de su página Web se sitúa en el cuerpo del documento Html, es decir entre las etiquetas **<body> ... </body>**

## Línea horizontal

Etiqueta **<hr>**: línea horizontal (horizontal rules) que permite los atributos **align** para la alineación de la línea horizontal, tomando los valores "left" o "right", y **width**, que permite modificar el ancho.

## Salto de línea

En html, los retornos de carro o saltos de línea que aparecen en el código se ignoran y se sustituyen por un espacio. Para forzar el salto de línea es preciso usar la etiqueta **<br>**.

<b> **negrita** </b>.

<i> *cursiva* </i>

<del> ~~tachado~~ </del>

<em> *énfasis* (similar a cursiva)</em>

<small>texto pequeño</small>

<sup> texto como superíndice de otro texto </sup>.

<sub> subíndice de otro texto </sub>

<mark> texto subrayado (nuevo en html5).</mark>

<q> pequeña cita </q>

<blockquote> poner citas largas </blockquote>

<cite> título de una referencia bibliográfica </cite>

<time> mostrar horas </time>

<address> direcciones </address>

<abbr title="Señor">Sr.</abbr> para [abreviaturas](#).

Se muestra al pasar el cursor por encima.

Varias de ellas se utilizarán en css en lugar de directamente en el documento html

Si los servidores no admiten los caracteres de nuestro idioma podemos utilizar lo siguiente para poder mostrar acentos, y demás símbolos:

á	<u>&amp;aacute;</u>	Á	<u>&amp;Aacute;</u>	“	<u>&amp;quot;</u>
é	<u>&amp;eacute;</u>	É	<u>&amp;Eacute;</u>		<u>&amp;nbsp;</u>
í	<u>&amp;iacute;</u>	Í	<u>&amp;Iacute;</u>	<	<u>&amp;lt;</u>
ó	<u>&amp;oacute;</u>	Ó	<u>&amp;Oacute;</u>	>	<u>&amp;gt;</u>
ú	<u>&amp;uacute;</u>	Ú	<u>&amp;Uacute;</u>	&	<u>&amp;amp;</u>
ñ	<u>&amp;ntilde;</u>	Ñ	<u>&amp;Ntilde;</u>	€	<u>&amp;euro;</u>

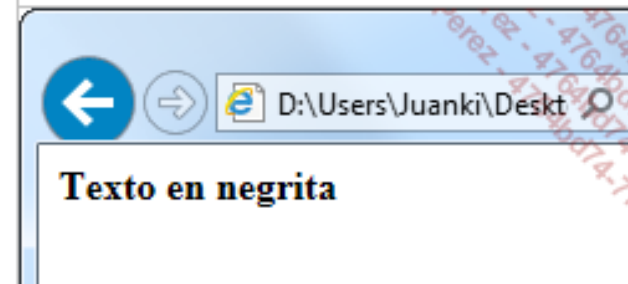
Ejemplo <p> Es un &aacute;rbol </p>



**Etiqueta <b>**: pone el texto contenido en ella en negrita (bold en ingles).

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
</head>
<body>
<b>Texto en negrita</b>
</body>
</html>
```



Del forma similar a la etiqueta `<b>`, la etiqueta `<strong> ... </strong>` permite igualmente escribir el texto en negrita en los navegadores **visuales** (como contraposición a los de síntesis vocal)

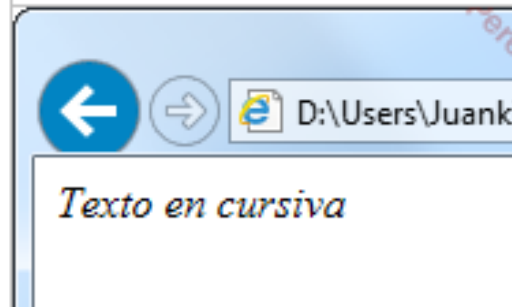
La **diferencia** entre las etiquetas `<b>` y `<strong>` es, no obstante, sensible en HTML. La etiqueta `<b>` provoca un efecto puramente tipográfico y visual. La etiqueta `<strong>` tiene como objetivo principal **reforzar** el texto.

Si bien los navegadores visuales han adoptado la misma representación que para la negrita, la interpretación es muy distinta en los programas de síntesis **vocal** que leen las páginas Web a personas con deficiencias visuales. La etiqueta `<strong>` **refuerza** el contenido mediante una entonación de voz distinta o un nivel sonoro más elevado. El sentido que se da al texto con la etiqueta `<strong>` es por tanto muy diferente.

*Cursiva.* Para escribir un texto en cursiva (italic en inglés), hay que incluirlo entre las etiquetas `<i> ... </i>`.

## Ejemplo

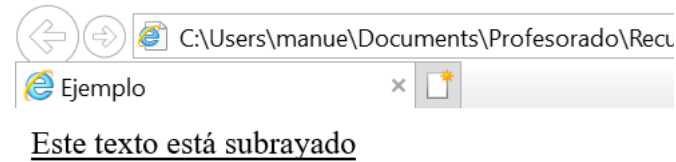
```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
</head>
<body>
<i>Texto en cursiva</i>
</body>
</html>
```



Existe también la etiqueta `<em>` (énfasis). Esta etiqueta también escribe el texto en cursiva en los navegadores. La **diferencia** con la etiqueta `<i>` está en el sentido que se da a ambas etiquetas. La etiqueta `<i>` es puramente tipográfica y visual. La etiqueta `<em>` tiene como objetivo poner de **relieve** el texto que la contiene. Este significado visual puede interpretarse de otro modo en los navegadores **no visuales**.

Subrayado. Para escribir un texto subrayado, (underline en inglés) hay que incluirlo entre las etiquetas `<u> ... </u>`.

```
1 <!DOCTYPE html>
2
3 <html lang="es">
4   <head>
5     <meta charset="UTF-8">
6     <title>Ejemplo</title>
7   </head>
8   <body>
9     <u>Este texto está subrayado</u>
10  </body>
11 </html>
```



**Etiqueta <p>**: esta etiqueta se usa para crear párrafos, siendo la de cierre opcional, siempre y cuando no se use para alineación del texto. La alineación se realiza con el atributo *align* cuyos valores pueden ser , "left", "right", "justify", "center", aunque existe otra serie de atributos genéricos menos importantes. (el atributo *align* está obsoleto en html5)

La etiqueta de apertura <p> actúa como salto de línea, similar a la etiqueta <br>

**Etiqueta <big>**: (Obsoleto en HTML5) para aumentar texto.

**Etiqueta <small>**: (Obsoleto en HTML5) para disminuir texto.

La alineación de texto se realiza utilizando estilos. Se pueden poner directamente en la etiqueta (<p>, <h1>, etc.) o en css. Los valores posibles son center, left, right o justify

En versiones anteriores de html se utilizaba *align*, pero está obsoleto en html5.

Ejemplo:

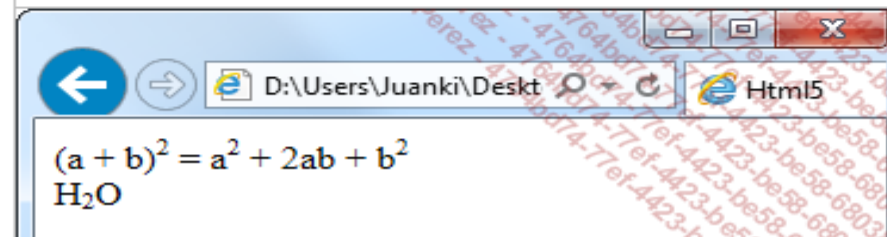
```
<p style="text-align: right;"> texto articulo1</p>
```

**Exponentes y subíndices.** En ocasiones es necesario escribir algunos caracteres como exponentes o subíndices, por ejemplo en fórmulas matemáticas o químicas.

La etiqueta `<sup> ... </sup>` escribe el contenido en exponente y la etiqueta `<sub> ... </sub>` en subíndice.

## Ejemplo

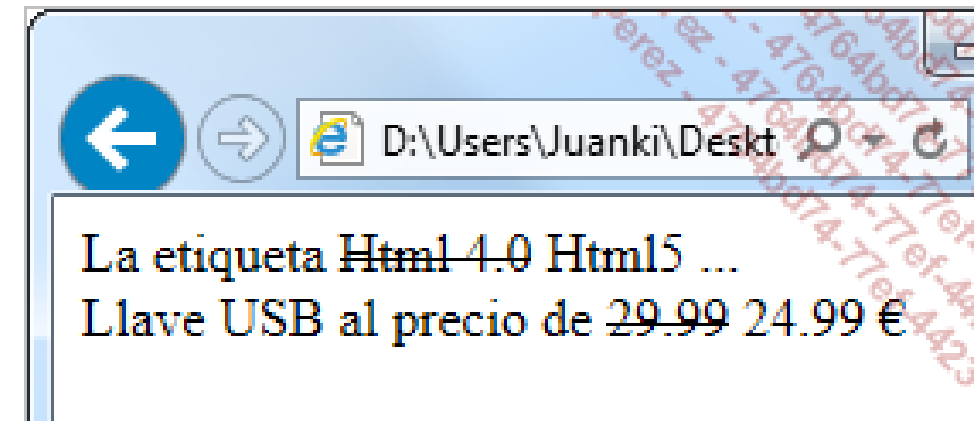
```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
</head>
<body>
(a + b)<sup>2</sup> = a<sup>2</sup> + 2ab + b<sup>2</sup><br>
H<sub>2</sub>O
</body>
</html>
```



**Texto tachado.** La etiqueta `<del> ... </del>` permite marcar un elemento de texto como suprimido (delete) o caducado, por ejemplo en la actualización de un precio en un sitio comercial. El texto aparece tachado en pantalla. Similar a la etiqueta `<strike>` (La cual ha sido obsoleta en HTML5)

## Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
</head>
<body>
La etiqueta <del>Html 4.0</del> Html5 ...<br>
Llave USB al precio de <del>29.99</del> 24.99 €
</body>
</html>
```

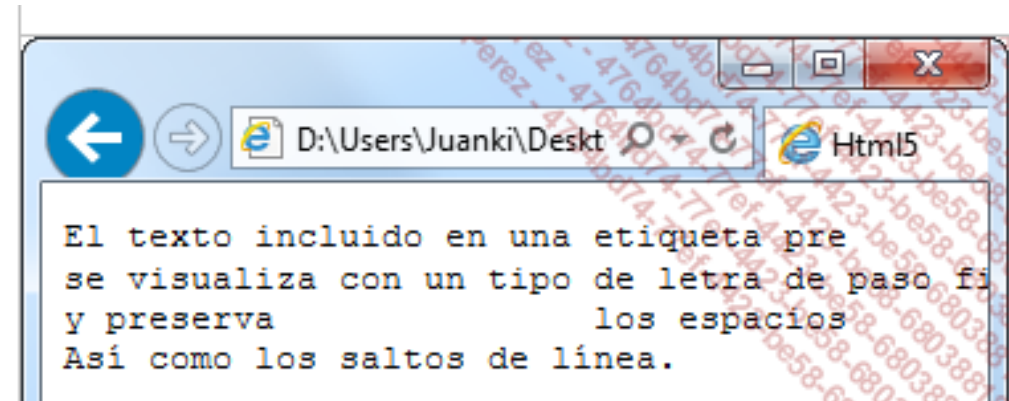




**Texto preformateado.** La etiqueta `<pre> ... </pre>` permite visualizar el texto **tal y como se ha codificado** en el editor de texto. Los espacios, tabulaciones y retornos de carro se respetan tal cual en la pantalla. Si incluye texto, se visualizará con un tipo de letra de paso fijo.

## Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
</head>
<body>
<pre>
El texto incluido en una etiqueta pre
se visualiza con un tipo de letra de paso fijo
y preserva          los espacios
Así como los saltos de línea.
</pre>
</body>
</html>
```



**Cambios de fuente.** La etiqueta `<font> ... </font>` permitía modificar el tamaño, el color y la fuente de un texto, pero no está permitió en HTML5. En su lugar hay que usar **style con los atributos entre comillas y separados por ;** `<p style="color:greenyellow; font-size:30px; font-family:fantasy;">Estilo diferente.</p>`

- **font-size** : para el tamaño de la fuente y con un valor que va del 1 al 7, donde 3 es el valor por defecto.
- **font-family** : es el que especifica la fuente del texto
- **color** : especifica el color del texto.



Texto con fuente cambiada

- Actividad: Realiza pruebas con la opción `style` para visualizar texto en diferentes formatos.

**Dirección del texto.** Algunos sistemas de escritura, tales como los alfabetos árabe y hebreo, se escriben de **derecha a izquierda** al contrario que el sentido de escritura convencional de izquierda a derecha de las lenguas que usan el alfabeto latino (como por ejemplo el castellano). Html, lenguaje universal, debe tener en cuenta de algún modo esta especificación.

La etiqueta `<bdo> ... </bdo>` indica el sentido de visualización del texto (de izquierda a derecha o de derecha a izquierda). Los atributos son:

- `dir="ltr"` (left to right) para el sentido de lectura de izquierda a derecha (por defecto).
- `dir="rtl"` (right to left) para el sentido de derecha a izquierda.

```
1 <!DOCTYPE html>
2
3 <html lang="es">
4   <head>
5     <meta charset="UTF-8">
6     <title>Ejemplo</title>
7   </head>
8   <body>
9     <bdo dir="ltr">Este texto se lee de izquierda a derecha</bdo><br/>
10    <bdo dir="rtl">Este texto se lee de derecha a izquierda</bdo>
11  </body>
12 </html>
```

← → ↺ 🏠 ⓘ Archivo | C:/

Este texto se lee de izquierda a derecha  
adreiuzi a ahcered ed eel es otzet etsE

span - abarcar. Es un contenedor en línea. Sirve para aplicar estilo al texto o agrupar elementos en línea. Sus etiquetas son: `<span>` y `</span>` (ambas obligatorias).

Al estar vinculado con el estilo, sus atributos más importantes son `class`, `id` and `style`.

```
<div style="color: blue;">
  <h2> Ejemplo de div y span </h2>
  <p>
    Esto es un párrafo dentro de un div,
    <span style="color: red;"> y esto un span dentro de un párrafo.</span>
  </p>
</div>
```

## Ejemplo de div y span

Esto es un párrafo dentro de un div, y esto un span dentro de un párrafo.

**Comentarios.** En ocasiones es útil comentar el código Html (como cualquier otro código de programación), para facilitar la comprensión a la hora de hacer alguna modificación.

Los **comentarios** están precedidos por la etiqueta `<!--` y se cierran con la etiqueta `-->`.

Todo lo que escribamos entre estas etiquetas no se visualizará en la pantalla del navegador. Los comentarios pueden ocupar varias líneas y pueden situarse de forma indistinta en el `<head> ... </head>` o en el `<body> ... </body>`.

## Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
</head>
<body>
<b><i>Texto en negrita y cursiva</i></b>
<!-- comentario particular -->
</body>
</html>
```

**Caracteres especiales.** Un poco como para las palabras reservadas en un lenguaje de programación, existe una lista de elementos a evitar en la codificación del texto. Estos elementos son, por ejemplo:

El signo < que indica la apertura de una etiqueta.

El signo > que marca el cierre de una etiqueta.

El signo " usado en los atributos de la etiqueta.

El signo & que marca el inicio de una referencia de carácter.

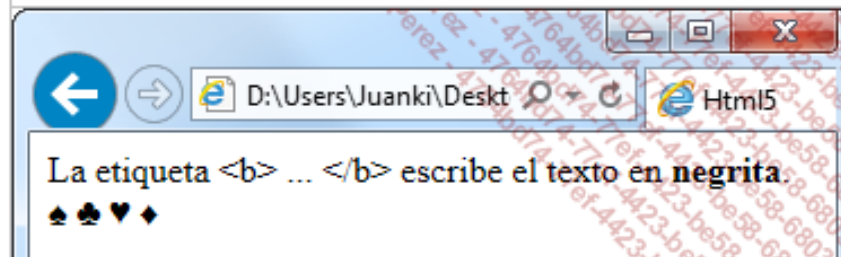
Los distintos signos deben codificarse usando las entidades siguientes.

<	&lt;
>	&gt;
"	&quot;
&	&amp;

Por otro lado, existe toda una serie de caracteres que no existen en el teclado normal como el símbolo © de copyright o el símbolo ® de marca registrada. Hay también otros caracteres específicos de las matemáticas como el signo  $\int$  para las integrales. Aquellos lectores interesados podrán encontrar toda una larga lista de caracteres especiales usados en Html en la dirección <http://www.ascii.cl/es/codigos-html.htm>

Por otro lado, existe toda una serie de caracteres que no existen en el teclado normal como el símbolo © de copyright o el símbolo ® de marca registrada. Hay también otros caracteres específicos de las matemáticas como el signo  $\int$  para las integrales. Aquellos lectores interesados podrán encontrar toda una larga lista de caracteres especiales usados en Html en la dirección <http://www.ascii.cl/es/codigos-html.htm>

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
</head>
<body>
La etiqueta <b> ... </b> escribe el texto en <b>negrita</b>.<br>
&spades; &clubs; &hearts; &diamonds;
</body>
</html>
```





## Actividad 1. Crea el siguiente texto usando HTML

Seguía la **escuela matemática**, donde el profesor enseñaba a sus discípulos por un método difícilmente comprensible en *Europa*.

Las proposiciones y demostraciones se escribían sobre una delgada oblea, con tinta compuesta por una mezcla encefálica.

El estudiante tragaba la oblea hallándose en ayunas, y en los tres días sucesivos *no debía comer ni beber nada, fuera de agua y pan*. Una vez digerida la oblea, la tintura ascendía al cerebro, llevando consigo la demostración.

Pero el éxito hasta entonces no había sido completo, en parte por error en las dosificaciones y en parte por la perversidad de los muchachos, para quienes aquella ingestión era tan nauseabunda, que generalmente ocultaban la oblea en la boca y la escupían luego, aparte de que *nunca podía persuadirseles de que guardaran la abstinencia tan larga como el método requería*.

**Jonathan Swift:**  
*Los viajes de Gulliver*

**Citas.** Las citas se escriben por delante del contenido de tipo texto mediante las etiquetas `<blockquote>` ... `</blockquote>`. También se puede utilizar `<q>`...`</q>` para textos pequeños. Ambas aparecen entre comillas dobles.

**Acronimos.** `<abbr>`.. `</abbr>` Esta etiqueta ha sido sustituida en HTML5 por `abbr`

```
Usando acrónimos <acronym title="Impuesto sobre  
el valor añadido">IVA</acronym>.<br />
```

Textos a los que *enfaticar*.

Usando acrónimos IV[Impuesto sobre el valor añadido]  
Añadiendo abreviaturas: Sr.

**Abreviatura.** `<abbr title= «texto»>`, también tiene el atributo **title**. En navegadores como IE no habrá ninguna referencia para saber que hay una leyenda. En cambio en Edge o Chrome aparecerá con un subrayado punteado.

```
Añadiendo abreviaturas: <abbr title="Señor">Sr.  
</abbr><br />
```

Usando acrónimos y A.A.  
Añadiendo abreviaturas: S Señor

Citando grandes textos: Arquitecto qui

**Actividad 2.** Crea la siguiente web

## El Ártico ha perdido el 14% de su hielo marino perenne en un solo año

a la derecha

**WASHINGTON.-** El hielo perenne del Ártico se redujo en un 14% durante el Último año, al perder **720.000 kilómetros cuadrados**, una superficie superior a la Península Ibérica, según datos de la NASA.

Según el JPL, la pérdida del hielo perenne, que debiera mantenerse durante todo el verano, fue todavía mayor y se acercó a un 50% en el momento en que ese hielo se desplazaba desde el ártico oriental hacia el oeste.

**Son Nghiem**, investigador del JPL ha declarado que:

*"Los cambios registrados en esos años en el hielo ártico son rápidos y espectaculares. De mantenerse la situación, ésta tendrá un impacto profundo en el ambiente, así como en el transporte marino y el comercio."*

# Color

---

## ETIQUETAS Y ATRIBUTOS

**Atributo color**: Atributo para asignar color usado en algunas etiquetas. La manera de asignar el color es muy diversa: en hexadecimal, mediante el nombre del color y en RGB.

Ejemplo: `<p style="background-color:rgb(125,0,255); color: yellow; ">aaaa</p>`

`<p style="background-color: #FF00FF; color: yellow; ">bbbb</p>`

**Hexadecimal**: el valor es "#rrggbb", siendo ese grupo de letras 3 agrupaciones numéricas con valores hexadecimales (0 1 2 3 4 5 6 7 8 9 A B C D E F). Los valores de baja intensidad se representan con 00, donde FF son los de mayor intensidad. El color negro se representa con el código #000000 mientras que el blanco es #FFFFFF. Ayuda en

[https://www.w3schools.com/colors/colors\\_hexadecimal.asp](https://www.w3schools.com/colors/colors_hexadecimal.asp)

- rr es la intensidad de rojo
- gg es la intensidad de verde.
- bb es la intensidad de azul.

**Actividad 3.** Asigna valores hexadecimales para los colores rojo, verde y azul.

- **Por nombre:** red, yellow, blue, brown, silver, gray, black, white, etc...
- **RGB:** son colores aditivos o primarios y están basados en la combinación de píxeles de colores rojo (RED), verde (GREEN) y azul (BLUE).
  - Los colores se codifican con un número.
  - Los valores para cada color van desde 0 a 255, por lo tanto, hay 256 valores posibles para cada valor. Con la unión de los tres colores en su máximo valor, se obtiene el color blanco (255,255,255) y, con la ausencia de todos, el negro (0,0,0).
  - Permite representar  $256 \times 256 \times 256$  combinaciones diferentes de colores, con lo que hay 16 millones de colores.
  - Debe tenerse en cuenta que puede que no todos los navegadores admitan esta codificación de colores.

**Actividad 4.** Crea una web que tenga el siguiente texto con el fondo en celeste y con el siguiente formato:

Q. La ventura va guiando nuestras cosas mejor de lo que acertáramos a desear porque ves allí, amigo *Sancho Panza*, donde se descubren treinta o ~~pocos más desaforados gigantes~~, con quien pienso hacer batalla y ~~quitarles a todos las vidas~~, con cuyos despojos comenzaremos a enriquecer, que esta es buena guerra, y es gran servicio de **Dios** quitar tan mala simiente de la faz de la tierra

S. ¿Qué gigantes?

Q. Aquellos que allí ves, de los *brazos largos*, que los suelen tener algunos de casi **dos leguas**.

S. Mire vuestra merced, que aquellos que allí se parecen no son *gigantes*, sino **molinos de viento**, y lo que en ellos parecen brazos son las aspas, que, *volteadas del viento*, hacen andar **la piedra del molino**.

# Encabezamiento

---

ETIQUETAS Y ATRIBUTOS



Estas etiquetas suelen usarse para **posicionamiento web** y añadir los títulos de las páginas web "recomendando que sean cortos debido al uso de palabras claves".

Esta etiqueta tiene 6 posibles valores, siendo 1 el de mayor tamaño y 6 el de menor, por tanto el texto de una etiqueta <h1> es mayor que el de una etiqueta <h2>.

Suelen usarse las etiquetas <h1> y <h2> para el título principal de la página y los otros tamaños menores para subtítulos.

```
<!DOCTYPE html>

<html lang="es">
  <head>
    <meta charset="UTF-8">
    <title>Ejemplo</title>
  </head>
  <body>
    <h1>Lenguaje de marcas</h1>
    <h2>Lenguaje de marcas</h2>
    <h3>Lenguaje de marcas</h3>
    <h4>Lenguaje de marcas</h4>
    <h5>Lenguaje de marcas</h5>
    <h6>Lenguaje de marcas</h6>
  </body>
</html>
```

---

## Lenguaje de marcas

### Lenguaje de marcas

#### Lenguaje de marcas

##### Lenguaje de marcas

##### Lenguaje de marcas

###### Lenguaje de marcas

**Actividad 5.** Crea la siguiente página web:

**Encabezado H1 sin atributo align**

**Encabezado H2 alineado a la izquierda**

**Encabezado h3 centrado**

**Encabezado h4 alineado a la derecha**

**Encabezado h5 alineado a la izquierda**

**Encabezado h6 alineado a la izquierda**

# Listas

---

## ETIQUETAS Y ATRIBUTOS

Existen distintos tipos de listas que pueden definirse en HTML, pero entre ellas las más importantes son las siguientes:

➤ **Etiqueta <ol>**: *listas numéricas u ordenadas*. Esta es la etiqueta que se usa para este tipo de listas, anidando dentro de ellas los elementos o líneas mediante la etiqueta **<li>**. Entre los atributos, destacan:

- **type**: se usa para especificar el estilo de numeración (1 ,a,i,l,A). Si se omite este atributo, aparecerá como ordenada numéricamente. Está obsoleto. se recomienda utilizar [list-style-type](#) dentro de style.
- **start**: se usa para indicar el primer número de la lista. Si no se indica, se entiende que empezará por el número 1. es numérico (1, 2,3... aunque la lista sea de letras.

`<ol style="list-style-type: upper-roman; " start="3">`

- Valores posibles en <https://developer.mozilla.org/en-US/docs/Web/CSS/list-style-type> o en [https://www.w3schools.com/cssref/pr\\_list-style-type.asp](https://www.w3schools.com/cssref/pr_list-style-type.asp)

- **Etiqueta <ul>**: *listas sin orden, sin numeración*. Al igual que las listas ordenadas, usa la etiqueta <li>. El atributo a destacar es el siguiente:
  - **type**: informa del tipo de viñeta que ha de usarse . Está obsoleto. Se recomienda utilizar **style** con list-style-type ("disc", "square" o "circle").

```
<body>
  <ol style="list-style-type: decimal; " start="5">
    <li>uno</li>
    <li>dos</li>
    <li>
      <ul style="list-style-type:square">
        <li>2.1</li>
        <li>2.2</li>
      </ul>
    </li>
    <li>tres</li>
  </ol>
  <ul style="list-style-image: url('iconoPequeño.JPG');">
    <li>2.2.1</li>
    <li>2.2.2</li>
  </ul>
</body>
```

5. uno  
6. dos  
7.   ▪ 2.1  
      ▪ 2.2  
8. tres

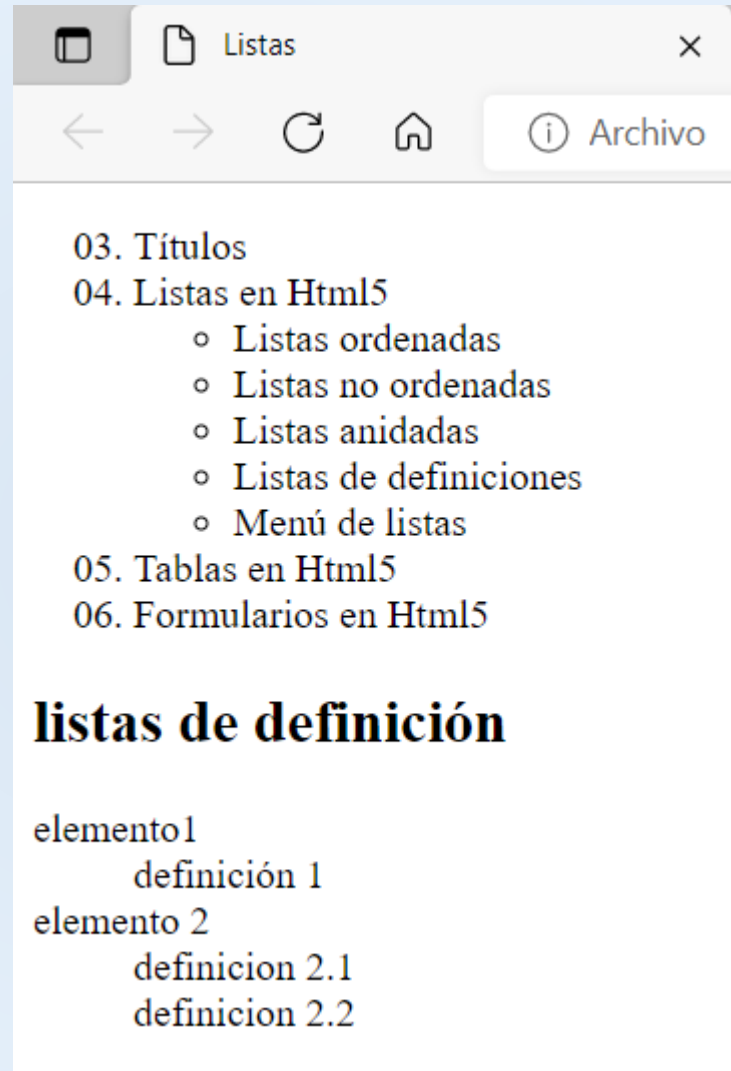
∞ 2.2.1  
∞ 2.2.2

- **Etiqueta <dl>**: *listas de definición*. Dentro de estas se usan las etiquetas:
  - **<dt>**: para cada elemento.
  - **<dd>**: para realizar la definición de cada elemento.

```
<!DOCTYPE html>

<html lang="es">
  <head>
    <meta charset="UTF-8">
    <title>Ejemplo</title>
  </head>
  <body>
    <dl>
      <dt>Capítulo 1</dt>
      <dd>Lenguajes de Marcas</dd>
      <dt>Capítulo 2</dt>
      <dd>HTML</dd>
    </dl>
  </body>
</html>
```

Capítulo 1  
Lenguajes de Marcas  
Capítulo 2  
HTML

**Actividad 6.** Crea la siguiente web:

# Enlaces

---

## ETIQUETAS Y ATRIBUTOS



Hoy en día, no se concibe una web sin hipervínculos a otras páginas o a otros apartados de esta, a la dirección de correo o a documentos. Pueden clasificarse en tres tipos: enlaces absolutos (a páginas de otras páginas web), enlaces relativos (a páginas situadas dentro del servidor local) y enlaces dentro de la misma página.

- **Etiqueta <a>.** Anchor (ancla) tiene una de apertura y otra de cierre. Entre dichas etiquetas, puede insertarse un elemento como texto, imagen u otro objeto. Entre los atributos que posee, destacan:
  - **href** (hypertext reference): el valor que se le asigna suele ser una URL.
  - **name**: se usa para definir una ubicación dentro del documento. También puedes utilizar id, pero id es único y name se utiliza, a veces, para identificación de grupo como en los input radio.
  - **target**: se usa para indicar dónde desplegar la URL. Los valores posibles son:
    - ❖ **\_blank**: se usa para abrir el contenido en una nueva ventana o pestaña
    - ❖ **\_self**: es el valor por defecto. Se abre la URL en el mismo lugar donde se ha pulsado el enlace.
    - ❖ **\_parent**: la URL se carga en el contexto padre o marco asociado.
    - ❖ **\_top**: la URL se carga en el contexto más alto y ocupa toda la ventana
  - **download** : descarga el documento en lugar de mostrarlo `<a href="/docs/doc.pdf" download>`

## Enlaces absolutos:

- Son los que se usan para poner una URL completa y deben incluir el protocolo `http://` acompañado del dominio. Por lo general, las URL pertenecen a otros servidores. Hay que poner `http://` antes de la dirección  
`<a href="http://www.google.es">Google</a>`

## Enlaces relativos:

- Se usan para acceder a URL alojadas en el mismo servidor, ya sea dentro o fuera del directorio donde se encuentra el enlace. La manera de actuar cuando quiere accederse a otra página dentro del mismo directorio es poner el nombre de la página junto a la extensión como valor del atributo `href`. Si quiere accederse a una URL que está dentro de un subdirectorio, se tendrá que indicar el nombre de este junto a la URL. Otro aspecto que puede darse es estar situado en un subdirectorio y querer acceder a una página del directorio padre. En ese caso, hay que poner dos puntos seguidos `..` por cada directorio que quiera subirse.  
`<a href="autores.html">Autores</a>`  
`<a href="contenido/tema.html">Autores</a>`  
`<a href="../librería/autores.html">Autores</a>`

Si se añade la **etiqueta base (sólo puede aparecer una vez)** en el **head** las referencias se tomarán desde el directorio indicado en base y no desde el actual `<base href=D:\ies\Asignaturas\lenguajeMarcas\.>`

**Dentro de la misma página.** Al disponer de páginas con mucho contenido, puede ser interesante crear un pequeño índice que acceda al contenido de ese apartado.

Lo primero es **crear los marcadores con name o id**, que serán las zonas de la página a las que pretende accederse. Para ello, se usa el **atributo name (o id)** de la etiqueta <a> (u otras) con el valor del marcador que se desee. El nombre de la etiqueta se toma como case-sensitive.

El siguiente paso es añadir **el enlace que lleve a un marcador**. Para ello, dentro del atributo **href**, se pondrá el carácter **almohadilla (#)** seguido del nombre del marcador.

```
<a href="#Becquer">Gustavo Adolfo Becquer</a>
```

...

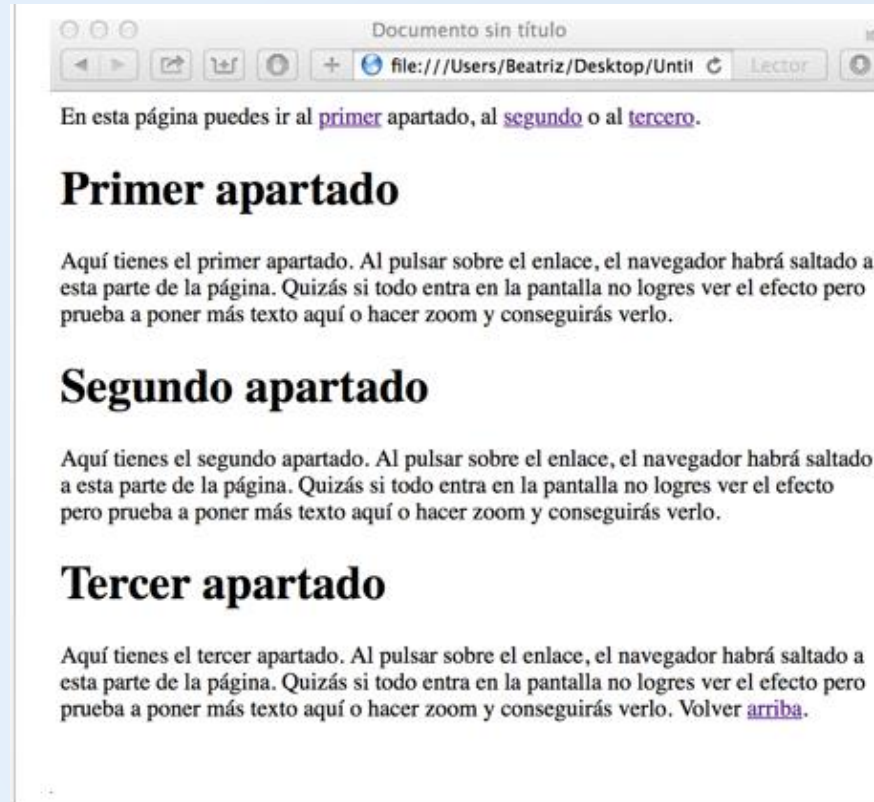
```
<a name="Becquer">Gustavo</a>
```

También se pueden poner la referencia a la que ir con otras etiquetas:

```
<p id="lugarParalr"> algo </p>
```

Si se pone la etiqueta base los enlaces relativos no funcionarán en la página local. Interpretará que son referidos a lo indicado en base.

## Actividad 7. Crea la siguiente web:



## Actividad 8. Crea la siguiente web:

- Buscadores
  - [Bing](#)
  - [Google](#)
- Redes sociales
  - [Facebook](#)
  - [Twitter](#)

# Imágenes

---

ETIQUETAS Y ATRIBUTOS

La etiqueta **<img />** se usa para añadir imágenes en una página y no tiene etiqueta de cierre. Por lo general, la imagen puede situarse en la misma carpeta donde se encuentra el documento HTML o situar todas las imágenes en una carpeta concreta y acceder a ella mediante rutas absolutas o relativas.

La etiqueta **<img>** es un elemento en línea, pero cuando especificamos el ancho y el alto, se convierte en un elemento de bloque.

Las extensiones de las imágenes que pueden usarse son .jpg, .bmp, .png, .gif, etc. Por defecto, siempre aparecen en la parte superior izquierda de la página web. Los atributos más importantes son los siguientes:

- **src:** se la ruta con el nombre de la imagen. **Atributo obligatorio.**
- **width:** se establece el ancho de la imagen. Mejor ponerlo en style
- **height:** establece el alto de la imagen. Mejor ponerlo en style
- **alt:** se usa para dar una breve descripción de la imagen por si esta no se puede representar. **Se debe poner siempre.**
- **title:** Texto explicativo

- **border:** se le asigna un borde con valor numérico. En html5 se debe usar dentro de style: `style="border:1px"`
- **align:** para la alineación de la imagen con valores top, middle, bottom, left y right . **No debe usarse en html5.** En su lugar hay que utilizar estilos.
  - Para la alineación vertical **vertical-align:** baseline|length|sub|super|top|text-top|middle|bottom|text-bottom|initial|inherit
  - Para la alineación horizontal margin-top | margin-bottom | margin-left | margin-right con los valores:
    - auto - margen calculado por el navegador
    - length - tamaño del margen en px, pt, cm, etc.
    - % - % sobre el ancho del elemento contenido
    - inherit - se hereda el margen del padre.

```

```



La etiqueta **<figure>** permite agregar contenido variado asociadas a una descripción o título. Este título es asociado semánticamente a la imagen de tal forma que podemos agregar una imagen, ilustración, diagrama, o incluso código. Como regla general, el contenido que agreguemos debe de estar relacionado al texto principal de la página. Su eliminación o cambio de posición no afectará al flujo principal del documento. Es un elemento en línea.

(<https://www.oscarblancarteblog.com/2016/11/28/etiqueta-figure-html5/>):

<figure> es la etiqueta padre.

<img ../> es una etiqueta para la imagen

<figcaption> es una etiqueta con el texto asociado a la imagen.

Ejemplo:

<figure>



<figcaption>Logotipo de josema con la etiqueta figure

</figcaption>

</figure>

### Actividad 9. Crea la siguiente web:

#### Patatas fritas

Receta de patatas fritas caseras.



Patatas FRITAS con ajos.

#### Burbujas

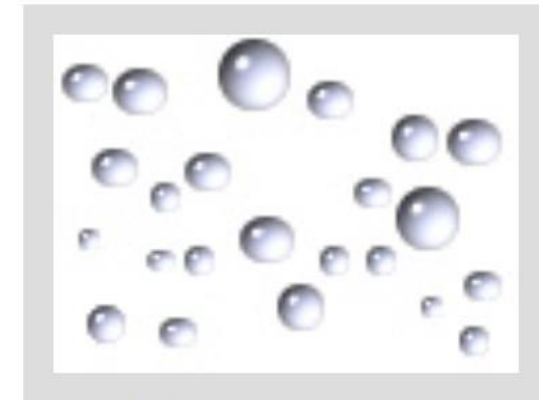


Imagen de burbujas

# Tablas

---

## ETIQUETAS Y ATRIBUTOS

Etiqueta **<table>**: Sirve para crear una tabla.

- Etiqueta **<caption>**: para añadir un título a la tabla.
- Etiqueta **<th>**: celda de cabecera en la primera fila.
- Etiqueta **<tr>**: se usa para crear filas.
- Etiqueta **<td>**: se sitúa anidada dentro de una etiqueta **<tr>** y se usa para asignar columnas o celdas dentro de una fila. La **etiqueta td tiene los** siguientes **atributos**:
  - **colspan=x**: combina columnas, siendo x el número de columnas por incluir.
  - **rowspan=x**: combina filas, siendo x el número de filas por incluir.
  - **text-align**: alineación del contenido en horizontal con los valores ("left", "right" y "center"). Usar dentro de style
  - **valign**: alineación vertical del contenido con valores ("bottom", "top" y "middle"), siendo esta última la que se usa por defecto. Usar vertical-align dentro de style

- Atributos de una tabla:
  - **border**: se añade un borde a la tabla. En html5 utilizar **style="border:1px solid"**
  - **cellpadding**: para separar el borde del contenido. El valor que se le asigna es en píxeles  
En html5 utilizar padding dentro de style.
  - **bgcolor**: puede añadirse a tablas, filas o columnas.  
En html5 utilizar background-color en style

Ejemplo

Nombre	Apellidos
Daniel	Rojo Arcenegui
José	Saramgo Pérez
Luis	Andrade Camacho

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8"/>
    <title>Lenguaje de Marcas</title>
  </head>
  <body>
    <header>
      <h1>Table</h1>
    </header>
    <table border="1">
      <caption>Ejemplo</caption>
      <th>Nombre</th>
      <th>Apellidos</th>
      <tr>
        <td>Daniel</td>
        <td>Rojo Arcenegui</td>
      </tr>
      <tr>
        <td>José</td>
        <td>Saramago Pérez</td>
      </tr>
      <tr>
        <td>Luis</td>
        <td>Andrade Camacho</td>
      </tr>
    </table>
  </body>
</html>
```

**Actividad 10.** Crea la siguiente web:

## PAISES

**Climas de América del Sur**

Parte de arriba de América del Sur.  
Países como:

Venezuela  
Colombia  
Ecuador  
Perú

Parte de abajo de América del Sur. Países como:

Argentina  
Chile  
Uruguay  
Paraguay

Bosque tropical, clima de sabana,  
clima marítimo con inviernos secos.

Climas marítimos con veranos secos, con  
inviernos secos, climas fríos, clima de estepa,  
clima desértico.

# Formularios

---

ETIQUETAS Y ATRIBUTOS

Son uno de los elementos principales que se introducen en un documento HTML. Es la manera que tienen los usuarios de interactuar con la página web mediante una serie de elementos (botones, campos de texto, casillas de verificación, etc.), y su misión es la de obtener la información que los usuarios proporcionan a la hora de utilizar algunos de sus elementos.

Aunque, normalmente, al emplear formularios se trabaja en una arquitectura cliente servidor, donde los datos obtenidos a través del formulario se envían a un servidor para ser procesados y devolver en algunos casos la información tratada, puede que se usen de manera local, aunque es lo menos habitual. Entre los formularios que pueden crearse, destacan:

- Registro de usuarios.
- Consultas.
- Libro de visitas.
- Pasarela de pago.
- Motor de búsqueda.
- Recordar contraseña.
- Identificación de usuarios.



La estructura está compuesta por el elemento `<form>`, que actúa como contenedor de distintos controles que permiten interactuar con el usuario.

- Elemento `<form>`: tiene una serie de atributos que se detallan a continuación:
  - [https://www.w3schools.com/html/html\\_forms\\_attributes.asp](https://www.w3schools.com/html/html_forms_attributes.asp)
  - [https://developer.mozilla.org/es/docs/Learn/Forms/Your\\_first\\_form](https://developer.mozilla.org/es/docs/Learn/Forms/Your_first_form)

```
<html>
<head>
  <title>LM</title>
</head>
<body>
  <form>
    <!-- controles de formulario-->
  </form>
</body>
</html>
```

Atributos	Descripción
<b>action</b>	Indica la acción o URI del programa que procesa la información.
<b>method</b>	Determina el método HTTP usado por el navegador y tiene dos posibles valores: <ul style="list-style-type: none"><li>• <i>Post</i>: los datos se incluyen en el cuerpo del formulario.</li><li>• <i>Get</i>: los datos se adjuntan en la URI después del carácter separador ?</li></ul>
<b>enctype</b>	= <code>"TEXT/PLAIN"</code> indica que los datos se envían sin codificación.
<b>target</b>	Es una palabra clave que informa de dónde se mostrarán las respuestas una vez procesados los datos del formulario. Algunos valores: <code>_blank</code> , <code>_parent</code> , <code>_top</code> , etc.

- Elemento `<input>`: se usa para la creación de distintos tipos de controles que interactúan con el usuario. Dependiendo del valor del atributo `type`, aparecerá un control u otro.
- Ejemplos:
  - `<input type="text">`
  - `<input type="password">`
  - `<input type="submit">`
  - `<input type="reset">`
  - `<input type="radio">`
- [https://www.w3schools.com/html/html\\_form\\_input\\_types.asp](https://www.w3schools.com/html/html_form_input_types.asp)
- <https://developer.mozilla.org/es/docs/Web/HTML/Element/input>

El Elemento HTML <label> representa una etiqueta para un elemento en una interfaz de usuario. Este puede estar asociado con un control utilizando el atributo for que tiene que coincidir con el id del input asociado.

El elemento label se usa generalmente para poner una etiqueta a un campo de un formulario, y haciendo uso del atributo for, se puede relacionar con dicho campo del formulario.

```
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
  </head>
  <body>
    <form>
      <label for="nombre">Nombre:</label>
      <input id="nombre" type="text" name="nombre">
    </form>
  </body>
</html>
```

Nombre:

Permiten introducir información a través de unos campos de texto de una o varias líneas. Los que se estudiarán en este apartado son:

**Texto de una sola línea:** aplicando el valor "**text**" al atributo type del elemento <input>, muestra un campo de texto que permite introducir una sola línea.

**Texto de múltiples líneas:** de características similares al anterior, pero permite añadir más líneas. Puede ser interesante usarlo cuando se pide una descripción, opinión personal o un campo que vaya a ocupar más caracteres de los esperados. Para añadir un control de este tipo, se usa el elemento <**textarea**>, junto al número de filas y columnas que deben aparecer en el control.

**Texto oculto:** aplicando el valor "**password**" al atributo type del elemento <input>, muestra un campo que sirve para ocultar la información y es de utilidad para añadir la contraseña.

**Subir archivos:** puede considerarse de esta categoría y se usa como examinador para subir archivos. Aplicando el valor "**file**" al atributo type del elemento <input>, muestra un campo que sirve para subir documentos al servidor.

La sintaxis `<input type="text" />` Los posibles atributos son:

**name:** Define un nombre (name) único para este elemento. **Este atributo se utiliza para recoger el contenido del campo de texto cuando se envía el formulario.**

**size:** Define el número de caracteres visibles del campo de texto y, por tanto, la longitud de la zona de texto. El usuario puede, no obstante, introducir tantos caracteres como quiera, incluso aunque sobrepasen la zona visible. El valor por defecto de este campo es 20.

**maxlength:** Determina el número máximo de caracteres que el usuario puede introducir en el campo de texto. Este atributo es particularmente útil para datos con un número concreto de caracteres como, por ejemplo, cinco cifras para introducir un código postal español.

**value:** Define el valor por defecto del campo de texto. Aparece como valor del campo de texto cuando se carga la página. Por ejemplo: `<input type="text" value="¡Escriba su nombre aquí!">`

**readonly:** Indica que el usuario no puede modificar el valor por defecto del campo de texto.

**placeholder:** Permite incluir una sugerencia acerca del valor que se debe introducir en el campo de texto. Esta sugerencia aparece de color gris en el campo de formulario tras cargar la página y desaparece cuando el usuario sitúa el foco en el elemento afectado.

**autocomplete:** especifica si un formulario o campo de entrada debe tener autocompletado activado o desactivado. Cuando la función de autocompletar está activada, el navegador completa automáticamente los valores de entrada en función de los valores que el usuario ha ingresado anteriormente. Sugerencia: es posible tener el autocompletado "activado" para el formulario y "desactivado" para campos de entrada específicos, o viceversa. El atributo autocomplete funciona con `<form>` y para los siguientes type `<input>`: texto, búsqueda, url, tel, correo electrónico, contraseña, fechadores, rango y color.

**autofocus.** Sitúa el foco sobre el elemento una vez cargada la página.

**required.** Convierte un elemento en obligatorio a la hora de enviar el formulario para su procesamiento posterior. Es muy útil para definir aquellos elementos obligatorios como el nombre o la dirección de correo electrónico del usuario

**pattern.** Define una expresión regular JavaScript que se usará para realizar la validación del valor del campo. Por ejemplo `pattern="[0-9]"` indica que el valor del elemento debe ser un número entero comprendido entre 0 y 9. Además hay que ponerlo dentro de un **form** y utilizar un **input type="submit"**, si pones un `button type="submit"` no funcionará

Tomemos como ejemplo un campo de texto que va a contener un código postal (5 dígitos).

```
<form action="">  
Código postal : <input type="text" pattern="[0-9]{5}"><br>  
<input type="submit">  
</form>
```

Cambiar el formato del mensaje de error: [HTML5 Validación de Formularios con el atributo "pattern" \(tutsplus.com\)](https://www.tutsplus.com/es/html5-validation-forms-with-the-pattern-attribute)

Usar el atributo **title** para poner otro mensaje:

```
<form action="somefile.php">  
<input type="text" name="username" placeholder="Username" pattern="[a-z]{1,15}" title="Solo se  
admiten letras en minúscula y entre 1 y 15 caracteres.">  
<input type="submit">  
</form>
```

Una **expresión regular** define un patrón de búsqueda/comprobación para cadenas de caracteres.

- La podemos utilizar para comprobar si una cadena contiene o coincide con el patrón.

El contenido de la cadena de caracteres puede coincidir con el patrón 0, 1 o más veces.

- Algunos ejemplos de uso de expresiones regulares pueden ser:
- para comprobar que la fecha leída cumple el patrón dd/mm/aaaa
- para comprobar que un NIF está formado por 8 cifras, un guión y una letra
- para comprobar que una dirección de correo electrónico es una dirección válida.
- para comprobar que una contraseña cumple unas determinadas condiciones.
- Para comprobar que una URL es válida.

El patrón se busca en el String de izquierda a derecha. Cuando se determina que un carácter cumple con el patrón este carácter ya no vuelve a intervenir en la comprobación

Puedes validarlas con

- <https://www.site24x7.com/es/tools/analizador-de-regex.html>
- [Free Online HTML Formatter - FreeFormatter.com](https://www.freeformatter.com/html-formatter.html)
- <https://www.html5pattern.com>
- [regex101: build, test, and debug regex](https://regex101.com)

[Lenguaje de expresiones regulares - Referencia rápida | Microsoft Learn](#)  
[JavaScript RegExp Object \(w3schools.com\)](#)



## Símbolos comunes en expresiones regulares

- `.`: Un punto indica cualquier carácter
- `^expresión`: El símbolo `^` indica el principio del String. En este caso el String debe contener la expresión al principio.
- `expresión$`: El símbolo `$` indica el final del String. En este caso el String debe contener la expresión al final.
- `[abc]`: Los corchetes representan una definición de conjunto. En este ejemplo el String debe contener las letras a ó b ó c.
- `(abc)` similar a `[ ]`, pero conserva significado de caracteres especiales.
- `[abc][12]`: El String debe contener las letras a ó b ó c seguidas de 1 ó 2
- `[^abc]`: El símbolo `^` dentro de los corchetes indica negación. En este caso el String debe contener cualquier carácter excepto a ó b ó c.
- `[a-z1-9]`: Rango. Indica las letras minúsculas desde la a hasta la z (ambas incluidas) y los dígitos desde el 1 hasta el 9 (ambos incluidos)
- `A|B`: El carácter `|` es un OR. A ó B
- `AB`: Concatenación. A seguida de B

## Meta caracteres

- \d: Dígito. Equivale a [0-9]
- \D: No dígito. Equivale a [^0-9]
- \s: Espacio en blanco. Equivale a [\t\n\r\f]
- \S: No espacio en blanco. Equivale a [^\s]
- \w: Una letra mayúscula o minúscula, un dígito o el carácter \_ (guión bajo)
- \_: Equivale a [a-zA-Z0-9\_]
- \W: Equivale a [^\w]
- \b: Límite de una palabra.

## Cuantificadores

- {X}: Indica que lo que va justo antes de las llaves se repite X veces
- {X,Y}: Indica que lo que va justo antes de las llaves se repite mínimo X veces y máximo Y veces. También podemos poner {X,} (o {X} en html5) indicando que se repite un mínimo de X veces sin límite máximo.
- \*: Indica 0 ó más veces. Equivale a {0,}
- +: Indica 1 ó más veces. Equivale a {1,}
- ?: Indica 0 ó 1 veces. Equivale a {0,1}
- !?: Indica negación

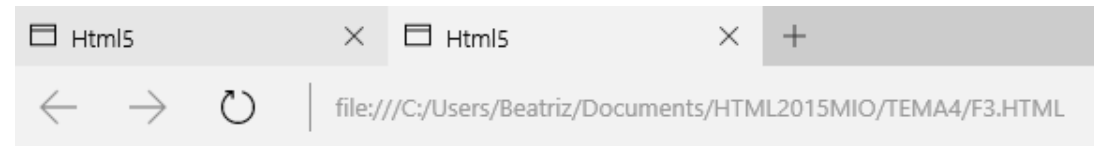
## Actividad Expr Regulares:

1. Contiene la cadena **axxa**
2. La cadena empieza por **axxa**
3. La cadena empieza por **a**(mayúscula o minúscula) seguida de **bc** y después cualquier cosa
4. La cadena tiene letras mayúsculas o minúsculas de una longitud entre 5 y 10 caracteres ambos incluidos.
5. La cadena no empieza por un dígito.
6. La cadena no acaba con un dígito
7. La cadena solo tiene aes o bs en minúscula, ejemplo abc no es válida, aabbab es válida.
8. Número de matrícula : Debe empezar por GS o GM y después 4 números.
9. DNI: 8 dígitos y luego una letra de entre las admitidas para DNIs (buscar en la web cuáles son).

En ciertas situaciones, es preciso prever algo más de espacio para que el usuario pueda expresarse. Es el caso, por ejemplo, de los comentarios o sugerencias. Se utiliza en este caso la etiqueta

`<textarea> ... </textarea>`

que incluye un área de texto de varias líneas.



**Comentarios:**

A large, empty rectangular box with a thin black border, intended for multi-line text input, such as comments or suggestions.

Los posibles atributos son:

**name.** Define un nombre para el elemento.

**cols.** Define el número de caracteres visibles en anchura para el área de texto. Puede reemplazarse por la propiedad CSS width.

**rows.** Define el número de líneas visibles en altura para el área de texto. Puede reemplazarse por la propiedad CSS height.

**readonly.** Indica que el valor informado por defecto en el área de texto no puede modificarse.

**autofocus.** Sitúa el foco sobre el elemento una vez cargada la página.

**maxlength.** Determina el número máximo de caracteres que el usuario puede introducir en el área de texto.

**required.** Hace que sea obligatorio informar este elemento.

**wrap.** Especifica la forma en que se gestionan los saltos automáticos de línea en el texto a la hora de enviar el contenido. Con wrap="hard", se inserta un carácter de salto de línea junto con el texto. Con wrap="soft" (por defecto), no se inserta ningún carácter de salto de línea a la hora de transmitir la información.

**placeholder.** Inserta un texto en el campo del formulario tras la carga de la página. Este texto se borra automáticamente cuando el usuario sitúa el foco en el área de texto.

```
<!DOCTYPE html>

<html lang="es">
  <head>
    <meta charset="UTF-8">
    <title>Ejemplo</title>
  </head>
  <body>
    <form action="">
      <p>Usuario: <input type="text" name="usuario" size="45" maxlength="40"/></p>
      <p>Password: <input type="password" name="clave" size="45" maxlength="40"/></p>
      <p>Comentario: <textarea name="mensaje" rows="5" cols="50"></textarea></p>
      <p>Subir fichero: <input type="file" name="fichero" size="60"/></p>
    </form>
  </body>
</html>
```

Usuario:

Password:

Comentario:

Subir fichero:  Ningún archivo seleccionado

Dependiendo del control que se use, podrá seleccionarse una o varias de las opciones que se presentan:

- **Botón de opciones:** aplicando el valor "**radio**" al atributo type del elemento **<input>**, muestra un campo que sirve para **seleccionar una de las opciones** que aparecen en el control. Necesitan ser agrupados para que funcionen correctamente añadiendo el mismo valor en el atributo name. En el atributo value, se añade la opción seleccionada, que debe ser diferente en cada grupo. Mediante CHECKED, se selecciona uno de ellos.
- **Control de selección o verificación:** aplicando el valor "**checkbox**" al atributo type del elemento **<input>**, muestra un campo que sirve para **seleccionar una o varias opciones** que aparecen en el control y, mediante CHECKED, se activa por defecto. El atributo value es opcional
- **Combo o listas despleables:** es similar a los estudiados con anterioridad, pero con la peculiaridad de que, para ver las distintas opciones, hay que pulsar sobre el control y **seleccionar solo una de las opciones que aparecen (salvo configuración con multiple)**. Se usa el elemento **<select>** anidando cada valor que ha de elegirse dentro del elemento **<option>**.

```
<!DOCTYPE html>
<html>
<head>
  <title>LM</title>
</head>
<body>
  <form action="">
    <p>
      <b>Ejemplo de botón de opciones</b><br>
      Selecciona un producto...<br>
      <input type="radio" name="productos" value="aguacate"/> El aguacate de la Axarquía<br>
      <input type="radio" name="productos" value="platano"/> El platano de Canarias<br>
      <input type="radio" name="productos" value="chirimolla"/> La chirimolla de Motril<br>
      <input type="radio" name="productos" value="fresa"/> La fresa de Lepe<br>
    </p>
    <br>
    <p>
      <b>Control de selección o verificación</b><br>
      Seleccione sus productos favoritos...<br>
      <input type="checkbox" name="a1" value="aguacate"/> El aguacate<br>
      <input type="checkbox" name="a2" value="platano" checked/> El aguacate<br>
      <input type="checkbox" name="a3" value="chirimollo"/> El aguacate<br>
      <input type="checkbox" name="a4" value="fresa" checked/> La fresa de Lepe<br>
    </p>
    <br>
    <p>
      <b>Ejemplo de listas</b><br>
      Selecciona una opcion...<br>
      <select name="Color">
        <option>aguacate</option>
        <option>platano</option>
        <option>chirimollo</option>
        <option>fresa</option>
      </select>
    </p>
  </form>
</body>
</html>
```

## Ejemplo de botón de opciones

Selecciona un producto...

- ☐ El aguacate de la Axarquía
- ☐ El platano de Canarias
- ☐ La chirimolla de Motril
- ☐ La fresa de Lepe

## Control de selección o verificación

Seleccione sus productos favoritos...

- ☐ El aguacate
- ☒ El aguacate
- ☐ El aguacate
- ☒ La fresa de Lepe

## Ejemplo de listas

Selecciona una opcion...

aguacate ▼



La etiqueta `<select> ... </select>` indica al navegador el uso de una lista desplegable. Los elementos de la lista se introducen mediante las etiquetas `<option> ... </option>`.

Los posibles atributos son:

**name.** Define un nombre para la lista desplegable, para un posible procesamiento posterior.

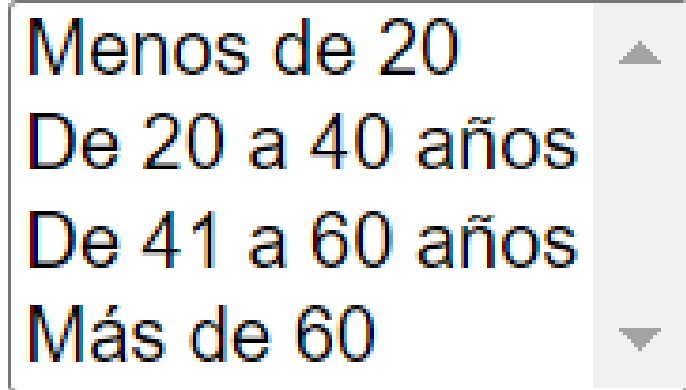
**size.** Por defecto, el atributo `size` de la etiqueta `<select>` es igual a 1. No obstante, mediante el atributo `size=«x»` puede definir el número de elementos del menú que será visible

**multiple.** Por defecto, el usuario no puede **seleccionar** más de un elemento del menú desplegable. Con el atributo múltiple de la etiqueta `<select>`, es posible seleccionar varios elementos a la vez. Para hacer esto, el usuario debe mantener pulsada la tecla [Ctrl] en el teclado y hacer clic sobre los elementos con el ratón.

**selected.** Por defecto aparece **seleccionado** el primer elemento de la lista. No obstante es posible preseleccionar algún otro elemento mediante el atributo `selected` de la etiqueta `<option>`.

**value.** En principio es el texto del elemento seleccionado, situado tras la etiqueta `<option>`, el que se transmite a la hora de enviar el formulario. No obstante es posible especificar un valor diferente (generalmente un valor numérico) para transmitirlo, mediante el atributo `value="valor"`.

**Actividad 11.** Crea la siguiente web:



Menos de 20  
De 20 a 40 años  
De 41 a 60 años  
Más de 60

Los botones de selección única, también llamados botones radio, tienen la particularidad de que sólo permiten seleccionar una opción a la vez

La etiqueta `<input>` no tiene etiqueta de cierre. La sintaxis `<input type = "radio"/>`

Los posibles atributos son:

**name.** Aquí el atributo name es obligatorio. Además, en el caso de los botones radio, el **nombre** debe ser **idéntico** para todos los botones.

**checked.** Permite preseleccionar un botón radio.

**value.** En vista de un procesamiento posterior, es posible atribuir un valor a cada botón radio mediante el atributo `value="valor"`.

## Actividad 12. Crea la siguiente web:

Nombre:

Jose

descripción

Sexo > Masculino ☐ Femenino ☐

CP:

Enviar

El uso de estos botones de selección múltiple, también llamados checkbox, es similar a los botones de selección única aunque, en este caso, es posible seleccionar varias opciones. La etiqueta `<input>` no tiene etiqueta de cierre. La sintaxis `<input type="checkbox" />`,

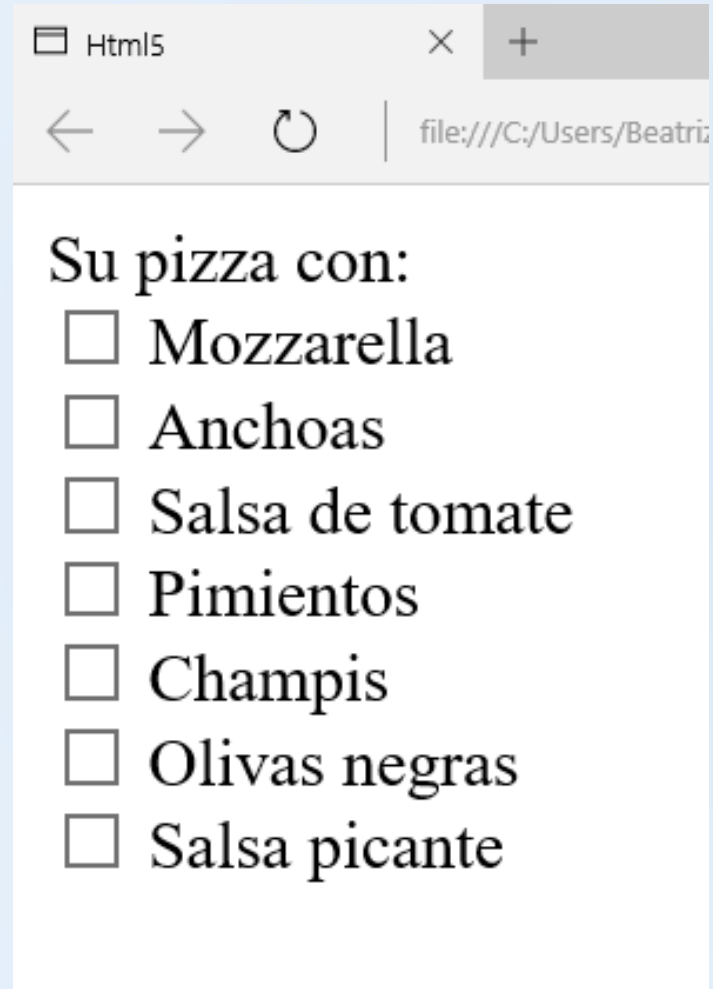
Los posibles atributos son:

**name.** El atributo `name="nombre"` es obligatorio.

**checked.** El atributo `checked` permite preseleccionar una casilla.

**value.** Para poder procesarlos posteriormente es preciso indicar un valor a cada botón checkbox mediante el atributo `value="valor"`.

## Actividad 13. Crea la siguiente web:



A screenshot of a web browser window. The title bar shows 'Html5' and a close button. The address bar shows 'file:///C:/Users/Beatri...'. The main content area displays a form titled 'Su pizza con:' followed by a list of seven items, each with an unchecked checkbox:

- ☐ Mozzarella
- ☐ Anchoas
- ☐ Salsa de tomate
- ☐ Pimientos
- ☐ Champis
- ☐ Olivas negras
- ☐ Salsa picante

Gracias a este tipo de controles, el usuario puede interactuar con el servidor, ya que, al presionarlo, se ejecuta un evento o acción. En este apartado, se verán los controles de botones más usados:

- **Botón de envío:** es uno de los controles que no suele faltar a la hora de crear un formulario. Aplicando el valor "***submit***" al atributo *type* del elemento `<input>`, muestra un botón que, al pulsarlo, envía el formulario. Otro atributo que hay que cumplimentar es *value*, cuyo valor será el texto del botón. Se puede añadir una acción adicional a la indicada en el formulario utilizando el atributo *formaction*.
- **Botón de reinicio o borrado:** aplicando el valor "***reset***" al atributo *type* del elemento `<input>`, muestra un botón que, al pulsarlo, "*limpia o restaura a valor inicial*" los valores

```
<p><input type="submit" value="Enviar">&nbsp;<input type="reset" value="Limpiar"></p>
```

Enviar Limpiar

&nbsp; espacio en blanco que no se puede omitir ni romper (por ejemplo, con salto de línea)

**Botón de acción:** `<button> ... </button>`. Esta etiqueta permite ejecutar, al hacer clic sobre el botón, una **acción** específica definida por el diseñador del sitio Web, generalmente mediante algún código de JavaScript.

Por otro lado, esta etiqueta `<button>` tiene la ventaja de tener una etiqueta de apertura y otra de cierre. De este modo es posible incluir texto, imágenes o incluso contenido Html. Destaquemos también que esta etiqueta `<button>` **no** tiene por qué estar anidada obligatoriamente en un **formulario** `<form>` y puede usarse en múltiples contextos. El texto del botón se declara entre ambas etiquetas, y no en el atributo value.

Por ejemplo `<button>Haga clic aquí</button>`.

```
<button type="button" onclick="alert('Hola!')">pulsa Me!</button>
```

Un button no valida el formulario de la misma manera que lo hace un input type submit, no valida patterns



Los posibles atributos para los botones de acción son:

**name.** Especifica un nombre para el botón.

**type.** El atributo type admite tres argumentos:

- **button:** se utiliza cuando el botón inicia un script (es el valor por defecto del atributo en la mayoría de navegadores).
- **submit:** para enviar el formulario.
- **reset:** para reinicializar el formulario.

**disabled.** Permite desactivar el botón cuando se carga la página. En este caso el botón aparece sombreado en gris. Es posible activarlo (enabled) con un script mediante una acción del usuario, por ejemplo después de haber aceptado las condiciones de uso.

```
<button onclick="location.href='correcto.html'">2 a Correcto</button>
```

Estos campos permiten almacenar datos que no estarán visibles al visitante de la página. porque les permite almacenar datos de formularios anteriores, datos en bruto, o datos que provienen de algún otro script.

## Ejemplo

```
<!DOCTYPE html> <html lang="es"> <head> <title>Html5</title> <meta charset="UTF-8"> </head> <body> <input type="hidden" name="c" value="3.1416"> </body> </html>
```

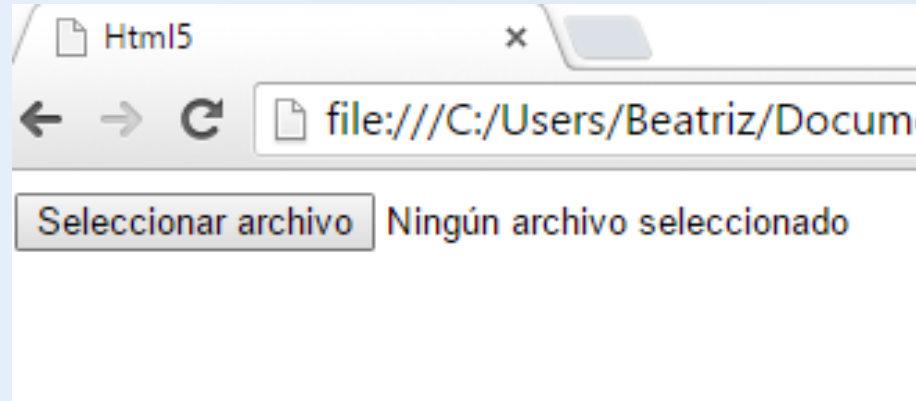
No tiene sentido realizar una captura de pantalla dado que el campo del formulario está oculto (hidden).

Esta etiqueta es perfectamente compatible entre los diferentes navegadores.

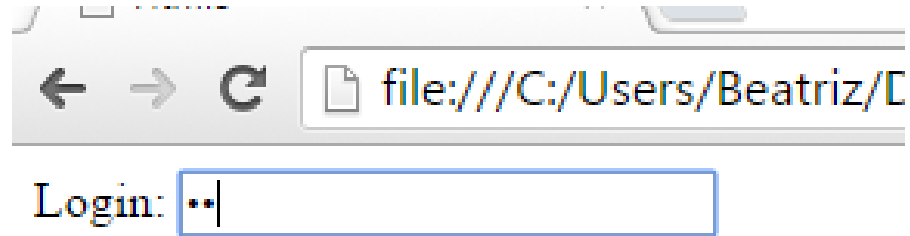
La etiqueta `<input type="file">` permite transferir un archivo (file) desde el puesto del usuario a otro ordenador de tipo servidor.

Un clic en el botón **Examinar** (de nombre variable según el navegador) abre el explorador de carpetas del ordenador del usuario para seleccionar el archivo que se quiere transferir.

## Actividad 14. Crea la siguiente web:



Este tipo de campo es, de hecho, un simple campo de texto donde la codificación se sustituye, en la visualización, por puntos o asteriscos. La sintaxis `<input type="password">`,  
`</>`,



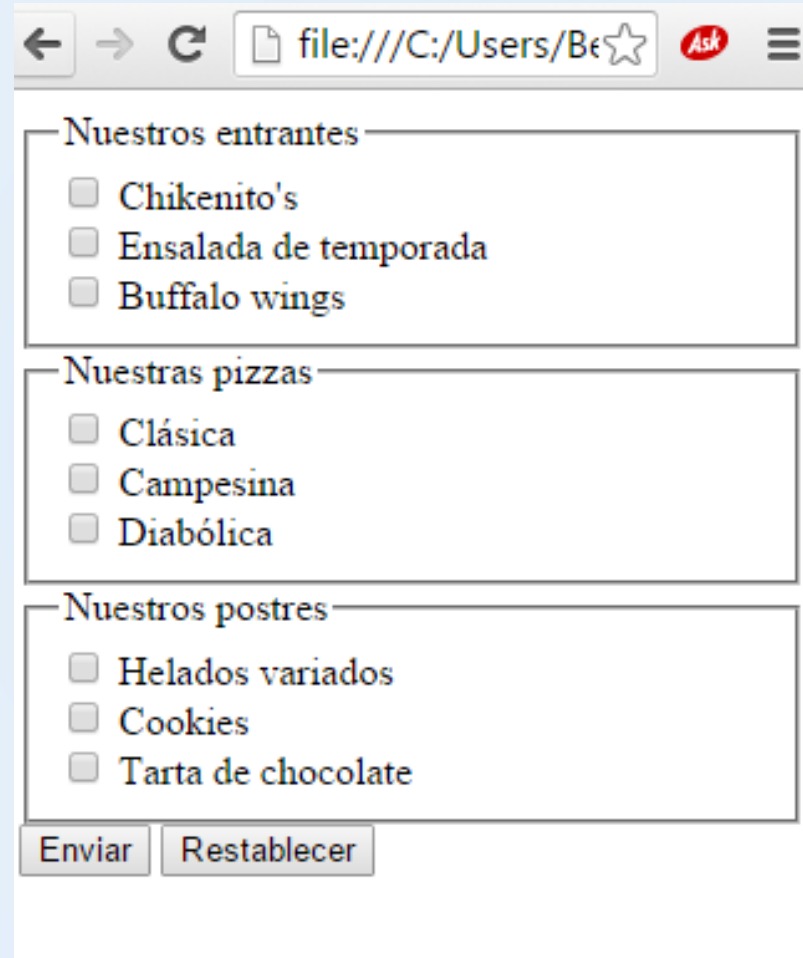
En el caso de formularios largos y complejos, en ocasiones resulta útil agrupar gráficamente ciertos elementos para organizar la página de forma lógica. Las etiquetas **<fieldset>** y **<legend>** permiten mejorar sensiblemente la ergonomía y la usabilidad de los formularios.

La etiqueta `<fieldset> ... </fieldset>` engloba los campos de formulario que se determine. Estos campos se muestran en la pantalla rodeados por un borde. Con el atributo `disabled` deshabilitas todo el `fieldset`. **disabled** Permite desactivar

La etiqueta `<legend> ... </legend>`, situada justo después de la etiqueta `<fieldset>`, agrega una leyenda relacionada con aquellos campos que hayamos agrupado mediante la etiqueta **<fieldset>**

```
<fieldset>
<legend>Nuestros postres</legend>
<input type="checkbox" name="n7">
Helados variados<br>
<input type="checkbox" name="n8">
Cookies<br>
<input type="checkbox" name="n9">
Tarta de chocolate<br>
```

**Actividad 15.** Crea la siguiente web, deshabilitar los postres:



file:///C:/Users/Be☆ Ask

Nuestros entrantes

- ☐ Chickenito's
- ☐ Ensalada de temporada
- ☐ Buffalo wings

Nuestras pizzas

- ☐ Clásica
- ☐ Campesina
- ☐ Diabólica

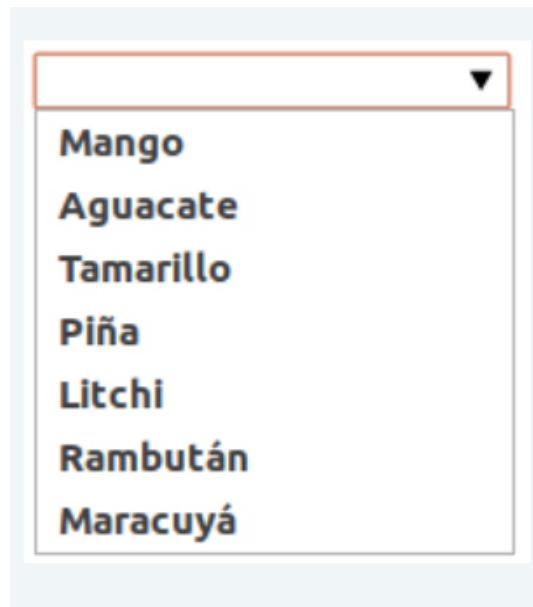
Nuestros postres

- ☐ Helados variados
- ☐ Cookies
- ☐ Tarta de chocolate

Enviar Restablecer

Los formularios en HTML5 han sufrido una importante mejora, permitiendo validaciones sin tener que utilizar JavaScript, especificar listas predefinidas de valores de entrada, definir campos calculados, etc. A continuación se muestran algunos ejemplos de nuevas etiquetas en HTML5 para los formularios. Se diferencia del select en que puedes escribir en la caja una opción que no esté en el desplegable.

La etiqueta `<datalist>`. El tipo de elemento *datalist* era una demanda de hace tiempo de los desarrolladores web, puesto que en muchos formularios hay que ofrecer al usuario una lista de valores predefinidos.

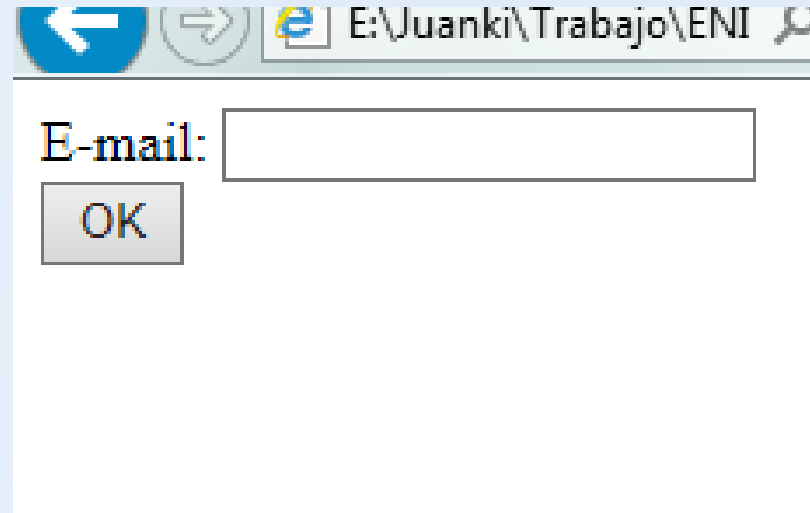


```
<!DOCTYPE html>
<html>
<body>
  <form>
    <input list="frutas" name="frutas">
    <datalist id="frutas">
      <option value="Mango">
      <option value="Aguacate">
      <option value="Tamarillo">
      <option value="Piña">
      <option value="Litchi">
      <option value="Rambután">
      <option value="Maracuyá">
    </datalist>
  </form>
</body>
</html>
```



`<input type="email">`. A priori, el campo de texto no se distingue de un campo de texto normal. No obstante, cuando intentamos enviar una dirección de correo electrónico mal formada, el navegador nos informa de que la dirección es incorrecta. Es posible agregar el atributo `required` que indica que se trata de un campo obligatorio

## Actividad 16. Crea la siguiente web:



The image shows a screenshot of a web browser window. The address bar at the top displays the path 'E:\Juanki\Trabajo\ENI'. Below the address bar, there is a form with the label 'E-mail:' followed by a text input field. Below the input field is a button labeled 'OK'.

la etiqueta `<input type="url">` se define para introducir direcciones Web (url). La dirección debe ser correcta.

Url de su sitio Web:

OK



Introduce una URL

Otro campo nuevo incluido en Html5 son los contadores numéricos.

Esta nueva etiqueta `<input type="number">` tiene atributos específicos:

**min\***. Indica el valor mínimo del contador.

**max\***. Indica el valor máximo del contador.

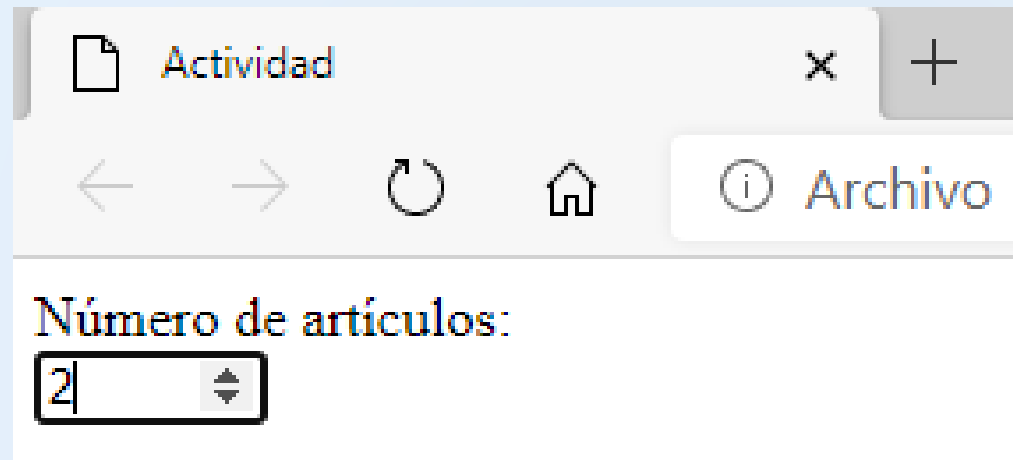
**step**. Determina el incremento del contador cada vez que se hace clic en las flechitas .

**value**. Indica el valor inicial del contador.

```
<input type="number" id="quantity" name="quantity" min="10" max="50" step="5">
```

\*Los atributos max y min solo funcionan con los input types: number, range, date, datetime-local, month, time y week. Para input type text utilizar minlength y maxlength.

## Actividad 17. Crea la siguiente web:



A screenshot of a web browser window. The tab is titled 'Actividad'. The address bar shows a home icon, back and forward arrows, a refresh button, and a dropdown menu with 'Archivo'. Below the address bar, the text 'Número de artículos:' is followed by a text input field containing the number '2' and a spinner control.

Los calendarios han cobrado especial importancia en las aplicaciones de la Web 2.0. Su uso es indiscutible en todos los sitios de reservas on-line.

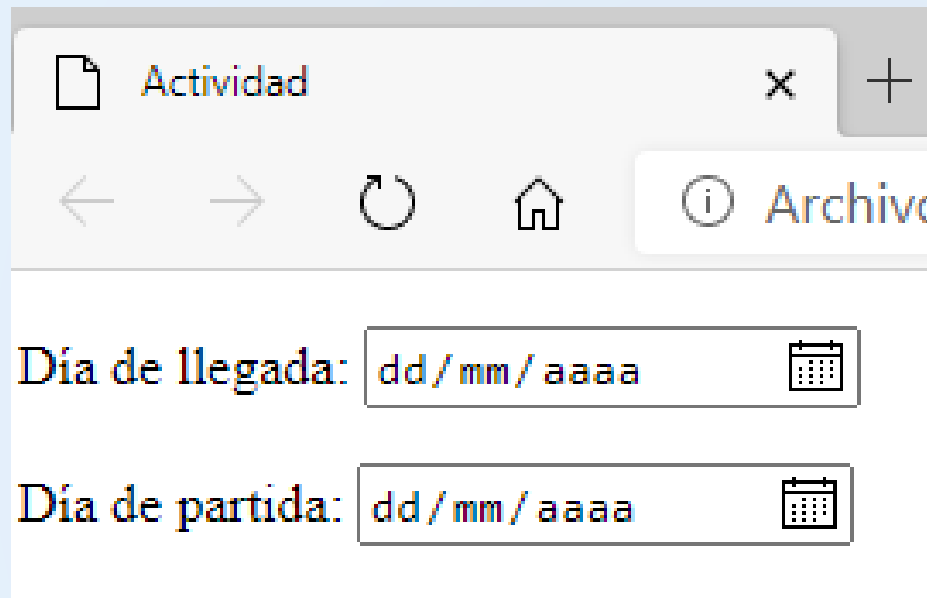
**<input type="date">** La más general. Permite seleccionar el año, el mes y el día.

**<input type="month">** Para seleccionar el mes y el año. Es útil, por ejemplo, para introducir la fecha de expiración de una tarjeta de crédito.

**<input type="week">** Para una semana determinada.

**<input type="time" min="11:00" max="23:00" step="900" value="11:00">** Para una hora. El atributo min permite fijar una hora de inicio del contador, max una hora de fin y step el paso de avance, en este ejemplo de 15 en 15 minutos (900 segundos) cada vez que se hace clic en una flechita.

## Actividad 18. Crea la siguiente web:



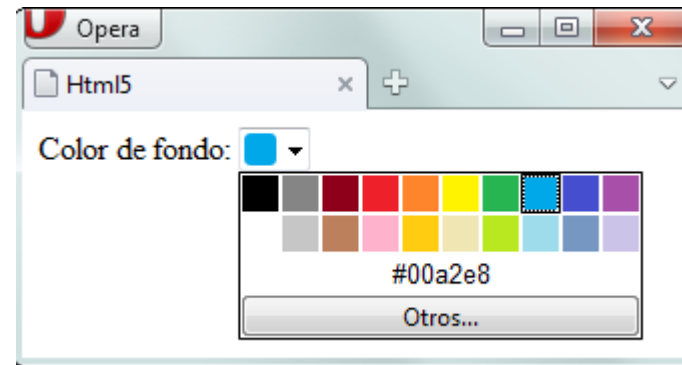
The screenshot shows a web browser window with a single tab titled 'Actividad'. The address bar contains the text 'Archivo'. Below the address bar, there are two form fields. The first field is labeled 'Día de llegada:' and contains the text 'dd/mm/aaaa' with a calendar icon to its right. The second field is labeled 'Día de partida:' and also contains the text 'dd/mm/aaaa' with a calendar icon to its right.

Son muy numerosos los sitios o aplicaciones Web que permiten realizar una búsqueda. Normalmente toman el aspecto de un campo de texto.

Html5 proporciona, con la etiqueta `<input type="search">` un campo de texto dedicado especialmente a esta posibilidad de búsqueda.



`<input type="color">`. Cuando se sitúa el foco en este campo de texto, se abre automáticamente una paleta de colores para realizar la selección del color, cuyo valor se reporta



Esta nueva etiqueta `<input type="range">` tiene atributos específicos:

**min.** Indica el valor mínimo del contador.

**max.** Indica el valor máximo del contador.

**step.** Determina el paso de avance del contador cada vez que se hace clic con el ratón en la flecha para aumentar o disminuir

**value.** Determina el valor inicial del contador.

```
<form >
```

```
<p><label for="rango">Elige un valor entre 0</label>
```

```
<input id="rango" type="range" min="0" max="6" step="1"
```

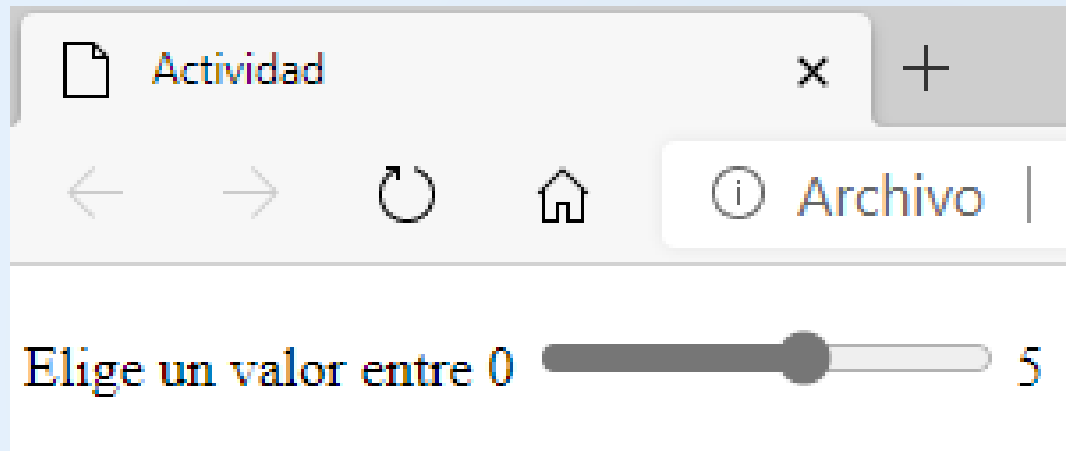
```
name="rango" oninput= "z.value=this.value"/>
```

```
<br>
```

```
Valor <output name="z"></output>
```

```
</form>
```

## Actividad 19. Crea la siguiente web:



## Actividad 20

Crear una página (index.html) que de la bienvenida a la empresa y tenga 2 botones o enlaces : Login y Registro que nos enviarán a la página de login o a la de registro.

- Página registro.html: Será un formulario para registrarse en una web. Se pedirá el nombre de la persona (solo se admitirán blancos y letras), la fecha de nacimiento, el color favorito, el sueldo (entre 0 y 100.000 con saltos de 5000), el email, el nombre de usuario (solo se admitirán números y letras y longitud entre 6 y 15 caracteres, y la password (debe tener una longitud mínima de 8 caracteres). Tendrá un botón de enviar que nos mandará a index.html, otro de borrar todos los campos de la página y un enlace a la página index.html nombrado con volver.
- Página login.html : Será un formulario con 2 campos nombre usuario (mismas restricciones que en registro) y password (mismas restricciones que en registro) y un botón de enviar que te redirigirá a la página de hola.html
- Página hola.html. Dará la bienvenida al usuario con un mensaje de Hola.

**Actividad 21.** Crea un formulario usando las herramientas vistas hasta ahora para recoger las sugerencias y comentarios para una web.

1. La etiqueta `<output>` en los formularios permite calcular valores dentro de estos y almacenar el resultado en otro campo. En el siguiente ejemplo se utiliza
  - a. El campo `z` para visualizar el valor seleccionado con `input type="range"`
  - b. El campo `x` para calcular el perímetro de la circunferencia (en `oninput`) y mostrarlo en pantalla.
  - c. El campo `c` para enviar el valor de la circunferencia

```
<form action="correcto.html" method="get" oninput=
  "x.value=parseInt(a.value * 2 * 3.1415);
  z.value=parseInt(a.value);
  c.value=x.value;" >
```

Longitud del radio: `<input type="range" id="ida" name="a" value="0">`

100

`<br>`

Valor del radio: `<output name="z" for="a"></output>`

`<br>`

Valor circunferencia: `<output name="x" for="a"></output>`

`<input type="hidden" name="c" value="3.1416">` // para enviar el valor  
de la circunferencia.

`<input type="submit">`

`</form>`

Longitud del radio:  100

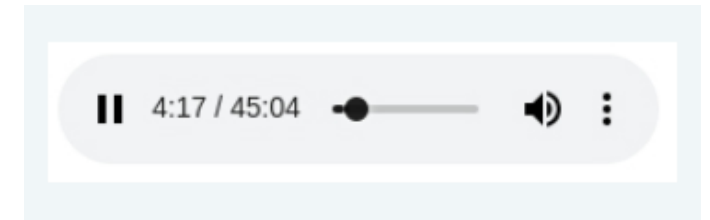
Valor del radio:

Valor circunferencia:

Etiqueta <audio> y <video>. Dentro de las mejoras que ofrece HTML5 están los elementos multimedia, como pueden ser las etiquetas <audio> y <video>. Antes había que utilizar un reproductor externo, con lo cual no existía una normalización y aspecto o utilización estándar. Ahora, estas nuevas etiquetas permiten a los desarrolladores incluir audio, vídeo y otros elementos multimedia directamente en sus páginas web.

```
<audio controls autoplay loop>  
  <source src="grabacionPrueba1.m4a" type="audio/mp4">  
  <source src="grabacionPrueba1.ogg" type="audio/ogg">
```

Your browser does not support the audio element.  
</audio>



Los atributos de la etiqueta <audio> son:

**src.** Obligatorio. Define la ruta del archivo de sonido.

**controls.** Muestra los controles del reproductor de audio Incluye las funcionalidades de reproducción, pausa, avance y volumen.

**autoplay.** Define la reproducción automática del archivo de sonido desde el momento en que esté disponible, tras la carga de la página. No siempre funciona.

**loop.** Especifica que el archivo de sonido se reproduzca en bucle. De este modo, el sonido comienza de nuevo una vez termina.

**preload.** Indica al navegador que debe descargar el archivo de audio durante la carga de la página de modo que esté disponible para una reproducción inmediata una vez la solicite el usuario. Este atributo preload se ignorará si el atributo autoplay está activado. Este atributo puede tomar los valores:

- preload="none". Sin carga previa.
- preload="metadata". Carga previa de los metadatos (metadata) asociados al archivo.
- preload="auto". Carga previa automática.

- Formato ogg. Esta ha sido la opción de Firefox, primer navegador en implementar la etiqueta <audio>. El formato ogg se ha mantenido.
- Formato mp3
- Formato acc
- Formato wav
- Formato mp4

	ogg	mp3	wav
<b>Internet Explorer</b>	no	Explorer 9+	no
<b>Firefox</b>	Firefox 3.6+	si	Firefox 3.6+
<b>Google Chrome</b>	Chrome 6.0+	Chrome 6.0+	Chrome 9.0+
<b>Safari</b>	no	Safari 5.0+	Safari 5.0+
<b>Opera</b>	Opera 10.0+	si	Opera 10.0+
<b>iOS Safari</b>	no	iOS Safari 3+	si

La etiqueta <source> nos va a permitir resolver la problemática de los distintos formatos de archivo. La etiqueta <source> se utiliza para especificar **varios tipos de archivos de audio**. Cada navegador escogerá el formato que mejor le convenga o el que pueda reproducir. Conviene poner varios formatos para que el navegador pueda elegir entre otros si el primero indicado no funciona en ese navegador.



Los atributos de la etiqueta <source> son:

**src.** Obligatorio. Define la ruta del archivo de sonido.

**type.** Define el tipo MIME del contenido. Puede valer:

- type="audio/ogg"
- type="audio/mpeg"
- type="audio/mp3"

<audio controls>

<source src="piano\_ogg.ogg">

<source src="piano\_mp3.mp3">

<source src="piano\_acc.acc">

Su navegador no soporta la etiqueta audio.

</audio>

Html5 proporciona, a día de hoy, una etiqueta nueva para reproducir los archivos de vídeo, independientemente de los plug-ins instalados en el puesto del usuario.

Insertar un archivo de vídeo se realiza de forma muy sencilla mediante la etiqueta:

```
<video src="archivo_video">
```

Su navegador no soporta la etiqueta video.

```
</video>
```

Los atributos de la etiqueta <video> son:

**src.** Obligatorio. Define la ruta del archivo de vídeo.

**width.** Determina la anchura del vídeo.

**height.** Determina la altura del vídeo.

Igual que con la etiqueta <img>, puede definir de forma explícita las dimensiones del vídeo.

**poster.** El atributo poster permite especificar una imagen que el navegador usará mientras se está descargando el vídeo o hasta que el usuario inicie su reproducción. Si no se especifica este atributo, se inserta la primera imagen del vídeo en su lugar.

```
<video src="video_ogv.ogv" poster="imagen.png">
```

**controls.** Muestra los controles del reproductor del vídeo. Se incluyen las funciones de reproducción, pausa, avance y volumen.

```
<video src="video_ogv.ogv" controls>
```

En ausencia del atributo controls, no se mostrarán los controles del reproductor en el navegador.

**autoplay.** Define la lectura automática del archivo de vídeo una vez esté disponible, tras la carga de la página.

```
<video src="video_ogv.ogv"
```

**loop.** Especifica que el archivo de vídeo se reproducirá en bucle. De este modo, el vídeo se reproduce de nuevo una vez ha llegado a su fin.

```
<video src="video_ogv.ogv" loop>
```

**preload.** Indica al navegador que debe descargar el archivo de vídeo durante la carga de la página de modo que esté disponible para una reproducción inmediata una vez lo solicite el usuario. Este atributo preload se ignora si está presente el atributo autoplay.

```
<video src="video_ogv.ogv" preload>
```

Este atributo puede tomar los valores:

- preload="none". No existe carga previa.
- preload="metadata". Carga previa de los metadatos asociados al archivo.
- preload="auto". Carga previa automática.

- Formato ogv
- Formato H.264
- Formato WebM

	Ogv	H.264	WebM
<b>Internet Explorer</b>	no	Explorer 9+	no
<b>Firefox</b>	Firefox 3.6+	En proyecto	Firefox 4.0+
<b>Google Chrome</b>	Chrome 4.0+	Chrome 4.0+	Chrome 6.0+
<b>Safari</b>	no	Safari 3.2+	no
<b>Opera</b>	Opera 10.5+	no	Opera 10.6+
<b>iOS Safari</b>	no	iOS Safari 3.2+	no
<b>Android Browser</b>	no	Android 2.1+	Android 2.3+

La etiqueta <source> nos va a permitir resolver la problemática de los distintos formatos de archivo. La etiqueta <source> se utiliza para especificar **varios tipos de archivos de video**. Cada navegador escogerá el formato que mejor le convenga o el que pueda reproducir.

Los atributos de la etiqueta <source> son:

**src.** Obligatorio. Define la ruta del archivo de sonido.

**type.** Define el tipo MIME del contenido. Puede valer:

- type="video/ogv"
- type="video/mp4"
- type="video/webm"

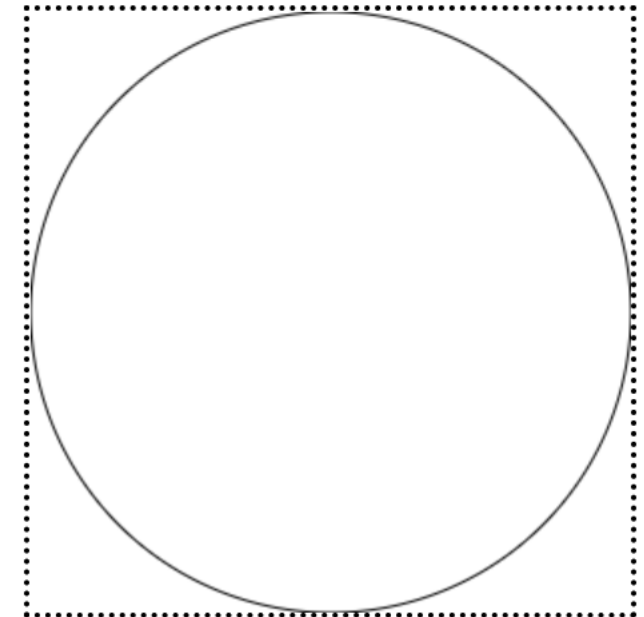
También puede especificarse el codec utilizado. El atributo type queda:

- type="video/ogg; codecs='theora, vorbis'"
- type="video/webm; codecs='vp8, vorbis'"
- type="video/mp4; codecs='avc1.42E01E, mp4a.40.2'"

Especificando el atributo type acelerará el proceso de carga de la etiqueta <source> más adecuada para el navegador.

La etiqueta `<canvas>`. HTML5 ofrece la posibilidad de utilizar elementos gráficos mediante la nueva etiqueta `<canvas>`. Aunque esta etiqueta solamente proporciona un lienzo (sin borde, salvo que se especifique, y sin contenido) para poder dibujar sobre él mediante scripts (JavaScript), es posible dibujar gráficos como círculos, líneas, etc.

```
<html>
  <head>
  </head>
  <body>
    <canvas id="miLienzo" width="300" height="300" style="border:3px dotted #000000">
    </canvas>
    <script>
      var lienzo = document.getElementById("miLienzo");
      var elem = lienzo.getContext("2d");
      elem.beginPath();
      elem.arc(150, 150, 150, 0, 2 * Math.PI);
      elem.stroke();
    </script>
  </body>
</html>
```



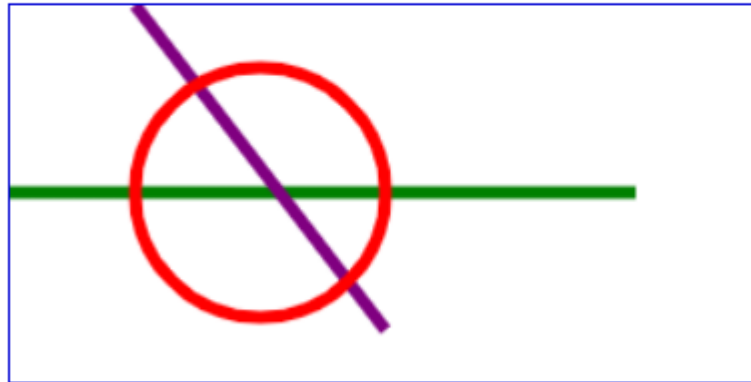
[HTML Canvas Tutorial for Beginners \(simplilearn.com\)](https://www.simplilearn.com/html5-canvas-tutorial-for-beginners)

[HTML5 Canvas Tutorial for beginners - w3resource](https://www.w3resource.com/html5/canvas-tutorial-for-beginners)

[https://developer.mozilla.org/es/docs/Web/API/Canvas\\_API/Tutorial/Drawing\\_shapes](https://developer.mozilla.org/es/docs/Web/API/Canvas_API/Tutorial/Drawing_shapes)

**Actividad 22.** Crea una web que tenga un audio, un video y represente el siguiente canvas :

**dibujo con canvas3**





**Recurso Web.**

**w3schools.com** es una web dónde puedes encontrar diferentes tutoriales, referencias, ejemplos y ejercicios para aprender todos los lenguajes para hacer paginas webs. En concreto el siguiente enlace muestra todo lo relacionado con HTML:

<https://www.w3schools.com/html/>

**Developer.mozilla.org** proporciona documentación sobre html y css.

[MDN Web Docs \(mozilla.org\)](https://developer.mozilla.org/)

**Recurso Web.** Herramientas de generación

- [WordPress](#)
- [Vue.js - The Progressive JavaScript Framework | Vue.js \(vuejs.org\)](#)
- <https://sergioescriba.com/listado-programas-crear-disenar-paginas-web/>

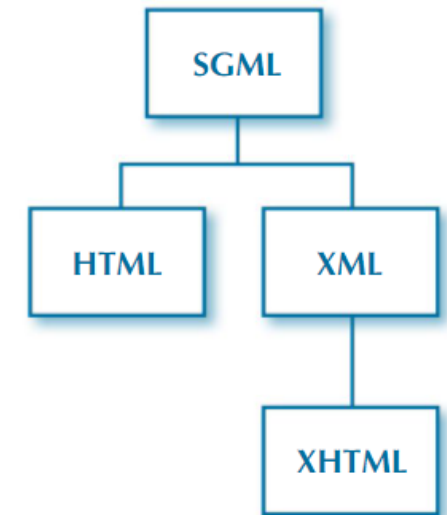
# xHTML

---

Se considera XHTML, sigla de extensible hypertext markup language (lenguaje de marcado de hipertexto extensible), como la adaptación de HTML en XML, pues mantiene las características de HTML e incluye normas XML. HTML descende de SGML, mientras que XHTML lo hace de XML.

En la actualidad, existen varias versiones de XHTML:

- XHTML 1.0. Surge en el año 2000 es la primera de todas y se considera similar a HTML 4.0 reformulado, cuya finalidad es que, gracias a ella, se use como lenguaje de contenidos. Esta versión presenta tres definiciones distintas para DTD (definición del tipo de documento) y es compatible con algunas versiones de HTML.
- XHTML 1.1. Esta versión no es compatible con versiones anteriores de HTML. Se considera una versión basada en módulos que sirven de base para futuros documentos extendidos, quitando características que estaban ya en desuso.



**Recurso Web.** En el siguiente enlace tenéis una guía de referencia escrita por la w3c sobre XHTML: [XHTML™ Basic 1.1 - Second Edition \(w3.org\)](https://www.w3.org/TR/xhtml1/)

[site:w3.org xhtml](https://www.w3.org/xhtml/) at DuckDuckGo

otros:

<https://developer.mozilla.org/es/docs/Glossary/XHTML>

## Diferencias html5- xHTML

[Diferencia entre XHTML y HTML5 – Acervo Lima](#)

Validador

<https://validator.w3.org/>

# Diferencias sintácticas y estructurales

---

XHTML

A la hora de crear documentos XHTML, no existe una gran diferencia con HTML, pero, para que esté bien formado, es necesario que cumpla una serie de reglas sintácticas heredadas.

XML es sensible a las mayúsculas y minúsculas, por lo que es necesario añadir los **elementos y atributos** en minúsculas.

## EJEMPLO 1

- |   |                                     |   |
|---|-------------------------------------|---|
| 1 | <input checked="" type="checkbox"/> | <code>&lt;h1&gt;Bienvenido a Lenguaje de Marcas&lt;/h1&gt;</code> |
| 2 | <input type="checkbox"/>            | <code>&lt;H1&gt;Bienvenido a Lenguaje de Marcas&lt;/H1&gt;</code> |

Mientras la opción 1 del ejemplo es correcta, ya que todos los nombres de los elementos están en minúsculas, la 2, a la hora de validarlo, informará de que el nombre de la etiqueta contiene caracteres en mayúsculas, lo cual no se permite en XHTML y sí en HTML.

Los atributos tendrán obligatoriamente un valor que deberá ir **entre comillas dobles**, ya sean valores de texto, números o alfanuméricos, deberán ir entrecomillados. Como ejemplo, se ha usado la etiqueta de imagen con tres atributos: texto alternativo, fuente y ancho de la imagen.

## EJEMPLO 2

- |   |                                     |   |
|---|-------------------------------------|---|
| 1 | <input checked="" type="checkbox"/> | <code>&lt;img alt="casa de campo" src="casa.jpg" width="25%" /&gt;</code> |
| 2 | <input type="checkbox"/>            | <code>&lt;img alt="casa de campo" src="casa.jpg" width=25% /&gt;</code>   |

La opción 1 del ejemplo es correcta, ya que todos los valores de los atributos van con comillas, mientras que la opción 2 no lo es porque el valor del atributo width va sin comillas, aunque sí sería válida en HTML.

Todas las etiquetas o elementos tienen que estar cerrados, incluyendo las etiquetas vacías.

En HTML, la gran mayoría de etiquetas tienen una de apertura y otra de cierre, aunque existen algunas, como las de comienzo de nueva línea o de imagen, en las que no es necesario su cierre.

## EJEMPLO 3

1 ☒ `<br />`

2 ☐ `<br>`

En el ejemplo, la opción número 1 es la correcta al estar en minúsculas y cerrada, mientras que a la opción 2 le falta el cierre, aunque es válida en HTML, tal y como se ha mencionado.



A la hora de anidar etiquetas, hay que tener en cuenta que no deben solaparse (tienen que cerrarse conforme a cómo se abren).

## EJEMPLO 4

- 1 ☒ `<i><u>Texto en cursiva y subrayado</u></i>`
- 2 ☐ `<i><u>Texto en cursiva y subrayado</i></u>`

En el caso primero del ejemplo, se aprecia que las etiquetas de subrayado y cursiva no se solapan, siendo esta la manera correcta. En el caso 2, se observa que, para poner el texto en subrayado, entre la etiqueta de apertura y la de cierre, aparece la etiqueta cerrada de cursiva, inválido en XHTML y válido en HTML.

Aunque los atributos minimizados no son los más usados, el nombre debe coincidir con su valor. Todos los atributos tienen que llevar un valor.

## EJEMPLO 5

```
1 ☒ <select>
    <option value="longan">Longan</option>
    <option value="carambola">Carambola</option>
    <option value="Litchi">Litchi</option>
    <option value="Rambutan" selected="selected">Rambutan</option>
</select>

2 ☐ <select>
    <option value="longan">Longan</option>
    <option value="carambola">Carambola</option>
    <option value="Litchi">Litchi</option>
    <option value="Rambutan" selected>Rambutan</option>
</select>
```

Aunque la segunda opción del ejemplo 5 es válida en HTML, para XHTML, hay que añadir un valor al atributo selected con el mismo nombre, tal y como se ve en la primera opción.

Respecto a la estructura, no varía mucho a la de un documento HTML y su mayor diferencia es la declaración del DTD para validar el documento, que es obligatoria. El elemento raíz tiene que ser la etiqueta <html>, que tendrá como atributos xmlns, que indica el espacio nominal XHTML (<http://www.w3.org/1999/xhtml>).

Al heredar XHTML de XML, comparte parte de su sintaxis y normas, por ello se utiliza DTD, sigla de document type definition o (definición del tipo de documento). Dentro de un DTD, se encuentra una serie de reglas que deben cumplirse.

En la versión 1.0 de XHTML, existen tres posibilidades para la declaración del DTD (estricto, transitorio y *frameset*), incluyendo alguna de las tres fuera de la etiqueta <html> mediante la declaración DOCTYPE, pues esta es la primera que hay que añadir a la hora de crear un nuevo documento.

- Declaración **estricta** se usa en el caso de **no emplear** elementos de **estilo**,
- Declaración **transición** **sí** usa estilos.
- Declaración **frameset** en el caso de **usar marcos** dentro del documento,

## Declaración del DTD

Tipo DTD	Declaración
Estricto	<code>&lt;!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"&gt;</code>
Transitorio	<code>&lt;!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"&gt;</code>
Frameset	<code>&lt;!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd"&gt;</code>

[W3C QA - Recommended list of Doctype declarations you can use in your Web document](#)

En el ejemplo siguiente se aprecia un documento XHTML versión 1.0 con una declaración de transición, que permite elementos en desuso.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transi-  
tional//EN" "http://www.w3.org/TR/xhtml1/DTD/xht-  
ml1-transitional.dtd">  
<html xmlns="http://www.w3.org/1999/xhtml" >  
<head>  
  <title>LM</title>  
</head>  
<body>  
</body>  
</html>
```

XHTML presenta una serie de ventajas respecto a HTML, entre las que destacan:

- A los documentos XHTML pueden agregárseles distintas aplicaciones (scripts y applets).
- Pueden validarse con herramientas XML.
- Compatible con los navegadores antiguos.
- Pueden presentar mayor rendimiento.
- Formato abierto.
- **Facilidad de mantener y editar.**

Refs:

<https://www.w3.org/International/articles/serving-xhtml/index.es.html>

Existen otros lenguajes de marcas populares en la web como Markdown que está teniendo bastante éxito últimamente y se puede utilizar, por ejemplo, en github.

<https://markdown.es/>



**Actividad.** Realiza una web en html5 sobre una temática a elegir por ti que cumpla los siguientes requisitos:

- Al menos 6 páginas
- Al menos 1 formulario, con uso de expresiones regulares. Debe haber etiquetas de al menos 7 de los tipos vistos.
- Al menos 1 lista ordenada y otra no ordenada con sublistas (ambas)
- Al menos 1 tabla, con varias filas y columnas. Debe tener hacerse uso de, al menos alguna fusión de celdas.
- Varios tipos de encabezado.
- Enlaces y botones para pasar de una página a otra. Enlaces relativos en alguna de las páginas.
- Inserción de vídeo, audio e imágenes.
- Utilización de etiquetas semánticas.

No es necesario que te esmeres en el formato porque esa parte la realizaremos con css3.

## Presentación:

Elige una herramienta de las que puedes encontrar en el artículo siguiente :  
<https://sergioescriba.com/listado-programas-crear-disenar-paginas-web/>

Realiza, por grupos de máximo 3 personas, una pequeña presentación sobre su uso. La presentación debe durar unos 15 minutos y la realizarás en clase ante tus compañeros.

No puede haber 2 grupos que elijan la misma herramienta así que debes comunicar a tu profesor cuál has elegido antes de empezar el trabajo.