

UT2. Introducción a HTML5



Contenido

2.1 ¿Qué es HTML?	5
2.1.1 Breve historia de HTML	5
2.1.2 HTML y XHTML	6
2.2 Editores de texto	6
2.3 Edición básica en HTML	6
2.3.1 Etiquetas	6
2.3.2 Atributos	7
2.4 Estructura de un documento HTML	7
2.4.1 Estructura básica de una página en HTML5	8
2.4.2 Etiquetas obsoletas de HTML	9
2.5 Comentarios	9
2.6 Etiquetas básicas	9
2.6.1 Párrafo simple <p>	9
2.6.2 Salto de línea 	10
2.6.3 Línea horizontal <hr>	10
2.6.4 Títulos o encabezados <h>	11
2.7 Elementos de texto	11
2.7.1 Texto en negrita 	11
2.7.2 Texto enfatizado o cursiva 	12
2.7.3 Texto resaltado <mark>	12
2.7.4 Texto tachado 	12
2.7.5 Texto subrayado <ins>	13
2.7.6 Texto subíndice <sub>	13
2.7.7 Texto superíndice <sup>	13
2.7.8 Texto pequeño <small>	13
2.7.9 Abreviatura <abbr>	14
2.7.10 Definiciones <dfn>	14
2.8 Escritura de código y texto informático	15
2.8.1 Escritura de código <code>	15
2.8.2 Texto preformateado <pre>	15
2.8.3 Resultados de operaciones en un ordenador <samp>	16
2.9 Citas	16
2.9.1 Cita en bloque <blockquote>	16
2.9.2 Cita corta en línea <q>	17
2.9.3 Cita de trabajo creativo <cite>	17
2.10 Otras etiquetas	18

2.10.1 Etiqueta de direcciones <address>	18
2.11 Listas	18
2.11.1 Listas NO ordenadas (viñetas) 	18
2.11.3 Listas ordenadas (numéricas) 	19
2.11.4 Listas anidadas	20
2.11.5 Listas de términos <dl> <dt> <dd>	21
2.12 Enlaces <a>	22
2.12.1 Enlaces globales	22
2.12.2 Enlaces locales	23
2.12.3 Enlaces internos	23
2.12.4 URLs absolutas	24
2.12.5 URLs relativas	24
2.13 Imágenes 	26
2.14 Tablas <table> <tr> <td>	28
2.14.1 Creación de tablas simples	28
2.14.2 Encabezados	29
2.14.3 Títulos de las tablas.	30
2.14.4 Agrupación de filas	30
2.14.5 Combinar celdas	32
2.14.6 Tablas dentro de tablas	34
2.15 Formularios <form>	36
2.15.1 Métodos de paso de datos	36
2.15.2 Elemento formulario	36
2.15.3 Elemento <input>	37
2.15.3.1 Campos de texto type="text"	39
2.15.3.2 Campos de contraseñas type="password"	40
2.15.3.3 Botones	40
2.15.3.4 Opción única type="radio"	40
2.15.3.5 Campos numéricos type="number"	41
2.15.3.6 Campo de fecha type="date"	42
2.15.3.7 Campo de hora type="time"	42
2.15.3.8 Casillas de verificación type="checkbox"	43
2.15.3.9 Campo de email type="email"	43
2.15.3.10 Campo oculto type="hidden"	44
2.15.3.11 Otros campos	45
2.16 Cuadro de texto multilínea <textarea>	45
2.17 Listas de opciones <select> <option>	45

2.18 Elementos mostrar/ocultar <details> <summary>	46
2.19 Insertar audio <audio>	47
2.20 Insertar vídeo <video>	48
2.20.1 Insertar vídeos de YouTube <iframe>	50
2.21 Otras etiquetas	51

2.1 ¿Qué es HTML?

Definiéndolo de forma sencilla, **HTML** (*HyperText Markup Language*) es el lenguaje que se utiliza para crear páginas web. Estas páginas son ejecutadas por los navegadores (Chrome, Firefox...)

El lenguaje **HTML** es un estándar reconocido en todo el mundo y cuyas normas define un organismo sin ánimo de lucro llamado [World Wide Web Consortium](#), más conocido como **W3C**. Como se trata de un estándar reconocido por todas las empresas relacionadas con el mundo de Internet, una misma página HTML se visualiza de forma muy similar en cualquier navegador de cualquier sistema operativo.

El propio **W3C** define el lenguaje **HTML** como "un lenguaje reconocido universalmente y que permite publicar información de forma global".

2.1.1 Breve historia de HTML

La historia completa de **HTML** es tan interesante como larga, por lo que a continuación se muestra su historia resumida a partir de la información que se puede encontrar en la Wikipedia.

El origen de **HTML** se remonta a 1980, cuando el físico **Tim Berners-Lee**, trabajador del [CERN](#) (*Organización Europea para la Investigación Nuclear*) propuso un nuevo sistema de "hipertexto" para compartir documentos.

El primer documento formal con la descripción de **HTML** se publicó en 1991 bajo el nombre *HTML Tags (Etiquetas HTML)* y todavía hoy puede ser consultado online a modo de *reliquia informática*.

A partir de ese momento van apareciendo diferentes versiones de HTML, siendo el [W3C](#) (*World Wide Web Consortium*) el encargado de publicar estos estándares.

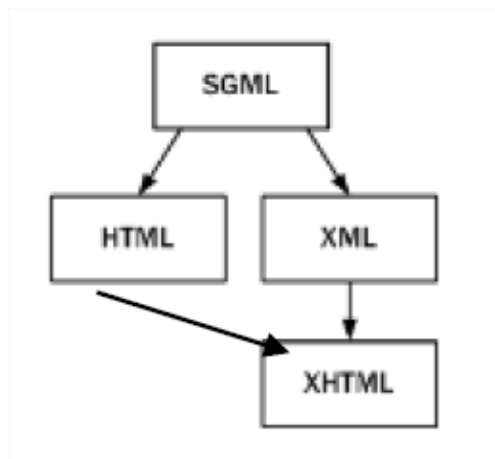
HTML 4.0 se publicó el 24 de abril de 1998 y supone un gran salto desde las versiones anteriores. Entre sus novedades más destacadas se encuentran las hojas de estilos CSS, la posibilidad de incluir pequeños programas o *scripts* en las páginas web, tablas complejas y mejoras en los formularios.

Poco después, apareció HTML 4.01 con pocas novedades significativas.

Desde la publicación de HTML 4.01, no apareció nada nuevo hasta 2009 que aparece HTML 5.

2.1.2 HTML y XHTML

El lenguaje **XHTML** es muy similar al lenguaje **HTML**. De hecho, **XHTML** no es más que una adaptación de **HTML** al lenguaje **XML**. Técnicamente, **HTML** es descendiente directo del lenguaje **SGML**, mientras que **XHTML** lo es del **XML** (que, a su vez, también es descendiente de **SGML**).



2.2 Editores de texto

Nuestra herramienta principal para crear una página web es el editor de código. Tenemos varias opciones:

- **Sublime Text:** este editor no es gratuito, pero tiene una licencia de prueba indefinida.
- **NotePad ++:** es un editor gratuito que soporta muchos lenguajes y tiene soporte para plugins.
- **Brackets:** es un editor de Adobe, también es gratuito.
- **Atom:** es gratuito y es de los creadores de GitHub.
- **Visual Studio Code:** es un editor de código fuente desarrollado por Microsoft para Windows, Linux, macOS y Web. Incluye soporte para la depuración, control integrado de Git, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código.

2.3 Edición básica en HTML

Un documento HTML contiene **texto** y **etiquetas**.

2.3.1 Etiquetas

Las **etiquetas** sirven para delimitar elementos de la página. Por ejemplo, un párrafo es un elemento de la página, una tabla también. Incluso hay elementos que contienen otros elementos (las tablas constan de filas y las filas de celdas...).

Las **etiquetas** aparecen entre los signos de menor y mayor (< >). Cuando un navegador se encuentra un texto dentro de estos dos signos, interpreta que lo que hay dentro es un elemento.

Ejemplo:

``, `<p>`, `<h1>`

La mayoría de las etiquetas afectan a un determinado texto, que estará delimitado por una etiqueta de apertura y otra de cierre. Se interpretará como el inicio y fin de un determinado elemento de la página.

El cierre de una etiqueta se marca poniendo el símbolo `/`.

Ejemplo:

Texto normal ``texto negrita``

Ya se ha comentado que un elemento puede contener dentro más elementos, por lo que entre la apertura y el cierre de una etiqueta puede haber más etiquetas. **Una norma importante es que siempre debemos cerrar primero las etiquetas abiertas más tarde.**

El siguiente código no es correcto:

Texto normal ``texto negrita y cursiva``

Se está cerrando la etiqueta *em* después de *strong* y eso no es correcto porque *em* es la última que se abrió.

Lo correcto sería:

Texto normal ``texto negrita y cursiva``

Es decir, se cierra primero el elemento ***em*** que fue el último en abrirse.

2.3.2 Atributos

Algunas etiquetas tienen atributos. Los atributos son propiedades de cada elemento a las que podemos asignar un valor, de modo que dicho valor varía el comportamiento del elemento. Por ejemplo, puede cambiar el color o el tamaño.

La sintaxis de los atributos es:

`<etiqueta atributo1="valor1" atributo2="valor2"...>` Texto `</etiqueta>`

A los atributos se les asignan valores que deben ir entre comillas. El símbolo de igualdad (=) es obligatorio.

Ejemplo:

`<p align="center">`

2.4 Estructura de un documento HTML

HTML, como ya hemos comentado, es un **lenguaje de etiquetas** (también llamado **lenguaje de marcado**) y las páginas web habituales están formadas por cientos o miles de etiquetas.

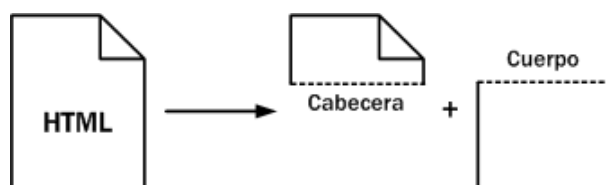
La principal ventaja de los lenguajes de etiquetas es que son muy sencillos de leer y escribir por parte de las personas. La principal desventaja es que pueden aumentar mucho el tamaño del documento, por lo que en general se utilizan

etiquetas con nombres muy cortos.

Un documento **HTML** se divide en dos partes principales: **la cabecera y el cuerpo**.

La **cabecera** incluye información sobre la propia página, como por ejemplo su título y su idioma.

El **cuerpo** de la página incluye todos sus contenidos, como párrafos de texto, imágenes, tablas y demás elementos.



El cuerpo (llamado **body**) contiene todo lo que el usuario ve en su pantalla y la cabecera (llamada **head**) contiene todo lo que no se ve (con la única excepción del título de la página, que los navegadores muestran como título de sus ventanas).

2.4.1 Estructura básica de una página en HTML5

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="utf-8">
  <title>Primer Ejercicio HTML 5</title>
</head>
<body>
  <p>Esperamos que <strong> el curso </strong> vaya bien y os guste</p>
</body>
</html>
```

- La primera etiqueta es **<!DOCTYPE>** Esta etiqueta indica que versión de HTML se utiliza en el documento para que los navegadores puedan aplicar la gramática y sintaxis correspondiente al documento. En nuestro ejemplo le estamos indicando que vamos a usar **HTML 5**.
- Seguidamente se encuentra la etiqueta **<html lang="es">**: html indica el comienzo y el final de un documento (</html>). Ninguna etiqueta o contenido puede colocarse antes de la etiqueta <html> ni después de la etiqueta </html> (con la excepción del tipo). En el interior de la etiqueta <html> se definen la **cabecera** y el **cuerpo** del documento.

El atributo **lang** puede ser utilizado por los navegadores y otras aplicaciones para aportar a los usuarios la información más adecuada o para presentar información a los usuarios de la manera más correcta. También es útil para los lectores de pantalla.

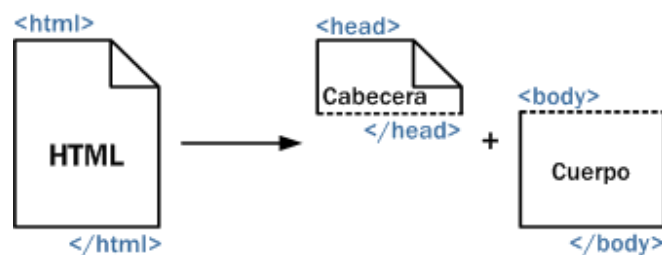
- **<head>**: delimita la parte de la cabecera del documento. Los elementos contenidos en ella sirven para indicar información al navegador web y a los

buscadores. En ella se pone información del propio documento HTML, como por ejemplo su título y el idioma de la página.

Los contenidos indicados en la cabecera no son visibles para el usuario, con la excepción de la etiqueta **<title>**, que se utiliza para indicar el título del documento y que los navegadores lo visualizan en la pestaña de la ventana del navegador.

También tenemos la etiqueta **<meta charset="utf-8">** que se utiliza para especificar la codificación de caracteres que se va a utilizar para visualizar la página. Esto permite al navegador reconocer caracteres propios de nuestro idioma, como las tildes o el carácter ñ.

- **<body>**: delimita el cuerpo del documento HTML. El cuerpo encierra todos los contenidos que se muestran al usuario (párrafos de texto, imágenes, tablas). En general, el **<body>** de un documento contiene cientos de etiquetas HTML, mientras que el **<head>** no contiene más que unas pocas.



2.4.2 Etiquetas obsoletas de HTML

De todas las etiquetas disponibles en HTML, las siguientes se consideran **obsoletas** y no se deben utilizar: **applet**, **basefont**, **center**, **dir**, **menú**, **marquee**, **strike**, **acronym**.

2.5 Comentarios

Los **comentarios** sólo se aparecen en el código fuente, no se muestran cuando el navegador procesa la página. No son etiquetas.

Un **comentario** comienza con **<!--** y se cierra con **-->**.

```
<!-- Este es mi primer comentario -->
```

2.6 Etiquetas básicas

2.6.1 Párrafo simple <p>

En una página web todo el texto está dentro de alguna etiqueta. Aunque en HTML 5 se permite escribir texto que no esté dentro de una etiqueta, no es una buena práctica, ya que no podríamos indicar más adelante el formato del texto (alineaciones, márgenes...).

Por ello, **no se debe poner texto directamente dentro de la etiqueta <body>**, tiene que haber algún contenedor (etiqueta) para ese texto.

El elemento más sencillo es **la etiqueta <p>** que indica un elemento de **párrafo** normal.

```
<p>Primer párrafo.</p>
<p>Segundo párrafo.</p>
```

Resultado:

Primer párrafo.

Segundo párrafo.

Por defecto los párrafos **<p>** hacen un **salto de línea** y dejan **espacio** arriba y abajo. **Dentro de un párrafo, no podemos indicar otro párrafo.** Pero si podemos tener otras etiquetas de texto simple: **** o ****.

Si ponemos más de un **espacio en un texto** es ignorado por el navegador, para ello tenemos caracteres especiales ** **

**2.6.2 Salto de línea
**

A veces es necesario dentro del texto contenido en un párrafo hacer un salto de línea. La etiqueta que se usa para esto es **
**.

```
<p>Primera línea <br> Segunda líneas. </p>
```

Resultado:

Primera línea

Segunda líneas.

Aparecerá cada texto en una línea distinta.

2.6.3 Línea horizontal <hr>

La etiqueta **<hr>** se utiliza para poner una línea horizontal que ocupa toda la pantalla. Automáticamente se inserta un espacio antes y después de la línea.

```
<p>Primar párrafo.</p>
<hr>
<p>Segundo párrafo.</p>
```

Resultado:

Primar párrafo.

Segundo párrafo.

2.6.4 Títulos o encabezados <hx>

Los títulos son encabezados similares a los de los artículos de periódicos, nos permiten dar diferente importancia a los contenidos de la página.

Para ello tenemos seis etiquetas que comienzan con la letra **h** a la que le sigue un número del 1 al 6. El elemento **h1** marcará títulos de primer nivel, **h2** de segundo nivel, ... y así hasta **h6** (sexto nivel).

```
<h1>Encabezado 1.</h1>  
<h2>Encabezado 2.</h2>  
<h3>Encabezado 3.</h3>  
<h4>Encabezado 4.</h4>  
<h5>Encabezado 5.</h5>  
<h6>Encabezado 6.</h6>
```

Resultado:

Encabezado 1.

Encabezado 2.

Encabezado 3.

Encabezado 4.

Encabezado 5.

Encabezado 6.

La etiqueta **<h1>** sirve para indicar a los buscadores las palabras clave para posicionar nuestros contenidos. En una página web es aconsejable que haya un solo titular (**h1**), en caso de haber más de uno, los buscadores penalizan y ponen la página en posiciones inferiores.

2.7 Elementos de texto

En este caso, detallaremos los elementos que se emplean para señalar texto interior a un párrafo.

2.7.1 Texto en negrita

En HTML 5 el texto importante (negrita) se indica con la etiqueta ****. En general todos los navegadores marcan el texto **strong** en negrita (letra más gruesa).

En versiones anteriores se utilizaba la etiqueta ****, actualmente no está recomendada, aunque es empleada para llamar la atención y no para dar importancia a un texto o palabra como si lo hace **strong**.

```
<p>Hola <strong>mundo!</strong></p>
```

Resultado:

Hola **mundo!**

**2.7.2 Texto enfatizado o cursiva **

La etiqueta **** marca texto enfatizado. El resultado visual en los navegadores es el texto en cursiva.

Antes se utilizaba **<i>**, actualmente no está recomendada.

```
<p>Hola <em>mundo!</em></p>
```

Resultado:

Hola *mundo!*

2.7.3 Texto resaltado <mark>

Para resaltar texto tenemos la etiqueta **<mark>**. En este caso, el efecto visual es un color de fondo amarillo que simula un subrayador.

```
<p>Yo soy texto normal. <mark>Y yo estoy marcado.</mark></p>
```

Resultado:

Yo soy texto normal. Y yo estoy marcado.

Este elemento no es tan antiguo como los anteriores porque apareció con HTML5 (los navegadores antiguos no lo reconocen). La idea semántica es marcar texto que queremos resaltar de forma muy llamativa indicando texto a recordar o sobre el que profundizar.

**2.7.4 Texto tachado **

En la versión de HTML 4 la etiqueta **<s>** quedó obsoleta, no se recomendaba su uso. Con HTML 5 se ha redefinido su uso para tachar texto con la etiqueta ****. En el navegador se muestra con efecto de tachado.

```
<p>La capital de Alemania en <del>Bee</del> Berlín.</p>
```

Resultado:

La capital de Alemania en ~~Bee~~ Berlín.

2.7.5 Texto subrayado <ins>

Para el texto subrayado se utiliza la etiqueta **<u>**. Esta etiqueta está deprecada, no se recomienda su uso. En HTML5 se utiliza la etiqueta **<ins>** para obtener el mismo efecto.

```
<p><ins>Este texto está subrayado.</ins></p>
```

Resultado:

Este texto está subrayado.

Nota: Para las etiquetas que permiten efectos visuales (tachado, negrita, cursiva, subrayado...) pueden conseguirse este mismo efecto con estilos CSS que veremos en temas siguientes.

Atención: Ya **NO** se recomienda utilizar el auto cierre (/>) en etiquetas sin etiqueta de cierre. Aunque se muestra correctamente, no se valida correctamente.

2.7.6 Texto subíndice <sub>

Permite que el texto aparezca por debajo de la línea base y en un tamaño más pequeño. Se utiliza la etiqueta **<sub>**.

```
<p>Texto<sub>subíndice</sub></p>
```

Resultado:

Textosubíndice

2.7.7 Texto superíndice <sup>

Parecida al anterior, pero ahora el texto por encima de la línea base y también en pequeño. La etiqueta es **<sup>**

```
<p>Texto<sup>superíndice</sup></p>
```

Resultado:

Textosuperíndice

2.7.8 Texto pequeño <small>

La etiqueta **<small>** se utiliza para escribir texto con **letra pequeña**. Se usa, por ejemplo, en textos que marquen copyright, derechos de uso, comentarios, notas anexas y letra pequeña en general.

El efecto visual es que el texto sale con un tamaño más pequeño. Es posible incluso anidar un elemento **small** dentro de otro, de modo que cada vez sale más

pequeño el texto.

```
<p>Texto normal. <small>Texto pequeño. <small>Texto aún más  
mequeño.</small></small></p>
```

Resultado:

Texto normal. Texto pequeño. Texto aún más mequeño.

2.7.9 Abreviatura <abbr>

La etiqueta **<abbr>** se utiliza para texto que indica una **abreviatura o un acrónimo**. Su uso requiere utilizar el atributo **title** para indicar el significado de la abreviatura. En general, los navegadores no muestran ningún resultado, pero sí un cartel con el contenido de **title** cuando se acerca el ratón.

```
<p>Se ha coronado a <abbr title="Su Alteza Real">SAR</abbr> El Rey Juan  
Pepe.</p>
```

Resultado:

Se ha coronado a SAR. El Rey Juan Pepe.

Su Alteza Real

2.7.10 Definiciones <dfn>

La etiqueta **<dfn>** sirve para indicar definiciones para un término o palabra. También necesita el atributo **title** para indicar el significado del término.

```
<p>Un dispositivo <dfn title="Componentes físicos de un sistema  
informático">Hardware.</dfn></p>
```

Resultado:

Un dispositivo *Hardware*.

Componentes físicos de un sistema informático

Los navegadores suelen mostrar el contenido del elemento en cursiva y al posicionar el ratón sobre la palabra muestran el contenido del atributo **title** en un cartelito “**tooltips**”.

2.8 Escritura de código y texto informático

2.8.1 Escritura de código <code>

HTML proporciona una etiqueta llamada **<code>** que se utiliza cuando en un documento deseamos escribir ejemplos de código escritos en un lenguaje de programación.

```
<p>Este es un ejemplo de código JavaScript:</p>
<code>
    window.onload=function(){
        alert("Hola");
    }
</code>
```

Resultado:

```
Este es un ejemplo de código JavaScript:

window.onload=function(){ alert("Hola"); }
```

Se puede observar cómo los navegadores modifican el tipo de letra (utilizan un tipo de letra monoespaciada) pero no respetan los espacios y tabuladores utilizados para indicar el código.

2.8.2 Texto preformateado <pre>

Como es muy normal querer respetar los espacios y tabuladores cuando se escriben ejemplos de código, lo habitual es combinar con la etiqueta **<pre>**.

La etiqueta **pre** representa un bloque de texto **preformateado** donde, los espacios continuos y los saltos de línea son respetados.

Ejemplo:

```
<p>Este es un ejemplo de código JavaScript:</p>
<pre>
    <code>
        window.onload=function(){
            alert("Hola");
        }
    </code>
</pre>
```

Resultado:

```
Este es un ejemplo de código JavaScript:

        window.onload=function(){
            alert("Hola");
        }
```

2.8.3 Resultados de operaciones en un ordenador `<samp>`

El elemento HTML Sample `<samp>` se utiliza para incluir texto en línea que representa una muestra (o cita) de la salida de un programa de ordenador. El contenido de esta etiqueta es renderizado generalmente usando la tipografía monoespaciada por defecto del navegador.

Ejemplo:

```
<p>Estaba tratando de arrancar mi computadora, pero recibí este divertido mensaje:</p>
<p><samp>Teclado no encontrado <br>Presione F1 para continuar</samp></p>
```

Este es el resultado:

```
Estaba tratando de arrancar mi computadora, pero recibí este divertido mensaje:

Teclado no encontrado
Presione F1 para continuar
```

2.9 Citas

Una **cita** es cuando se extraen fragmentos o ideas **textuales** de un texto. Existen varias formas de marcar citas en html5.

2.9.1 Cita en bloque `<blockquote>`

`<blockquote>` crea **citas en bloque**, marca las citas a otros autores o documentos. Es decir, indica párrafos completos que son citas procedentes de fuentes externas al documento. Se le puede el **atributo cite** para indicar **una URL** a la fuente de la cita.

- **NO genera automáticamente comillas alrededor del contenido.**
- **Incluye un margen izquierdo cuando se visualiza.**

Ejemplo:

```
<p>Esto es un ejemplo de cita en bloque.</p>
<blockquote cite='http://html.conclase.net/w3c/html401...def-BLOCKQUOTE'>
  <p><strong>Nota.</strong> Recomendamos que las implementaciones de
  hojas de estilo proporcionen un mecanismo para insertar signos de
  puntuación de citas antes y después de una cita delimitada por un
  BLOCKQUOTE de un modo apropiado según el contexto del idioma actual y
  el grado de anidamiento de las citas.</p>
</blockquote>
```


Resultado:

Esto es un ejemplo de cita en bloque.

Nota. Recomendamos que las implementaciones de hojas de estilo porporcionen un mecanismo para insertar signos de puntuación de citas antes y después de una cita delimitada por un BLOCKQUOTE de un modo apropiado según el contexto del idioma actual y el grado de anidamiento de las citas.

Puede utilizar el **atributo cite** (como se observa en el ejemplo), para **indicar la URL** que se ha utilizado como fuente de la cita.

2.9.2 Cita corta en línea <q>

<q> indica que el texto adjunto es una cita corta en línea. La mayoría de los navegadores modernos implementan esto **rodeando el texto entre comillas**. Este elemento está destinado a citas breves en la misma línea; para citas de bloque independiente, utiliza el elemento <blockquote>.

Ejemplo:

```
<p>Conforme al sitio web de Mozilla,  
<q cite="https://www.mozilla.org/en-US/about/history/details/">Firefox 1.0 fue  
lanzado en 2004 y se convirtió en un gran éxito. </q></p>
```

Resultado:

Conforme al sitio web de Mozilla, “Firefox 1.0 fue lanzado en 2004 y se convirtió en un gran éxito.”

Nota: La mayoría de los navegadores modernos **agregarán automáticamente comillas** alrededor del texto dentro de un elemento <q>. Es posible que se necesite una regla de estilo para agregar comillas en navegadores antiguos.

Aunque no es obligatorio utilizar el atributo **cite**, se puede usar para indicar una fuente para consultar la cita original (los navegadores, sin embargo, no hacen nada con el contenido de este enlace).

2.9.3 Cita de trabajo creativo <cite>

<cite> es uno de los pocos identificadores de HTML que se emplea tanto como etiqueta como atributo. Como etiqueta, <cite> se emplea para referenciar un trabajo creativo como un libro, una película, un programa de televisión o una canción.

Ejemplo:

```
<p>Yo llegué en el momento justo, interesado, cuando el hipertexto e Internet  
habían visto ya la luz. La tarea que me correspondía era hacer que  
casaran.<br>Tim Berners-Lee, <cite>Tejiendo la Web</cite>.</p>
```

Resultado:

Yo llegué en el momento justo, interesado, cuando el hipertexto e Internet habían visto ya la luz. La tarea que me correspondía era hacer que casaran.
Tim Berners-Lee, *Tejiendo la Web*.

2.10 Otras etiquetas

Existen otras muchas etiquetas de las cuales podemos destacar:

2.10.1 Etiqueta de direcciones <address>

La etiqueta **<address>** se utiliza para indicar información de contacto, como una dirección postal, una dirección de correo electrónico, un número telefónico, una ciudad, un país, un código postal, etc.

Los navegadores suelen mostrar este tipo de párrafos en **cursiva**, y aunque el elemento representa el texto con el mismo estilo que los elementos <i> o , es más apropiado utilizarlo cuando se trata información de contacto, ya que **transmite información semántica adicional a los buscadores**.

Ejemplo:

```
<address>
IES Castelar <br>
Av. Santiago Ramón y Cajal, 2 <br>
06001 <br>
Badajoz <br>
España <br>
</address>
```

Resultado:

IES Castelar
Av. Santiago Ramón y Cajal, 2
06001
Badajoz
España

2.11 Listas

Las listas permiten crear datos agrupados y alineados mediante símbolos como viñetas o números para facilitar la lectura y organización de la información.

2.11.1 Listas NO ordenadas (viñetas)

Las listas con viñetas se deben englobar dentro de una etiqueta **** (acrónimo de *unordered list*, lista no ordenada), después cada ítem de la lista estará dentro de elementos de tipo **** (de *list item*, elemento de lista).

Ejemplo:

```
<h3>Lista de la compra</h3>
<ul>
<li>Agua</li>
```

```
<li>Pan</li>
<li>Huevos</li>
<li>Leche</li>
</ul>
```

Resultado:

Lista de la compra

- Agua
- Pan
- Huevos
- Leche

Atributos específicos		
Atributo	Valor	Descripción
type	Uno de los siguientes: 'disc' , 'square' , o 'circle' . Por defecto: depende del anidamiento, lo fija el navegador.	Indica el estilo de los items de la lista.
compact	Solo puede ser: 'compact' . Por defecto: Lo fija el navegador.	Indica que la lista debe mostrarse compactada.

Nota: Se recomienda no usar el atributo type, se debe emplear CSS. Si se usa, no se valida el documento. Lo usaremos temporalmente para visualizar su efecto en las listas.

2.11.3 Listas ordenadas (numéricas)

Las listas numéricas aparecen dentro del elemento **** (de *ordered list*, lista ordenada), después cada ítem de la lista estará dentro de elementos de tipo ****, al igual que las anteriores. La diferencia ahora es que cada párrafo con ****, aparece con un número y no con una viñeta.

Ejemplo:

```
<h3>Lista de la compra</h3>
<ol>
  <li>Agua</li>
  <li>Pan</li>
  <li>Huevos</li>
  <li>Leche</li>
</ol>
```

Resultado:

Lista de la compra

1. Agua
2. Pan
3. Huevos
4. Leche

Atributos específicos		
Atributo	Valor	Descripción
type	(1 a A i I)	Indica el estilo de los items de la lista.
start	'un número'	Número inicial de la secuencia
compact	Solo puede ser: ' compact '. Por defecto: Lo fija el navegador.	Indica que la lista debe mostrarse compactada.
reversed	reversed	Invierte el orden de la lista.

2.11.4 Listas anidadas

Es posible meter una lista dentro de otra, por ejemplo:

Ejemplo:

```
<h3>Lista de la compra</h3>
<ul>
  <li>Bebidas
    <ul>
      <li>Agua</li>
    </ul>
  </li>
  <li>Otros
    <ul>
      <li>Pan</li>
      <li>Huevos</li>
      <li>Leche</li>
    </ul>
  </li>
</ul>
```

Resultado:

Lista de la compra

- Bebidas
 - Agua
- Otros
 - Pan
 - Huevos
 - Leche

También es posible anidar mezclando tipos de listas:

Ejemplo:

```
<h3>Lista de la compra</h3>
<ol>
  <li>Bebidas
    <ul>
      <li>Agua</li>
    </ul>
  </li>
  <li>Otros
    <ul>
      <li>Pan</li>
      <li>Huevos</li>
      <li>Leche</li>
    </ul>
  </li>
</ol>
```

Resultado:

Lista de la compra

1. Bebidas
 - Agua
2. Otros
 - Pan
 - Huevos
 - Leche

2.11.5 Listas de términos <dl> <dt> <dd>

Permite crear una lista de definiciones de términos. En ellas se indica el término a definir y su significado. Para ello usamos las etiquetas **<dl>** **<dt>** y **<dd>**

Ejemplo:

```
<h3>Lista de vehículos</h3>
<dl>
  <dt>Coche:</dt>
  <dd>Vehículo de cuatro ruedas.</dd>

  <dt>Moto:</dt>
  <dd>Vehículo de dos ruedas.</dd>
</dl>
```

Resultado:

Lista de vehículos

Coche:

Vehículo de cuatro ruedas.

Moto:

Vehículo de dos ruedas.

2.12 Enlaces <a>

Una **URL (Uniform Resource Location)** es la **dirección única y específica a un recurso** en **Internet** (una página web, una imagen, un vídeo, un directorio...).

Un **enlace o hipervínculo**, es un texto o imagen en el que el usuario puede pulsar para ir a otro contenido. Su función es permitirnos navegar por las diferentes páginas que tengamos.

La etiqueta que permite realizar **enlaces** es la etiqueta **<a>**.

Atributos específicos		
Atributo	Valor	Descripción
href	URL o enlace a recurso.	URL o dirección hacia un recurso.
target	_self : Carga el recurso en la misma pestaña de navegación que el actual. Este es el comportamiento por defecto. _blank : Carga el recurso en una nueva pestaña del navegador.	Especifica en donde abrir el recurso enlazado.

2.12.1 Enlaces globales

Son los enlaces que nos llevan a una URL.

Ejemplo:

```
<p>Visita mi web <a href="https://tuelectronica.es"
target="_blank">TuElectronica.es</a>.</p>
```

Resultado:

Visita mi web TuElectronica.es.

2.12.2 Enlaces locales

Muchos enlaces de un sitio web en Internet no saltan a otros sitios, sino que son **saltos a recursos del mismo servidor web**.

En ese caso no se indica una URL global tal cual se explicó anteriormente, sino que se indica un salto local relativo. Es decir, un salto a un recurso en nuestro propio servidor web que toma **como punto de partida el directorio en el que se encuentra la página web que realiza el salto**. Para ello se usarán **rutasy locales**.

Ejemplo:

```
<ul>
  <!-- Abrimos el recurso pagina1.html que está en el directorio -->
  <li><a href="pagina1.html" target="_blank">Pagina 1</a></li>
  <!-- Abrimos el recurso pagina2.html que está en la carpeta "carpeta" del
  directorio de este HTML -->
  <li><a href="carpeta/pagina2.html" target="_blank">Pagina 2</a></li>
</ul>
```

2.12.3 Enlaces internos

Hay un tipo de enlace que permite posicionarnos en un **lugar concreto del documento**. Este tipo de enlace se denomina interno, ya que salta dentro del propio documento.

Este salto requiere **primero marcar la posición a la que deseamos saltar**. Para realizar esto se utiliza el atributo **id**. Después se crea un enlace a esa posición.

El **id** es un atributo presente en cualquier elemento de la página web. Sirve para identificar a dicho elemento dentro de la página, por ello no podemos repetir valor.

Por ejemplo, al hacer clic en el enlace que se indica, el navegador hará un scroll hasta llegar al id (en este caso el título H1) que se ha referenciado en el enlace mediante el uso de #.

HTML

HTML, siglas en inglés de *HyperText Markup Language* ('lenguaje de marcado de hipertexto'), hace referencia al [lenguaje de marcado de páginas web](#). Es un estándar que sirve de referencia del software que conecta con la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código (denominado código HTML) para la definición de contenido de una página web, como texto, imágenes, videos, juegos, entre otros. Es un estándar a cargo del [World Wide Web Consortium \(W3C\)](#) o Consorcio WWW, organización dedicada a la estandarización de casi todas las tecnologías ligadas a la web, sobre todo en lo referente a su escritura e interpretación. HTML se considera el lenguaje web más importante siendo su invención crucial en la aparición, desarrollo y expansión de la [World Wide Web](#) (WWW). Es el estándar que se ha impuesto en la visualización de páginas web y es el que todos los navegadores actuales han adoptado.¹

El lenguaje HTML basa su filosofía de desarrollo en la diferenciación. Para añadir un elemento externo a la página (imagen, video, [script](#), entre otros.), este no se incrusta directamente en el código de la página, sino que se hace una referencia a la ubicación de dicho elemento mediante texto. De este modo, la página web contiene solamente texto mientras que recae en el navegador web (interpretador del código) la tarea de unir todos los elementos y visualizar la página final. Al ser un estándar, HTML busca ser un lenguaje que permita que cualquier página web escrita en una determinada versión, pueda ser interpretada de la misma forma (estándar) por cualquier navegador web actualizado.

[HTML](#) es un lenguaje de marcado que nos permite indicar la estructura de nuestro documento mediante etiquetas. Este lenguaje nos ofrece una gran adaptabilidad, una estructuración lógica y es fácil de interpretar tanto por humanos como por máquinas.

2.12.4 URLs absolutas

Además de los links internos y externos, la otra característica que diferencia a los enlaces (y, por tanto, también a las URL) es si el enlace es absoluto o relativo.

Las URL absolutas incluyen todas las partes de la URL (protocolo, servidor y ruta) por lo que no se necesita más información para obtener el recurso enlazado.

2.12.5 URLs relativas

Las URL relativas prescinden de algunas partes de las URL para hacerlas más breves. **Como se trata de URL incompletas, es necesario** disponer de información adicional para obtener el recurso enlazado. En concreto, para que una URL relativa sea útil es imprescindible **conocer la URL del origen del enlace**.

Si queremos indicarle al navegador que el recurso está en un nivel superior, se incluyen dos puntos y una barra (`../`) en la ruta del recurso enlazado.

Elemento	Valor
Página origen	<code>https://www.dominio.com/curso/seo/introduccion/curso_SEO_parte_1.html</code>
Página enlazada	Página web llamada <code>examen_1.html</code> y que se encuentra en el directorio superior llamado <code>/seo</code>
URL absoluta	<code>https://www.dominio.com/curso/seo/examen_1.html</code>

Elemento	Valor
URL relativa	<code>../examen_1.html</code>

Si el destino se encuentra un par de niveles por encima debemos incluir «../» dos veces seguidas:

Elemento	Valor
Página origen	<code>https://www.dominio.com/curso/seo/introduccion/curso_SEO_parte_1.html</code>
Página enlazada	Página web llamada <code>inscripcion_curso_SEO.html</code> y que se encuentra en el directorio superior llamado <code>/curso</code>
URL absoluta	<code>https://www.dominio.com/curso/inscripcion_curso_SEO.html</code>
URL relativa	<code>../../inscripcion_curso_SEO.html</code>

Además de que se pueden subir niveles, también **se puede entrar en otros directorios diferentes** para obtener los propios recursos:

Elemento	Valor
Página origen	<code>https://www.dominio.com/curso/seo/introduccion/curso_SEO_parte_1.html</code>
Página enlazada	Página web llamada <code>curso_wordpress.html</code> y que se encuentra en un directorio llamado <code>/wordpress</code> que se encuentra en la raíz del servidor
URL absoluta	<code>https://www.dominio.com/wordpress/curso_wordpress.html</code>
URL relativa	<code>../../..../wordpress/curso_wordpress.html</code>

Si el recurso enlazado se encuentra en algún **directorio inferior** al que se encuentra el origen, solo es necesario indicar el nombre de los directorios a los que debe entrar el navegador.

Elemento	Valor
Página origen	<code>https://www.dominio.com/curso/seo/introduccion/curso_SEO_parte_1.html</code>
Página enlazada	Página web llamada <code>examen_1.html</code> y que se encuentra en un directorio inferior llamado <code>/examen</code>
URL absoluta	<code>https://www.dominio.com/curso/seo/introduccion/examen/examen_1.html</code>
URL relativa	<code>/examen/examen_1.html</code>

2.13 Imágenes

Las imágenes son fundamentales para que una página web sea más atractiva. Son el primer contenido no textual que se planteó poder incorporar al estándar HTML. Prácticamente no hay páginas sin imágenes.

Los navegadores tienen capacidad de mostrar imágenes, pero sólo las que pertenecen a tipos concretos. Los tipos de imágenes más usados y reconocidos por la mayoría de los navegadores son:

- Formato jpg. Son imágenes que ocupan muy poco espacio gracias a su alta compresión. No admiten animaciones ni zonas marcadas con transparencia. Son el formato habitual de la imagen digital en una página web.
- Formato gif. Imágenes con hasta 256 colores, por lo que son buenas para dibujos y logotipos, pero no para fotografías. Pueden incluso contener animaciones (uno de sus usos habituales).
- Formato png. Imágenes fotográficas comprimidas al estilo de jpg pero con más calidad (ocupan más espacio, normalmente) y transparencia alfa.
- Formato webp. Formato de imagen desarrollado por Google para conseguir mejorar a PNG y JPG haciendo que ocupen menos sin pérdida de calidad respecto a estos formatos. Sin embargo, no todos los navegadores son capaces de interpretarlas.

Las imágenes se colocan mediante la etiqueta ****. El atributo **src** indica la ruta de la imagen y es obligatorio. De igual forma, el atributo **alt** es necesario para validar el documento. Este atributo permite mostrar un texto alternativo si la imagen no puede cargarse.

Ejemplo:

```

```

Resultado:



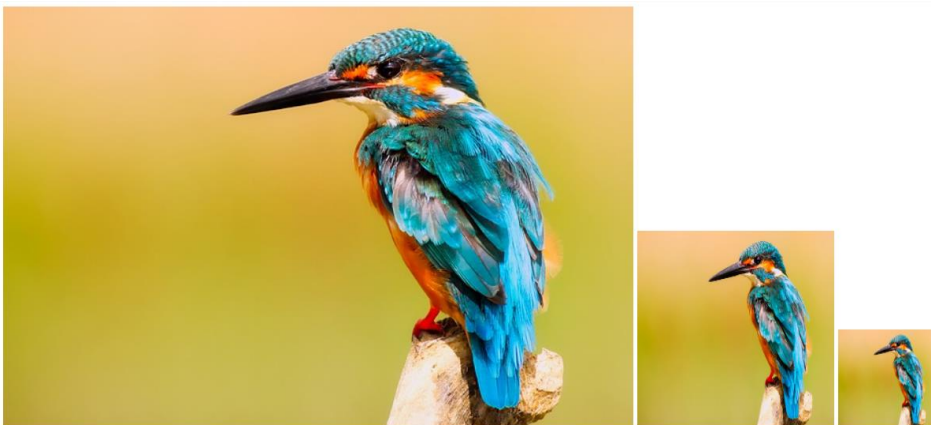
Atributos específicos		
Atributo	Valor	Descripción
alt	Obligatorio. Texto	Indica un texto alternativo. Ese texto aparece cuando la imagen no se ha podido cargar.
width	Números enteros no negativos. Pixels	Anchura de la imagen.
height	Números enteros no negativos. Pixels	Altura de la imagen.

Ejemplo:

```
  
  

```

Resultado:



2.14 Tablas <table> <tr> <td>

Las tablas son uno de los elementos fundamentales de HTML para mejorar la puesta visual de las páginas web. Una tabla es un conjunto de datos presentados en forma de filas y columnas. Una tabla típica podría ser esta:

Company	Contact	Country
Alfreds Futterkiste	Maria Anders	Germany
Centro comercial Moctezuma	Francisco Chang	Mexico
Ernst Handel	Roland Mendel	Austria
Island Trading	Helen Bennett	UK
Laughing Bacchus Winecellars	Yoshi Tannamuri	Canada
Magazzini Alimentari Riuniti	Giovanni Rovelli	Italy

En HTML las tablas tienen muchas posibilidades, lo que ha permitido, tradicionalmente, ser utilizadas como elemento de maquetación de páginas complejas, hoy en día, ya no se usan para maquetación. Para este fin se utilizan las etiquetas <div>.

2.14.1 Creación de tablas simples

En HTML una tabla se crea con la etiqueta <table>, dentro de este elemento se indican las **filas mediante el elemento <tr>** y dentro de cada fila se indican las **celdas mediante elementos <td>**.

Ejemplo:

```
<table>
  <tr>
    <td></td>
    <td>Lunes</td>
    <td>Martes</td>
    <td>Miércoles</td>
  </tr>
  <tr>
    <td>10:30</td>
    <td>Seguridad informática</td>
    <td>Hacking ético</td>
    <td>Gestión de incidentes</td>
  </tr>
  <tr>
    <td>11:30</td>
    <td>Lenguaje de marcas</td>
    <td>Hacking ético</td>
    <td>Base de datos</td>
  </tr>
</table>
```

Resultado:

Lunes	Martes	Miércoles
10:30 Seguridad informática	Hacking ético	Gestión de incidentes
11:30 Lenguaje de marcas	Hacking ético	Base de datos

Atributos específicos		
Atributo	Valor	Descripción
border	Longitud de píxeles.	Permite poner un borde a la tabla
width	Longitud de píxeles o un porcentaje.	Permite poner un ancho a la tabla.
height:	Longitud de píxeles o un porcentaje.	Permite poner un alto a la tabla.

Nota: Se recomienda no usar estos atributos, se debe emplear CSS. Si se usan, no se valida el documento. Lo usaremos temporalmente para visualizar mejor las tablas.

2.14.2 Encabezados

Es muy habitual que las tablas muestren datos y que estos tengan celdas que sirvan de título. Esas celdas se consideran encabezados y se crean con **<th>**. El navegador muestra su contenido en negrita y centrado en la celda.

Ejemplo:

```
<table border="1px">
  <tr>
    <th></th>
    <th>Lunes</th>
    <th>Martes</th>
    <th>Miércoles</th>
  </tr>
  <tr>
    <td>10:30</td>
    <td>Seguridad informática</td>
    <td>Hacking ético</td>
    <td>Gestión de incidentes</td>
  </tr>
  <tr>
    <td>11:30</td>
    <td>Lenguaje de marcas</td>
    <td>Hacking ético</td>
    <td>Base de datos</td>
  </tr>
</table>
```

Resultado:

	Lunes	Martes	Miércoles
10:30	Seguridad informática	Hacking ético	Gestión de incidentes
11:30	Lenguaje de marcas	Hacking ético	Base de datos

2.14.3 Títulos de las tablas.

A las tablas se les puede poner un título con la etiqueta **<caption>**.

Ejemplo:

```
<table border="1px">
  <caption>Horario</caption>
  <tr>
    <th></th>
    <th>Lunes</th>
    <th>Martes</th>
    <th>Miércoles</th>
  </tr>
  <tr>
    <td>10:30</td>
    <td>Seguridad informática</td>
    <td>Hacking ético</td>
    <td>Gestión de incidentes</td>
  </tr>
  <tr>
    <td>11:30</td>
    <td>Lenguaje de marcas</td>
    <td>Hacking ético</td>
    <td>Base de datos</td>
  </tr>
</table>
```

Resultado:

	Lunes	Martes	Miércoles
10:30	Seguridad informática	Hacking ético	Gestión de incidentes
11:30	Lenguaje de marcas	Hacking ético	Base de datos

2.14.4 Agrupación de filas

Hay tres elementos HTML que sirven para diferenciar las tres partes principales de una tabla. Son los siguientes:

- **<thead>**. Sirve para indicar las filas que forman la cabecera de la tabla.
- **<tfoot>**. Indica el pie de la tabla.

- **<tbody>**. Indica el cuerpo de la tabla.

El uso de estos elementos permite dar con CSS diferente formato a cada parte.

Ejemplo:

```
<table border="1px">
  <thead>
    <tr>
      <th></th>
      <th>Lunes</th>
      <th>Martes</th>
      <th>Miércoles</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>10:30</td>
      <td>Seguridad informática</td>
      <td>Hacking ético</td>
      <td>Gestión de incidentes</td>
    </tr>
    <tr>
      <td>11:30</td>
      <td>Lenguaje de marcas</td>
      <td>Hacking ético</td>
      <td>Base de datos</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td>Total</td>
      <td>2</td>
      <td>1</td>
      <td>2</td>
    </tr>
  </tfoot>
</table>
```

Resultado:

	Lunes	Martes	Miércoles
10:30	Seguridad informática	Hacking ético	Gestión de incidentes
11:30	Lenguaje de marcas	Hacking ético	Base de datos
Total	2	1	2

2.14.5 Combinar celdas

Es posible **unir celdas**. Las etiquetas de columna (td y th) son las que poseen los atributos que permiten esta operación. En concreto son los atributos:

Atributos específicos		
Atributo	Valor	Descripción
colspan	Por ejemplo, colspan="3". Une esta celda con las dos que tiene a su derecha, formando una combinación de tres celdas en horizontal	Combina la celda actual con e. número de celdas a la derecha que se indique.
rowspan	Por ejemplo, rowspan="3". Une esta celda con las dos que tiene hacia abajo, formando una combinación de tres celdas en vertical.	Combina la celda actual con e. número de celdas hacia abajo que se indique.

Ejemplo:

```
<h1>Celdas combinadas horizontal</h1>
<table border="1px" width="250px">
  <tr>
    <td colspan="3">A</td>
    <td>B</td>
  </tr>
  <tr>
    <td>C</td>
    <td colspan="2">D</td>
    <td>E</td>
  </tr>
  <tr>
    <td colspan="4">F</td>
  </tr>
  <tr>
    <td>G</td>
    <td>H</td>
    <td>I</td>
    <td>J</td>
  </tr>
</table>
```


Resultado:

Celdas combinadas horizontal

A			B	
C	D			E
F				
G	H	I	J	

Ejemplo:

```
<h1>Celdas combinadas vertical</h1>
<table border="1px" width="250px">
  <tr>
    <td>A</td>
    <td>B</td>
    <td rowspan="3">C</td>
    <td>D</td>
  </tr>
  <tr>
    <td rowspan="2">E</td>
    <td>F</td>
    <td rowspan="3">G</td>
  </tr>
  <tr>
    <td>H</td>
  </tr>
  <tr>
    <td>I</td>
    <td>J</td>
    <td>K</td>
  </tr>
</table>
```

Resultado:

Celdas combinadas vertical

A	B	C	D
E	F		G
	H		
I	J	K	

2.14.6 Tablas dentro de tablas

Para generar tablas aún más complejas, se pueden meter elementos `<table>`, dentro de otras tablas. Lo que se hace es meter una etiqueta `<table>` con todos sus elementos de fila y columna dentro de un elemento `<td>` o `<th>`, **NUNCA dentro de un `<tr>`**.

Ejemplo:

```
<h1>Tabla dentro de tabla</h1>
<table border="1px" width="500px">
  <tr>
    <td>A</td>
    <td>B</td>
    <td rowspan="3">C</td>
    <td>D</td>
  </tr>
  <tr>
    <td rowspan="2">E</td>
    <td>F</td>
    <td rowspan="3">
      <table border="1px" width="100%">
        <tr>
          <td colspan="3">A</td>
          <td>B</td>
        </tr>
        <tr>
          <td>C</td>
          <td colspan="2">D</td>
          <td>E</td>
        </tr>
        <tr>
          <td colspan="3">F</td>
        </tr>
        <tr>
          <td>G</td>
          <td>H</td>
          <td>I</td>
          <td>J</td>
        </tr>
      </table>
    </td>
  </tr>
  <tr>
    <td>I</td>
    <td>J</td>
    <td>K</td>
  </tr>
</table>
```

```

        </table>

    </td>
</tr>
<tr>
    <td>H</td>
</tr>
<tr>
    <td>l</td>
    <td>J</td>
    <td>K</td>
</tr>
</table>

```

Resultado:

Tabla dentro de tabla

A	B	C	D			
E	F		A		B	
	H		C	D	E	
I	J	K	F			
			G	H	I	J

2.15 Formularios <form>

Los formularios son un elemento cuya labor es recoger información del usuario. **La función habitual de un formulario es enviar esa información a un programa para que la procese adecuadamente.**

Por ejemplo, podemos pedir al usuario su nombre y contraseña y enviar esa información a un programa PHP.

En HTML5 se han mejorado muchísimo las posibilidades de los formularios. Una vez más, el problema es que hay características que no todos los navegadores soportan.

La página <https://www.wufoo.com/html5/> presenta una tabla para saber, sobre cada nueva característica de los formularios HTML5 qué navegadores la soportan (e incluso desde qué versión). También podemos consultarlo en la página: <https://caniuse.com/>

2.15.1 Métodos de paso de datos

Hay dos métodos de envío de datos desde un formulario:

- **GET**, los datos se envían mediante la URL según el formato:

nombre_programa.php?variable1=valor1&variable2=valor2&...

Este formato implica la separación entre el nombre del programa y el conjunto de pares variable - valor mediante un signo "?", y la separación dentro del dicho conjunto de las distintas variables con sus valores con el símbolo "&".

Este método plantea un **gran problema de seguridad** ya que cualquier usuario puede ver todos los datos enviados. Pensemos en sistemas que envíen nombres de usuarios o contraseñas de esta forma, serían claramente inseguros.

Además, otro problema que se presenta con este método es la **restricción en la longitud de la URL**, con lo cual en formularios que envíen gran cantidad de información, no podría realizarse el envío de la información completa.

- **POST**, los datos no son visibles en la URL. Los datos se envían al programa de forma oculta al usuario. Esto no significa que la información se envíe encriptada.

2.15.2 Elemento formulario

Todo formulario HTML comienza con una etiqueta **<form>**, dentro de ella se colocan todos los controles del formulario encargados de recabar los datos del usuario.

Ejemplo:

```
<form action="directorio/programa" method="get o post">
    ...
</form>
```

Atributos específicos		
Atributo	Valor	Descripción
action	URL.	Se trata de la URL de la página o programa al que se enviarán los datos para su procesamiento.
method	get o post.	Método utilizado para el envío de datos.

Se puede utilizar la siguiente URL para probar el envío de datos:

<https://jkorpele.fi/cgi-bin/echo.cgi>

Los formularios tienen otros atributos específicos que vale la pena mencionar.

Otros atributos específicos		
Atributo	Valor	Descripción
autocomplete	Booleano.	Si se activa, el navegador automáticamente completará los datos del formulario basándose en los valores que el usuario haya rellenado en otros formularios.
novalidate	Booleano	Los navegadores compatibles con HTML5 tienen la capacidad de validar algunos datos (por ejemplo, en los cuadros numéricos valida que el usuario realmente ha escrito un número).
target	_self : Carga el procesamiento del formulario en la misma pestaña de navegación que el actual. Este es el comportamiento por defecto. _blank : Carga el procesamiento del formulario en una nueva pestaña del navegador.	Especifica en donde abrir el procesamiento del formulario.

2.15.3 Elemento <input>

La mayoría de los controles de un formulario se inserta mediante la etiqueta **<input>**. El atributo principal es **type** que indica el tipo de control deseado.

Atributos específicos		
Atributo	Valor	Descripción

pattern	Expresión regular.	Especifica una expresión regular con la que se comprueba el valor del campo de entrada cuando se envía el formulario.
placeholder	Texto, numérico.	Especifica una sugerencia breve que describe el valor esperado de un campo de entrada (un valor de muestra o una breve descripción del formato esperado).
type		Tipo de control.
name		Atributo fundamental que da un nombre al control. Este nombre es el que se le pasa al servicio receptor de los datos del formulario.
size	Numero de caracteres.	Tamaño en caracteres que tendrá el control.
maxlength	Numero de caracteres.	Tamaño máximo de caracteres.
value		Valor que tendrá por defecto el control.
disabled		El control se deshabilita. El contenido del controlador no se envía para ser procesado.
readonly		No se permite la edición.
autocomplete		Permite al navegador autocompletar el control en función de la información que posee del usuario.
form		Recibe el identificador del formulario al que pertenece el control. Se usa cuando el elemento está fuera de La etiqueta form.
required		Hace que el elemento sea de obligado rellenado, es decir, no se puede dejar vacío, el usuario tendrá que darle algún valor.
id		Identificador único. Se recomienda su uso para poder referenciar las etiquetas <label> asociadas a cada <input>.

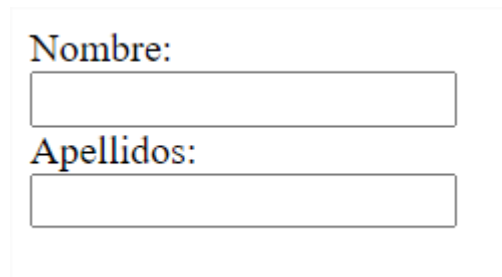
Existe la etiqueta **<label>** que acompaña a los <input> para describir que dato se requiere del usuario. El atributo **for** de la etiqueta <label> debe ser igual al id del

<input> al cual está asociado.

Ejemplo:

```
<form action="https://jkorpele.fi/cgi-bin/echo.cgi" method="post">
  <label for="fname">Nombre:</label><br>
  <input type="text" id="fname" name="fname"><br>
  <label for="lname">Apellidos:</label><br>
  <input type="text" id="lname" name="lname">
</form>
```

Resultado:



Nota: se recomienda usar siempre los atributos id y name en los inputs.

2.15.3.1 Campos de texto type="text"

Los cuadros de texto usan el atributo type con valor text. Permiten recoger información escrita por el usuario. Su sintaxis básica es:

```
<input type="text" id="nombre" name="nombre">
```

Además, podemos usar los atributos comunes a los elementos <input> comentados anteriormente.

```
<form action="https://jkorpele.fi/cgi-bin/echo.cgi" method="post">
  <label for="nombre">Nombre:</label><br>
  <input type="text" id="nombre" name="nombre" size="50" maxlength="5"
  required><br>
  <label for="codigo">Código:</label><br>
  <input type="text" id="codigo" name="codigo" value="1234"
  readonly><br><br>
  <input type="submit" value="Submit">
</form>
```

Resultado:



2.15.3.2 Campos de contraseñas type="password"

Funcionan como los cuadros de texto, solo que el texto que se introduce se oculta, mostrando solo puntos o asteriscos. Se usan, lógicamente, para introducir contraseñas.

La sintaxis es:

```
<input type="password" id="pcode" name=" pcode">
```

Resultado:



Usa los mismos atributos que los cuadros de texto. **Si usamos método GET, la contraseña es visible en la parte superior del navegador.**

2.15.3.3 Botones

Los botones son controles del formulario en los que no se puede escribir, sino que, simplemente, reciben un clic de ratón para hacer algo como enviar datos o borrar datos.

- **Botón de envío:**

Sirve para llevar a cabo la comunicación entre el formulario y la página que recoge sus datos. En cuanto se pulsa este botón, los datos del resto de controles se envían a la página receptora del formulario.

La sintaxis básica es:

```
<input type="submit" value="Enviar">
```

- **Botón de restablecer:**

Este tipo de botón lo que hace es borrar los datos que se han introducidos en los controles del formulario. Deja todos los controles en su estado inicial. La sintaxis básica es:

```
<input type="reset" value="Limpiar">
```

2.15.3.4 Opción única type="radio"

Se trata de un control fácil de usar que permite elegir una de entre varias opciones. Todas las opciones deben de tener el mismo nombre y solo cambia el valor (ese valor es el que se manda).

El atributo **checked** hace que el botón en el que se usa aparezca chequeado por defecto.

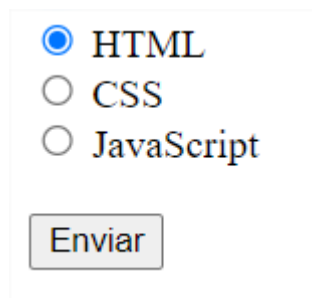
Ejemplo:

```
<form action="https://jkorpele.fi/cgi-bin/echo.cgi" method="post">
  <input type="radio" id="html" name="fav_language" value="HTML" checked>
  <label for="html">HTML</label><br>
```



```
<input type="radio" id="css" name="fav_language" value="CSS">
<label for="css">CSS</label><br>
<input type="radio" id="javascript" name="fav_language" value="JavaScript">
<label for="javascript">JavaScript</label><br><br>
<input type="submit" value="Enviar">
</form>
```

Resultado:



2.15.3.5 Campos numéricos type="number"

Sirven para introducir números (decimales o no). Los navegadores que reconocen este tipo de control presentan un cuadro de texto con botones para subir y bajar el valor del número (el incremento o decremento que hacen depende del atributo step) y además no permiten introducir datos no numéricos (salvo que se use el atributo novalidate).

Hay una serie de atributos reservados a este tipo de cuadros:

Atributos específicos		
Atributo	Valor	Descripción
max		Establece el máximo valor que se puede introducir.
min		Establece el mínimo valor que se puede introducir.
step		Valor de incrementos de los números. Por defecto 1.

Ejemplo:


```
<form action="https://jkorpele.fi/cgi-bin/echo.cgi" method="post">
  <label for="edad">¿Cuál es tu edad?</label><br>
  <input type="number" id="edad" name="edad" value="20" max="99"
  min="18"><br><br>
  <input type="submit" value="Enviar">
</form>
```

Resultado:

¿Cual es tu edad?

Si se establece un valor fuera del rango de min o max no se permite el envío:

¿Cual es tu edad?

 El valor debe ser superior o igual a 18

2.15.3.6 Campo de fecha type="date"

Permite seleccionar una fecha. Los navegadores que reconocen este tipo de control permiten pulsar en un botón que invoca a un calendario para elegir cómodamente la fecha.

También admite min, max y step.

El formato para el atributo value, min o max es **año-mes-día**

Ejemplo:

```
<form action="https://jkorpele.fi/cgi-bin/echo.cgi" method="post">
  <label for="cumple">Cumpleaños:</label>
  <input type="date" id="cumple" name="cumple" value="1987-02-02">
  <input type="submit" value="Enviar">
</form>
```

Resultado:

Cumpleaños:

febrero de 1987

L	M	X	J	V	S	D
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	1
2	3	4	5	6	7	8

 Borrar Hoy

2.15.3.7 Campo de hora type="time"

Permite seleccionar una hora, minutos y segundos. También admite los atributos min, max y step (segundos).

Ejemplo:

```
<form action="https://jkorpela.fi/cgi-bin/echo.cgi" method="post">
  <label for="appt">Selecciona una hora:</label>
  <input type="time" id="appt" name="appt">
  <input type="submit" value="Enviar">
</form>
```

Resultado:

Selecciona una hora: -- : -- ⌚

2.15.3.8 Casillas de verificación type="checkbox"

Se usan igual que los botones de radio, pero en este caso están pensadas para ser independientes unas de otras. Es decir, **se pueden activar varias a la vez**.

El atributo **checked** de los botones de radio se puede usar en las casillas de verificación con la misma finalidad, para que inicialmente la casilla aparezca marcada.

Ejemplo:

```
<form action="https://jkorpela.fi/cgi-bin/echo.cgi" method="post">
  <input type="checkbox" id="html" name="fav_language" value="HTML"
checked>
  <label for="html">HTML</label><br>
  <input type="checkbox" id="css" name="fav_language" value="CSS">
  <label for="css">CSS</label><br>
  <input type="checkbox" id="javascript" name="fav_language"
value="JavaScript">
  <label for="javascript">JavaScript</label><br><br>
  <input type="submit" value="Enviar">
</form>
```

Resultado:

☒ HTML
☐ CSS
☐ JavaScript

2.15.3.9 Campo de email type="email"

Funcionan como los cuadros de texto, solo que el texto que se introduce se es verificado por el navegador.

La sintaxis es:

```
<form action="https://jkorpela.fi/cgi-bin/echo.cgi" method="post">
```


```
<label for="email">Email:</label>
<input type="email" id="email" name="email">
<input type="submit" value="Enviar">
</form>
```

Resultado:

Email:

Si el texto introducido no tiene un formato de email (abcd@dominio.com), no se permite el envío del formulario.

Email:

 Incluye un signo "@" en la dirección de correo electrónico. La dirección "javier" no incluye el signo "@".

2.15.3.10 Campo oculto type="hidden"

Permite ocultar al usuario el contenido de un input. Es muy empleado para enviar información de sesión al servidor para prevenir problemas de seguridad. Funcionan como los campos de texto.

Debido a que la información se oculta al usuario, no se acompaña con un <label>.

La sintaxis es:

```
<form action="https://jkorpela.fi/cgi-bin/echo.cgi" method="post">
  <input type="hidden" id="token" name="token" value="1234567890">
  <input type="submit" value="Enviar">
</form>
```

Resultado:

Tras enviar el formulario:

Echoing submitted form data

token	1234567890
-------	------------

Processed 2022-11-07T17:43Z

2.15.3.11 Otros campos

Existen una multitud de tipos de inputs más específicos que se pueden consultar en la siguiente URL:

<https://developer.mozilla.org/es/docs/Web/HTML/Element/input>

2.16 Cuadro de texto multilínea <textarea>

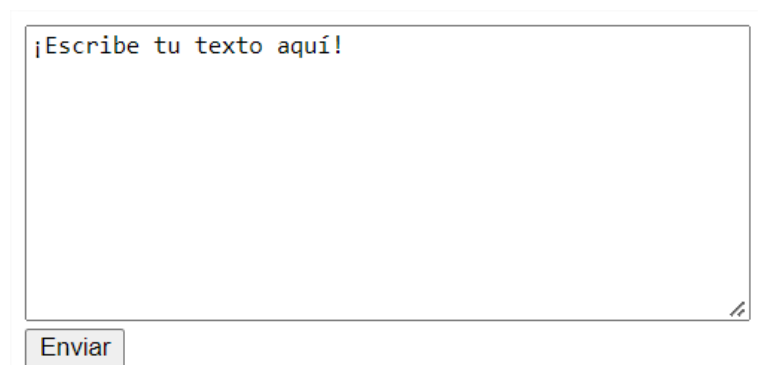
Permite colocar un cuadro de texto de varias líneas para que el usuario pueda introducir un texto largo.

Atributos específicos		
Atributo	Valor	Descripción
rows	Valor numérico entero positivo.	El número de líneas visibles en el control
cols	Valor numérico entero positivo.	La anchura visible del control de texto, en caracteres de anchura media.

La sintaxis es:

```
<form action="https://jkorpele.fi/cgi-bin/echo.cgi" method="post">
  <textarea name="textarea" rows="10" cols="50">¡Escribe tu texto
aquí!</textarea>
  <br>
  <input type="submit" value="Enviar">
</form>
```

Resultado:



2.17 Listas de opciones <select> <option>

Permite mostrar una lista de opciones. Comienza con la etiqueta <select> que es la encargada de dar nombre (name) al control. Dentro cada opción del cuadro se indica con la etiqueta <option> a la que se da valor mediante el atributo **value**. Dentro de la etiqueta <option> se coloca el texto que verá el usuario.

La sintaxis es:

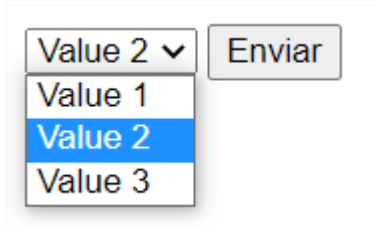
```
<form action="https://jkorpele.fi/cgi-bin/echo.cgi" method="post">
  <label for="eleccion"></label>
  <select id="eleccion" name="eleccion">
```

```

<option value="value1">Value 1</option>
<option value="value2" selected>Value 2</option>
<option value="value3">Value 3</option>
</select>
<input type="submit" value="Enviar">
</form>

```

Resultado:



The screenshot shows a web form with a dropdown menu and a submit button. The dropdown menu is open, showing three options: 'Value 1', 'Value 2' (which is highlighted in blue), and 'Value 3'. The button next to it is labeled 'Enviar'.

Atributos específicos <select>		
Atributo	Valor	Descripción
multiple		Este elemento booleano indica que se pueden seleccionar múltiples opciones de la lista. Si no está especificado, solo se podrá seleccionar una opción cada vez.

Atributos específicos <option>		
Atributo	Valor	Descripción
selected		Si está presente, este atributo booleano indica si esta opción es la inicialmente seleccionada.

2.18 Elementos mostrar/ocultar <details> <summary>

Se trata de secciones ocultas que están dentro de un elemento llamado <details>. Dentro de ese elemento se puede poner una sección de tipo <summary> la cual muestra un texto de resumen de la sección oculta. Al hacer clic en el apartado summary, se muestra todo el contenido de la sección.

La sintaxis es:

```

<details>
  <summary>Contenido oculto</summary>
  <p>Este es el contenido que está oculto.</p>
</details>

```

Resultado:

▼ Contenido oculto

Este es el contenido que está oculto.

Atributos específicos <details>		
Atributo	Valor	Descripción
open		Indica si los detalles serán mostrados al usuario cuando la página se cargue.

2.19 Insertar audio <audio>

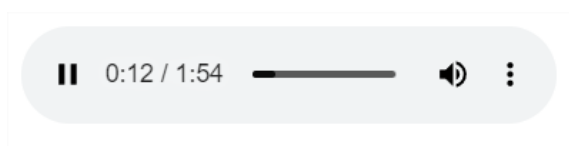
La etiqueta <audio> permite agregar sonidos.

Atributos específicos		
Atributo	Valor	Descripción
src	URL.	La URL del audio que se va a insertar.
controls		Si está presente este atributo, el navegador ofrecerá controles para permitir que el usuario controle la reproducción de audio, incluyendo volumen, búsqueda y pausar/reanudar reproducción. Requerido por algunos navegadores.
autoplay		Si se especifica (incluso aunque el valor sea "false"), el sonido comenzará a reproducirse automáticamente en cuanto sea posible
loop		Inicia la reproducción desde el principio una vez ha finalizado.

La sintaxis es:

```
<audio  
src="https://upload.wikimedia.org/wikipedia/commons/f/f2/Median_test.ogg"  
controls autoplay loop></audio>
```

Resultado:



El audio se reproduce automáticamente cuando esté disponible y se repetirá cuando finalice.

Los formatos de audio más utilizados soportados son:

Formato	Tipo de fuente
.ogg	audio/ogg
.mp3	audio/mpeg
.wav	audio/wav

Podemos establecer audios alternativos con la etiqueta <source>. Si un archivo no puede ser reproducido o no está disponible, pasa al siguiente.

La sintaxis es:

```
<audio controls autoplay loop>
  <source src="https://upload.wikimedia.org/fake_audio.ogg" type="audio/ogg">
  <source src="https://file-
examples.com/storage/fe8c7eef0c6364f6c9504cc/2017/11/file_example_MP3_700KB.m
p3" type="audio/mpeg">
</audio>
```

Se puede apreciar que el atributo src ahora se establece en las etiquetas <source>. Además, se debe establecer el tipo de fuente.

2.20 Insertar vídeo <video>

Esta etiqueta es muy similar a la etiqueta <audio>, pero en este caso, nos permite insertar vídeos.

Atributos específicos		
Atributo	Valor	Descripción
src	URL.	La URL del vídeo que se va a insertar.
controls		Si está presente este atributo, el navegador ofrecerá controles para permitir que el usuario controle la reproducción del vídeo, incluyendo volumen, búsqueda y pausar/reanudar reproducción. Requerido por algunos navegadores.
autoplay		Si se especifica (incluso aunque el valor sea "false"), el vídeo comenzará a reproducirse automáticamente en cuanto sea posible
poster	URL.	Muestra una imagen a modo de presentación.

width		Indica el tamaño de ancho del vídeo.
height		Indica el tamaño de alto del vídeo.
muted		Silencia el vídeo.
loop		Inicia la reproducción desde el principio una vez ha finalizado.

La sintaxis es:

```
<video controls width="450" src="http://clips.vorwaerts-gmbh.de/big_buck_bunny.mp4"></video>
```

Resultado:



Los formatos de vídeos más utilizados soportados son:

Formato	Tipo de fuente
.mp4	video/mp4
.ogg	video/ogg
.webm	video/webm

Podemos establecer vídeos alternativos con la etiqueta <source>. Si un archivo no puede ser reproducido o no está disponible, pasa al siguiente.

La sintaxis es:

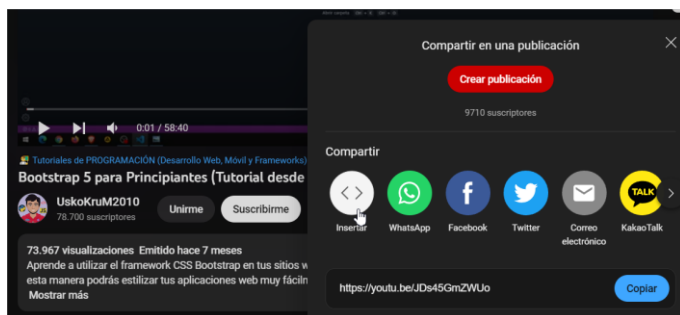
```
<video controls width="450">
  <source src="http://clips.vorwaerts-gmbh.de/fake_video.mp4">
  <source src="http://clips.vorwaerts-gmbh.de/big_buck_bunny.ogv">
</video>
```

Se puede apreciar que el atributo src ahora se establece en las etiquetas <source>. Además, se debe establecer el tipo de fuente.

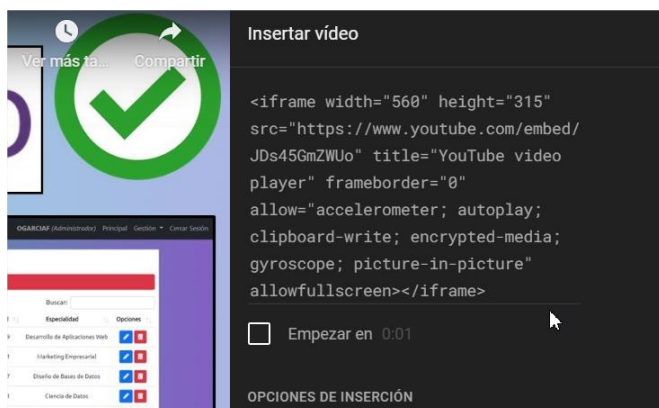
2.20.1 Insertar vídeos de YouTube <iframe>

Podemos insertar en un documento HTML vídeos de YouTube, Vimeo o sitios WEB similares. Para poder realizar esto, aparece una etiqueta denominada <iframe>, que permite insertar el contenido de una web dentro de otra. En este caso, lo usaremos para insertar vídeos.

En cualquier video de YouTube, puedes pulsar la opción de compartir. Podremos ver una opción que indica insertar:



El elegir esta opción de compartir, nos facilitan un código HTML:



Solo debemos copiarlo y pegarlo donde queramos:

```
<iframe width="560" height="315"
src="https://www.youtube.com/embed/JDs45GmZWUo" title="YouTube video player"
frameborder="0" allow="accelerometer; autoplay; clipboard-write; encrypted-
media; gyroscope; picture-in-picture" allowfullscreen></iframe>
```

Resultado:



2.21 Otras etiquetas

HTML5 establece muchas etiquetas que en cualquier momento pueden sernos útiles. Podemos consultarlas en la siguiente URL:

<https://developer.mozilla.org/es/docs/Web/HTML>