

Unidad 2

HTML básico

"El viaje es la recompensa"

Proverbio chino

En este segundo apartado trataremos los elementos más básicos de HTML siguiendo las siguientes normas:

- Nos centraremos en la versión 5, tratando de respetar la norma XHTML 5 más estricta (aunque siempre hablaremos de html por simplificar)
- Veremos en algunos casos algunas de las etiquetas y/o argumentos de uso desaconsejado indicándolo siempre a continuación. Puesto que en nuestro desempeño no sólo haremos páginas web sino que tendremos que modificar las que otros han creado necesitamos saber para que sirven aunque no sea conveniente su uso!
- Trataremos, también, de proporcionar siempre que podamos un contenido semántico a nuestras etiquetas
- Separaremos la parte de diseño (que en un futuro aprenderemos a aplicar mediante CSS) del contenido del documento. No te preocupes si, por el momento, tus páginas resultan demasiado simples.

El Doctype

Todas las páginas HTML deberían de comenzar por un DOCTYPE o Declaración de tipo de documento que es la que asocia dicha página con un DTD (Document Type Definition) o Definición de Tipo de Documento e indica esto al navegador. En realidad, y para no hacerlo más complicado de lo que necesitamos saber, se trata meramente de una línea formal en la que indicamos al navegador la norma que sigue el lenguaje de marcas que debe de interpretar.

En la Wikipedia tenemos diferentes tipos de DOCTYPE:

<https://tinyurl.com/2q28zoxw>

La definición en HTML 5 es esta:

```
<!DOCTYPE html>
```

Mientras que en HTML 4.01 Transicional, la anterior, era esta:

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

Hemos mejorado bastante, verdad? :-)

Importante: El DOCTYPE debe de ocupar siempre la primera línea del documento html sin que exista ni un sólo espacio antes de ella o no tendremos garantías de que todos los navegadores la reconozcan.

Los elementos básicos

Los elementos básicos en una página HTML son html, head y body. La expresión mínima formal de un documento HTML sería este:

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

```

        <meta charset="UTF-8" />
        <title>Lenguajes de Marcas</title>
    </head>
    <body>
        <p>¡Hola mundo!</p>
    </body>
</html>

```

El contenido del elemento html es toda nuestra página web (salvo el DOCTYPE). Lleva un atributo denominado lang que indica el idioma con que está hecha la web.

El contenido del elemento head está formado, a su vez, por una serie de elementos que no se visualizan directamente (salvo el título o title que aparecerá en la barra del navegador).

El elemento meta puede aparecer con diferentes indicaciones dentro de esta sección de cabecera. La más importante es, tal vez, el juego de caracteres con que hemos escrito el documento HTML. Si nos equivocamos puede que algunos caracteres especiales (vocales acentuadas, signos de puntuación, etc.) no aparezcan correctamente. En Linux se suele usar UTF-8 mientras que si usas windows suele ser el ISO-8859-1. De todas formas, lo mejor es consultarlo en el editor de texto que uses

Importante: No confundas juego de caracteres con idioma. ¡Son cosas diferentes!

Por curiosidad: La anterior declaración del meta que define el juego de caracteres también era mas compleja que la actual. Mira:

```

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

```

Por último, el elemento body contendrá toda la parte visible de la web. En este caso se trata de un único párrafo (etiqueta p) con un mensaje típico de bienvenida.

Elementos de bloque y elementos de línea

Las páginas web están hechas siempre de elementos rectangulares que se colocan uno debajo de otro, uno junto a otro o uno encima de otro. No hay nada más. Cualquier cosa que nos parezca diferente se trata de un truco de diseño.

Dentro de esto, tenemos que distinguir entre los elementos de bloque y los de línea. Los de bloque podríamos asemejarlos con los párrafos de un texto: ocupan todo el ancho de la superficie útil del documento (o del área del documento al que han sido restringidos) y pueden tener otros elementos, iguales o diferentes, arriba, abajo o, en algunas ocasiones, en su interior (aunque esto último no siempre es correcto). Un párrafo, por ejemplo, no debería de contener a otro párrafo en su interior. También pueden contener elementos de línea.

Los elementos de línea se sitúan uno debajo de otro, uno al lado de otro (como, por ejemplo, un texto en negrita que puede ir seguido de otro en cursiva) o, incluso, uno dentro de otro (una cita puede tener una palabra en negrita en su interior). Los elementos de línea no pueden tener en su interior un elemento de bloque.

***Nota:** No siempre diremos en este documento cuando un elemento es de bloque y cual es de línea... Pero puedes comprobarlo por ti mismo. Simplemente pon el elemento del que tienes dudas detrás de un elemento con la etiqueta b. Así:*

Etiqueta b<cite>Etiqueta cite</cite>

Si en el navegador se ven uno debajo del otro en lugar de uno al lado de otro es que el elemento que has puesto en segundo lugar es de bloque. Si se vieran uno al lado de

otro es que es se línea. Fácil ¿no?

Otra característica importante de HTML es que por defecto los saltos de línea consecutivos que hagamos manualmente en nuestro documento no se respetan. Por ejemplo, si yo escribiera lo siguiente:

<p>Hola a todos

los compañeros

del curso de HTML</p>

En el navegador se mostraría lo siguiente:

Hola a todos los compañeros del curso de HTML

Cuando aparecen múltiples espacios en blanco sólo se respeta el primero. Por ejemplo:

**<p>Hola a todos los compañeros del curso
de HTML</p>**

Se vería en el navegador así:

Hola a todos los compañeros del curso de HTML

Comentarios

Los comentarios en HTML se escriben entre los símbolos <!-- y -->

<!-- Esto es un comentario y jamás se visualizará

Un comentario puede tener varias líneas -->

***Nota:** Todo, absolutamente todo lo que aprendamos durante este curso a escribir en un documento HTML o CSS llegará al navegador del usuario tal y como las hayamos escritos, sea o no interpretado por este. Los comentarios y algún otro argumento puede que no resulten visibles, pero si el usuario quiere (y sabe hacerlo) siempre podrá verlos, por lo general con la combinación de teclas Ctrl+U o eligiendo la opción adecuada en el menú del navegador. Por tanto, tenemos que tener mucho cuidado. Como norma general lo que no queramos que vea ningún usuario no debería de escribirse.*

Algunas etiquetas comunes

<code><p></p></code>	Define un párrafo
<code>
</code>	Salto de línea
<code></code>	Negritas
<code></code>	Negritas
<code><i></i></code>	Cursivas
<code></code>	Cursivas
<code><sup></sup></code>	Superíndice
<code><sub></sub></code>	Subíndice
<code></code>	Texto tachado
<code><pre></pre></code>	Preformateado
<code><mark></mark></code>	Destaca el texto con un fondo amarillo
<code><h1></h1></code>	Encabezado de nivel 1
<code><h2></h2></code>	Encabezado de nivel 2
...	
<code><h6></h6></code>	Encabezado de nivel 6
<code><cite></cite></code>	Cita corta
<code><q></q></code>	Cita corta
<code><blockquote></blockquote></code>	Cita larga
<code><hr /></code>	Línea horizontal
<code><bdo></bdo></code>	Define la dirección del texto
<code><abbr></abbr></code>	Permite definir abreviaturas (y es la etiqueta aconsejada también para siglas)

Todas las etiquetas llevan asociada una información de estilo por defecto, pero esta puede variar entre cada navegador. Más adelante, cuando veamos CSS, veremos como modificar dicho estilo y tratar de que sea lo más independiente posible del navegador que usemos para verlas.

En las etiquetas en las que podemos elegir, preferiremos siempre las que conllevan información semántica (strong, em, cite) frente a las que sólo nos dan información de estilo (b, i, q).

La etiqueta pre de texto “preformateado” define un área en el que se respeta de forma íntegra los saltos de línea y espacios en blanco que hayamos realizado al escribir el texto. El tipo de letra cambia también a una monoespaciada. Veremos mas adelante las distintas familias tipográficas que podemos usar.

Las etiquetas h1, h2, h3, h4, h5, h6, p, blockquote, y pre son etiquetas que definen elementos de bloque, mientras que strong, b, em, i, sup, sub, del, bdo, cite o q son etiquetas de elementos de línea.

Para indicar si el texto va de izquierda a derecha o de derecha a izquierda se usa la etiqueta bdo con el atributo dir:

```
<p><bdo dir="ltr">dábale arroz a la zorra el  
abad</bdo></p>
```

```
<p><bdo dir="rtl">dábale arroz a la zorra el  
abad</bdo></p>
```

ltr indica que el texto va de izquierda a derecha (left to right) y rtl de derecha a izquierda. El valor por defecto es ltr.

***Nota:** dir es un atributo universal, es decir, que puede usarse con cualquier etiqueta y no solo con bdo*

Usamos la etiqueta abbr para definir una abreviatura o unas siglas. Podemos usarla en conjunto con el atributo title que mostrará una nota explicativa o “tooltip” al pasar el ratón por encima.

```
Mi ordenador tiene 512 <abbr  
title="Megabytes">Mb</abbr> de memoria
```

***Nota:** tooltip es también un atributo universal*

Caractéres especiales

Los caracteres especiales que se usan para las etiquetas no pueden escribirse tal cual en un documento HTML, así que hay que usar una codificación especial.

< <

> >

" “

& &

Existen también un caracter especial que pueden sernos úti:

** ** **espacio irrompible y no eliminable**

Si necesitamos algún otro caracter que no aparece en nuestro teclado (símbolos tipográficos, etc.) podemos recurrir a la siguiente tabla:

<http://www.ascii.cl/es/codigos-html.htm>

¿Qué tendrías que hacer para mostrar en el navegador uno de estos códigos especiales?

Listas

Las listas se definen mediante las etiquetas ul, li y ol.

**** Define una lista no ordenada (con “boliches”)

**** Define una lista ordenada

**** Define cada item de la lista en cualquiera de los anteriores

Por ejemplo:

```
<ol>  
    <li>Lunes</li>  
    <li>Martes</li>  
    <li>Miércoles</li>  
    <li>Jueves</li>  
</ol>
```

Las listas se pueden anidar, pero teniendo en cuenta que la nueva lista tiene que abrirse antes de cerrar el item anterior. Por ejemplo:

```
<ul>  
    <li>Punto 1</li>  
    <li>Punto 2  
        <ul>  
            <li>Apartado A del punto 2</li>  
            <li>Apartado B del punto 2</li>  
        </ul>  
    </li>  
    <li>Punto 3</li>  
</ul>
```

Los únicos atributos que aún pueden usarse en lo son reversed, start y type.

start sirve para variar el número por el que se inicia la lista ordenada (que por defecto es el 1) y type para cambiar el formato de enumeración que usamos.

Ejemplo:

```
<ol start="50" type="A">
```

Otros tipos válidos para el argumento type son "1", "I", "i" o "a".

Reversed ordenaría la lista al revés. Es un atributo que no necesita valor, funciona simplemente con poner su nombre. Pero como nosotros vamos a trabajar con xhtml nos inventamos uno. Por ejemplo así:

```
<ol start="50" type="A" reversed="yes">
```

ul no tiene permitido el uso de atributos. Pero el atributo type (¡¡¡de uso desaconsejado, recuerda !!!) nos permitiría cambiar el tipo de símbolo en las listas desordenadas. Por ejemplo <ul type="square">. Otros valores válidos para este atributo son "disc" o "circle".

La etiqueta li puede llevar el atributo value (solo cuando se usa junto con ol) para alterar el orden de la numeración en la lista. Ejemplo:

```
<li value="333">
```

Tanto value como start sólo pueden llevar valores numéricos en formato decimal aunque el tipo de lista sea de otro tipo (con letras, números romanos, etc.). El navegador reemplazará al valor correcto. Por ejemplo, si usamos type="a" y start="5" nuestra lista comenzará por la letra e

La etiqueta li se puede usar también fuera de un bloque ol o ul

Paciencia: Ya aprenderemos a hacer todas estas cosas que se desaconsejan desde HTML con CSS en el siguiente capítulo

Imágenes

Para insertar imágenes se utiliza la etiqueta img. Es una etiqueta única (sin contenido) por lo que tenemos que usar el método de auto-cerrado para cumplir las normas de XHTML. La sintaxis básica es esta:

```

```

El atributo src es obligatorio e indica el lugar donde está la imagen. Se puede usar una ruta relativa, absoluta o indicar una URL web.:

```

```

```

```

Importante: A quién enlaza imágenes de otras webs sin autorización se le conoce como leecher (de leech, sanguijuela). Un buen administrador es capaz de bloquear estas prácticas o incluso hacer que en lugar de las imágenes en cuestión aparezcan escenas de abuso, desagradables u ofensivas. Piensa, además, que desde el punto de vista legal necesitas tener los derechos de autor de las imágenes que publicas en tu web o tener la certeza de que estás se publican con una licencia libre

El atributo alt también es obligatorio en XHTML. Lo que pongamos como valor del atributo alt será lo que se verá en la web si por algún problema la imagen desaparece o no está disponible y, mucho más

importante, la descripción que oirán los navegantes invidentes cuando visiten nuestra web.

```

```

El atributo title muestra un “tooltip” al pasar el ratón sobre la imagen. width y height nos permiten modificar el ancho y/o el alto de la imagen expresando el tamaño deseado en píxeles. Si no los usamos la imagen se mostrará con su tamaño original. Si modificamos sólo uno de ellos el otro se ajustará de forma proporcional.

La etiqueta img es una etiqueta de línea.

Los hiperenlaces

Los enlaces a otras páginas se realizan mediante la etiqueta a

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

El atributo href es obligatorio e indica la dirección destino del enlace. Puede ser una página web en el mismo disco, en el mismo servidor, en otro servidor, a un punto de la misma página, a una dirección de correo, etc.

```
<a href="album.html">Álbum de fotos</a>
```

```
<a href="mailto:josemaria@uponaday.net">Enviar un
correo</a>
```

Un ejemplo más completo para un enlace a email en el que rellenamos no solo la dirección de destino, sino también el asunto y el campo cc:

```
<a href="mailto:josemaria@uponaday.net?subject=De un
alumno de DAW&cc=departamento@reyfernandovi.com">Enviar
correo</a>
```

Importante: Para que valide correctamente como XHTML estricto debemos de sustituir el signo & por el código & visto antes. No se ha hecho en el ejemplo anterior por claridad.

Otros atributos válidos son title y target:

```
<a href="album.html" title="Texto resaltado al pasar el  
ratón" target="_blank">Álbum de fotos</a>
```

El atributo title, como ya hemos visto en otros elementos, nos muestra un “tooltip” al pasar el ratón sobre el enlace, mientras que target nos permite indicar donde queremos que se abra la nueva página a la que hace referencia el enlace. El valor por defecto de target es “_self” e indica que se abrirá en la misma página donde lo pulsamos. Si usamos el valor “_blank” la página se abrirá en una solapa o ventana nueva. Otro parámetro válido es _parent que, como su nombre indica, abre la página en la solapa padre de la referencia.

Otra utilidad de esta etiqueta es hacer un enlace a un punto dentro de la misma página. Para ello usamos la siguiente sintaxis:

```
<a href="#C4">Ver el capítulo 4</a>
```

...

```
<h2 id="C4">CAPÍTULO 4</a>
```

Como vemos, usamos un elemento a para crear el hiperenlace usando un término a nuestra elección precedido del carácter especial #. En el punto de destino del hiperenlace debemos de usar el atributo id y como contenido del mismo pondremos el mismo término que pusimos como valor del atributo href pero en este caso prescindiendo del carácter #. Este atributo id que nos sirve como referencia para el destino del

hipervínculo puede ir en cualquier etiqueta. El valor del atributo id debe de ser único y no debe repetirse nunca en la misma página web.

Importante: *Hablaremos mucho en el futuro del parámetro id porque es particularmente importante para algunos selectores de CSS*

Para ir a un punto dentro de otra página usamos un formato similar:

```
<a href="otrapagina.html#C4">Ver el capítulo 4 de  
otrapagina.html</a>
```

En esa otra página deberíamos de tener un elemento a con un atributo id de forma similar al caso que hemos visto antes:

```
<h3 id="C4">CAPÍTULO 4</a>
```

La etiqueta a es una etiqueta de línea.

Tablas

Las tablas se han usado durante mucho tiempo en HTML como el sustituto de otras alternativas de diseño. Nosotros no deberíamos de hacer esto jamás. Una tabla debe de usarse para lo que es: para presentar datos en un formato tabulado y no, por ejemplo, para presentar información en cuatro columnas separadas. Para esto ya veremos la forma correcta de hacerlo.

La sintaxis que muestra una tabla sencilla quedaría con dos filas y tres columnas sería así:

```
<table>  
  <tr>  
    <td>1</td><td>2</td><td>3</td>
```

`</tr>`

`<tr>`

`<td>4</td><td>5</td><td>6</td>`

`</tr>`

`</table>`

La etiqueta `<table></table>` es la que define y contiene al elemento tabla. Cada fila va encerrada como contenido de la etiqueta `<tr></tr>` y dentro de estas cada celda va delimitada por la etiqueta `<td></td>`. El número de celdas que hay en una fila determina el número de columnas de la tabla.

La mayoría de los atributos que podían usarse con la etiqueta `table` (`width`, `cellpadding`, `cellspacing`, `frame`, `rules` y `summary`, `border`...) han sido desaconsejados y/o han desaparecido en HTML5.

Importante: En clase y en los ejemplos siguientes seguiremos usando el atributo `border` para poder ver lo que hacemos claramente, pero en cuanto que aprendamos a hacer lo mismo con CSS nos olvidaremos de él!

Las etiquetas `tr` también han visto desaparecer como desaconsejados sus atributos `align` y `valign`. La etiqueta `td`, por último, tampoco dispone de bendiciones para usar sus atributos `width`, `height`, `align`, `valign`, `abbr`, `axis` y `scope`.

La etiqueta `td` continúa disponiendo de los atributos `colspan` y `rowspan` que sirven para fusionar celdas horizontal o verticalmente.

`<table border="1">`

`<tr>`

`<td colspan="3">1, 2 y 3</td>`

```

</tr>

<tr>

    <td>4</td><td>5</td><td>6</td>

</tr>
</table>
<table border="1">

    <tr>

        <td rowspan="2">1 y 4</td><td>2</td><td>3</td>

    </tr>

    <tr>

        <td>5</td><td>6</td>

    </tr>
</table>

```

La etiqueta <th> se usa para definir una celda como de encabezado y puede usarse tanto en filas como en columnas:

```

<table border="1">

    <tr>

        <th></th><th>A</th><th>B</th><th>C</th>

    </tr>

    <tr>

        <th>D</th><td>1</td><td>2</td><td>3</td>

    </tr>

    <tr>

```



```

        <th>E</th><td>4</td><td>5</td><td>6</td>
    </tr>
</table>

```

La etiqueta `caption` se usa para dar título a la tabla. Debe de aparecer una sólo vez e inmediatamente después de la apertura de la etiqueta `table`:

```

<table border="1">
    <caption>Título de la tabla</caption>
    <tr>
        ...
    </table>

```

Las etiquetas `thead`, `tbody` y `tfoot` nos permiten estructurar de forma lógica la información de la tabla, pudiendo incluso variar el orden de los mismos (definiendo el cuerpo por delante de la cabecera, por ejemplo) si así nos conviniera:

```

<table border="1">
    <thead>
        <tr>
            <th>Mes</th>
            <th>Ingresos</th>
        </tr>
    </thead>
    <tfoot>
        <tr>

```

```

        <td>Total</td>

        <td>2800,00 €</td>

    </tr>

</tfoot>

<tbody>

    <tr>

        <td>Enero</td>

        <td>1200,00 €</td>

    </tr>

    <tr>

        <td>Febrero</td>

        <td>1600,00 €</td>

    </tr>

</tbody>

</table>

```

Por último, disponemos también de la etiqueta `colgroup` que nos permitirá agrupar de forma lógica algunas columnas de la tabla para aplicarles estilos de forma conjunta, pero no tiene sentido ver como funciona hasta que no trabajemos con CSS.

Nuevas etiquetas para la organización

Dentro de las nuevas etiquetas, el primer grupo que nos interesa ver es el de aquellas que introducen una mayor información semántica a la estructura del documento. Se trata de `header`, `nav`, `footer`, `aside`, `section` y `article`. Se trata de etiquetas que identifican las secciones más habituales

que existen en la gran mayoría de las páginas web y que hasta ahora identificábamos usando la etiqueta div en función con algún id o class, por ejemplo `<div id="footer">` o `<div class="article">`.

Estas etiquetas no hacen nada especial y no introducen ninguna información de estilo. Su único propósito es proporcionar información semántica a nuestras páginas de cara, sobre todo, a facilitar su lectura y la valoración del contenido por parte de los buscadores. Pensemos, por ejemplo, en una página web que se titule “Carnicerías de calidad en Sevilla” y otra en cuyo pie de página aparezca el nombre de su autor: “Javier Carnicería Sevilla”. Google no debería de tratar a ambas al mismo nivel cuando alguien realice una búsqueda de carnicerías en Sevilla. Estas etiquetas le proporcionan, a él y al resto de buscadores, una forma clara de valorar el contenido de una web según donde se presente cada texto.

Etiquetas y atributos desaconsejados

Existen otras etiquetas y atributos que se usaban habitualmente en HTML4 y aunque ya desaconsejadas siguen siendo interpretadas correctamente por la mayoría de los navegadores. Para no crear malos hábitos de trabajo hemos preferido mencionarlas lo menos posible (o no hacerlo en absoluto) en este documento. Si te las encuentras por ahí y necesitas ayuda para entender lo que hacen ya sabes: w3schools es tu amigo ;-)

Etiquetas multimedia

HTML5 incluye dos etiquetas, audio y video para facilitar la interpretación directa por parte del navegador de contenido multimedia y embed para incrustar archivos en formato flash. Veremos estas etiquetas mas adelante.