

Introducción a HTML

HTML (Hyper Text Markup Language) es un lenguaje de marcado de texto, lo cual nos indica que no es un lenguaje de programación como Java, PHP, C o VisualBasic. HTML es el lenguaje para escribir páginas web. Las marcas, conocidas como etiquetas, describen la forma en la que se estructura el contenido de una página web.

Las etiquetas describen diferentes elementos de la página, por ejemplo, la etiqueta <p> se utiliza para delimitar párrafos de texto y la etiqueta para definir imágenes. El encargado de interpretar estas etiquetas es un programa llamado navegador (browser), por ejemplo, el navegador Chrome de Google o el Firefox de Mozilla. Los archivos que contienen el código HTML tienen la extensión .html, y son archivos de texto plano (sin formato). Más adelante veremos qué editores se pueden utilizar para crearlos, en este curso vamos a utilizar el editor visual Studio Code.

ETIQUETAS

Las etiquetas (marcas) delimitan cada uno de los elementos que componen un documento HTML.

El lenguaje HTML se basa en etiquetas, por eso es importante entender bien la sintaxis de las etiquetas. El siguiente esquema muestra las partes de una etiqueta HTML genérica.

La etiqueta de apertura está delimitada por el signo < "menor que", a continuación, el propio nombre de la etiqueta (en este caso p) y opcionalmente, un atributo (en este caso class="fondo"), finalmente el signo > "mayor que" para cerrar.

A continuación, está el texto del contenido de la etiqueta, lo más frecuente es un texto, pero podría ser otra etiqueta, una imagen, ...

<p>se coloca el texto del párrafo </p>

Algunas etiquetas no tienen contenido, y se cierran sobre sí mismas, no tienen etiqueta de cierre, se llaman etiquetas vacías, como la etiqueta
. La etiqueta
 produce un salto de línea. Las etiquetas vacías no tienen contenido, pero si pueden tener atributos. Algunas otras etiquetas vacías son: <hr>, , <input>, <link>, <meta>, <param> y <source>.

En una etiqueta, la primera palabra que aparece es el nombre de la etiqueta. También podemos encontrar atributos con sus valores, siguiendo el formato <etiqueta atributo1="valor" atributo2="valor">. La forma más correcta es escribir el nombre de la etiqueta y los atributos en minúscula y los valores entre comillas dobles.

Por ejemplo, Visita Colsubsidio

En general las etiquetas se pueden anidar pero no se pueden 'cruzar'. Estas dos etiquetas <p> y <div>, están anidadas correctamente:

```
<div>
<p>
Texto del párrafo
</p>
</div>
```

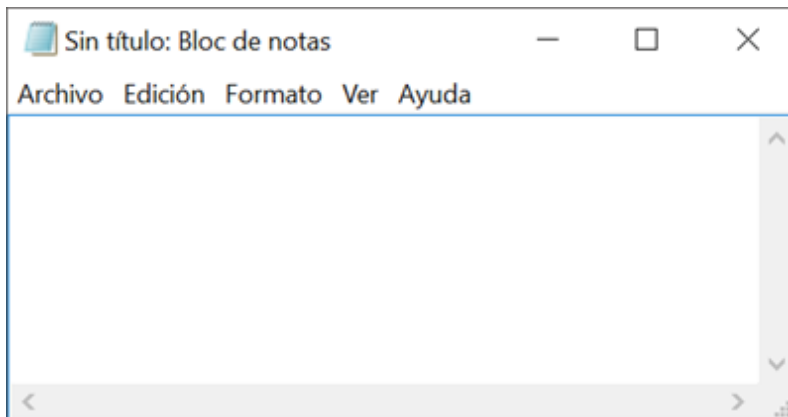
Es incorrecto cruzarlas así:

```
<div>
<p>
Texto del párrafo
</div>
</p>
```

Mi primera página web

Vamos a comenzar por escribir una página web muy sencilla. Para ello podemos utilizar cualquier editor de texto sin formato, por ejemplo, el Bloc de Notas (Notepad) incluido en Windows. En ordenadores Apple puedes usar el TextEdit. Para abrir el Bloc de Notas en Windows 8 o posterior, teclea Bloc de Notas en la caja de búsqueda de la barra de tareas y ábrelo. En Windows 7 desde el Menú Inicio → Programas → Accesorios → Bloc de notas

El aspecto del Bloc de Notas es muy simple, una hoja en blanco con una barra de menús.



Escribiremos el siguiente código. Más adelante ya veremos qué es cada elemento:

```
<html>
<head>
  <title>Mi primera página</title>
</head>
<body>
  <p>Esta es una frase de ejemplo</p>
</body>
</html>
```

Una vez escrito, vamos a guardarlo. Pulsamos en el menú Archivo y elegimos Guardar. Introducimos el nombre, por ejemplo, primera. Si ahora pulsásemos Guardar, se guardaría con la extensión txt, que indica que es un documento de texto sin formato. Para guardarlo como una página web, debemos de emplear la extensión **.html** Por lo que completamos el nombre para que quede **primera.html** y pulsamos Guardar.

Ahora, utilizando el Explorador de Windows, abrimos la carpeta en la que hemos guardado el archivo y hacemos doble clic sobre él. Como el archivo tiene extensión .html, se abrirá la página con el navegador que tengamos como predeterminado.

Editores de HTML

Como decíamos, los archivos que contienen el código HTML tienen la extensión .html, y son archivos de texto plano, es decir, lo que se ve es lo que hay, no hay código oculto como ocurre con los archivos de los procesadores de texto (como Microsoft Word y otros) esto quiere decir que un archivo HTML puede ser escrito con cualquier editor que sea capaz de escribir archivos de texto plano, como el Bloc de notas de Windows, aunque lo recomendable es utilizar editores específicos para páginas web como visual Studio Code. , Atom, NotePad++, Sublime, Dreamweaver, etc.

Estos editores específicos nos ayudan a escribir de forma más rápida el código HTML. Podemos dividirlos en dos grandes grupos: 1. Los editores de código. 2. Los editores en modo diseño.

1. Los editores de código. Por ejemplo, Brackets, Atom, Sublime, NotePad++, son editores en los que nosotros debemos escribir el código HTML, es decir, las etiquetas. Por lo tanto, debemos saber qué etiqueta debemos escribir en cada momento según lo que queramos hacer. Por ejemplo, para poner una palabra en negrita hay que utilizar la etiqueta , o para insertar una imagen la etiqueta con sus correspondientes atributos
2. Los editores con modo diseño. Por ejemplo, Dreamweaver y Kompozer, son editores en los que no hace falta que nosotros escribamos el código HTML ya que el propio editor se encarga de hacerlo según nuestras indicaciones. Por ejemplo, para poner una palabra en negrita basta seleccionarla y pulsar el "botón negrita". Para insertar una imagen bastará pulsar en el comando "Insertar imagen", y el editor nos abrirá una ventana para elegir la imagen. Con estos editores se suele trabajar en la vista Diseño que nos muestra directamente el resultado de la página, es decir, al aplicar negrita veremos directamente la

palabra en negrita, y al insertar una imagen, esta aparecerá en la pantalla. Estos editores también permiten escribir directamente las etiquetas en la vista código. Este tipo de editores también son conocidos como WYSIWYG, (What You See Is What You Get, lo que ves es lo que obtienes).

En el primer tipo de editores para ver el resultado de lo que estamos escribiendo deberemos abrir el archivo con un navegador.

Cada uno de estos tipos tiene sus ventajas e inconvenientes, pero para el propósito de este curso es mucho mejor utilizar un editor de código ya que de lo que se trata es de aprender HTML y por lo tanto deberemos aprender a utilizar las etiquetas de HTML. Más adelante, cuando dominemos el HTML podremos utilizar un editor en modo diseño, pero siendo capaces de entender lo que el editor hace por nosotros.

En este curso vamos a trabajar con el editor Visual Studio Code.

Etiquetas de texto

En esta unidad vamos a empezar a ver las etiquetas que sirven para colocar el texto en una página web. Antes, de forma sencilla, veremos la estructura básica de una página web, también veremos una primera introducción a las hojas de estilo CSS.

Estructura básica de la página

La primera línea `<!DOCTYPE ... >` no es propiamente una etiqueta de HTML, sino una declaración o instrucción al navegador sobre la versión de HTML que utiliza la página. Para la versión HTML 5, el tipo de documento es html: `<!DOCTYPE html>`

El resto del documento HTML está contenido dentro de la etiqueta `<html></html>`. Para indicar el idioma es conveniente añadir el atributo `lang="es"`: `<html lang="es"></html>`

Dentro de la etiqueta `<html>`, encontramos dos subdivisiones, la cabecera, delimitada por las etiquetas `<head></head>` y el cuerpo, delimitado por las etiquetas `<body></body>`. Por lo tanto, el aspecto básico de cualquier página web, es el siguiente:

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    </head>
```

```
  <body>
```

```
    </body>
```

```
</html>
```

La etiqueta `<head>` contiene información sobre la página. Por ejemplo, contiene etiquetas que pueden decir de qué trata la página, el título que debe de mostrar en la barra del navegador, o código javascript y estilos, que pueden estar en el propio encabezado o como enlaces a otros archivos. Normalmente, lo que contiene esta etiqueta no se muestra en el navegador, es una etiqueta que contiene declaraciones sobre la página, pero no contenido propiamente dicho como texto o imágenes.

Las etiquetas que pueden ir dentro de `<head>` son `<title>` `<style>` `<base>` `<link>` `<meta>` `<script>` `<noscript>`. Iremos viendo estas etiquetas a medida que nos hagan falta ya que ahora no seríamos capaces de entender bien su significado en todos los casos. De momento sólo comentaremos que obligatoriamente debe de contener la etiqueta `<title></title>`, que contiene el título de la página, que es lo que se ve en la barra de título del navegador.

Dentro de `<head>`, para optimizar la página para diferentes anchos de pantalla, también es conveniente definir el viewport, por ejemplo de esta forma: `<meta name="viewport" content="width=device.width, initial-scale=1.0">`.

En el `<body>` encontramos el contenido de la página, lo que se ve a través del navegador: texto, imágenes, enlaces, tablas, etc... Por lo tanto, dentro del `<body>` pueden ir la gran mayoría de las etiquetas html.

Escribiendo texto en el cuerpo de la página

Como estamos diciendo todo el texto de la página estará dentro del `<body>`, el texto dentro del `<body>` debe estar, principalmente, dentro de párrafos. En HTML, los párrafos se identifican con la etiqueta `<p></p>`. Dentro de los párrafos colocaremos texto e imágenes y algunas etiquetas que nos permitan dar formato al texto. Dentro de un párrafo no debe haber otros párrafos anidados.

A su vez, dentro del `<body>`, disponemos de encabezados que nos permiten organizar los títulos o rótulos del texto. Los encabezados van desde el `<h1></h1>` de mayor tamaño, al `<h6></h6>`, el más pequeño. Por ejemplo, utilizaríamos un `<h1>` para poner el rótulo principal de la página, `<h2>` para los títulos de los apartados, `<h3>` para los apartados de segundo nivel, etc. El propio texto de cada apartado iría fuera del encabezado contenido en párrafos `<p></p>` u otras etiquetas.

Una de las peculiaridades del texto, es que los espacios en blanco consecutivos y saltos de línea no se muestran como en el código fuente. Si hay varios espacios seguidos, sólo se mostrará uno. Por ejemplo, si en el código fuente escribimos Bienvenido a este curso se verá Bienvenido a este curso. Si queremos poner varios espacios seguidos, utilizaremos el código html para el espacio, ` `;

El navegador también ignora los saltos de línea. Así si dentro de un párrafo colocamos varios saltos de línea pulsando Intro, estos no se verán. Para crear un salto de línea dentro de un párrafo, utilizamos la etiqueta `
`.

Cada etiqueta `<p></p>` lleva implícito un salto de línea.

Etiqueta `<p>`

La etiqueta <p> define un párrafo.

Ejemplo: <p>Un párrafo de texto</p>

Produce este resultado:

Un párrafo de texto

La etiqueta <p> es de tipo bloque, es decir, por defecto ocupa una línea completa en la página web. Además, también incluye un margen superior e inferior. Por ejemplo, si escribimos dos párrafos en la misma línea: <p>Un párrafo de texto.</p><p>Otro párrafo.

Produce este resultado:

Un párrafo de texto.

Otro párrafo.

Si el contenido del párrafo no cabe en una línea será el navegador web quien lo divida en varias líneas según el ancho disponible. Por lo tanto, el número de líneas que ocupe un párrafo dependerá del ancho disponible. Si queremos incluir un salto de línea explícito dentro de un párrafo podemos utilizar la etiqueta
. Como ya hemos dicho, los saltos de línea que escribamos en el código html no tienen efecto al mostrar la página.

Texto preformateado <pre>

La etiqueta <pre> define texto preformateado, es decir, que conserva los espacios y saltos de línea presentes en el código html.

Ejemplo: <pre>Aquí hay más de un espacio

y un salto de línea</pre>

Produce este resultado:

Aquí hay más de un espacio

y un salto de línea

Por defecto, el navegador presenta el texto de esta etiqueta en una fuente de la familia monospace (Courier).

Esta etiqueta se utiliza cuando queremos que el navegador muestre los espacios y saltos de línea, como en un poema, datos en forma de tabla, etc. Por ejemplo:

Semifinal A Semifinal B

\ /

\ /

\ /

Final 15 Julio

Encabezados <h1> <h2> ...

Las etiquetas <h1> <h2> ... <h6> crean texto para encabezados. Se suelen utilizar para títulos, subtítulos, rótulos, etc.

Ejemplo: <h1>Encabezado de primer nivel h1 </h1>

En la siguiente imagen puedes ver el tamaño de cada tipo de cabecera en relación al texto de párrafo:

Cabecera h1

Cabecera h2

Cabecera h3

Cabecera h4

Cabecera h5

Cabecera h6

Texto normal de párrafo

Para los títulos y subtítulos es preferible utilizar etiquetas de encabezado que cambiar el tamaño de una etiqueta de párrafo ya que las etiquetas de encabezado tienen significado semántico y son utilizadas por los buscadores para dar significado al texto en función de su importancia.

Separadores <hr>

La etiqueta <hr> crea un separador horizontal.

Ejemplo: <hr>

Produce este resultado:



Etiqueta <q>

La etiqueta <q> se utiliza para poner entre comillas un texto corto.

Ejemplo:

<p> Al despedirse dijo: <q>Hasta nunca</q> y cerró la puerta.</p>

Al despedirse dijo: “Hasta nunca” y cerró la puerta.

Etiqueta

La etiqueta se utiliza para dar importancia a un texto corto. Los navegadores ponen el texto en negrita.

Ejemplo:

<p> Un coche histórico debe tener, al menos, 25 años desde la primera matricula</p>

Un coche histórico debe tener, al menos, **25 años** desde la primera matricula.

Etiqueta

La etiqueta se utiliza para dar énfasis a un texto corto. Los navegadores ponen el texto en cursiva.

Ejemplo:

<p> Una página web puede ser consultada desde cualquier parte del mundo.</p>

Una página web puede ser consultada desde *cualquier* parte del mundo.

Etiqueta

La etiqueta se utiliza para destacar un texto corto, sin que ello signifique que es importante. Los navegadores ponen el texto en negrita.

Ejemplo:

<p> El futbolista es el deportista mejor pagado.</p>

El futbolista es el deportista mejor **pagado**.

En muchas ocasiones es difícil distinguir si un texto corto es importante, queremos darle énfasis o merece ser destacado, por lo que utilizar las etiquetas o puede ser un poco ambiguo. Podríamos simplificar diciendo que si queremos ponerlo en cursiva utilizaremos , y si queremos ponerlo en negrita y que sea tenido en cuenta por los buscadores como un texto importante utilizaremos , y si sólo queremos ponerlo en negrita utilizaremos .

Etiqueta <sub>

La etiqueta <sub> se utiliza para ver texto como subíndice, es decir, colocado en la mitad inferior de la línea.

Ejemplo:

<p> H₂O. Texto _{subíndice}</p>

H₂O. Texto subíndice.

Etiqueta <sup>

La etiqueta <sup> se utiliza para ver texto como superíndice, es decir, colocado en la mitad superior de la línea.

Ejemplo:

<p> x². Texto ^{superíndice}</p>

x². Texto ^{superíndice}

Imágenes

Las imágenes son un elemento esencial en las páginas web. Hasta la versión 5 de HTML se utilizaba la etiqueta para incluirlas en una página web. Pero debido a la necesidad de crear páginas web que se adapten a dispositivos de diferente tamaño se ha incorporado la etiqueta <picture>.

Etiqueta

Con la etiqueta representamos imágenes en una página web. Esta etiqueta no tiene etiqueta de cierre, se cierra con el carácter >. Para indicar el origen de la imagen se utiliza el atributo src="ruta_imagen". Por ejemplo:

```

```

Que produce el siguiente resultado:



Atributo src de la etiqueta

Como acabamos de ver el atributo src sirve para indicar la ruta donde se encuentra la imagen que se va a mostrar. Por tanto, la imagen estará en un archivo distinto al archivo que contiene la página web, no puede estar incrustada en el mismo archivo (salvo que se genere dinámicamente).

Este archivo de imagen puede encontrarse en cualquier lugar accesible desde la página web, en el mismo directorio, en otro directorio, en otro servidor de internet, etc. En cada caso la ruta para acceder al archivo será distinta, y puede escribirse de forma absoluta o de forma relativa. Una ruta es absoluta cuando incluye todos los elementos de la ruta, y una ruta es relativa cuando parte del directorio en el que se encuentra la página web.

Atributo alt de la etiqueta

El atributo alt contiene texto alternativo, que se mostrará en lugar de la imagen cuando no sea posible verla.

Por ejemplo, si escribimos el siguiente código con el nombre del directorio erróneo (grafico en lugar de graficos) se mostrará el texto de atributo alt:

```

```

Si no existe la imagen ni tampoco el atributo alt algunos navegadores (Firefox) mostrarán un



pequeño gráfico con un icono similar a este: , y otros navegadores (Chrome) no mostrarán nada:

Aunque el atributo Alt es obligatorio, si se omite no se producirá ningún error apreciable por el usuario. Si se omite este atributo, al validar la página con la [herramienta W3C](#) no dará una validación correcta. Aparte de la función principal de mostrar el texto cuando no existe la imagen, es importante utilizar Alt por dos motivos.

El primero es porque lo usan los buscadores para indexar las imágenes en sus bases de datos de forma que podamos encontrar la imagen desde un buscador, el segundo motivo es porque el atributo alt es utilizado por los programas que ayudan a los ciegos y personas con problemas visuales a leer las páginas web. En cualquier caso, es importante dar un texto descriptivo y corto a

este atributo.

**Atributo title en la etiqueta **

El atributo title es un atributo global, es decir que puede usarse en la mayor parte de las etiquetas de HTML. El atributo title proporciona información descriptiva del elemento en el que está definido. El atributo title se muestra como un tooltip al colocar el cursor encima del elemento y es bastante utilizado en las etiquetas img ya que puede ser interesante proporcionar al usuario información sobre la imagen en forma de tooltip.

**Atributos width y height de la etiqueta **

El atributo width establece la anchura de visualización de la imagen, y el atributo height establece la altura de visualización de la imagen. No son obligatorios, si no aparecen la imagen tendrá su altura y anchura original. Si aparecen, modificarán el aspecto de la imagen, pero no afectarán al archivo de imagen, es decir, sólo afectarán a la visualización que realiza el navegador de la imagen. Por defecto, las unidades son píxeles. Por ejemplo, si queremos que la imagen anterior se vea a la mitad de su tamaño original (200 x 79) escribiremos este código.

```

```