

Introducción a HTML

Conceptos a aprender

- Introducción a HTML
- Etiquetas
- Atributos
- Imágenes
- Hipervínculos
- Listas

Introducción

HTML es una sigla en inglés que representa “HyperText Markup Language”, que significa “Lenguaje de marcado de hipertexto”. HiperTexto se refiere a que el formato busca pasar más allá de los límites impuestos por el texto mismo, hacerlo interactivo. Un lenguaje de marcado no tiene nada que ver con uno de programación, incorpora etiquetas para ‘marcar’ el contenido y que el browser (navegador de internet) lo entienda y pueda procesarlo.

Etiquetas HTML

La gran mayoría de los elementos funcionan rodeando el contenido que representan usando etiquetas de apertura y de cierre. De la siguiente forma:

```
<h1>El primer título</h1>
```

Podemos ver que este contenido “El primer título” está entre dos etiquetas h1, una de apertura, <h1>, y una de cierre </h1>. La etiqueta de cierre siempre tiene la barra delante del nombre del elemento para denotar que ahí termina el mismo.

Existen otras etiquetas que son ‘autocerrables’, es decir, solo tienen una parte. Por ejemplo:

```
<br />
```

La etiqueta `
` sirve para hacer saltos de línea. Como no requiere ningún contenido, no hace falta tener una etiqueta de apertura y una de cierre.

Etiquetas más utilizadas

Si bien existen más de 50 etiquetas para utilizar en HTML, en la siguiente tabla van a encontrar algunas de las más populares/utilizadas en el día a día.

<code><html>...</html></code>	Determina el comienzo y fin de la página que estamos escribiendo
<code><h1>...</h1></code>	La etiqueta h1 se utiliza para mostrar Títulos de gran tamaño
<code><h2>...</h2></code> , <code><h3>...</h3></code> , <code><h4>...</h4></code> , <code><h5>...</h5></code> y <code><h6>...</h6></code>	Todos los se utilizan para mostrar títulos, pero cuanto más grande el número, más chico va a ser el tamaño de letra.
<code><p>...</p></code>	P = "Paragraph" (párrafo). Se utiliza para contener todo un texto en un párrafo.
<code>
</code> o <code>
</br></code>	BR = Line Break (salto de línea). Agrega una línea en blanco.
<code><a>...</code>	A = "Anchor" (Ancla). Crea un enlace a otras páginas de internet, archivos o ubicaciones dentro de la misma página, direcciones de correo, o cualquier otra URL.
<code>...</code>	B = "Bold" (negrita). Convierte el texto en negrita
<code></code> <code>..</code> <code></code>	UL y LI = Listas. Se utiliza para crear una lista de elementos.
<code><table>...</table></code>	Table = Tabla. En conjunto con otros elementos como TR, TD, etc, se utiliza para crear estructuras de tablas.
<code><div>...</div></code>	Div = Division. Sirve para crear secciones o agrupar contenidos en un bloque.
<code>...</code>	Span = Abarcar. Es un contenedor en línea. Sirve para aplicar estilo al texto o agrupar elementos en línea.
<code><input></code>	Input = Entrada. Se utiliza para solicitar información a los usuarios de la página.

Lista de todas las etiquetas: <https://www.w3schools.com/TAGS/default.asp>

Atributos de las etiquetas

Las etiquetas además pueden tener atributos que las modifican, por ejemplo:

```
<p class="foo" id="bar">Párrafo con atributos</p>
```

Este párrafo tiene una clase "foo" y un id "bar".

En otro ejemplo, la etiqueta `<input>` utiliza el atributo `type` para indicar que tipo de ingreso de datos queremos solicitar:

<code><input type="text"></code>	→	<input type="text" value="abc"/>
<code><input type="password"></code>	→	<input type="password" value="*****"/>
<code><input type="email"></code>	→	<input type="email" value="abc@gmail.com"/>
<code><input type="number"></code>	→	<input type="number" value="34567"/>
<code><input type="checkbox"></code>	→	<input checked="" type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>
<code><input type="radio"></code>	→	NO <input checked="" type="radio"/> YES

Anidar elementos HTML

Los elementos HTML no siempre tienen que contener texto, también pueden contener otros elementos HTML.

```
<div>
  <div>
    <div>
      <h1>Título 1</h1>
      <p>Parrafo 1</p>
      <p>Parrafo 2</p>
    </div>
    <div>
      <h1>Título 1</h1>
      <p>Parrafo 1</p>
      <p>Parrafo 2</p>
    </div>
    <div>
      <h1>Título 1</h1>
      <p>Parrafo 1</p>
      <p>Parrafo 2</p>
    </div>
  </div>
</div>
```

Aquí, por ejemplo, tenemos un elemento `<div>` que contiene otros `<div>`, cada uno de los cuales contienen un `<h1>` y dos `<p>`.

Siempre hay que tratar de tener una sangría en cada nivel nuevo de anidación, para mejorar la lectura.

La etiqueta ``

La etiqueta ``, como `
`, es 'auto-cerrable', esto significa, que cuando la escribamos, no hace falta usar otra etiqueta de cierre. Es válido escribirla con la barra al final, o sin ella, si no nos preocupa que nuestro HTML sea también XML válido:

```
<img />  
<img>
```

La etiqueta de imagen por si sola, no hace nada. Depende de tener atributos para poder mostrar contenido. El atributo principal para que una imagen sea mostrada es 'src', que viene del inglés, 'source', que significa 'fuente', y es justamente la fuente del archivo que queremos colocar en el documento html. Supongamos que tenemos una imagen 'tigre.jpg' en el mismo directorio que el documento html, para colocarla, necesitaríamos escribir el siguiente código:

```

```

Esta 'fuente' de la imagen debe ser una URL válida. En este caso, es una URL relativa al documento html, la imagen del tigre está en la misma carpeta que el archivo html y por lo tanto no es necesario incluir más información sobre la ubicación de esta. Si estuviera en una subcarpeta, 'animales', tendríamos que hacer algo como esto:

```

```

Con lo que estamos diciendo que la imagen del tigre está en la subcarpeta animales.

También podríamos poner direcciones absolutas. Es decir, si por algún motivo queremos incluir la imagen directamente de un sitio externo, podemos hacer algo como esto:

```

```

Sin embargo, a menos que la web externa lo permita expresamente, está considerado 'de mala educación' tomar una imagen de un sitio externo al nuestro. En el ejemplo dado, está bien porque precisamente estamos tomando la imagen de un sitio que se dedica a albergar imágenes.

También es muy recomendable agregar un texto alternativo que provea una breve descripción de la imagen para quienes no pueden verla. En otras palabras, es un texto alternativo que sirve para presentar la misma información que la imagen.

```

```

Hipervínculos

La etiqueta <a> obtiene su nombre del inglés 'anchor', o ancla. Para funcionar necesita la propiedad 'href' que requiere una dirección válida (puede ser absoluta o relativa).

```
<a href="http://www.google.com.ar" target="_blank" rel="nofollow">
  Enlace absoluto</a>
<a href="relativo.html"> Enlace relativo </a>
<a href="../relativo.html">
  Enlace relativo, apuntando un nivel arriba del directorio actual
</a>
<a href="relativo/relativo.html"> Enlace relativo, a un directorio </a>
<a href="/relativo.html"> Enlace relativo, desde la base del sitio </a>
```

Tiene también otras propiedades que dependen de la de 'href', de los cuales el más importante y usado es 'target', que define si el enlace se abre en la ventana actual o en otra, generalmente no va nada si se quiere que se abra en la misma ventana, pero se coloca el valor de '_blank' si se quiere que el enlace se abra en una nueva ventana. También podemos ponerle otro nombre definido por nosotros (que, por convención, no tenga guión bajo como primer carácter), lo que provocará que todos los enlaces que se abran con ese 'target', se abran en esa ventana particular.

Otra propiedad muy usada es 'rel', que provee una descripción a la máquina de qué hacer con el enlace. Uno de los valores más usados es tal vez el 'nofollow', que sugiere a los robots de búsqueda que no sigan el enlace. Esto se utiliza generalmente para enlaces externos.

Además de URLs, también podemos colocar IDs en nuestros enlaces. Esto significa que si tenemos un elemento HTML en nuestra página con el id de 'tigres', podemos enviar al usuario a ese elemento mediante un enlace como este:

```
<a href="#tigres"> Enlace a la etiqueta cuyo id es 'tigres' </a>
```

También podemos combinar el uso de URL con ID, de la siguiente forma:

```
<a href="animales.html#tigres">
  Enlace a la etiqueta cuyo id es 'tigres' dentro de la página
  'animales.html'
</a>
```

Otra opción es incluir un mail mediante la palabra clave 'mailto', que iniciará el cliente de mail predeterminado del usuario para enviar un mail a la casilla indicada:

```
<a href="mailto:xacademy@technologywithpurpose.org/"> Para enviar un
```

```
mail </a>
```

Existen otras palabras clave como 'mailto', cuya función es ejecutar un programa del cliente y realizar una acción (Se lo ve en, por ejemplo, enlaces de descarga de torrents o en llamadas por teléfono).

Listas

Listas desordenadas

La lista de elementos se marca con la etiqueta (del inglés, unordered list), mientras que los elementos en sí de la lista se los marca con la etiqueta (de list).

```
<ul>
  <li>Rojo</li>
  <li>Verde</li>
  <li>Azul</li>
</ul>
```

Notemos que en el código fuente no se escriben los punteros de las listas. Estos se agregan automáticamente por el navegador.

Listas ordenadas

Para marcar la lista ordenada usamos la etiqueta (del inglés, ordered list), y como en las listas desordenadas, también usamos la etiqueta para los elementos de la lista. Se las usa para listar elementos que necesiten un cierto orden, por ejemplo, eventos o procesos:

```
<ol>
  <li>Primera Guerra Mundial (1914-1918)</li>
  <li>Segunda Guerra Mundial (1939-1945)</li>
  <li>Guerra fría (1947-1991)</li>
</ol>
```

O bien, un ranking:

```
<ol>
  <li>Michael Schumacher ( 7 veces campeón)</li>
  <li>Juan Manuel Fangio ( 5 veces campeón)</li>
```

```
<li>Alain Prost ( 4 veces campeón)</li>
<li>Sebastian Vettel ( 4 veces campeón)</li>
<li>Jack Brabham ( 3 veces campeón)</li>
</ol>
```

La etiqueta puede llevar el atributo "start" para comenzar desde un número distinto a 1.
Por ejemplo:

```
<ol start="17">
  <li>El Elemento 17</li>
  <li>El Elemento 18</li>
  <li>El Elemento 19</li>
</ol>
```

Cabe aclarar que dentro de un , pueden ir otros o . Es decir, podemos hacer algo como esto:

```
<h1>Rankings</h1>
<ul>
  <li>
    <h2>Mejores actores</h2>
    <ol>
      <li>Jack Nicholson</li>
      <li>Ralph Fiennes</li>
      <li>Daniel Day-Lewis</li>
    </ol>
  </li>
  <li>
    <h2>Mejores actrices</h2>
    <ol>
      <li>Meryl Streep</li>
      <li>Julia Roberts</li>
      <li>Julianne Moore</li>
    </ol>
  </li>
  <li>
    <h2>Mejores películas</h2>
    <ol>
      <li>Mulholland Drive</li>
    </ol>
  </li>
</ul>
```



```
<li>Schindler's List</li>
<li>Forrest Gump</li>
</ol>
</li>
</ul>
```

Ejemplo de estructura de HTML

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Título de mi página</title>
  </head>
  <body>
    <h1>El primer título</h1>
    <p>Hola mundo</p>
    <h2>Un subtítulo</h2>
    <p>
      Más texto. Esto sigue arriba. <br />
      Pero esto está abajo
    </p>
  </body>
</html>
```

Lo primero que notamos de esta pieza de código es el **doctype**.

El doctype es lo que le dice a los browsers qué tipo de documento está leyendo. Si entienden que el documento es un HTML, continúan en 'modo estándar', pero si no lo entienden, pueden entrar en el 'modo quirks', donde básicamente ejecutan la página como si fuera HTML 1.0. Como vemos, es necesario para que el documento HTML funcione correctamente, y lo mínimo necesario para que los browsers lo entiendan, es <!doctype html>, que curiosamente funciona a la perfección incluso con navegadores antiguos.

Siguiendo el orden de etiquetas en nuestro código de ejemplo, veremos que la etiqueta ‘padre’ es `<html>`. Todas las otras etiquetas derivan de esta.

Dentro de la etiqueta `<html>`, podemos distinguir dos ‘sub-etiquetas’, que son la etiqueta `<head>` y la etiqueta `<body>`. Dentro de la etiqueta `<head>` tenemos la cabecera, que es donde incluimos toda la meta-información relativa al documento, generalmente todo lo que no se ve a simple vista.

En este ejemplo, la etiqueta `<head>` tiene dos etiquetas hijas, la primera es una etiqueta `<meta>` que trae un atributo `charset` de ‘utf-8’. Con esto le informamos al navegador que el documento usa la codificación de caracteres UTF-8 (Unicode de 8 bits). Un documento puede tener muchas `<meta>` que hagan referencia a distintas cosas, como palabras clave, descripción del documento y mucho más. La segunda etiqueta ‘hija’ de `<head>` es la etiqueta `<title>`, que es justamente el título de la página. No confundir con los `h1- h6`, que son títulos dentro del texto, `<title>` es el título del documento general, se puede mostrar como nombre de la pestaña.

En la etiqueta `<body>` está el contenido principal, lo que vemos en la ventana de nuestro navegador. Como su nombre lo indica, es el cuerpo del documento y es dentro de esta etiqueta donde escribiremos la mayoría de nuestras etiquetas hijas. En este ejemplo, `<body>` tiene cuatro etiquetas hijas, `<h1>`, `<p>`, `<h2>` y otro `<p>` que a su vez tiene una etiqueta ‘autocerrable’ `
`. Las etiquetas de `<h1>` al `<h6>` son títulos. Van de mayor importancia a menor, `<h1>` siendo el título de mayor tamaño y `<h6>` la de menor. Las etiquetas `<p>` son de párrafos. El `
` es un salto de línea. Necesitamos de esta etiqueta si queremos hacer saltos de línea. Escribir directamente en la siguiente línea NO es interpretado como un salto de línea, como podemos ver en el ejemplo.

Próximos pasos

- Elementos semánticos
- Accesibilidad
- Formularios
- Tablas