

# Introducción al Diseño Web:

# HTML 5

Área Informática  
Escuela “Padre José Margis”  
Tercer año  
2013

**Autor: Ing Juan S. Marquez**

- Este documento fue realizado íntegramente con Software Libre
- Este documento está impreso en doble faz, para ahorrar papel, logrando que menos árboles sean talados, y reduciendo la contaminación producida por la industria papelera.
- Esta obra está bajo una licencia **Creative Commons BY-SA**. Ud. es libre de copiar, distribuir, exhibir y ejecutar la obra; así como de hacer obras derivadas de la misma, siempre que atribuya correctamente la autoría y redistribuya las obras derivadas bajo esta misma licencia.

## ¿Qué es HTML?

HTML significa HyperText Markup Language (Lenguaje de marcado de Hipertexto) y es la última versión del lenguaje HTML, utilizado para crear páginas web. HTML es un lenguaje *semántico*, lo que quiere decir que **no definimos el aspecto de las cosas, sino lo que significan**. Por ejemplo, si tenemos el título de nuestra página, en lugar de decir *"Que sea grande y en letras rojas"*, le indicamos al navegador que *"Este es el título principal de la página. Hacé algo para que se destaque"*. Y ese *"algo"* lo dejamos a decisión del navegador.

Obviamente, podemos controlar el aspecto que tienen nuestras páginas, pero eso es tarea de las hojas de estilo CSS, no del código HTML.

## ¿Cómo funciona HTML?

HTML es un lenguaje basado en etiquetas (en inglés, *tags*). Una etiqueta tiene la siguiente forma:

`<etiqueta>Algo aquí dentro </etiqueta>`

Las etiquetas se escriben siempre con minúsculas. La etiqueta para poner el título principal en la página es `<h1>`. Resultaría:

`<h1>Título principal</h1>`

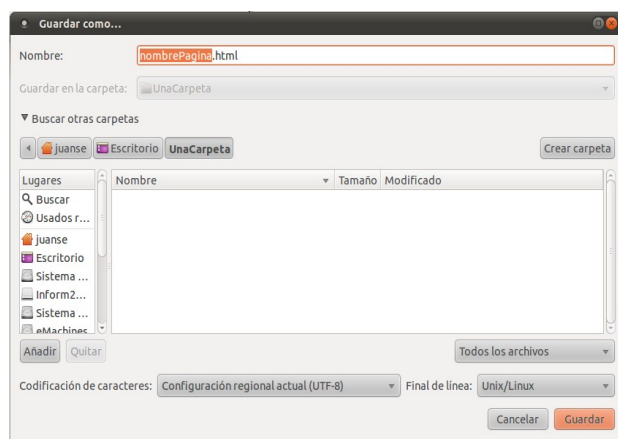
**<h1>** marca el inicio de la etiqueta, mientras que **</h1>** se encarga de cerrarla. Hay etiquetas que funcionan con una sola parte, y son así:

**<etiqueta>** (sin que exista un **</etiqueta>**)

## ¿Cómo crear un documento HTML?

Existe software especializado, pero es suficiente con un editor de texto sencillo como *gedit* o *nano* (GNU/Linux) o *Bloc de Notas* (Windows). **No utilizar un procesador de textos** (como *Word* o *Writer*).

Luego de haber escrito el código, debemos guardar el archivo. Podemos ponerle cualquier nombre, pero debemos darle la extensión **".html"**. Además, si usamos Windows, debemos guardarlo como archivo genérico ("Todos los archivos"). En GNU/Linux no es necesario tomar esta precaución.



## Estructura básica de un documento:

Explicaremos la estructura básica de cualquier documento HTML. Por lo general, es lo que copiamos y pegamos en el editor de texto al empezar con una página. No hace falta memorizar la sintaxis, basta con comprender algunos puntos. La estructura es la siguiente:

```
<!DOCTYPE html>
<html lang="es-ar">
<head>
<meta charset="utf-8">
<title>Título de la ventana del navegador </title>
</head>
<body>
(Aquí va el contenido)
</body>
</html>
```

### <!DOCTYPE html>

Sirve para avisarle al navegador qué tipo de archivo va a abrir. En este caso, HTML 5.

### <html lang="es-ar">

La etiqueta html indica el comienzo de la página. Su atributo indica el idioma ("es-ar" significa "español de Argentina").

**<head>** Comienza el encabezado.

### <meta charset="utf-8">

Indica la codificación de caracteres. UTF-8 es la recomendada para evitar problemas con los caracteres que no existen en inglés (ñ, Ñ, á, ü, ì, i, etc.).

### <title>Título de la ventana del navegador</title>

Título de la página. Aparece en la barra de título del navegador.

**</head>** Termina el encabezado.

**<body>...</body>** Comienza y termina el cuerpo de la página.

**</html>** Termina la página.

## Etiquetas básicas

```
<p>Párrafo </p>
<br> Salto de línea
<h1> Título de nivel 1 (el más
importante)</h1>
<h2> Título de nivel 2 (subtítulo) </h2>
...
<h6> Título de nivel 6 </h6>
<em>Texto enfatizado</em>
<strong>Texto muy
importante</strong>
<blockquote> Una cita textual
</blockquote>
<q> Una cita textual, pero muy breve </q>
<cite> El autor de una cita </cite>
<code>Código de una página o programa</code>
<!-- Esto es un comentario -->
```

## Caracteres especiales

&gt;	>	&frac14;	¼
&lt;	<	&frac12;	½
&amp;	&	&frac34;	¾
&nbsp;	Espacio	&quot;	"
&euro;	€	&ntilde;	ñ
&aacute;	á	&Ntilde;	Ñ
&eacute;	é	&iacute;	í
&oacute;	ó	&uacute;	ú

Estos caracteres se utilizan para casos muy particulares. La ñ y las tildes se pueden escribir directamente si se utiliza la codificación UTF-8

## Elementos de bloque y elementos en línea en HTML

HTML clasifica a todos los elementos en dos grandes grupos: elementos en línea (*inline elements*) y elementos de bloque (*block elements*).

**Bloque:** Los elementos de bloque siempre empiezan en una nueva línea y ocupan todo el espacio disponible hasta el final de la línea, aunque sus contenidos no lleguen hasta el final de la línea. La mayoría de elementos de bloque pueden contener elementos en línea y otros elementos de bloque. Por ejemplo, dentro de un párrafo `<p>` puede haber un texto con énfasis `<em>`.

**En línea:** Los elementos en línea empiezan en la misma línea, y solo ocupan el espacio necesario para mostrar sus contenidos. Los elementos en línea sólo pueden contener texto u otros elementos en línea.

### Lista de elementos de bloque y de línea:

Algunas etiquetas no se explicaron todavía. La lista sirve como referencia.

Elementos de bloque:	Elementos en línea:
blockquote	a
div	abbr
h1 ... h6	br
ol	cite
p	code
table	dfn
ul	em
li	img
td	q
th	span
tr	strong

## Atributos

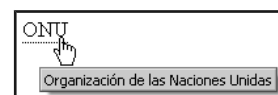
Un atributo es un valor que se añade a una etiqueta. El valor siempre va entre comillas. Tiene la forma

atributo="valor"

Como ejemplo del uso de atributos veremos dos etiquetas. La etiqueta `<abbr>` sirve para definir abreviaciones:

`<abbr title="Organización de las Naciones Unidas">ONU</abbr>`

Cuando pasamos el ratón sobre la abreviatura aparece el valor del atributo "title". Sobre la palabra ONU aparecerá un cartelito con la leyenda "Organización de las Naciones Unidas"



La etiqueta `<dfn>` es similar, pero para definiciones:

`<p>El <dfn title="Marsupial australiano">koala</dfn> está en peligro de extinción.</p>`



## Enlaces

Si queremos enlazar a una página o archivo que está en otro servidor (normalmente sitios que no son nuestros), usamos `<a>` de esta forma:

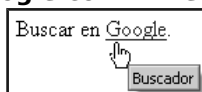
**`<a href="http://www.algo.com" title="Descripción"> Texto del enlace </a>`**

El atributo `href` contiene la URL (dirección) a la que queremos enlazar. Es muy importante que no se nos olvide el protocolo (en este caso `http://`).

En `title` escribimos una descripción de la página que enlazamos. En la mayoría de navegadores este título aparecerá al pasar el ratón por encima del link. No hay que confundir el título con el texto del enlace. Son completamente independientes.

Por ejemplo, para enlazar a Google:

Buscar en **`<a href="http://www.google.com" title="Buscador">Google</a>`**.



Para enlazar a una página que esté en nuestro servidor, necesitamos saber la ruta (*path*) desde donde estemos hasta la ubicación del archivo.

Si el origen (página donde está el link) está en la misma carpeta que el destino (página a la que apunta el link), entonces sólo tenemos que escribir su nombre:

**`<a href="perfil.html" title="Información sobre mí">Ficha personal </a>`**

Si el destino estuviera en un subdirectorio, utilizaríamos una barra `/` para indicar el camino<sup>1</sup>:

**`<a href="galeria/color.html" title="Galería color">Ver dibujos a color </a>`**

Si el destino estuviera una carpeta por encima, pondríamos dos puntitos y una barra `../` de esta manera:

**`<a href="../index.html" title="Página principal"> Volver al inicio </a>`**

Si queremos que el enlace se abra en una nueva pestaña del navegador, debemos agregar el atributo `target="_blank"`.

**`<a href="http://google.com" title="Buscador" target="_blank"> Buscar en Google </a>`**

## Anclas

Las anclas nos permiten enlazar a una posición concreta de la página. Primero creamos el ancla en el sitio al que queremos enlazar. Para esto, usamos el atributo `id` (podemos incluirlo prácticamente en cualquier etiqueta).

**`<h3 id="comentarios">Lista de comentarios </h3>`**

Ahora creamos un enlace que nos transporte al ancla. Ponemos en `href` el "id" de antes precedido de una almohadilla (`#`):

**`<a href="#comentarios" title="Lista de comentarios"> Leer comentarios </a>`**

También podemos enlazar a un ancla que esté en otra página:

**`<a href="post005.html#comentarios" title="Lista de comentarios">Comentarios del post número 5</a>`**

---

<sup>1</sup> En la Web se usa la convención de los sistemas GNU/Linux para las rutas de archivos (`/`, respetar mayúsculas y minúsculas, etc.)

## Listas

Las **listas numeradas** muestran sus elementos numerados. Se crean con la etiqueta `<ol>`:

**<p>Mis escritores favoritos (en orden de preferencia):</p>**

```
<ol>
<li>José Hernández</li>
<li>J.R.R. Tolkien</li>
<li>Isaac Asimov</li>
</ol>
```

Mis escritores favoritos (en orden de preferencia):

1. José Hernández
2. J.R.R. Tolkien
3. Isaac Asimov

### Listas sin numerar

Las listas sin numerar marcan cada elemento con un ítem, de modo que no se sigue una numeración. Se crean con la etiqueta `<ul>`.

**<p>Mis escritores favoritos:</p>**

```
<ul>
<li>José Hernández</li>
<li>J.R.R. Tolkien</li>
<li>Isaac Asimov</li>
</ul>
```

Mis escritores favoritos (en orden de preferencia):

- José Hernández
- J.R.R. Tolkien
- Isaac Asimov

Las listas pueden *anidarse*, es decir, definir una lista dentro de otra:

**<p>Los mejores libros de mis escritores favoritos:</p>**

```
<ol>
<li>José Hernández</li>
  <ul>
    <li>El gaucho Martín Fierro</li>
    <li>La vuelta de Martín Fierro</li>
  </ul>
<li>J.R.R. Tolkien</li>
  <ul>
    <li>El Señor de los anillos</li>
    <li>El hobbit</li>
  </ul>
<li>Isaac Asimov</li>
  <ul>
    <li>El hombre Bicentenario</li>
    <li>Fundación</li>
  </ul>
</ol>
```

Los mejores libros de mis escritores favoritos:

1. José Hernández
  - El gaucho Martín Fierro
  - La vuelta de Martín Fierro
2. J.R.R. Tolkien
  - El Señor de los anillos
  - El hobbit
3. Isaac Asimov
  - El hombre Bicentenario
  - Fundación

## Tablas básicas

Las principales etiquetas de las que disponemos para crear tablas son estas:

`<table>`: crea la tabla  
`<caption>`: pone título a la tabla  
`<tr>`: crea una fila  
`<td>`: crea una celda  
`<th>`: crea una celda de encabezamiento

Lo veremos con un ejemplo. En la siguiente página tenemos una tabla de dos filas y dos columnas:

```

<table>
<caption>Libros </caption>
<tr>
  <th>Título </th>
  <th>Autor</th>
</tr>
<tr>
  <td>La Odisea</td>
  <td>Homero</td>
</tr>
</table>

```

Daríá como resultado:

**Libros**

Título	Autor
La Odisea	Homero

Por cada fila, abrimos una etiqueta `<tr>` e insertamos allí las celdas que correspondan. Dentro de cada celda podemos incluir prácticamente cualquier cosa, pero debemos tener siempre en mente que el objetivo de las tablas es tabular información, no "acomodar" objetos en la página.

Las tablas se ven sin bordes, salvo que modifiquemos la etiqueta que da comienzo a la tabla del siguiente modo: **`<table border="1">`**

### Combinar celdas

Muchas veces necesitamos que una celda ocupe más de un espacio. ¿Qué pasa si en vez de un autor por cada libro, queremos mencionar tres? Podemos hacer que la celda de autor ocupe tres columnas. Así: `<table>`

```

<caption>Libros </caption>
<tr>
  <th>Título </th>
  <th colspan="3">Autor</th>
</tr>
<tr>
  <td>Cuentos varios</td>
  <td>E. A. Poe</td>
  <td>Horacio Quiroga</td>
  <td>Julio Cortázar</td>
</tr>
</table>

```

Libros			
Título	Autor		
Cuentos varios	E. A. Poe	Horacio Quiroga	Julio Cortázar

Para expandir una celda por varias columnas hemos utilizado el atributo `colspan`. Podemos hacer lo mismo con las filas, mediante `rowspan`. Vamos a hacer la misma tabla de antes, pero girada 90 grados:

```

<table>
<caption>Libros </caption>
<tr>
  <th>Título </th>
  <td>Cuentos varios</td>
</tr>
<tr>
  <th rowspan="3">Autor</th>
  <td>E. A. Poe</td>
</tr>
<tr>
  <td>Horacio Quiroga</td>
</tr>
<tr>
  <td>Julio Cortázar</td>
</tr>
</table>

```

Libros	
Título	Cuentos varios
Autor	E. A. Poe
	Horacio Quiroga
	Julio Cortázar

## Imágenes.

Las imágenes son un elemento importante a la hora de hacer más atractivo un sitio web, o de aportar más información. No obstante, hay que saber cuándo utilizarlas y no abusar de ellas.

Podemos usar tres formatos de imagen: GIF, JPEG y PNG.

El JPEG es el más adecuado para imágenes con muchos colores, como fotografías, y que no tengan grandes áreas de colores planos. No soporta transparencias ni animaciones.

Las imágenes GIF sólo pueden almacenar hasta 256 colores diferentes, pero uno de estos colores puede ser transparente. La imagen puede ser animada.

El formato PNG es el estándar y podemos elegir varias profundidades de paleta (número de colores), e incluir imágenes con transparencias, pero no animaciones.

Para poner una imagen, hacemos uso de la etiqueta `<img>` con unos cuantos atributos: ``

Con `src` establecemos qué imagen queremos mostrar. Al igual que con los links, hay que tener en cuenta la ruta hacia la imagen.

Por motivos de organización, normalmente tendremos las imágenes en una carpeta (o varias) llamada, por ejemplo, *imagenes*, así que escribiríamos `src="imagenes/algo.gif"`.

El atributo `alt` contiene una descripción de la imagen, que veremos cuando no se haya podido cargar por cualquier motivo. Este atributo es obligatorio, por cuestiones de accesibilidad: hay personas con problemas de visión, otras que deshabilitan las imágenes para ahorrar tiempo; otras, usan navegadores de texto como `w3m`.

`title` lleva una descripción que se muestra al pasar el ratón por encima de la imagen.

También podemos hacer **que una imagen sea**, a su vez, **un enlace** a una página. Los navegadores suelen mostrarla con un reborde para indicarnos que se trata de un link, pero lo podemos cambiar con CSS. Para poner una imagen como un link, la introducimos dentro de la etiqueta `<a>`:

```
<a href="http://www.escuelapadrese.com.ar" title="Sitio de la escuela">  
  </a>
```

## <audio> y <video>

De manera similar a `<img>`, HTML5 permite insertar audio y video. Estas etiquetas no existían en versiones anteriores a HTML, y todavía hay navegadores Web que no soportan todos los formatos, por lo que no siempre funcionarán. Ejemplos:

```
<video src="mi_video.mp4" controls></video>  
<audio src="mi_sonido.mp3" controls></audio>
```



## Clases e identificadores

Las clases sirven para dividir un elemento en categorías. Por ejemplo, el elemento `<p>` define párrafos, pero podríamos querer que nuestra página tenga distintas clases de párrafo, por ejemplo: "importante" y "explicación". Se utiliza el atributo "class":

`<p class="importante">Este es un párrafo importante.</p>`

`<p class="explicacion">Este es un párrafo de explicación.</p>`

El atributo *id* identifica a un elemento. Por lo tanto, no puede haber dos elementos con el mismo valor de *id*. Ejemplo

`<p id="principal">... </p>` (No puede haber otro elemento con *id*="principal").

## Etiquetas de sección

Son etiquetas de bloque invisibles, que sirven para indicar que comienza una sección determinada del documento. Su utilidad es la de organizar el código de la página. Luego podremos darle formato utilizando el lenguaje CSS. Las etiquetas son las siguientes:

- **<article>** representa un artículo, es decir, el o los textos que dan sentido a la página.
- **<aside>** representa un elemento de importancia secundaria en la página, como puede ser una barra lateral, por ejemplo.
- **<header>** representa el encabezado de la página. Típicamente lleva su título, logotipo, etc.
- **<footer>** representa el pie de página.
- **<nav>** representa un menú de navegación (una lista de enlaces que nos llevan a las distintas páginas del sitio).
- **<section>** representa cualquier otra sección de la página no contemplada en los casos anteriores.



## <div> y <span>

Son etiquetas invisibles, la primera de bloque y la segunda en línea. Sirven para dar un formato específico a un sector de la página mediante CSS.

`<p>Resalto las <span class="importante">palabras importantes</span> que contiene</p>`

`<!-- Luego, con CSS daremos formato a todos los <span> de clase "importante" -->`