

Conceptos de

HTML

El lenguaje HTML

Como ya se ha dicho, este lenguaje estructura documentos. La mayoría de los documentos tienen estructuras comunes (títulos, párrafos, listas...) que van a ser definidas por este lenguaje mediante tags. Cualquier cosa que no sea una tag es parte del documento mismo.

Este lenguaje no describe la apariencia del diseño de un documento sino que ofrece a cada plataforma que le de formato según su capacidad y la de su navegador (tamaño de la pantalla, fuentes que tiene instaladas...). Por ello y para no frustrarnos, no debemos diseñar los documentos basándonos en como lucen en nuestro navegador sino que debemos centrarnos en proporcionar un contenido claro y bien estructurado que resulte fácil de leer y entender.

No se desespere por lo que acaba de leer. HTML tiene dos ventajas que lo hacen prácticamente imprescindibles a la hora de diseñar una presentación web: Su compatibilidad y su facilidad de aprendizaje debido al reducido número de tags que usa.

Básicamente, los documentos escritos en HTML constan del texto mismo del documento y las tags que pueden llevar atributos. Esto llevado a la práctica, vendría a ser:

`<tag> texto afectado </tag>`

La tag del principio activa la orden y la última (que será la del principio precedida del signo /) la desactiva. No todas las tags tienen principio y final pero esto lo veremos más adelante.

Editores y convertidores

Antes de comenzar al trabajar sobre un editor, le recomendaría que visionase el código fuente de nuestra página principal. Todos los navegadores dan la opción de editarla (Menú ver / Código fuente). Si visita otras páginas y visualiza su código fuente encontrará similitudes en la forma en que están organizadas las páginas y en las tags utilizadas.

¿Dónde hay que editar el código fuente? Pues, si usted es usuario de Windows le bastaría con el Bloc de Notas y si utiliza Macintosh con el Simple Text. Si utiliza procesadores de texto más potentes debe guardar sus documentos como "solo texto" ya que HTML ignora todos los espacios en blanco. Una vez guardado convierta la extensión de texto por la extensión html o htm (en los sistemas DOS).

Los convertidores se utilizan para tomar los archivos de un procesador de textos y convertirlos a HTML. Pero debido a la propia limitación de este lenguaje, por muy elegante que hagamos un documento en nuestro procesador, un convertidor no obrará milagros y quizá acabe por crear cosas ilegibles en HTML. Además, la mayoría de los convertidores no convierten imágenes y no automatizan los vínculos hacia los documentos en Web debiendo corregir esto de manera manual.

A través de Internet o de revistas especializadas, usted podrá hacerse con editores y convertidores gratuitos o de muy reducidos costes. Quizá más adelante, cuando este acostumbrado a trabajar con HTML, puedan resultarle interesantes pero eso se lo dejo a su futura elección. De momento, hágame caso, si quiere aprender HTML use solo un procesador de texto simple

Documento HTML

<HTML> <HEAD> <TITLE> <BODY>

Estructura básica de un documento HTML: Cabecera y cuerpo del documento

Tres son las tags que describen la estructura general de un documento y dan una información sencilla sobre él. Estas tags no afectan a la apariencia del documento y solo interpretan y filtran los archivos HTML.

1. **<HTML>**: Limitan el documento e indica que se encuentra escrito en este lenguaje.
2. **<HEAD>**: Especifica el prólogo del resto del archivo. Son pocas las tags que van dentro de ella, destacando la del título **<TITLE>** que será utilizado por los marcadores del navegador e identificará el contenido de la página. Solo puede haber un título por documento, preferiblemente corto aunque significativo, y no caben otras tags dentro de él. En head no hay que colocar nada del texto del documento.
3. **<BODY>**: Encierra el resto del documento, el contenido.

```
<HTML>
<HEAD>
<TITLE>Ejemplo 1</TITLE>
</HEAD>
<BODY>
Hola mundo
</BODY>
</HTML>
```

Primeros pasos

**<H1> <P>
**

Tres son la tags que describen la estructura general de un documento y dan una información sencilla sobre él. Estas tags no afectan a la apariencia del documento y solo interpretan y filtran los archivos HTML.

1. **<H1>, <H2>, <H3>...:** Titulares. Sirven para dividir el texto en secciones. Se pueden definir seis niveles de titulares, el texto que deseamos que sea un titular se pone entre las tags **<H1> Titular </H1>**. Se definen mediante las tags **<H1>.....</H1>** hasta **<H6>.....</H6>**
2. **<P>:** Párrafos. En principio, sin entrar en detalles de alineación u otras características, digamos que se definen por las tags **<P>.....<P>**. Esta tag, en un principio, se diseñó para saltar de párrafo por lo que puede ir sola "**<P>**" al final de un texto indicando que a continuación se quiere una línea en blanco aunque le recomendamos que se acostumbre a utilizarla abriéndola y cerrándola.
3. **
:** Saltos de línea. Esta tag sirve para realizar un salto de línea, puede poner tantas como desee y realizará un salto de línea por cada una de ellas.
4. **<!-- -->:** Comentarios. Son directivas que nunca se mostrarán a través del navegador y que le servirán para recordatorios en futuras revisiones del documento.

Bien, recuperemos nuestro ejemplo y agreguemos nuestros nuevos conocimientos.

```
<HTML>
<HEAD>
<TITLE>Ejemplo 2</TITLE>
</HEAD>
<BODY>
  <H1>Mi primera página</H1>

  <!-- Aquí va un comentario que no es
        interpretado por el navegador -->

  <P>Hola mundo, esta es un página con titular,
que tiene también un párrafo y unos cuantos
saltos de línea.</P>

  Uno<br>
  Dos<br>
  Tres<br>
</BODY>
</HTML>
```

Creación de enlaces

<A>

Lo característico del lenguaje HTML es el poder generar vínculos de hipertexto para enlazar con ellos todos sus documentos en web.

Para generar un enlace a otro documento necesitamos el nombre de un archivo (o su dirección URL) y el texto que servirá de punto de activación del otro documento. Este segundo elemento será el que veamos en pantalla y que se servirá del primero para saltar de documento.

Los enlaces se generan mediante la tag `<A>.....` y, a diferencia de los vistos anteriormente, llevará siempre dentro de la tag un atributo ya sea `` o ``.

1. `.....`: Es el más habitual de los atributos y sirve para saltar entre diferentes URLs. De momento veremos:
 - o Saltar en una presentación del archivo 1 al archivo 2: En el archivo 1 incluiremos la directiva `Siguiente página`
 - o Saltar de nuestra presentación a otra presentación web llamada `www.bienvenidos.es`: `Visita esta página`

2. `Primera parte`: Utilizamos el atributo name para dar nombre a una sección de nuestro documento. Posteriormente, cuando en nuestro documento queramos incluir un vínculo a dicha sección escribiremos: `Ir a la primera parte`

```
<HTML>
<HEAD>
<TITLE>Ejemplo 3</TITLE>
</HEAD>
<BODY>
  <A NAME="arriba"><H1>Página de enlaces</H1></A>
  <A HREF="#abajo">Ir abajo</A><br>

  <A HREF="ej4.html">Ir a ejemplo 4</A><br><br>
  <A HREF="http://www.webestilo.com/">Ir a WebEstilo</A><br>

  <br>.<br>.<br>.<br>.<br>.<br>.<br>.<br>.<br><br>
  <br><br><br><br><br><br><br><br><br><br><br>
  <br><br><br><br><br><br><br><br><br><br><br>
  <br><br><br>.<br>.<br>.<br>.<br>.<br>

  <A NAME="abajo"><br></A>
  <A HREF="#arriba">Ir arriba</A>

</BODY>
</HTML>
```

URLs

URL: Localizador Universal de Recursos

Los URL son las direcciones de las informaciones que buscamos en Internet. Los URL constan de tres partes:

1. **Protocolo:** Es el programa que utilizará el navegador para obtener el archivo elegido. Les suena HTTP, FTP, Gopher...
2. **Nombre del host:** Se trata del sistema donde se encuentra almacenada la información que buscamos.
3. **Ruta del fichero:** Se trata de la ubicación del archivo dentro del host.

<http://www.bienvenidos.es/publico/saludos.html>

Entre los principales tipos de URL destacan:

1. **HTTP:** Son los más populares ya que son los utilizados por los servidores de WWW para mandar documentos a través de Internet.
2. **FTP:** Se utilizan para apuntar hacia los archivos que estén en servidores que usan el protocolo FTP (File Transfer Protocol). Este protocolo es normalmente utilizado para enviar y recibir ficheros. Es el protocolo que se usa para enviar nuestras páginas al servidor de internet. Como ya se puede imaginar en estos servidores se almacenan los archivos que forman parte de nuestra presentación web.
3. **File:** Apuntan hacia archivos contenidos en el mismo disco que se encuentra el navegador. No resulta muy interesante poner estos URL en nuestras presentaciones puesto que otra persona que desde otro sistema apunte hacia este URL, generalmente fallará en su intento y no podrá tener acceso a él.
4. **Mailto:** Se usa para mandar correos electrónicos. Cuando seleccionamos este tipo de URL se abre la aplicación de correo electrónico de nuestro ordenador para enviar un correo a la dirección hacia la que apunta el URL. La forma estándar es: <mailto:webmaster@bienvenidos.es>
5. **News:** Son URL de grupos de noticias, en estos servidores se almacenan mensajes en los que se discuten sobre diferentes temas.

```
<HTML>
<HEAD>
<TITLE>Ejemplo 5</TITLE>
</HEAD>
<BODY>
```

```
<H1>Diferentes tipos de URLs</H1>
```

```
Enlace a <A HREF="http://www.webestilo.com">WebEstilo</A>.
<br>
Enlace al servidor de FTP
de <A HREF="ftp://ftp.mozilla.org/pub/mozilla/">Mozilla</A>.
<br>
Enlace a <A HREF="file:///C:">la unidad C
```

```
de su ordenador</A>.  
<br>  
Enlace a <a href="mailto:webmaster@bienvenidos.es">  
mailto:webmaster@bienvenidos.es</a>.  
<br>  
Enlace a las <A HREF="news://msnews.microsoft.com">News de Microsoft</A>.  
<br>  
</BODY>  
</HTML>
```

1. Listas

Junto con encabezados y párrafos, son otro de los elementos HTML más comunes. Pueden darse cinco tipos diferentes de listas, cada uno con tags distintas aunque con alguna en común:

- i. Listas numeradas u ordenadas: Se engloban por las tags y cada elemento de la lista estará encabezado por la tag que puede o no llevar la tag de cierre . Es conveniente que cada elemento de la lista esté en una línea nueva aunque todo seguido consiga en la presentación el mismo efecto. Cuando el navegador interpreta una lista ordenada, numera y sangra cada elemento en forma secuencial, aunque se introduzcan modificaciones.
- ii. Listas con viñetas o sin orden: Se engloban por las tags y cada elemento de la lista, también estará encabezado por la tag . El resultado es que el navegador inserta viñetas (marcadores) delante de cada elemento.
- iii. Listas de menú y de directorio: Están en desuso puesto que su resultado suele ser, prácticamente, idéntico al de las listas con viñetas.
 1. Menú: Englobadas por las tags <menu>.....</menu> y cada elemento encabezado por la tag .
 2. Directorio: Englobadas por las tags <dir>.....</dir> y cada elemento encabezado por la tag .
- iv. Listas de glosario: Cada elemento de la lista está compuesto por un término y una definición y cada una de estas partes tiene su propia tag. Estas listas se engloban con las tags <dl>.....</dl>. Para el término se usa la tag <dt> y para la definición la tag <dd>. Generalmente el navegador colocará término y definición en dos líneas diferentes pero esto se puede evitar añadiendo a la tag de apertura el atributo compact: <dl compact>.
- v. Listas anidadas: Consiste en poner una lista dentro de otra, de manera que la lista secundaria sangre respecto a la principal. Puede jugar con los diferentes tipos de lista pero recuerde estructurar bien las tags: Las tags de la lista principal englobarán todo el conjunto de las listas y las tags de las listas secundarias se cerraran antes de volver a la lista principal. Ahora quizá le empiece a convenir sangrar el propio código conforme lo va escribiendo en HTML.

```
<HTML>  
<HEAD>  
<TITLE>Ejemplo 6</TITLE>  
</HEAD>  
<BODY>  
  
<H1>Listas</H1>
```

```

Una lista ordenada
<ol>
<li>Uno</li>
<li>Dos</li>
<li>Tres</li>
</ol>
<br><br>
Una lista sin ordenar
<ul>
<li>Uno</li>
<li>Dos</li>
<li>Tres</li>
</ul>
<br><br>
Una lista de glosario
<dl>
<dt>T&eacute;mino 1</dt>
<dd>Definici&oacute;n 1</dd>
<dt>T&eacute;mino 2</dt>
<dd>Definici&oacute;n 2</dd>
</dl>
<br><br>

Listas anidadas
<ul>
<li>Uno
  <ul>
    <li>Uno</li>
    <li>Dos</li>
    <li>Tres</li>
  </ul>
</li>
<li>Dos</li>
<li>Tres</li>
</ul>
</BODY>
</HTML>

```

Estilos de caracter

** <U> <I>**

Estos estilo son tags que afectan a palabras o caracteres dentro de otras entidades de HTML modificando el aspecto de ese texto para que sea diferente del texto que lo rodea. Existen dos tipos de estilos:

1. **Estilos lógicos: Indican como se va a emplear el texto que realizamos, no el como se va a formatear.**

- [.....](#): Indica que los caracteres estarán enfatizados de alguna manera, generalmente en cursiva aunque dependerá del navegador.
- [.....](#): Los caracteres tendrán mayor énfasis, generalmente en negrita.
- [<code>.....</code>](#): Muestra como una fuente monoespaciada, generalmente Courier.
- [<samp>.....</samp>](#): Muy similar a code.
- [<kdb>.....</kdb>](#): Texto que el usuario debe escribir.
- [<var>.....</var>](#): Nombre de una variable que deba ser reemplazada por su valor real. Generalmente en cursiva o subrayada.
- [<dfn>.....</dfn>](#): Se usa para resaltar una palabra que se va a definir.
- [<cite>.....</cite>](#): Se usa para citas cortas.

2. [Estilos físicos: Modifican la presentación real del texto.](#)

- [.....](#): Pone el texto en negrita.
- [<i>.....</i>](#): Pone el texto en cursiva.
- [<tt>.....</tt>](#): Pone el texto en fuente monoespaciada.
- [<u>.....</u>](#): Subraya el texto afectado.

```

<HTML>
<HEAD>
<TITLE>Ejemplo 7</TITLE>
</HEAD>
<BODY>

<H1>Estilos de caracter</H1>
Un ejemplo de texto de <EM>realizado</EM>.<br>
Un ejemplo de texto de <STRONG>realizado</STRONG>.<br>
Un ejemplo de texto de <CODE>realizado</CODE>.<br>
Un ejemplo de texto de <SAMP>realizado</SAMP>.<br>
Un ejemplo de texto de <KDB>realizado</KDB>.<br>
Un ejemplo de texto de <VAR>realizado</VAR>.<br>
Un ejemplo de texto de <DFN>realizado</DFN>.<br>
Un ejemplo de texto de <CITE>realizado</CITE>.<br>
<br>
Un ejemplo de texto de <B>realizado</B>.<br>
Un ejemplo de texto de <I>realizado</I>.<br>
Un ejemplo de texto de <U>realizado</U>.<br>
Un ejemplo de texto de <TT>realizado</TT>.<br>

</BODY>
</HTML>

```

Texto preformateado

<PRE>

Ya dijimos que HTML elimina cualquier espacio en blanco adicional que se inserte pero nos encontramos con una excepción a esta regla cuando utilizamos las tags `<pre>.....</pre>`. Sin embargo esta tag convertirá el texto afectado a fuente monoespaciada (posiblemente Courier).

Esta tag se usaba para hacer tablas en versiones anteriores del lenguaje HTML ahora su utilidad puede reducirse a convertir a HTML, rápida y fácilmente, archivos de correo electrónico y publicaciones de grupos Usenet.

```
<HTML>
<HEAD>
<TITLE>Ejemplo
8</TITLE>
</HEAD>
<BODY>
```

```
<H1>Texto
preformateado</H1>
```

```
Diferencia entre un texto normal y
un texto
preformateado. En el texto
preformateado, se respetan los
```

```
car&aacute;cteres y los espacios.<br>
iiiiii<br>
```

```
wwwww<br>
```

```
<PRE>
Diferencia entre un texto normal y

un texto preformateado. En el texto
preformateado, se respetan los

car&aacute;cteres y los espacios.<br>
iiiiii<br>

wwwww<br>
</PRE>

</BODY>
</HTML>
```

Saltos y líneas

<HR>

1. Líneas divisorias: Se crean con la tag `<hr>` que no tiene tag de cierre ni lleva texto asociado. Se puede especificar el ancho de la línea con el siguiente atributo
`<hr width="80%">`.
2. Saltos de línea: La tag `
` inserta un salto de línea donde se coloque. Puede colocar tantas como desee y se insertará un salto de línea por cada una de ellas.

```
<HTML>
<HEAD>
<TITLE>Ejemplo 9</TITLE>
</HEAD>
<BODY>

<H1>Saltos y líneas</H1>

<HR>
<HR width="80%">
<HR width="60%">
<HR width="40%">
<HR width="20%">

----
<br><br><br>
----
</BODY>
</HTML>
```

Carácteres especiales

Es posible que en un momento dado deseemos insertar un carácter que no sea ASCII en nuestro documento HTML. Muy probablemente, si lo creamos nosotros con las combinaciones de teclas habituales no llegue a poder ser leído por otra plataforma una vez publicado en Internet. Para solventar este conflicto, HTML ofrece la posibilidad de insertar unas entidades de carácter que podrán ser interpretados por todas las plataformas. Estas entidades pueden ser:

1. Nombradas: Se suelen construir con el signo &, la abreviatura del carácter y acabado con el signo punto y coma. El símbolo de registrado se construiría: ®
2. Numeradas: Se construyen con los signos &#, el número correspondiente a la posición del carácter en el conjunto ISO-Latin-1 (ISO-8859-1). El mismo símbolo de registrado, sería ahora: ®
3. Existen unas entidades reservadas, exclusivamente, para caracteres usados en tags pero que nosotros no los pretendemos usar dentro de ninguna de ellas. Estos son:
 - o signo <, entidad <
 - o signo >, entidad >
 - o signo &, entidad &
 - o signo ", entidad "

Hemos puesto algunos ejemplos de caracteres no ASCII, si desea puede consultar la [referencia completa del conjunto de caracteres ISO-Latin-1](http://www.webestilo.com/html/referencia_completa_del_conjunto_de_caracteres_ISO-Latin-1) (http://www.webestilo.com/html/referencia_completa_del_conjunto_de_caracteres_ISO-Latin-1)

Tablas

<TABLE> <TR> <TD>

Las tablas surgieron con la versión HTML 3.0. Las tablas nos permiten representar y ordenar cualquier elemento de nuestra presentación en diferentes filas y columnas de modo que podamos resumir grandes cantidades de información de una manera que puede representarse rápida y fácilmente.

El contenido de una tabla lo debemos desarrollar entre las tags `<table>.....</table>`.

Las tablas se definen fila a fila, celda a celda, comenzando desde la celda superior izquierda. Las columnas se calcularán automáticamente según las celdas que hay en cada fila.

Cada fila de la tabla se indica mediante las tags `<tr>.....</tr>`. Las tags `<th>` y `<td>` con sus correspondientes tags de cierre, indican para indicar las filas individuales dentro de cada fila. Las tags `<th>.....</th>` indican que se trata de celdas que sirven como encabezado de tabla y suelen visualizarse en negrita. Las tags `<td>.....</td>` indican que se trata de celdas comunes.

```

<HTML>
<HEAD>
<TITLE>Ejemplo 10</TITLE>
</HEAD>
<BODY>

<H1>Tablas b&aacute;sicas</H1>

<TABLE BORDER="1">
<TR>
  <TH>Cabereca 1</TH>
  <TH>Cabereca 2</TH>
  <TH>Cabereca 3</TH>
</TR>
<TR>
  <TD>Dato 1</TD>
  <TD>Dato 2</TD>
  <TD>Dato 3</TD>
</TR>
<TR>
  <TD>Dato 4</TD>
  <TD>Dato 5</TD>
  <TD>Dato 6</TD>
</TR>
</TABLE>

</BODY>
</HTML>

```

Accesorios

Título

Cuando queramos titular una tabla, podemos escribirlo como texto normal o usando las tags `<caption>.....</caption>`. Las tags de título van dentro de las tags de la tabla y son opcionales, el título no es obligatorio. La tag de apertura puede llevar consigo el atributo `align` que indicará si el título va encima o debajo de la tabla. `align="top"` indicaría encima de la tabla y `align="bottom"` indicaría en la parte de abajo.

Alineación de celdas

Una vez colocadas las celdas, hay que alinear los datos dentro de cada celda. Así, dentro de cada tag de celda podemos encontrar:

1. El atributo `align=` define horizontalmente los datos al margen izquierdo (`left`), al derecho (`right`) o centrado (`center`).
2. El atributo `valign=` define verticalmente los datos en la parte superior (`top`), en la parte inferior (`bottom`) o centrado (`middle`).

Celdas extendidas

Para crear una celda que abarque varias filas o columnas, debemos colocar en las tags `<th>` o `<td>` los atributos:

1. `rowspan=` más un valor para indicar el número de filas que se quiere abarcar.
2. `colspan=` más un valor para indicar el número de columnas que se quiere abarcar.

Si opta por poner celdas extendidas en su presentación web, la cosa se complica bastante. Nuestra recomendación: Coja lápiz y papel y dibuje la tabla con todas las filas y columnas que se quieren formar porque así tendrá mucho más claro los valores que debe asignar a los atributos `rowspan` y `colspan` y las tags a las que hay que asignarlos.

Espaciado

Podemos modificar el aspecto de la tabla cambiando el ancho de los bordes, el espaciado entre celdas y el ancho de las mismas.

1. `width=` Acompaña a `<table>` y especifica el ancho de la tabla, tanto en número de píxeles como en porcentaje respecto al ancho de la pantalla. También puede acompañar a las tags `<th>` o `<td>` para especificar el ancho de las columnas.
2. `Border=` Anteriormente, ya hemos hablado de este atributo. Ahora le diremos que puede darle un valor que indicará el ancho del borde en píxeles. `Border="0"` indicaría la ausencia de borde.
3. `Cellspacing=` Suele acompañar a la tag `<table>`. Indica el número de píxeles que separan cada celda. El valor predeterminado suele ser 2.
4. `Cellpadding=` También acompaña a la tag `<table>`. Indica el espacio en píxeles entre el borde de la celda y su contenido. El valor predeterminado suele ser 1.

```
<HTML>
<HEAD>
<TITLE>Ejemplo 11</TITLE>
</HEAD>
<BODY>

<H1>Tablas avanzadas</H1>

<TABLE WIDTH="50%" BORDER="1" CELLSPACING="3" CELLPADDING="2">
<TR>
  <TD COLSPAN="2" ALIGN="right">Dato 1</TD>
  <TD>Un texto cualquiera</TD>
  <TD ROWSPAN="2">Dato 3</TD>
</TR>
<TR>
  <TD>Dato 4</TD>
  <TD>Dato 4</TD>
  <TD ALIGN="center">Dato 5</TD>
```

```
</TR>
</TABLE>

</BODY>
</HTML>
```

Imágenes

El uso de imágenes es uno de los factores que ha popularizado tanto World Wide Web. Incluir imágenes en una presentación web es muy sencillo, solo debe tener en cuenta que las imágenes tienen que tener los formatos GIF, JPEG o PNG. Las imágenes en línea, se especifican a partir de la tag `` que no tiene una tag correspondiente de cierre pero que puede acompañarse de los siguientes atributos:

1. `src=` Este atributo es obligatorio e indica el nombre del archivo de imagen (entre comillas) o la URL que se va a representar.
2. `Align=` Permite controlar la alineación de una imagen con respecto a una línea de texto adyacente o a otras imágenes en esa línea. Los tres valores posibles son los ya conocidos `left`, `right`, `top`, `middle` y `bottom`.
3. `Alt=` Es la alternativa que se estableció cuando todavía existían visualizadores de solo texto. Entre comillas podremos escribir un texto que suplantara a esta imagen si no se carga o mientras se carga o cuando, visualizando ya la imagen, pasamos el ratón por encima.
4. `WIDTH=` Este atributo es opcional pero es recomendable ponerlo para ayudar al navegador a representar la imagen, significa el ancho de la imagen que vamos a representar.
5. `HEIGHT=` Al igual que el atributo `WIDTH`, es opcional y recomendable ponerlo, este significa el alto de la imagen.
6. `BORDER=` Con `BORDER` especificamos el ancho de un borde que rodea la imagen.

```
<IMG SRC="/graficos/bebel.jpg" WIDTH=140 HEIGHT=210 BORDER=0 ALT="Un bebé">
```



```
<HTML>
<HEAD>
<TITLE>Ejemplo 12</TITLE>
</HEAD>
<BODY>
```

```

<H1>Imágenes</H1>

<IMG SRC="/graficos/bebel.jpg" WIDTH=140 HEIGHT=210 BORDER=0 ALT="Un bebé;" ALIGN="RIGHT">
<IMG SRC="/graficos/bebel.jpg" WIDTH=140 HEIGHT=210 BORDER=3 ALT="Un bebé;" ALIGN="LEFT">
Un texto cualquiera.
</BODY>
</HTML>

```

Mapas de imágenes

<MAP> <AREA>

Puede hacer que parte de la imagen sea un enlace a otra página, es decir, puede hacer un mapa sobre la imagen de manera que secciones de la imagen sean enlaces. Las tags usadas para esto son:

- Las tags `<map>.....</map>` identifican que vamos a crear un mapa de imágenes. Generalmente, lleva asociado el atributo `name=` al que le sigue entre comillas el nombre del mapa.
- La tag `<area>` define las áreas que vamos a poder activar en esa imagen. A esta tag le acompañan los siguientes atributos:
 - `shape=` Entre comillas estableceremos el tipo de área a definir. Puede tratarse de `rect` (rectangular), `poly` (poligonal) o `circle` (circular).
 - `Coords=` Entre comillas indicaremos los pares de coordenadas de cada punto del área a activar. Estas coordenadas las podemos averiguar utilizando un programa de edición de imágenes. En las áreas rectangulares deben especificarse las coordenadas de la esquina superior izquierda y las de la esquina inferior derecha. En las poligonales especificaremos las coordenadas de todos los vértices del área. En las circulares indicaremos las coordenadas del centro del círculo y el valor del radio.
 - `Href=` Como ya sabe, indica la dirección, entre comillas, de la página web a la que accede si pinchamos en un área determinada.



Finalmente, debe saber que para que una imagen sea tratada como un mapa, además de el código anteriormente descrito, debe incluir en la tag de imagen correspondiente a la imagen a mapear el atributo `usemap="#nombre del mapa"`.

```

<IMG SRC="/graficos/bebel.jpg" WIDTH=140 HEIGHT=210
      BORDER=0 ALT="Un bebé" USEMAP="#bebel">

<MAP NAME="bebel">

```

```
<AREA SHAPE=CIRCLE COORDS="60,56,47"
  HREF="#" ALT="Cabeza">
<AREA SHAPE=POLY COORDS="3,182,36,178,
  44,165,60,169,66,184,62,196,
  43,201,35,190,0,193,0,183"
  HREF="#" ALT="Sonajero">
</MAP>
```

```
<HTML>
<HEAD>
<TITLE>Ejemplo 13</TITLE>
</HEAD>
<BODY>

<H1>Mapas de im&aacute;genes</H1>

<IMG SRC="/graficos/bebel.jpg" WIDTH=140 HEIGHT=210 BORDER=0 ALT="Un beb&eacute;" USEMAP="#bebel">

<MAP NAME="bebel">
<AREA SHAPE=CIRCLE COORDS="60,56,47" HREF="#" ALT="Cabeza">
<AREA SHAPE=POLY COORDS="3,182,36, 178,44,165, 60,169,66, 184,62,196, 43,201,35, 190,0,193,0,183" HREF="#" ALT="Sonajero">
</MAP>

</BODY>
</HTML>
```

Formularios <FORM>

La Web se ha convertido en una poderosa arma para las empresas que se dedican a realizar encuestas y, los formularios han sido una de las herramientas que han ayudado a este auge.

Los formularios nos van a permitir, desde dentro de una presentación web, solicitar información al visitante. Estos formularios estarán compuestos por tantos campos como informaciones queramos obtener. Una vez introducidos los valores en estos campos serán enviados a una URL donde se procesará toda esta información.

Un formulario sigue siendo lenguaje html y por lo tanto necesita de unas tags que lo especifiquen. La declaración de formulario queda recogida por las tags <form>.....</form> y dentro de ellas se recogerán todas las variables de entrada.

A la tag de apertura `<form>` le acompañarán estos atributos:

1. `action=""` Entre comillas se indica el programa que va a tratar las variables enviadas con el formulario, un guión CGI o la URL mailto.
2. `Method=""` Indica el método de transferencia de las variables. `Post`, si se envía a través del STDIO. `Get`, si se envía a través de la URL.

```
<HTML>
<HEAD>
<TITLE>Ejemplo 14</TITLE>
</HEAD>
<BODY>

<H1>Formularios</H1>

<FORM ACTION="mailto:unaprueba" METHOD="POST">
<INPUT TYPE="text" NAME="nombre"><BR>
<INPUT TYPE="submit"><INPUT TYPE="Reset">
</FORM>

</BODY>
</HTML>
```

Campos de entrada

<INPUT>

La tag `<input>` define la introducción de variables. Junto a esta tag encontraremos los siguientes atributos:

1. `type=""` Indicará el tipo de variable a introducir.
 - o `text` Indica que el campo a introducir será un texto. Sus atributos:
 - `maxlength=""` Seguido de un valor que limitará el número máximo de caracteres a introducir en ese campo.
 - `size=""` Seguido de un valor que limitará el número de caracteres a mostrar en pantalla.
 - `value=""` Indica que no hay valor inicial del campo.
 - o `Password` Indica que el campo a introducir será una palabra de paso. Mostrará asteriscos en lugar de letras escritas. Sus atributos serán los mismos que para `text`.
 - o `Checkbox` El campo se elegirá marcando de entre varias opciones una casilla cuadrada.
 - `value=""` Entre comillas se indicará el valor de la casilla.
 - `checked` La casilla aparecerá marcada por defecto.
 - o `Radio` El campo se elegirá marcando de entre varias opciones una casilla circular.
 - `value=""` Entre comillas se indicará el valor de la casilla.

- o **Image** El campo contendrá el valor en coordenadas del punto de la imagen que haya pinchado. Atributo obligatorio:
 - **src=" "** Entre comillas escribiremos el nombre del archivo de imagen.
- o **hidden** El visitante no puede modificar su valor ya que no está visible. Se manda siempre junto al atributo value= seguido de su valor entre comillas.

2. **Name=" "** Indicará el nombre que se asigna a un determinado campo.

```
<HTML>
<HEAD>
<TITLE>Ejemplo 15</TITLE>
</HEAD>
<BODY>

<H1>Formularios</H1>

<FORM ACTION="mailto:unaprueba" METHOD="POST">
Texto: <INPUT TYPE="text" NAME="nombre"><BR>
Password: <INPUT TYPE="password" NAME="contra"><BR>
Sexo:<INPUT TYPE="radio" NAME="boton1" VALUE="1"> Hombre
<INPUT TYPE="radio" NAME="boton1" VALUE="2">Mujer<BR>
Vehiculo:<INPUT TYPE="checkbox" NAME="Moto" VALUE="Si">Moto
<INPUT TYPE="checkbox" NAME="Coche" VALUE="" CHECKED>Coche

<BR><BR>
<INPUT TYPE="submit"><INPUT TYPE="Reset">
</FORM>

</BODY>
</HTML>
```

Campos de selección

<SELECT>

Las tags `<select>.....</select>` encierran los valores que podremos elegir a partir de una lista. Los atributos que acompañan a la tag de apertura son:

1. **name=" "** Indicará el nombre del campo de selección.
2. **Size=" "** Indicará el número de opciones visibles. Si le asignamos 1, la selección se presentará como un menú desplegable. Si le asignamos un valor mayor se presentará como una lista con barra de desplazamiento.
3. **multiple** Indica si se pueden realizar multiples selecciones.

Las diferentes opciones de la lista se indicarán mediante la tag `<option>` que puede acompañarse del atributo `selected` para indicar cual es la opción que aparecerá por defecto. Si no lo especificamos, siempre será la primera de la lista.

```

<HTML>
<HEAD>
<TITLE>Ejemplo 16</TITLE>
</HEAD>
<BODY>

<H1>Formularios</H1>

<FORM ACTION="mailto:unaprueba" METHOD="POST">
<SELECT NAME="Colores" MULTIPLE>
  <OPTION VALUE="r">Rojo</OPTION>
  <OPTION VALUE="g">Verde</OPTION>
  <OPTION VALUE="b">Azul</OPTION>
</SELECT>
<BR><BR>
<SELECT NAME="Colores" SIZE="1">
  <OPTION VALUE="r">Rojo</OPTION>
  <OPTION VALUE="g">Verde</OPTION>
  <OPTION VALUE="b">Azul</OPTION>
</SELECT>
<BR><BR>
<INPUT TYPE="submit"><INPUT TYPE="Reset">
</FORM>

</BODY>
</HTML>

```

Areas de texto

<TEXTAREA>

Con las tags `<textarea>`;.....`</textarea>` definimos un texto de múltiples líneas para que el visitante pueda incluir un comentario junto a sus datos.

Junto a la tag de apertura pueden aparecer los siguientes atributos:

1. `name=" "` Nombre del campo.
2. `cols=" "` Numero de columnas de texto visible.
3. `rows=" "` Numero de filas de texto visible.

```

<HTML>
<HEAD>
<TITLE>Ejemplo 16</TITLE>
</HEAD>
<BODY>

<H1>Formularios</H1>

```

```
<FORM ACTION="mailto:unaprueba" METHOD="POST">
<TEXTAREA COLS=20 ROWS=10 NAME="Texto">
</TEXTAREA>
<BR><BR>
<INPUT TYPE="submit"><INPUT TYPE="Reset">
</FORM>

</BODY>
</HTML>
```

Botones

Se definen mediante la tag `<input>` a la que le acompañan los atributos:

1. `type=" "` Seguido de `submit` para enviar los datos del formulario y seguido de `reset` para borrar los datos que se han introducido.
2. `Value=" "` Indica el texto que incorporaran los botones. Normalmente, enviar y borrar.

```
<HTML>
<HEAD>
<TITLE>Ejemplo 16</TITLE>
</HEAD>
<BODY>

<H1>Formularios</H1>

<FORM ACTION="mailto:unaprueba" METHOD="POST">
<TEXTAREA COLS=20 ROWS=10 NAME="Texto">
</TEXTAREA>
<BR><BR>
<INPUT TYPE="submit"><INPUT TYPE="Reset">
</FORM>

</BODY>
</HTML>
```

Lista de las principales etiquetas Html

Aquí esta una lista de las principales etiquetas Html

Puesta en forma de los caracteres

`...`

Texto en negrita

`<BIG>...</BIG>`

Ampliación del tamaño de los caracteres

`<BLINK>...</BLINK>`

Texto parpadeante (Netscape solo)

`...`

Texto en itálico

``

Texto en color donde

`...`

XXXXXX es un valor hexadecimal

`...`

Tamaño de los caracteres donde

X es un valor de 1 a 7

`<I>...</I>`

Texto en itálico

`<NOBR>...</NOBR>`

Impide las rupturas automáticas
de línea de los browser

`<PRE>...</PRE>`

Texto preformateado, o sea con una visualización
de todos los espacios y saltos de línea

`<SMALL>...</SMALL>`

Reducción del tamaño de los caracteres

`...`

Puesta en negrita del texto

`_{...}`

Texto en indicio

`^{...}`

Texto en exponente

`<U>...</U>`

Texto subrayado

Puesta en forma del texto

`<!--...-->`

Comentarios ignorado por el navegador

`
`

A la línea

`<BLOCKQUOTE>...`

Citación (introduce un retracto de texto)

<code></BLOCKQUOTE></code>	
<code><CENTER>...</CENTER></code>	Centra cada elemento comprendido en la etiqueta
<code><DIV align=center> ...</DIV></code>	Centra el elemento encuadrado por la etiqueta
<code><DIV align=left> ...</DIV></code>	Alinea el elemento a la izquierda
<code><DIV align=right> ...</DIV></code>	Alinea el elemento a la derecha
<code><Hx>...</Hx></code>	Título o x tiene un valor de 1 à 7
<code><Hx align=center>...</Hx></code>	Título centrado
<code><Hx align=left>...</Hx></code>	Título alineado a la izquierda
<code><Hx align=right>...</Hx></code>	Título alineado a la derecha
<code><P>...</P></code>	Nuevo párrafo
<code><P align=center>...</P></code>	Párrafo centrado
<code><P align=left>...</P></code>	Párrafo alineado a la izquierda
<code><P align=right>...</P></code>	Párrafo alineado a la derecha
Listas	
<code></code>	Lista no numerada
<code></code>	Elemento de lista
<code></code>	
<code></code>	Lista numerada
<code></code>	Elemento de lista
<code></code>	
<code><DL></code>	Lista de glosario
<code><DT>...</DT></code>	Término de glosario (sin retracto)
<code><DD>...</DD></code>	Explicación del término (con retracto)
<code></DL></code>	
Rayas	
<code><HR></code>	Línea de separación. Raya horizontal
<code><HR width="x%"></code>	Anchura de la raya en %
<code><HR width=x></code>	Anchura de la raya en píxeles
<code><HR size=x></code>	Altura de la raya en píxeles
<code><HR align=center></code>	Raya centrada

<HR align=left>
<HR align=right>
<HR noshade>

Raya alineada a la izquierda
Raya alineada a la derecha
Raya sin efecto de sombreado

Enlaces

...
...
...

...
...
...

Enlace hacia una página Web
Enlace hacia una dirección Email
Enlace hacia la página fichero.htm
situada en el mismo directorio
Definición de una ancla
Enlace hacia una ancla

Imágenes

Inserción de una imagen al formato Gif o Jpg
(ver enlaces para la dirección)
Puesta a la escala de la imagen en pixeles
Definición del borde de una imagen con un enlace
Texto alternativo cuando la imagen no esta mostrada
Alinea la imagen abajo
Alinea la imagen en el medio
Alinea la imagen arriba
Alinea la imagen a la izquierda
Alinea la imagen a la derecha
Espaciamiento horizontal entre la imagen y el texto
Espaciamiento vertical entre la imagen y el texto

Tablas

<TABLE>...</TABLE>
<TABLE width="x%">
<TABLE width=x>
<TABLE border=x>

Definición de una tabla
Anchura de la tabla en %
Anchura de la tabla en pixeles
Anchura del borde

<code><TABLE cellpadding=x></code>	Espacio entre el borde y el texto
<code><TABLE cellspacing=x></code>	Espesor de la raya entre las celdas
<code><TR>...</TR></code>	Línea de la tabla
<code><TD>...</TD></code>	Celda de la tabla
<code><TD bgcolor="#XXXXXX"></code>	Color de una celda de la tabla
<code><TD width="x%"></code>	Anchura de columna en %
<code><TD width=x></code>	Anchura de columna en pixeles
<code><TD align=center></code>	Texto centrado en la celda
<code><TD align=left></code>	Texto alineado a la izquierda en la celda
<code><TD align=right></code>	Texto alineado a la derecha en la celda
<code><TD valign=bottom></code>	Alineación hacia arriba del contenido de la celda
<code><TD valign=middle></code>	Centrado vertical del contenido de una celda
<code><TD valign=top></code>	Alineación hacia el bajo del contenido de la celda
<code><TD colspan=x></code>	Numero de celdas para fusionar horizontalmente
<code><TD rowspan=x></code>	Numero de celdas para fusionar verticalmente

Frames

<code><FRAMESET>...</FRAMESET></code>	Define una estructura de frames (reemplaza la etiqueta BODY)
<code><FRAMESET rows="x%,y%,..."></code>	División horizontal de la ventana en %
<code><FRAMESET cols="x%,y%,..."></code>	División vertical de la ventana en %
<code><FRAME src="fichier.htm"></code>	Fichero mostrado en una ventana de frames
<code><NOFRAMES>...</NOFRAMES></code>	Contenido para los browser no previstos para los frames

Fichero Html

<code><HTML>...</HTML></code>	Principio y fin del fichero Html
<code><HEAD>...</HEAD></code>	Zona de encabezamiento de un fichero Html
<code><TITLE>...</TITLE></code>	Título visualizado por el browser (elemento de HEAD)
<code><BODY>...</BODY></code>	Principio y fin del cuerpo del fichero Html
<code><BODY bgcolor="#XXXXXX"></code>	Color del fondo (en hexadecimal)
<code><BODY background="xyz.gif"></code>	Imagen del fondo

