

2. Introducción a HTML

LM&SGI

Desarrollo de Aplicaciones Web

| | | |
|----|---|----|
| 1. | Restricciones XHTML | 2 |
| 2. | Estructura básica de un documento | 3 |
| 3. | Formato básico | 4 |
| 4. | Listas | 5 |
| 5. | Enlaces | 7 |
| 6. | Imágenes | 12 |

1. Restricciones XHTML

El lenguaje HTML original era muy permisivo en su sintaxis, por lo que era posible escribir sus etiquetas y atributos de muchas formas diferentes. Las etiquetas por ejemplo podían escribirse en mayúsculas, en minúsculas e incluso combinando mayúsculas y minúsculas. El valor de los atributos de las etiquetas se podían indicar con y sin comillas ("). Además, el orden en el que se abrían y cerraban las etiquetas no era importante.

La flexibilidad de HTML puede parecer un aspecto positivo, pero el resultado final son páginas con un código HTML desordenado, difícil de mantener y muy poco profesional. Afortunadamente, XHTML soluciona estos problemas añadiendo ciertas normas en la forma de escribir las etiquetas y atributos.

A continuación se muestran las cinco restricciones básicas que introduce XHTML respecto a HTML en la sintaxis de sus etiquetas:

■ Las etiquetas se tienen que cerrar de acuerdo a como se abren:

Ejemplo correcto en XHTML:

```
<p>Este es un párrafo con <a>un enlace</a></p>
```

Ejemplo incorrecto en XHTML (pero correcto en HTML):

```
<p>Este es un párrafo con <a>un enlace</p></a>
```

■ Los nombres de las etiquetas y atributos siempre se escriben en minúsculas:

Ejemplo correcto en XHTML:

```
<p>Este es un párrafo con <a href="http://www.google.com">un enlace</a></p>
```

Ejemplo incorrecto en XHTML (pero correcto en HTML):

```
<P>Este es un párrafo con <A HREF="http://www.google.com">un  
enlace</A></P>
```

■ El valor de los atributos siempre se encierra con comillas:

Ejemplo correcto en XHTML:

```
<p>Este es un párrafo con <a href="http://www.google.com">un enlace</a></p>
```

Ejemplo incorrecto en XHTML (pero correcto en HTML):

```
<p>Este es un párrafo con <a href=http://www.google.com>un enlace</a></p>
```

■ Los atributos no se pueden comprimir:

Ejemplo correcto en XHTML:

```
<dl compact="compact">...</dl>
```

Ejemplo incorrecto en XHTML (pero correcto en HTML):

```
<dl compact>...</dl>
```

■ Todas las etiquetas deben cerrarse siempre:

La mayoría de etiquetas HTML encierran un contenido de texto entre la etiqueta de apertura y la etiqueta de cierre. Sin embargo, algunas etiquetas especiales llamadas **etiquetas vacías** no necesitan encerrar ningún texto.

La etiqueta `
` por ejemplo, se utiliza para indicar el comienzo de una nueva línea, tal y como se verá más adelante. Por sus características, la etiqueta `
` nunca encierra ningún contenido de texto.

Como el estándar XHTML obliga a cerrar todas las etiquetas abiertas, siempre que se incluya la etiqueta `
` se debería cerrar de forma seguida: `
</br>`. Para que el código resulte más cómodo de escribir, XHTML permite en estos casos escribir de forma abreviada una etiqueta que se abre y se cierra de forma consecutiva.

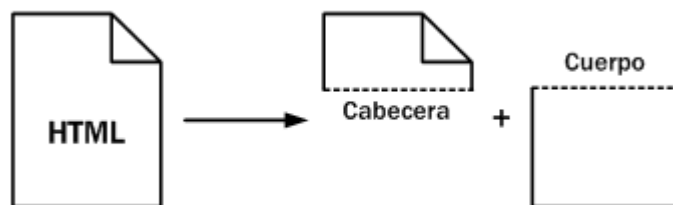
En lugar de abrir y cerrar de forma consecutiva la etiqueta (`
</br>`) se puede utilizar la sintaxis `
` para indicar que es una etiqueta vacía que se abre y se cierra en ese mismo punto. En la forma compacta es habitual equivocarse con la posición del carácter `/`.

Ejemplo correcto en XHTML: `
`

Ejemplo incorrecto en XHTML (pero correcto en HTML): `
`

2. Estructura básica de un documento

Las páginas HTML se dividen en dos partes: la cabecera y el cuerpo. La cabecera incluye información sobre la propia página, como por ejemplo su título y su idioma. El cuerpo de la página incluye todos sus contenidos, como párrafos de texto e imágenes.



A continuación vemos la estructura básica de un documento HTML.

```

<html>
  <head>
    <title>El primer documento HTML</title>
  </head>
  <body>
    Este es el cuerpo
  </body>
</html>
  
```

3. Formato básico

■ Etiquetas básicas

| | |
|-----------|---|
| body | Delimita el inicio y fin del cuerpo de la página. Como atributos interesantes tiene: <ul style="list-style-type: none"> ■ bgcolor. Permite dar un color de fondo uniforme a la página. ■ background. El fondo es una imagen. |
| p | Delimita los límites de un párrafo. |
| h1,...,h6 | Indican que el texto es un título. El número indica el nivel, siendo el 1 el mayor. Además a mayor nivel mayor es la letra. |
| hr | Dibuja una línea horizontal tan grande como la pantalla |
| br | Crea un salto de línea |
| b | Delimita una zona de texto en negrita. |
| strong | Define que el texto es el más importante de la página. Se resalta con texto en negrita. |
| em | Define que el texto es el más importante del párrafo. Se resalta con texto en cursiva. |
| i | Delimita una zona de texto en cursiva. |
| sup | Texto en formato superíndice |
| sub | Texto en formato subíndice |
| strike | Tacha el texto. (No soportado en HTML 5) |
| del | Indica que el texto está marcado para su eliminación. Se muestra con texto tachado. |
| ins | Se emplea para marcar una modificación en los contenidos originales consistente en la inserción de un nuevo contenido. Se muestra como texto subrayado. |
| u | Subraya un texto. (En HTML 4.01 fue desaprobado) |
| font | Permite especificar el tamaño de la fuente, el color y la familia, mediante los siguientes atributos: <ul style="list-style-type: none"> ■ Color. Indica mediante rgb(x,x,x) o #xxxxxx ■ Face. Nombre de la letra ■ Size. Tamaño de la letra |
| center | El texto se muestra centrado en la línea |
| pre | Muestra el texto tal y como se ha escrito, respetando todos los espacios en blanco y todas las nuevas líneas. |

■ Entidades HTML

Ya se habló en el tema 1 del problema del formato en cuanto a lenguas diferentes del inglés y se explico que la solución es usar entidades HTML, a continuación se muestran las entidades HTML más usuales.

| Carácter | Entidad con nombre | Entidad numérica | | Carácter | Entidad con nombre | Entidad numérica |
|----------|--------------------|------------------|--|----------|--------------------|------------------|
| á | á | á | | Á | Á | Á |
| é | é | é | | É | É | É |
| í | í | í | | Í | Í | Í |
| ó | ó | ó | | Ó | Ó | Ó |
| ú | ú | ú | | Ú | Ú | Ú |
| ü | ü | ü | | Ü | Ü | Ü |
| ñ | ñ | ñ | | Ñ | Ñ | Ñ |
| | | | | | | |
| ¿ | ¿ | ¿ | | α | α | α |
| ¡ | ¡ | ¡ | | β | β | β |
| – | – | – | | © | © | © |
| → | → | → | | ® | ® | ® |
| ← | ← | ← | | € | € | € |
| | | | | | | |
| < | < | < | | espacio | | |
| > | > | > | | & | & | & |

4. Listas

Las listas permiten crear párrafos agrupados y alineados mediante símbolos como viñetas o números y además crean párrafos alineados de forma especial para su correcta visibilidad.

■ Con viñetas

Las listas con viñetas se deben englobar dentro de un elemento ul (de unordered list, lista no ordenada), después cada párrafo de la lista estará dentro de elementos de tipo li (de list item, elemento de lista).

```
<ul>
  <li>Agua</li>
  <li>Vino</li>
  <li>Cerveza</li>
</ul>
```

■ Numéricas

Las listas numéricas aparecen dentro del elemento ol. La diferencia ahora es que cada párrafo con li, aparece con un número y no con una viñeta.

```
<ol>
  <li>Agua</li>
  <li>Vino</li>
  <li>Cerveza</li>
</ol>
```

■ Listas de términos

Permite crear una lista de definiciones de términos. En ellas se indica el término a definir y su definición.

```
<dl>
  <dt>Windows</dt>
  <dd>
    Sistema operativo de <strong>Microsoft</strong> disponible para PC
    disponible en versiones de 32 y 64 bits y para servidores,
    ordenadores e incluso tabletas y móviles.<br />
    La última versión es la 8 y la 2012 para servidores.
  </dd>
  <dt>Linux</dt>
  <dd>
    Sistema operativo de código abierto disponible
    en numerosas distribuciones gratuitas y de pago.
    Es la base del sistema <strong>Android</strong>.
  </dd>
  <dt>Mac OS</dt>
  <dd>
    Sistema operativo de los ordenadores de la empresa <strong>Apple</strong>
  <br />
    La última versión es la <strong>Snow Lion</strong>
  </dd>
</dl>
```

■ Listas anidadas

```
<ul>
  <li>
    No alcohólicas
    <ul>
      <li>Agua</li>
    </ul>
  </li>
  <li>
    Alcohólicas
    <ul>
      <li>Vino</li>
      <li>Cerveza</li>
    </ul>
  </li>
</ul>
```

```
<ol>
  <li>
    No alcohólicas
    <ul>
      <li>Agua</li>
    </ul>
  </li>
  <li>
    Alcohólicas
    <ul>
      <li>Vino</li>
      <li>Cerveza</li>
    </ul>
  </li>
</ol>
```


5. Enlaces

El lenguaje de marcado HTML se definió teniendo en cuenta algunas de las características que existían en ese momento para la publicación digital de contenidos. Entre los conceptos utilizados en su creación, se encuentra el mecanismo de "hipertexto".

De hecho, las letras "HT" de la sigla HTML significan "hipertexto" (hypertext en inglés), por lo que el significado completo de HTML podría traducirse como "lenguaje de marcado para hipertexto".

La incorporación del hipertexto fue una de las claves del éxito del lenguaje HTML, ya que permitió crear documentos interactivos que proporcionan información adicional cuando se solicita.

El elemento principal del hipertexto es el "hiperenlace", también llamado "enlace web" o simplemente "enlace". Los enlaces se utilizan para establecer relaciones entre dos recursos. Aunque la mayoría de enlaces relacionan páginas web, también es posible enlazar otros recursos como imágenes, documentos y archivos.

Una característica que no se suele tener en cuenta en los enlaces es que están formados por dos extremos y un sentido. En otras palabras, el enlace comienza en un recurso y apunta hacia otro recurso. Cada uno de los dos extremos se llaman "anchors" en inglés, que se puede traducir literalmente como "anclas".

■ URL

Una URL (Uniform Resource Location) es la dirección concreta de un recurso (una página web, una imagen, un vídeo, un directorio,...) en Internet.

La gracia es que cada URL es única con lo que es una especie de DNI de un recurso. Si la URL es correcta sólo habrá un solo recurso posible al que se puede referir dicha.

Una URL se forma usando:

► **El protocolo.** Ejemplos:

- [http://](#) (para recursos de la web)
- [https://](#) (para recursos de la web contenidos en un servidor seguro)
- [ftp://](#) (recursos contenidos en un servidor de ficheros)
- [ftps://](#) (recursos contenidos en un servidor de ficheros seguro)
- [mailto:](#) (dirección de correo electrónico)
- [file:///](#) (recurso dentro de nuestra propia computadora)

► **Servidor.** Nombre completo en Internet del servidor que aloja el recurso al que deseamos acceder. Por ejemplo: [www.nasa.gov](#).

► **Puerto.** Puerto por el que se debe conectar con el servidor para obtener el recurso. Si no se indica (que es lo habitual) se toma el puerto por defecto. Por

ejemplo en http se usa el 80. Si queremos usar uno en particular se indica tras el servidor poniendo dos puntos y el puerto. Por ejemplo `www.midb.com:1521`

► **Ruta.** Indica el recorrido dentro del servidor que hay que hacer en sus directorios para llegar al recurso que queremos. Se pone después del servidor.

Ejemplos:

- `/index.html` Accede a la página `index.html` situada en la raíz del servidor.
- `/imagenes/paisajes/foto001.jpg` Accede a la imagen `foto001.jpg` dentro del directorio `paisajes` dentro, a su vez, del directorio `paisajes`

► **Cadena de búsqueda.** Sólo aparece tras direcciones a páginas web que admitan recibir parámetros (como las páginas PHP, ASP o JSP por ejemplo).

Ejemplo `?pagina=5&idioma=es`, pasará los parámetros `pagina` y `lenguaje` usando los valores 5 y es respectivamente.

Ejemplo de URL completa:

`www.ejemplourl.com:9000/dirs/srv/pagina1.php?a=90&r=7`

Por otro lado los caracteres presentes en una URL deben de pertenecer al ASCII, de otro modo tendremos el sempiterno problema de que no se lea bien la URL por diferentes tipos de codificación del texto. De hecho los caracteres permitidos en una URL son:

```

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
a b c d e f g h i j k l m n o p q r s t u v w x y z
0 1 2 3 4 5 6 7 8 9 0 - _ ~ .

```

Cualquier otro no es válido aunque sí aparecen los siguientes pero sólo para indicar elementos con significado especial (recorridos de la ruta, elementos de la parte de consulta de la URL,...). Estos símbolos están reservados para un uso concreto y no se pueden utilizar para ningún otro:

```
! # $ % & ' ( ) * + , / : ; = ? @ [ ]
```

Si necesitamos codificar en la URL símbolos fuera de los caracteres permitidos entonces debemos codificarlos usando una notación que comienza con el símbolo % y le sigue el código hexadecimal correspondiente en el código ASCII.

■ Etiqueta HTML

| | |
|--------|--|
| a | Indica el enlace |
| href | Atributo para definir la URL a la que accederemos al pinchar en el enlace |
| name | Nombra al enlace. Nos permite ir directamente a este enlace desde otros enlaces. |
| target | Permite indicar cómo se muestra la página de destino. Con la caída en desuso de los marcos el único valor útil es: <code>_blank.</code> , que provoca la apertura del enlace en una nueva página |

Un enlace HTML se compone básicamente de dos partes:

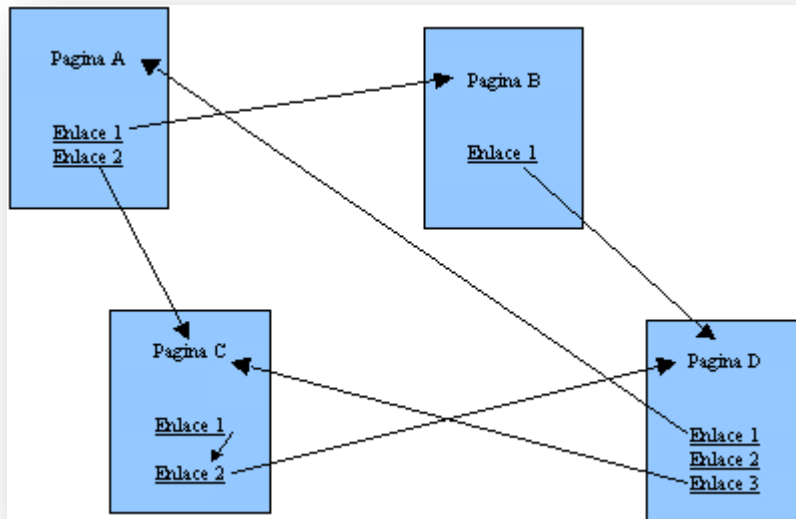
- Un texto para pinchar.
- Un URL a la que ir.

Ejemplo: ` vamos a google`

■ Estructura de una sitio web

Al incorporar los enlaces a los documentos HTML estamos dando la posibilidad a un sitio web de estar compuesto por varios documentos enlazados. Es decir, cada documento HTML se puede considerar una hoja de papel y cada enlace una indicación de donde están las otras hojas.

Por otra parte, siempre que se accede a un sitio web existe una página inicial, normalmente denominada **home**, que hace las funciones de portada de nuestro sitio. Desde esta podemos acceder por medio de los enlaces al resto de “hojas” de nuestro sitio. Un sitio web, por pequeño que sea, estará formado por decenas de páginas que enlazadas entre sí, creando una red de interconexión como se esquematiza en la imagen:



■ Enlaces relativos y absolutos

Observemos las siguientes URLs:

<http://www.ejemplo.com/ruta1/ruta2/pagina1.html>.

<http://www.ejemplo.com/ruta1/ruta2/pagina2.html>.

<http://www.ejemplo.com/ruta1/ruta2/pagina3.html>.

Se puede observar que toda la primera parte es igual en todas las URLs. Supongamos que las dos segundas URL aparecen en el código HTML de la primera, entonces podemos escribir las URLs de la siguiente forma.

[pagina2.html](#).

[pagina3.html](#).

Si en el navegador estamos viendo la `pagina1.html` y se pinchamos en un enlace como los anteriores, el navegador deduce que el resto de la URL que falta es:

<http://www.ejemplo.com/ruta1/ruta2/>

que es donde está `pagina1.html`.

El primer tipo de URL se denominan absolutas y el segundo relativas.

Se pueden dar varias situaciones:

- El origen y el destino del enlace se encuentran en el mismo directorio

| | |
|------------------|--|
| URL absoluta | <code>http://www.ejemplo.com/ruta1/ruta2/ruta3/pagina2.html</code> |
| URL relativa | <code>pagina2.html</code> |
| Origen | <code>http://www.ejemplo.com/ruta1/ruta2/ruta3/pagina1.html</code> |
| Recurso enlazado | Página web llamada <code>pagina2.html</code> y que se encuentra en el mismo directorio |

- El destino del enlace se encuentra cerca de su origen y en un nivel superior

| | |
|------------------|--|
| Origen | <code>http://www.ejemplo.com/ruta1/ruta2/ruta3/pagina1.html</code> |
| Recurso enlazado | Página web llamada <code>pagina2.html</code> y que se encuentra en el directorio superior llamado <code>ruta2</code> |
| URL absoluta | <code>http://www.ejemplo.com/ruta1/ruta2/pagina2.html</code> |
| URL relativa | <code>../pagina2.html</code> |

- El destino del enlace se encuentra cerca de su origen y en un nivel inferior

| | |
|------------------|--|
| Origen | <code>http://www.ejemplo.com/ruta1/ruta2/ruta3/pagina1.html</code> |
| Recurso enlazado | Página web llamada <code>pagina2.html</code> y que se encuentra en un directorio inferior llamado <code>ruta4</code> |
| URL absoluta | <code>http://www.ejemplo.com/ruta1/ruta2/ruta3/ruta4/pagina2.html</code> |
| URL relativa | <code>ruta4/pagina2.html</code> |

■ Enlaces internos

El atributo **name** permite definir enlaces dentro de una misma página web. Si una página es muy larga, puede ser útil mostrar enlaces de tipo "Saltar hasta la segunda sección", "Volver al principio de la página", etc.

Este tipo de enlaces son especiales, ya que la URL de la página siempre es la misma para todas las secciones y por tanto, debe añadirse otra parte a las URL para identificar cada sección.

```

<a name="primera_seccion"></a>
<p>
    Tenga cuidado. A la gente le gusta que les
    digan lo que ya saben. Recuerde eso. Se ponen incómodos cuando uno
    les cuenta cosas nuevas. Las cosas nuevas... bueno,
    las cosas nuevas no son lo que se esperan
</p>
...
<a name="segunda_seccion"></a>
<p>
    ...William, pero yo siempre he pensado que
    debemos avanzar hacia el futuro. -Sí, señor.
    Es bastante difícil avanzar hacia otro sitio
</p>
...

```

El atributo name permite crear "enlaces vacíos" que hacen referencia a secciones dentro de una misma página. Una vez definidos los "enlaces vacíos", es posible crear un enlace que apunte directamente a una sección concreta de una página:

```

<a href="http://www.ejemplo.com/pagina1.html#segunda_seccion">
    Enlace a la sección 2 de la página 1
</a>

```

La sintaxis que se utiliza con estos enlaces es la misma que con los enlaces normales, salvo que se añade el símbolo # seguido del nombre de la sección a la que se apunta. Cuando el usuario pincha sobre uno de estos enlaces, el navegador accede a la página apuntada por la URL y baja directamente a la sección cuyo nombre se indica después del símbolo #.

También es posible utilizar este tipo de enlaces con URL relativas en una misma página. El siguiente ejemplo añade enlaces de tipo "Volver al inicio de la página":

```

<a name="inicio"></a>

```

Los enlaces directos a secciones también funcionan con el atributo id de cualquier elemento. De hecho, se recomienda utilizar los atributos id de los elementos ya existentes en la página en vez de crear "enlaces vacíos".

El siguiente ejemplo es equivalente al ejemplo anterior:

```

<h1 id="inicio">Título de la página</h1>

```

■ Enlaces a e-mail

En muchas webs el correo electrónico de administración del sitio se resalta como un enlace y al pinchar en él se lanza el gestor de correos del cliente, por ejemplo el Outlook. Este efecto es posible conseguirlo simplemente indicado en el href la dirección e-mail, precedido de **mailto:**.

```

<A HREF="mailto:sgarrid2@airtel.net">Sergio Garrido-Lestache </A>

```

6. Imágenes

| | |
|--------|--|
| img | Imagen |
| src | URL del archivo de imagen |
| alt | Descripción de la imagen |
| width | Anchura de la imagen |
| height | Altura de la imagen |
| align | Alineación de la imagen de acuerdo a los elementos de alrededor. Puede tomar los siguientes valores: <i>top</i> , <i>bottom</i> , <i>middle</i> , <i>left</i> y <i>right</i> . |
| hspace | Especifica los espacios en blanco a derecha e izquierda de la imagen. |
| vspace | Especifica los espacios en blanco arriba y debajo de la imagen. |
| border | Especifica la anchura en píxeles del borde de la imagen |

Ejemplo:

```

```

En el ejemplo anterior vemos las dos formas de indicar los tamaños. Si solo ponemos un número estamos indicando el tamaño absoluto en píxeles. Si ponemos un porcentaje estamos indicando el tamaño relativo en función del elemento que contiene la imagen.