1. **HTML INTRODUCCIÓN**

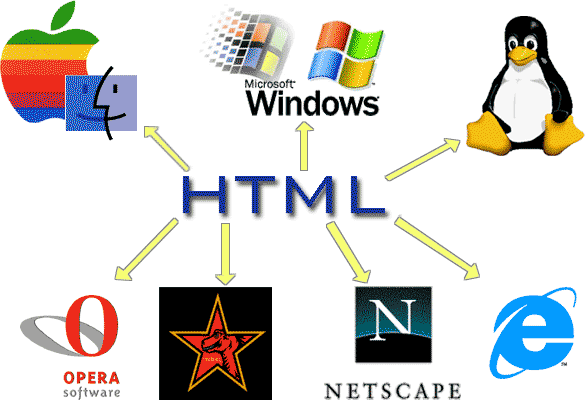
**¿Qué es y para qué sirve?**

HTML (Hypertext Markup Language) es un lenguaje de presentación de documentos. Haciendo uso de este lenguaje, a través de una serie de etiquetas podremos definir el contenido de un documento de texto con algunos elementos multimedia. Este tipo de documento escrito en HTML es lo que llamamos página Web.

HTML no es un lenguaje de programación, por lo que no es posible hacer un programa con este lenguaje. Para añadir alguna capacidad de proceso a una página Web es necesario recurrir a otros lenguajes, como Javascript o VBSscript.

Se debe considerar HTML como un lenguaje para estructurar documentos y hacer su contenido más accesible, pero resulta frustrante observar la misma página en otro navegador, ya que probablemente no se visualice correctamente y resulte una página poco estética.

HTML es estandarizado y multiplataforma. Un documento preparado utilizando marcas HTML puede ser leído utilizando una variedad de navegadores Web, como Firefox, Opera, Netscape, o Explorer Chrome. Por ello, no conviene abusar de las capacidades específicas de un único programa de visualización. Es mejor utilizar una versión estándar del lenguaje, y desarrollar las páginas probándolas con más de un navegador.



El navegador interpreta las marcas HTML en un archivo y presenta la información formateada como una pagina Web.

HTML es un formato basado en texto, diseñado para ser leído y editado directamente usando un **editor de texto**. No obstante, escribir y mantener un gran número de páginas a mano requiere esfuerzo y conocimiento en profundidad de HTML. Para evitar este problema existen  **editores visuales** como Macromedia Dreamweaver, Microsoft FrontPage, etc., que permiten editar páginas Web de una forma similar a como se edita un documento de texto. Sin embargo, estos editores pueden llegar a generar código HTML de muy baja calidad sobre el que tarde o temprano habrá que realizar una operación de limpieza y reestructuración, utilizando un editor de texto.

La **extensión de los ficheros** que contienen código HTML **es  .html**. Sin embargo, debido a las limitaciones del sistema de ficheros del sistema operativo DOS de Windows, la extensión **.htm también es** aceptada como **válida**.

Asimismo, el código HTML se puede generar de forma automática desde el servidor como respuesta a la ejecución de un script de servidor (escrito en Perl, PHP, ASP, JSP, etc.).

El lenguaje de marcas **HTML**(*Hyper Text Markup Language*) surgió por la complejidad del lenguaje SGML creándose un lenguaje mucho más simple y adaptado expresamente al cometido de representar contenido para la web. En este sentido, el número de etiquetas del que se dotó a HTML era considerablemente reducido. HTML fue por primera vez plasmado como HTML 2.0 en un RFC en 1995 y este RFC fue dando lugar a sucesivos RFC hasta la aparición en 1997 de un RFC que propició HTML 3.2 y que fue publicado como recomendación por parte del W3C.Poco después, en 1998, W3C publicó una nueva versión denominada HTML 4.0 en tres versiones distintas (*strict*, *transitional*y *framese*) y, enseguida, se publicó la versión 4.1 para incorporar algunas erratas. En 2008 el consorcio W3C publicó un borrador de HTML5 en el cual se encuentran trabajando hoy en día, aunque varias empresas las están usando de modo experimental. El consorcio internacional W3C marcó las siguientes fechas para liberación de los estándares de especificación: 2014 para HTML 5.0, 2016 para HTML 5.1 y 2018 para HTML 5.2

* Las instrucciones HTML están encerradas entre los caracteres: < y >.Muchos elementos HTML requieren una marca de comienzo y otra de finalización. Todo aquello que está fuera de las marcas del lenguaje se imprime en la pantalla (dentro del navegador).
* Una página HTML es un archivo que generalmente tiene como extensión los caracteres html. Por ejemplo podemos llamar a nuestra primer página con el nombre: pagina1.html

<html>

<head>

</head>

<body>

Cuerpo de la página.

</body>

</html>

**Definición de Tipos de Documentos (DTDs)**

**(Veremos este tema ampliado más adelante)**

Para que los navegadores sepan que versión de HTML es el documento que debe presentar, es necesario incluir una cabecera DOCTYPE que lo identifique. A continuación, se muestra un ejemplo para HTML 4.01 Transitional.

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">.

La primera línea de un documento HTML debe especificar la versión en la que está desarrollado. Esto se consigue insertando una referencia a un documento donde se especifica formalmente la sintaxis y estructura de todos los elementos que se pueden utilizar en esa versión de HTML.

Este documento se llama DTD (Data Type Definition, en castellano *Definición de Tipos de Documento).*

La DTD **indica qué elementos, atributos, etcétera, tiene un documento y cómo se anidan**. En función de la DTD que se defina para ese documento, se permitirán o se prohibirán el uso de ciertos elementos.

**- Tipos de DTD en HTML 4.01:**

* **EL HTML 4.01 Strict** o tipo de documento HTML 4.0 estricto, en el que prima la estructura del mismo sobre su presentación, es decir, que el documento debe tener estar conformado de acuerdo con las reglas del estándar HTML 4.0 de una forma estricta, sin desviarse de él en lo más mínimo.

No se permite en el documento la presencia de elementos o atributos desaprobados por el estándar HTML 4.0 (etiquetas IMG sin el atributo ALT,etc.). Declaración de estilos mediante Hojas de Estilos en Cascada (CSS) y una perfecta definición de todos los atributos establecidos como necesarios para cada etiqueta. Como ventaja presenta el que con un documento de esta forma tenemos garantizada la accesibilidad y compatibilidad de nuestra página, y como defectos el que navegadores antiguos no lo soportan y que la construcción del documento no es flexible en absoluto (no se admiten errores en el código).

Su definición es del tipo:

**<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">**

* **EL HTML 4.01 Transational** Las DTD transicionales (Transitional DTD) permiten algunos elementos y atributos antiguos que están en desuso y elementos desaprobados por el Consorcio para este estándar. Con él damos apoyo a navegadores que soportan mal las Hojas de Estilos en Cascadas (CSS) y disponemos de más flexibilidad a la hora de escribir el código de nuestra página, a costa de no entrar de lleno en el estándar y de perder un poco de accesibilidad.

Su definición es del tipo:

**<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">**

* **HTML 4.01 Frameset** que es una variación del HTML 4.01 Transitional para aquellos documentos que empleen frames (marcos) puedan obtener resultados válidos desde el validador SGML Recordar que en estos documenntos HTML, no existe <body>…</body> siendo sustituido <frameset>.. . . .</frameset>

Su definición es del tipo:

**<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">**

<!DOCTYPE HTML PUBLIC "**FPI**" "**URI\_DTD**">

Donde:

* FPI es el (*Format Public Identifier o Identificador Público de Formato*)
* URI\_DTD (URI de la DTD que define formalmente el FPI seleccionado) varían en función de las necesidades del documento.

Los posibles juegos de valores para HTML 4.01 son:

|  |  |  |
| --- | --- | --- |
| **FPI** | **URI\_DTD** | **Tipo de Documento** |
| **"-//W3C//DTD HTML 4.01//EN"** | **"http://www.w3.org/TR/html4/strict.dtd"** | HTML estricto |
| **"-//W3C//DTD HTML 4.01 Transitional//EN"** | **"http://www.w3.org/TR/1999/REC-html401-19991224/loose.dtd"** | HTML transicional |
| **"-//W3C//DTD HTML 4.01 Frameset//EN"** | **"http://www.w3.org/TR/1999/REC-html401-19991224/frameset.dtd"** | HTML con marcos (frames) |

**Versión de HTML utilizada**

A lo largo del desarrollo de este curso se utilizará el estándar de HTML 4.01 que es compatible con los navegadores Internet Explorer 6.0, Firefox 1.5 y Netscape 7.0 (programas que se utilizarán para visualizar los documentos realizados).

Podemos encontrar el estándar "HTML 4.01 Specification" en <http://www.w3.org/TR/html401/> (en inglés) y también en <http://www.w3.org/MarkUp/html4-updates/translations#Spanish>(en castellano).

En esta especificación nos encontraremos que algunos elementos son considerados como "**deprecated**" o "**desaprobados**", teniendo una alternativa en las Hojas de Estilo CSS que estudiaremos en otro tema posterior. Estas hojas de estilo nos ayudarán a realizar un código más limpio, profesional y reutilizable.

**Estructura básica de un documento**

**Los elementos básicos que toda página HTML debe llevar:**

* La marca:<html> y finaliza con la marca:</html>
* Los fines de marcas tienen el mismo nombre que el comienzo de marca, más el caracter /
* Una página HTML tiene dos secciones muy bien definidas que son:

**Cabecera:**

**<head>** (Etiqueta de apertura del Título)

Contiene información sobre el título de la página, el autor, palabras clave, etc. que no se presentarán en la ventana del navegador, salvo el título que aparecerá en la barra de título de la parte superior

**</head>** (Etiqueta de cierre del Título)

**Cuerpo:**

**<body>** (Etiqueta de apertura del cuerpo)

Aquí va el contenido de la página que será lo que se presente en pantalla.

**</body>** (Etiqueta de cierre del cuerpo)



* Todo el texto que dispongamos dentro del <body> aparece dentro del navegador tal como lo hayamos escrito.
* El lenguaje HTML no es sensible a mayúsculas y minúsculas, es decir podemos escribirlo como más nos guste, además no requiere que dispongamos cada marca en una línea (podríamos inclusive escribir toda la página en una sola línea! cosa que no conviene ya que somos nosotros quienes tendremos que modificarla en algún momento).

El lenguaje HTML consta de una serie de *etiquetas* o *tags* que definen la estructura del texto de un documento. Una etiqueta es una palabra clave encerrada entre ángulos, por ejemplo:

<HTML>

Existen dos clases de etiquetas:

* **Etiquetas vacías**

Se utilizan para introducir saltos de línea, líneas horizontales y otros elementos no asociados al formato de textos o imágenes.

Su formato es:

<etiqueta>

* **Etiquetas contenedoras**

Constan de una marca inicial y una marca final igual que la inicial pero precedida del carácter “/”. Todo lo que esté incluido entre ambas marcas quedará sujeto al formato indicado por la etiqueta.

La mayoría de estas etiquetas pueden contener a su vez otras etiquetas de cualquiera de los dos tipos.

Su formato es:

<etiqueta>texto</etiqueta>

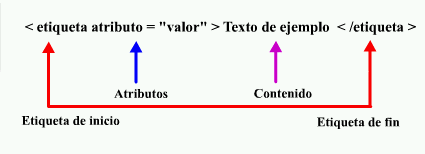
Las etiquetas pueden tener atributos.

Un atributo es un valor que define una propiedad especifica de la etiqueta, que modifica su significado y debe ir entre comillas (aunque su uso no es obligatorio sí es muy aconsejable).

Los atributos se escriben dentro de los ángulos que delimitan una etiqueta, después del nombre de la misma. El navegador dará a esa etiqueta las modificaciones en función de los valores de esos atributos y veremos como resultado que el contenido de las marcas, el elemento que encontramos entre las marcas de inicio y fin, se ve afectado por esos atributos.

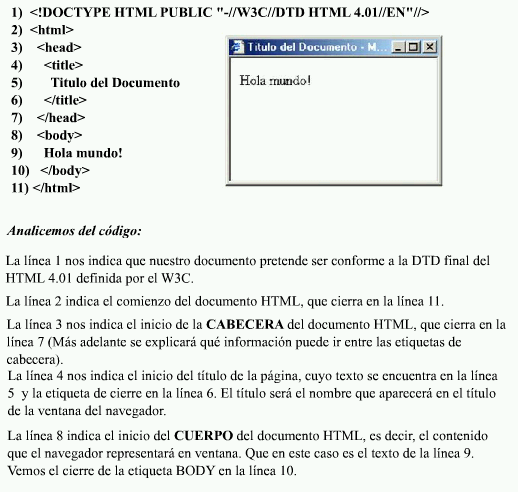
Cada atributo tiene un identificador, seguido de un signo "=" y el valor que se le quiere asignar entre comillas simples (' ') o comillas dobles (" ").

La mayoría de las etiquetas tienen un inicio y un cierre y afectan a la parte del contenido que se encuentra entre medias. El componente de inicio es el nombre de etiqueta y sus atributos, si los hay. La etiqueta de cierre es sólo el nombre de la etiqueta precedido de una barra (/).

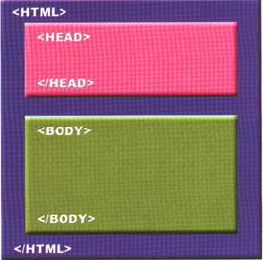


El resto de contenido de un documento HTML suele ser básicamente texto de contenido.

Veamos el código de una página Web simple:



Cualquier página HTML tiene el esqueleto anterior.



Entre las etiquetas <body> y </body> se incluye el contenido del documento, que estará formado, básicamente por texto, imágenes, enlaces y formularios, etc.

**Contenido de una página HTML**

Prácticamente todo lo que no son etiquetas se considera contenido del documento. Se debe tener en cuenta que todas las etiquetas que se insertan en un documento para formatear el contenido del mismo son sólo un consejo para el navegador. Éste puede ignorarlas o interpretarlas de una forma distinta a la que has estado viendo en el navegador que ha servido para probar tus páginas durante el desarrollo. De hecho, cada persona puede configurar su navegador para que interprete cada etiqueta de la forma que le apetezca.

En este apartado vamos a comenzar a introducir elementos dentro de un documento HTML. Veremos que existen distintos tipos, como son cabeceras de distintos tamaños, párrafos, elementos para destacar textos con negritas, subrayados o itálicas... También comenzaremos a dotar de color a los elementos, tanto a textos como al fondo de un documento HTML.

***Caracteres, entidades y caracteres 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:

* **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: **&reg;**

* **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: **&#174;**

* **Reservadas**

Se usan exclusivamente para caracteres usados en etiquetas pero que nosotros no los pretendemos usar dentro de ninguna de ellas.

Un ejemplo de estos tipos de caracteres especiales son:



El conjunto de caracteres ASCII no es suficiente para un sistema de información global como la Web, de modo que HTML usa un conjunto de caracteres mucho más completo llamado *Conjunto Universal de Caracteres (Universal Character Set, UCS)*, definido en ISO10646. Este estándar define un repertorio de miles de caracteres usados por comunidades de todo el mundo.

El conjunto de caracteres del documento no es suficiente para permitir a los navegadores interpretar correctamente los documentos HTML del modo en que éstos se intercambian normalmente: codificados como una secuencia de bytes en un fichero. Los navegadores deben conocer además la codificación de caracteres específica usada para transformar el flujo de caracteres del documento en un flujo de bytes.

Para utilizar caracteres especiales como por ejemplo: < y >. No los podemos usar directamente ya que el navegador los confundiría con los caracteres que se utilizan para las marcas HTML.

La solución es utilizar otra sintaxis para dichos caracteres, los más utilizados:

< &lt;

> &gt;

& &amp;

" &quot;

  &nbsp; //Espacio en blanco.

© &copy;

€ &euro;

la sintaxis es utilizar un ampersand (&) seguido de un nombre significativo y un punto y coma.

**Ejemplo: 10+x\*y < 12\*z**

pagina1.html

<html>

<head>

<title>Título de la primer página</title>

</head>

<body>

<h1>

10+x\*y &lt; 12\*z

</h1>

</body>

</html>

***Comentarios <!-->***

En el código HTML se pueden incluir comentarios utilizando los símbolos:

* “**<!--**“ para iniciar la zona de comentarios
* “**-->**” para acabar la zona de comentarios

Ejemplo:

<!-- Esto es un comentario -->

Un comentario es un texto que solo le interesa a la persona que desarrolló la página, el navegador ignora todo el contenido que se encuentra dentro de esta marca. Los comentarios son muy útiles para el desarrollador de la página. Uno deja anotaciones para facilitar el mantenimiento del sitio. Además hay que tener en cuenta que puede ser otra persona la que desarrolle en otro momento el mantenimiento de las páginas que desarrollamos nosotros. Lo que para uno puede ser muy obvio a otro desarrollador puede no ser tan obvio. Otro uso muy habitual cuando estamos desarrollando la página si queremos deshabilitar una parte del código podemos encerrarla entre los caracteres de comentarios.

**La sintaxis para definir un comentario es:**

**<!-- Aquí va el comentario -->**

**Es obligatorio el carácter de menor < disponer el signo de admiración y los dos guiones seguidos. Cerramos el comentario con dos guiones y el signo de mayor >**

Ejemplo:

<html>

<head>

</head>

<body>

<!-- Corresponden a datos del año 2011. -->

<table border="1">

<tr>

<th rowspan="4">Recursos</th><th colspan="4">Facturación

de los últimos tres meses</th>

</tr>

<tr>

<td>Discos Duros</td><td>23000</td><td>27200</td><td>26000</td>

</tr>

<tr>

<td>CPU</td><td>73000</td><td>67300</td><td>51000</td>

</tr>

<tr>

<td>Monitores</td><td>53000</td><td>72000</td><td>88000</td>

</tr>

</table>

</body>

</html>

Un comentario puede abarcar varias líneas:

<!--

comentarios - comentarios - comentarios

comentarios - comentarios - comentarios

comentarios - comentarios - comentarios

comentarios - comentarios - comentarios

comentarios - comentarios - comentarios

comentarios - comentarios - comentarios

comentarios - comentarios - comentarios

comentarios - comentarios - comentarios

-->

De todos modos hay que tener en cuenta que cuando un navegador pide la página a un sitio el servidor envía el archivo HTML completo, es decir con los comentarios. Los comentarios consumen ancho de banda del servidor.

***Interpretación de espacios, tabulaciones y retornos de carro***

HTML fue inicialmente pensado para texto. Los saltos de línea del código de un documento HTML no se representan, el texto aparece a continuación separado por un espacio, aunque se trate de varios saltos de línea.

|  |  |
| --- | --- |
| <html>  <head>  <title>Párrafos,líneas y espacios</title>  </head>  <body>  **Esto**  **es**  **una**  **sola**  **línea**  **de**  **texto**  </body>  </html> |  |

La longitud de esa línea vendrá determinada por el ancho de la ventana.

Los espacios en blanco y tabulaciones tampoco se representan, tan solo uno de ellos aparecerá representado en el navegador.

|  |  |
| --- | --- |
| <html>  <head>  <title>Párrafos,líneasy espacios</title>  </head>  <body>  **Esto es otra**  **linea de texto**  <body>  </html> |  |

Para representar un salto de línea se utiliza la etiqueta **<BR>**.

|  |  |
| --- | --- |
| <html>  <head>  <title>Párrafos,líneas y espacios</title>  </head>  <body>  Esto es una línea  de texto.**<br>**  Esto es otra línea  de texto.  </body>  </html> |  |

Por cada **<BR>** el texto saltará una línea en blanco.

Para forzar un espacio en blanco se debe utilizar el carácter especial “**&nbsp;”**