

UD2.1 INTRODUCCIÓN AL HTML

ÍNDICE

1. INTRODUCCIÓN.....	2
1.1. ¿Qué es el HTML?	2
1.2. Origen del HTML.....	2
1.3. Fichero HTML	3
2. ELEMENTOS DEL HTML	5
2.1. Etiquetas	5
2.2. Elementos	6
2.3. Atributos	6
2.4. Jerarquía	7
2.5. Elementos vacíos	7
3. ANATOMÍA DEL HTML	8
3.1. Visión General	8
3.2. Declaración Inicial DOCTYPE.....	8
3.3. Etiqueta HTML.....	9
3.4. Etiqueta HEAD	9
3.5. Etiqueta BODY	9

1. INTRODUCCIÓN

1.1.¿Qué es el HTML?

HTML es el lenguaje utilizado para crear la mayor parte de las páginas web. Es un estándar reconocido en todos los navegadores, por lo tanto, todos ellos visualizan una página HTML de forma muy similar independientemente del sistema operativo sobre el que se ejecuta.

- HTML describe la estructura de las páginas web.
- Los elementos HTML son los componentes básicos del HTML.
- Los elementos HTML se representan mediante etiquetas.
- HTML define la manera en que las imágenes, multimedia y el texto se muestra en los navegadores web.
- El HTML incluye elementos para conectar vuestros documentos (hipertexto) y hacer vuestra web documentos interactivos (como con formularios).
- Los navegadores no muestran las etiquetas HTML, sino que las utilizan para representar el contenido.
- Con HTML podéis crear vuestro propio sitio web.

1.2.Origen del HTML

En **1992**, cuando había 50 servidores web en el mundo, el **CERN lanzó el navegador web** portátil como software gratuito. Marc Andreessen, que trabajaba en el Centro Nacional de Aplicaciones de Supercomputación (NCSA), creó un navegador llamado Mosaic que se lanzó en 1993. Poco después, dejó NCSA para fundar Netscape. La primera versión del navegador Netscape implementaba HTML 1.0.

En **1992**, Berners-Lee y el equipo del CERN publicaron **el primer borrador HTML 1.0**, que se finalizó en 1993. Esta especificación era tan sencilla, pero contenía la idea básica que se ha convertido en central en la reciente evolución de HTML.

En **1995**, el grupo de trabajo HTML del Internet **Engineering Task Force desarrolló HTML 2.0**. Este grupo se disolvió más tarde a favor del **World Wide Web Consortium (W3C)** que continúa desarrollando HTML e intentó hacer una estandarización en HTML 3.0.

Finalmente, en **1997** se publicó una **versión de consenso, HTML 3.2**. Esto añadió funciones como tablas, texto que fluye por imágenes, miniaplicaciones y hojas de estilo a la especificación oficial, manteniendo la compatibilidad con HTML 2.0.

El **W3C** publicó la especificación **HTML 4.0 a finales de 1997**, y siguió con **HTML 4.01** en 1999, que añadió **scripts, CSS, tablas complejas y mejores formularios**. Esta versión intentó corregir algunos de los errores más flagrantes que la 3.2 había permitido cometer los diseñadores, especialmente en la introducción de Hojas de estilo en cascada. Pero,

de hecho, el W3C ha abandonado el HTML como estándar por defecto a favor de un retorno hacia la raíz de SGML, un lenguaje más grande y complejo.

El W3C publicó en el año 2000 el XHTML 1.0 (Extensible HTML). El contenido de la especificación XHTML 1.0 era idéntico al del HTML 4.01. La única diferencia estaba en la sintaxis de la lengua. Mientras que HTML permitía a los autores mucha libertad en cómo escribían sus elementos y atributos, XHTML exigía a los autores que siguieran las reglas de XML (Extensible Markup Language), un lenguaje de marcas más estricto en el que el W3C se basaba la mayoría de sus tecnologías.

En el año 2004, W3C comenzó a trabajar en XHTML 1.1, pero **los representantes de Opera, Apple y Mozilla estaban descontentos con esta dirección porque el W3C no se prestó atención a las necesidades de los diseñadores web**. Querían que se ponía más énfasis en los formatos que permitieran la creación de aplicaciones web.

Así que crearon **su propia organización**, la Web Hypertext Application Technology Working Group (**WHATWG**). Comenzaron a trabajar en una única especificación llamada simplemente **HTML5**.

Mientras que HTML5 se estaba desarrollando en el WHATWG, el W3C continuó trabajando en XHTML 2. En octubre de 2006, Sir Tim Berners-Lee escribió una entrada en el blog en la que admitió que el intento de trasladar la web de HTML a XML no funcionaba.

En el año 2007 los dos grupos comenzaron a trabajar juntos y en enero de 2008 se publicó HTML5 como borrador de trabajo. Por otro lado, **en 2009 el W3C cerró XHTML 2**.

HTML5 ha crecido desde entonces hasta convertirse en extremadamente popular. El auge de los navegadores de teléfonos móviles también ayudó a su crecimiento. Apple (iOS) y Google (Android OS), los dos fabricantes de sistemas operativos de teléfonos móviles más grandes han incorporado capacidades HTML5 a sus navegadores.

En 2012, HTML5 se convertirá en el formato de publicación web recomendado tanto del WHATWG como del W3C. En 2014, lanzaron HTML5 como recomendación estable.

1.3. Fichero HTML

Básicamente, podemos simplificar y decir que una página web es un documento HTML. Este documento es un fichero de texto con extensión *.html*.

Aunque es una simplificación, es verdad que el fichero HTML es el estructurador principal de la página, y si hacen falta otros elementos (hojas de estilo, scripts) son dirigidas al HTML principal. Para empezar, podemos decir que el HTML luce como se representa a continuación:

```
<!DOCTYPE html>
<html>
  <head>
    <title>My First Webpage </title>
  </head>
  <body>
    <p>This is my first homepage. <b>This text is bold</b></p>
  </body>
</html>
```

2. ELEMENTOS DEL HTML

El HTML, como lenguaje de marcas está formado por etiquetas que forman elementos. Los elementos pueden tener atributos y otros elementos anidados.

Aunque HTML define una gran cantidad de etiquetas, estas no son suficientes para crear páginas complejas ya que la definición completa de ciertos elementos, como las imágenes y los enlaces, requiere información adicional. Como no es posible crear una etiqueta por cada elemento diferente, se afianza la información adicional a las etiquetas mediante los atributos dando lugar a los elementos.

Para cada uno de los atributos hay definido un conjunto de valores que se le puede asignar, si el valor de un atributo no es válido, el navegador le ignora.

Cada una de las etiquetas HTML define los atributos que puede utilizar, aunque algunos de ellos son comunes a muchas etiquetas.

La jerarquía es una parte fundamental del HTML y entenderla es el primer paso para estructurar de una forma extensa y elaborada los documentos web.

En los siguientes puntos se establecen los principales elementos de los documentos HTML.

2.1. Etiquetas

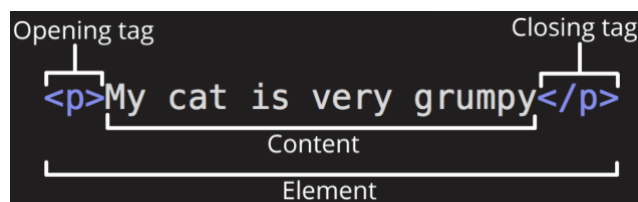
Las etiquetas son elementos que están rodeados por paréntesis angulares.

```
<etiqueta> CONTENIDO </etiqueta>
```

Algunos puntos importantes sobre las etiquetas son:

- Las etiquetas HTML suelen venir en pares como <p> y </p>.
- La primera etiqueta de un par es la etiqueta inicial, la segunda etiqueta es la etiqueta final
- La etiqueta final se escribe como la etiqueta inicial, pero con una barra inclinada insertada antes de nombre de la etiqueta
- El texto entre las etiquetas inicial y final es el contenido del elemento
- Las etiquetas HTML no distinguen entre mayúsculas y minúsculas, significa lo mismo que . Sin embargo el Mundo
- El Wide Web Consortium (W3C) recomienda etiquetas en minúsculas.

2.2.Elementos



Las partes principales de nuestro elemento son las siguientes:

- **La etiqueta de apertura:** consiste en el nombre del elemento (en este caso, p), envuelto entre claudátores angulares de apertura y cierre. Esto indica donde empieza o empieza a tener efecto el elemento; en este caso, donde empieza el párrafo.
- **La etiqueta de cierre:** es la misma que la etiqueta de apertura, excepto que incluye una barra inclinada antes del nombre del elemento. Esto indica donde termina el elemento; en este caso, donde acaba el párrafo. No añadir una etiqueta de cierre es uno de los errores estándar para principiantes y puede provocar resultados extraños.
- **El contenido:** este es el contenido del elemento, que en este caso es sólo texto.
- **El elemento:** la etiqueta de apertura, la etiqueta de cierre y el contenido juntos forman el elemento.

2.3.Atributos



Los atributos contienen información adicional sobre el elemento que no quisiera que aparezca en el contenido real. Aquí, class es el nombre del atributo y editor-note es el valor del atributo. El atributo class os permite dar al elemento un identificador no único que se puede utilizar para orientarlo (y cualquier otro elemento con el mismo valor de clase) con información de estilo y otras cosas. Algunos atributos no tienen valor, como obligatorio.

Los atributos que establecen un valor siempre tienen:

- Un espacio entre éste y el nombre del elemento (o el atributo anterior, si el elemento ya tiene uno o más atributos).
- El nombre del atributo seguido de un signo igual.
- El valor del atributo envuelto con comillas de apertura y de cierre.

Nota: los valores de atributos simples que no contienen espacios en blanco ASCII (o ninguno de los caracteres " ' ' = < >) pueden permanecer sin comillas, pero se recomienda

citar todos los valores de atributos, ya que hace que el código sea más coherente y comprensante.

2.4. Jerarquía

También podéis poner elementos dentro de otros elementos, esto se llama nidificación. Si quisiéramos afirmar que nuestro gato es muy malhumorado, podríamos envolver la palabra "mucho" en un elemento ``, lo que quiere decir que la palabra debe subrayarse mucho:

```
<p>My cat is <strong>very</strong> grumpy.</p>
```

Sin embargo, tenéis que asegurarnos de que vuestros elementos estén imbricados correctamente. En el ejemplo anterior, primero hemos abierto el elemento `<p>`, después el elemento ``; por lo tanto, primero tenemos que cerrar el elemento ``, después el elemento `<p>`. El siguiente es incorrecto:

```
<p>My cat is <strong>very grumpy.</p></strong>
```

Los elementos deben abrirse y cerrarse correctamente para que estén claramente dentro o fuera los unos de los otros. Si se superponen como se muestra más arriba, vuestro navegador web intentará adivinar lo que estáis intentando decir, lo que puede dar lugar a resultados inesperados. ¡Así que no lo hagáis!

2.5. Elementos vacíos

Algunos elementos no tienen contenido y se llaman elementos vacíos. Por ejemplo el elemento `img` que incrusta una imagen en nuestra web:

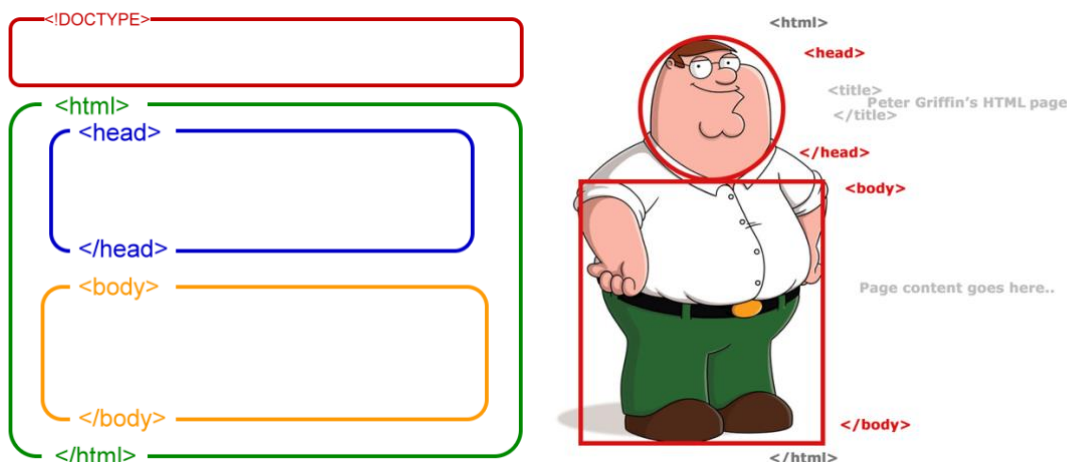
```

```

Contiene dos atributos, pero no hay ninguna etiqueta de cierre `` ni contenido interno. Esto se debe a que un elemento de imagen no envuelve el contenido para afectarlo. Su finalidad es incrustar una imagen en la página HTML en el lugar donde aparece.

3. ANATOMÍA DEL HTML

3.1. Visión General



El contenido ilustrado en la imagen es el básico de un fichero HTML. Ahora, identifica a continuación los elementos al ejemplo:

```
<!doctype html>
<html lang="en-US">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width" />
    <title>My test page</title>
  </head>
  <body>
    
  </body>
</html>
```

En los siguientes apartados se estudian los elementos por separado.

3.2. Declaración Inicial DOCTYPE

```
<!doctype html>
```

Es un preámbulo obligatorio. En la niebla de los tiempos, cuando HTML era joven (alrededor de 1991/92), los doctypes debían actuar como enlaces a un conjunto de reglas que la página HTML debía seguir por considerarse un buen HTML, que podría significar la comprobación automática de errores y otras cosas útiles. Sin embargo, en estos días, no hacen falta y básicamente sólo se necesitan para asegurarnos de que vuestro documento se comporta correctamente. Eso es todo lo que necesitas saber de momento.

3.3. Etiqueta HTML

Este elemento envuelve todo el contenido de la página entera y a veces se conoce como elemento raíz. También incluye el atributo lang, que define el idioma principal del documento.

3.4. Etiqueta HEAD

Este elemento actúa como contenedor para todas las cosas que queréis incluir en la página HTML que no sean el contenido que estáis mostrando a los espectadores de vuestra página. Esto incluye cosas como palabras clave y una descripción de la página que quisiera que aparezca en los resultados de la búsqueda, para diseñar nuestro contenido, declaraciones de conjunto de caracteres y mucho más.

En el ejemplo, dentro de la etiqueta HEAD tenemos otros elementos del estándar que deben estar presentes:

- `<meta charset="utf-8">` Este elemento establece el conjunto de caracteres que debe utilizar vuestro documento en UTF-8, que incluye la mayoría de caracteres de la gran mayoría de los idiomas escritos. Básicamente, ahora puede gestionar cualquier contenido textual que podamos poner. No hay ninguna razón para no configurarlo, y puede ayudar a evitar algunos problemas más adelante.
- `<meta name="viewport" content="width=device-width">` Este elemento de la ventana gráfica garantiza que la página se renderiza a la anchura de la ventana gráfica, evitando que los navegadores móviles muestren páginas más anchas que la ventana gráfica y luego las reduzcan.
- `<title></title>` El elemento `<title>`. Esto establece el título de vuestra página, que es el título que aparece en la pestaña del navegador donde se carga la página. También se utiliza para describir la página cuando la añadió como favorito.

3.5. Etiqueta BODY

Contiene todo el contenido que queréis mostrar a los usuarios web cuando visitan vuestra página, ya sea texto, imágenes, vídeos, juegos, pistas de audio reproducibles o cualquier otra cosa.

→ Haz la Práctica 2.1 ←