

Introducción al HTML

HTML y lenguajes de marcado. Lenguaje HTML
Documentos HTML, Edición

HTML y lenguajes de marcado.

Lenguaje HTML

HTML y Lenguajes de marcado

HTML. Origen: Tim Berners-Lee. Primeros años
Lenguajes de marcado. SGML. Evolución. Características
Evolución del HTML. Versiones.

XML. Formación y validez. DTD. Espacio de nombres

HTML5. XHTML (XML + HTML.). HTML5. Origen.

Componentes. Tecnologías asociadas. Nuevo paradigma. Semántica

Documentación. Libros. Web

Lenguaje HTML

Elementos del lenguaje. Etiquetas HTML. Atributos.

Convenciones del lenguaje. Comentarios

Atributos globales. Identificadores (Id)y clases (class)

Lenguajes de marcado generales: SGML

Standard Generalized Markup Language (Lenguaje de Marcado Estándar Generalizado).

ISO 8879:1986, Information processing — Text and office systems — Standard Generalized Markup Language (SGML).

El lenguaje SGML sirve para especificar las reglas de etiquetado de documentos y no impone en sí ningún conjunto de etiquetas en especial.

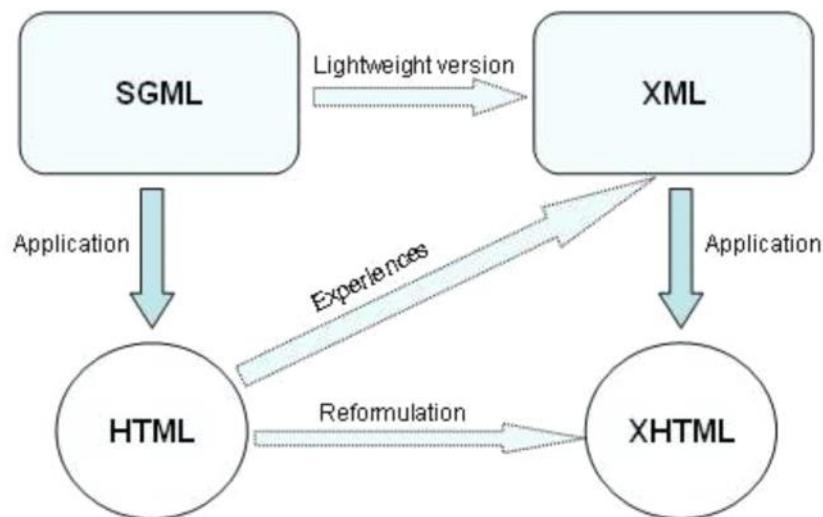
1974 – IBM define GLM, un sistema para la organización y etiquetado de documentos

1986 - La Organización Internacional de Estándares (ISO) normalizó este lenguaje

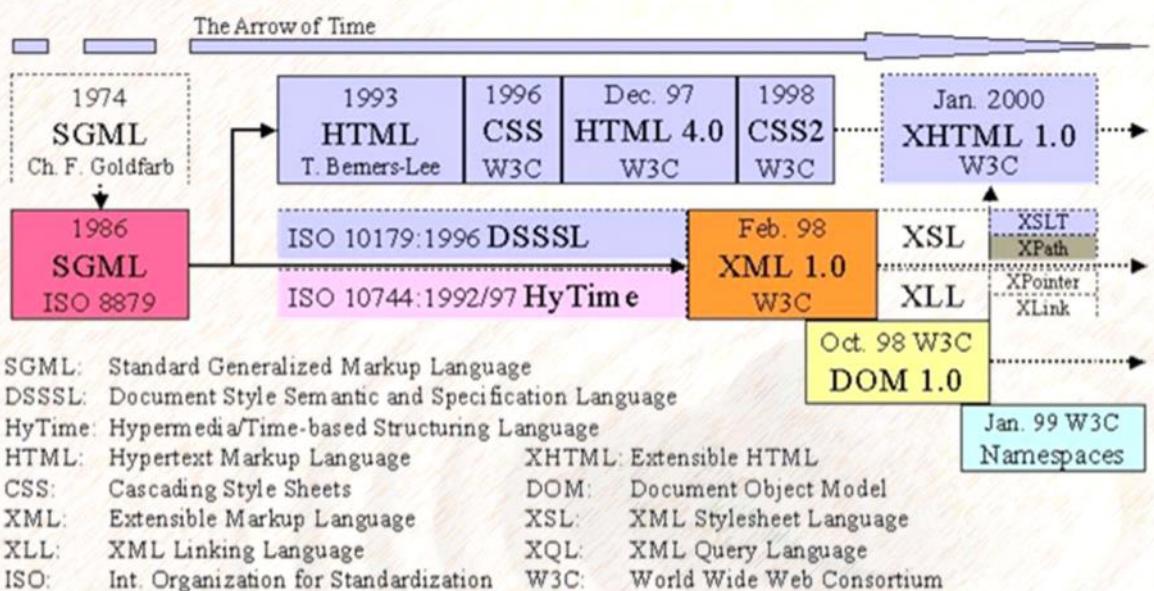
Document: Bungler OED At: "<entry>"

```
<entry>
  <hwsec>
    <hwgrp>
      <hwlem>bungler</hwlem>
      <pron>b<I>ʌŋglər</I>-ŋglər</pron>. </hwgrp>
    <vfl>Also <vd>b</vd> <vf>bungler</vf>,
      </vfl>
    <etym>f. as prec. + <xra><xlem>-ER</xlem>
      <sen>One who bungles; a clumsy unskillful
        <quot>
          <qdat>1533 </qdat>
          <auth>More </auth>
          <wk>answ. Poyson, Bk. </wk>Wks. (1557)
          <qtxt>He is even but a very bungler.
            </qtxt>
```

Lenguajes de marcado generales: SGML – HTML – XML...



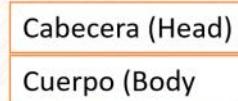
Evolución de los lenguajes de marcado



Características generales

Lenguajes de marcado

Estructura general
de un documento



Etiquetas o marcas
<Tag> </tag>

Descripción de la estructura
y la sintaxis de un
documento

- Metadatos
 - Codificación de caracteres
 - Instrucciones de proceso
 - Comentarios <!-->
 - Secciones: contenido no del lenguaje
- Elementos
- Atributos (modificadores de la etiqueta)
- Entidades (Caracteres escape): &< &&

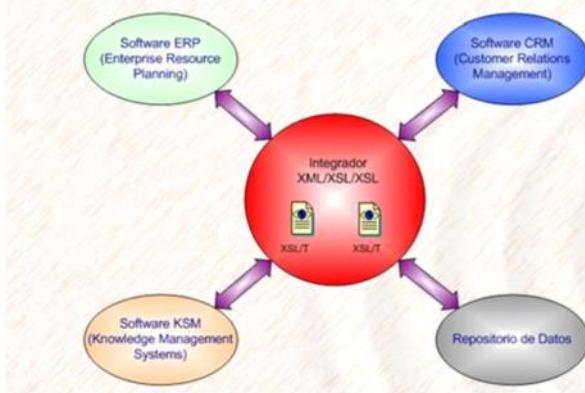
DTD: **document type definition**
(definición de tipo de
documento):

XML. Lenguaje e interface

eXtensible Markup Language
(lenguaje de marcas extensible)

W3C (1998)

No se crea pensando en Internet, sino como estándar para el intercambio de información estructurada entre diferentes plataformas



Permite definir la **gramática de lenguajes** específicos para **almacenar datos** en forma legible en documentos estructurados de gran tamaño.

Varias aplicaciones se pueden comunicar entre sí o integrar información

- bases de datos,
- editores de texto,
- hojas de cálculo
- casi cualquier cosa.

XML. Características

- No incorpora ningún comportamiento
- No predefine ninguna forma de representación de los datos

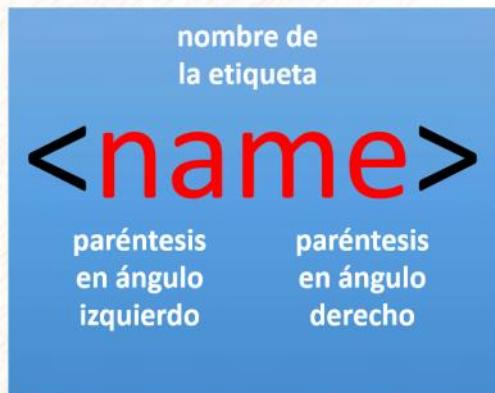
Vemos como lo muestra el navegador

- Independiza los datos (e.g. frente a HTML)
- Simplifica el uso compartido de los datos
- Simplifica el transporte de los datos
- Simplifica el uso de los datos en múltiples plataformas

Es la base de diversos lenguajes utilizados en internet

- XHTML
- XML Schema (web semántica)
- SVG (gráficos vectoriales)
- WSDL (servicios web)
- RSS (sindicación de contenidos)

Etiquetas en XML



Etiqueta de apertura

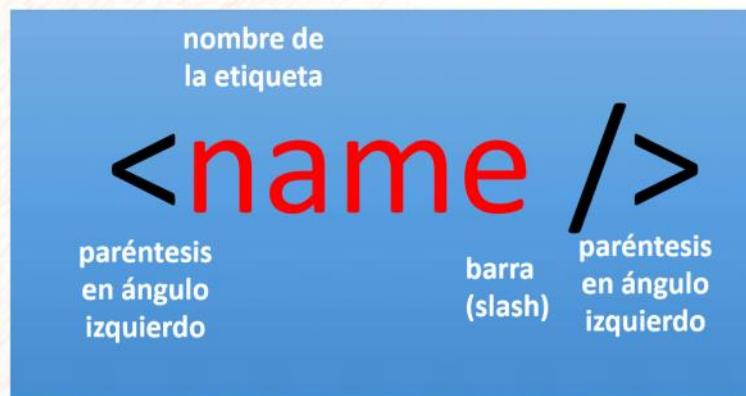


Etiqueta de cierre

Etiquetas dobles (two-sided tag)

```
<titulo>Everything</titulo>
```

Etiquetas en XML



La etiqueta es auto conclusiva: se cierra ella misma y contiene toda la información necesaria.

Siempre termina con />

Etiquetas sencillas (one-sided tag)

<informacion/>

Documentos bien formados

Los documentos denominados como «bien formados» (*well formed*) son aquellos que cumplen con todas las definiciones básicas de formato

Pueden ser analizados por cualquier **analizador sintáctico** (*parser*) que cumpla con la norma. .

- Estructura estrictamente jerárquica:
 - Etiquetas correctamente **anidadas**
 - Elementos con contenido correctamente **cerrados**
 - Un sólo **elemento raíz** o inicial del que todos los demás sean parte.
- Valores de atributos entre **comillas** simples o dobles.
- Sensible a **mayúsculas y minúsculas**.
- Un conjunto de caracteres llamados **espacios en blanco** (espacios, tabuladores, retornos de carro, saltos de línea) se definen específicamente.
- Las estructuras, tipos de elementos, entidades, elementos particulares, etc tienen un nombre asignado -> Generar espacios de nombres

Se separa esto del concepto de validez



Ejemplo de documento bien formado en XML

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE Edit_Mensaje SYSTEM "Edit_Mensaje.dtd">

<Edit_Mensaje>
  <Mensaje>
    <Remitente>
      <Nombre>Nombre del remitente</Nombre>
      <Mail>Correo del remitente</Mail>
    </Remitente>
    <Destinatario>
      <Nombre>Nombre del destinatario</Nombre>
      <Mail>Correo del destinatario</Mail>
    </Destinatario>
    <Texto>
      <Asunto>
        Este es mi documento con una estructura muy sencilla no contiene atributos ni entidades...
      </Asunto>
      <Parrafo>
        Este es mi documento con una estructura muy sencilla no contiene atributos ni entidades...
      </Parrafo>
    </Texto>
  </Mensaje>
</Edit_Mensaje>
```

Head

Body

Metadatos

Etiquetas

Elementos (Datos)

Elementos

Los elementos
contienen

Otros elementos
Atributos
Texto (Datos)

```
<Nombre_Completo genero="Masculino">
  <Nombre>Pepito</Nombre>
  <Apellidos>Perez</Apellidos>
</Nombre_Completo>
```

Reglas de
nombrado

- Sensible al caso
- Comenzar por letra o guion bajo
- No pueden empezar por las letras xml (XML, Xml...)
- Pueden contener letras, dígitos, guiones, guiones bajos y puntos.
- No pueden contener espacios

Documentos válidos

Que un documento esté «bien formado» solamente se refiere a su estructura sintáctica básica, es decir, que se componga de elementos, atributos y comentarios como XML especifica que se escriban.

Ahora bien, cada aplicación de XML, es decir, cada lenguaje definido con esta tecnología, necesitará especificar cuál es exactamente la **relación** que debe verificarse entre los **distintos elementos** presentes en el documento.

Se especifica en un documento externo

- **DTD** —Document Type Definition, (Definición de Tipo de Documento)
- **XSchema**

Crear una definición equivale a crear un nuevo lenguaje de marcado, para una aplicación específica.

En HTML o XHTML, el DTD es un estándar implícito al lenguaje: no lo define el usuario

El DTD correspondiente al ejemplo anterior en XML

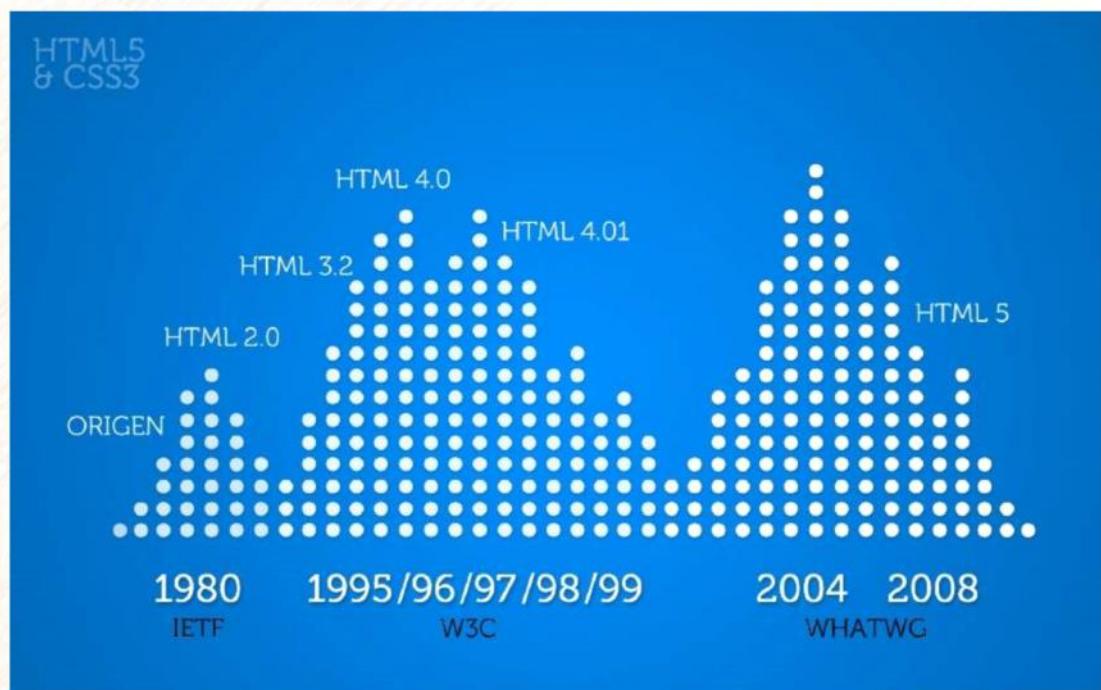
```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<!-- Este es el DTD de Edit_Mensaje --&gt;

&lt;!ELEMENT Mensaje (Remitente, Destinatario, Texto)*&gt;
&lt;!ELEMENT Remitente (Nombre, Mail)&gt;
&lt;!ELEMENT Nombre (#PCDATA)&gt;
&lt;!ELEMENT Mail (#PCDATA)&gt;
&lt;!ELEMENT Destinatario (Nombre, Mail)&gt;
&lt;!ELEMENT Nombre (#PCDATA)&gt;
&lt;!ELEMENT Mail (#PCDATA)&gt;
&lt;!ELEMENT Texto (Asunto, Parrafo)&gt;
&lt;!ELEMENT Asunto (#PCDATA)&gt;
&lt;!ELEMENT Parrafo (#PCDATA)&gt;</pre>
```



DTD: **document type definition**
(definición de tipo de documento)

Evolución del HTML: versiones



Evolución del HTML: versiones

“HTML 1”

1993, primera proposición para una especificación de HTML (IETF) a partir del lenguaje inicial desarrollado por Tim Berners-Lee

“HTML 2”

1993, Boceto competidor de Dave Raggett. Incorpora imágenes. Raggett *HTML+ (Hypertext Markup Format)*

HTML 3.0

Estándar propuesto por el recién formado W3C en marzo de 1995.

HTML 3.2, abandonaba la mayoría de las nuevas características del HTML 3.0 y, a cambio, adoptaba muchos elementos desarrollados inicialmente por los navegadores web

HTML 4.0

Recomendación del W3C en 1997, que inicia una renovación del estándar.
Se consolida en 1999 con la 4.01

XHTML

HTML 5

XHTML

HTML + XML

XHTML

- El DTD limitado del HTML
- Restricciones de formales propias del XML

Exige documentos válido y bien formados
Que puedan ser parseados siguiendo
estrictamente los estándares XML



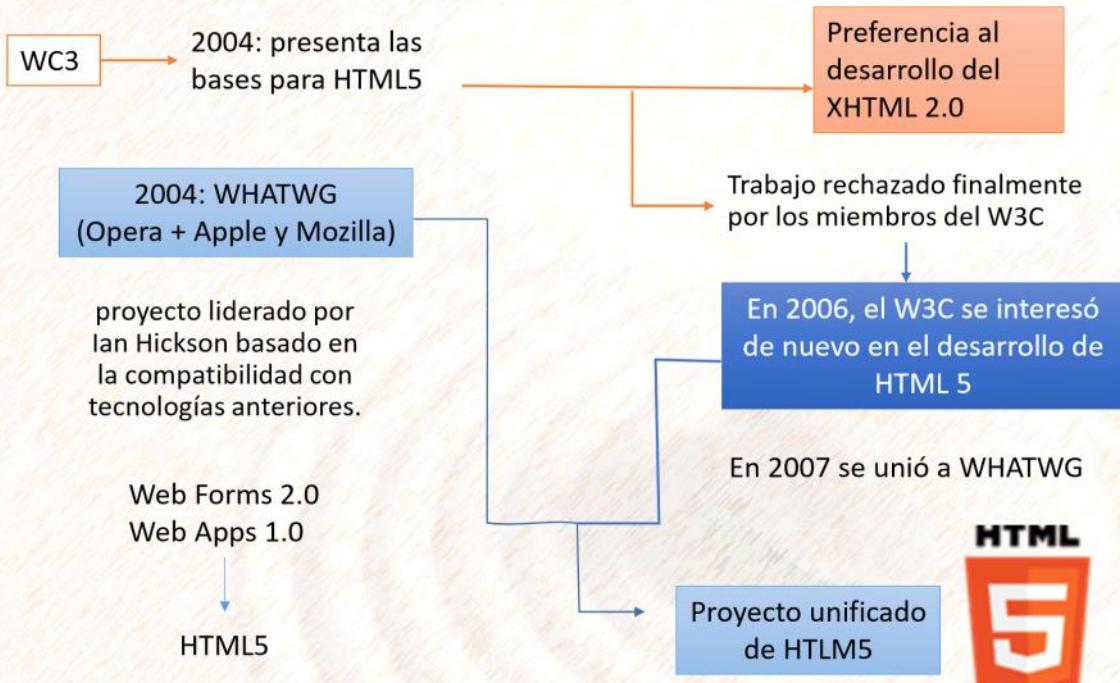
Desarrollado
por el W3C

En **2000** la recomendación 1.0
En **2001** la recomendación 1.1



W3C se vuelca en el borrador de XHTML 2.0

De XHTML a HTML5



HTML5 / CSS3

Conceptos semánticos Nuevas Etiquetas Nuevas APIs JavaScript

The image displays a grid of eight orange icons, each representing a different feature or aspect of HTML5 and CSS3. The icons are arranged in two rows of four. The top row includes: 1) Three stacked horizontal bars, labeled 'Semantics'; 2) A stylized 'E' shape, labeled 'CSS3'; 3) A clapperboard icon, labeled 'Multimedia'; and 4) A hexagonal cube icon, labeled 'Graphics & 3D'. The bottom row includes: 1) An icon resembling a television screen, labeled 'Device Access'; 2) A flower-like starburst icon, labeled 'Performance'; 3) An 'Offline' symbol inside a circle, labeled 'Offline & Storage'; and 4) A stylized 'W' or network icon, labeled 'Connectivity'.

Conceptos semánticos	Nuevas Etiquetas	Nuevas APIs JavaScript
Semantics	CSS3	Multimedia
Device Access	Performance	Offline & Storage
Connectivity		

Tecnologías asociadas



CSS3
JS / JQUERY
SVG
MOBILE

Elementos clave en HTML5

Cambio de paradigma → Nueva filosofía basada en el pragmatismo
En cierto modo lo opuesto a XHTML

Predominio de los aspectos semánticos → Redefinición de etiquetas
Toda una serie de etiquetas nuevas
↓
Construidas en base al principio anterior

Soporte en los navegadores

domingo, 14 de mayo de 2017 13:39

Can I use <https://caniuse.com/>

The screenshot shows the CanIuse.com homepage. At the top, there's a navigation bar with 'Home' (highlighted in yellow), 'News', the date 'May 13, 2017 - New feature added: CSS text-orientation', and links for 'Compare browsers' and 'About'. Below the navigation is a large orange header with the text 'Can I use' and a settings icon. A message box asks if the user wants to import usage data for Spain, with 'Import' and 'No thanks' buttons. The main content area has sections for 'Index of features', 'Latest features', 'Most searched features', 'Did you know?', 'Browser scores', and 'Third party tools'. The 'Latest features' section lists items like CSS text-orientation, Encrypted Media Extensions, CSS display: contents, Link type "noreferrer", and DOMMatrix. The 'Most searched features' section lists Flexbox, CSS Grid, CSS transforms, SVG, and CSS calc(). The 'Did you know?' section notes that usage data can be imported via the Settings panel. The 'Browser scores' section shows Chrome 58: 290, Firefox 53: 287, Safari 10.1: 244, and Edge 15: 221. The 'Third party tools' section lists various CanIuse-related tools. A 'See full list' link is also present.

Detected your country as "Spain". Would you like to import usage data for that country?

Import No thanks

Index of features

Latest features

- CSS text-orientation
- Encrypted Media Extensions
- CSS display: contents
- Link type "noreferrer"
- DOMMatrix

Most searched features

- Flexbox
- CSS Grid
- CSS transforms
- SVG
- CSS calc()

Did you know?

Usage data for all countries and continents can be imported via the **Settings panel**.

Browser scores

Current version	Dev version
Chrome 58: 290	
Firefox 53: 287	
Safari 10.1: 244	
Edge 15: 221	

About these scores

Third party tools

- The CanIuse Embed — Add support tables to your site
- Caniuse Component — Add support tables to your presentations
- Caniuse command line tool
- Doluse..? — Lint your CSS to check what features work
- I want to use — Select multiple features and see what % of users can use them

See full list

Etiquetas en HTML

nombre de
la etiqueta

paréntesis en ángulo izquierdo

paréntesis en ángulo izquierdo

Etiqueta de apertura

nombre de
la etiqueta

</p>

paréntesis en ángulo izquierdo **barra (slash)**

Etiqueta de cierre

Etiquetas dobles (two-sided tag)

<i>Everything</i>

```
<a href="index.htm">Click Here</a>
```

Etiquetas en HTML



La etiqueta es auto conclusiva: se cierra ella misma y contiene toda la información necesaria

En XHTML sería:

**Etiquetas
sencillas
(one-sided tag)**

HTML

XHTML

Etiquetas en HTML: atributos

nombre de
la etiqueta

nombre del
atributo

valor del
atributo

< p lang="es">

<nombre_etiqueta nombre_atributo="valor del atributo">

Click Here

Etiqueta: <a>

Atributo: href="..."

Elemento Click Here

Etiqueta:

Atributo: scr="..."

Etiquetas en HTML: formato y “convenciones”

Los nombres de etiqueta, los nombres de atributos y los valores pueden ir en mayúscula o minúscula

lo habitual es que , los nombres de atributos y los valores vayan en **minúscula**

No es necesario encerrar los valores de los atributos entre comillas

lo habitual es que los valores de los atributos se indiquen **entre comillas**

Los atributos suelen ser específicos de una o varias etiquetas
Existen unos pocos **atributos globales**: válidos para todas las etiquetas

Comentarios

texto del comentario

<!--Comentario-->

paréntesis en
ángulo izquierdo,
admiración,
dos guiones

dos guiones,
paréntesis en
ángulo
izquierdo

<!--Esto es un comentario en HTML-->

Atributos globales

class	Nombre de estilo aplicable a diversos elementos
id	Nombre de estilo aplicable a un elemento
style	Estilo CSS en línea
title	Información extra sobre un elemento
lang	Idioma asociado a un elemento
translate	Elemento que se debe traducir en una página “localizado”
spellcheck	Activa o no la comprobación de ortografía y gramática
dir	Dirección del texto
tabindex	Orden tabular de un elemento
acceskey	Tecla rápida que activa/da el foco a un elemento
contenteditable	Carácter editable o no
contextmenu	Menú contextual de un elemento
draggable	Posibilidad de arrastrar de un elemento
dropzone	Acción por defecto al arrastrar un elemento
hidden	Indica el carácter no relevante de un elemento

Atributos *id* y *class*

Permiten identificar a cualquier etiqueta utilizada.

```
<etiqueta id="id_1">
```

La etiqueta utilizada queda identificada exclusivamente como *id_1*
En un documento nunca debe asignarse la misma id a más de una etiqueta

```
<tag_2 class="cls_1 cls_2 ...">
```

La etiqueta utilizada queda identificada como perteneciente a una clase,
En la que se puede incluir cualquier número de todo tipo de etiquetas

HTML. Documentos

Documentos HTML

Inicio del. Etiquetas !DOCTYPE y HTML

Jerarquía: Cabecera y cuerpo. Etiquetas HEAD y BODY

Cabecera. Etiqueta TITLE. Inclusiones e iconos. Metadatos.

Conjunto de caracteres. Caracteres especiales

Primera página Web. Ejercicio “Hola Mundo”

Omisiones y errores: funcionamiento del navegador

Validación; Sitios

Inicio del documento HTML

<!DOCTYPE ...>

Especificación del tipo de documento
(opcional, excepto en XHTML)

Le indica al navegador que el documento cumple determinado estándar, que será el que aquel utilizará para procesar el documento

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

```
<!DOCTYPE HTML PUBLIC "-//W3C/DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

<!DOCTYPE html>

HTML5



Inicio del documento HTML

Documento HTML
(todo lo demás)

<html>
...
</html>

Es el contenedor raíz en la estructura HTML, es decir dentro del que vamos a incluir todo el documento

```
<html lang="es">  
...  
</html>
```

HTML5



En la etiqueta html se incorpora el **atributo global “lang”**
lang= idioma (es, en, fr, ge, it, ru...)
Los idiomas pueden incluir dialectos: es~es (español de España)

Partes del documento HTML: cabecera (head) y cuerpo (body)

<!DOCTYPE ...>

<html>

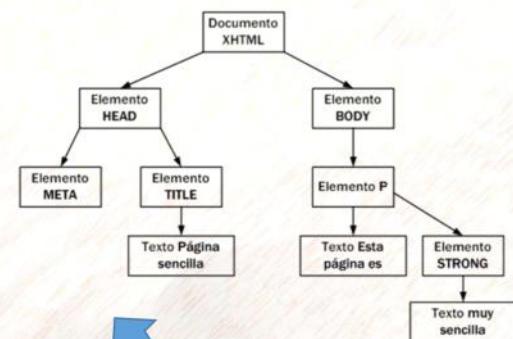
```
<head>  
...  
</head>
```

Cabecera

```
<body>  
...  
</body>
```

Cuerpo

</html>



Refleja el carácter de estructuras jerárquicas de los documentos HTML

La cabecera (head)

Cabecera

```
<head>  
...  
</head>
```

Titulo de la página

```
<title>Curso HTML5</title>
```

Metadatos

```
<meta name="valor" content="valor">
```

Inclusiones de archivos externos

```
<link rel="valor" href="valor">
```

Estilos y script embebidos en el documento

```
<style>.....</style>
```

Etiqueta <base>

<base>

especifica la url por defecto para todas las urls relativas y el destino por defecto de los hiperenlaces incluidos en al página

Atributos

href

corresponde a la url por defecto

target

indicael destino por defecto de los hiperenlaces

_blank

_parent

_self

_top

framename

```
<base href="http://www.miweb.com/images/" target="_blank">
```

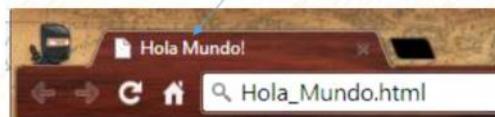
Etiqueta <title>

```
<title>Título</title>
```

Corresponde al título del documento web

```
<title>Hola Mundo!</title>
```

Normalmente será el texto mostrado en la barra superior de la ventana o pestaña del navegador



<title> es también un **atributo global**, que incorpora información adicional a cualquier etiqueta. En muchos casos se muestra como un recuadro emergente al pasar el ratón sobre un objeto

Inclusiones: css y scripts

<link>

usado para incorporar archivos externos, correspondientes a .

- estilos,
- imágenes o iconos

```
<link rel="stylesheet" href="misestilos.css">
```

Atributos

rel

significa “relación”; establece el uso que se hará del archivo:



- stylesheet,
- shortcut icon,
- apple-touch-icon

href

corresponde a la referencia del archivo

<script>

usado para incorporar archivos externos, correspondientes a códigos de **scripts** (i.e. JavaScript)

```
<script type='text/javascript' src='.../jquery.min.js'>
```

Iconos: favicon

```
<link rel="shortcut icon" href="fav_icon.png">
```

.ico → problemas fuera
de Windows
.png



Fijar
pestaña



25x25 px

Metadatos: name/content

<meta ...>

Cada etiqueta <meta> establece un metadato, utilizando los atributos: name y content, http-equiv y charset. Pueden sucederse tantos <meta> como sea necesario

Atributos
name
... content

```
<meta name="keywords"  
      content="pests, weeds, plants, flowers, ....">  
<meta name="author" content="Hege Refsnes">  
<meta name="application-name" content="...">  
<meta name="description" content="...">  
<meta name="generator" content="...">
```

Combinando los atributos *name / content* se establece el valor de una serie de metadatos o descriptores con información sobre el documento

- nunca se muestra en el navegador
- la utilizan motores de búsqueda y dispositivos que necesitan hacer una vista previa del documento u obtener un sumario de la información que contiene.

Metadatos: http-equiv

Atributo
http-equiv
... content

```
<meta http-equiv="refresh"  
      content="5; url=http://.../newpage.htm">  
<meta http-equiv="content-type" content=  
      "text/html; charset=UTF-8">  
<meta http-equiv="default-style" content="....">
```

- después de **content-type**, el atributo *content* permite indicar el tipo MIME del contenido (text/html) y el charset o página de códigos de caracteres (no se usa en HTML5)
- después de **default-style**, el atributo *content* permitía indicar la hoja de estilo preferida (si se han incluido varias)
- después de **refresh**, el atributo *content* permite definir un intervalo de tiempo para que se refresque la pantalla,
Además se puede indicar que cargue una URL diferente: permite por tanto redirigir a una URL diferente de la indicada en el navegador

Character Sets (<meta charset>)

Atributo
charset

<meta charset="utf-
8">

HTML5



La innovación de este elemento en HTML5, como en la mayoría de los casos, es solo **simplificación**. La nueva etiqueta <meta> para la definición del tipo de caracteres es más corta y simple.

Definición de la páginas de códigos que utilizará el agente de usuario (navegador) al renderizar un documento HTML

es crítico para la correcta presentación del documento, al determinar la página de códigos que utilizará el navegador debe coincidir con la codificación empleada en el programa que genera el documento HTML

Páginas de códigos: ASCII

Las páginas de códigos son tablas que definen el conjunto de caracteres, es decir que código lingüístico corresponde a cada conjunto de bits, dependiendo del alfabeto utilizado por un documento.

Inicialmente se utilizaba el conjunto de caracteres **ASCII** (*American Standard Code for Information Interchange*):

Cada carácter está representado por un número de 7 bits, lo que permite codificar únicamente 128 caracteres: letras mayúsculas y minúsculas, números y algunos caracteres habituales.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DEL	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2	SP	!	"	#	\$	%	&	'	()	*	+	,	-	.	
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	
5	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
															DEL	

Estos caracteres son entendidos en cualquier ordenador y por tanto pueden ser usados con libertad en un documento HTML

Páginas de códigos: ISO

Las páginas de **códigos iso-8859** son todos super sets del estándar US-ASCII, donde cada carácter se codifica mediante 1 Byte (8 bits), por lo que solo tienen cabida 256 caracteres diferentes: los 128 del ASCII y 128 del ASCII “extendido”
En este último existen variaciones entre los distintos conjuntos de códigos ISO

Una cadena de un Byte, de valor 11100001, (corresponde decimal 225)

Si se aplica la página de códigos iso-8859-1 (Latín)

Si se aplica la página de códigos iso-8859-6 (Árabe)

Si se aplica la página de códigos iso-8859-7 (Griego)

á
ف
α

Páginas de códigos: ISO

iso-8859-1	Latin 1	Lenguajes de Europa Occidental (Inglés, Francés...)
iso-8859-2	Latin 2	Lenguajes de Europa Central y Oriental (Checo, Polaco...)
iso-8859-3	Latin 3	Lenguajes de Europa del Sur, Esperanto y otros
iso-8859-4	Latin 4	Lenguajes de Europa del Norte (Letón, Lituano, Groenlandés...)
iso-8859-5	Cyrillic	Lenguajes de alfabeto cirílico (Búlgaro, Ruso, Serbio...)
iso-8859-6	Arabic	Lenguajes de alfabeto arábico
iso-8859-7	Greek	Griego moderno, signos matemáticos
iso-8859-8	Hebrew	Lenguajes de alfabeto hebreo
iso-8859-9	Latin 5	Turco
iso-8859-10	Latin 6	Lenguajes Nórdicos (Islandés, Inuit)
iso-8859-15	Latin 9 / Latin 0	Modificación del iso-8859-1 que incluye algunos caracteres nuevos, incluyendo el símbolo del Euro
ISO-2022-JP	Japanese 1	Lenguaje japonés
ISO-2022-JP-2	Japanese 2	Lenguaje Japonés
ISO-2022-KR	Korean	Lenguaje coreano

UNICODE

A partir de HTML 4.0 se soporta el **conjunto de caracteres (charset)** UNICODE.

Definido por el consorcio internacional UNICODE (<http://www.unicode.org/>) encargado de mantener el estándar en colaboración con ISO/IEC

- sistemas de escritura **modernos**: árabe, braille, copto, cirílico, griego, hebreo, latino, devanagari (sánscrito), sinogramas (hanja coreano, hanzi chino y kanji japonés), silabarios japoneses (hiragana y katakana), ...
- escrituras **históricas** extintas, para propósitos académicos: cuneiforme, griego antiguo, lineal B micénico, fenicio y rúnico.
- caracteres **no alfabéticos** como símbolos musicales y matemáticos, fichas de juegos como el dominó, flechas, iconos etc.



La versión 6.0 contiene
109.449 caracteres

Planos UNICODE

Cada conjunto definible por 2 bytes constituye un “plano” (*Plane*). En teoría podría corresponder a 65.536 caracteres (depende del sistema de codificación)

- BMP (*Basic Multilingual Plane*)
Agrupa los símbolos más habituales de la mayoría de lenguas actuales
- SMP (*Supplementary Multilingual Plane*): Lenguas antiguas y mas
- SIP (*Supplementary Ideographic Plane*): Ext. CJK (China, Japón, Corea)
- TIP (*Tertiary Ideographic Plane*): Lenguas antiguas asiáticas
- SSP (*Supplementary Special-purpose Plane*): usos especiales

Compatibilidad. BMP comienza por los caracteres del ASCII (a los que asigna el mismo número que tenían originalmente)

Páginas de códigos: UTF

UNICODE, permite varios sistemas de codificación conocidos como los conjuntos UTF (*Unicode Transformation Format*)

- UTF-8 utiliza un sistema no modal de longitud variable, de 1 a 4 Bytes , donde el valor de los primeros bits determina cuantos Bytes corresponden a un determinado código. Con ello se supera sin limitaciones el numero de posibles caracteres optimizando el uso de memoria para los primeros signos codificados (i.e. lenguas latinas)
- UTF-16 es también un sistema de codificación de caracteres de longitud variable, utiliza 2 Bytes para codificar el plano BMP y 4 Bytes para otros planos
- UTF 32 es de longitud fija de 4 Bytes, resultando más eficiente con los planos que no son BMP.
Teóricamente se podría llegar a codificar $2^{32} = 4.294.967.296$

Niveles de codificación

Al generar un documento

Le indicamos al editor en que código estamos trabajando

En un editor tan simple como el bloc de notas podemos elegir varias formas de codificación

Nombre: Nuevo documento de texto.txt

Tipo: Documentos de texto (*.txt)

Ocultar carpetas

Codificación:

- ANSI
- ANSI
- Unicode
- Unicode big endian
- UTF-8

Al leer un documento en el navegador

El navegador configurado correctamente utiliza el código indicado en los metadatos del documento

Carácteres especiales

Entidades

Secuencia de escape con nombre : &Nombre;
Secuencia de escape sin nombre : \$#123;

& (ampersand)	&	&	© (copyright)	©	©
< (menor que)	<	<	® (marca registrada)	®	®
> (mayor que)	>	>	° (grado)	°	°
(espacio fuerte)	 	 	± (más / menos)	±	±
¢ (centavo)	¢	¢	“ (comillas)	"e	"
£ (libra)	£	£	† (cruz o dagger)	†	†
¥ (yen)	¥	¥	™ (trademark)	™	™

El espacio fuerte (*nonbreaking space*) suele ser usado en exceso, incluso por algunos editores WYSIWYG y conversores de texto. Su uso debe ser el imprescindible y en ningún caso puede sustituir una correcta disposición (*layout*)

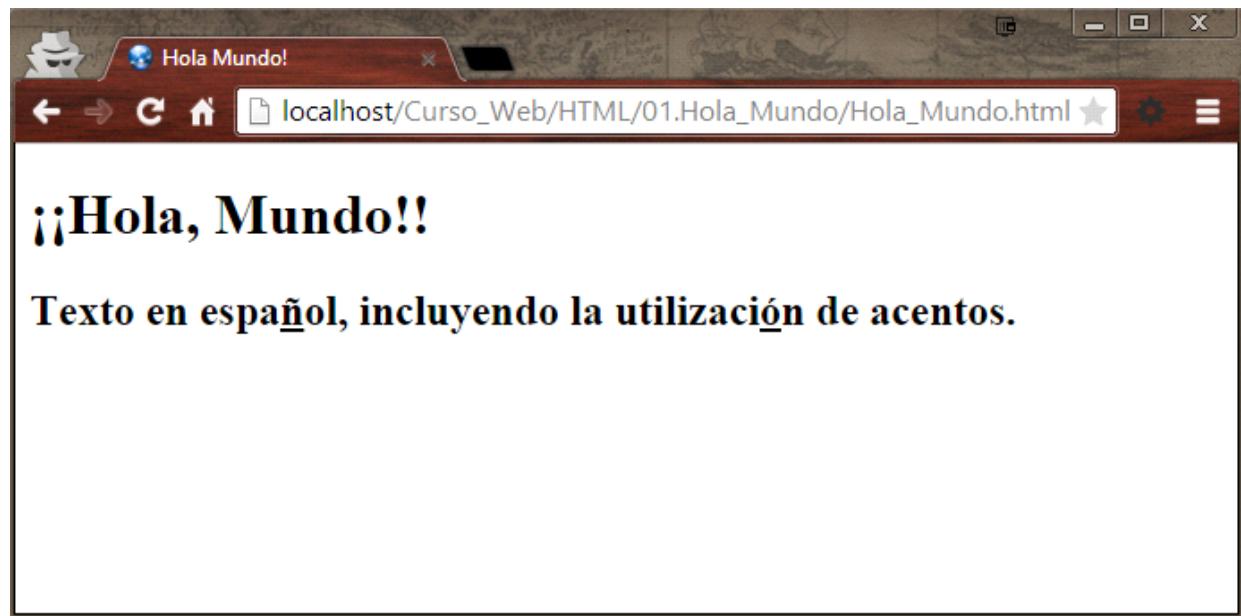
Carácteres especiales del español

En español, el uso de vocales acentuadas, la eñe y los símbolos de apertura de admiración e interrogación suponía el empleo de otros tantos caracteres especiales

Carácter	Secuencia escape	Sec. escape numérica	Carácter	Secuencia escape	Sec. escape 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	¥
¡	-----	¡	¿	-----	¿

Hola Mundo

jueves, 4 de mayo de 2017 13:47



Ejercicio (1)

Hola Mundo: Usando un editor de texto plano (e.g. Bloc de notas) generamos dos documentos HTML que muestren el mensaje “Hola Mundo”.

- en HTML4 y
- en HTML5

En la cabecera indicamos el título de la página (`<title>`) y todos los metadatos posibles. En el cuerpo presentaremos un párrafo, que incluirá el atributo `title`.

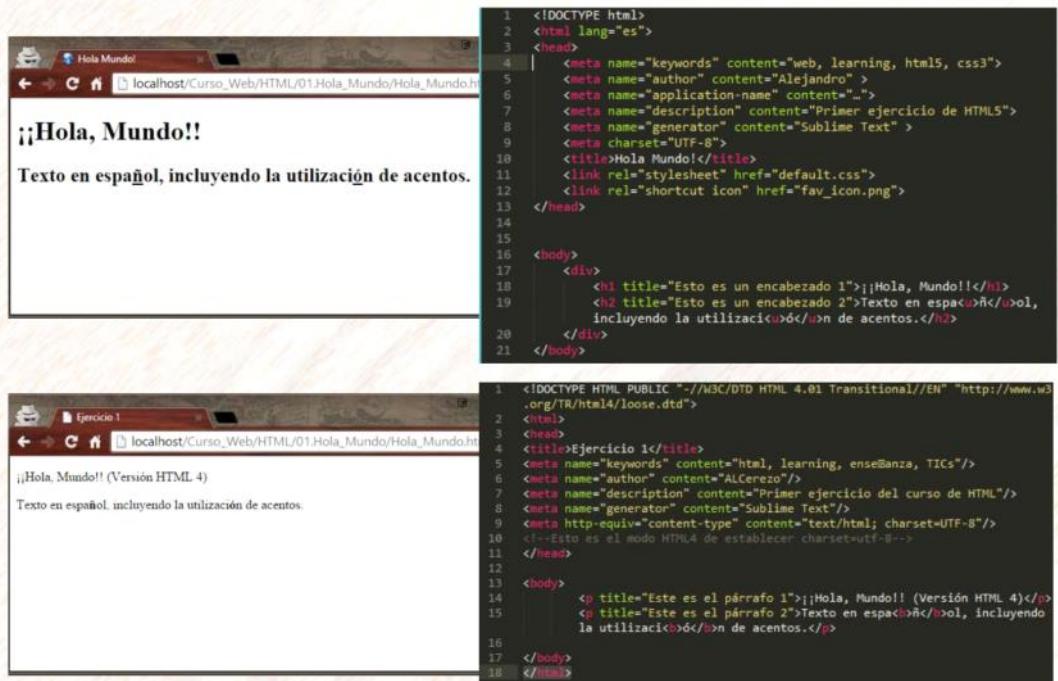
Objetivo: Distinguir cabecera y cuerpo, generando las primeras en HTNL4 y 5. Incluir titulo y metadatos en las cabeceras. Comprobar la validez del concepto “documento HTML” con total independencia de Internet y la Web.

Un segundo objetivo es configurar block de notas como editor de HTML y un navegador por defecto (e.g. Chrome)

Resultado. Validación

jueves, 4 de mayo de 2017 13:49

Ejercicio (1): solución



¡¡Hola, Mundo!!
Texto en español, incluyendo la utilización de acentos.

```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4   <meta name="keywords" content="web, learning, html5, css3">
5   <meta name="author" content="Alejandro" >
6   <meta name="application-name" content="">
7   <meta name="description" content="Primer ejercicio de HTML5">
8   <meta name="generator" content="Sublime Text" >
9   <meta charset="UTF-8">
10  <title>Hola Mundo!</title>
11  <link rel="stylesheet" href="default.css">
12  <link rel="shortcut icon" href="fav_icon.png">
13 </head>
14
15
16 <body>
17   <div>
18     <h1 title="Esto es un encabezado 1">¡¡Hola, Mundo!!</h1>
19     <h2 title="Esto es un encabezado 2">Texto en español, incluyendo la utilización de acentos.</h2>
20   </div>
21 </body>
```


¡¡Hola, Mundo!! (Versión HTML 4)
Texto en español, incluyendo la utilización de acentos.

```
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
2 <html>
3 <head>
4   <title>Ejercicio 1</title>
5   <meta name="keywords" content="html, learning, enseñanza, TICs"/>
6   <meta name="author" content="ALCerezo"/>
7   <meta name="description" content="Primer ejercicio del curso de HTML"/>
8   <meta name="generator" content="Sublime Text"/>
9   <meta http-equiv="content-type" content="text/html; charset=UTF-8"/>
10  <!-- Esto es el modo HTML4 de establecer charset=utf-8-->
11 </head>
12
13 <body>
14   <p title="Este es el párrafo 1">¡¡Hola, Mundo!! (Versión HTML 4)</p>
15   <p title="Este es el párrafo 2">Texto en español, incluyendo la utilización de acentos.</p>
16
17 </body>
18 </html>
```

Ejercicio (1): solución

```
<!DOCTYPE html>
<html lang="es">
<head>
    <meta name="keywords" content="web, learning, html5, css3">
    <meta name="author" content="Alejandro" >
    <meta name="application-name" content="...">
    <meta name="description" content="Primer ejercicio de HTML5">
    <meta name="generator" content="Bluefish 2.2.4" >
    <meta charset="UTF-8">
    <title>Hola Mundo!</title>
    <link rel="stylesheet" href="misestilos.css">
    <link rel="shortcut icon" href="fav_icon..ico">
</head>

<body>
    <div>
        <h1>;Hola, Mundo!!</h1>
        <h2>Texto en español, incluyendo la utilización de acentos.</h2>
    </div>
</body>
```

Validación

Herramienta para detectar errores o ausencias en el código que puedan provocar problemas de visualización en algún navegador

- El estilo en la sintaxis es independiente de la validación.
- Queda como opción de cada desarrollador.
- Conviene respetar cierta coherencia (Buenas prácticas)

W3C Markup Validation Service (<http://validator.w3.org/>)

(X)HTML5 Validator (<http://html5.validator.nu/>)

Validator.nu (X)HTML5 Validator (Living Validator)

Ejercicio (1): "solución mínima"



A screenshot of a web browser window titled "Ejercicio 1 - Hola Mundo". The address bar shows "localhost/Curso_Web/". The page content is "Hola mundo. Viva la diversión". Below the browser is a code editor window showing the following HTML code:

```
1 <!DOCTYPE html>
2 <html lang="es">
3   <head>
4     <meta charset="UTF-8">
5     <title>Ejercicio 1 - Hola Mundo</title>
6     <link rel="shortcut icon" href="fav_icon.png">
7   </head>
8   <body>
9     Hola mundo. Viva la diversión
10    </body>
11  </html>
```



A screenshot of a terminal window. The title bar says "localhost/Curso_Web/HTT". The address bar shows "localhost/Curso_Web/HTML/01.Hola_Mundo/Hola_Mundo.bad". The output window shows "¡¡Hola Mundo!!". Below the terminal is a code editor window showing the following command:

```
1 ¡¡Hola Mundo!!
```

Omisiones y navegadores



Si ignoramos todas las normas del HTML

El navegador intenta llenar los huecos (e.g. crea elementos html, head y body)

Al final presenta tantos elementos como pueda

- El papel del **navegador** es muy activo: siempre intenta presentar el documento
- Las **diferencias** entre navegadores cobran mayor importancia