

Tecnologías Web para la presentación

HTML/XHTML

Lenguajes de Marcas

¿Qué es HTML?

- **HTML (HyperText Markup Language):** Lenguaje con el que se definen páginas Web
 - Permite describir el contenido de una página, incluyendo texto y otros elementos (imágenes, videos, pequeñas aplicaciones, etc.)
 - Una página HTML consta de texto y marcas especiales que permiten indicar algún tratamiento especial (estructura, formato, hiperenlace, etc.)
 - Las marcas se indican en formato `<marca>...</marca>`

```
<html>
  <head>
    <title>Título de la página</title>
  </head>
  <body>
    <h1>Título de sección</h1>
    <p>Texto...</p>
    <p>Texto <b>en negrita</b></p>
  </body>
</html>
```



Historia HTML

- **1989:** Inicio de su desarrollo
- **1991:** Lanzamiento de la web
 - Tim berners-Lee desarrolla y lanza su proyecto de investigación llamado World Wide Web.
 - Se define HTML como un subconjunto del lenguaje SGML.
 - Se publica el primer documento formal con la descripción de HTML bajo el nombre "HTML Tags".
- **1993:** Primera propuesta de estandarización de HTML
 - El organismo IETF (Internet Engineering Task Force) realiza la primera propuesta oficial para crear un estándar sobre HTML.
- **1994:** Creación de la W3C.
- **1995:** HTML 2.0 primer estándar oficial.
 - IETF organizó grupo de trabajo y publicaron el estándar HTML 2.0.
 - A pesar de su nombre fué el primer estándar de HTML.

HTML/XHTML

Historia HTML

- **1996:** La W3C entra en juego
 - A partir de este año los estándares de HTML los publica el W3C.
- **1997:** Se publica HTML 3.2
 - Primer estándar oficial de la W3C.
 - Incluye applets de java y texto que fluye alrededor de las imagenes.
- **1998:** Publicación de HTML 4.0
 - Gran salto tecnológico respecto a versiones anteriores.
 - Incluye hojas de estilo CSS.
 - Permite la ejecución de código Javascript en navegadores.
- **1999:** HTML 4.01 - XHTML
 - HTML 4.01 última versión HTML, es un HTML 4.0 con pocas novedades.
 - El w3C decide abandonar HTML y combinarlo con XML para producir XHTML.

HTML/XHTML

Historia HTML

- **2000:** Publicación XHTML 1.0
- **2004:** Creación WHATWG
 - Mozilla, Google y Opera no están de acuerdo con el rumbo de la W3C y XHTML.
 - Organización al margen de W3C.
- **2008:** Publicación primera versión HTML5
 - La W3C da un paso atrás y retoma la actividad de estandarización de HTML.
- **2014:** HTML 5 estandarizado por parte del W3C.
 - Incluye CSS3.
 - Se abandona definitivamente XHTML
- **2016:** HTML 5.1

HTML/XHTML

HTML/XHTML

HTML vs. XHTML vs. HTML5

- **HTML:** Lenguaje original
 - Basado en SGML (*Standard Generalized Markup Language*)
 - Es casi un lenguaje XML, pero tiene elementos no compatibles con XML
 - Empieza a cobrar fuerza la idea de no mezclar marcado de estructura (tabla, sección, encabezado, etc.) con marcado de presentación (fuente, color, alineamiento, etc.)

- **HTML 4.01:** Última versión publicada de HTML
 - Dos modos:
 - Strict: no permite marcado de presentación
 - Transitional: permite marcado de presentación (center, font, align, bgcolor, ...)
 - Framesets:
 - Primera incorporación al estándar de la etiqueta <frameset> (aunque ya se usaba comúnmente en Netscape)

HTML/XHTML

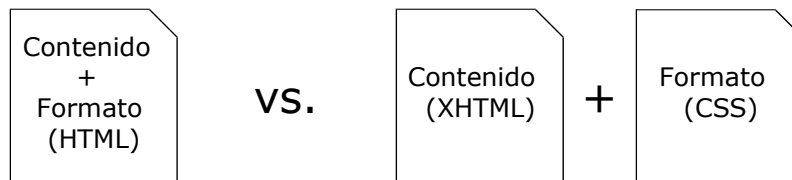
HTML vs. XHTML vs. HTML5

- **XHTML:** versión basada en XML de HTML 4.01
 - Compatibilidad con otros documentos XML
 - Combinar con otros documentos en lenguajes basados en XML, XForms, MathML, SMIL, SVG, etc.)
 - Usar herramientas (XSLT)
 - XHTML es más estricto que HTML
 - Exige que los documentos estén bien formados (requisito de XML)
 - Se pueden validar con <http://validator.w3.org/> o con HTML Tidy <http://www.w3.org/People/Raggett/tidy/>
 - Hay herramientas que permiten pasar de HTML a XHTML
 - Por ejemplo, el navegador Amaya (<http://www.w3.org/Amaya/>)

HTML/XHTML

HTML vs. XHTML vs. HTML5

- XHTML separa claramente contenido y presentación
 - **XHTML Strict** obliga a que la asignación del formato de presentación se haga de manera separada mediante hojas de estilo **CSS**
 - Esta separación tiene ventajas:
 - La gestión del estilo de todo el sitio web está localizada en un punto concreto: las hojas de estilo
 - Es posible modificar todo el estilo del sitio (manteniendo su coherencia) sin editar todas sus páginas
 - Se pueden tener varios estilos definidos para el mismo sitio web
 - Los usuarios de un sitio web pueden seleccionar el conjunto de colores y estilos (tema) del sitio web
 - Soporte a la navegación desde múltiples dispositivos, adaptando las páginas a sus características (tamaño de pantalla, etc.)



HTML/XHTML

HTML vs. XHTML vs. HTML5

- Al estar basado en XML, XHTML permite integrar fácilmente otros lenguajes de la familia XML
 - Mayor extensibilidad.
- XHTML tiene reglas más estrictas para definir la estructura de un documento
 - Por ejemplo: todos los elementos tienen que estar bien anidados y tener siempre la etiqueta de cierre
 - Al estar bien escritos se emplean menos recursos para procesar los documentos
 - Ventaja para hacer navegadores más eficientes

HTML/XHTML

HTML vs. XHTML vs. HTML5

- **HTML5:** Evolución de XHTML y HTML 4.01
 - Es el estándar vigente a día de hoy
 - XHTML 2.0 ha sido abandonado
 - Desarrollado por el WHATWG, adoptado por el W3C
 - Participación de Opera, Mozilla, Apple
 - Un paso más
 - "Estándar viviente": W3C publica, pero WHATWG no para de añadir
 - Añaden cosas y recomiendan no usar otras, pero quitan poco o nada
 - HTML5 vs. XHTML5
 - "Un estándar, dos formas de escribir"
 - La recomendación vigente es codificar en HTML5 (la transición hacia navegadores optimizados para XML puro no funcionó).
 - XHTML5 esta abandonado.

HTML y XHTML

- Versiones
 - HTML 4.01 (1999)
 - <http://www.w3.org/TR/html401/>
 - XHTML 1.0 (2000, revisado en 2002)
 - <http://www.w3.org/TR/xhtml1/>
 - XHTML 1.1 (2001)
 - XHTML™ Basic 1.1 - Second Edition (2010)
 - <http://www.w3.org/TR/xhtml-basic/>
 - XHTML™ 1.1 - Module-based XHTML - Second Edition (2010)
 - <http://www.w3.org/TR/xhtml11/>
- HTML5 standard
 - **W3C Publicación oficial del estándar HTML5 en 2014**
 - Todos los navegadores soportan HTML5, estándar de facto.

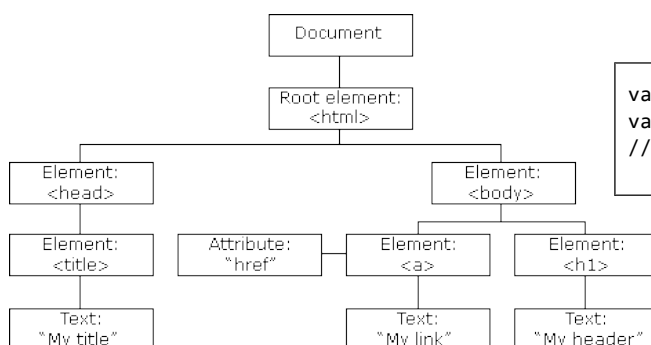
Herramientas para crear una página Web

- Mínimo
 - Un editor de texto
 - Por ejemplo, Notepad++: <http://notepad-plus-plus.org/>
 - Algún navegador
 - Para visualizar las páginas

HTML/XHTML

DOM

- Definido por W3C
- Modelo de Objetos del Documento (*Document Object Model*)
 - DOM define objetos y propiedades de los elementos HTML y XML, y los métodos para acceder a ellos
 - Representación de documentos HTML y XML
 - API para consultar y manipular los documentos (contenido, estructura, estilo)
- Los objetos de un documento se organizan en una jerarquía (árbol): jerarquía DOM



```
var elemento=document.getElementById("intro");
var cabecera=document.getElementnntByTag("title");
// ...
```

HTML/XHTML

Recomendaciones sobre los ficheros

- Una página Web involucra muchos ficheros
 - Estructurar adecuadamente los ficheros en directorios
 - En el raíz, index.html, index.xhtml o index.php
 - /img
 - /res
 - etc.
 - Nombrar los ficheros sin caracteres extraños, ni espacios, etc. para mayor compatibilidad entre sistemas operativos (Windows, Unix)
 - Ser consistente en el uso de la extensión de los ficheros
 - .html o .xhtml
- Para imágenes
 - Utilizar formatos estándar (PNG o JPEG) mejor que propietarios (GIF)
 - Comprobar que se adecúan al tamaño de la pantalla
 - Comprobar que su tamaño es moderado
- En general, comprobar que la página se ve correctamente con todos sus componentes remotamente
 - Pudiera haber referencias absolutas a ficheros locales en vez de relativas

Convenciones nombres ficheros

Es importante que siga estas reglas y convenciones cuando nombre sus archivos:

- Utiliza sufijos adecuados para sus archivos:
 - Los archivos HTML deben terminar con .html o .htm. Los gráficos web deben etiquetarse de acuerdo con su formato de archivo: .gif, .png, .jpg (.jpeg también es aceptable, aunque menos común), o .svg.
- Nunca uses espacios dentro de los nombres de archivo:
 - Es común usar un carácter de subrayado o un guión para separar visualmente palabras dentro de los nombres de archivo, como robbins_bio.html o robbins-bio.html.
- Evita caracteres especiales como:
 - ?, %, #, /, :, ;, •, etc. Limita los nombres de archivo a letras, números, guiones bajos y guiones. También es mejor evitar los caracteres internacionales, como el sueco å.
- Los nombres de archivo pueden distinguir entre mayúsculas y minúsculas, dependiendo de la configuración del servidor. Usar consistentemente todas las letras en minúscula en nombres de archivo, aunque no es obligatorio, es una buena forma de hacer que tus nombres de archivo sean más fáciles de administrar.
- Mantenga los nombres de archivo tan cortos como sea posible.

Fichero inicial

- La configuración del servidor permite determinar el fichero que se enviará cuando se recibe una petición genérica (del tipo *http://www.dominio.com*)
 - index.html
 - index.xhtml
 - index.php

Lenguajes de marcado

- La idea básica es que un documento está formado por un conjunto de **elementos**
 - El tipo de elemento se identifica mediante marcas o **etiquetas** (*tags*)
 - Con cada etiqueta se pueden asociar varios **atributos**
 - Así se puede especificar características de formato, de tipo de información, etc. que pueda ser procesada por un programa (por ejemplo, por un navegador)
- Ejemplos:
<oscuro>Este texto va en negrita</oscuro>
→ **Este texto va en negrita**
Esta palabra es <importante estilo="cursiva">fundamental</importante>
→ Esta palabra es *fundamental*
- En XML, los elementos (etiquetas, atributos) de un tipo de documentos se especifican en ficheros **DTD** (*Document Type Definition*)
 - Los elementos válidos en un documento XHTML están especificadas en el fichero <http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd>

Etiquetas

- Las etiquetas van emparejadas:
`<etiqueta> texto </etiqueta>`
`<h1>Introducción</h1>`
- Algunas etiquetas pueden ir sin emparejar:
`<hr />`
`<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />`
- HTML permite que las etiquetas sin emparejar no lleven cierre (/)
`<hr>`
`
`
- En HTML las etiquetas se pueden escribir en mayúsculas o minúsculas, indistintamente. Se recomienda escribir todas las etiquetas en minúsculas.
- En XHTML se tienen que escribir siempre en minúsculas

HTML/XHTML

Atributos

- Un elemento puede tener una etiqueta con atributos, separados por espacios, y que se especifican como pares atributo="valor".

`<etiqueta atributo="valor"> texto </etiqueta>`

- En XHTML el valor siempre tiene que ir entre comillas (en HTML no es obligatorio)
- Los atributos permiten añadir información adicional
 - Necesaria
 - Por ejemplo, un hiper-enlace:
`Universidad Complutense Madrid`
 - Opcional
 - Por ejemplo, formato:
`<p align="center">Centrado</p>`

HTML/XHTML

Atributos

- Los atributos van después del nombre del elemento sólo en la etiqueta de apertura, nunca en la etiqueta de etiqueta de cierre.
- Puede haber varios atributos aplicados a un elemento, separados por espacios en la etiqueta de apertura. Su orden no es importante.
- La mayoría de los atributos toman valores, estos valores se presentan en formato **nombre atributo = "valor"**.
- En HTML algunos valores de atributos son palabras descriptivas únicas. Por ejemplo, el atributo atributo **checked**, que hace que una casilla de verificación de un formulario esté marcada. **checked** es equivalente a **checked="checked"**. Es posible que oiga llamar a este tipo de atributo atributo booleano porque describe una característica que está activada o desactivada.
- El valor de un atributo puede ser un número, una palabra, una cadena de texto, una URL o una medida (px, cm, em, pt, ...), dependiendo del propósito del atributo.

HTML/XHTML

Atributos

- El uso de comillas dobles para los valores de los atributos es una buena convención, pero tenga en cuenta que las comillas no son obligatorias y pueden omitirse.
- Además, las comillas simples o dobles son aceptadas siempre que las comillas de apertura y cierre coincidan. Tenga en cuenta que las comillas en los archivos HTML deben ser las comillas rectas(") .
- Los nombres y valores de los atributos disponibles para cada elemento se definen en las especificaciones HTML; en otras palabras, no se puede inventar un atributo para un elemento.
- Algunos atributos son obligatorios, como los atributos src y alt en el elemento img. La especificación HTML también define qué atributos son obligatorios para que el documento sea válido.

HTML/XHTML

Documento XHTML

Opcional, declaración XML que indica el juego de caracteres del documento

```
<?xml version="1.0" encoding="utf-8"?>
```

Tipo de documento y URL de la DTD (definición de tipo de documento) que especifica la versión del lenguaje de etiquetas utilizado en el documento

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

El atributo xmlns especifica el espacio de nombres del documento

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
```

```
<title>Título del documento</title>
```

```
<meta name="keywords" contents="ejemplo xhtml, xhtml mínimo" />
```

```
</head>
```

Las etiquetas <meta /> proporcionan información sobre el documento a los programas que analicen la página

```
<body>
```

```
<p>Ejemplo de documento XHTML</p>
```

```
</body>
```

```
</html>
```

Documento HTML5

Declaración que indica que el documento es HTML5

```
<!DOCTYPE html>
```

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

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
```

```
<title>Título del documento</title>
```

```
</head>
```

```
<body>
```

```
<p>Ejemplo de documento HTML5</p>
```

```
</body>
```

```
</html>
```

Atributos básicos

- Se pueden asociar a cualquier etiqueta
 - **id** = "texto"
 - Asigna un identificador único al elemento dentro de la página
 - Evitar el atributo **name** por estar obsoleto a partir de XHTML 1.0
 - **class** = "texto"
 - Asocia una clase CSS que define un conjunto de estilos de formato para el elemento
 - **style** = "texto"
 - Establece de forma directa los estilos CSS del elemento
 - **title** = "texto"
 - Asigna un título a un elemento
 - Es útil para mejorar la accesibilidad ya que los navegadores lo muestran cuando el usuario pasa el ratón por encima del elemento

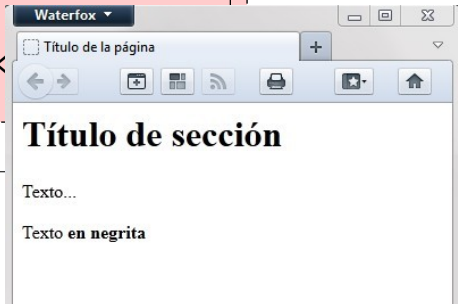
HTML/XHTML

Documentos HTML

- Todo documento HTML comienza por la etiqueta **<HTML>**
 - Y acaba con **</HTML>**
- Todo documento HTML tiene dos partes:

	<code><html></code>
Cabecera	<code><head></code> <code><title>Título de la página</title></code> <code></head></code>
Cuerpo	<code><body></code> <code><h1>Título de sección</h1></code> <code><p>Texto...</p></code> <code><p>Texto en negrita</p></code> <code></body></code>
	<code></html></code>

Resultado ->



Etiquetas de cabecera

La etiqueta **head** es responsable, entre otras cosas, de definir:

- El juego de caracteres en el que está escrito el documento, y que en el caso del idioma español será utf-8 a través de una etiqueta tipo meta **charset**
- El título, a través de la etiqueta **title**
- La descripción de nuestro documento, a través de la etiqueta meta **description**
- Las palabras clave de nuestro documento, separadas por comas a través de la etiqueta meta **keywords**
- Enlaces a archivos **CSS** y de fuentes, a través de etiquetas tipo **link**, que veremos más adelante
- Código **Javascript**, a través de la etiqueta **script** y que también estudiaremos más adelante
- Y muchas otras cosas...

Etiquetas de cabecera

- **Metadatos <meta />**
- Meta-información sobre el documento (información procesable automáticamente por programas que analicen la página)
 - En la descripción o meta **description** podemos describir brevemente el contenido de la página web.
`<meta name="description" content="Web venta productos bricolaje"/>`
 - Definición del juego de caracteres empleado en la web. Así se evita, por ejemplo, que la ñ o las tildes no se muestren correctamente
`<meta charset="utf-8"/>`
 - Con esta etiqueta meta los administradores tienen la posibilidad de definir palabras clave para los motores de búsqueda como Google.
`<meta name="keywords" content="palabra 1, palabra 2, palabra clave 3"/>`
 - Estos dos meta tags, permiten hacer referencia al diseñador de una página web y al propietario de los derechos del código fuente de una página HTML
`<meta name="author" content="Juan Pavón" />`
`<meta name="copyright" content="Propietario del copyright" />`

Etiquetas de cabecera

■ Metadatos <meta http-equiv/>

El atributo **http-equiv** sirve para indicar información sobre cómo debe interaccionar el navegador con el servidor, contiene información equivalente a las indicaciones del encabezado HTTP.

- Para impedir el uso de la caché se emplea el meta tag "**cache-control**" con el valor **no-cache**:

```
<meta http-equiv="cache-control" content="no-cache"/>
```

- Es posible asignar a las páginas HTML la fecha de caducidad que se deseen. Una vez se supera esta fecha, el navegador tiene que cargar el documento HTML desde la página de origen

```
<meta http-equiv="expires" content="43200"/>
```

```
<meta http-equiv="expires" content="Sat, 07 Feb 2016 00:00:00 GMT"/>
```

- Con este atributo también se define una fecha de vencimiento, a partir de la cual los navegadores o los motores de búsqueda serán dirigidos a una URL establecida previamente. En este ejemplo, en 10 segundos se produce el reenvío a la página principal de la W3C

```
<meta http-equiv="refresh" content="10; url=https://www.w3.org/">
```

Etiquetas de cabecera

■ <head>

- Delimita la cabecera del documento
- En la cabecera se describe información del documento (título, meta-información, scripts, estilos)

■ <title>

- Indica el título del documento
- Es obligatoria y tiene que aparecer una y solo una vez en el documento
- El navegador lo visualiza en la barra de título de su ventana

■ <link />

- Enlace a otros archivos (generalmente hojas de estilo)

```
<link rel="stylesheet" type="text/css" href="/css/formato.css" />
```

■ <base />

- Indica una URI que sirve de referencia para direcciones relativas

```
<base href="https://www.w3schools.com/">
```

```
<a href="tags/tag_base.asp">HTML base Tag</a>
```

Etiquetas de cabecera

- **<script>**
 - Permite incluir código de script
 - Aunque por defecto se toma JavaScript, conviene especificarlo
- **<noscript >**
 - Indica un mensaje a mostrar en navegadores que no pueden procesar la etiqueta <script>

```
<script type="text/javascript">
  // Código JavaScript
</script>

<noscript>
  <p>Esta página requiere el uso de JavaScript.
  Por favor compruebe la configuración de su navegador.</p>
</noscript>
```

HTML/XHTML

Atributo style

- **Atributo style**
 - Permite definir propiedades de apariencia que se aplicarán al contenido de la etiqueta donde se define.
 - Lo ideal es separar el contenido y estructura de la página (código HTML) de como se debe mostrar (hojas CSS). Se sigue utilizando bastante.
 - Algunas propiedades:
 - Fuente del texto: **font-family**: fuente (Arial, Times, Verdana, etc.)
 - Tamaño del texto (en pt, px, cm, em,...): **font-size**: 10pt, 12px, ...
 - Color del texto: **color**: rgb(red,green,blue) o #RRGGBB
 - Color de fondo: **background-color**: rgb(red,green,blue) o #RRGGBB
 - Color de borde: **border-color**: rgb(red,green,blue) o #RRGGBB
 - Colores en HTML
 - r,g,b: cantidad de rojo, verde y azul, valores entre 0-255 cada uno.
 - RR,GG,BB: en hexadecimal, valores entre 00-FF cada uno.
 - Link buscar colores por código
<https://html-color-codes.info/codigos-de-colores-hexadecimales>
 - Link nombres de colores aceptados:
<https://www.cdmon.com/es/apps/tabla-colores>

HTML/XHTML

Comentarios

- Texto que ignora el navegador

`<!-- texto del comentario (una o más líneas) -->`

- Los comentarios pueden estar insertados en cualquier lugar de la página web.
- Normalmente se usan para:
 - Marcar el comienzo y el final de las secciones de las páginas

```
<!-- Inicio de las noticias -->
<div id="noticias"> ... </div>
<!-- Fin de las noticias -->
```
 - Incluir notas para otros diseñadores

```
<!-- Esto se puede mejorar -->
```
 - Incluir explicaciones sobre el código de la página

```
<!-- Script para identificar las preferencias del usuario -->
```

Codificación de caracteres

- Todos los caracteres se pueden especificar con su código precedido del símbolo **&** acabando con punto y coma (**;**)
- El juego de caracteres Unicode se puede consultar en <http://www.unicode.org/charts/>
- Caracteres especiales
 - **<** <
 - **>** >
 - **&** &
 - **"** "
 - ** ** (espacio en blanco)
 - **'** '
 - **–** -
- Lista de caracteres especiales en HTML:
http://en.wikipedia.org/wiki/List_of_XML_and_HTML_character_entity_references
- Se puede emplear el código **&#{número unicode};**
 - **@** se muestra el carácter @
 - **∑** se muestra el carácter Σ

Codificación de caracteres

- Acentos y letras especiales:
 - Lo mejor es declarar el uso de caracteres UTF-8 para que se vean bien los acentos y letras como la ñ.
 - Hoy día la mayoría de los navegadores interpretan bien **Utf-8**
- En ocasiones, sin embargo, algunas herramientas no lo interpretan bien, y convierten el texto a ISO-8859 y habría problemas.
- Para evitar problemas es recomendable usar codificación **utf-8**:
 - € €
 - ñ ñ
 - á á
 - é é
 - í í
 - ó ó
 - ú ú
 - ü ü
 - Ñ Ñ
 - Á Á
 - É É
 - Í Í
 - Ó Ó
 - Ú Ú
 - Ü Ü

HTML/XHTML

Lo que el navegador ignora

- Múltiples espacios en blanco: Cuando un navegador encuentra más de un espacio en blanco consecutivo, muestra un solo espacio.
 - Por lo tanto, si el documento contiene:
long, long ago
 - el navegador muestra:
long, long ago
- Saltos de línea (retornos de carro).
 - Los navegadores convierten los saltos de línea a espacios en blanco, por lo que siguiendo la regla anterior de ignorar múltiples espacios en blanco, los saltos de línea no tienen efecto sobre el formato de la página.
- Tabulador
 - La introducción de tabuladores también se convierten en espacios en blanco, ¿así que adivina qué? No sirven para sangrar texto en la página web (aunque pueden hacer que su código sea más legible).
- Comentarios
 - Los navegadores no muestran el texto entre las etiquetas especiales
 - <!-- y --> utilizadas para indicar un comentario.

HTML/XHTML

Estilo HTML

■ Algunas propiedades

- Ajuste del texto: **text-align**: ajuste (left, right, center o justify)
- Indentado de la primera línea: **text-indent**: 5px
- Margen izquierdo: **margin-left**: 10px
- Margen derecho: **margin-right**: 3cm
- Margen arriba y abajo: **margin-top** y **margin-bottom** respectivamente
- Tipo de borde: **border-style**: los valores disponibles son: solid, dotted, dashed, double, etc.
- Altura elemento: **height**: para definir altura, su valor puede ser una medida con su unidad de medida, por ejemplo: 10px, 12em, 100%.
- Anchura element: **width**: define el ancho y su valor puede ser cualquier cantidad con su respectiva unidad de medida.

HTML/XHTML

Estilo HTML

```
<p style="font-size: 14px; font-family: Lato; color: #1d1778; background-color: yellow; padding: 12px; margin: 10px; width: 100px; border-style: dotted; border-color: #000000; border-width: 5px;"> Este es un párrafo con estilo, gracias al atributo style...</p>
```



HTML/XHTML

Texto

- El texto de un documento debe estar contenido por alguna de las etiquetas de una de las tres categorías siguientes:
 - **Etiquetas de bloque:** no necesitan estar contenidas dentro de ninguna etiqueta (salvo `<body>` y `<html>`)
 - `<title>`
 - `<p>` `<pre>` `<h1>` `<h2>` `<h3>` `<h4>` `<h5>` `<h6>` `<address>`
 - **Etiquetas en-línea:** se aplican a una parte de texto dentro de un bloque
 - Salto de línea `
`
 - Tipo de información `<abbr>` `<cite>` `<code>` `` `<kbd>` ``
 - Elemento carácter `` `<i>` `<small>` `<sub>` `<sup>`
 - **Etiquetas de contenedores de texto:** sirven para estructurar el texto y pueden contener en su interior etiquetas de bloque
 - Bloque de cita: `<blockquote>`
 - División `<div>`, `<article>`, `<header>`, `<section>`, `<aside>`, `<nav>`, `<footer>`

HTML/XHTML

Etiquetas de bloque – Párrafos

- **`<p>`**
 - Contiene el texto de un párrafo
 - El texto del párrafo se muestra con el formato por defecto del navegador
 - El navegador no muestra los espacios en blanco ni los saltos de línea del código fuente
 - Entre párrafos se pone una línea en blanco (y solo una)
 - Si un párrafo `<p>` no contiene nada, no se muestra (salvo que la hoja de estilo incluya bordes o márgenes)
- **`<pre>`**
 - Texto preformateado: igual que `<p>` pero se tienen en cuenta espacios en blanco y líneas en blanco
 - Se usa un tipo de letra de ancho fijo

HTML/XHTML

Etiquetas de bloque – Secciones

- **<h1> <h2> <h3> <h4> <h5> <h6>**

- Encabezado (de nivel 1 a 6)

```
<h1>Sección 1</h1>
<h2>Sección 1.1</h2>
<p>Texto normal.</p>
<h3>Sección 1.1.1</h3>
<p>Texto normal.</p>
...
```

- **<address>**

- Información de contacto del autor de la página

```
<address>Autor: Juan Pavón (UCM) <br /> Última
modificación: 10 de enero de 2013
</address>
```

HTML/XHTML

Etiquetas en línea

- **
**

- Salto de línea (sin dejar una en blanco entre medias)

- ** **

- Espacio en blanco
- Cuando se usa la etiqueta <pre> se mantienen los espacios en blanco

- **<hr />**

- Línea horizontal (*horizontal rule*)
- Se usa cada vez menos ya que con CSS se pueden crear mejor los bordes

HTML/XHTML

Etiquetas en línea

■ Importancia del texto

■ ****

- Para resaltar una porción de texto dándole énfasis (aparecerá en cursiva)
- Más recomendable que <i>

■ ****

- Mayor énfasis que con (aparecerá en negrita)
- Más recomendable que

Etiquetas en línea

■ Tipo de informaci3

■ **<abbr>**

- Abreviatura

`<abbr title="etc3tera">etc.</abbr>`

`<abbr title="Universidad Complutense de Madrid">UCM</abbr>`

■ **<cite>**

- Para incluir una referencia o cita (aparecerá en cursiva)

`<p>Como dice el refrán,`

`<cite>A la tercera va la vencida</cite>.</p>`

■ **<dfn>**

- Definici3n de un t3rmino (aparecerá en cursiva)

- `<p>`

El `<dfn>HTML</dfn>` es un lenguaje de marcado para hipertextos.

`</p>`

Etiquetas en línea

■ Tipo de información

■ **<code>**

- Parecido a <pre> (pero <code> es un elemento de línea y <pre> un elemento de bloque)
- Fragmento de código de programa (aparecerá en letra monospace)

```
<code>main() { printf ("Hola, mundo"); }</code>
```

■ **<samp>**salida producida por un programa</samp>

■ **<kbd>**texto a introducir por el usuario</kbd>

HTML/XHTML

Etiquetas en línea

■ Para indicar edición del texto

■ ****

- Para mostrar que se elimina un texto (aparecerá tachado)

■ **<ins>**

- Para mostrar que se ha insertado un texto (aparecerá subrayado)

```
<p>La nota final es <del>suspenseo</del>  
<ins datetime="20130625">aprobado</ins>.</p>
```

→ La nota final es ~~suspenseo~~ aprobado.

HTML/XHTML

Etiquetas en línea

■ Elemento carácter

- Estas etiquetas están en su mayoría desaconsejadas, ya que pueden conseguirse sus efectos utilizando las propiedades correspondientes de las hojas de estilo
 - **** para etiquetar texto en **negrita**: Texto en `negrita`
 - **<i>** para etiquetar texto en *itálica*: Texto en `<i>itálica</i>`
 - **<small>** para etiquetar texto de menor tamaño
 - **<sub>** para identificar texto en subíndice: `H₂O`
→ H₂O
 - **<sup>** para identificar texto en superíndice: `xⁿ` → xⁿ
- Estas etiquetas existían en HTML, pero han sido eliminadas de HTML5.
 - **<big>** para etiquetar texto de mayor tamaño
 - **<tt>** para etiquetar texto de fuente de espaciado fijo

HTML/XHTML

Etiquetas de contenedores de texto

■ **<blockquote>**

- Para incluir una cita larga, que puede contener varios párrafos u otras etiquetas
- Se suele mostrar el texto dentro de esta etiqueta con márgenes a izquierda y derecha
- Lo mejor es definir el formato deseado en la hoja de estilo
- Se puede indicar el origen de la cita con el atributo **cite**

```
<p>El candidato a la presidencia fue locuaz en su discurso:</p>
<blockquote cite="http://www.buenasnoticias.org/entrevista12.html">
  <p>Prometo que voy a respetar los servicios sociales, naturalmente.
    Miente quien infunda alguna sospecha al respecto.
  </p>
  <p>Y bla bla bla...</p>
</blockquote>
```

HTML/XHTML