

# Tecnologías Web para la presentación HTML(5)/XHTML

## Aplicaciones Web/Sistemas Web



Juan Pavón Mestras  
Dep. Ingeniería del Software e Inteligencia Artificial  
Facultad de Informática  
Universidad Complutense Madrid

Material bajo licencia Creative Commons



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



## 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 vs. XHTML vs. HTML5

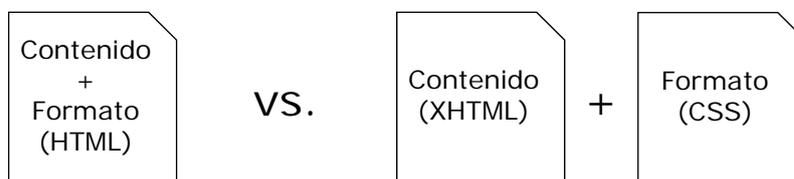
---

- **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 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 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 simples para definir la estructura de un documento
  - Por ejemplo: todos los elementos tienen que estar bien anidados y tener siempre la etiqueta de cierre
  - Menos recursos para procesar los documentos
    - Ventaja para hacer navegadores más eficientes

## 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 (a fecha de hoy)
  - Un paso más
    - Construye sobre 'strict'; acaba con 'frameset' y 'transitional'
    - "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 ha funcionado).

## 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 está en proceso de estandarización
  - **Publicación oficial en 2014**
  - Casi todos los navegadores soportan casi todas sus características
    - Pero no todos: <http://www.caniuse.com>

## Ejercicio

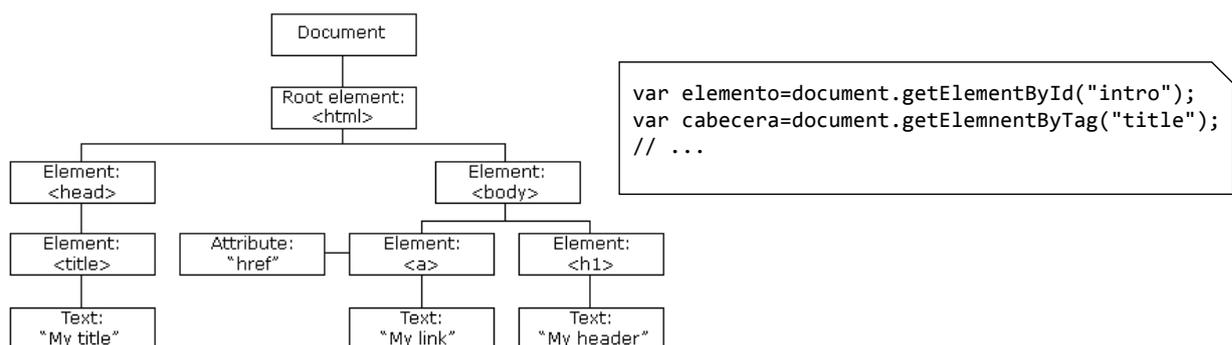
---

- ¿Cuál será el mejor lenguaje de marcado para móviles?
  - ¿Qué características debería tener?
  - ¿Qué opciones se están barajando hoy día?

## 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



## 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

## Herramientas para crear una página Web

---

- Recomendable
  - Editor de páginas Web
    - Amaya: editor de referencia del W3C – <http://www.w3.org/Amaya/>
    - Gratuitos: Netbeans, HTML-Kit, Kompozer, Nvu
    - Comerciales: Adobe Dreamweaver, Microsoft Expression Web
  - Navegadores Web (varias versiones, varios SOs)
    - Chrome, Firefox, Internet Explorer, Safari, Opera
      - Tienen buenas herramientas para ayuda a desarrolladores
    - Considerar navegadores en móviles (iOS, Android)
  - Editores gráficos y de imágenes
    - GIMP, Adobe Photoshop
  - Herramienta de desarrollo para código de scripts o del servidor
    - Eclipse, Netbeans, Aptana Studio 3
  - Servidor Web
    - El que se vaya a utilizar
    - Se puede usar Apache sobre Windows para desarrollo pero es recomendable sobre Linux para implantación por mejor seguridad

# Eclipse IDE for Java EE Developers

---

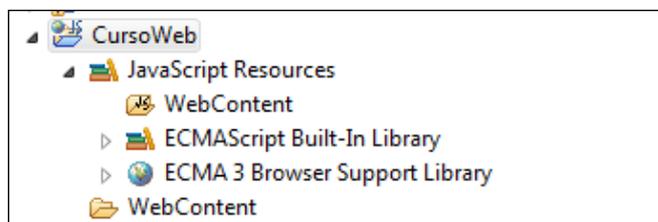
- Entorno de desarrollo de software: <http://www.eclipse.org/>
  - Descargar la versión *Eclipse IDE for Java EE Developers*
- Windows, Linux, Mac OS X
- Muchos plugins
  - Java EE
  - Git
  - HTML 5
  - JavaScript
  - Maven
  - etc.

## Creación de páginas web estáticas

---

### (1) Crear un proyecto

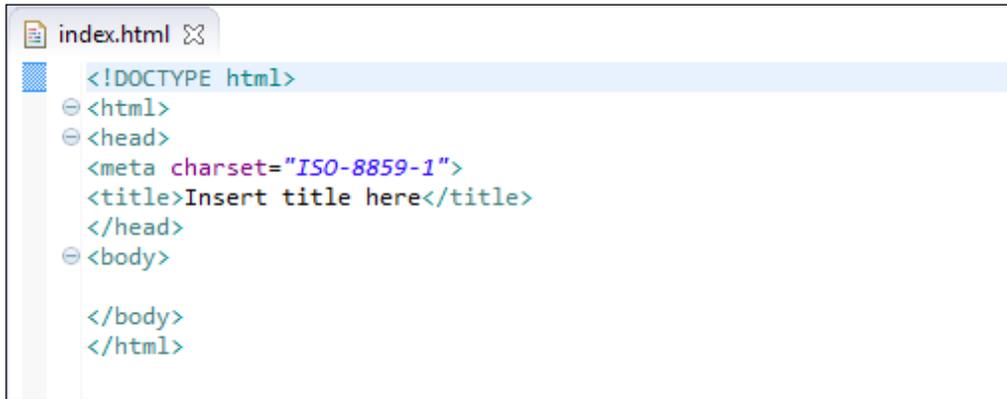
- File → New → Project...
  - Seleccionar Web → Static Web Project
  - Indicar un nombre
  - Al crear el proyecto web eclipse solicita cambiar la perspectiva



## Creación de páginas web estáticas

### (2) Crear una página HTML

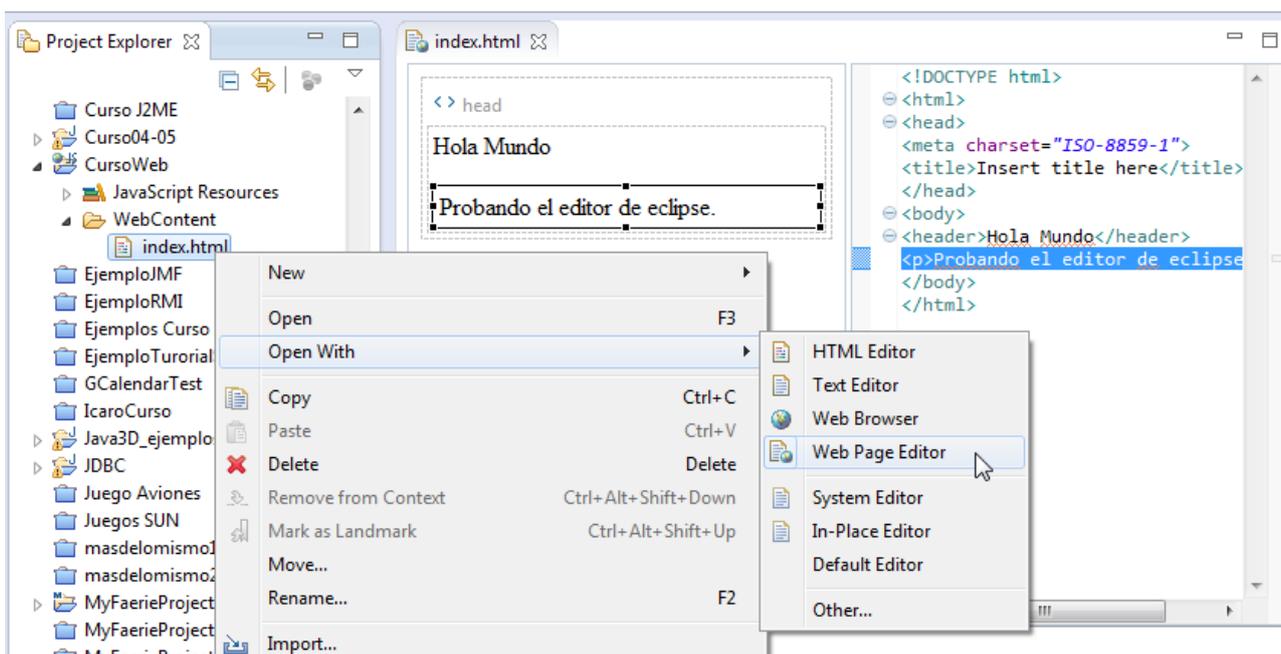
- En WebContent seleccionar New → HTML File
  - Seleccionar Web → Static Web Project
  - Indicar un nombre (por ejemplo, index.html)
  - A continuación seleccionar la plantilla New HTML File (5)



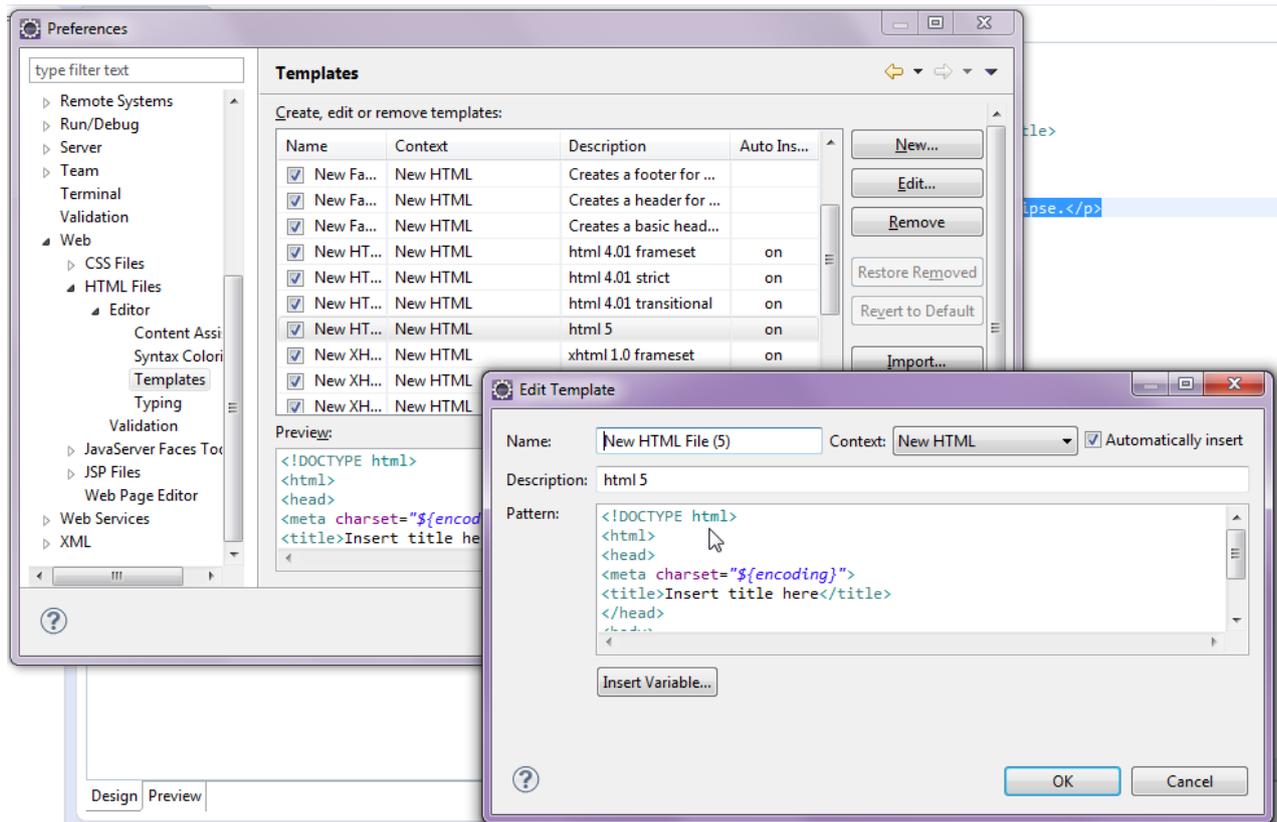
```
index.html
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>

</body>
</html>
```

## Varias opciones de editor



## Plantillas de páginas HTML5



## Ejercicio: Instalar Eclipse IDE for Java EE

<http://www.eclipse.org>

- Una vez instalado, probar a crear una página XHTML que visualice el típico "Hola Mundo"
- Validar la página en <http://validator.w3.org/>

## 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

## 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

## De las pruebas a la implantación del sitio Web

---

- Las pruebas se realizan en un entorno de desarrollo local
  - Herramientas como DreamWeaver proporcionan un análisis bastante exhaustivo del código
  - También se puede validar la corrección del documento en <http://validator.w3.org/>
  - Firefox tiene el complemento HTML Validator <https://addons.mozilla.org/es/firefox/addon/html-validator/>
- La implantación debe ser sencilla: subir los ficheros al servidor
  - Requiere previamente la configuración del servidor
  - Subir contenidos nuevos/modificados
    - Normalmente la herramienta de desarrollo automatiza este proceso
    - Se suele utilizar una conexión ftp (segura preferiblemente)
- Administración del sitio Web
  - Configuración
  - Monitorización
  - Estadísticas de uso

## Alojamiento del sitio Web (*Web hosting*)

---

- En un servidor particular
  - Requiere su instalación, configuración y mantenimiento
    - Total flexibilidad
  - Tiene que estar siempre disponible en internet
    - Tanto el ordenador como la conexión (p.ej., ADSL)
  - Requiere contratar un dominio
    - Hay varios proveedores: OVH,
    - Gestionar la IP dinámica: lo suele ofrecer el mismo gestor del dominio
- En un servidor externo
  - De pago
  - Gratuitos (suelen llevar asociados publicidad)
    - Básico
    - Con PHP y base de datos
    - Con WordPress u otros gestores de contenidos
    - Blogs
  - En la nube (*cloud hosting*)
    - Utiliza numerosos servidores en lugar de sólo uno

## Ejercicio

---

- Seleccionar y comentar en el campus virtual las mejores ofertas para montar un sitio Web
  - Definir criterios para una selección adecuada
    - Fiabilidad
    - Facilidad de uso
    - Herramientas disponibles
    - Lenguajes soportados por el servidor (PHP, J2EE, etc.)
    - Orientación (webs personales, profesionales, empresariales)
    - Precio
- Para la entrega de las prácticas habrá que montarlas en un sitio y pasar la URL correspondiente al profesor
  - Decidir el servidor donde se instalarán las prácticas que sea accesible a través de internet

## 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=iso-8859-1" />
  - HTML permite que las etiquetas sin emparejar no lleven cierre (/)  
<hr>
- En HTML las etiquetas se pueden escribir en mayúsculas o minúsculas, indistintamente
- En XHTML se tienen que escribir siempre en minúsculas

## 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:  
<a href="http://www.ucm.es/">Universidad Complutense Madrid</a>
  - Opcional
    - Por ejemplo, formato:  
<p align="center">Centrado</p>

## 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

## Documentos HTML

---

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

<b>Cabecera</b>	<pre>&lt;html&gt; &lt;head&gt;   &lt;title&gt;Título de la página&lt;/title&gt; &lt;/head&gt;</pre>
<b>Cuerpo</b>	<pre>&lt;body&gt;   &lt;h1&gt;Título de sección&lt;/h1&gt;   &lt;p&gt;Texto...&lt;/p&gt;   &lt;p&gt;Texto &lt;b&gt;en negrita&lt;/b&gt; &lt;/body&gt;</pre>
	<pre>&lt;/html&gt;</pre>

Resultado ->



## Documento XHTML

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Título del documento</title>
<meta name="keywords" contents="ejemplo xhtml, xhtml mínimo" />
</head>
<body>
<p>Ejemplo de documento XHTML</p>
</body>
</html>
```

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

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

El atributo xmlns especifica el espacio de nombres del documento

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

## Documento HTML5

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Título del documento</title>
</head>
<body>
<p>Ejemplo de documento HTML5</p>
</body>
</html>
```

Declaración que indica que el documento es HTML5

## 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 **&**
  - El juego de caracteres Unicode se puede consultar en <http://www.unicode.org/charts/>
- Caracteres especiales
  - **&lt;**; <
  - **&gt;**; >
  - **&amp;**; &
  - **&quot;**; "
  - **&nbsp;**; (espacio en blanco)
  - **&apos;**; '
  - **&ndash;**; –
- Lista de 256 caracteres especiales en HTML:  
[http://en.wikipedia.org/wiki/List\\_of\\_XML\\_and\\_HTML\\_character\\_entity\\_references](http://en.wikipedia.org/wiki/List_of_XML_and_HTML_character_entity_references)

## 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:
    - &euro; €
    - &ntilde; ñ                      &Ntilde; Ñ
    - &aacute; á                        &Aacute; Á
    - &eacute; é                        &Eacute; É
    - &iacute; í                        &Iacute; Í
    - &oacute; ó                      &Oacute; Ó
    - &uacute; ú                      &Uacute; Ú
    - &uuml; ü                      &Uuml; Ü

## Codificación de caracteres en los hiperenlaces

---

Carácter	Código
/	%2F
:	%3A
=	%3D
"	%22
'	%60
(espacio)	%20
?	%3F
@	%40
&	%26
\	%5C
~	%7E
	%7C

## 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

## Etiquetas de cabecera

---

- **Metadatos**
  - **<meta />**
    - Metainformación sobre el documento (información procesable automáticamente por programas que analicen la página)
    - Normalmente se usan los atributos **name** (para definir un tipo de metadato) y **content** (para definir el valor)
      - Algunos habituales:  
`<meta name="author" content="Juan Pavón" />`  
`<meta name="robots" content="index, follow" />`  
`<meta name="keywords" content="sistemas web, html" />`  
`<meta name="description" content="Curso sobre diseño web" />`
    - El atributo **http-equiv** sirve para indicar información sobre cómo interactuar el navegador con el servidor
      - Número de segundos para refrescar la página y cual recargar  
`<meta http-equiv="refresh" content="20;URL=./indice.htm" />`
      - Fecha (formato GMT) en que expirará la página (y deja de ser válida en el caché)
        - Si se pone 0, el navegador tendrá que recargarla cada vez que se visite  
`<meta http-equiv="expires" content="Sat, 16 Nov 2002 16:05:00 GMT" >`

## Etiquetas de cabecera

---

- **<script>**
  - Permite incluir código de script
  - Aunque por defecto se toma JavaScript, conviene especificarlo
  - Puede referirse a un fichero externo  
`<script type="text/javascript" src="http://www.ejemplo.com/js/miscript.js"></script>`
- **<noscript >**
  - Indica un mensaje a mostrar en navegadores que no pueden procesar `<script>`

```
<script type="text/javascript">
  //
  // Código JavaScript
  //]]&gt;
&lt;/script&gt;
&lt;noscript&gt;&lt;p&gt;Esta página requiere el uso de JavaScript.
Por favor compruebe la configuración de su navegador.&lt;/p&gt; &lt;/noscript&gt;</pre></div><div data-bbox="108 451 258 462" data-label="Page-Footer"><p>Juan Pavón - UCM 2012-13</p></div><div data-bbox="464 449 539 460" data-label="Page-Footer"><p>HTML/XHTML</p></div><div data-bbox="865 449 889 460" data-label="Page-Footer"><p>37</p></div><div data-bbox="142 557 456 579" data-label="Section-Header"><h2>Etiquetas de cabecera</h2><hr/></div><div data-bbox="150 602 869 842" data-label="List-Group"><ul><li>■ <b>&lt;style&gt;</b><ul><li>■ Permite definir propiedades que se aplicarán a lo largo del documento</li><li>■ Generalmente se indica la hoja de estilo que se va a aplicar</li><li>■ O propiedades del documento:<ul><li>• Fuente del texto: <b>font-family</b>: fuente (Arial, Times, Verdana, etc.)</li><li>• Tamaño del texto (en puntos –pt–): <b>font-size</b>: 10pt, 12pt, ...</li><li>• Color del texto: <b>color</b>: rgb(r,v,a) o #RRVVAA<ul><li>• r,v,a: cantidad de rojo, verde y azul, 0..255 cada uno</li><li>• RR,VV,AA: en hexadecimal, 00..FF cada uno</li></ul></li><li>• Ajuste del texto: <b>text-align</b>: ajuste (left, right, center o justify)</li><li>• Indentado de la primera línea: <b>text-indent</b>: espacio</li><li>• Margen izquierdo: <b>margin-left</b>: espacio</li><li>• Margen derecho: <b>margin-right</b>: espacio</li></ul></li></ul></li></ul></div><div data-bbox="205 866 829 936" data-label="Text"><pre>&lt;style type="text/css"&gt;
P { font-family: Arial; font-size: 10pt; color: rgb(0,0,128);
text-indent: 15px; text-align: justify; margin-left: 10px }
&lt;/style&gt;</pre></div><div data-bbox="108 950 258 962" data-label="Page-Footer"><p>Juan Pavón - UCM 2012-13</p></div><div data-bbox="464 948 539 959" data-label="Page-Footer"><p>HTML/XHTML</p></div><div data-bbox="865 948 890 959" data-label="Page-Footer"><p>38</p></div>
```

## 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 <br />
    - Tipo de información <abbr> <cite> <code> <em> <kbd> <strong>
    - Elemento carácter <b> <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>

## 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)
  - En XHTML el estilo del párrafo (alineación, tipo de letra, etc.) se debe declarar en los ficheros de estilo
    - En HTML se permite en la propia etiqueta, pero no es aconsejable

```
<p align="center">Centrado</p>
<p align="right">Alineado a la derecha</p>
```
- **<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

## 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>
```

## Etiquetas en línea

---

- **<br />**
  - Salto de línea (sin dejar una en blanco entre medias)
- **&nbsp;**
  - 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

## Etiquetas en línea

---

### ■ Importancia del texto

#### ■ `<em>`

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

#### ■ `<strong>`

- Mayor énfasis que con `<em>` (aparecerá en negrita)
- Más recomendable que `<b>`

## Etiquetas en línea

---

### ■ Tipo de información

#### ■ `<abbr>`

- Abreviatura

```
<abbr title="etcétera">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>`

- Definición de un término (aparecerá en cursiva)

## 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>`

## Etiquetas en línea

---

### ■ Para indicar edición del texto

#### ■ `<del>`

- 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.

## 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
    - **<b>** para etiquetar texto en **negrita**: Texto en `<b>negrita</b>`
    - **<i>** para etiquetar texto en *itálica*: Texto en `<i>itálica</i>`
    - **<small>** para etiquetar texto de menor tamaño
    - **<sub>** para identificar texto en subíndice: `H<sub>2</sub>O` → H<sub>2</sub>O
    - **<sup>** para identificar texto en superíndice: `x<sup>n</sup>` → x<sup>n</sup>
  - 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

## 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>
```

## Estructura de la página

---

- **<div>**
  - División: mecanismo más importante para agrupar diversos elementos de bloque (párrafos, encabezados, listas, tablas, divisiones, etc.)
  - El formato hay que definirlo en una hoja de estilo
  - Una división no puede insertarse dentro de una etiqueta en-línea (<strong>, <em>, etc.) o de un bloque de texto (párrafo <p>, encabezado <h1> ... <h6>, dirección <address>, pre-formateado <pre>, lista, etc)
    - Pero sí puede insertarse dentro de una tabla, de un bloque de cita <blockquote> o de otra división <div>
- Con CSS se puede luego definir la posición de los distintos elementos
  - Ver capítulo de CSS

## Estructura de la página

---

- Estructuras semánticas (**sólo HTML5**)
  - Etiquetas similares a <div> pero con significados más específicos:
    - <header>
    - <nav>
    - <section>
    - <article>
    - <aside>
    - <footer>
    - <details>
- Con CSS se puede luego definir la posición de los distintos elementos de forma semántica
  - Ver capítulo de CSS

## Agrupación de texto

---

### ■ `<span>`

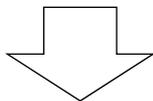
- Se usa para dar formato con la hoja de estilo a un grupo de elementos en línea seguidos dentro de un mismo bloque (por ejemplo, varias palabras seguidas en un párrafo)
- Se suele asociar a clases definidas en un fichero de estilos (.css)

#### Fichero .css

```
span.feroz {
  font-style: italic;
  text-decoration: underline;
}
span.rojo {
  color: red;
}
```

#### Fichero .html (incluye al .css)

```
<p>Andaba sola
<span class="rojo">Caperucita
Roja</span>
y se encontró
<span class="feroz">al lobo
feroz</span></p>
```



Andaba sola **Caperucita Roja** y se encontró *al lobo feroz*

## Hiperenlaces

---

### ■ `<a>`

- Hiperenlace o hipervínculo (hyperlink)  
`<a href="URL">texto asociado al enlace</a>`
- El texto asociado al enlace aparece destacado
- El URL puede constar de
  - Protocolo (http://)
  - Servidor (www.ucm.es)
  - Ruta (/info/cursos)
  - Consulta (?curso=web)
  - Sección (#segunda)`http://www.ucm.es/ info/cursos?curso=web#segunda`
- En las URL pueden usarse caracteres especiales codificados:
  - # %23 (espacio en blanco) %20
  - / %2F ? %3F
  - : %3A @ %40
  - = %3D & %26
  - " %22 \ %5C
  - ' %60 ~ %7E

## Hiperenlaces

---

- La referencia puede ser:
  - A una página de otro sitio web: `http://www.ucm.es`
  - A otra página del mismo sitio web
    - Recomendable que sea una ruta relativa al documento actual:  
`<a href="../img/foto.jpg">Mi foto</a>`
    - Mejor que una ruta absoluta:  
`<a href="c:/web/img/foto.jpg">Mi foto</a>`
  - A un marcador definido en la misma página
    - En algún sitio se tiene que haber definido un marcador:
      - Con el atributo `name` en una etiqueta **a (anchor)**:  
`<a name="marcador">Nota</a>`
      - O con el atributo `id`:  
`<h2 id="marcador">Nota</h2>`
    - Que luego se puede enlazar:  
`<a href="#marcador">Ir al lugar marcado</a>`  
`<a href="../documento.html#marcador">...</a>`  
`<a href="http://www.abc.es/documento.html#marcador">...</a>`
- Para que el enlace se abra en otra ventana: `target="_blank"`

## Imágenes

---

- `<img />`
  - Inserta una imagen de mapa de bits (de algún archivo aparte)
    - JPEG (Joint Photographic Experts Group): muy buena compresión
    - PNG (Portable Network Graphics): compresión sin pérdida de calidad
    - GIF (Graphics Interchange Format): imágenes animadas, pero menos calidad
  - Atributos
    - **src**: camino absoluto o relativo a la imagen desde la página web
    - **alt**: texto que debe mostrarse si la imagen no está disponible
    - **title**: (opcional) texto que se muestra en forma de "tip" (cuadrado amarillo que aparece cuando se sitúa el ratón encima de la imagen)
    - **width y height**: (opcionales) anchura y altura de la imagen
      - Es conveniente usar estos atributos porque permite al navegador reservar espacio para la imagen mientras se carga

```
<p>El bonito escudo de mi universidad:  
</p>
```

## Ejercicios

---

- Prueba varios tipos de enlaces en una página web
  - Para enviar un mensaje de correo electrónico

```
<a href="mailto:sugerencias@papelera.com">Buzón de sugerencias</a>
```

    - Explora cómo se puede hacer para que el mensaje vaya con un asunto (*subject*) concreto
    - También cómo se podría hacer que el email empezara con un texto de ejemplo
  - Cómo se haría para poner la dirección de correo como una imagen en vez de como texto (y evitar así sucumbir al spam)
  - Para ir al principio de una página

```
<a href="/">Inicio</a>
```
  - Cómo se puede hacer para poner un enlace que permita bajar un fichero mediante ftp
- Más ejemplos en:  
[http://librosweb.es/xhtml/capitulo\\_4/ejemplos\\_de\\_enlaces\\_habituales.html](http://librosweb.es/xhtml/capitulo_4/ejemplos_de_enlaces_habituales.html)

## Listas

---

- Tres tipos de listas:
  - **Listas ordenadas** (`<ol> ... </ol>`)
  - **Listas no ordenadas** (`<ul> ... </ul>`)
    - `<li> ... </li>` delimitan cada elemento de la lista
  - **Listas de definición** (`<dl> ... </dl>`)
    - `<dt> ... </dt>` delimitan los términos
    - `<dd> ... </dd>` delimitan las definiciones
- Las listas se pueden anidar

```
<ol>
  <li>Primer elemento </li>
  <li>Segundo elemento </li>
  <ul>
    <li>Elemento de lista desordenada anidada</li>
  </ul>
  <li>Definiciones: </li>
  <dl>
    <dt>Término</dt>
    <dd>Definición del término 1</dd>
  </dl>
</ol>
```

## Ejercicio

---

- Escribir el código para generar las siguientes listas anidadas:

### Ejercicio de listas anidadas

- Introducción
- Edición de páginas Web
  - [HTML y XHTML](#)
  - [CSS](#)
  - [HTML5](#)
- Programación de páginas web
  1. JavaScript
  2. CGI
  3. PHP
  4. J2EE
- Profesor

*Email*

**nombre@direccion.com**

*Teléfono*

91 123 4567

*Tutorías*

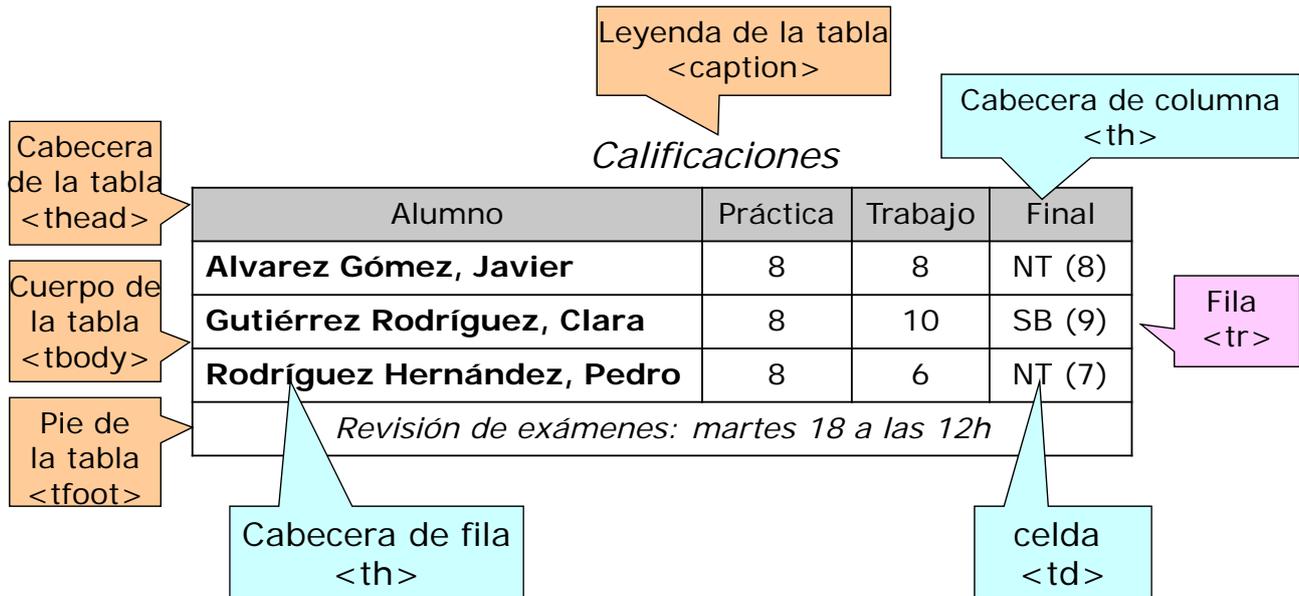
Los martes de 10 a 12

## Tablas

---

- Permiten presentar información tabular, en filas y columnas, con cabeceras
- Cada elemento de la tabla puede ser simple o a su vez ser otra agrupación de filas y de columnas, cabeceras y pies de tabla, subdivisiones, cabeceras múltiples y otros elementos complejos
  - Esto ha dado lugar a que a veces se usan para estructurar la página, lo cual no es en absoluto recomendable

## Tablas – Componentes



## Tablas

- **<table>**
  - Define una tabla
  - Atributos
    - border – en píxeles
    - width – anchura de la tabla en píxeles o en porcentaje del espacio horizontal del elemento padre
    - cellspacing – espacio entre el borde de la celda y su contenido
    - cellpadding – espacio entre celdas o entre celda y borde de tabla
    - summary – aporta información sobre el contenido de la tabla (para dispositivos que no la pueden visualizar, por ejemplo)
- **<caption>**
  - Leyenda de la tabla: texto opcional que se muestra fuera de la tabla (por defecto, arriba)
  - Se suele poner a continuación de <table>
  - No puede incluir párrafos ni otros elementos de bloque, aunque sí etiquetas en línea (<strong>, <em>, etc.)

## Tablas sencillas

---

- Una tabla sencilla se define con
  - La etiqueta **<table>**
  - A continuación se definen las filas, con **<tr>** (*table row*)
  - Y para cada fila, los elementos con **<td>** (*table data cell*)
    - Algunas celdas se usan como cabeceras de fila o columna: **<th>** (*table header*)
      - Aunque se pueden utilizar en cualquier celda
      - Se muestran resaltadas (normalmente, en negrita y centradas en horizontal)
  - Se pueden agrupar celdas
    - En una fila con el atributo **colspan**
    - En una columna con el atributo **rowspan**

## Tablas sencillas

---

```
<table border="1" width="30%">
<caption>Fusión de filas y columnas</caption>
<tr>
  <th colspan=3>Números</th>
</tr>
<tr>
  <th>Nombre</th>
  <th>Valor</th>
  <th>Idioma</th>
</tr>
<tr>
  <td>Uno</td>
  <td>1</td>
  <td rowspan="2">Español</td>
</tr>
<tr>
  <td>Dos</td>
  <td>2</td>
</tr>
</table>
```

Fusión de filas y columnas

Números		
Nombre	Valor	Idioma
Uno	1	Español
Dos	2	

## Tablas complejas

---

- Pueden comenzar con una cabecera (opcional y como mucho una)
  - **<thead>**
- Pueden acabar con un pie (opcional y como mucho uno)
  - **<tfoot>**
    - Se declara antes de los **<tbody>**
- Tienen al menos un cuerpo
  - **<tbody>**
  - Agrupa filas
  - Se pueden agrupar columnas:
    - **<col />** para asignar los mismos atributos a varias columnas
    - **<colgroup>** para agrupar de forma estructural varias columnas
      - El atributo **span** indica el número de columnas

## Ejercicios

---

- Crear un calendario con una tabla
  - Ejemplos:  
[http://www.mclibre.org/consultar/amaya/ejercicios/calendarios\\_1/calendarios\\_1\\_formateado.html](http://www.mclibre.org/consultar/amaya/ejercicios/calendarios_1/calendarios_1_formateado.html)
- Crear tablas como las que aparecen en  
[http://www.mclibre.org/consultar/amaya/ejercicios/tablas\\_1/tablas\\_1.html](http://www.mclibre.org/consultar/amaya/ejercicios/tablas_1/tablas_1.html)
- y en  
[http://librosweb.es/xhtml/capitulo\\_7/tablas\\_basicas.html](http://librosweb.es/xhtml/capitulo_7/tablas_basicas.html)

## Formularios

---

- **<form>**
- Conjunto de controles que permiten al usuario interactuar
  - Generalmente para introducir datos y enviarlos al servidor web
  - El navegador envía únicamente los datos de los controles contenidos en el formulario
  - En una misma página puede haber varios formularios que envíen datos al mismo o a diferentes agentes
- Ejemplo:

```
<form action="http://www.miweb.com/procesaform.php" method="post">
Escribe tu nombre:
<input type="text" name="nombre" value="" />
<br/>
<input type="submit" value="Enviar" />
</form>
```

### Formulario muy sencillo

Escribe tu nombre:

## Formularios

---

- Dentro de un formulario puede haber:
  - Cualquier elemento típico de una página web
    - Párrafos, imágenes, divisiones, listas, tablas, etc.
  - Controles de formularios
    - <input />
    - <button>
    - <select>
    - <optgroup>
    - <option>
    - <textarea>
  - Estructura de formularios
    - <fieldset>
    - <legend>
  - Información para accesibilidad
    - <label> permite mejorar la accesibilidad de los controles
  - Controles avanzados (sólo HTML5)
    - <datalist>, <keygen>, <output>...

## Formularios

---

- Atributos de <form>
  - **action="URL"**: aplicación del servidor que procesará los datos remitidos (por ejemplo, un script de PHP)
  - **method**: método HTTP para enviar los datos al servidor
    - **GET**: como añadido a la dirección indicada en el atributo action
      - Limitado a 500 bytes
      - Los datos enviados se añaden al final de la URL de la página y por tanto se ven en la barra del navegador
      - Se suele usar cuando se envía información que no modifica el servidor (por ejemplo, términos para una búsqueda)
      - Si no se especifica, los navegadores suelen hacer GET
    - **POST**: en forma separada
      - Puede enviar más información
      - Permite enviar ficheros adjuntos
      - Los datos enviados no se ven en la barra del navegador
      - Se suele usar cuando se envía información que puede modificar el servidor
  - **enctype**: Tipo de codificación al enviar el formulario al servidor
    - "application/x-www-form-urlencoded" o "multipart/form-data"
    - Sólo se indica cuando se adjuntan archivos

## Formularios

---

- <input />
  - type = "text | password | checkbox | radio | submit | reset | file | hidden | image | button" - Indica el tipo de control que se incluye en el formulario
  - name = "texto" - Nombre del control (para que el servidor pueda procesar el formulario)
  - value = "texto" - Valor inicial del control
  - size - Tamaño inicial del control (en píxeles, salvo para campos de texto y de password que se refiere al número de caracteres)
  - maxlength = "numero" - Máximo tamaño de texto y de password
  - checked = "checked" - Opción preseleccionada para los controles checkbox y radiobutton
  - disabled = "disabled" - El control aparece deshabilitado y su valor no se envía al servidor junto con el resto de datos
  - readonly = "readonly" - El contenido del control no se puede modificar
  - src = "url" - Para el control que permite crear botones con imágenes, indica la URL de la imagen que se emplea como botón de formulario
  - alt = "texto" - Descripción del control

## Formularios

---

- Cuadro de texto

Nombre <br/>

```
<input type="text" name="nombre" value="" />
```

Nombre

- Se enviará al servidor cuando se pulse un botón de enviar
- El nombre asignado en *name* tiene que concordar con el que se use en la aplicación en el servidor
  - No se deben utilizar caracteres problemáticos en programación (espacios en blanco, acentos y caracteres como ñ o ç)
- *value* permite establecer un valor inicial en el cuadro de texto

- Contraseñas

Contraseña <br/>

```
<input type="password" name="contrasena" value="" />
```

Contraseña

- Igual que el cuadro de texto por el valor introducido no se ve

## Formularios

---

- Cuadro de texto de varias líneas

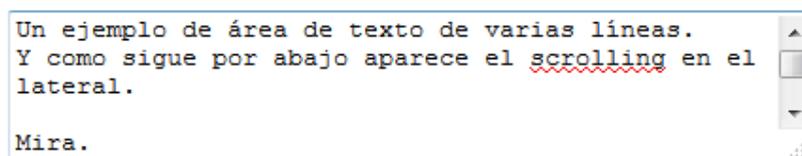
Nombre <br/>

```
<textarea name="nombre" rows="4" cols="50">
```

Contenido inicial del cuadro de texto

```
</textarea>
```

- filas: número de filas visibles (sale una barra de desplazamiento si se hay más)
- columnas: anchura en caracteres



Un ejemplo de área de texto de varias líneas.  
Y como sigue por abajo aparece el scrolling en el lateral.

Mira.

## Formularios

- Botón de envío de formulario

```
<input type="submit" name="enviar" value="Enviar" />
```



- El navegador se encarga de enviar automáticamente los datos cuando el usuario pincha el botón

- Botón de reseteo de formulario

```
<input type="reset" name="borrar" value="Borrar formulario" />
```

- El navegador borra toda la información introducida y muestra el formulario en su estado original

## Formulario de email

```
<h3>Envíanos tus sugerencias por e-mail</h3>
```

```
<form action="MAILTO:sugerencias@tehacemoscaso.com"
      method="post" enctype="text/plain">
```

```
Nombre:<br>
```

```
<input type="text" name="name" value="tu nombre"><br>
```

```
E-mail:<br>
```

```
<input type="text" name="mail" value="tu email"><br>
```

```
Sugerencia:<br>
```

```
<input type="text" name="comment" value="Tus sugerencias" size="50"><br><br>
```

```
<input type="submit" value="Enviar">
```

```
<input type="reset" value="Borrar">
```

```
</form>
```

### Envíanos tus sugerencias por e-mail

Nombre:

E-mail:

Sugerencia:

## Formularios

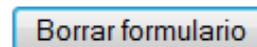
---

- Botones en general: **<button>**

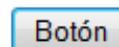
`<button type="submit">Enviar</button>`



`<button type="reset">Borrar formulario</button>`



`<button type="button">Botón</button>`



- El navegador se encarga de enviar automáticamente los datos cuando el usuario pincha el botón

## Formularios

---

- Casillas de verificación (*checkbox*)

Lenguajes de programación: `<br/>`

`<input name="java" type="checkbox" value="on"/>` Java

`<input name="cplusplus" type="checkbox" value="on"/>` C++

`<input name="csharp" type="checkbox" value="on"/>` C#

`<input name="otros" type="checkbox" value="on"/>` Otros

- `value` indica el tipo de casilla: on/off, yes/no, true/false

Lenguajes de programación:

Java  C++  C#  Otros

- *Radiobutton*

Sexo `<br/>`

`<input type="radio" name="sexo" value="hombre" checked="checked" />`  
Hombre

`<input type="radio" name="sexo" value="mujer" />` Mujer

Sexo

Hombre

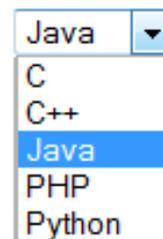
Mujer

## Formularios

---

### ■ Listas de selección

```
<form action="">
<select name="lenguajes">
  <option value="c">C</option>
  <option value="cplusplus">C++</option>
  <option value="java" selected>Java</option>
  <option value="php">PHP</option>
  <option value="python">Python</option>
</select>
</form>
```



- Atributos de option:
  - *value* determina el valor que se envía al servidor
  - *selected* permite definir la opción por defecto

## Formularios

---

### ■ Incluir un fichero

- El atributo `enctype` en la etiqueta `<form>` del formulario tiene que ser `multipart/form-data`

```
<form name="fichero" action="procesa_fichero.php" method="post"
enctype="multipart/form-data">
Fichero: <input type="file" name="archivo" />
<input type="submit" value="Enviar">
</form>
```



## Formularios

---

- Agrupación de elementos
  - Permite ver mejor las partes de un formulario agrupando elementos relacionados
  - `<legend>` es el título que se visualiza con el grupo

```
<form action="">
```

```
<fieldset>
```

```
  <legend>Información personal:</legend>
```

```
  Nombre: <input type="text" size="50"><br>
```

```
  E-mail: <input type="text" size="50"><br>
```

```
  Ciudad: <input type="text" size="20">
```

```
</fieldset>
```

```
</form>
```



## Ejercicios de formularios

---

- Crear un formulario para registrar un nuevo cliente en un sitio web, solicitando datos personales necesarios así como la creación de una clave de acceso (password)
  - Avanzado: Incluir un *captcha* para verificar que se trata de una persona real
    - Esto requiere usar algún script para generar el captcha
- Crear un formulario para entrega de prácticas en una asignatura
  - Investiga cómo se puede comprobar el tipo de fichero
- Crea un formulario para realizar una encuesta
  - Ejemplo:

[http://www.mclibre.org/consultar/amaya/ejercicios/encuesta\\_pc/encuesta\\_pc\\_formateado.html](http://www.mclibre.org/consultar/amaya/ejercicios/encuesta_pc/encuesta_pc_formateado.html)

## Objetos

---

- Es una forma genérica de incluir elementos que son interpretados por algún plugin
- **<object>**
- Atributos:
  - data="URL" - Los datos que utiliza el objeto
  - type="tipo-mime" - Tipo de contenido de los datos
    - El navegador decidirá el plugin o acción que corresponda en función del tipo

```
<object data="pelicula.mpeg" type="application/mpeg" />
```
- height="alto" y width="ancho"
- classid, codebase, codetype - Información específica que depende del tipo de objeto

## Objetos

---

- Recursos que podemos añadir mediante **<object>**
  - Imágenes
    - Aunque es preferible usar <img>
  - Vídeos
  - Archivos de sonido
  - Applets de Java
    - En HTML se usaba la etiqueta <applet>, que desaparece en HTML5
  - Archivos PDF
  - Controles ActiveX
  - ...
- Si el navegador no sabe abrir el contenido, solicitará al usuario que descargue un plugin para ejecutar.

## Objetos

---

- Se pueden proporcionar alternativas para un objeto

```
<object title="Película interesante"
  classid="http://www.miweb.com/peli.py">
  <!-- Formato alternativo en forma de vídeo -->
  <object data="pelicula.mpeg" type="application/mpeg">
    <!-- Otro formato alternativo mediante una imagen GIF -->
    <object data="pelicula.gif" type="image/gif">
      <!-- Si el navegador no soporta ningún formato, mostrar un texto -->
      Aquí debería haberse visto una película interesante.
    </object>
  </object>
</object>
```

- Cualquier texto que no sea una etiqueta se mostrará si el navegador no es capaz de reproducir el contenido.

## Objetos

---

- Para incluir vídeos en formato Flash

```
<object data="video.swf" type="application/x-shockwave-flash">
</object>
```

- Incluir un video de YouTube

```
<p>Aaron Swartz keynote - "How we stopped SOPA" </p>
<object width="640" height="360"
  type="application/x-shockwave-flash"
  data="http://www.youtube.com/v/Fgh2dFngFsg">
  <param name="movie" value="http://www.youtube.com/v/Fgh2dFngFsg" />
  <param name="wmode" value="transparent" />
</object>
```

## Vídeos o sonidos en HTML5

---

- Para incluir vídeos en HTML5

```
<video width="320" height="240" controls>  
  <source src="movie.mp4" type="video/mp4">  
  <source src="movie.ogg" type="video/ogg">  
  Tu navegador no puede mostrar el vídeo.  
</video>
```

- Para incluir sonidos en HTML5

```
<audio controls>  
  <source src="horse.ogg" type="audio/ogg">  
  <source src="horse.mp3" type="audio/mpeg">  
  Tu navegador no puede reproducir el sonido.  
</audio>
```

- Cuestiones importantes:

- El navegador reproducirá el primer vídeo compatible
- Si no puede reproducir ninguno, mostrará el texto que no esté incluido en ninguna etiqueta

## Vídeos o sonidos en HTML5

---

- Cuestiones importantes:

- El navegador reproducirá el primer vídeo/sonido compatible
- Si no puede reproducir ninguno, mostrará el texto

- No todos los navegadores soportan los mismos

- Chrome
  - Sonido: MP3, WAV, OGG
  - Video: H.264+AAC, VP8+Vorbis, OGG
- Firefox
  - Sonido: WAV, OGG (MP3 sólo en Windows)
  - Video: VP8+Vorbis, OGG (H.264 sólo en Windows)
- Internet Explorer
  - Sonido: MP3
  - Video: H.264+AAC
- Safari
  - Sonido: MP3
  - Video: H.264+AAC

Internet Explorer y Firefox son mutuamente excluyentes, por lo que siempre será necesario ofrecer más de una alternativa.

## Vídeos o sonidos en HTML5

---

- **<track>**

- Para incluir archivos de texto en vídeos o sonidos
- Subtítulos, metadatos, descripciones, anotaciones, etc.

```
<video width="320" height="240" controls>
  <source src="forrest_gump.mp4" type="video/mp4">
  <source src="forrest_gump.ogg" type="video/ogg">
  <track src="subtitles_en.vtt" kind="subtitles" srclang="en"
label="English">
  <track src="subtitles_no.vtt" kind="subtitles" srclang="no"
label="Norwegian">
</video>
```

- Sólo es compatible con Internet Explorer, Chrome y Opera.

## Ejercicios

---

- Definir una página web con la propuesta de proyecto para este curso
  - Publicarla y mandar la URL correspondiente al profesor a través del campus virtual

## Extendiendo XHTML con otros lenguajes

---

- Otros lenguajes derivados de XML se pueden combinar bien con XHTML
  - MathML – expresiones matemáticas
  - SVG – gráficos bidimensionales
  - SMIL – animación
- Los documentos que combinan XHTML con estos lenguajes se deben proporcionar al navegador como de tipo MIME *application/xhtml+xml*
  - Y el documento con extensión.xhtml
- Los navegadores modernos suelen dar buen soporte
  - MS Internet Explorer algo menos

## MathML

---

- Mathematical Markup Language
  - Lenguaje de marcas para la representación de fórmulas matemáticas
- Versiones
  - MathML 1.0 (1998)
  - MathML 2.0 (2001)
  - MathML 3.0 (2010)
- Para probar si un navegador interpreta bien MathML
  - [https://www.eyasme.com/Joe/MathML/MathML\\_browser\\_test](https://www.eyasme.com/Joe/MathML/MathML_browser_test)
  - Internet Explorer, por ejemplo, no es capaz de trabajar con MathML
    - Una solución es enviarle las fórmulas como imágenes
- Ejemplo: Resolución de ecuaciones de segundo grado:  
[http://www.mclibre.org/consultar/amaya/xhtml/mathml\\_ejemplo\\_segugra.xhtml](http://www.mclibre.org/consultar/amaya/xhtml/mathml_ejemplo_segugra.xhtml)

# SVG

---

- Scalable Vector Graphics
  - Lenguaje de marcas para la representación de gráficos vectoriales bidimensionales (dibujos y texto)
- Versiones actuales
  - Mobile SVG Profiles (2003)
    - Define SVG Tiny, para teléfonos móviles, y SVG Basic, para PDAs
  - SVG Tiny 1.2 (2008)
  - SVG 1.1 (2ª edición) (2011)
- Soportado cada vez mejor por todos los navegadores
  - Algo menos en Internet Explorer
- Ejemplos:
  - La Wikimedia Commons tiene muchas imágenes disponibles

# SVG

---

- Utilización de gráficos SVG
  - Como objeto externo (con extensión .svg)

```
<object
  data="ejemplo.svg"
  type="image/svg+xml"
  height="80"
  width="80">
</object>
```
  - HTML5: Anidando a partir de una etiqueta SVG

```
<body>
<h1>Prueba de SVG</h1>
<svg width="100" height="100">
  <circle cx="50" cy="50" r="40" stroke="green" stroke-width="4"
  fill="yellow" />
</svg>
</body>
```

## Bibliografía

---

- <http://www.w3schools.com/>
  - Tutoriales de tecnologías para desarrollo de aplicaciones web
  - Tiene ejemplos que se pueden cambiar y probar directamente
- Tutoriales muy comprensibles en español:
  - <http://librosweb.es/>
  - <http://www.mclibre.org/consultar/amaya/index.html>

**Comentar bibliografía, ejemplos y sugerencias sobre la presentación en el campus virtual**