

Comunicación Persona-Máquina

HTML

Diseño Web

- A la hora de diseñar un sitio web, distinguimos:
 - Contenido: Todo lo que representa **INFORMACIÓN** dentro de un sitio web, así como su estructura y organización.
 - Presentación: Todo lo relativo al aspecto visual del sitio web.

HTML

- Lenguaje de Marcado de Hipertexto (*HiperText Markup Language*).
 - Conjunto de normas para escribir una página web entendible por un navegador
 - Lenguaje utilizado para representar el **CONTENIDO** de las páginas Web.
- Estándar reconocido y cuyas normas define un organismo denominado W3C (*World Wide Web Consortium*).
 - Empresas, organismos públicos, desarrolladores de navegadores...



HTML

- Comenzó como **único** lenguaje Web (1991): Contenidos + Presentación
 - Utilidad prácticamente académica
 - Poca importancia al aspecto gráfico, apenas sin presentación
- La popularización de Internet creó la necesidad de introducir más elementos de presentación en HTML
 - Mezcla de elementos de presentación y contenidos en un mismo documento
 - Códigos demasiado extensos
 - Páginas poco mantenibles

HTML

- Nace CSS (*Cascade Style Sheets*) para posibilitar la separación de código (1996)
 - Ahora esa separación se convierte en obligatoria
 - Las etiquetas HTML relativas a presentación se mantienen únicamente por **retrocompatibilidad** → **NUNCA** se deben utilizar
- Versiones actuales HTML
 - HTML 4.01 es la última versión final
 - XHTML 1.0
 - Basado en XML añade semántica al HTML
 - HTML 5.0 es el presente y futuro

HTML 5

- Es la versión de HTML vigente
 - Vio la luz en 2014 pero sigue estando en continuo desarrollo
 - Da nombre al estándar web actual, que incluye HTML 5, CSS3 y JavaScript
 - El W3C decidió excluir de esta versión tecnologías externas muy demandadas en los 90 (Flash, Silverlight, etc.)
 - Cómo alternativas se proponen nuevas funcionalidades CSS y JavaScript → En continuo desarrollo

CSS

- *Cascade Style Sheet*
 - Complementa el HTML
- Define la **PRESENTACIÓN**, es decir, el formato y la apariencia de la página web
- Versiones
 - CSS I
 - CSS II
 - CCS3 (adaptado a HTML 5.0)

Estándares

- Tanto HTML como CSS son estándares, definidos por el W3C
 - <http://www.w3.org>
 - Marca las directrices de la Web que seguirán
 - los navegadores y ...
 - ...otras herramientas Web.
 - Al tratarse de estándares, la interpretación puede ser diferente según el navegador utilizado
 - Probar las páginas siempre con varios navegadores diferentes

Herramientas

- Editores Web
 - Un documento .html es un simple archivo de texto que es interpretado por un navegador web
 - Podemos utilizar Notepad++
 - Si queremos estructurar y organizar los proyectos
 - Aptana Studio, BlueGriffon, Aloha Editor, Maquette...
 - WebStorm:
 - Es un IDE para Javascript, pero sirve de editor de HTML5 y CSS
 - <https://www.jetbrains.com/webstorm/>
- Tutoriales
 - W3Schools: <http://www.w3schools.com/>

Herramientas

- Navegadores
 - Chrome, Firefox, Opera, etc...
- Para evaluar si tu navegador soporta HTML 5.0
 - <http://html5test.com/>
- Comprobación:
 - [HTML 5 Test](#) | [Find Me By IP](#) | [Can I Use](#)

Elementos HTML

- Un elemento HTML consta de:
 - **Etiqueta de apertura + Contenido + etiqueta de cierre**
 - <p>El contenido</p>
 - **Solo una etiqueta**
 - <hr> : Separa dos textos diferentes conceptualmente (se visualizará una línea horizontal)
- NOTA: En versiones anteriores a HTML5 se requería “autocierre” en estas etiquetas: <hr/>
- Algunas etiquetas tienen **parámetros**, cuyos valores van entre **comillas dobles**
 -

Parámetros en etiquetas HTML

- Algunas etiquetas tienen **parámetros**, cuyos valores van entre **comillas dobles**
 -
- Los parámetros booleanos no tienen valor
 - Si los ponemos → true
 - Si no los ponemos → false

<input type="checkbox" **checked**>

NOTA: En versiones anteriores a HTML5 los parámetros booleanos recibían su propio nombre como valor, para definirlos como ciertos:

<input type="checkbox" checked="checked">

Estructura de la página

```
<!DOCTYPE html>
```

Especifica el tipo de documento y las normas

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

Inicio de la página web

```
    <head>
```

```
        <meta charset = "utf-8">
```

Codificación

Cabecera

*Titulo, autores,
estilos...(Información no visible)*

```
        <title>Página simple</title>
```

```
    </head>
```

```
    <body>
```

```
        <h1>Página simple</h1>
```

Cuerpo

*Contenido de la página
(Información visible)*

```
        <p>Esto es un
```

```
        <a href="demo.html">ejemplo</a> simple.</p>
```

```
        <!-- this is a comment -->
```

```
    </body>
```

```
</html>
```

Fin de la página web

Visualización por defecto

- En ausencia de CSS, el navegador aplica estilos muy básicos al HTML
- Sirven de guía al diseñador para ver diferencias entre elementos antes de crear CSS
 - Cuando creamos HTML, no debemos pensar en la presentación

Página simple

Esto es un ejemplo simple.

Elementos semánticos

- Los elementos HTML sirven para **marcar** partes de una web: Títulos, párrafos, listas...
- Salvo dos elementos muy concretos (DIV y SPAN), **todos** responden a una semántica
- Debemos marcar elementos **únicamente atendiendo a la semántica de la etiqueta** y nunca a la visualización por defecto
- Ejemplo: Un título de nivel 1 se muestra más grande que de nivel 2, pero **nunca** pondremos por estética un título 2 sin estar precedido por un título 1.

Algunos elementos

- Párrafos
 - Delimitados por `<p> </p>`
- Títulos
 - Delimitados por `<h1> </h1>`, ..., `<h6> </h6>`
 - **Se debe respetar el orden**
- Separadores de contenido
 - `
`: Salto de línea
 - `<hr>`: Cambio de tema entre párrafos (El estilo por defecto, representa una línea horizontal)
- Los comentarios (de una o varias líneas)
 - `<!-- Comentario en una página HTML -->`

La cabecera <head></head> (I)

Su contenido no se muestra, pero proporciona información útil

- **<title> </title>**
 - Define el título de la página (obligatorio en HTML 5.0)
 - Se muestra en la barra de título del navegador (pestana), al hacer un marcador a la página o en los buscadores de Internet.
- **<link >**
 - Enlaza recursos externos, generalmente hojas de estilo (veremos más adelante cómo).

La cabecera <head></head> (II)

- **<style> </style>**
 - Define estilos específicos para la página actual (veremos cómo más adelante)
 - Es alternativo a **<link>**
- **<base>**
 - define la URL base en la propiedad href para el resto de los enlaces que incluyamos en la página web. En la propiedad target se especifica dónde se cargarán los enlaces (_blank, _self).
- **<script> </script>**
 - Encapsula instrucciones de otro lenguaje, como Javascript (lo veremos más adelante)

La cabecera <head></head> (III)

- <**meta**> </**meta**>
 - Define metadatos (no obligatorio pero recomendable)
 - La codificación de caracteres es imprescindible (UTF-8)
 - En HTML4.01
 - <**meta** http-equiv="content-type" content="text/html; charset=UTF-8">
 - En HTML5
 - <meta charset="UTF-8">
 - Parámetros: el tipo de etiqueta (**name**) y el contenido (**content**)
 - <meta **content**=“Perico Palotes” **name**=“author”>
 - <meta **content**=“Prácticas de HTML” **name**=“description”>
 - <meta **content**=“html, etiquetas, estilos” **name**=“keywords”>

Estructura en el <body></body> (I)

- HTML 5 incluye varias etiquetas que nos permiten definir la estructura del documento
- Estas etiquetas no tienen una visualización por defecto en el navegador
- Nos permiten:
 - Definir bloques
 - Asignar roles a los bloques
 - Cada etiqueta define semánticamente al bloque

Estructura en el <body></body> (I)

Todas tienen **APERTURA** y **CIERRE**

- **<header>**
 - Define cabeceras (de la página, de artículos, de secciones...)
- **<footer>**
 - Define pies (de la página, de artículos, de secciones...)
- **<nav>**
 - Define zonas con enlaces de navegación entre páginas de nuestro sitio web
- **<section>**
 - Define grandes secciones, subsecciones, etc...

Estructura en el <body></body> (II)

Todas tienen **APERTURA** y **CIERRE**

- **<article>**
 - Define un bloque con formato de artículo (una noticia, una entrada en blog, etc.)
- **<aside>**
 - Define un contenido que está relacionado con la página, pero va separado del contenido principal.
- **<mark>**
 - Define textos destacados para referencias
 - Por defecto, fondo amarillo y caracteres en negro

Estructura en el <body></body> (III)

Todas tienen **APERTURA** y **CIERRE**

- **<figure>**
 - Define contenidos multimedia (imágenes, video, audio)
 - Dentro se puede utilizar **<figcaption></figcaption>** para especificar el pie de figura y quedando, de esta forma, vinculados
- **<time>**
 - Asigna la fecha real a la que nos referimos en el contenido, mediante el parámetro **datetime**.

Marcado genérico de bloques: La etiqueta <div></div>

Si se quiere definir un bloque que no tiene concordancia con las etiquetas semánticas vistas, usaremos <div>

- Los parámetros ***id*** y ***class*** nos permiten identificarlos en el css
 - Estos parámetros son comunes en todos los elementos del body
 - <div ***id***=“15” ***class***=“CPM”>El recurso 15 de CPM</div>
- Las etiquetas semánticas estructurales aparecieron en HTML5 → Antes se utilizaba <div> para todo
<div id=“cabecera”> → <header>

Etiquetas de texto (I) : Enfatizado de la información

Destacar y *enfatizar*

- ** **
 - Destacar (El estilo por defecto lo muestra en **negrita**)
- ** **
 - Enfatizar (El estilo por defecto lo muestra en *cursiva*)

NOTA: En las primeras versiones de HTML se incluían las etiquetas de formato

- **** Negrita (bold) ****
- **<i>** Cursiva (Italic) **</i>**

Estas etiquetas no deben utilizarse en ninguna circunstancia puesto que hacen referencia al diseño y no al tipo de información.

Etiquetas de texto (II): Tipos de información

Etiquetas con APERTURA y CIERRE

- **<cite>**
 - título de un trabajo, obra...
- **<code>**
 - bloques de código informático
- **<kbd>**
 - texto de teclado
- **<var>**
 - variable
- **<samp>**
 - ejemplos

Etiquetas de texto (III): Texto con información adicional

Etiquetas con **APERTURA** y **CIERRE**

- **<dfn>** Sirve para conceptos que se quieren definir.
- **<abbr>** Sirve para definir abreviaturas.
- **<acronym>** Sirve para definir acrónimos.
- Cualquiera de estos elementos utiliza el parámetro title para completar la información
 - *<acronym title=“Universidad de Oviedo”> UniOvi</acronym>*

El parámetro *style*

Disponemos de tres maneras de incluir CSS en HTML:

- Estilos en un **archivo externo**
 - Permiten que distintas páginas accedan al mismo CSS
 - Se enlazan en la cabecera
- Estilos en la **cabecera** de la página web
 - Estilos CSS específicos para una página
 - Reglas CSS especificadas en la cabecera de la página
- **Estilos en los elementos** de la página web
 - Estilo propio para un único elemento
 - A través del parámetro global *style*

El parámetro *style*

- Es un parámetro global: Común a todos los elementos
- Sirve para aplicar reglas CSS a un elemento
- La principal utilidad es hacer pruebas, no se recomienda su uso en sitios en explotación
 - Cada regla se separa con ;
 - Entre la propiedad y el valor, :
 - Ejemplo:

```
<p style="color: rgb(200, 20, 20); font-family: Arial,sans-serif; font-style: italic; text-align: center;">
```

Esto es un párrafo

```
</p>
```

Esto es un párrafo

Marcado genérico de texto: Etiqueta

- Existen muchos contenidos textuales para los cuales HTML no especifica etiquetas.
- <**span**> permite marcar cualquier elemento que no se puede marcar con las otras etiquetas definidas.
- Esta etiqueta nos permitirá aplicar un estilo CSS a los fragmentos de texto que no coincidan semánticamente con ninguna de las etiquetas vistas
- Al igual que con los elementos <**div**>, los parámetros ***id*** y ***class*** cobran especial relevancia

Los enlaces

- `<a> ` con la propiedad `href` para especificar la URL
- Ejemplo:
 - `<p> Esto es un enlace a Google </p>`

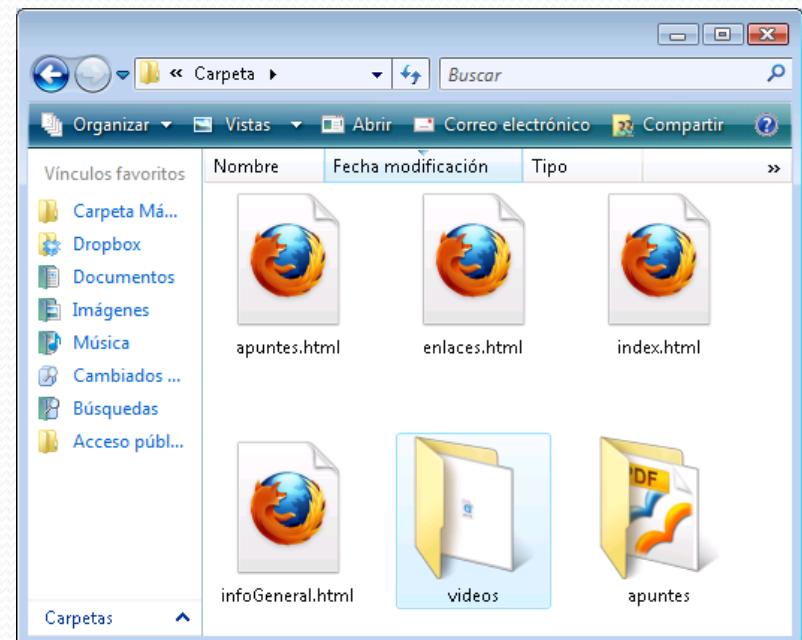
Esto es un enlace a [Google](http://www.google.es)

Tipos de enlaces

- En función de la ubicación de la página:
 - Enlaces **externos**: Apuntan a páginas ubicadas en otros servidores.
 - Enlaces **internos**: Apuntan a páginas ubicadas en el mismo servidor en el que nos encontramos.
- En función del formato del enlace:
 - Enlaces **absolutos**: Incluyen toda la información necesaria (protocolo, servidor y ruta). **Normalmente se usan para enlaces externos**.
 - Enlaces **relativos**: Prescinden de parte de la información necesaria. **Normalmente se usan para enlaces internos**.

Ejemplos enlaces

- Referencias absolutas (enlace externo)
 - <a href=<http://www.google.es>> Google
- Referencias relativas (enlaces internos)
 -
Apuntes
 -
Video
 -
Inicio



Destino de los enlaces

- Con el parámetro *target*

```
<a href=http://www.google.es target="_blank"Google </a>
```

- Valores de *target*

- **_blank** (abrir en pestaña nueva o ventana nueva)
- **_self** (abrir en la misma ventana, es el valor por defecto)
- **_parent** y **_top** (abrir en el marco anterior o en el espacio principal de la página)

Correo electrónico (I)

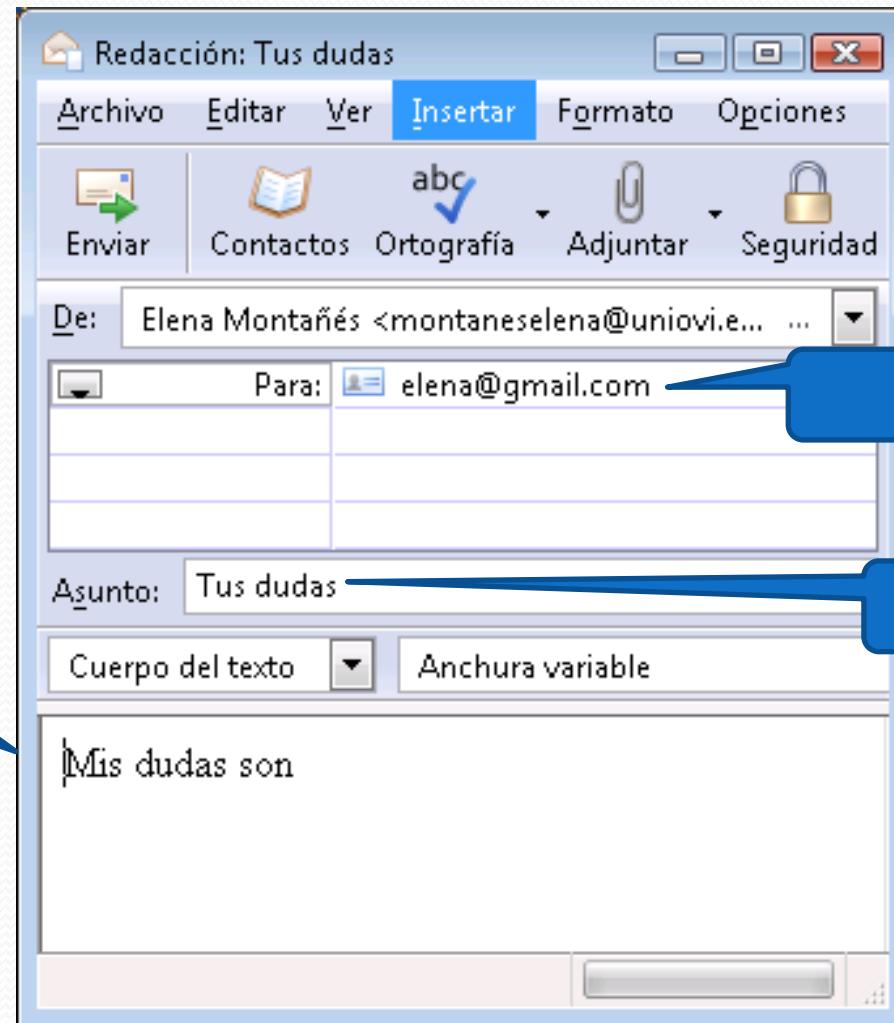
- Utilizamos el servicio **mailto** en vez del protocolo http
- También podemos utilizar *cc* y *bcc*

< a href="mailto:elena@gmail.com?subject=Tus%20dudas&body=

Mis%20dudas%20son: %0A1%0A2%0A...> Tutorías

- ? Separa el enlace de sus propiedades
- & separa cada propiedad
- %20 es el espacio en blanco
- %0A para saltos de línea

Correo electrónico (II)



Imágenes (I)

- Imágenes que formen parte de la información
 - Las imágenes decorativas → CSS
- Típicamente en formato de **mapa de bits** (pixel a pixel)
 - **png** (para imágenes sencillas, fondos)
 - **jpg** (fotos e imágenes más complicadas)
 - **gif** (como png, pero pueden tener movimiento)
- La etiqueta **img** y los parámetro **src** y **alt**
 - ``
¡NO tiene etiqueta de CIERRE!

El valor de **src** puede ser un enlace a una imagen de una página web, como flickr

Imágenes (II)

- Los parámetros ***width*** and ***height***
 - En píxeles (por defecto, si no se especifica nada)
 - ``
 - En porcentaje
 - ``

¡Especificiar solo uno para mantener la relación de aspecto!
- La recomendación es no modificar el tamaño si es posible
 - Aumentar tamaño → Perdida de calidad
 - Reducir tamaño → Espacio en disco mayor al necesario
 - Solución: Utilizar una herramienta de reescalado

Videos (I)

- Diferentes formatos
 - MPEG4 (.mp4)
 - Flash (.flv o .f4v)
 - Avi (.avi)
 - WebM (.webm)
 - Ogg Vorbis (.ogg)
- La etiqueta **<video>** y el parámetro **src**
 - **<video src="videos/mivideo.webm"></video>**
- También soporta los parámetros **width** y **height**

Videos (II)

- Si falla algún formato (muy común dada la heterogeneidad), se pueden poner alternativas
 - <video>
 - <**source** src="videos/mivideo.mp4" type="video/mp4" />
 - <**source** src="video/mivideo.ogg" type="video/ogg" />
 - <**source** src="video/mivideo.webm" type="video/webm" />
 - </video>

Videos (III)

- Más parámetros
 - **Controls** (añade controles para pararlo y arrancarlo)
 - **Autoplay** (se reproduce en cuanto se cargue)
 - **Preload** (comienza a cargarse al cargar la página)
 - **Loop** (se reproduce cíclicamente)
 - **Muted** (al iniciar el video no hay audio)
¡Basta indicar el parámetro, no requieren valores!
- Para videos externos utilizar **<object></object>** y “embeber” el código HTML

Audio

- Diferentes formatos
 - .mp3
 - .wav
 - .ogg
- La etiqueta **<audio>** y el parámetro **src**
 - **<audio src=“audio/miaudio.mp3”></audio>**
- También tiene las propiedades **controls** y **autoplay**
- Para audios externos utilizar **<object></object>** y “embeber” el código HTML

Listas (I)

- Cada item con ``
- Viñetas con ``
 - ``
`CPM`
`TPP`
``
 - Numeración con ``
 - ``
`CPM`
`TPP`
``
 - CPM
 - TPP

Listas (II)

- Listas de definiciones

- <dl>

- <dt>CPM</dt>

- <dd>Comunicación Persona-Máquina</dd>

- <dt>TPP</dt>

- <dd>Tecnologías y Paradigmas de Programación</dd>

- </dl>

CPM

Comunicación Persona-Máquina

TPP

Tecnologías y Paradigmas de Programación

Listas (III)

- Las propiedades del parámetro **style** tomadas del CSS ***list-style-type***, ***list-style-position*** y ***list-style-image***
 - <li style="**list-style-type**: square; **list-style-position**: inside;">CPM
 - <li style="**list-style-image**: url(miimagen.gif); **list-style-position**: outside;">TPP

También pueden ir en **** y ****

Tablas (I): Sin cabecera ni pie

- Con **<table></table>**, **<tr></tr>** y **<td></td>**

```
<table border="2">
```

```
<tr>
```

```
<td>CPM</td>
```

```
<td>TPP</td>
```

```
</tr>
```

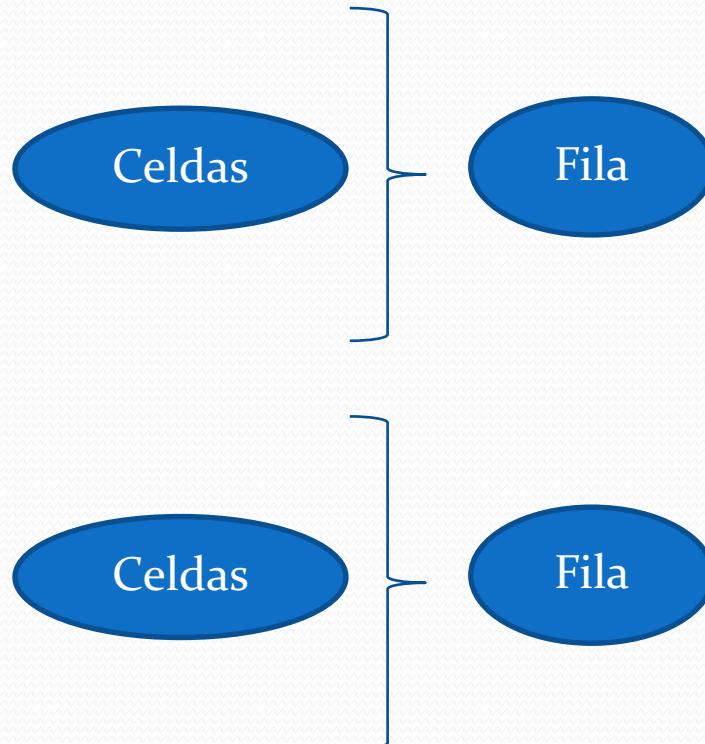
```
<tr>
```

```
<td>ALG</td>
```

```
<td>SO</td>
```

```
</tr>
```

```
</table>
```



Tablas (II): Título y formato de columnas

- Con **<caption></caption>** para el título de la tabla
 - <table border="2"><caption>Asignaturas</caption>
...
</table>
- Con **<colgroup></colgroup>** para formatear un grupo de columnas (en **span** se indica cuántas)
 - <table border="2">
<colgroup style="background-color: rgb(20, 20, 20);"
span="1"></colgroup >
...
</table>

Tablas (III): Con cabecera y pie

- La cabecera con `<thead></thead>`, el cuerpo con `<tbody></tbody>` y el pie de tabla con `<tfoot></tfoot>`
- ```
<table border="2">
 <thead>
 <tr><th>1ºCuat.</th><th>2ºCuat.</th></tr> ←
 </thead>
 <tbody>
 <tr><td>CPM</td><td>TPP</td></tr>
 <tr><td>ALG</td><td>SO</td></tr>
 </tbody>
 <tfoot><tr><td>2</td><td>2</td></tr></tfoot>
</table>
```
- `<th></th>`  
para las  
celdas de la  
cabecera

# Tablas (IV): Combinar columnas

- Con **colspan** combinamos columnas

- ```
<table border="1">
<tbody>
<tr> <td colspan="3" style="text-align: center;">Asignaturas
```

Asignaturas		
CPM	TPP	ALG

Tablas (V): Combinar filas

- Con **rowspan** combinamos filas

- ```
<table border="2">
<tbody>
<tr> <td rowspan="3"> Asignaturas </td>
<td> CPM </td> </tr>
<tr><td> TPP </td> </tr>
<tr><td> ALG </td> </tr>
</tbody>
</table>
```

Asignaturas	CPM
	TPP
	ALG