



Nombre:

Fecha: / 10 /2010

Grupo: 1 ☐ 2 ☐ 3 ☐ 4 ☐

PRÁCTICA 5
HTML. TABLAS Y FORMULARIOS.

En esta práctica introduciremos dos nuevos elementos de nivel de bloque que nos permitirán conseguir dos efectos nuevos en nuestros sitios web. En primer lugar, trabajaremos con tablas en HTML. En segundo lugar, veremos cómo se pueden crear formularios a través de los cuales los usuarios de una web pueden enviar información a un servidor.

Antes de entrar a explicar en más detalle los tags que permiten definir tablas y formularios vamos a explicar alguna propiedad más de los tags HTML que nos permitan entender mejor lo que haremos más adelante. En general, cualquier tag HTML (p. ej. <a>...) tiene asociados dos tipos de componentes (diferentes para cada tipo de tag, y que son opcionales):

1. Atributos: los atributos nos permiten definir características adicionales de los tags. Su sintaxis de uso es:

`<tag atributo1="valor1" atributo2="valor2">...</tag>`

Por ejemplo, si recuperamos el tag <a>... que nos permitía definir orígenes y destinos de vínculos (anclas o enlaces), podemos ver que en su lista de atributos (<http://html.conclase.net/w3c/html401-es/struct/links.html#edef-A>) contiene algunos de los siguientes:

- name: asigna un nombre, de modo que el mismo pueda actuar como destino de un vínculo.
- href: especifica la localización de un sitio en la web (o en nuestra propia página), enlazando así el elemento actual con dicho sitio.

Puedes encontrar la lista exhaustiva y sus funciones en el enlace anterior.

2. Elementos: los elementos de un tag son aquel conjunto de tags que podemos encontrar dentro del mismo. Por ejemplo, dentro de un tag ... no podemos encontrar un tag <body>, porque el navegador no sabría interpretarlo de forma correcta. Para conocer los tags que pueden ser elementos de otro, puedes consultar la especificación de HTML 4.01. Por ejemplo, dentro de una lista (... o ...), consultando <http://html.conclase.net/w3c/html401-es/struct/lists.html#h-10.2> podrás comprobar que los elementos disponibles son:

- : Una o más copias del atributo .

Por lo tanto, dentro de una lista, el único elemento que podemos encontrar son . Dentro de cada podremos encontrar los elementos que permita el mismo. ¿Cuáles son? (Verás que los mismos están definidos como "flow", y pulsando el enlace puedes comprobar que esto incluye cualquier elemento de bloque, como H1...H6, p, ol, ul, table... o de línea como sub, sup, span, br...)

Con las ideas anteriores intentaremos presentar ahora cómo construir tablas y formularios en HTML, basándonos tanto en los atributos como los elementos que pueden contener ambas.

Parte 1: Tablas en HTML.

Las tablas en HTML son una forma de organizar la información para que la misma se muestre por medio de filas y columnas. Son extremadamente útiles y permiten conseguir efectos que con otros elementos de nivel de bloque (H1...H6, p, ol, ul...) no se podrían conseguir. Eso ha llevado a que incluso muchos desarrolladores web las utilicen para dar formato a sus sitios web, aunque esta práctica está desaconsejada incluso por la especificación estándar de HTML 4.01 (Capítulo 7, Tablas): "No deberían usarse tablas con la única finalidad de organizar la presentación de los contenidos de un documento (es decir, de crear el "layout"), ya que esto puede ocasionar problemas cuando se represente en un medio no visual. Además, al incluir gráficos, estas tablas pueden forzar a los usuarios a hacer desplazar horizontalmente la pantalla para ver una tabla diseñada en un sistema con una pantalla más grande. Para minimizar estos problemas, los autores deberían usar hojas de estilo en lugar de tablas para organizar la presentación."

Vamos a ver cuáles son los principales "tags" que pueden ser utilizados en el desarrollo de una tabla en HTML:

- `<TABLE>...</TABLE>`: Puedes encontrar su especificación en <http://html.conclase.net/w3c/html401-es/struct/tables.html#edef-TABLE>. El tag table marca el principio y el final del contenido de la tabla (dentro del mismo se incluirá el título de la misma, las cabeceras, el pie o cualquier información que contenga la misma).

Atributos más relevantes del tag table (por tanto, irán situados como `<table atributo1="value" atributo2="value" ... >`):

- `summary`: resumen del contenido de la tabla, pensado para dispositivos (navegadores) que se presentan en medios no visuales (para invidentes...). Muchos de los navegadores convencionales no lo muestran de manera visual, pero es conveniente incluirlo.
- `width`: anchura de la tabla. La anchura se puede especificar por medio de un valor absoluto en píxeles (`width="600"`), o un valor porcentual con respecto a la anchura de la ventana de visualización (`width="75%"`). De las dos opciones, la segunda es más recomendable porque permite al navegador redimensionar las tablas dinámicamente. Si no ponemos un valor, el navegador tomará uno por defecto.
- `border`: permite definir la anchura del marco que rodea a una tabla (y las celdas de la misma). Se mide en píxeles, y se puede especificar como `border="5"` (borde de 5 píxeles).
- `cellspacing`: permite definir la distancia que debe haber entre los bordes de la tabla y las celdas de la misma, así como entre las celdas de la tabla. Por ejemplo, `cellspacing="2"`, definiría 2 píxeles de distancia.
- `cellpadding`: permite definir la distancia entre el borde de las celdas de la tabla y el contenido de las mismas. También se define, como

cellspacing, en píxeles (o en porcentaje, aunque su significado es menos claro). Puedes encontrar una representación gráfica del significado de cellpadding y cellspacing en <http://html.conclase.net/w3c/html401-es/struct/tables.html#h-11.3.3>.

Hay algunos otros atributos que puedes encontrar en <http://html.conclase.net/w3c/html401-es/struct/tables.html#edef-TABLE> que aquí no detallaremos.

Elementos propios del tag table (irán todos ellos situados dentro del bloque <TABLE>...</TABLE>):

- <caption>...</caption>: el elemento caption debe definir el título, contenido o descripción de la tabla. Los agentes visuales (navegadores) lo mostrarán por defecto encima de la tabla.
- <colgroup>...</colgroup>: permite definir grupos de columnas que vayan a tener unas mismas propiedades (especificadas en los atributos de colgroup). Los más representativos son:
 - o span: define el número de columnas que comprenden el grupo de columnas afectado por colgroup.
 - o width: permite definir la anchura por defecto para las columnas afectadas por el colgroup.

Por ejemplo, el tag <COLGROUP span = "6" width = "16%"> </COLGROUP> define un grupo de 6 columnas, la anchura de las cuales será cada una del 16% del total (serán todas de la misma anchura).

- <col>: el elemento col tiene atributos similares a los de colgroup. Además, lo podemos usar dentro de colgroup para definir subgrupos de columnas que compartan características.

Pasamos ahora a ver los tags que nos van a permitir definir las distintas filas de la tabla (no es obligatorio usar los tres tipos en cada tabla):

- <THEAD>...</THEAD>: Define las cabeceras de la tabla. No es obligatorio incluirla. La misma debe estar compuesta por, al menos, una fila, que será un elemento de la forma <TR>...</TR> (veremos más adelante cómo definir las filas).
- <TBODY>...</TBODY>: Define el cuerpo de la tabla. Al igual que thead, debe estar compuesta por, al menos, una fila, de la forma <TR>...</TR>.
- <TFOOT>...</TFOOT>: Define el pie de la tabla. Al igual que los anteriores estará compuesto por filas de la forma <TR>...</TR>.

El único elemento que podemos encontrar en los anteriores tags thead, tbody y tfoot es:

- <TR>...</TR>: permite definir las distintas filas (tr = table row) de la tabla. Cada fila, a su vez, sólo puede estar compuesta por los siguientes elementos:
 - o <TH>...</TH>: celda de encabezado. Generalmente mostrada en negrita. El tag de cierre es opcional.
 - o <TD>...</TD>: celda de datos. El tag de cierre es opcional.Dentro de los tags td y th podemos encontrar cualquier elemento de nivel de línea (a, strong, em...) o de bloque (p, ol, ul, table...).

Los tags th y td tienen varios atributos. De ellos, los que más nos interesan son los siguientes:

- rowspan: permite hacer que una celda ocupe varias filas (<TD rowspan="3"> haría que la celda definida ocupara 3 filas).
- colspan: permite hacer que una celda ocupe varias columnas (<TD colspan="2"> haría que la celda definida ocupara 2 columnas).

Ambos tags se pueden usar de forma conjunta para definir una celda que ocupe varias celdas y columnas.

- align: define la alineación del contenido de la celda con respecto a la misma. Puede tomar, entre otros, los valores "left", "center" y "right".
- valign: define la alineación vertical del contenido de la celda con respecto a la misma. Sus valores pueden ser "top", "middle", "bottom" y "baseline".

Los dos atributos anteriores también se pueden aplicar a tr, afectando a todas las celdas de una fila, a thead, tbody y tfoot, afectando a todas las celdas de esas áreas, y a colgroup y col, afectando a todas las celdas de esas columnas.

En la página web <http://www.htmlquick.com/es/tutorials/tables.html> puedes encontrar varios ejemplos de tablas que hacen uso de los anteriores elementos.

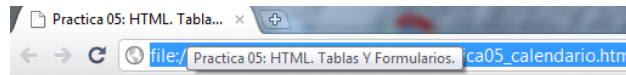
1. Crea un fichero practica05_horario.htm (incluye en el mismo cabeceras y metas como vimos en la práctica 4). Haz uso de los anteriores ejemplos para realizar la tabla HTML correspondiente a tu horario del primer curso, primer cuatrimestre de GII o GM. Puedes encontrar el horario en http://www.unirioja.es/facultades_escuelas/fceai/horarios/horarios_10_11/pdf/gii_1.1.pdf o en

http://www.unirioja.es/facultades_escuelas/fceai/horarios/horarios_10_11/pdf/gm_1.1.pdf. Selecciona el grupo de prácticas (INFOR A1... INFOR A4) y el grupo reducido (REDUC A1...REDUC A3) que te corresponda. Recuerda validarla al terminar, e incluir el icono correspondiente. Por ejemplo, la tabla HTML, para una persona de GII que esté en el grupo REDUC A2 y en el grupo INFOR A1 debería quedar como:

Periodo Lectivo: Curso 2010-11 Semestre 1
Titulación: Grado en Ingeniería Informática

HORARIO DEL CURSO 1 - GRUPO 1.1					
Horario	Lunes	Martes	Miércoles	Jueves	Viernes
9:00 10:00		Sistemas Informáticos GRUPO: GRANDE A Prof a Contratar 06-Aula 107	Cálculo Matricial y Vectorial GRUPO: REDUC A2 J. LALIENA 06-Aula 107		
10:00 11:00		Metodología de la Programación GRUPO: GRANDE A J. SAN JUAN 06-Aula 107	Cálculo Matricial y Vectorial GRUPO: GRANDE A J. LALIENA 06-Aula 107	Metodología de la Programación GRUPO: GRANDE A J. SAN JUAN 06-Aula 107	Sistemas Informáticos GRUPO: INFOR A1 A. ROMERO 05-Sala 2 inf.
11:00 12:00	Cálculo Matricial y Vectorial GRUPO: GRANDE A J. LALIENA 06-Aula 107	Cálculo Infinitesimal GRUPO: GRANDE A J. MÍNGUEZ 06-Aula 107	Cálculo Infinitesimal GRUPO: GRANDE A J. MÍNGUEZ 06-Aula 107	Cálculo Infinitesimal GRUPO: GRANDE A J. MÍNGUEZ 06-Aula 107	
12:00 13:00	Matemática Discreta GRUPO: GRANDE A J. PÉREZ 06-Aula 107	Cálculo Matricial y Vectorial GRUPO: GRANDE A J. LALIENA 06-Aula 107	Matemática Discreta GRUPO: GRANDE A J. PÉREZ 06-Aula 107	Matemática Discreta GRUPO: GRANDE A J. PÉREZ 06-Aula 107	Matemática Discreta GRUPO: INFOR A1 Prof. a contratar 05-Sala 2 inf
13:00 14:00	Matemática Discreta GRUPO: REDUC A2 J. PÉREZ 06-Aula 107	Cálculo Matricial y Vectorial GRUPO: GRANDE A J. LALIENA 06-Aula 107	Cálculo Infinitesimal GRUPO: REDUC A2 J. MÍNGUEZ 06-Aula 107		
16:00 17:00	Cálculo Infinitesimal GRUPO: INFOR A1 J. ARREGUI 04-Sala 1 Mac (S. par)				
17:00 18:00	Cálculo Matricial y Vectorial GRUPO: INFOR A1 J. LALIENA 04-Sala 1 Mac (S. impar)				
18:00 19:00	Metodología de la Programación GRUPO: INFOR A1 B. PÉREZ 04-Sala4 BSCH	Sistemas Informáticos GRUPO: INFOR A1 A. ROMERO 05-Sala 2 inf.			
19:00 20:00					

2. Crea un fichero practica05_calendario.htm (incluye en el mismo cabeceras y metas como vimos en la práctica 4). Realiza una tabla que contenga el calendario del primer cuatrimestre del curso 2010/2011. Puedes encontrar el mismo en http://www.unirioja.es/estudiantes/calendario_academico/Calend_presencial10.pdf. El resultado final debería ser algo parecido a:



Calendario

CALENDARIO ACADÉMICO 2010/2011
Titulaciones presenciales

SEPTIEMBRE	OCTUBRE	NOVIEMBRE
L M X J V S D	L M X J V S D	L M X J V S D
 1 2 3 4 5	1 2 3	1 2 3 4 5 6 7
6 7 8 9 10 11 12	4 5 6 7 8 9 10	8 9 10 11 12 13 14
13 14 15 16 17 18 19	11 12 13 14 15 16 17	15 16 17 18 19 20 21
20 21 22 23 24 25 26	18 19 20 21 22 23 24	22 23 24 25 26 27 28
27 28 29 30	25 26 27 28 29 30 31	29 30
DICIEMBRE		
L M X J V S D		
 1 2 3 4 5		
6 7 8 9 10 11 12		
13 14 15 16 17 18 19		
20 21 22 23 24 25 26		
27 28 29 30		



Puedes encontrar las dos imágenes que hemos incrustado en los meses de Septiembre y de Diciembre en <http://www.unirioja.es/cu/jearansa/1011/ficheros/luna1.jpg> y <http://www.unirioja.es/cu/jearansa/1011/ficheros/luna2.jpg>.

El tag para introducir imágenes en HTML tiene la siguiente sintaxis:

- ****: es un tag sin contenido, y por tanto no es necesario que se incluya ni tag de cierre ni ningún texto adicional. Al ser un tag sin contenido, no hay elementos que podamos introducir dentro del mismo. Sin embargo, sí dispone de varios atributos que es importante conocer:

- **src**: define la dirección URL donde se encuentra la imagen. En general será preferible que la ruta de la misma se dé en local, en lugar de facilitar la ruta entera.
- **alt**: es un atributo obligatorio que ofrece un texto alternativo para dispositivos que no sean capaces de mostrar imágenes. Debe ser explicativo sobre el contenido de la imagen.
- **height**: especifica la altura (en píxeles o en porcentaje) con la que se mostrará la imagen en el navegador. Por defecto, se muestra la de la imagen original.
- **width**: especifica la anchura con la que se muestra la imagen.

Puedes encontrar la lista exhaustiva de atributos del tag img en <http://html.conclase.net/w3c/html401-es/struct/objects.html#edef-IMG>.

Parte 2: Formularios en HTML.

Hasta ahora, todos los elementos que hemos visto en páginas web eran estáticos, desde el punto de vista de que sólo nos permitían mostrar información. Los formularios nos permiten también recoger información de los usuarios de nuestra página web. La forma de recoger esa información puede ser variada. Por ejemplo, puede haber un programa en el servidor (por ejemplo, en php, jsp...) que se encargue de recoger la información y enviárnosla a nosotros. También podemos hacer que la información

recogida en el formulario nos sea enviada por correo electrónico. En esta sección veremos los distintos tipos de tags y sus atributos que nos permitirán definir formularios.

El tag que permite definir formularios en HTML es form:

- `<form>...</form>`: el mismo debe especificar por medio de atributos algunas de las cuestiones relevantes sobre el formulario:
 - o `action`: debe contener la url del procesador del formulario. Por ejemplo, `<form action="http://miweb/programa"...>`. También podemos usar `action="mailto:midireccion@correo"` para que el formulario sea enviado por mail.
 - o `method`: especifica el método de envío de la información, entre GET y POST. Puedes ver sus diferencias en <http://html.conclase.net/w3c/html401-es/interact/forms.html#submit-format>.
 - o `accept-charset`: codificación de caracteres que debe ser aceptada por el servidor para manejar el formulario enviado.Existen algunos atributos adicionales cuya definición puedes encontrar en <http://html.conclase.net/w3c/html401-es/interact/forms.html#h-17.3>.

Además, un formulario puede contener como elementos cualquiera de los elementos de nivel de bloque, así como controles. Los controles son la forma de interacción del usuario de la web con la misma. Algunos de los controles más habituales son:

- `<input>`: permite definir diferentes controles. Está prohibido incluir tag de cierre. Sus atributos principales son:
 - o `type`: puede tomar el valor "text" (para un campo de texto de una línea), "password" (un campo contraseña), "checkbox" (casilla de verificación), "radio" (botón de radio), "submit" (botón de envío del formulario), "reset" (botón de reinicio del formulario). Puedes encontrar ejemplos de cómo se muestra cada uno de los tipos en http://www.w3schools.com/html/html_forms.asp.
 - o `name`: asigna el nombre al control. Se usa al enviar el formulario al destinatario del mismo.
 - o `value`: define el valor inicial del control.
 - o `size`: define el tamaño del control. Por ejemplo, para los campos de texto define el número de caracteres inicial.
 - o `checked`: para los campos de tipo "radio" o "checkbox" especifica que el botón esté marcado.En <http://html.conclase.net/w3c/html401-es/interact/forms.html#h-17.4.2> puedes encontrar ejemplos de controles creados con input.
- `<select>...</select>`: permite crear menús de selección simple o múltiple. Debe contener como elemento uno o varios elementos `<option>` que detallamos un poco más adelante. Veamos primero los atributos que lo componen:
 - o `name`: nombre del control.
 - o `size`: número de filas que son visibles del menú (puede ser distinto del número de opciones que contiene el menú).
 - o `multiple`: valor booleano ("true" o "false") que permite seleccionar varias opciones de forma simultánea. Por defecto, su valor es "false" (y la selección es simple).

Un tag select estará formado por uno o varios elementos <option>:

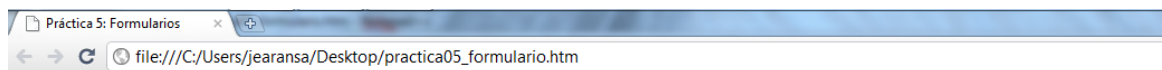
- <option>...</option>: Cada elemento option representa una de las opciones que se muestran en el menú. Debe contener una secuencia de caracteres. Como atributos dispone de los siguientes:
 - o selected: especifica que la opción indicada está preseleccionada.
 - o value: define el valor que se envía al usuario por esa opción. Si no se detalla, el valor será el texto del elemento option.

En la página http://www.pcweb.es/manual_html/mostrar.php?opcion=select puedes encontrar algunos ejemplos de formularios select con selección simple y múltiple de opciones.

- <textarea>...</textarea>: a diferencia de <input type="text"> (que genera un campo de texto de una línea), textarea permite definir un campo de texto de varias líneas. Lo que escribamos dentro del control será el valor inicial de la caja de texto. Admite los siguientes atributos:
 - o name: nombre del control.
 - o rows: número de líneas de texto visibles de la caja de texto.
 - o cols: anchura visible de la caja de texto en número de caracteres.

En la página <http://es.kioskea.net/contents/html/htmlform.php3> puedes encontrar un tutorial sobre formularios web y ejemplos de cada uno de los elementos anteriores.

3. Con las anteriores herramientas, crea un formulario en un fichero practica05_formulario.htm (incluye en el mismo cabeceras y metas como vimos en la práctica 4) que responda al siguiente aspecto. Puedes encontrar la lista de Departamentos de la UR en <http://www.unirioja.es/dptos/> y la lista de edificios en <http://www.unirioja.es/universidad/presentacion/campus.shtml>:



ENCUESTA CAMPUS VIRTUAL

NOMBRE:	
DEPARTAMENTO:	Agricultura y Alimentación
EDIFICIO:	Edificio Vives
E-MAIL:	
TELÉFONO:	
¿Utiliza habitualmente el campus virtual?	si: ☒ no: ☐
¿Estaria interesado/a en recibir formación personalizada sobre las posibilidades de esta herramienta informática?	si: ☒ no: ☐

CALENDARIO APROXIMADO DE FORMACIÓN

TURNO PREFERENTE:	☒ Mañana ☐ Tarde
MES PREFERENTE:	Octubre
HORARIO PREFERENTE:	9:00
COMENTARIOS:	

