

TABLAS EN HTML

Una tabla es un conjunto de celdas organizadas dentro de las cuales podemos alojar distintos contenidos. HTML dispone de una gran variedad de etiquetas para crear tablas, con sus atributos.

En un principio nos podría parecer que las tablas son raramente útiles y que pueden ser utilizadas principalmente para listar datos como agendas, resultados y otros datos de una forma organizada. En general, sirven para representar información tabulada, en filas y columnas. Esto es una realidad en los últimos años, desde que las tablas se han descartado para fines relacionados con la maquetación.

Como veremos a continuación, existen diversas etiquetas que se deben utilizar en una forma determinada para la creación de tablas. Por ello, puede que en un principio nos resulte un poco complicado trabajar con estas estructuras, pero, con un poco de práctica podremos crear tablas con absoluta soltura. Si deseamos mostrar datos de una manera sencilla de leer, dispuestos en filas y columnas, tarde o temprano observaremos que las tablas son la mejor solución y apreciaremos las posibilidades nos ofrecen.

ETIQUETAS BÁSICAS PARA TABLAS EN HTML

Para empezar, nada más sencillo que por el principio: las tablas son definidas por las etiquetas TABLE y su cierre.

Dentro de estas dos etiquetas colocaremos todas las otras etiquetas de las tablas, hasta llegar a las celdas. Dentro de las celdas ya es permitido colocar textos e imágenes que darán el contenido a la tabla.

Las tablas son descritas por líneas de arriba a abajo (y luego por columnas de izquierda a derecha). Cada una de estas líneas, llamada fila, es definida por otra etiqueta y su cierre: TR (table row).

Asimismo, dentro de cada línea, habrá diferentes celdas. Cada una de estas celdas será definida por otra etiqueta: TD (table data). Dentro de ésta y su cierre será donde coloquemos nuestro contenido, el contenido de cada celda.

Aquí tenéis un ejemplo de estructura de tabla:

```
<table>
<tr>
  <td>Celda 1, línea 1</td>
  <td>Celda 2, línea 1</td>
</tr>
<tr>
  <td>Celda 1, línea 2</td>    <td>
Celda 2, línea 2</td>
</tr>
</table>
```

Esto en una página se vería así:

Celda 1, línea 1 Celda 2, línea 1

Celda 1, línea 2 Celda 2, línea 2

También existe la etiqueta TH (table header), que sirve para crear una celda cuyo contenido esté formateado como un título o cabecera de la tabla. En la práctica, lo que hace es poner en negrita y centrado el contenido de esa celda, lo que se puede conseguir aplicando las correspondientes etiquetas dentro de la celda.

Aquí tenéis un ejemplo de estructura de tabla con la etiqueta th:

```
<table>
<tr>
  <th>Titulo Celda 1</th>
  <th> Titulo Celda 2</th>
</tr>
<tr>
  <td>Celda 1, línea 1</td>
  <td> Celda 2, línea 1</td>
</tr>
<tr>
  <td> Celda 1, línea 2</td>    <td>
Celda 2, línea 2</td>
</tr>
</table>
```

Esto en una página se vería así:

Titulo Celda 1 Titulo Celda 2

Celda 1, línea 1 Celda 2, línea 1

Celda 1, línea 2 Celda 2, línea 2

ATRIBUTOS PARA TABLAS, FILAS Y CELDAS

A partir de esta idea simple y sencilla, las tablas adquieren otra magnitud cuando les incorporamos toda una cantidad de atributos aplicados sobre cada tipo de etiquetas que las componen.

En cuanto a atributos para tabla hay unos cuantos. Muchos los conoces ya de otras etiquetas, como width, height, align, etc. Hay otros que son especialmente creados para las etiquetas TABLE.

- cellpadding: es el espacio entre el borde de la celda y su contenido.
- cellspacing: es el espacio entre celdas de la tabla.

- **border:** es el número de píxeles que tendrá el borde de la tabla.
- **bordercolor:** es el rgb que le vas a asignar al borde de la tabla.

En cuanto a las etiquetas "interiores" de una tabla, nos referimos a TR y TD, ten en cuenta:

- Podemos usar prácticamente cualquier tipo de etiqueta dentro de la etiqueta TD para, de esta forma, escribir su contenido.
- Las etiquetas situadas en el interior de la celda no modifican el resto del documento.
- Las etiquetas de fuera de la celda no son tenidas en cuenta por ésta.

Así pues, podemos especificar el formato de nuestras celdas a partir de etiquetas introducidas en su interior o mediante atributos colocados dentro de la etiqueta de celda TD o bien, en algunos casos, dentro de la etiqueta TR, si deseamos que el atributo sea válido para toda la línea. La forma más útil y actual de dar forma a las celdas es a partir de las hojas de estilo en cascada que ya tendréis la oportunidad de abordar más adelante.

Veamos a continuación algunos atributos útiles para la construcción de nuestras tablas. Empecemos viendo atributos que nos permiten modificar una celda en concreto o toda una línea:

- **align:** Justifica el texto de la celda del mismo modo que si fuese el de un párrafo.
- **valign:** Podemos elegir si queremos que el texto aparezca arriba (top), en el centro (middle) o abajo (bottom) de la celda.
- **bgcolor:** Da color a la celda o línea elegida.
- **bordercolor:** Define el color del borde.

Otros atributos que pueden ser únicamente asignados a una celda y no al conjunto de celdas de una línea son:

- **background:** Nos permite colocar un fondo para la celda a partir de un enlace o una imagen.
- **height:** Define la altura de la celda en píxeles o porcentaje.
- **width:** Define la anchura de la celda en píxeles o porcentaje.
- **colspan:** Expande una celda horizontalmente.
- **rowspan:** Expande una celda verticalmente.

Estos últimos cuatro atributos descritos son de gran utilidad. Concretamente, height y width nos ayudan a definir las dimensiones de nuestras celdas de una forma absoluta (en píxeles o puntos de pantalla) o de una forma relativa, es decir por porcentajes referidos al tamaño total de la tabla.

Los atributos rowspan y colspan son también utilizados frecuentemente. Gracias a ellos es posible expandir celdas fusionando éstas con sus vecinas. El valor que pueden tomar estas etiquetas es numérico. El número representa la cantidad de celdas fusionadas.

Así:

```
<td colspan="2">
```

Esta celda tiene un colspan 2
Celda 1, línea 2 Celda 2, línea 2

Por otro lado:

`<td rowspan="2">`

Esta celda tiene rowspan="2", por eso tiene fusionada la celda de abajo.	Celda Normal Otra celda normal
--	--

El resto de los atributos presentados presentan una utilidad y uso bastante obvios. Los dejamos a vuestra propia investigación.

FORMULARIOS HTML

Hasta ahora hemos visto la forma en la que el HTML gestiona y muestra la información, esencialmente mediante texto, imágenes y enlaces. Nos queda por ver de qué forma podemos intercambiar información con nuestro visitante. Desde luego, este nuevo aspecto resulta primordial para gran cantidad de acciones que se pueden llevar a cabo mediante la Web: comprar un artículo, rellenar una encuesta, enviar un comentario al autor, registrar un usuario, etc.

Los formularios son esas famosas cajas de texto y botones que podemos encontrar en muchas páginas web. Son muy utilizados para realizar búsquedas o bien para introducir datos personales por ejemplo en sitios de comercio electrónico. Los datos que el usuario introduce en estos campos son enviados al correo electrónico del administrador del formulario o bien a un programa que se encarga de procesarlo automáticamente. Nosotros en esta guía no vamos a mostrar como enviar la información al mail, ya que nos interesa, más adelante poder manejar esa información.

QUÉ SE PUEDE HACER CON UN FORMULARIO

Usando HTML podemos únicamente enviar el contenido del formulario a un correo electrónico, es decir, construir un formulario con diversos campos y, a la hora pulsar el botón de enviar, generar un mensaje de que se ha registrado con éxito la información.

Pero para todo lo que sea manejar esa información y guardarla, por ejemplo, en una base de datos vamos a tener que utilizar Java. Como lo haremos lo veremos más adelante en el curso.

Así pues, en resumen, con HTML podremos construir los formularios, con diversos tipos de campos, como cajas de texto, botones de radio, cajas de selección, menús desplegables, etc. Sin embargo, debe quedar claro que desde HTML no se puede manejar esta información para guardarla o enviarla a algún correo, etc. Eso será trabajo de Java.

CÓMO HACER UN FORMULARIO EN HTML

Los formularios son definidos por medio de las etiquetas FORM y su cierre. Entre estas dos etiquetas colocaremos todos los campos y botones que componen el formulario. Dentro de esta etiqueta FORM debemos especificar algunos atributos:

action: define el tipo de acción a llevar a cabo con el formulario. Como ya hemos dicho, existen dos posibilidades:

- El formulario es enviado a una dirección de correo electrónico. Para esto hay que poner el mail en el action.
- El formulario es enviado a un programa o script que procesa su contenido. Esta es la posibilidad que más nos interesa.

```
<form action="ruta del método que va a manejar la información"></form>
```

method: Este atributo se encarga de especificar la forma en la que el formulario es enviado. Los dos valores posibles que puede tomar este atributo son POST y GET. A efectos prácticos y, salvo que se os diga lo contrario, daremos siempre el valor POST. Estos conceptos de POST y GET, lo veremos más adelante en el curso.

enctype: Se utiliza para indicar la forma en la que viajará la información que se mande por el formulario. En el caso más corriente, enviar el formulario por correo electrónico, el valor de este atributo debe de ser "text/plain". Así conseguimos que se envíe el contenido del formulario como texto plano dentro del email. Si fuéramos a enviar una imagen dentro del formulario, este atributo debería ser "multipart/form-data". También todos estos conceptos vamos a verlos más adelante.

Este último atributo puede que esté como que no esté, las otras dos si vamos a guardar la información de nuestro formulario en Java, van a estar siempre.

EJEMPLO DE ETIQUETA FORM COMPLETA

Entonces con todo lo anterior ya explicado, la etiqueta completa nos quedaría así:

```
<form action="ruta del método que va a manejar la información" method="POST"
enctype="multipart/form-data"></form>
```

Entre esta etiqueta y su cierre colocaremos el resto de etiquetas que darán forma a nuestro formulario.

CAMPOS DE TEXTO

El lenguaje HTML nos propone una gran diversidad de alternativas a la hora de crear nuestros formularios, es decir, una gran variedad de elementos para diferentes propósitos. Estas van desde la clásica caja de texto, hasta la lista de opciones en un menú desplegable, pasando por las cajas de validación, etc.

Las etiquetas que tenemos que utilizar para crear campos de texto, pueden ser de dos tipos. Veamos en qué consiste cada una de estas modalidades y cómo podemos implementarlas en nuestro formulario.

ETIQUETA INPUT

Las cajas de texto son colocadas por medio de la etiqueta INPUT. Dentro de esta etiqueta hemos de especificar el valor de dos atributos: type y name.

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

Como todos sabrán un input se ve así:

De este modo expresamos nuestro deseo de crear una caja de texto cuyo contenido será llamado "nombre" (por ejemplo, en el caso de la etiqueta anterior, pero podemos poner distintos nombres a cada uno de los campos de texto que habrá en los formularios).

ATRIBUTO TYPE

Como hemos visto el atributo type nos sirve para especificar el tipo de dato que se va a ingresar en nuestro input, en el ejemplo anterior lo habíamos puesto como tipo text, para que sea una caja de texto y poder ingresar texto. Pero existen otros tipos de valores para el atributo type

NUMBER

Este tipo permite al usuario ingresar números. Los navegadores vienen con validaciones para evitar que el usuario ingrese algo que no sea números. Además, en los navegadores modernos, los campos numéricos suelen venir con controles que permiten a los usuarios cambiar su valor de forma gráfica.

`<input type="number">` Se

vería así:

DATE

Este le permite al usuario ingresar una fecha, ya sea mediante una caja de texto o una interfaz gráfica con selector de fecha.

`<input type="date">` Se vería

así:

A calendar interface for January 2008. The header shows "03/01/2008" and a calendar icon. Below the header, the month "enero de 2008" is displayed with a dropdown arrow. The calendar grid shows days of the week (L, M, M, J, V, S, D) and dates. The date "3" is highlighted in blue. The word "Hoy" is at the bottom right.

EMAIL

Este tipo permite al usuario ingresar un mail. Los navegadores vienen con validaciones para validar que se esté ingresando con el formato correcto de un mail. Este input se va a ver como un input de texto común y corriente.

```
<input type="email">
```

TEXTO OCULTO

Hay determinados casos en los que podemos desear esconder el texto escrito en el campo input, por medio de círculos negros, de manera que aporte una cierta confidencialidad. Para esto vamos a usar el type password.

```
<input type="password">
```

Se vería así:

Más adelante veremos otros valores para el atributo type con otras utilidades

ATRIBUTO NAME

Si vemos de nuevo el ejemplo del principio:

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

En este ejemplo creamos una caja de texto cuyo contenido será llamado "nombre", elegimos nombre, pero podemos ponerles el nombre que queramos.

El nombre del elemento del formulario es de gran importancia para poder identificarlo en nuestro programa de procesamiento (Java).

Además de estos dos atributos, esenciales para el correcto funcionamiento de nuestra etiqueta, existen otra serie de atributos que pueden resultarnos de utilidad pero que no son imprescindibles:

size: define el tamaño de la caja de texto, en número de caracteres visibles. Si al escribir el usuario llega al final de la caja, el texto que escriba a continuación también cabrá dentro del campo, pero irá desfilando, a medida que se escribe, haciendo desaparecer la parte de texto que queda a la izquierda.

maxlength: indica el tamaño máximo del texto, en número de caracteres, que puede ser escrito en el campo. En caso que el campo de texto tenga definido el atributo maxlength, el navegador no permitirá escribir más caracteres en ese campo que los que hayamos indicado.

value: en algunos casos puede resultarnos interesante asignar un valor definido al campo en cuestión. Esto puede ayudar al usuario a rellenar más rápidamente el formulario o darle alguna idea sobre la naturaleza de datos que se requieren. Este valor inicial del campo puede ser expresado mediante el atributo value. Veamos su efecto con un ejemplo sencillo:

```
<input type="text" name="instituto" value="Wikipedia">
```

placeholder: este atributo especifica una pequeña pista que describe el valor esperado de para el campo (input).

La pequeña sugerencia se muestra en el campo de entrada antes de que el usuario ingrese un valor. Una vez que escriba, esa pista va a desaparecer.

`<input type="text" name="nombre" placeholder="Nombre del usuario">` Genera un campo de este estilo:

Nombre del usuario

Nota: recordemos que todos estos ejemplos de input deben ir entre las etiquetas de apertura y de cierre form.

`<form>`

`<input type="text" name="instituto" value="Club San Martín">`

`</form>`

ETIQUETA TEXTAREA PARA TEXTO LARGO

Si deseamos poner a la disposición de usuario un campo de texto donde pueda escribir cómodamente sobre un espacio compuesto de varias líneas, hemos de invocar una nueva etiqueta: TEXTAREA y su cierre correspondiente.

Este tipo de campos son prácticos cuando el contenido a enviar no es un nombre, teléfono, edad o cualquier otro dato breve, sino más bien, un comentario, opinión, etc. en los que existe la posibilidad que el usuario desee rellenar varias líneas.


Dentro de la etiqueta textarea deberemos indicar, como para el caso visto anteriormente, el atributo name para asociar el contenido a un nombre que será asemejado a una variable en un lenguaje de programación. Además, podemos definir las dimensiones del campo a partir de los atributos siguientes:

- rows: define el número de líneas del campo de texto.
- cols: define el número de columnas del campo de texto. La etiqueta

queda por tanto de esta forma:

`<textarea name="comentario" rows="10" cols="40"></textarea>` El

resultado es el siguiente:



Asimismo, es posible predefinir el contenido del campo. Para ello, no usaremos el atributo value, sino que escribiremos dentro de la etiqueta el contenido que deseamos atribuirle. Veámoslo:

```
<textarea name="comentario" rows="10" cols="40">Escribe tu comentario...</textarea>
```

Esta etiqueta al igual que el input debe ir dentro de la etiqueta form.

ETIQUETA LABEL

El elemento LABEL y su etiqueta de cierre, provee una descripción corta para el campo de texto y que puede ser asociada a un campo de texto. Podemos asociar una etiqueta label a un campo de texto para que el usuario pueda acceder al campo de texto con solo clicar el label. También, como veremos más adelante, cuando veamos las cajas de opciones, clicar en el nombre de la opción para acceder a ella, “tickear” esa opción.

La etiqueta se ve de esta forma:

```
<label>Nombre del Usuario</label> <input  
type="text" name="nombre">
```

Esto en una página se vería así:

Nombre del Usuario

Podríamos poner un salto de línea para que el label quede arriba del input, si lo quisiéramos.

Nombre del Usuario

ATRIBUTO LABEL

La etiqueta label solo consta del atributo for. Mediante la utilización del atributo for podemos asociar el label con el input. Para lograr esto vamos a tener que utilizar también el atributo ID, este atributo lo explicamos previamente y lo vamos a ver más en detalle en la parte de CSS.

La manera que anclamos un label a un input es, al label le vamos a dar un valor en su atributo for, este va a representar el dato que se va a ingresar en el input y en el input vamos a poner el mismo valor, pero en el atributo ID. Entonces, el primer elemento input en el documento con un ID que coincida con el dispuesto en el atributo for puesto en el label, será el control etiquetado para este elemento.

Esto se vería así:

```
<label for="nombre">Nombre del Usuario</label>
```

```
<input type="text" id="nombre" name="nombre">
```

El label y el input se verán igual pero ahora cuando el usuario clickee el label se va a activar el campo de texto del input para poder ingresar el valor que el usuario necesite. Después vamos a ver un ejemplo más útil con las cajas de opciones.

OTROS ELEMENTOS DE FORMULARIOS

Seguramente hayan notado que los inputs son una manera muy práctica de hacernos llegar la información del navegante. No obstante, en muchos casos, permitir al usuario que escriba cualquier texto permite demasiada libertad y puede que la información que éste escriba no sea la que nosotros estamos necesitando.

Por ejemplo, pensemos que queremos que el usuario indique su país de residencia. En ese caso podríamos ofrecer una lista de países para que seleccione el que sea. Este mismo caso se puede aplicar a gran variedad de informaciones, como el tipo de tarjeta de crédito para un pago, la puntuación que da a un recurso, si quiere recibir o no un boletín de novedades, etc...

Este tipo de opciones predefinidas por nosotros pueden ser expresadas por medio de diferentes campos de formulario. Veamos a continuación cuales son:

LISTAS DE OPCIONES

Las listas de opciones son ese tipo de menús desplegables que nos permiten elegir una (o varias) de las múltiples opciones que nos proponen. Para construirlas emplearemos una etiqueta SELECT, con su respectivo cierre.

Como para los casos ya vistos, dentro de esta etiqueta definiremos su nombre por medio del atributo name. Cada opción será incluida en una línea precedida de la etiqueta OPTION.

Podemos ver, a partir de estas explicaciones, la forma más típica y sencilla de esta etiqueta:

```
<select name="estación">
```

```
<option>Primavera</option>
```

```
<option>Verano</option>
```

```
<option>Otoño</option>
```

```
<option>Invierno</option>
```

```
</select>
```

Esto en una página se vería así:



Y cuando el usuario clickea en el select muestra las opciones así:



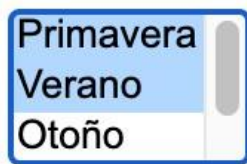
Esta estructura puede verse modificada principalmente a partir de otros dos atributos:

size: indica el número de valores mostrados a la vez en la lista. Lo típico es que no se incluya ningún valor en el atributo size, en ese caso tendremos un campo de opciones desplegable, pero si indicamos un valor para el atributo size aparecerá un campo donde veremos las opciones definidas por size y el resto podrán ser vistos por medio de la barra lateral de desplazamiento.

multiple: permite la selección de más elementos de la lista. Este atributo se expresa sin valor alguno, es decir, no se utiliza con el igual, simplemente se pone para conseguir el efecto, o no se pone si queremos una lista desplegable común.

```
<select name="estación" size="3" multiple>
```

Esto en una página se vería así:



La etiqueta `OPTION` puede asimismo ser modificada por medio de otros atributos.

selected: del mismo modo que multiple, este atributo no toma ningún valor, sino que simplemente indica que la opción que lo presenta esta elegida por defecto.

```
<option selected>Otoño</option>
```

BOTONES DE RADIO

Existe otra alternativa para plantear una elección, en este caso, obligamos al usuario a elegir únicamente una de las opciones que se le proponen.

La etiqueta empleada en este caso es `INPUT` en la cual usaremos el atributo `type` con el de valor `radio`. Este atributo colocara una casilla pinchable al lado del valor del input. Veamos un ejemplo:

```
<input type="radio" name="estación" value="1">Primavera  
<br>  
<input type="radio" name="estación" value="2">Verano  
<br>  
<input type="radio" name="estación" value="3">Otoño  
<br>  
<input type="radio" name="estación" value="4">Invierno
```

Esto en una página se vería así:

- ☒ Primavera
- ☐ Verano
- ☐ Otoño
- ☐ Invierno

En este tipo de input para elegir una opción debemos tocar en la casilla clickeable, pero habíamos explicado previamente en la etiqueta label, que podíamos hacer que la etiqueta label al clickearla se active la caja de texto del input. Ahora, podemos usar eso para que cuando el usuario clickee la palabra primavera se seleccione esa opción. Esto se vería así:

```
<input type="radio" id="primavera" name="estación" value="1">
<label for="primavera">Primavera</label>
<br>
<input type="radio" id="verano" name="estación" value="2">
<label for="verano">Verano</label>
<br>
<input type="radio" id="otoño" name="estación" value="3">
<label for="otoño">Otoño</label>
<br>
<input type="radio" id="invierno" name="estación" value="4">
<label for="invierno">Invierno</label>
```

Esto en la página se verá igual que el anterior:

- ☐ Primavera
- ☒ Verano
- ☐ Otoño
- ☐ Invierno

La única diferencia va a ser que el usuario va a poder clickear el nombre de la estación que quiere para seleccionar esa opción, además de poder clickear la casilla.

CAJAS DE VALIDACIÓN

Este tipo de elementos pueden ser activados o desactivados por el visitante por un simple clic sobre la caja en cuestión. Para esto vamos a usar la etiqueta INPUT con el valor checkbox en el atributo type.

```
<input type="checkbox" name="estación" value="1">Primavera
```

Esto se verá así:

- ☐ Primavera
- ☐ Verano
- ☐ Otoño
- ☐ Invierno

ENVÍO, BORRADO Y DEMÁS

Ha llegado el momento de explicar cómo podemos hacer un botón para provocar el envío del formulario, entre otras cosas.

Como podremos imaginarnos, en formularios no solamente habrá elementos o campos donde solicitar información del usuario, sino también habrá que implementar otra serie de funciones. Concretamente, han de permitirnos su envío mediante un botón. También puede resultar práctico poder proponer un botón de borrado o bien acompañar el formulario de datos ocultos que puedan ayudarnos en su procesamiento.

BOTÓN DE ENVÍO DE FORMULARIO (BOTÓN DE SUBMIT)

Para dar por finalizado el proceso de relleno del formulario y hacerlo llegar a su gestor, el usuario ha de enviarlo por medio de un botón previsto a tal efecto. Para esto vamos a utilizar la etiqueta `BUTTON` y su respectivo cierre. Dentro el elemento `button` se puede poner texto (y etiquetas como `<i>`, ``, ``, `
`, ``, etc.). Se vería así:

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

Esto en la página se verá así:



Como puede verse, tan solo hemos de especificar que se trata de un botón de envío (`type="submit"`) y hemos de definir el mensaje que queremos que aparezca escrito en el botón.

BOTÓN DE BORRADO (BOTÓN DE RESET)

Este botón nos permitirá borrar el formulario por completo, en el caso de que el usuario desee rehacerlo desde el principio. Su estructura sintáctica es parecida a la anterior:

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

A diferencia del botón de envío, indispensable en cualquier formulario, el botón de borrado resulta meramente optativo y no es utilizado frecuentemente. Hay que tener cuidado de no ponerlo muy cerca del botón de envío y de distinguir claramente el uno del otro, para que ningún usuario borre el contenido del formulario que acaba de rellenar por error.

BOTONES NORMALES

Dentro de los formularios también podemos colocar botones normales, pulsables como cualquier otro botón. Estos botones por si solos no tienen mucha utilidad, pero podremos necesitarlos para realizar acciones en el futuro. Su sintaxis es la siguiente:

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

DATOS OCULTOS (CAMPOS HIDDEN)

En algunos casos, aparte de los propios datos rellenos por el usuario, puede resultar práctico enviar datos definidos por nosotros mismos que ayuden al programa en su procesamiento del formulario. Este tipo de datos, que no se muestran en la página, pero si pueden ser detectados solicitando el código fuente, no son frecuentemente utilizados por páginas construidas en HTML, son más bien usados por páginas que emplean tecnologías de servidor. No se asusten, veremos más adelante qué quiere decir esto. Tan solo queremos dar constancia de su existencia y de su modo creación. He aquí un ejemplo:

```
<input type="hidden" name="instituto" value="Club San Martín">
```

Esta etiqueta, incluida dentro de nuestro formulario, enviara un dato adicional al programa encargado de la gestión del formulario.

EJEMPLO COMPLETO DE FORMULARIO

Con esto último finalizamos el tema de formularios. Pasemos ahora a ejemplificar todo lo aprendido a partir de la creación de un formulario.

```
<form action="ruta del método que va a manejar la información" method="POST"
enctype="multipart/form-data"></form>
```

```
    <label>Nombre del usuario</label> <br>
    <input type="text" name="nombre"> <br>
    <label>Edad del usuario</label> <br>
    <input type="number" name="edad "> <br>
    <label>Fecha de nacimiento del usuario</label> <br>
    <input type="date" name="fechanac"> <br>
    <label>Sexo del usuario</label> <br>
    <input type="radio" name="sexo" value="Hombre"> Hombre <br>
    <input type="radio" name="sexo" value="Mujer"> Mujer <br>
    <label>País nacimiento del usuario</label> <br>
    <select name="país">
        <option>Argentina</option>
        <option>Brasil</option>
        <option>Chile</option>
        <option>Uruguay</option>
    </select>
    <br>
    <button type="submit">Enviar</button>
    <button type="reset">Borrar</button>
```

Este formulario se verá así:

Nombre del usuario

Edad del usuario

Fecha de nacimiento del usuario

Sexo del usuario

☐ Hombre

☐ Mujer

País nacimiento del usuario

 ▼

Enviar

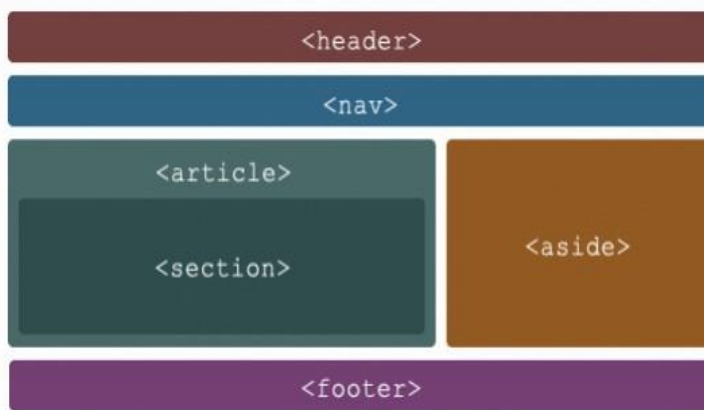
Borrar

SECCIONES EN HTML

Las páginas web se trabajan con lo que se conoce como un esquema. El esquema (outline) de una página web es un índice de los apartados de una página web que muestra la relación de jerarquía entre los diferentes apartados y subapartados. El concepto de esquema se formalizó en HTML 5 con más precisión que en HTML 4 / XHTML 1 y explica algunas características y formas de utilización de las etiquetas de secciones y bloques de contenido.

En relación a esto se pensó que las páginas de HTML se pueden dividir en secciones y en HTML 5 se introdujo una serie de etiquetas que nos van a ayudar con la división de nuestra página en secciones. Dentro de cada sección van a haber más etiquetas, esto es simplemente para que podemos tener un índice de los apartados de la página web.

La página dividida en secciones con sus respectivas etiquetas se ve así:



```
<body>
  <header>
    <a href="/"><img src=logo.png alt="home"></a>
    <hgroup>
      <h1>Title</h1>
      <h2 class="tagline">
        A lot of effort went into making this effortless.
      </h2>
    </hgroup>
  </header>
  <nav>
    <ul>
      <li><a href="#">home</a></li>
      <li><a href="#">blog</a></li>
      <li><a href="#">gallery</a></li>
      <li><a href="#">about</a></li>
    </ul>
  </nav>
  <section class="articles">
    <article>
      <time datetime="2009-10-22">October 22, 2009</time>
      <h2>
        <a href="#" title="link to this post">Travel day</a>
      </h2>
      <div class="content">
        Content goes here...
      </div>
      <section class="comments">
        <p><a href="#">3 comments</a></p>
      </section>
    </article>
  </section>
  <aside>
    <div class="related"></div>
```

<section>: se utiliza para representar una sección "general" dentro de un documento o aplicación, como un capítulo de un libro. Puede contener subsecciones y si lo acompañamos de h1-h6 podemos estructurar mejor toda la página creando jerarquías del contenido, algo muy favorable para el buen posicionamiento web.

<article>: representa un componente de una página que consiste en una composición autónoma en un documento, página, aplicación, o sitio web con la intención de que pueda ser reutilizado y repetido.

<aside>: representa una sección de la página que abarca un contenido relacionado con el contenido que lo rodea, por lo que se le puede considerar un contenido independiente. Este elemento puede utilizarse para efectos tipográficos, barras laterales, elementos publicitarios u otro contenido que se considere separado del contenido principal de la página.

<header>: representa un grupo de artículos introductorios o de navegación. Está destinado a contener por lo general la cabecera de la sección (un elemento h1-h6 o un elemento hgroup).

<nav>: representa una sección de una página que enlaza a otras páginas o a otras partes dentro de la página. No todos los grupos de enlaces en una página necesita estar en un elemento nav, sólo las secciones que constan de bloques de navegación principales son apropiadas para el elemento de navegación.

<footer>: representa el pie de una sección, con información acerca de la página/sección que poco tiene que ver con el contenido de la página, como el autor, el copyright o el año.

<hgroup>: representa el encabezado de una sección. El elemento se utiliza para agrupar un conjunto de elementos h1-h6 cuando el título tiene varios niveles, tales como subtítulos o títulos alternativos.

ETIQUETAS EXTRAS

En este apartado que va a ser el último de nuestra parte de html vamos a ver unas etiquetas que no hemos visto todavía y que son importantes.

ETIQUETA DIV

La etiqueta div se conoce como etiqueta de división. La etiqueta div se usa en HTML para hacer divisiones de contenido en la página web como (texto, imágenes, encabezado, pie de página, barra de navegación, etc.). La etiqueta Div tiene etiquetas de apertura (<div>) y de cierre (</div>) y es obligatorio cerrar la etiqueta. Div es la etiqueta más útil en el desarrollo web porque nos ayuda a separar datos en la página web y podemos crear una sección particular para datos o funciones particulares en las páginas web. Cabe aclarar que la etiqueta div genera un salto de línea.

- La etiqueta Div es una etiqueta de nivel de bloque
- Es una etiqueta de contenedor genérica
- Se utiliza para agrupar varias etiquetas de HTML para que se puedan crear secciones y aplicarles estilo.

Un ejemplo que podríamos usar para la etiqueta div es, supongamos que tenemos 3 párrafos que queremos alinear a la izquierda, esto recordemos lo haríamos con el atributo align. Nosotros haríamos algo así:

```
<p align="left">Párrafo 1</p>
<p align="left">Párrafo 2</p> <p
align="left">Párrafo 3</p>
```

Una forma de simplificar nuestro código anterior y de evitar introducir continuamente el atributo align sobre cada una de nuestras etiquetas es utilizando la etiqueta DIV. Vamos a usar un div para generar una sección de todos los párrafos y le pones el atributo align al div.

Esto se vería así:

```
<div align="left">
<p>Párrafo 1</p>
<p>Párrafo 2</p>
<p>Párrafo 3</p>
</div>
```

Como hemos visto, la etiqueta DIV marca divisiones en las que definimos un bloque de contenido, y a los que podríamos aplicar estilo de manera global, aunque lo correcto sería aplicar ese estilo del lado del CSS.

ETIQUETA SPAN

El elemento span HTML es un contenedor en línea genérico para elementos y contenido en línea. Solía agrupar elementos con fines de estilo (mediante el uso de los atributos de clase o id). La mejor manera de usarlo es cuando no hay ningún otro elemento semántico disponible. span es muy similar a la etiqueta div, pero div es una etiqueta a nivel de bloque y span es una etiqueta en línea. La etiqueta Span es una etiqueta emparejada, lo que significa que tiene una etiqueta de apertura (<) y de cierre (>), y es obligatorio cerrar la etiqueta.

- La etiqueta span se utiliza para agrupar elementos en línea.
- La etiqueta span no realiza ningún cambio visual por sí misma.
- span es muy similar a la etiqueta div, pero div es una etiqueta a nivel de bloque y span es una etiqueta en línea.

Un ejemplo de la etiqueta span, es poner una parte de un párrafo de un color concreto, ya que es una etiqueta en línea, la podemos meter dentro de una etiqueta p.

```
<p>My mother has <span style="color: blue">blue</span> eyes.</p>
```

Esto en una página se vería así:

My mother has blue eyes.

La etiqueta span no crea un salto de línea similar a una etiqueta div, sino que permite al usuario separar cosas de otros elementos a su alrededor en una página dentro de la misma línea. Al evitar el salto de línea, solo da como resultado el texto seleccionado para cambiar, manteniendo todos los demás elementos a su alrededor iguales.

Nota: en el apartado de CSS vamos a ver mejor el atributo style y el atributo color. Ahora los estamos usando para el ejemplo.

PREGUNTAS DE APRENDIZAJE

- 1) ¿Qué significa el acrónimo HTML?:
 - a) Hyper Text Mask Language.
 - b) Hard Text Markup Language.
 - c) Hyper Text Markup Language.
 - d) Hard Text Mask Language.
- 2) HTML es un lenguaje:
 - a) De programación.
 - b) De marcado.
 - c) Hablado.
 - d) Ninguno de los anteriores.
- 3) La etiqueta que encierra todo el contenido de una página es:
 - a) <head></head>
 - b) <body></body>
 - c) !DOCTYPE
 - d) <html></html>
- 4) La etiqueta que es el contenedor de todo aquello que se va incluir en el html es:
 - a) <head></head>
 - b) <body></body>
 - c) !DOCTYPE
 - d) <html></html>
- 5) La etiqueta que encierra todo el contenido visible al usuario es:
 - a) <head></head>
 - b) <body></body>
 - c) !DOCTYPE
 - d) <html></html>
- 6) ¿Cuál de estos no es un atributo básico?
 - a) id
 - b) class
 - c) style
 - d) align
- 7) En relación a HTML cuál de estas es verdadera:
 - a) Todas las etiquetas deben ir en minúsculas

- b) Todas las etiquetas tienen apertura y cierre
 - c) HTML5 es la última versión de HTML
 - d) Ninguna es cierta
- 8) ¿Cuál es la etiqueta para indicar un salto de línea?
- a) `</br>`
 - b) `<breakLine>`
 - c) `
`
 - d) `<hr>`
- 9) ¿Cuál es la etiqueta para poner texto en negrita?
- a) ``
 - b) `<bold></bold>`
 - c) `<i></i>`
 - d) `<u></u>`
- 10) ¿Qué etiqueta se usa para escribir una lista no ordenada?
- a) ``
 - b) ``
 - c) ``
 - d) `<list>`
- 11) ¿Qué etiqueta se usa para denominar cada fila en una tabla html?
- a) `<tr>`
 - b) `<td>`
 - c) `<th>`
 - d) ``
- 12) ¿Qué atributo debe ir siempre en la etiqueta `<a>`?
- a) `src`
 - b) `align`
 - c) `href`
 - d) `alt`

EJERCICIOS DE APRENDIZAJE

1. Vamos a crear una página web de nuestra banda favorita, vamos a mostrar un ejemplo con los Beatles:

Los Beatles

Es una banda de rock formada en el año 1960 en Liverpool.



Ingrantes

- Paul McCartney
- John Lennon
- Ringo Star
- George Harrison

Año	Disco
1965	Help!
1968	The Beatles
1969	Abbey Road
"Abbey Road fue su ultimo disco".	

La tabla tendrá bordes que se lo debemos agregar sin css. Investigar atributo border.

2. Por ultimo, vamos a crear un formulario para registrar un usuario que se vea de la siguiente manera:

Registrar un usuario

Nombre del usuario

Contraseña del usuario

Edad del usuario

Fecha de nacimiento del usuario

Sexo del usuario

- ☐ Hombre
☐ Mujer
☐ Prefiero no decir

País nacimiento del usuario

Es importante que en las casillas de sexo del usuario se puede clicar la/s palabra/s hombre, mujer o prefiero no decir para seleccionar la opción. Recordemos que eso lo podemos hacer con la etiqueta label.