

PROGRAMACION WEB

GUIA DE LABORATORIO Nº 3

Nombre de la practica: Uso de tablas y formularios HTML5 y CSS3

I. Objetivos

Que el estudiante sea capaz de:

1. Utilizar los elementos HTML relacionados con tablas de forma apropiada.
2. Mostrar capacidad para crear tablas con agrupamientos de filas y de columnas en una página Web.
3. Entienda el funcionamiento y la utilidad de los formularios como medio para obtener información de parte del usuario
4. Comprenda los dos métodos de envío de formularios
5. Conozca las nuevas opciones de HTML 5 para los formularios

II. Introducción Teórica

TABLAS EN HTML

Una tabla es una manera muy clara y compacta de mostrar la información. Una tabla en HTML se presenta como un conjunto de filas (formadas horizontalmente) y columnas (formadas verticalmente). Las filas están apiladas una sobre otra. Cada fila contiene a su vez un conjunto de celdas, colocadas una al lado de la otra.

Estructura de una tabla.

Las tablas están formadas por celdas organizadas en filas y columnas. Una serie de **celdas organizadas horizontalmente** constituyen una **fila** y un **conjunto de celdas organizadas de forma vertical** forman una columna.

Se puede decir que una **celda es el espacio formado por la intersección de una fila y una columna**.

	Columna 1	Columna 2	Columna 3
Fila 1	Celda (1,1)	Celda (1,2)	Celda (1,3)
Fila 2	Celda (2,1)	Celda (2,2)	Celda (2,3)
Fila 3	Celda (3,1)	Celda (3,2)	Celda (3,3)

El esquema de etiquetas necesario para insertar una tabla puede pensarse como se muestra a continuación:

<TABLE>			
<TR>	<TH>Celda (1,1) </TH>	<TH>Celda (1,2) </TH>	<TH>Celda (1,3) </TH>
<TR>	<TH>Celda (2,1) </TH>	<TH>Celda (2,2) </TH>	<TH>Celda (2,3) </TH>
<TR>	<TH>Celda (3,1) </TH>	<TH>Celda (3,2) </TH>	<TH>Celda (3,3) </TH>
</TABLE>			

El esquema general anterior es sólo una ayuda visual para comprender el código HTML que será necesario introducir en el documento web para producir una tabla. El código sería el siguiente:

```
<TABLE>
  <CAPTION>Estructura de una tabla
HTML</CAPTION> <TR>
  <TD> celda (1,1) </TD>
  <TD> celda (1,2) </TD>
  <TD> celda (1,3) </TD>
</TR>
<TR>
  <TD> celda (2,1) </TD>
  <TD> celda (2,2) </TD>
  <TD> celda (2,3) </TD>
</TR>
<TR>
  <TD> celda (3,1) </TD>
  <TD> celda (3,2) </TD>
  <TD> celda (3,3) </TD>
</TR>
</TABLE>
```

La visualización de una tabla se genera automáticamente a partir de las filas y las columnas definidas. Sin embargo para un navegador no es fácil interpretar a tiempo como debe aparecer la tabla. Primero tiene que leer la tabla completa antes de poder visualizar alguna cosa. En caso de tablas muy grandes eso puede conducir a que en la pantalla aparecen espacios vacíos desagradables mientras se carga la página.

HTML ofrece una nueva sintaxis para comunicarle al navegador al principio de la tabla cuántas columnas tiene.

De esta manera el navegador puede cargar la tabla mucho más rápido, o sea que una parte de la tabla puede ser visualizada antes de que la totalidad de la tabla haya sido leída. Esto se debe implementar mediante las etiquetas <COLGROUP>...</COLGROUP> y sus hijas <COLS>. No obstante, estas etiquetas, que veremos más adelante, sólo son compatibles con Internet Explorer, cosa que las hace desaconsejables para nuestras páginas, pues debemos siempre crear páginas compatibles con todos los navegadores de cuarta generación y todas las resoluciones habituales.

Etiquetas básicas de una tabla

El siguiente cuadro presenta una explicación del significado de cada una de las etiquetas utilizadas en la sección anterior:

Etiqueta	Función
<code><TABLE>...</TABLE></code>	Son las etiquetas principales de definición de una tabla. Con ellas se indica el inicio y el final de una tabla.
<code><CAPTION>...</CAPTION></code>	Esta etiqueta cerrada tiene el propósito de colocar un título descriptivo para la tabla.
<code><TR>...</TR></code>	Este par de etiquetas delimitan una fila. La etiqueta de cierre <code></TR></code> es opcional, sin embargo, se recomienda su uso.
<code><TH>...</TH></code>	Se utilizan estas etiquetas para crear una celda de tipo encabezado (con letra negrita y centrada en la celda). Esta etiqueta puede utilizarse como encabezados de las columnas. La etiqueta de cierre <code></TH></code> es opcional.
<code><TD>...</TD></code>	Permite crear una celda normal dentro de una fila. La etiqueta de cierre <code></TD></code> es opcional.

Atributos principales de las etiquetas básicas

Atributos de `<TABLE>`:

width = "n": donde n puede ser expresado en píxeles o en tanto por ciento (%). Este atributo determina el ancho que va a tener la tabla. Si le damos un valor en píxeles el ancho será absoluto; es decir, el ancho de la tabla será el mismo siempre independientemente del tamaño de la pantalla o del elemento que contenga la tabla, mientras que si le damos un valor en porcentaje (%) el ancho será relativo al tamaño de pantalla o del elemento que contenga la tabla. Así, si queremos que la tabla ocupe toda la ventana del navegador, bastará con especificar el atributo `width="100%"`.

border = "n": donde n es un número entero. Este atributo permite que veamos los bordes de la tabla y de las celdas que la componen, estos bordes van a tener un ancho de n píxeles, Si no se usa este atributo no veremos los bordes, ya que su valor por defecto es `border = "0"`. Las celdas sin contenido no aparecerán con bordes, por lo que si queremos que se muestren deberemos insertar en la celda un gif transparente de 1x1 píxeles, un espacio (` `) o una etiqueta `
`.

bgcolor = "color": donde color va a venir expresado del mismo que se hizo en `bordercolor`; es decir, mediante su nombre web en inglés o mediante su código hexadecimal. Con este atributo podemos definir el color de fondo que va a tener la tabla.

background = "ruta imagen": permite establecer una imagen de fondo para toda la tabla, y en donde "ruta imagen" va a ser la ruta de directorios o una URL de Internet en la que se encuentra la imagen . Así, si tenemos nuestra página colgando del directorio raíz de la aplicación web y dentro de este hay una carpeta "images" que contiene a nuestra imagen de fondo "fondo1.gif", la sintaxis correcta sería:

```
<TABLE width="200" height="750" border="1 "
background="images/fondo1.gif">
  <TR>
    <TD>UNO</TD>
    <TD>DOS</TD>
  </TR>
</TABLE>
```

cellspacing = "n": donde n es un número entero. Nos permite establecer el espacio que va a haber entre las celdas, con lo que podemos dar una mayor o menor separación a las mismas, determinado por n= "nº de píxeles entre ellas". Si la tabla está definida con bordes, este atributo modificara el grosor del borde interior de la misma.

cellpadding = "n": con n= "nº de píxeles". Este valor nos va a determinar el espacio interior en las celdas; es decir, el espacio que habrá entre los bordes de la celda y el texto, imagen o componente que hay dentro de esta.

summary = "descripcion": Permite introducir un resumen, o un comentario para especificar el objeto y / o contenido de la tabla. No se muestra en el navegador.

OBSERVACIÓN: Los atributos que tienen que ver con el formato y la presentación de la tabla están desaprobadados por el estándar HTML 4.0 de la W3C. Entre ellos están: align, background y bgcolor.

Atributos de <TR>:

Esta pareja de etiquetas es necesaria para especificar y delimitar cada una de las filas que va a tener la tabla

valign = "top/middle/baseline/bottom": Nos permite alinear verticalmente el contenido de las celdas de la fila en la parte superior, en medio, a la altura del texto de la celda colindante o abajo respectivamente.

lang = "idioma": Se utiliza para especificar el idioma del contenido de una fila completa de la tabla. Los valores posibles son los códigos de idioma definidos en la RFC 1766 para documentos HTML.

dir = "dirección": Se usa para indicar la dirección del texto y de las columnas. Toma los valores: ltr (de izquierda a derecha) y rtl (derecha a izquierda). Cuando el valor es rtl, el ordenamiento de las columnas comenzara por el tope arriba a la derecha.

Atributos de <TH> y <TD>:

```
<TH align = "left|center|right|justify|decimal" valign =  
"top|middle|bottom|baseline" width = "Tamaño"  
bgcolor="color" rowspan="numFilas" colspan= "numColumnas" nowrap>  
  
<TD align = "left|center|right|justify" valign =  
"top|middle|bottom" width="Tamaño" bgcolor="color"  
rowspan="numFilas" colspan= "numColumnas" nowrap>
```

bgcolor ="color": donde color puede venir en su nombre web en inglés o en su valor hexadecimal. Al igual que con la etiquetas <TABLE> y <TR>, este atributo va a establecer el color del fondo de cada celda, prevaleciendo este color sobre el que hayamos definido para la tabla en conjunto y para la fila en general.

background = "ruta de imagen": donde ruta de imagen puede ser la ruta relativa en el directorio de archivos del servidor o una URL de Internet donde se encuentra la imagen. Este atributo es del todo

análogo al de las etiquetas <TABLE> y <TR>, tiene sus mismas limitaciones respecto a especificar una ruta URL y nos permite definir una imagen de fondo para toda una fila.

Hay que tener mucho cuidado, ya que la imagen utilizada debe ser del mismo tamaño que la fila de la tabla, ya que si no el efecto no será el correcto.

align = "left/center/right/justify/char": Este atributo va a definir si el texto o la imagen que va a ir en la celda va a estar alineados a la izquierda, en el centro o a la derecha de cada celda. Su valor por defecto es LEFT, por lo que si no especificamos nada el contenido de la celda aparecerá alineado a la izquierda.

valign = " top / middle / baseline / bottom": Nos permite alinear verticalmente el contenido de la celda en la parte superior, en medio, que todas las celdas sean alineadas a una línea base común (o sea, de tal manera que el primer renglón de cada celda comienza a una misma altura) o abajo respectivamente. Su valor por defecto es "middle", situando el contenido de la celda verticalmente en medio de la misma

colspan = "n": donde n = nº entero. Este atributo nos permite combinar varias columnas en una sola, de tal forma que esta ocupara todo el espacio reservado para las que contiene.

rowspan = "n": donde n = nº entero. Este atributo nos permite combinar varias celdas en una sola, de tal forma que esta ocupara todo el espacio reservado para las que contiene.

Estos dos últimos atributos son de los mas útiles de los que disponemos para crear tablas, pero su uso entraña ciertas dificultades, que sólo se evitan con un buen conocimiento y practica.

width = "n": donde n=nº de píxeles ó tanto por ciento. Con este atributo se especifica el ancho que va a tener cada celda. Si queremos que las celdas sean del mismo tamaño es necesario que especifiquemos el ancho en tanto por ciento, ya que si no es así, dependiendo del espacio que necesite el contenido de la celda, esta se ampliara por sí sola.

height = "n": donde n=nº de píxeles ó tanto por ciento. Con este atributo se especifica el alto que va a tener cada celda. Normalmente el alto de la celda se suele dejar que lo coja ella misma, adaptándose a las necesidades del texto o imagen que contiene. La especificación "height", al contrario de la especificación "width", no es estándar HTML, sin embargo es interpretada por los navegadores que interpretan tablas.

nowrap: Este atributo evita que el texto de un párrafo contenido en la celda dé un salto de línea automático cuando se llega al límite derecho de la celda. Como conclusión, si se utiliza este atributo, cuando se llega al final de ancho especificado por width, la celda seguirá expandiéndose más, si es necesario, para dar cabida al contenido que mostrará (texto o imágenes). Si no se utiliza nowrap, al llegar el texto al límite derecho de la celda se producirá un salto de línea. Como observación, no siempre implementan muy bien este atributo los navegadores, comportándose a veces la tabla de una forma inesperada dependiendo del navegador utilizado.

OBSERVACIONES:

1. Los atributos colspan y rowspan sólo se utilizan con los elementos TH y TD, nunca con TR.
2. Atributos que tienen que ver con formato y presentación están desaprobados. Entre ellos: bgcolor, background, align, valign, width, height y nowrap.

Otras etiquetas utilizadas en la definición de una Tabla HTML

Etiqueta	Función
<code><CAPTION>...</CAPTION></code>	Se utiliza para colocar una descripción de la naturaleza de la tabla. Podría ser el título de la tabla. Sólo se permite una etiqueta <code><CAPTION></code> por tabla. Además, sólo puede colocarse inmediatamente después de la etiqueta inicial <code><TABLE></code> .
<code><THEAD>...</THEAD></code>	Sirve para agrupar un conjunto de filas, como cabecera de la tabla.
<code><TFOOT>...</TFOOT></code>	Agrupar un conjunto de filas como pie de la tabla.
<code><TBODY>...</TBODY></code>	Agrupar un conjunto de celdas como el cuerpo de la tabla. Pueden definirse una o más secciones de cuerpo de tabla.

NOTA: La cabecera de tabla y el pie de tabla deberían contener información sobre lo descrito en las columnas de la tabla. El cuerpo de tabla debería contener filas de datos de tabla.

Etiqueta `<CAPTION align="top|bottom">`

Define un título para la tabla completa, que irá en una zona fuera de cualquier celda o fila de la tabla. Cada definición de una tabla HTML solo puede tener una etiqueta `<CAPTION>`. Ejemplo:

```
<TABLE>
<CAPTION> Titulo de mi tabla </CAPTION>
...
</TABLE>
```

Con el parámetro `ALIGN`, podemos controlar que el título aparezca arriba (por defecto `TOP`) o abajo de la tabla, con el valor `BOTTOM`.

Etiquetas `<THEAD>` `<TFOOT>` y `<TBODY>`

Con estas 3 etiquetas, las filas de una tabla pueden agruparse en una cabecera de tabla, un pie de tabla, y una o más secciones como el cuerpo de tabla, respectivamente.

El encabezado y el pie de página se usan cuando se elaboran / imprimen tablas largas, que contienen varias filas de datos y permiten que el texto definido en la cabecera y pie de una tabla pueda repetirse en todas las páginas necesarias para mostrar los datos de la tabla.

Cuando se usan las etiquetas `THEAD`, `TFOOT` y `TBODY`, cada una de ellas debe contener un grupo de filas. Cada

grupo de filas debe contener al menos una fila, definida por la etiqueta `<TR>`.

A continuación se ilustra un ejemplo del orden y la estructura del encabezado y pie de la tabla, más cada uno de los cuerpos de esa tabla.

```
<TABLE>
<THEAD>
<TR> ... Información de Cabecera de la tabla...
</TR> </THEAD>
```

```
<TFOOT>
<TR> ... Información del Pie de la tabla... </TR>
</TFOOT>
<TBODY>
<TR> ... primera fila de los datos del bloque UNO... </TR>
<TR> ... primera fila de los datos del bloque UNO... </TR>
</TBODY>
<TBODY>
<TR> ... primera fila de los datos del bloque DOS...
<TR> ... primera fila de los datos del bloque DOS...
</TBODY>
</TABLE>
```

Analizando el orden de etiquetas anterior dentro de <TABLE>, primero se debe definir el <THEAD>, luego el <TFOOT> y a continuación se definen cada uno de los <TBODY>. Esto se hace así para que el Navegador pueda mostrar el Encabezado y el Pie de la tabla antes de recibir todas las (potencialmente numerosas) filas de datos del cuerpo de la tabla

Si desea agregar o no un encabezado <THEAD> o un pie <TFOOT> a la definición de su tabla, debe tener en cuenta estas 2 reglas:

1. Si quiere agregar el encabezado/pie, deberá declarar primero el <THEAD> y luego el <TFOOT>. Después define con la pareja de etiquetas <TBODY> el contenido de las celdas del cuerpo de su tabla.
2. Si no desea declarar un encabezado o un pie para su tabla, solo declare la definición de celdas (con etiquetas <TR> y (<TH> o <TD>)) sin usar <TBODY>

Estas reglas obedecen a razones de compatibilidad con versiones de Navegadores antiguas. Además tenga en cuenta que las secciones THEAD, TFOOT y TBODY deben **contener el mismo número de columnas**.

Formularios

Un formulario es una sección del documento web en la que puede incluirse texto normal y otros elementos especiales, denominados controles. Los usuarios interactuarán con el formulario modificando los controles. Por ejemplo, introduciendo texto, activando una casilla de verificación, seleccionando un botón de opción, seleccionando un elemento de un menú desplegable, etc.

Los datos que son ingresados al formulario son enviados normalmente a un programa en el servidor web donde se hospeda la página web para su procesamiento, o a nosotros directamente, mediante correo electrónico.

Los programas que procesan los datos del formulario se alojan en el servidor web, llamados CGI (Common Gateway Interface), suelen estar escritos en Perl o en C++. También es posible procesar esa información mediante páginas PHP, ASP (Active Server Pages) o JSP (Java Server pages), que son páginas que contienen dentro de su código HTML fragmentos de código escrito en php, Visual Basic Script o Java respectivamente. Mediante alguno de estos métodos podremos incorporar los datos recibidos del formulario a una base de datos, crear un registro, enviar un mensaje automático de respuesta al usuario, validar un pedido, confirmar el acceso a un sistema remoto, etc. Una vez procesada de esta forma la información recibida podremos enviar una página de respuesta definida a

nuestro visitante, con lo que podremos personalizar esta respuesta de antemano.

Tipos de controles

1. Cuadros o cajas de texto y contraseñas

Son controles que permiten la introducción de texto en una sola línea. Puede configurarse el tamaño del cuadro de texto, el número de carácter es máximo que se pueden introducir en él, etc.

2. Botones

Los desarrolladores Web pueden crear tres tipos de botones:

- **Botones de envío (submit buttons):** Cuando se activa, un botón de envío se envía un formulario. Un formulario puede contener más de un botón de envío.
- **Botones de reinicialización (reset buttons):** Cuando se activa, un botón de reinicialización reinicializa todos los controles a sus valores iniciales.
- **Botones pulsadores (push buttons):** Los botones pulsadores no tienen un comportamiento por defecto. Cada botón pulsador puede tener asociados scripts en el lado del cliente a través del atributo event del elemento. Cuando ocurre un evento (p.ej., el usuario aprieta el botón, lo suelta, etc.), se acciona el script asociado.

Se pueden crear botones con el elemento `BUTTON` o el elemento `INPUT`.

Los botones creados con `<button>` funcionan igual que los creados con la etiqueta `<input type="button">`. La única diferencia es que el contenido de un botón creado con `<button>` es lo que pondremos entre las etiquetas de apertura y cierre, pudiendo definir estilos o incluir imágenes, mientras que al crearlo con `<input>` sólo podremos incluir un texto mediante el atributo `value`.

3. Casillas de verificación (checkboxes)

Las casillas de verificación (y los botones de opción) actúan como interruptores de encendido/apagado que pueden ser activados por el usuario. Una casilla de verificación está "marcada" cuando se establece el atributo `checked` del elemento de control. Cuando se envía un formulario, solamente pueden tener éxito los controles de casillas de verificación que estén marcadas. Varias casillas de verificación de un formulario pueden compartir el mismo nombre de control.

Así, por ejemplo, las casillas de verificación permiten a los usuarios elegir varios valores para la misma propiedad. Para crear un control de casilla de verificación se utiliza el elemento `INPUT`.

4. Botones de opción (radio buttons)

Los botones de opción son como las casillas de verificación, excepto en que cuando varios comparten el mismo nombre de control, son mutuamente exclusivos: cuando uno está "encendido", todos los demás con el mismo nombre se "apagan". Para crear un control de botón de opción se usa el elemento `INPUT`.

Si ninguno de los botones de opción de un conjunto que comparten el mismo nombre tiene un valor inicial de "encendido", el comportamiento del agente de usuario al elegir cuál es el control que está inicialmente "encendido" queda sin definir.

Al diferir los comportamientos de los agentes de usuario, los autores deberían asegurarse de que en cada conjunto de botones de opción hay uno que inicialmente está "encendido".

5. Menús (menus) y cuadros de lista

Los menús ofrecen al usuario opciones entre las cuales elegir. El elemento SELECT crea un menú, en combinación con los elementos OPTGROUP y OPTION.

<pre><form action="rutaprograma" method="metodo"> ...elementos del formulario... ...botones para enviar los datos... </form></pre>	
action = "rutaprograma".	Indica el programa del servidor que va a "tratar" las variables que se envíen con el formulario o el envío de los datos mediante correo electrónico mediante el valor "mailto:direccion_de_correo", en cuyo caso deberemos añadir el parámetro ENCTYPE="text/plain" para que lo que recibamos resulte legible. También puede ser una URL usando el método GET (por ejemplo, un botón para ir a otra página). En el caso de que definamos aquí un programa para el tratamiento de datos deberemos especificar su ruta relativa respecto al directorio de carpetas del servidor en dónde tengamos situado nuestra aplicación web o bien una URL completa, si el programa está en otra dirección de Internet.
method="metodo"	Indica el método que se utilizará para enviar los datos del formulario. Los valores posibles son GET (valor por defecto) o POST. En el caso de que queramos enviar los datos de forma oculta a un programa en el servidor o a una dirección de correo utilizaremos method="POST". Si usamos method="GET", los datos serán enviados a través de la URL de la página.
enctype = " tipo "	Indica el tipo de documento en formato MIME. El más usado para que lleguen el contenido a través de un mail con el método POST sería "text/plain". Especifica el tipo de encriptación que se va a realizar con los datos que se van a enviar. Este atributo solamente se aplica si method es POST. Indica cómo se codificará el formulario si utiliza el método POST. Por defecto es "application/x-www-form-urlencoded". Otro posible valor es "multipart/form-data".
name="nombre"	Indica el nombre del formulario para futuras referencias con rutinas de scripts.

En resumen, el elemento FORM actúa en primer lugar como contenedor de los elementos del formulario y, en segundo lugar, indica la acción a desarrollar con los mismos, a través del uso de parámetros o atributos.

Etiqueta <INPUT> ... </INPUT>

La etiqueta <input> es la que controla la mayoría de los posibles controles de un formulario en HTML. El tipo de control lo indicaremos mediante el atributo type. Sus atributos y valores son:

* TYPE = "tipo", donde tipo puede ser uno cualquiera de los elementos que veremos a continuación:

- **text**, que sirve para introducir un cuadro de texto simple, y admite los siguientes parámetros: ○ name="nombre", que asigna de forma unívoca un nombre identificador para la caja de texto. Este identificador se utilizará para hacer referencia al control dentro del código de programa o script que procesará los datos del formulario.
○ maxlength="n", que fija el número máximo de caracteres que se pueden introducir en el cuadro de texto.

- `size="n"`, que establece el tamaño de la caja de texto en pantalla, medida en número de caracteres.
- `value="texto"`, que establece el valor por defecto del texto que aparecerá en inicialmente en la caja de texto.
- `disabled`, que desactiva la caja de texto, por lo que el usuario no podrá escribir nada en ella.
- `accept="lista_de_content-type"`, Indica el tipo de contenido que aceptara el servidor.

Sus posibles valores son:

`_text/html`

`_application/msexcel`

`_application/msword`

`_application/pdf`

`_image/jpg`

`_image/gif`

`_audio/basic`

`_video/mpeg` etc.

`readonly`, que establece que el texto no puede ser modificado por el usuario.

`tabindex="n"`, que especifica el orden de tabulador que tendrá el campo respecto todos los elementos que incluye el formulario.

`alt="texto"`, que asigna una pequeña descripción al cuadro de texto.

- **radio**, que define un conjunto de elementos de formulario de tipo circular, en los que el usuario debe optar por uno solo de ellos, que se marca con el ratón o tabulador. Admite los parámetros:
 - `name="nombre"`, que asigna un nombre identificador único a la variable definida por el elemento. Este identificador debe ser el mismo para todos los elementos radio de un grupo.
 - `value="valor"`, que define un valor posible de la variable para cada uno de los radio botones.
 - `checked`, que marca por defecto uno de los radio botones del grupo.
 - `disabled`, que desactiva el radio botón, por lo que el usuario no podrá marcarlo.
 - `tabindex="n"`, que especifica el orden de tabulador que tendrá el campo respecto todos los elementos que incluye el formulario.
- **checkbox**, que define una o más casillas de verificación, pudiendo marcar el usuario las que desee del conjunto total. Si pinchamos una casilla con el ratón o la activamos con el tabulador y le damos a la barra espaciadora la casilla se marca; si volvemos a hacerlo, la casilla se desmarca.

Sus parámetros complementarios son:

- `name="nombre"`, que asigna un nombre identificador único a la variable definida por el elemento. Este identificador debe ser el mismo para todos los elementos conjunto de casillas.
- `value="valor"`, que define un valor posible de la variable para cada uno de casillas de verificación.
- `checked`, que marca por defecto una o más de las casillas del grupo.
- `disabled`, que desactiva la casilla de verificación, por lo que el usuario no podrá marcarla.
- `tabindex="n"`, que especifica el orden de tabulador que tendrá el campo respecto todos los elementos que incluye el formulario.
- `button`, que define un botón estándar. Este botón puede ser usado para diferentes acciones, pero normalmente se le da utilidad mediante JavaScript, mediante el evento "OnClick". Sus parámetros son:
 - `name="nombre"`, que asigna un nombre al botón, que nos puede servir para acciones con lenguaje de script.
 - `value=" valor "`, que define el texto que va a figurar en el botón.
 - `disabled`, que desactiva el botón, de tal forma que no se produce ninguna acción cuando se pulsa, debido a que permanece inactivo.
 - `tabindex = "n"`, que especifica el orden de tabulador que tendrá el campo respecto todos los elementos que incluye el formulario.
- **image**, que introduce un botón definido por una imagen, en vez del formato estándar de botones, con lo que podemos así personalizar el botón. Inicialmente su función era contener una localización de las coordenadas que pinchara el usuario, para que luego el programa CGI realizara una acción u otra dependiendo de estas. Pero actualmente se usa más para personalizar un botón de envío de datos; es decir, su funcionalidad es análoga a la de submit, pero nos permite personalizar este elemento. Además añadirá como información en el formulario las coordenadas x e y donde el usuario lo pulsó. Admite los parámetros:
 - `name="nombre"`, que asigna un nombre al botón para identificarlo de forma única y poder así acceder luego a sus propiedades.
 - `src="ruta imagen "`, que establece la ruta dónde localizar el fichero de imagen. Esta ruta puede ser relativa al directorio Web en el servidor o una URL absoluta que define unívocamente la localización de la imagen.
 - `width="w"`, que fija la anchura del botón de imagen.
 - `height="h"`, que fija la altura del botón de imagen.
 - `align= "left / middle / right / bottom / baseline"`, que define la posición del texto que rodea el botón respecto a este.
 - `hspace= "x"`, que fija el espacio horizontal que habrá entre el botón y el texto que lo circunda.
 - `vspace= "x"`, que fija el espacio vertical que habrá entre el botón y el texto que lo circunda.
 - `alt = " texto "`, que asocia un texto explicativo al botón de imagen. Al situar el cursor encima del botón se mostrara este texto en forma de tool tip de Windows. Siempre es conveniente poner este atributo, para que en caso de navegadores de sólo texto, de que la imagen no se encuentre en la ruta por cualquier motivo o de que la página tarde mucho en cargarse por su peso, el usuario pueda ver qué hace el botón de imagen y pueda en consecuencia utilizarlo para tal fin.

- disabled, que desactiva el botón, de tal forma que no se produce ninguna acción cuando se pulsa, pues permanece inactivo.
 - tabindex = "n", que especifica el orden de tabulador que tendrá el campo respecto todos los elementos que incluye el formulario.
 - usemap= "#nombre de mapa", que nos permite el uso de un mapa de imágenes en el cliente como un dispositivo más de entrada de datos.
- **hidden**, que define un campo invisible, por lo que no se ve en pantalla. Aunque parece que así definido no tiene utilidad, sus usos son varios e importantes, y los veremos más tarde. Sus atributos son:
 - name="nombre", que asigna un nombre identificador único al campo oculto.
 - value="valor", que va a ser el valor fijo que se le va a pasar al programa del servidor, ya que el usuario no puede modificarlo. En realidad este valor no tiene porqué ser fijo, ya que lo vamos a poder modificar mediante código de script, lo que nos va a permitir ir pasando una serie de variables ocultas de una página a otra.
- **file**, que define un archivo que puede ser enviado junto con los datos del formulario. En este tipo de elemento encontramos asociados una caja de texto y un botón en el que encontramos escrito bien "examinar..." bien "browse..." , dependiendo del lenguaje de nuestro navegador, que al ser pulsado nos abre la típica ventana de exploración de nuestras unidades de disco, para que seleccionemos el archivo que queremos enviar al servidor. Cuando elegimos uno su ruta aparece en la caja de texto. Sus principales atributos son:
 - title="titulo": que muestra un texto en tipo tip al situar el cursor encima del botón de examinar (sólo en Internet Explorer).
 - accept="tipo_archivo": que determina el tipo de archivo que se admite como para enviar.
 - disabled : que desactiva tanto el botón como la caja de texto, impidiendo al usuario seleccionar un archivo.
 - size="numero_entero": que fija la anchura de la caja de texto asociada.
- **submit**, que incorpora al formulario un botón de envío de datos. Cuando el usuario pulsa este botón los datos que ha introducido en los diferentes campos del formulario son enviados al programa del servidor o a la dirección de correo indicada en action. Sus atributos son:
 - value="valor", que define el texto que va a aparecer en el botón de envío. Disabled que desactiva el botón, de tal forma que no se produce ninguna acción cuando se pulsa, debido a que permanece inactivo.
 - tabindex="n", que especifica el orden de tabulador que tendrá el campo respecto todos los elementos que incluye el formulario.
- **reset**, que define un botón que al ser pulsado por el usuario borra todos los datos que hubiera introducido en cualquiera de los campos del formulario. Sus atributos son los mismos que los de SUBMIT.

Etiqueta <SELECT> ... </SELECT>

Este elemento se utiliza para presentar al usuario varias opciones y que elija entre ellas una o más. La forma general es:

```
<SELECT name="nombre" size="num" multiple>
<OPTION value="valor_asociado_A" selected>Opción
A <OPTION value="valor_asociado_B" >Opción B
<OPTION value="valor_asociado_C" >Opción C
</SELECT>
```

Los atributos del elemento SELECT permiten especificar el nombre lógico al que se asociará la opción elegida de entre las posibles. Sólo serán visibles las que se especifiquen en size. Si es 1, el elemento SELECT tendrá la apariencia de un menú desplegable (cuidado con el número de opciones, pues podría desbordar la pantalla). Normalmente, el elemento selección permite seleccionar sólo una de sus opciones, pero con multiple podremos hacer que se pueda seleccionar a varios de los elementos.

Las opciones se especifican con la etiqueta OPTIO que puede aparecer con un atributo value que indica el valor que se asociará al elemento de selección en caso de que esa opción sea la elegida. Este atributo puede obviarse, en cuyo caso el valor devuelto será la descripción de la opción misma. Se puede especificar con selected la opción seleccionada por defecto.

Los atributos del elemento SELECT y sus posibles valores son:

- name = "nombre", que asigna un nombre identificador al campo, de tal forma que al enviar los datos del formulario la opción elegida será asociada a este nombre.
- size = "n", con n= nº entero, que define el número de opciones visibles. Si n=1 el campo de selección se presenta como una lista desplegable, mientras que si se indica otro valor se presenta como una caja de lista con barra de desplazamiento.
- multiple, que permite elegir varias de las opciones a la vez. Si no se especifica este atributo solamente se podrá escoger una de las opciones. Para ello hay que mantener pulsada la tecla CONTROL mientras se seleccionan y/o deseleccionan las diversas opciones.
- disabled, que desactiva la lista, de tal forma que no se produce ninguna acción cuando se pulsa una opción, pues permanece inactiva.
- tabindex = "n", que especifica el orden de tabulador que tendrá el campo respecto todos los elementos que incluye el formulario.

Para introducir cada una de las diferentes opciones de la lista se utiliza la etiqueta <OPTION>, que admite como parámetros:

- value, que fija el valor que será asociado al parámetro name de <SELECT> cuando se envíe el formulario. Este valor debe ser único para cada opción.
- selected, que establece la opción por defecto. Si no se especifica este parámetro en ninguna opción se tomara la primera de ellas por defecto.

- label, permite especificar un rótulo para la opción, más corto que el contenido del elemento OPTIO . Cuando se utiliza este atributo el navegador debería utilizar como rótulo de la opción el valor asignado a este atributo en lugar del contenido del elemento OPTIO .

Etiqueta <TEXTAREA>... </TEXTAREA>

La introducción de un texto, con un número de filas y columnas determinado se puede conseguir con la utilización de la etiqueta TEXTAREA cuya forma general es la siguiente:

```
<TEXTAREA name="nombre" rows="filas" cols="columnas">
Texto por defecto
</TEXTAREA>
```

Con ello, se presenta una caja donde el usuario puede introducir varias líneas de texto. La caja de texto tendrá barras de desplazamiento y unas dimensiones de ROWS x COLS. El texto por defecto (optativo) aparecerá entre las etiquetas de apertura y cierre del área de texto.

Atributos y valores de <TEXTAREA>:

name="nombre", que asigna un nombre identificador al campo, que será asociado en el envío del formulario al texto introducido en la caja de texto.

cols="x", que define el número de columnas visibles de la caja de texto.

rows="y", que define el número de filas visibles de la caja de texto.

wrap= "valor", que justifica automáticamente el texto en el interior de la caja. Este atributo es de uso complicado.

Según las recomendaciones si su valor es VIRTUAL se enviará todo el texto seguido, en una línea, mientras que si vale PHYSICAL el texto se enviará separado en líneas físicas, pero, así como se define no lo admiten los navegadores. Usemos cual usemos de los dos el resultado es el mismo: justificación por líneas físicas, y así y todo se producen saltos de línea indeseados.

Si construimos la página web con un programa de ayuda tipo HomeSite nos encontramos con tres posibles valores de wrap: SOFT y HARD, que hacen lo mismo, alineando el contenido en líneas físicas, como en los casos anteriores y es el valor por defecto, y OFF, que coloca todo el texto de la caja en una sola línea, apareciendo entonces una barra de desplazamiento horizontal para poder visualizar todo el contenido de la caja.

disabled, que desactiva la caja de texto. Su utilidad es escasa.

readonly, que impide que el contenido de la caja sea modificado por el usuario.

tabindex = "n", que especifica el orden de tabulador que tendrá el campo respecto todos los elementos que incluye el formulario.

Etiqueta <BUTTON >... </BUTTON >

A partir de la implementación de los estándares HTML 4.0 contamos con varias etiquetas nuevas para construir formularios, siendo BUTTON una de ellas, bastante útil, por cierto.

Esta etiqueta proporciona un método único para la implementación de cualquier tipo de botón de formulario. Sus principales atributos son:

type= " tipo ", que puede tomar los ya conocidos valores submit (por defecto), reset y button.

name= " nombre ", que asigna un nombre identificador único al botón.

value= " texto ", que define el texto que va a aparecer en el botón.

La principal ventaja que aportan estas etiquetas, con respecto a los botones generados con <INPUT> es que con <BUTTON> vamos a poder introducir dentro de ellas cualquier elemento de HTML, como imágenes, tablas, etc.

III. Requerimientos

Nº	Cantidad	Descripción
1	1	PC con uno de los editores de páginas Web: Sublime Text, Brackets, Notepad ++, BlueFish y HTML Kit y cualquiera de los navegadores siguiente: Internet Explorer 7 u 8, Firefox (última versión disponible), Google Chrome (última versión disponible), Opera (última versión disponible), Safari (última versión disponible).
2	1	Guía 4

IV. Procedimiento

1. Abra el Editor y digite los siguientes códigos HTML.
2. Guarde los archivos HTML con los nombres sugeridos.

Ejercicio#1: planetas.html

```
1  <!DOCTYPE HTML>
2  <HTML>
3  <HEAD>
4      <TITLE>El sistema solar</TITLE>
5      <STYLE type="text/css">
6          BODY {background-color:#666699;}
7          H1 {text-align:center; font-family:"Bookman Old Style";
8              font-weight:bolder; color:#FFFF33;}
9          HR {background-color:gold;}
10         P {text-align:justify; font-family:Century; color:white;}
11         TABLE {background-color:#aa6633; color:#EEEE66; font-family:Garamond;}
12         TH {text-align:center; font-weight:bolder; font-size:14pt;}
13         TD#number {text-align:center;}
14         SPAN{font-size:13pt;font-weight:bolder; font-style:italic;}
15     </STYLE>
16 </HEAD>
```

```

17 <BODY>
18 <H1>Sistema Solar</H1>
19 <HR size="8"><BR>
20 <TABLE align="center" border="2">
21   <TR>
22     <TH colspan="8">Planetas del Sistema Solar</TH>
23   </TR>
24   <TR>
25     <TH colspan="4">Terrestres o Telúricos</TH>
26     <TH colspan="4">Gigantes de gas o Jovianos</TH>
27   </TR>
28   <TR>
29     <TH>Planeta</TH>
30     <TH>Distanacia del Sol (kms)</TH>
31     <TH>Distancia de la tiera (kms)</TH>
32     <TH>Satélites</TH>
33     <TH>Planeta</TH>
34     <TH>Distanacia del Sol (kms)</TH>
35     <TH>Distancia de la tiera (kms)</TH>
36     <TH>Satélites</TH>
37   </TR>
38   <TR>
39     <TD>Mercurio</TD>
40     <TD align="center">57,900,000</TD>
41     <TD>De 91,700,000 a 218,900,000</TD>
42     <TD id="number">0</TD>
43     <TD>Júpiter</TD>
44     <TD id="number">778,400,000</TD>
45     <TD>De 628,760,000 a 970,900,000</TD>
46     <TD id="number">16</TD>
47   </TR>
48   <TR>
49     <TD>Venus</TD>
50     <TD id="number">108,200,000</TD>
51     <TD>De 41,400,000 a 257,000,000</TD>
52     <TD id="number">0</TD>
53     <TD>Saturno</TD>
54     <TD id="number">1,429,400,000</TD>
55     <TD>De 1,277,400,000 a 1,658,000,000</TD>
56     <TD id="number">18</TD>
57   </TR>
58   <TR>
59     <TD>Tierra</TD>
60     <TD id="number">149,600,000</TD>
61     <TD>-----</TD>
62     <TD id="number">1</TD>
63     <TD>Urano</TD>
64     <TD id="number">2,875,000,000</TD>
65     <TD>De 2,587,000,000 a 3,156,000,000</TD>
66     <TD id="number">21</TD>
67   </TR>
68   <TR>
69     <TD rowspan="2">Marte</TD>
70     <TD id="number" rowspan="2">227,900,000</TD>
71     <TD rowspan="2">De 55,700,000 a 399,000,000</TD>

```



```

72 <TD>Neptuno</TD>
73 <TD id="number" align="center">4,504,300,000</TD>
74 <TD>De 4,313,000,000 a 4,683,000,000</TD>
75 <TD id="number">8</TD>
76 </TR>
77 <TR>
78 <TD>Plutón</TD>
79 <TD id="number">5,900,100,000</TD>
80 <TD>De 4,290,000,000 a 7,520,000,000</TD>
81 <TD id="number">1</TD>
82 </TR>
83 </TABLE>
84 <P lang="es">
85 Los cuatro planetas más cercanos al Sol -Mercurio, Venus, la Tierra y Marte- son llamados planetas
86 <SPAN>"terrestres"</SPAN> o <SPAN>"telúricos"</SPAN>, ya que poseen superficies rocosas sólidas.
87 Los cuatro planetas gigantes que se encuentran más allá de la órbita de Marte -Júpiter, Saturno,
88 Urano y Neptuno- son llamados <SPAN>"gigantes de gas"</SPAN>, o planetas <SPAN>"jovianos"</SPAN>.
89 Diminuto y distante, Plutón, tiene una sólida superficie, pero más helada que la de los planetas
90 terrestres.
91 </P>
92 </BODY>
93 </HTML>

```

border-collapse

El estilo border-collapse determina como se mostrarán y se comportarán los bordes de elementos contiguos en las tablas. Este estilo admite dos valores: collapse y separate. El atributo collapse le indica a los bordes que deben solaparse, mostrando uno sólo. El atributo separate muestra los dos bordes contiguos, sin que éstos se solapen.

Ejercicio #2: tabla_css3.html

Copiar el siguiente código:

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>
5 Ejemplo de tabla con CSS3
6 </title>
7 <style type="text/css">
8
9 </style>
10 </head>
11 <body>
12 <table class="tabla1">
13 <thead>
14 <tr>
15 <th></th>
16 <th>Smart Starter</th>
17 <th>Smart Medium</th>
18 <th>Smart Business</th>
19 <th>Smart Deluxe</th>
20 </tr>
21 </thead>
22 <tfoot>

```

```

23     <tr>
24         <th>Precio por mes</th>
25         <td>$ 2.90</td>
26         <td>$ 5.90</td>
27         <td>$ 9.90</td>
28         <td>$ 14.90</td>
29     </tr>
30 </tfoot>
31 <tbody>
32     <tr>
33         <th>Almacenamiento</th>
34         <td>512 MB</td>
35         <td>1 GB</td>
36         <td>2 GB</td>
37         <td>4 GB</td>
38     </tr>
39     <tr>
40         <th>Ancho de banda</th>
41         <td>50 GB</td>
42         <td>100 GB</td>
43         <td>150 GB</td>
44         <td>Ilimitado</td>
45     </tr>
46     <tr>
47         <th>Base de datos MySQL</th>
48         <td>Ilimitado</td>
49         <td>Ilimitado</td>
50         <td>Ilimitado</td>
51         <td>Ilimitado</td>
52     </tr>
53     <tr>
54         <th>Actualizaci&ocute;n</th>
55         <td>$19.90</td>
56         <td>$12.90</td>
57         <td>libre</td>
58         <td>libre</td>
59     </tr>
60     <tr>
61         <th>PHP 5</th>
62         <td><span class="check"></span></td>
63         <td><span class="check"></span></td>
64         <td><span class="check"></span></td>
65         <td><span class="check"></span></td>
66     </tr>
67     <tr>

```

```

68         <th>Ruby on Rails</th>
69         <td><span class="check"></span></td>
70         <td><span class="check"></span></td>
71         <td><span class="check"></span></td>
72         <td><span class="check"></span></td>
73     </tr>
74 </tbody>
75 </table>
76 </body>
77 </html>

```

Agregar dentro de la etiqueta STYLE la siguiente regla de estilo:

```

table.tabla1{
    font-family: "Trebuchet MS", sans-serif;
    font-size: 16px;
    font-weight: bold;
    line-height: 1.4em;
    font-style: normal;
    border-collapse: separate;
    margin: auto;
}

```

Guardar los cambios y actualizar la página Web en el navegador
 Los elementos th de la tabla tendrá el siguiente estilo:

```

.tabla1 th{
    padding: 15px;
    color: #fff;
    text-shadow: 1px 1px 1px #568F23;
    border: 1px solid #93CE37;
    border-bottom: 3px solid #9ED929;
    background-color: #9DD929;
    background: -moz-linear-gradient(top, #65b6cf, #2c2c2c);
    background: -webkit-gradient(linear, 0 0, 0 100%, from(#65b6cf), to(#2c2c2c));
    -webkit-border-top-left-radius: 5px;
    -webkit-border-top-right-radius: 5px;
    -moz-border-radius: 5px 5px 0px 0px;
    border-top-left-radius: 5px;
    border-top-right-radius: 5px;
}

```

background: -webkit-gradient(linear, 0 0, 0 100%, from(#65b6cf), to(#2c2c2c));

Se interpreta de la siguiente manera:

¿Qué tipo de degradado? Lineal

Coordenadas de inicio "Eje X e Y" (0 0 = Esquina Superior Izquierda)

Coordenadas de Finalización (0 100% = Esquina Inferior Derecha)

De qué color empezar (#65b6cf)

En qué color terminar (#2c2c2c)

Guardar los cambios y actualizar la página Web en el navegador

Ahora se colocará el siguiente estilo a la celda vacía (empty) de la cabecera de la tabla

```
.tabla1 thead th:empty{
    background:transparent;
    border:none;
}
```

Guardar los cambios y actualizar la página Web en el navegador
El pie de página de la tabla tendrá el siguiente estilo:

```
.tabla1 tfoot td{
    color: #9CD009;
    font-size:32px;
    text-align:center;
    padding:10px 0px;
    text-shadow:1px 1px 1px #444;
}
```

Agregando sombra al texto:

text-shadow:1px 1px 1px #444;

La propiedad es muy sencilla y tiene cuatro parámetros:

text-shadow: posicionX posicionY amplitud color;

- **posicionX** indica el desplazamiento horizontal de la sombra con respecto al texto (positivos hacia la derecha, negativos hacia la izquierda)
- **posicionY** indica el desplazamiento de la sombra con respecto al texto (positivos hacia abajo, negativos hacia arriba)
- **amplitud** es el tamaño de esa sombra también se le conoce como difumado o desenfoque
- **color** es el color de la sombra

Guardar los cambios y actualizar la página Web en el navegador

Las celdas internas de la tabla tendrán un fondo verde claro, un radio de 5px en los bordes de la celda y una sombra de texto en blanco:

```
.tabla1 tbody td{
    padding:10px;
    text-align:center;
    background-color:#DEF3CA;
    border: 2px solid #E7EFE0;
    -moz-border-radius:5px;
    -webkit-border-radius:5px;
    border-radius:5px;
    color:#666;
    text-shadow:1px 1px 1px #fff;
}
```


Guardar los cambios y actualizar la página Web en el navegador

Ahora se quiere agregar un icono a todas las celdas que hacen referencia a la clase "check". Colocar la siguiente regla de estilo CSS:

```
.tabla1 tbody span.check::before{
    content : url(imagenes/check.jpg)
}
```

::before, selecciona la parte anterior al contenido de un elemento para insertar un nuevo contenido generado (en este caso se trata de una imagen)

Guardar los cambios y actualizar la página Web en el navegador

Ejercicio #3: formulario.html

Crear un formulario sencillo

1. Digite el siguiente código HTML

```
1  <!DOCTYPE HTML>
2  <html>
3  <head>
4  <title>formularios - ejemplo n°1</title>
5  <style type="text/css">
6  <!--
7  BODY {background-color:#ffffaa;}
8  H3 {text-align:center; font-weight:bold; font-size:16pt; color:blue;}
9  HR {background-color:blue;}
10 //-->
11 </style>
12 </head>
13 <body>
14 <h3>Ejemplo de formulario</h3><hr>
15 <form method="get" enctype="text/plain">
16 Digita tu nombre y apellidos:&nbsp;
17 <input type="text" name="nombre" size="40" maxlength="40"><br><br> Elije un tema:<br>
18 <input type="radio" name="tema" value="ciencia" checked>ciencia<br>
19 <input type="radio" name="tema" value="arte">arte<br>
20 <input type="radio" name="tema" value="literatura">literatura<br><br> Marca tus aficiones:<br>
21 <input type="checkbox" name="aficion" value="deporte" checked>Deporte<br>
22 <input type="checkbox" name="aficion" value="musica" checked>Música<br>
23 <input type="checkbox" name="aficion" value="lectura">Lectura<br>
24 <input type="checkbox" name="aficion" value="pesca">Pesca<br>
25 <input type="checkbox" name="aficion" value="pasear">Pasear<br>
26 <input type="checkbox" name="aficion" value="cine">Cine<br><br>
27 Digita la contraseña de usuario:&nbsp;
28 <input type="password" name="password" size="10" maxlength="10"><br><br>
29 <input type="submit" value="enviar datos">
30 <input type="reset" value="borrar datos"><br><br>
31 </form>
32 </body>
33 </html>
```

2. Abra un navegador y visualice la página “formulario.html” para que llene el formulario y lo envíe por correo electrónico dando clic en el botón enviar datos que incluyó en el formulario.

Ejercicio #4: misdatos.html

1. Creación de un formulario con los principales controles dentro de una tabla y envío de los datos con el método POST.
2. Digite en un editor de página Web y digite el siguiente código HTML

```
1  <!DOCTYPE HTML PUBLIC>
2  <HTML>
3  <HEAD>
4  <TITLE>Formulario de datos personales</TITLE>
5  <STYLE type="text/css">
6  <!--
7  BODY {background-color:#5C4033; color:#CCAFAF;}
8  H1 {text-align:center; font-size:20pt;}
9  HR {background-color:#CCAFAF;}
10 //-->
11 </STYLE>
12 </HEAD>
13 <BODY>
14 <H1>Datos personales</H1>
15 <HR size="5">
16 <BR><BR>
17 Llena el siguiente formulario con los datos que se solicitan, por favor:
18 <FORM method="POST" action="mailto:materialic2019@gmail.com"
19 enctype="text/plain">
20 <TABLE border="0">
21 <TR>
22 <TD><B>Nombre: </B></TD>
23 <TD><INPUT type="text" maxlength="20" size="15" name="nombre"></TD>
24 </TR>
25 <TR>
26 <TD><B>Apellido: </B></TD>
27 <TD><INPUT type="text" maxlength="20" size="15" name="apellido"></TD>
28 </TR>
29 <TR>
30 <TD><B>Nacionalidad: </B></TD>
31 <TD>
32 <SELECT>
33 <OPTION value="Salvadoreño">Salvadoreño(a)
34 <OPTION value="Hondureño">Hondureño(a)
35 <OPTION value="Guatemalteco">Guatemalteco(a)
36 <OPTION value="Costarricense">Costarricense
37 <OPTION value="Nicaraguense">Nicaraguense
38 <OPTION value="Panameño">Panameño(a)
39 </SELECT>
40 </TD>
41 </TR>
42 <TR>
43 <TD><B>Sexo: </B></TD>
44 <TD>
45 <INPUT type="radio" name="sexo" value="mas" checked><B>Masculino</B><BR>
46 <INPUT type="radio" name="sexo" value="fem"><B>Femenino</B>
```

[illegible]

- Abra un navegador y cargue la página “misdatos.html” para que llene el formulario y lo envíe por correo electrónico dando clic en la imagen de Enviar correo que incluyó en el formulario.

Ejercicio #5: datos.html

1. Digite el siguiente código HTML.

```

1 <HTML>
2 <HEAD>
3   <TITLE>Ejemplo de Formularios</TITLE>
4   <link rel="stylesheet" type="text/css" href="css/EstiloForm.css">
5 </HEAD>
6 <BODY>
7   <div id="stylized" class="myform">
8     <form id="form" name="form" method="post" action="enviado.html">
9       <h1>Formulario de Registro</h1>
10      <p>Llenar los siguientes campos: </p>
11      <label>Nombre
12        <span class="small">Agregue su nombre</span>
13      </label>
14      <input type="text" name="name" id="name" />
15      <label>Año de nacimiento
16        <span class="small">Seleccione su año de nacimiento</span>
17      </label>
18      <input type="number" name="nacido" min="1900" />
19      <label>Email
20        <span class="small">Agregue un email válido</span>
21      </label>
22      <input type="text" name="email" id="email" />
23      <label>Password
24        <span class="small">Min. 6 caracteres </span>
25      </label>

```



```

26     <input type="password" name="password" id="password" />
27     <button type="submit">Enviar datos</button>
28     <div class="spacer"></div>
29 </form>
30 </div>
31 </BODY>
32 </HTML>

```

Nota: el archivo css se encuentra en los recursos de la guía

- Abra un navegador y cargue la página “datos.html”. Cuando se cargue conteste el formulario y presione el botón Enviar datos cuando haya finalizado.
- Como el formulario anterior se va a enviar a otra página HTML, tal y como se especificó en el atributo action de la etiqueta <FORM>, el código de esta página únicamente mostrará el mensaje formulario enviado. El código de esta página se muestra a continuación. Digítelo y nómbrelo enviado.html.

```

1  <!DOCTYPE HTML>
2  <HTML>
3  <HEAD>
4  <TITLE>Encuesta enviada</TITLE>
5  <STYLE type="text/css">
6  BODY {background-color:#4A766E; color:#FF7F00; font-size:40pt;}
7  H1 {text-align:center;}
8  P {text-align:center; font-size:16pt;}
9  </STYLE>
10 </HEAD>
11 <BODY>
12 <H1>El formulario ha sido enviado</H1>
13 <P>
14 <A href="datos.html" target="_self">Regresar al formulario</A>
15 </P>
16 </BODY>
17 </HTML>

```

Ejercicio #6: FormHTML5.html

- Digite el siguiente código HTML.

```

1  <!DOCTYPE HTML>
2  <HTML>
3  <HEAD>
4  <TITLE>Uso de HTML5</TITLE>
5  <STYLE type="text/css">
6  BODY {background-color:#C0FFF0;}
7  LABEL {font-family:Verdana; font-style:oblique; font-weight:bold;
8  font-size:10pt; color:#205020;}
9  </STYLE>
10 </HEAD>

```



```

11 <BODY>
12 <form>
13 <p><label>Nombre: <input name="name" required autofocus tabindex="1"></label></p>
14 <p><label>E-mail: <input name="email" type="email" required tabindex="3" value="lenguaje@gmail.com" readonly></label></p>
15 <p><label>URL: <input name="url" type="url" tabindex="2"></label></p>
16 <p><label>Edad: <input name="edad" type="number" min="1" max="90" value="1"></label></p>
17 <p><label>Fecha: <input type="date" name="fecha" size="20"></label></p>
18 <p><label>Mes: <input type="month" name="mes" size="20"></label></p>
19 <p><label>Comentario: <textarea name="comment" required></textarea></label></p>
20 <p><input type="submit" value="Enviar"></p>
21 </form>
22 </BODY>
23 </HTML>

```

V. Ejercicios complementarios

En la clase realice los siguientes ejercicios:

1. Cambiar los estilos de los ejercicios a paginas globales (tablas y formularios)
2. Determinar las reglas CSS necesarias para mostrar la siguiente tabla con el aspecto final mostrado en la imagen (modificar el código HTML que se considere necesario añadiendo los atributos class oportunos).

Primero deberá codificarse la tabla en HTML.

Nombre Producto	Venta Semana 1	Venta Semana 2	Venta Semana 3	Venta Semana 4
Pastas	125.89	159.78	130.56	116.57
Detergente	67.45	78.56	90.56	69.34
Sopas	45.67	34.45	45.89	67.80
Espicias	17.56	18.67	15.89	13.23

El resultado final deberá ser como el siguiente:

Nombre Producto	Venta Semana 1	Venta Semana 2	Venta Semana 3	Venta Semana 4
Pastas	125.89	159.78	130.56	116.57
Detergente	67.45	78.56	90.56	69.34
Sopas	45.67	34.45	45.89	67.80
Espicias	17.56	18.67	15.89	13.23

Cumpliendo con las siguientes especificaciones:

1. Alinear el texto de las celdas, cabeceras y título al centro.
2. Definir los bordes de la tabla de color gris oscuro #333.
3. Implementar la pseudoclase hover en las celdas de la tabla donde se encuentran los valores numéricos con el color #FFFF66
4. En los nombres de los productos colocar como fondo una imagen llamada fondo_gris.gif la cual se encuentra en la carpeta imágenes de los recursos de la guía 10

5. Modificar el tipo de letra de la tabla y utilizar Arial (en su defecto Helvetica y posteriormente sans-serif) a tamaño 0.9em. El color azul claro es #E6F3FF.
6. Mostrar un color alterno en las filas de datos (color amarillo claro #FFFFCC).
7. Deberá centrar la tabla

VI. Análisis de resultados

1. El siguiente formulario debe de enviar la información a su correo electrónico

Formulario de registro y evaluación:

Escribe tu nombre:

Introduce tu contraseña:

Envíame una foto tuya:

Este manual te parece:

☒ Muy bueno

☐ Bueno

☐ Malo

☐ Muy malo

☒ Avisarme de actualizaciones

Aplicar estilos CSS al formulario

2. Al diseñar una Web es muy común crear lo que se llama "Libro de visitas". En el podemos pedir a los que nos visitan sus datos y opiniones y con esto mejorar nuestra web. Cree el siguiente formulario con el nombre libro.html y tome en cuenta las indicaciones que siguen después de la imagen del formulario:

Libro de visitas

Datos personales:

Nombre:

Apellidos:

Indica un pseudónimo y una contraseña:

Pseudónimo: Contraseña:

Otros datos:

Edad:

- ☐ 0 - 10
- ☐ 11 - 20
- ☐ 21 - 30
- ☐ 31 - 40
- ☐ 41 o más

¿Dónde vives?

- ☐ Santiago
- ☐ Santo Domingo
- ☐ La Romana
- ☐ San Cristóbal
- ☐ Otros:

Aficiones:

- ☐ Música
- ☐ Deportes
- ☐ Pintura
- ☐ Lectura
- ☐ Otras:

Indicaciones:

- Pseudónimo no puede tener más de 16 caracteres y Contraseña más de 12
- El atributo name de la etiqueta <form> debe ser miformulario
- Cada control ha de tener un atributo name diferente. Para el nombre debe ser name="nombre".
- Los datos del formulario se deben enviar a su cuenta de correo.
- Aplicar estilos CSS al formulario
- Guarde el archivo y ábralo en el navegador para comprobar el resultado.