

Ejercitación 1 - CUESTIONARIO:

1. Qué es HTML, cuando fue creado, cuáles fueron las distintas versiones y cuál es la última?
2. ¿Cuáles son los principios básicos que el W3C recomienda seguir para la creación de documentos con HTML?
3. En las Especificaciones de HTML, ¿cuándo un elemento o atributo se considera desaprobado? ¿y obsoleto?
4. Qué es el DTD y cuáles son los posibles DTDs contemplados en la especificación de HTML 4.01?
5. Qué son los metadatos y cómo se especifican en HTML?

- 1- Es un lenguaje de marcado para la elaboración de páginas web. Es un estándar que sirve de referencia del software que conecta con la elaboración de páginas web en sus diferentes versiones. Define una estructura básica y un código (denominado código HTML) para la definición de contenido de una página web, como texto, imágenes, videos, juegos, entre otros.

HTML es un derivado de SGML y nació en 1991, gracias a Tim Berners-Lee (que trabajaba en el CERN), que ante la necesidad de compartir información entre científicos creó la primera definición del lenguaje.

Los estándares oficiales HTML son el HTML 2.0, el HTML 3.2, el HTML 4.0, el HTML 4.01 y el HTML 5. El HTML 5 es la última especificación oficial y se espera que continúe evolucionando a lo largo de los próximos años. El XHTML, una forma más avanzada del HTML que se suponía iba a sustituir a éste, podemos considerar que ha quedado integrado dentro del HTML 5.

- 2- Separar estructura y presentación El HTML tiene sus raíces en SGML, que siempre ha sido un lenguaje para la especificación de código estructural. A medida que el HTML madura, un número cada vez mayor de sus elementos y atributos presentacionales ha sido reemplazado por otros mecanismos, en particular las hojas de estilo. La experiencia ha demostrado que separando la estructura de un documento de sus aspectos presentacionales se reduce el coste de servir a un amplio espectro de plataformas, medios, etc. y se facilitan las revisiones del documento. Considerar la accesibilidad universal a la Web Para hacer la Web más accesible a todos, en especial a aquéllos con discapacidades, los autores deberían considerar cómo pueden representarse sus documentos en diferentes plataformas: navegadores basados en voz, lectores braille, etc. El HTML ofrece un número de mecanismos con este fin (p.ej., el atributo alt, el atributo accesskey, etc.) Además de esto, los autores deberían recordar que sus documentos pueden llegar a una audiencia muy lejana con diferentes computadoras y configuraciones. Para que los

documentos sean correctamente interpretados, los autores deberían incluir en sus documentos información sobre el idioma natural y la dirección del texto, cómo está codificado el documento, y otras cuestiones relacionadas con la Internacionalización. Ayudar a los agentes de usuario con la representación Incremental Mediante un diseño cuidadoso de las tablas y haciendo uso de las nuevas características de las tablas de HTML 4, los autores pueden ayudar a los agentes de usuario a representar los documentos más rápidamente. Los autores pueden aprender a diseñar tablas para su representación incremental. Los implementadores deberían consultar las notas sobre tablas del apéndice para obtener información sobre algoritmos incrementales.

3-

Desaprobado:

Un elemento o atributo desaprobado es aquel que ha quedado anticuado por la presencia de estructuras nuevas. Los elementos desaprobados se definen en el manual de referencia en los lugares apropiados, pero claramente marcados como desaprobados. Los elementos desaprobados pueden declararse obsoletos en versiones futuras de HTML.

Obsoleto:

Un elemento o atributo obsoleto es aquél para el cual no hay garantía de soporte por parte de un agente de usuario. Los elementos obsoletos han dejado de estar definidos en la especificación, pero se enumeran por motivos históricos en la sección de cambios del manual de referencia.

4- Un DTD, del inglés document type definition, especifica cuáles son los tipos de elementos y atributos que podemos utilizar en nuestra página web. También define las reglas de cómo podemos utilizar estos elementos juntos, de acuerdo a la especificación del language de marcado. La DTD también declara cuales juegos de caracteres podemos utilizar.

Para el HTML 4.01 existen 3 tipos de DTD los cuales son:

Strict, el DTD «HTML 4.01 Strict» hace énfasis en la separación del contenido de la presentación y el comportamiento. Este DTD es el que la W3C recomienda para todas las páginas web actuales.

Transitional, el DTD «HTML 4.01 Transitional» significa que puede ser usado transicionalmente para convertir un documento de las versiones anteriores de HTML 4.01 en uno más moderno, no es recomendable para nuevas páginas web. Este tipo incluye 11 tipos de elementos que están obsoletos pero que se pueden usar.

Frameset, el DTD «HTML 4.01 Frameset» se utiliza para página que usan marcos. debes tener en cuenta que **los marcos son obsoletos** fueron descotinuados por la W3C, se recomienda que en este caso se utilice algún scrip del lado del servidor para reemplazar los marcos.

- 5- Los metadatos son datos que describen datos, es decir, información sobre un documento más que contenido del propio documento. En general, la especificación de metadatos implica dos pasos: - Declaración de una propiedad y de un valor para esta propiedad. Esto puede hacerse de dos maneras: - Desde dentro de un documento, por medio del elemento META. - Desde fuera de un documento, vinculando los metadatos por medio del elemento LINK (véase la sección sobre tipos de vínculos). - Referencia a un perfil en el que se definen la propiedad y sus valores legales. Para designar un perfil, se usa el atributo profile del elemento HEAD.

Ejercicio 2:

a) **<!-- Código controlado el día 12/08/2009 -->**

<!-- -->

Son etiquetas para dar inicio y cierre a un comentario, pueden ser colocados en cualquier parte del documento HTML.

En este punto, le falta la etiqueta de cierre, en caso de ser omitida todo lo que siga a la derecha de la etiqueta de apertura será comentado.

En la salida del navegador no se muestra ni un cambio, porque justamente está comentado.

b) **<div id="bloque1">Contenido del bloque1</div>**

<div> **</div>** son etiquetas que permiten contenedores y son de carácter obligatorias.

El atributo **id**= "bloque1" establece un indicador único al elemento.

Es un elemento en bloque cuya función principal es agrupar en secciones / bloques

Lo que se visualiza en el navegador es:

Contenido del bloque1

c) ****

**** es una etiqueta que representa una imagen en un documento.

Es un elemento vacío solo tiene una etiqueta de apertura, y que no se debe especificar la etiqueta de cierre para ser válido.

El atributo **src** es para colocar la url de la imagen, es obligatorio usarlo al colocar el elemento **img**.

El atributo **alt** define el texto alternativo que describe la imagen, texto que los usuarios verán si la URL de la imagen es errónea o la imagen tiene un formato no soportado o si la imagen aún no se ha descargado.

El atributo **id** establece un indicador único al elemento.

El atributo **name** indica el nombre para el elemento.

El atributo **width** indica el ancho.

El atributo **height** indica el alto.

El atributo **longdesc** complementa al atributo alt y se emplea para ofrecer una descripción más larga del elemento que la proporcionada por el atributo alt. Mientras que el atributo alt contiene el texto alternativo de la imagen, el atributo longdesc contiene una dirección de Internet a otra página web o a la misma página web donde se encuentra la descripción larga de la imagen.

d) `<meta name="keywords" lang="es" content="casa, compra, venta, alquiler" />`
`<meta http-equiv="expires" content="16-Sep-2019 7:49 PM" />`

La **<meta>** etiqueta define metadatos sobre un documento HTML y siempre van dentro del elemento `<head>`.

Las meta **keywords** son una serie de etiquetas que incluyen palabras clave para indicar al buscador cuál es el contenido o la temática de la web.

El atributo **lang** proporciona el idioma, en este caso español.

El atributo **name** hace referencia al nombre al que se asocia la metainformación.

El atributo **content** indica los datos que se quieren asociar al name.

El atributo **http-equiv** aporta información sobre los encabezado de respuesta HTTP, puede usarse en lugar de name.

Esto indica que el documento que contiene esta etiqueta META caducará en esta fecha. Si el documento se solicita después de esta fecha, el navegador debería cargar una nueva copia desde el servidor, en lugar de usar la copia en su caché.

e) `<a href="http://www.e-style.com.ar/resumen.html" type="text/html"`
`hreflang="es" charset="utf-8" rel="help">Resumen HTML `

a es un elemento en línea que representa anclaje o hipervínculo.

El atributo **type** crea campos de texto básicos de una sola línea.

El atributo **Hreflang** especifica el idioma y las restricciones geográficas opcionales para un documento.

El atributo **charset="utf-8"** es el código dominante de internet. UTF-8 se utiliza en la mayoría de los sitios web. Codifica los caracteres más comunes, números básicos e inglés con 8 bits.

El atributo **rel** especifica la relación entre el documento actual y el documento/recurso vinculado y **help** hace referencia a un vínculo a la ayuda sensible al contexto.

f) `<table width="200" summary="Datos correspondientes al ejercicio vencido"><caption`
`align="top"> Título </caption>`
`<tr>`
`<th scope="col"> </th>`
`<th scope="col">A</th>`
`<th scope="col">B</th>`
`<th scope="col">C</th>`

```

</tr>
<tr>
<th scope="row">1</th>
<td>&nbsp;</td>
<td>&nbsp;</td>
<td>&nbsp;</td>
</tr>
<tr>
<th scope="row">2</th>
<td>&nbsp;</td>
<td>&nbsp;</td>
<td>&nbsp;</td>
</tr>
</table>

```

los elementos **table** (representando el contenedor principal), **tr** (representando a las filas contenedoras de las celdas) y **td** (representando a las celdas).

El atributo **summary** se utiliza para especificar el resumen del contenido de la tabla.

El atributo **width** establece el ancho.

Las etiquetas `<caption>` y `</caption>` (ambas obligatorias) dan un título descriptivo a las tablas.

El atributo **align** se utiliza para especificar la alineación del elemento caption, en este caso es top (parte superior).

El atributo **scope** especifica si una celda de encabezado es un encabezado para una columna, una fila o un grupo de columnas o filas. Donde **col** especifica que la celda es un encabezado para una columna, del mismo modo para **row** solo que en este caso se trata de una fila.

La entidad ** ** sirve para representar un espacio en blanco

Ejercicio 3

a) `Click aquí para ir a Google`

En este caso se utiliza la etiqueta `<a>` ``, donde `a` es un elemento en línea que representa un hipervínculo, que con el atributo `href` referenciamos a un enlace web de Google con un texto indicando que al hacer click te llevara a dicho sitio.

`Click aquí para ir a Google`

Este caso es similar al anterior solo que se agrega el atributo `target` que especifica dónde abrir el documento vinculado, al ser `target="_blank"`, `_blank` nos indica que abre el documento vinculado en una nueva ventana o pestaña, en cambio en el anterior el link se abre en la misma ventana.

En este caso no se visualiza en el navegador ya que luego de cerrar la primer etiqueta <a> no se encuentra cerrada y no hay un entro dentro de esta etiqueta para que se pueda visualizar el link en el navegador.

Click aquí para ir a Google

En este caso al hacer click en el texto ‘ Click aquí para ir a Google’ al existir el atributo href="#" es un enlace a ningún lado, se suele utilizar al bosquejar un sitio web y simplemente representar enlaces, por lo tanto el link no redirecciona.

Click aquí para volver arriba

En este caso lo que hace es subir al principio de la página sin tener que arrastrar el dichoso cursor, esto se logra gracias a href="#arriba", sin embargo es necesario declarar anteriormente debajo de la etiqueta <body>:

**** para que funcione, de lo contrario no lo va a hacer.

En este caso no muestra nada en el navegador puesto que no existe el atributo href y no tiene como redireccionar, y al no existir texto entre las etiquetas no se muestra en el navegador.

Se debería colocar la línea de código:

debajo de la etiqueta <body> y agregar debajo **Ir arriba** para que funcione.

b)

<p>Click aquí</p>

La etiqueta <p> se puede cerrar o no, es opcional, es para distribuir textos en párrafos y puede contener elementos en línea dentro, como es el caso, ya que img y a son elementos en línea. Muestra en el navegador una imagen con un texto que proporciona una equivalencia con la imagen y un link hacia Google.

<p> Click aquí</p>

En este caso p > a > img por lo tanto “Click aquí” no es un link, es solo texto ya que se encuentra dentro de la etiqueta <p> </p>. Luego lo que se hace es vincular una imagen con un enlace, que es justamente lo que se muestra en el navegador.

Dentro la etiqueta <a> se encuentra la etiqueta , donde por medio del atributo href nos referimos al url del destino y con src el url de la imagen.

```
<p><a href="http://www.google.com.ar">Click aquí</a></p>
```

Este caso es similar al anterior solo que “Click aquí” se encuentra dentro de la etiqueta <a> . Por lo que en el navegador se muestra vinculado una imagen con “Click aquí” en un enlace

```
<p><a href="http://www.google.com.ar"></a> <a
href="http://www.google.com.ar">Click aquí</a></p>
```

Este caso es similar al anterior solo que se muestra en pantalla por un lado el texto vinculado a la url y por otro la imagen vinculada a la url. Esto se da porque el atributo href es él mismo en ambas etiquetas <a> , solo que aquí están separadas en dos diferentes.

```
c)
<ul>
<li>xxx</li>
<li>yyy</li>
<li>zzz</li>
</ul>
```

En este caso podemos ver en el navegador una lista que no posee orden. El elemento ul define una lista desordenada. El elemento contiene uno o más elementos li que especifican los ítems de la lista.

```
<ol>
<li>xxx</li>
<li>yyy</li>
<li>zzz</li>
</ol>
```

En este caso podemos ver en el navegador una lista ordenada orden. El elemento ol define una lista ordenada. El elemento contiene uno o más elementos li que especifican los ítems de la lista.

```
<ol>
<li>xxx</li>
</ol>
<ol>
<li value="2">yyy</li>
</ol>
<ol>
<li value="3">zzz</li>
```


En este caso se muestra una lista ordenada, el atributo value sirve para colocar un valor específico al ítem de la lista. Se puede decir que establece el valor de un elemento de la lista. Los siguientes elementos de la lista se incrementarán a partir de ese número. El valor debe ser un número y sólo puede ser utilizado en las listas ordenadas ().

<blockquote>

<p>1. xxx

2. yyy

3. zzz</p>

</blockquote>

<blockquote> es una etiqueta que especifica una sección que se cita de otra fuente. Puede contener: texto, y/o cero o más elementos en bloque o en línea.

Por lo que se concluye que muestra en el navegador lista/texto que se encuentra dentro de <p> </p>

d)

<table border="1" width="300">

<tr>

<th>Columna 1</th>

<th>Columna 2</th>

</tr>

<tr>

<td>Celda 1</td>

<td>Celda 2</td>

</tr>

<tr>

<td>Celda 3</td>

<td>Celda 4</td>

</tr>

</table>

En este caso podemos ver en el navegador una tabla con un tamaño indicado en el atributo width y un borde indicado por el atributo border.

<table> (el contenedor principal), <tr> (representando a las filas contenedoras de las celdas) y <td> (representando a las celdas).

<table border="1" width="300">

<tr>

<td><div align="center">Columna 1</div></td>

<td><div align="center">Columna 2</div></td>

</tr>

<tr>

<td>Celda 1</td>


```

<td>Celda 2</td>
</tr>
<tr>
<td>Celda 3</td>
<td>Celda 4</td>
</tr>
</table>

```

El elemento strong es para marcar con especial énfasis las partes más importantes de un texto.

```

e)
<table width="200">
<caption>
Título
</caption>
<tr>
<td bgcolor="#dddddd">&nbsp;</td>
<td bgcolor="#dddddd">&nbsp;</td>
<td bgcolor="#dddddd">&nbsp;</td>
</tr>
<tr>
<td bgcolor="#dddddd">&nbsp;</td>
<td bgcolor="#dddddd">
&nbsp;</td>
<td bgcolor="#dddddd">&nbsp;</td>
</tr>
</table>

```

En este caso se puede ver en el navegador una tabla con un título y celdas color gris, esto se debe por el atributo bgcolor ya que es usado para determinar el color de fondo de un párrafo, tabla o cualquier otra parte del HTML. Las etiquetas <caption> y </caption> (ambas obligatorias) le dan un título/descripción significativa a la tabla.

```

<td colspan="3"><div align="center">Título</div>
</td>
</tr>
<tr>
<td bgcolor="#dddddd">&nbsp;</td>
<td bgcolor="#dddddd">&nbsp;</td>
<td bgcolor="#dddddd">&nbsp;</td>
</tr>
<tr>
<td bgcolor="#dddddd">&nbsp;</td>
<td bgcolor="#dddddd">&nbsp;</td>
<td bgcolor="#dddddd">&nbsp;</td>
</tr>

```

`</table>`

Este caso es muy similar al anterior solo que el título se encuentra dentro de las etiquetas `<div>` `</div>` ambas obligatorias con el atributo `align` para centrar y el uso del atributo `colspan` que define el número de columnas que debe abarcar una celda.

f)

```
<table width="200">
<tr>
<td colspan="3"><div
align="center">Título</div></td>
</tr>
<tr>
<td rowspan="2" bgcolor="#dddddd">&nbsp;</td>
<td bgcolor="#dddddd">&nbsp;</td>
<td bgcolor="#dddddd">&nbsp;</td>
</tr>
<tr>
<td bgcolor="#dddddd">&nbsp;</td>
<td bgcolor="#dddddd">&nbsp;</td>
</tr>
</table>
```

En este caso podemos ver en el navegador la tabla con las tres primeras celdas unidas del título centrado y dos celdas unidas en vertical.

Los atributos `colspan` y `rowspan` permiten unir una celda con las celdas contiguas, tanto horizontal como verticalmente. El valor de `colspan` indica la cantidad de celdas unidas en horizontal y el valor de `rowspan` indica la cantidad de celdas unidas en vertical.

```
<table width="200">
<tr>
<td colspan="3"><div
align="center">Título</div></td>
</tr>
<tr>
<td colspan="2"
bgcolor="#dddddd">&nbsp;</td>
<td bgcolor="#dddddd">&nbsp;</td>
</tr>
<tr>
<td bgcolor="#dddddd">&nbsp;</td>
<td bgcolor="#dddddd">&nbsp;</td>
<td bgcolor="#dddddd">&nbsp;</td>
</tr>
</table>
```

En este caso podemos ver en el navegador la tabla con las tres primeras celdas unidas del título centrado y dos celdas unidas en horizontal.

g)

```
<table width="200" border="1">
<tr>
<td colspan="3"><div
align="center">Título</div></td>
</tr>
<tr>
<td colspan="2" rowspan="2">&nbsp;</td>
<td>&nbsp;</td>
</tr>
<tr>
<td width="50%">&nbsp;</td>
</tr>
</table>
```

En este caso se puede ver en el navegador una tabla con celdas unidas horizontal y verticalmente con bordes y celdas del título centrados unidos horizontalmente.

```
<table width="200" border="1" cellpadding="0"
cellspacing="0">
<tr>
<td colspan="2"><div
align="center">Título</div></td>
</tr>
<tr>
<td rowspan="2">&nbsp;</td>
<td>&nbsp;</td>
</tr>
<tr>
<td width="50%">&nbsp;</td>
</tr>
</table>
```

Igual al caso anterior solo que los bordes modificados, el atributo Cellpadding define el espacio entre el borde de la celda de una tabla y el contenido presente en ella, en cambio, el atributo Cellspacing define el espacio entre celdas adyacentes individuales.

h)

```
<form id="form1" name="form1" action="procesar.php" method="post"
target="_blank">
<fieldset>
<legend>LOGIN</legend>
Usuario: <input type="text" id="usu1" name="usu1" value="xxx" /><br />
Clave: <input type="password" id="clave1" name="clave1" value="xxx" />
```

```

</fieldset>
<input type="submit" id="boton1" name="boton1" value="Enviar" />
</form>

```

En este caso se puede apreciar en el navegador un formulario con un marco y un botón de envío mediante el método post que lo que hace es enviar los datos de manera que no podamos verlo, ocultos al usuario.

```

<form id="form2" name="form2" action="" method="get" target="_blank">LOGIN<br />
<label>Usuario: <input type="text" id="usu2" name="usu2" /></label><br />
<label>Clave: <input type="text" id="clave2" name="clave2" /></label><br />
<input type="submit" id="boton2" name="boton2" value="Enviar" />
</form>

```

En este caso se puede apreciar en el navegador un formulario sin marco y un botón de envío mediante el método get que añade los datos a la URI definida en el atributo de acción del formulario.

```

<form id="form3" name="form3" action="mailto:xx@xx.com" enctype="text/plain"
method="post" target="_blank">
<fieldset>
<legend>LOGIN</legend>
Usuario: <input type="text" id="usu3" name="usu3" /><br />
Clave: <input type="password" id="clave3" name="clave3" />
</fieldset>
<input type="reset" id="boton3" name="boton3" value="Enviar" />
</form>

```

En este caso podemos ver un formulario con envío de mails **fieldset** (grupo de campos) permite organizar en grupos los campos de un formulario.

i)

```

<label>Botón 1
<button type="button" name="boton1" id="boton1">
<br />
<b>CLICK AQUÍ</b></button></label>

```

En este caso se ve en el navegador un botón con una imagen.

```

<label>Botón 2
<input type="button" name="boton2" id="boton2" value="CLICK AQUÍ" />
</label>

```

En este caso se ve en el navegador un botón sin imagen.

j)

```
<p><label><input type="radio" name="opcion" id="X" value="X" />X</label><br />
<label><input type="radio" name="opcion" id="Y" value="Y" />Y</label></p>
```

Se puede ver en el navegador campos de opción donde el elemento input, teniendo el valor "radio" en su atributo type, representa una opción que pertenece a un grupo en el que no más de una opción puede ser seleccionada al mismo tiempo. Para que un conjunto de botones de opción pertenezcan al mismo grupo, todos ellos deberían tener el mismo valor en el atributo name.

```
<p><label><input type="radio" name="opcion1" id="X" value="X" />X</label><br />
<label><input type="radio" name="opcion2" id="Y" value="Y" />Y</label></p>
```

Caso contrario al anterior

k)

```
<select name="lista">
<optgroup label="Caso 1">
<option>Mayo</option>
<option>Junio</option>
</optgroup>
<optgroup label="Caso 2">
<option>Mayo</option>
<option>Junio</option>
</optgroup>
</select>
```

Se puede ver una lista desplegable dividida en dos casos.

La etiqueta **optgroup** se utiliza para agrupar opciones relacionadas en una lista desplegable. Se debe abrir y cerrar

```
<select name="lista[]" multiple="multiple">
<optgroup label=" Caso 1">
<option>Mayo</option>
<option>Junio</option>
</optgroup>
<optgroup label=" Caso 2">
<option>Mayo</option>
<option>Junio</option>
</optgroup>
</select>
```

Similar al caso anterior solo que hay un scroll para poder ver todas las opciones