

Ejercicio 1

1. Qué es HTML, cuando fue creado, cuáles fueron las distintas versiones y cuál es la última?

HTML, siglas en inglés de *HyperText Markup Language* ('lenguaje de marcas de hipertexto'), hace referencia al 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. Es un estándar a cargo del *World Wide Web Consortium* (W3C).

Durante sus primeros cinco años (1990-1995), HTML pasó por una serie de revisiones y experimentó una serie de extensiones, el primer lanzamiento fue en 1992. Con la creación del W3C, el desarrollo de HTML cambió de lugar de nuevo. Un primer intento frustrado de extender el HTML en 1995 conocido como HTML 3.0 luego dio paso a un enfoque más pragmático conocido como HTML 3.2, que se completó en 1997. HTML 4.01 siguió rápidamente más tarde ese mismo año. En 2006, el W3C indicó su interés en participar en el desarrollo de HTML 5.0 después de todo, y en 2007 formó un grupo de trabajo autorizado para trabajar con WHATWG en el desarrollo de la especificación HTML. Apple, Mozilla y Opera permitieron al W3C publicar la especificación bajo los derechos de autor del W3C, manteniendo una versión con la licencia menos restrictiva en el sitio WHATWG.

2. ¿Cuáles son los principios básicos que el W3C recomienda seguir para la creación de documentos con HTML?

1. Separar estructura y presentación.
2. Considerar la accesibilidad universal a la Web.
3. Ayudar con la representación incremental

3. En las Especificaciones de HTML, ¿cuándo un elemento o atributo se considera desaprobado? ¿y obsoleto?

Un elemento o atributo **desaprobado (deprecated)** es aquel que ha quedado anticuado por la presencia de estructuras nuevas. Los elementos desaprobados pueden declararse obsoletos en el futuro.

Un elemento o atributo **obsoleto (obsolete)** es aquél para el cual no hay garantía de soporte por parte de un agente de usuario.

4. Qué es el DTD y cuáles son los posibles DTDs contemplados en la especificación de HTML 4.01?

Un documento HTML válido declara qué versión de HTML se utiliza en él. La declaración del tipo de documento especifica la definición del DTD que se usa en él ISO8879. HTML4.01 especifica tres DTDs, que varían en cuanto a los elementos que soportan:

- El DTD HTML 4.01 Estricto (Strict DTD) incluye todos los elementos y atributos que no han sido desaprobados o que no aparecen en documentos con marcos.

- El DTD HTML 4.01 Transicional(Transitional DTD) incluye todo lo que incluye el DTD estricto más los elementos y atributos desaprobados (la mayoría de los cuales están relacionados con la presentación visual).
- El DTD HTML 4.01 para Documentos con Marcos(Frameset DTD) incluye todo lo que incluye el DTD Transicional más los marcos.

5. Qué son los metadatos y cómo se especifican en HTML?

Los metadatos son definiciones e información que van más allá del contenido que se muestra en la página. Muchas veces estos son usados para que las páginas puedan ser ubicadas en los motores de búsqueda así como tener un mejor puntaje en estos.

Ejercicio 2

2.a)

`<!-- Código controlado el día 12/08/2009 -->`

Este es un comentario, se puede colocar en cualquier parte del documento y no es tenido en cuenta a la hora de mostrar el contenido de la página

2.b)

`<div id="bloque1">Contenido del bloque1</div>`

Sirve para crear secciones o agrupar contenidos. Se utiliza frecuentemente para dar un estilo a otros elementos de HTML utilizando CSS o para realizar ciertas tareas con JavaScript.

El atributo id no es obligatorio, es simplemente una manera de identificar unívocamente un elemento en HTML.

2.c)

``

`` es un tag que se utiliza para insertar imágenes al documento.

alt: Define una línea de texto que provee contenido equivalente para aquellas situaciones en que las imágenes no pueden ser vistas. (Es requerido).

id: (explicado en el punto 2.b.).

Name: OBSOLETO EN HTML5. Un nombre para el elemento, utilizado con propósitos visuales únicamente.

width: ancho de la imagen, por defecto en píxeles.

height: alto de la imagen, por defecto en píxeles.

longdesc: OBSOLETO EN HTML5. Proporciona el URI de un recurso con una explicación más extensa del contenido/propósito de la imagen que la provista en el atributo alt.

2.d)

`<meta>` es una etiqueta que provee metadatos que no pueden ser expresados con otros elementos del encabezado.

`<meta name="keywords" lang="es" content="casa, compra, venta, alquiler" />`

lang: indica el lenguaje

name y keywords: El elemento provee metadatos a nivel de documento organizados en pares nombre-valor. El "nombre" en este par es provisto por el atributo name, mientras que el "valor" es provisto por el atributo content. Keywords sirve por ejemplo, para indicar qué palabras clave se encuentran por ejemplo si alguien realiza una búsqueda.

`<meta http-equiv="expires" content="16-Sep-2019 7:49 PM" />`

http-equiv y content: El elemento http-equiv representa una directiva pragma y su tipo es declarado en el atributo http-equiv. Los

valores para su interpretación son definidos en el atributo content.

2.e)

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

El Elemento HTML Anchor <a> crea un enlace a otras páginas de internet, archivos o ubicaciones dentro de la misma página, direcciones de correo, o cualquier otra URL.

type: Especifica el tipo de medio (media type) en la forma de MIME type para la URL enlazada. Esto es únicamente informativo, sin ninguna funcionalidad incorporada.

hreflang: Este atributo indica el lenguaje humano del recurso al que se enlaza.

charset: OBSOLETO EN HTML5. Define la codificación de caracteres (character encoding) de la URL enlazada.

rel: Especifica la relación del objeto de destino con el objeto de enlace. El valor es una lista separada por espacios de tipos de enlace (link types).

2.f)

```
<table width="200" summary="Datos correspondientes al ejercicio vencido">
  <caption align="top"> Título </caption>
  <tr>
    <th scope="col">&nbsp;</th>
    <th scope="col">A</th>
    <th scope="col">B</th>
    <th scope="col">C</th>
  </tr>
  <tr>
    <th scope="row">1º</th>
    <td>&nbsp;</td>
    <td>&nbsp;</td>
    <td>&nbsp;</td>
  </tr>
  <tr>
    <th scope="row">2º</th>
    <td>&nbsp;</td>
    <td>&nbsp;</td>
    <td>&nbsp;</td>
  </tr>
</table>
```

table: define una tabla dentro del navegador (representa datos en dos o más dimensiones).

caption: Es el encargado de darle un título descriptivo a las tablas.

align: define la alineación del texto en relación a la tabla.

tr: define una fila de celdas en una tabla.

scope: define las celdas con las que el encabezado (definido en <th>) se relaciona.

th: define una celda como la cabecera de una columna en una tabla. La naturaleza exacta de esta cabecera está definida por los atributos.

td: define la celda de una tabla que contiene datos.

Ejercicio 3

Click aquí para ir a Google

Link a google, se abre en la misma pestaña.

Click aquí para ir a Google

Link a google, se abre en una pestaña nueva

No tiene etiqueta de cierre, no muestra nada.

Click aquí para ir a Google

La referencia no hace nada

Click aquí para volver arriba

La referencia te envía al elemento de id arriba

no muestra nada, tiene el identificador "arriba" por el que otros elementos lo pueden detectar o brindar estilos de css.

B)

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

Busca im1.jpg y si no le encuentra muestra el texto "imagen1", luego un link para ir a google.

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

Muestra un link para ir a google, luego busca im1.jpg y si no le encuentra muestra el texto "imagen1".

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

Busca im1.jpg y si no le encuentra muestra el texto "imagen1" (para cualquiera de los dos si se le realiza click redireccionará a google ya que la imagen está dentro de la etiqueta <a>), luego un link para ir a google.

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

Igual que el elemento anterior, solo que dentro de un párrafo, en este contexto no influye en su visualización, sin embargo el párrafo puede permitir una diferente presentación si se añade texto u otros elementos.

C)

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

Lista desordenada, por defecto esta se muestra itemizado.

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

Lista ordenada, por defecto se muestra numerada.

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

Plantea 3 listas ordenadas diferentes, el ítem de la primer lista no recibe un valor específico en el atributo value, por lo que se muestra como 1, los valores de los respectivos ítems de la segunda y tercer lista son seteados mediante el atributo value, por lo que se muestran como 2 y 3 más allá de ser los primeros elementos de sus listas.

```
<blockquote>
    <p>1. xxx<br />
        2. yyy<br />
        3. zzz</p>
</blockquote>
```

Se utiliza la etiqueta <blockquote> para citar elementos provenientes de otras fuentes, en el ejemplo se escriben los números de los elementos manualmente y simplemente se agrega un brake(
) cuando se quiere pasar a la siguiente línea.

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>

```

Tabla, se le da un ancho de 300, con encabezado sin ningún atributo definido y un borde de 1px.

```

<table border="1" width="300">
<tr>
<td><div align="center"><strong>Columna1</strong></div></td>
<td><div align="center"><strong>Columna 2</strong></div></td>
</tr>
<tr>
<td>Celda 1</td>
<td>Celda 2</td>
</tr>
<tr>
<td>Celda 3</td>
<td>Celda 4</td>
</tr>
</table>

```

Tabla, se le da un ancho de 300, se inserta el contenido de las columnas del encabezado dentro de un div para alinearlos al centro (lo que de todas maneras es el default en el encabezado de la tabla) también se le encierra entre etiquetas `` que resaltan el texto en negrita, pero esto también se hace por defecto por ser el encabezado de la tabla, esta tabla también tiene un borde de 1px.

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>
```

Esta tabla se presenta sin bordes y sin encabezado en las columnas, se puede visualizar un título brindado por <caption> que es el encargado de darle un título descriptivo a las tablas, luego se agregan las filas, y a cada uno de sus elementos de celdas(<td>) se le setea un color de fondo mediante el atributo bgcolor (background color), luego dentro del contenido, se utiliza la entidad ** ** (Non Breaking Space que significa espacio sin ruptura) sirve para representar en HTML un espacio en blanco.

```
<table width="200">
<tr>
<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>
```

esta tabla es idéntica a la anterior visualmente, pero se establece el título usando el atributo de fila colspan, donde se configura para que ocupe 3 celdas, luego se utiliza un div con el atributo align="center" para centrar el texto.

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>
```

Muy similar en su estructura al ejercicio anterior, este tiene una celda que ocupa dos espacios de columna.

```
<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>
```

Muy similar en su estructura al ejercicio anterior, este tiene una celda que ocupa dos espacios de fila.

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>
```

Esta tabla propone la primer fila ocupando 3 casilleros pero como solo se utilizan dos simplemente toma todos los espacio disponibles, luego también usa un casillero que ocupa dos columnas con rowspan.

```
<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>
```

Esta tabla es igual que la de arriba visualmente pero se le quita el padding y spacing a las celdas por lo que el borde se vuelve “simple” visualmente.

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>
```

Form que se enviará con método post y atributo action procesar.php el usuario y la contraseña tienen valores por defecto.

```
<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>
```

Form con método get no define atributo en el action y los valores se enviarán en la url, aparte define una etiqueta label donde pone los input y no usa fieldset por lo que no aparece encerrado.

```
<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>
```

Form visualmente igual al primero pero este presenta un action mailto que se utiliza para enviar emails.

l)

<label>Botón 1

<button type="button" name="boton1" id="boton1">

**
**

CLICK AQUÍ</button></label>

Botón al que se le agrega una imagen dentro y el texto se define con para user letras negritas.

<label>Botón 2

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

</label>

Botón al que el contenido se le setea a través del atributo value.

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>
```

```
<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>
```

Entre estas dos opciones de inputs de radio buttons cambia la definición del name y hace que se pueda seleccionar sólo uno, o los dos al mismo tiempo.

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>
```

Lista desplegable para seleccionar con dos grupos en su interior que definen secciones. Se despliega haciendo click en el elemento.

```
<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>
```

Lista desplegable para seleccionar con dos grupos en su interior que definen secciones. Se encuentra desplegada y se recorre scrolleando.