**Ejercitación HTML.**

**Ejercicio 1**

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

HTML es un lenguaje de marcado para la elaboración de páginas web. Sus siglas son HyperText Markup Language (que en español seria, Lenguaje de marcas de hipertexto).

El origen de HTML se remonta a 1980, cuando el físico Tim Berners-Lee, investigador del CERN propuso un nuevo sistema de “hipertexto” para compartir documentos.

En 1990 Berners-Lee escribió y especificó el HTML. Ese mismo año, Berners-Lee y el ingeniero Robert Cailliau del CERN, colaboraron en una solicitud conjunta, pero el CERN no adoptó formalmente el proyecto.

Finalmente, el primer documento formal con la descripción de HTML se publicó en 1991 bajo el nombre “HTML Tags” (Etiquetas HTML).

Las versiones principales fueron HTML 2.0, HTML 3.2, HTML 4.01 y HTML 5. Además de estas versiones, también existen las variantes.

La última versión es HTML 5.

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

Los principios de accesibilidad web que el W3C recomienda seguir son:

**Principio 1: Perceptible:** la información y los componentes de la interfaz de usuario deben ser mostrados a los usuarios en formas que ellos puedan entender.  
**Principio 2: Operable:** Los componentes de la interfaz de usuario y la navegación debe ser manejable.  
**Principio 3: Comprensible:** La información y las operaciones de usuarios deben ser comprensibles.  
**Principio 4: Robustez:** el contenido debe ser suficientemente robusto para que pueda ser bien interpretado por una gran variedad de agentes de usuario, incluyendo tecnologías de asistencia.

https://www.w3.org/WAI/fundamentals/accessibility-principles/es

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

Un elemento o atributo es desaprobado cuando 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.

Los agentes de usuario deberían seguir dando soporte a los elementos desaprobados por razones de compatibilidad con versiones anteriores.

Un elemento o atributo es obsoleto cuando 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.

<http://html.conclase.net/w3c/html401-es/conform.html>

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

DTD o Definición del Tipo de Documento es una descripción de estructura y sintaxis de un documento XML o SGML. Su función básica es la descripción de la estructura de datos, para usar una estructura común y mantener la consistencia entre todos los documentos que utilicen la misma DTD. De esta forma, dichos documentos pueden ser validados, conocen la estructura de los elementos y la descripción de los datos que trae consigo cada documento.

Los DTDs contemplados en HTML 4:

· "strict.dtd" -- DTD estricto por defecto

· "loose.dtd" -- DTD no estricto

· "frameset.dtd" -- DTD para documentos con marcos

Recomendamos a los autores que escriban documentos en DTD estricto.

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

Los Metadatos son elementos HTML que muestran información sobre la propia página web que los contiene. Son usados por los buscadores para definir la información principal de nuestra web (temática, descripción), por lo que será muy importante que lo tengamos correctamente configurados.

Estos elementos meta se encontrarán en el head del código de la web.

La sintaxis es:

<meta name="" content="">

**Ejercitación 2**

HTML está compuesto por un conjunto de elementos que son la base de su estructura. Los elementos están compuestos por dos tags (el de apertura y el de cierre) y el contenido

en el medio (con excepción de los elementos vacíos ). Cada tag puede tener atributos (proporcionan ciertas características como altura, ancho, color, etc.) y eventos (asocian un script que se ejecuta cuando el evento ocurre).

Analizar los siguientes segmentos de código indicando en qué sección del documento HTML se colocan, cuál es el efecto que producen y señalar cada uno de los elementos, etiquetas, y atributos (nombre y valor), aclarando si es obligatorio.

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

Es un comentario, se puede ubicar en cualquier sección del documento.

Las etiquetas de inicio **<!- -** y fin **-->** , son obligatorias.

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

La etiqueta **div** es un contenedor genérico sin un significado semántico en particular. Se utiliza comúnmente en el desarrollo de documentos con propósitos estilísticos, en conjunto con los atributos **style** y **class**. También puede resultar útil para proveer atributos comunes a los elementos contenidos por él, como por ejemplo **lang** o **title**.

El atributo **id=”bloque1”** sirve como identificación del elemento y debe ser único en todo el documento.

Los atributos de inicio **<div>** y cierre **</div>** son obligatorias.

c) <img src="" alt="lugar imagen" id="im1" name="im1" width="32" height="32" longdesc="detalles.htm" />

La etiqueta **img** representa una imagen en el documento.

El atributo **src** indica la url de la imagen y es obligatorio.

El atributo **alt** define el texto alternativo que describe la imagen.

El atributo **id**=”im1” identifica al elemento img en todo el documento.

El atributo **name** es el nombre del elemento y es compatible con versiones anteriores de html, en su lugar se puede utilizar el atributo **id**.

El atributo **width** indica el ancho de la imagen en pixeles o porcentaje.

El atributo **height** indica altura de la imagen en píxeles.

El atributo **longdesc** indica la URL como descripción de una imagen mostrada, complementa al texto de **alt**.

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

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

La etiqueta **meta** se define en el encabezado de una página web, sirve para incluir información de referencia sobre la página: autor, título, fecha, palabras clave, descripción, etc. Pero también se puede utilizar para especificar los encabezados del documento.

El atributo “**name**”especifica un nombre para los metadatos. (application-name, autor, description, generator, keywords, viewport)

El atributo “**http-equiv**” el servidor debe usar el nombre indicado como encabezado.

Ej: Caduca: 16-Sep-2019 7:49 PM

El atributo “**lang**”especifica el lenguaje del contenido web.

El atributo “**content**”da el valor asociado con el atributo **http-equiv** o **name**.

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

La etiqueta **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.

El atributo **href** indica el URL a la que apuntaremos con el enlace.

El atributo **type** especifica el tipo de medio para la URL enlazada. Esto es únicamente informativo, sin ninguna funcionalidad incorporada.

El atributo **hreflang** indica el lenguaje humano del recurso al que se enlaza.

El atributo **charset** define la codificación de caracteres (character encoding) de la URL enlazada. (obsoleto en html5).

El atributo **rel** especifica la relación del objeto de destino con el objeto de enlace. El valor **“help”** indica que el documento de destino contiene información de ayuda sobre el contenido del documento presente.

f) <table width="200" summary="Datos correspondientes al ejercicio vencido">

<caption align="top"> Título </caption> 39606822

<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>

Elemento **table** crea una tabla.

El atributo **width** especifica el ancho en pixeles de la tabla, en este caso de 200.

El atributo **summary** se utiliza para dar una descripción lo mas precisa posible del contenido o tema tratado en nuestra tabla. Esto se hace de cara a los lectores de pantalla que utilizan las personas con minusvalía visual.

El atributo **caption** define un título de tabla, con su atributo align que indica la alineación.

Las filas se crean a través del elemento **tr**, la cual define una fila de celdas en una tabla.

El atributo **th** define una celda como encabezado de un grupo de celdas en una tabla.

El atributo <th scope=""> enumerado define las celdas con las que se relaciona el elemento de encabezado (definido en el <th>).

<th scope="row"> el **row** se relaciona con todas las celdas de la fila a la que pertenece.

<th scope="col"> el **col** se relaciona con todas las celdas de la columna a la que pertenece.

El atributo **td** define la celda de una tabla que contiene datos**.**

“**&nbsp;**” es una cadena de caracteres sirve para representar un espacio en blanco.

**Ejercitación 3 - a**



Link con una URL **absoluta** que apunta a google



Especifica que el link redireccionado sera abierto en una nueva ventana o tab

El atributo target dentro de el tag **<a>** especifica al navegador donde el documento linkeado debe ser cargado , es opcional y su valor default es **\_self**



Este elemento contiene una referencia invalida a google ,

además no tiene el cierre de tag por lo que no es posible renderizar el componente

**Atributos:**

**-type:** espeficica el media type de el tag , los posibles se adjuntan aqui: http://www.iana.org/assignments/media-types/media-types.xhtml

**-hreflang**:especifica el lenguaje de el recurso linkeado , es puramente informativo sin ninguna funcionalidad

**-rel**: especifica la relacion entre el documento y el documento linkeado , los posibles se adjuntan aqui: https://developer.mozilla.org/en-US/docs/Web/HTML/Link\_types

**-charset**: no puede ser aplicado en el elemento <a> solo en <meta> (representa el encoding del documento) y <script> representa el encoding de el documento externo de script



Tanto **href="#top**" como **href="#"** referencian al inicio de la pagina actual



posee una referencia al elemento de id **arriba**

**Ejercitación 3 - b**



Etiqueta párrafo que contiene 2 etiquetas <img> y <a>

<img> contiene al atributo src , que en este caso mediante una url relativa a la pagina actual referencia una imagen

el atributo alt especifica un texto alternativo para la imagen en caso de no poder ser mostrada



**<p>** contiene a **<a>** que contiene a **<img>**  , a referencia a google , img tiene una referencia relativa al documento llamada im1.jpg con un texto alternativo alt que ocupa 2 lineas , al ser parseado elt texto queda como “ima gen1” , al lado de la imagen se agrega el texto “Click aquí” ambos al estar encerrados por el tag <a> funcionan como un solo link



A diferencia del anterior la imagen y el texto “Click aquí” estan cada uno separados englobados por diferentes etiquetas **<a>** funcionando como 2 link separados

**Ejercitación 3 - c**

|  |  |
| --- | --- |
| **Código** | **Renderización** |
| <ul>  <li>xxx</li>  <li>yyy</li>  <li>zzz</li>  </ul> |  |
| **<ul>** lista sin orden (unordered list) , es una colección de cosas sin orden específico , su organización es trivial. | |

|  |  |
| --- | --- |
| **Código** | **Renderización** |
| <ol>  <li>xxx</li>  <li>yyy</li>  <li>zzz</li>  </ol> |  |
| **<ol>** lista ordenada (ordered list) , es una lista de elementos donde el orden es importante.  posee 3 atributos , **reversed** ,que especifica que el orden de la lista debe ser revertido **; start** , que especifica en qué número la lista debe comenzar; **type** , que especifica qué tipo de marcador debe ser utilizado | |

|  |  |
| --- | --- |
| **Código** | **Renderización** |
| <ol>  <li>xxx</li>  </ol>  <ol>  <li value="2">yyy</li>  </ol>  <ol>  <li  value="3">zzz</li>  </ol> |  |
| Son tres listas ordenadas en las cuales su único elemento interno empieza con un valor asignado | |

|  |  |
| --- | --- |
| **Código** | **Renderización** |
| <blockquote>  <p>1. xxx<br />  2. yyy<br />  3. zzz</p>  </blockquote> |  |
| El elemento blockquote se usa para escribir una cita textual o un párrafo exacto y que éste se diferencie del resto del texto.  En este caso esta siendo mal utilizado para representar una lista de elementos.  se puede además apreciar una diferenciación de estilo con respecto a los tag **<ul>** y **<ol>** | |

**Ejercitación 3 - d**

|  |  |
| --- | --- |
| **Código** | **Renderización** |
| <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 con un borde que la rodea de 1 pixel de ancho , de 300 pixeles de ancho  tiene 3 filas , la primera que contiene 2 columnas headers  y las 2 ultimas filas contienen 2 celdas cada una | |

|  |  |
| --- | --- |
| **Código** | **Renderización** |
| <table border="1" width="300">  <tr>  <td>  <div align="center"><strong>Colum  na1</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> |  |
| En vez de utilizar los tags <th> utliza un div que alinea el contenido al centro y un tag strong que define la importancia de el contenido de texto | |

**Ejercitación 3 - e**

|  |  |
| --- | --- |
| **Código** | **Renderización** |
| <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> |  |
| Tabla de 200 pixeles de ancho , con el tag <caption> que debe ser puesto inmediato a el tag <table> que define el titulo descriptivo de la tabla  dentro de la tabla se tienen 2 filas definidas con <tr>ambas contienen 3 columnas <td> que tienen de estilo de fondo el color #dddddd | |

|  |  |
| --- | --- |
| **Código** | **Renderización** |
| <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> |  |
| A diferencia de el anterior la descripción de el titulo esta hecha con una fila que engloba una columna que dentro pose un tag <div> con alineamiento centrado que contiene el texto titulo , el atributo **colspan**  define el ancho de la celda dentro de la columna | |

**Ejercitación 3 - h**

|  |  |
| --- | --- |
| **Código** | **Renderización** |
| <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> |  |
| Se puede observar como se muestra un formulario con un grupo de campos(fieldset), el cual muestra una leyenda, dos cuerpos de texto (Usuario y Clave) seguidos de dos campos de entrada cuyo valor esta seteado en “xxx”, fuera del campo hay un botón el cual si se lo pulsa se abre una nueva pestaña con la URI modifica por el valor del atributo action en este caso “procesar.php” | |

|  |  |
| --- | --- |
| **Código** | **Renderización** |
| <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> |  |
| A diferencia del anterior se elimina el grupo de campos, se incorporan dos Labels que antes eran solo texto seguidos de un campo de entrada respectivamente. En este caso si se pulsa el boton se ejecuta el método del formulario “get” el cual envia la URI del valor del metodo action “” seguido de ? con los valores de las entradas (Boton y los textos) hacia el servidor. | |

|  |  |
| --- | --- |
| **Código** | **Renderización** |
| <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" /> |  |
| Se vuelve a incorporar el grupo de campos, solamente que ahora cuando se pulsa el botón, se resetean los valores de todas las entradas del formulario. | |

**Ejercitación 3 - i**

|  |  |
| --- | --- |
| **Código** | **Renderización** |
| <label>Botón 1  <button type="button" name="boton1" id="boton1">  <img src="logo.jpg" alt="Botón con imagen " width="30" height="20" />  <br />  <b>CLICK AQUÍ</b></button></label> |  |
| El elemento **label** muestra una etiqueta llamada “Botón 1” que contiene un elemento **button**(botón) con imagen incluida en él mediante el elemento **img** con texto alternativo **alt,** de 30(width) pixeles de ancho y 20(height) de alto. También añade el texto “Click aqui” en negrita **b** dentro del botón. | |

|  |  |
| --- | --- |
| **Código** | **Renderización** |
| <label>Botón 2  <input type="button" name="boton2" id="boton2" value="CLICK AQUÍ" />  </label> |  |
| A diferencia del anterior código, este botón no incluye la imagen dentro del mismo ni el texto en negrita. Sino que especifica el nombre del botón a través del atributo **value**. | |

**Ejercitación 3 - j**

|  |  |
| --- | --- |
| **Código** | **Renderización** |
| <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> |  |
| Es un elemento párrafo <p> que contiene un elemento label de nombre X y otro de nombre Y, acompañados de botones de tipo radio, donde se permite seleccionar solo uno. | |

|  |  |
| --- | --- |
| **Código** | **Renderización** |
| <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> |  |
| A diferencia del anterior código, estos botones de radio permiten seleccionar ambas opciones ya que tienen valor del atributo name distinto. | |

**Ejercitación 3 - k**

|  |  |
| --- | --- |
| **Código** | **Renderización** |
| <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> |  |
| Tag <select> que representa un elemento de control que muestra un menu de opciones , dentro de este tenemos el tag <optgroup> que agrupa <option> que sean de la misma especie o grupo | |

|  |  |
| --- | --- |
| **Código** | **Renderización** |
| <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> |  |
| Diferente a el anterior este selector posee el atributo multiple  Este elemento booleano indica que se pueden seleccionar múltiples opciones de la lista. Si no está especificado, solo se podrá seleccionar una opción cada vez | |

**Ejercicio 5**

El ejercicio se realizo en HTML, se adjunta en el repositorio