1)

a) HTML (HyperText Markup Language) es un lenguaje de marcado utilizado para crear páginas web. Permite definir la estructura y el contenido de una página web, utilizando etiquetas o elementos para describir diferentes partes del documento, como encabezados, párrafos, imágenes, enlaces, formularios, etc.

HTML fue creado por Tim Berners-Lee en 1990 como parte de su proyecto World Wide Web (WWW) en el CERN (Organización Europea para la Investigación Nuclear). La primera versión de HTML, llamada HTML 1.0, fue lanzada en 1993. Desde entonces, se han creado varias versiones de HTML, cada una con nuevas características y mejoras.

Estas son las distintas versiones de HTML:

* HTML 2.0: lanzada en 1995, agregó soporte para formularios y scripts.
* HTML 3.2: lanzada en 1997, agregó soporte para tablas, applets, marcos (frames) y capas (layers).
* HTML 4.01: lanzada en 1999, mejoró la semántica del lenguaje y agregó soporte para hojas de estilo en cascada (CSS).
* XHTML 1.0: lanzada en 2000, fue una versión más estricta de HTML que se basó en XML. XHTML 1.1 fue lanzada en 2001.
* HTML5: lanzada en 2014, es la versión más reciente de HTML. Agrega muchas características nuevas, como soporte para multimedia, gráficos vectoriales, geolocalización, almacenamiento en el lado del cliente, y API para dispositivos móviles. Además, HTML5 simplifica la sintaxis de HTML y agrega más semántica al lenguaje.

La última versión de HTML es HTML5. Es compatible con todos los navegadores modernos y es ampliamente utilizado para crear páginas web.

b) El W3C recomienda seguir los siguientes principios básicos para la creación de documentos con HTML:

1. Estructura bien formada: Utilizar una estructura de documentos bien formada, es decir, utilizar etiquetas y elementos de forma adecuada y anidada correctamente.
2. Semántica: Utilizar etiquetas y elementos que reflejen el significado de su contenido, en lugar de utilizarlos solamente para cambiar la apariencia de la página.
3. Accesibilidad: Crear documentos que sean accesibles para todos los usuarios, independientemente de sus habilidades o discapacidades. Utilizar atributos como "alt" para las imágenes y "aria" para la accesibilidad de los elementos interactivos.
4. Separación de contenido y presentación: Separar el contenido de la presentación utilizando hojas de estilo en cascada (CSS) para definir la apariencia de la página.
5. Validación: Validar el código HTML utilizando herramientas de validación para asegurarse de que cumple con los estándares y es compatible con todos los navegadores.

Siguiendo estos principios básicos, se puede asegurar que los documentos HTML sean legibles, accesibles, interoperables y compatibles con diferentes dispositivos y navegadores.

c) En las especificaciones de HTML, un elemento o atributo se considera desaprobado (deprecated en inglés) cuando se ha marcado para ser eliminado en futuras versiones de HTML, lo que significa que aún funciona en los navegadores actuales, pero no se recomienda su uso debido a que puede ser reemplazado por una alternativa más moderna y mejor. En lugar de utilizar elementos o atributos desaprobados, se recomienda utilizar elementos o atributos recomendados y compatibles con las últimas versiones de HTML.

Un elemento o atributo se considera obsoleto cuando se ha eliminado oficialmente de la especificación de HTML, lo que significa que no debería utilizarse en nuevos desarrollos. Los elementos o atributos obsoletos ya no son compatibles con los estándares modernos de HTML y pueden no funcionar en todos los navegadores. En lugar de utilizar elementos o atributos obsoletos, se recomienda utilizar elementos o atributos recomendados y compatibles con las últimas versiones de HTML.

d) DTD es una especificación formal que define la estructura y el contenido de un documento en un lenguaje de marcado. En el caso de HTML, un DTD define qué elementos y atributos son válidos y cómo deben ser utilizados en un documento HTML.

En la especificación de HTML 4.01, se contemplan tres posibles DTDs:

1. Strict DTD: Este DTD es la versión más estricta de HTML y no permite el uso de elementos y atributos obsoletos. Es útil para la creación de documentos web que necesiten ser compatibles con diferentes navegadores y dispositivos.
2. Transitional DTD: Este DTD es una versión menos estricta de HTML que permite el uso de elementos y atributos obsoletos. Es útil para la creación de documentos web que requieren características antiguas o que están en proceso de ser actualizadas.
3. Frameset DTD: Este DTD es una versión de HTML que permite el uso de elementos de marcos (frames). Se utiliza cuando se desea crear documentos web con marcos.

Cada uno de estos DTDs tiene sus propias reglas y restricciones, lo que ayuda a garantizar que los documentos HTML sean compatibles con diferentes navegadores y dispositivos. Al especificar un DTD en un documento HTML, se establece qué elementos y atributos son válidos en ese documento y cómo deben ser utilizados.

e) Los metadatos son información sobre el documento HTML que no se muestra directamente en la página, sino que se utiliza para describir su contenido, su estructura, su autoría y otros detalles relevantes que pueden ser útiles para los buscadores, redes sociales y otros sistemas automatizados.

Los metadatos en HTML se especifican utilizando la etiqueta "meta". La etiqueta "meta" se coloca dentro de la sección "head" del documento HTML y puede tener varios atributos para describir diferentes tipos de metadatos.

Por ejemplo, el atributo "name" se utiliza para indicar el nombre del metadato y el atributo "content" se utiliza para proporcionar el valor del metadato. Algunos de los metadatos más comunes que se pueden especificar en HTML son:

* Descripción: se utiliza para proporcionar una breve descripción del contenido del documento, que se puede mostrar en los resultados de búsqueda.
* Palabras clave: se utiliza para especificar las palabras clave que se relacionan con el contenido del documento, que también pueden ser utilizadas por los motores de búsqueda.
* Autor: se utiliza para indicar el nombre del autor del documento.
* Codificación de caracteres: se utiliza para especificar la codificación de caracteres utilizada en el documento, lo que puede ser útil para los navegadores y otros sistemas automatizados.
* Vista previa de página: se utiliza para proporcionar una imagen en miniatura o una vista previa de la página, que puede ser utilizada por las redes sociales y otros sistemas automatizados para mostrar una vista previa del contenido.

2.a)

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

Este bloque de código es un comentario se puede colocar en cualquier sección del documento ya que no produce ningún efecto en la vista del browser. Puede como bien lo indica su nombre usarse para comentar alguna sección y dar especificaciones del uso que se le da una sección del código HTML.

2.b)

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

Esta etiqueta hace referencia a un div, debe encontrarse dentro de la sección de <body></body>. Se utiliza para delimitar un contenedor. Tiene un atributo id que lo identifica

2.c)

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

Esta etiqueta es la etiqueta que representa a una imagen, lo cual es un recurso externo que puede ser incrustado en el cuerpo de un documento.

La ubicación de dicho recurso externo debe especificarse en el atributo src.

Tiene atributo de largo alto y una referencia hacia una descripción en otra página html.

Tambien tiene id y name como atributo.

Se puede colocar como la etiqueta div dentro de la sección de <body></body>.

2.d)

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

Sección del documento HTML: la etiqueta meta se coloca dentro de la sección "head" del documento HTML.

La etiqueta meta se utiliza para especificar las palabras clave que se relacionan con el contenido del documento.

Atributos:

"name" (obligatorio): se utiliza para indicar el nombre del metadato ("keywords" en este caso).

"lang" (opcional): se utiliza para especificar el idioma del contenido (en este caso, "es" para español).

"content" (obligatorio): se utiliza para proporcionar el valor del metadato, que en este caso son las palabras clave separadas por comas.

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

Sección del documento HTML: la etiqueta meta se coloca dentro de la sección "head" del documento HTML.

La etiqueta meta se utiliza para especificar cuándo el navegador debe considerar que la página ha caducado.

Elemento: meta.

Atributos:

"http-equiv" (obligatorio): se utiliza para especificar el nombre del metadato HTTP que se va a utilizar. En este caso, "expires" indica que se está especificando una fecha de caducidad para el documento.

"content" (obligatorio): se utiliza para proporcionar el valor del metadato, que en este caso es la fecha y la hora en la que la página caduca.

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>

Sección del documento HTML: la etiqueta "a" se coloca dentro del cuerpo del documento HTML, generalmente dentro de un párrafo o una lista.

Efecto: la etiqueta "a" se utiliza para crear un enlace a otra página o recurso.

Elemento: a.

Atributos:

"href" (obligatorio): se utiliza para especificar la dirección URL del recurso al que se va a enlazar

"type" (opcional): se utiliza para especificar el tipo MIME del recurso al que se va a enlazar.

"hreflang" (opcional): se utiliza para especificar el idioma del recurso al que se va a enlazar.

"charset" (opcional): se utiliza para especificar la codificación de caracteres utilizada por el recurso al que se va a enlazar.

"rel" (opcional): se utiliza para especificar la relación entre el documento actual y el recurso al que se va a enlazar.

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>

Este código es una tabla donde se declara la etiqueta <table></table> posse atributos de ancho y un summary para quien quiera ver información extra si es necesario cuando vea el código.

Tambien tiene un caption para poner el título, que tiene un alineamiento en la parte superior .

Las etiquetas <tr> hacen referencia a una fila de una tabla y las etiquetas <th> representa a una celda de encabezado en una tabla.

Por ultimo las etiquetas <td> son datos de una celda

3)

a)

El primer segmento de código crea un enlace a la página principal de Google utilizando el texto "Click aquí para ir a Google". Al hacer clic en el enlace, se redirige al usuario a la página de Google.

El segundo segmento de código es similar al primero, pero con el atributo "target" agregado y establecido en "\_blank". Esto hace que el enlace se abra en una nueva pestaña del navegador en lugar de reemplazar la página actual.

El tercer segmento de código tiene varios atributos adicionales, incluyendo "type", "hreflang", "charset" y "rel". Estos atributos brindan información adicional sobre el recurso al que se va a enlazar, como su idioma y codificación de caracteres. Sin embargo, el valor de la URL tiene un espacio en blanco extra antes de "google.com.ar", lo que puede causar un error de enlace roto.

El cuarto segmento de código crea un enlace de "ancla" que no lleva a ninguna página en particular. En cambio, el "#" en la URL indica que el enlace se utilizará para navegar a una sección específica de la página actual..

El quinto segmento de código crea un enlace de ancla que lleva al usuario de regreso a la sección con el id "arriba" de la página actual. Para que el enlace funcione, se necesita una etiqueta "a" con el atributo "name" establecido en "arriba" o un elemento con el atributo "id" establecido en "arriba" en la parte superior de la sección de la página a la que se desea navegar.

b)

El primer segmento de código incluye una imagen y un enlace. El enlace se establece en todo el texto "Click aquí” para ir a Google y la imagen está en un elemento separado dentro del mismo párrafo..

El segundo segmento de código es similar al primero, pero con la imagen envuelta en el enlace. Esto significa que la imagen en sí misma se convierte en un enlace y puede ser clicada para llevar al usuario a la página de Google. El texto del enlace "Click aquí" se encuentra fuera del elemento de la imagen.

El tercer segmento de código es similar al segundo, pero con el texto del enlace y la imagen en el orden inverso.

El cuarto segmento de código es similar al segundo, pero con un segundo enlace que envuelve el texto "Click aquí". Esto significa que tanto la imagen como el texto son enlaces separados y se pueden hacer clic en cada uno de ellos para llevar al usuario a la página de Google.

c)

El primer bloque de código es una lista desordenada con 3 items de lista

El segundo bloque de código es un lista ordenada con 3 items de lista

El tercer bloque de código son 3 listas ordenadas a las cuales la segunda y tercer lista se les indica a los ítems de la lista que tengan un valor especifico en su numeración.

El cuarto bloque de código es un blockquote que Crea un bloque de texto citado, que por lo general se utiliza para citas de otros autores. Y en este caso se le introdujo una lista manualmente a través de la etiqueta <p> añadiéndole la etiqueta <br/> para saltos de línea

d)

En este caso son 2 tablas distintas en las cuales la primera tabla en su primer fila utiliza la etiqueta table head <th> para indicar un encabezado de columna, en cambio, la segunda tabla utiliza en su primer fila la etiqueta <div>, alinea sus elementos en en el centro y al texto ingresado se le otorga la etiqueta <strong> para indicar que debe ser en negrita para simular ser un texto con más relevancia por ser los encabezados de columna

f)

En ambos casos se utilizan tablas en HTML para mostrar contenido. La primera tabla tiene un ancho de 200 píxeles y una etiqueta <caption> que se utiliza para proporcionar un título para la tabla. La tabla tiene dos filas con tres celdas cada una, cada una con un color de fondo de gris claro.

La segunda tabla también tiene un ancho de 200 píxeles y tiene una fila de encabezado que utiliza la etiqueta <colspan> para fusionar tres celdas en una sola celda que muestra el título de la tabla. La tabla tiene dos filas más, cada una con tres celdas y un color de fondo de gris claro.

La diferencia principal es que la primera tabla utiliza la etiqueta <caption> para mostrar el título de la tabla, mientras que la segunda tabla utiliza una fila de encabezado con la etiqueta <colspan> para fusionar varias celdas y mostrar el título de la tabla en una sola celda.

g)

En este caso ambas tablas son similares la diferencia radica en que a la segunda tabla se le coloca cellpading y cellspacing que cumplirían la función visual de unificar los bordes para que no se noten las líneas dobles de los bordes de la tabla a simple vista

h)

En este segmento hay 3 formularios que simulan una pantalla de login para un usuario.

En el primer y tercer caso tiene una etiqueta <fieldset> para englobar la sección del formulario y la etiqueta <legend> que define la leyenda Login como título de la sección. Tambien tienen 2 etiquetas input para Nombre y Clave la primera de tipo texto y la segunda de tipo password.

Los 3 formularios tienen una etiqueta input que hacen la función de botón:

El primer y segundo input son similares su diferencia radica en que tienen distinto ID y ambos son del tipo “submit” mientras que el input del tercer formulario es de tipo “reset” también tiene un ID diferente al de los demás.

El atributo action en cada formulario es diferente en el primero le pega a un método de php, en el segundo formulario no realiza ninguna acción y en el tecer formulario realiza un envío a un mail.

Todos los formularios tienen un method en el primer y tercer caso son de tipo POST mientras que en el segundo formulario es de tipo GET.

Por último el segundo formulario no posee etiqueta <fieldset>. Usa etiquetas <label>

i)

Aca tenemos 2 etiquetas <label> que dentro tienen botones

La diferencia radica en que el primer label crea el botón mediante la etiqueta <button> y una imagen con la etiqueta <img/> y un texto encerrado en <b>, mientras que el segundo label realiza el botón con la etiqueta <input/> con un atributo de tipo button y un atributo value para indicar su texto interior.

j)

Ambos bloques de código muestran un par de botones de opción (radio buttons) que permiten al usuario seleccionar una opción de entre dos posibles. Sin embargo, hay algunas diferencias importantes entre los dos bloques.

El primer bloque de código muestra dos botones una opción dentro del mismo grupo de opciones, ya que ambos tienen el mismo atributo name. Esto significa que el usuario sólo puede seleccionar una de las dos opciones y, al seleccionar una opción, la otra se deselecciona automáticamente.

Por otro lado, el segundo bloque de código muestra dos botones de opción en dos grupos de opciones diferentes. Cada botón de opción tiene un nombre de grupo diferente (opcion1 y opcion2) y, por lo tanto, el usuario puede seleccionar una opción de cada grupo lo que permite seleccionar ambas opciones al mismo tiempo.

k)

En estos dos bloques de código se puede ver una lista de opciones.

La gran diferencia es que en el primer caso se realiza un dropdown que despliega una lista de opciones y en el segundo caso se puede observar una especie de lista scrolleable esto se debe a que en el segundo caso tiene el atributo multiple.