

Práctica N° 1 HTML

Ejercitación 1

1. HTML son las siglas de Hyper Text Markup Language (Lenguaje de Marcado de Hipertexto). Es el lenguaje estándar de marcado para la creación de páginas web, describe la estructura de una página web y consiste en una serie de elementos. Estos elementos le dicen al navegador cómo mostrar el contenido (principalmente la estructura). La primera especificación se asocia a Tim Berners-Lee en 1991, compuesta por 18 elementos que conformaban el diseño inicial. Las distintas versiones recomendadas por las W3C se listan a continuación:
 - HTML 1.0 (1991).
 - HTML 2.0 (1995).
 - HTML 3.2 (1997).
 - HTML 4.01 (1999).
 - XHTML 1.0 (2000).
 - HTML 5 (2014).
 - HTML5.1 (2017).
 - HTML5.2 (2017).
2. Algunos de los aspectos que verifica W3C son: calidad en el código, accesibilidad para usuarios, accesibilidad para dispositivos, gestión del sitio, entre otros.
3. Un elemento o atributo desaprobado es aquel que ha quedado anticuado por la presencia de estructuras nuevas, pero que aún es soportado por los navegadores, se recomienda no utilizarlo.

Por otro lado, un elemento o atributo se considera obsoleto cuando su uso está completamente desaconsejado y aunque los navegadores puedan seguir soportándolo, no se garantiza que lo hagan de manera indefinida, se busca una transición a soluciones más modernas.
4. Un DTD o Document Type Definition define los elementos y atributos válidos que pueden usarse en un documento HTML y establece cómo estos elementos pueden estar organizados.

Los posibles DTDs contemplados en HTML 4.01 son: DTD Estricto, DTD Transicional y DTD de Conjunto de Marcos.

5. Los metadatos en HTML son datos sobre los datos de una página web. Proporcionan información sobre el contenido y las características del documento HTML que no se muestran directamente al usuario en la visualización del contenido de la página, pero que es importante para los navegadores, motores de búsqueda y otros servicios.

Se especifican en la sección `<head>` del documento HTML usando la etiqueta `<meta>`.

Ejercitación 2

- A. Representa un comentario. Se puede colocar en cualquier parte del documento. Se utilizan para dejar aclaraciones en el código para los desarrolladores, no afectan el contenido de la página. Las etiquetas de apertura y cierre, respectivamente, son: `<!--` y `-->`.
- B. Es un contenedor, utilizado para agrupar contenido y aplicar estilos. Se coloca en el cuerpo del documento HTML, es decir, dentro de la etiqueta `<body>`. Las etiquetas de apertura y cierre son: `<div>` y `</div>`. Y como atributo no obligatorio tenemos `id="valor"`, cuya utilidad es proporcionar un identificador único.
- C. Se utiliza para insertar imágenes. Se coloca dentro del `<body>`. La etiqueta es auto-cerrada y se representa con ``. Como atributos obligatorios tenemos `src=""`, que contiene como valor la URL de la imagen (en este caso vacía), y `alt="lugar imagen"` cuyo valor es el texto alternativo que se mostrará si la imagen no puede ser renderizada. Como atributos no obligatorios tenemos `id="im1"` (identificador único), `name="im1"` (nombre utilizado en forms o scripts), `width="32"` (ancho en píxeles), `height="32"` (alto en píxeles), `longdesc="detalles.htm"` (URL de una descripción larga).
- D. Se utilizan para proporcionar metadatos sobre el documento. No visibles para el usuario, se colocan dentro de `<head>`. Su etiqueta es auto-cerrada y se representa con `<meta/>`. Tenemos como atributos obligatorios:

name="keywords" (nombre del metadato), *content=""* (valor del metadato); y como opcionales: *lang="es"* (idioma de las palabras clave) y *http-equiv="expires"* (indica que el contenido de la página debe considerarse expirado después de la fecha proporcionada en un *content=""*).

- E. Se utiliza para crear enlaces a otras páginas o recursos. Se coloca dentro de `<body>`. Su etiqueta es `<a>`. Como atributos obligatorios encontramos: *href="http://www.e-style.com.ar/resumen.html"* (URL a la que lleva el enlace), mientras que como atributos opcionales tenemos: *type="text/html"* (tipo del recurso enlazado), *hreflang="es"* (idioma del recurso enlazado), *charset="utf-8"* (codificación de caracteres del recurso enlazado), *rel="help"* (tipo de relación entre el documento actual y el recurso enlazado). Además el contenido de la etiqueta es el texto que se mostrará como enlace clickeable *"Resumen HTML"*.
- F. Se utiliza para representar datos tabulares mediante la etiqueta `<table></table>` que se coloca dentro de la sección `<body>`. A su vez, dentro de la tabla encontramos otras etiquetas: `<caption>` cuyo contenido es el título que se muestra para la tabla y cuyo atributo *align="top"* alinea el título en la parte superior (deprecado); `<tr>` que representa una fila; `<th>` que representa un encabezado, encontramos el atributo *scope="col"* que indica que es para una columna o *scope="row"* que indica que es para una fila; `<td>` contiene celdas de datos, en este caso celdas vacías.

Ejercitación 3

- A. En la primera opción, el navegador abrirá la página de Google en la misma pestaña. En la segunda opción, debido al atributo *target="_blank"* abrirá la página en una nueva pestaña. En la tercera opción simplemente encontramos más atributos. En la cuarta opción encontramos que el valor de *href* es *#*, utilizado para anclajes en la misma página, sin embargo no apunta a ningún destino específico. En la quinta opción vemos que *href* apunta a *#arriba*, es decir un elemento con ese id, al hacer click el navegador se desplazará hacia ese elemento. Finalmente encontramos la etiqueta `<a>` que tiene el *id="arriba"*.

- B. En el primer ejemplo, el enlace y la imagen están en el mismo párrafo, pero la imagen no está dentro del enlace, por lo que el enlace solo afecta al texto. En el segundo ejemplo, la imagen está dentro del enlace, pero el texto no. En el tercer ejemplo, tanto la imagen como el texto están dentro del mismo enlace. Finalmente, en el cuarto ejemplo, hay dos enlaces en el mismo párrafo: uno que envuelve la imagen y otro que envuelve solo el texto; esto significa que la imagen y el texto actúan como enlaces separados, cada uno redirigiendo a la URL especificada por su respectivo enlace.
- C. La primera es una lista desordenada, representada con viñetas. La segunda es una lista ordenada, representada con números. La tercera es una lista ordenada, donde además, se representa la numeración con el atributo *value* que debe mostrarse independientemente de la posición del ítem. Finalmente la etiqueta `<blockquote>` se utiliza para citar un bloque de texto, que en este caso contiene un párrafo con texto que simula una lista ordenada.
- D. La primera tabla usa la etiqueta `<th>` para los encabezados, mientras que la segunda tabla usa `<td>` con `<div>` y estilos para crear el efecto visual del encabezado. La primera tabla utiliza la semántica correcta, mientras que la segunda logra un efecto visual similar pero no es una buena práctica.
- E. La primera tabla utiliza la etiqueta `<caption>` para el título de la tabla, que es la forma estándar de lograr ese propósito. Por otro lado la segunda tabla utiliza una celda con atributo `colspan="3"` más un `<div>` dentro para simular el título, lo cual no sería una buena práctica.
- F. En ambas tablas encontramos que el título está incluido en la misma tabla cuando debería ser mostrado a través de `<caption>`. Ambas tablas son de 2x3 (fila por columna). Como diferencias vemos que en la primera tabla la primera celda abarca las dos filas (atributo `rowspan="2"`), mientras que en la segunda tabla la primera celda abarca dos columnas (atributo `colspan="2"`) más una celda simple.
- G. La primera tabla tiene una celda que ocupa tanto columnas como filas, mientras que la segunda tabla sólo extiende verticalmente una celda mediante *rowspan*.

- H. El primer y tercer formulario utilizan métodos *POST*, mientras que el segundo *GET*. El primer formulario envía datos a procesar a *procesar.php* mediante el *action*, el segundo formulario no tiene definición en *action* y el tercero intenta enviar datos por correo electrónico. El primer y segundo formulario incluyen botones tipo *submit* mientras que el tercero tipo *reset*. El primer y tercer formulario utilizan *fieldset* y *legend*, además, el primer formulario incluye valores predeterminados, por otro lado el segundo formulario no tiene valores predeterminados y utiliza la etiqueta *label*.
- I. El primer botón utiliza la etiqueta *button* que permite incluir contenido HTML, como imágenes, mientras que el segundo utiliza la etiqueta *input* con atributo *type="button"*, que sólo permite mostrar texto plano.
- J. El primer ejemplo configura los botones de opción como un grupo único, asegurando que solo se pueda seleccionar una opción a la vez, mientras que el segundo ejemplo configura los botones en grupos separados, permitiendo la selección independiente de cada opción.
- K. El primer ejemplo está diseñado para seleccionar una sola opción entre varias organizadas en grupos, mientras que el segundo ejemplo permite la selección de múltiples opciones dentro de los mismos grupos, mediante el atributo *multiple="multiple"*.

Práctica Complementaria HTML5

Ejercitación 1

1. Ofrece una estructura más semántica a las páginas web mediante etiquetas nuevas como `<header>`, `<footer>`, `<article>`, y `<section>`. Esto hace que el código sea más claro y comprensible, tanto para los desarrolladores como para los motores de búsqueda. Además, HTML5 soporta directamente multimedia mediante los elementos `<video>` y `<audio>`, eliminando la necesidad de plugins externos como Flash para reproducir videos y audios. También mejora los formularios con nuevos tipos de campos como `<input type="email">` y `<input type="date">`, lo que facilita la recolección de datos y la validación. HTML5 incluye elementos como `<canvas>` y SVG para gráficos y animaciones, lo que permite crear visualizaciones interactivas y juegos sin depender de tecnologías adicionales. Otra ventaja importante es el almacenamiento local con Webstorage, que permite guardar datos en el navegador y mejora la rapidez y la experiencia del usuario. Además, ofrece nuevas APIs como Geolocation y WebSockets, que facilitan funcionalidades avanzadas y la comunicación en tiempo real. Finalmente, HTML5 está diseñado para ser compatible con una amplia gama de dispositivos y navegadores modernos, lo que garantiza un buen rendimiento y una experiencia consistente para los usuarios.
2. Porque es el estándar moderno para el desarrollo web y ofrece una serie de mejoras importantes sobre sus predecesores.

Ejercitación 2

1. Para audio, HTML5 es compatible con MP3, WAV, Ogg Vorbis, y Opus.

Ejercitación 3

1. Para video, los formatos soportados incluyen MP4, WebM y Ogg/Theora.