

Práctica 1: El lenguaje HTML

Introducción

HTML significa HyperText Markup Language, es un lenguaje de marcas de hipertexto, que es un elemento esencial de la Web.

Con HTML definimos la estructura de un documento y los diferentes navegadores son capaces de interpretar ese lenguaje y mostrar el contenido del documento de manera legible por los usuarios.

HTML se suele utilizar en combinación con JavaScript, un lenguaje de programación con el que implementa la interactividad y dinamismo del documento, y también con CSS, que es un lenguaje declarativo que permite mejorar la manera en que se muestra el contenido del documento.

Este documento introduce el lenguaje HTML mediante un ejercicio práctico que consiste en realizar una página personal tipo portfolio, que sirva para darnos a conocer en el ámbito profesional.

Este documento sigue el mismo esquema, con modificaciones, de un tutorial disponible en YouTube y realizado por Miguel Ángel Duran García (@midudev y @midulive en YouTube). Las modificaciones han consistido en ampliar información sobre los diferentes elementos HTML utilizando la referencia MSDN HTML.

El resultado final, la web de portfolio, se puede generar siguiendo este documento o también siguiendo el video-tutorial de midu. En cualquier caso, merece la pena conocer los canales de midu y los videos en los que habla de tecnología.

Al final del documento se incluyen las referencias completas.

La referencia oficial de MSDN es el lugar definitivo de consulta sobre esta y otras tecnologías Web.

Preparativos

Para realizar esta práctica hay que utilizar un editor cualquiera (Visual Studio Code, Sublime Text, etc) y un navegador cualquiera (Chrome, Opera, Safari, etc).

En el tutorial que usamos de referencia se utiliza Visual Studio Code (en adelante VSC). Si no tienes VSC o un editor similar, la primera tarea sería instalarlo.

Una vez instalado VSC, se recomienda instalar estos complementos:

- Live Preview: permite tener una vista previa del documento HTML en el que estamos trabajando

- Live server: es un complemento más potente ya que lanza un servidor Web con el que podemos ver el resultado.

Realiza las siguientes acciones:

1. Crea una carpeta (dentro de tu carpeta desarrollo o proyectos) para el curso: "curso-html".
2. Abre VSC
3. Abre la carpeta "curso-html"
4. Crea un archivo index.html (index suele usarse como punto de entrada)
5. Escribe una etiqueta h1 con el texto "Portfolio de tu-nombre": `<h1>Portfolio de tu-nombre</h1>`

Observa que el elemento `<h1>` tiene dos etiquetas: una de inicio `<h1>` y otra de cierre `</h1>`. No todos los elementos tienen etiqueta de inicio y fin.

Diferencia entre etiqueta y elemento: un elemento se construye a partir de etiquetas. Por ejemplo, el elemento div se construye con las etiquetas `<div>` y `</div>`.

Es recomendable escribir las etiquetas en minúsculas (aunque funciona igual con mayúsculas).

Mostrar el resultado:

Tenemos diferentes maneras de comprobar el resultado.

- opción 1: menú de VSC "Run without debugging" (deberemos refrescar para ver los cambios)
- opción 2: opción Preview (no requiere refrescar, pero no funcionan todas las opciones)

Para la opción anterior, en algunas versiones de VSC o editores, existe la opción "Abrir en el explorador". Existen más opciones (desde el Explorador de archivos o Finder) pero no tiene mayor importancia porque utilizaremos alguna de las anteriores.

1. Introducimos un párrafo: etiqueta p con el texto "Ingeniero de Software con X años de experiencia"
2. Queremos destacar el texto Ing. Sw: usamos strong

`<p>Ingeniero de Software con ene años de experiencia</p>`

Observa que las etiquetas no se muestran en la visualización.

3. Usamos etiqueta h2 para mostrar nuestra experiencia.
4. Usamos ul para enumerar cada una de nuestras experiencias profesionales.

```
<h2 id="experiencia">Experiencia</h2>
```

```
<ul>
```

```
<li class="">ESII</li>
```

```
<li class="">Maestranza Aérea</li>
```

```
</ul>
```

5. Comprueba lo que ocurre al eliminar un cierre de li .
6. Comprueba el resultado

Utilizar las DevTools de los navegadores

En el explorador (normalmente será Chrome), botón derecho "inspeccionar". Se abre un panel con las herramientas esenciales para el desarrollador Web.

Por ahora nos fijamos en la pestaña Elements. Despliega el elemento ul y observa que, aunque eliminemos la etiqueta de cierre de li, aquí aparece cerrada. Algunas etiquetas se pueden no cerrar (consultar la documentación)

Imágenes

Hay elementos HTML que se denominan de reemplazo, la etiqueta se reemplaza por contenido. Estos elementos no tienen etiqueta de cierre.

1. Localiza una imagen tuya y ubícala en la carpeta del curso.
2. Incluye un elemento imagen:

```

```

Atributos

Los atributos en HTML proporcionan información adicional. Por ejemplo, el atributo src de la etiqueta img.

Hay dos tipos de atributos:

- globales: como class
- específicos de la etiqueta: como src

Por ejemplo, el atributo alt de img sirve para describir en texto la imagen (esto resulta de gran ayuda para personas con problemas de visibilidad).

Estilos por defecto: user agent stylesheet (comprobar en la pestaña Styles)

1. Incluye el atributo alt en la etiqueta img
2. Comprueba otros atributos de img: width y height.

```

```

Las comillas en los atributos no son obligatorias, salvo que en la cadena tenga espacios como en "foto de JAG".

3. Atributo hidden: sirve para ocultar elementos. Comprueba el funcionamiento.

Basta con incluir el atributo, no hace falta asignarle valor porque es Booleano.

Atributos id y class:

- id sirve para identificar un elemento. Por ejemplo ``
- class: sirve para identificar un conjunto de elementos. Por ejemplo, `<li class="list-item">...`

Estructura básica de un documento HTML

En este apartado vamos a repasar la estructura de un documento HTML.

Los comentarios se definen de este modo:

`<!-- Comentario --->`

La etiqueta `<!DOCTYPE html>` indica el tipo de documento y es la primera que debe aparecer en un documento HTML.

Un documento HTML tiene dos secciones importantes: una zona de metadatos e información diversa que necesita nuestro documento (estilos, librerías, etc) que se define con la etiqueta `<head></head>`, y la sección del contenido que se define con las etiquetas `<body></body>`

Algunos ejemplos de información de metadatos para la sección `<head>`:

- Indicar la codificación que utiliza el documento utilizamos este elemento:
`<meta charset="utf-8">`
- Indicar al navegador que queremos que el documento ocupe todo el ancho disponible del dispositivo:
`<meta name="viewport" content="width=device-width">`
- Definir un color:
`<meta name="theme-color" content="#09f">`
- Indicar el título del documento:
`<title>Portfolio JAG</title>`
- Definir un icono de nuestra página:
`<meta rel="icon" type="image/jpeg" href="...">`

Incluye en tu documento todas las etiquetas meta indicadas.

Etiquetas para SEO (meta)

SEO significa Search Engine Optimization, que se traduce como optimización para motores de búsqueda. Se trata de introducir una serie de elementos en el encabezado de nuestra página para facilitar la búsqueda de los motores de búsqueda.

- meta name="robots"
- meta name="description" content="..."
- Etiquetas Open Graph (véase la siguiente web: <https://ogp.me/>)

Otras etiquetas a considerar:

- <link rel="alternate" href="url" hreflang="es_GB">: document alternativo en otro idioma
- <link rel="canonical" href="url">: indica cuál es la página original

Estilos

En la zona <head> del documento se pueden definir estilos CSS utilizando el elemento <style>.

En este tutorial no vamos a explicar nada sobre CSS, simplemente incluiremos esta sección:

```
<style>
  body {
    background: rgba(208,208,208,0.9);
  }
</style>
```

HTML semántico

El modo correcto de organizar la información en nuestro documento es hacer un buen uso de los elementos HTML. Para ello, conviene conocer las etiquetas semánticas en vez de abusar de etiquetas <div>.

Las etiquetas y <div> deben utilizarse para agrupar, pero no proporcionan ningún significado semántico. Suelen resultar de gran utilidad para agregar estilo a un contenido.

Sin embargo, en HTML hay un conjunto de etiquetas que proporcionan significado, de ahí que se utilice la expresión HTML semántico.

Algunos ejemplos:

- <p> describe un párrafo.
- describe una imagen.

Supongamos que queremos incluir en nuestro documento la siguiente información (tomada de nuestro CV):

- Tecnologías favoritas: HTML, CSS y JavaScript
- Proyectos en los que he participado:
 - o Sitio Web corporativo: Desarrollo de la web corporativa de una empresa. Tecnologías utilizadas: HTML, JavaScript, CSS
 - o Aplicación e-commerce: Desarrollo de una aplicación Web para una tienda. Tecnologías utilizadas: HTML, CSS, JavaScript
 - o Aplicación móvil: Desarrollo de una aplicación móvil para gimnasios. Tecnologías utilizadas: Kotlin, Android Studio

Ejercicio: incluye la información anterior (tecnologías favoritas y proyectos) utilizando únicamente elementos <div>. Comprueba el resultado en el navegador.

Veamos algunos ejemplos de etiquetas semánticas, incluyendo la descripción del elemento según MSDN HTML (puede ser muy conveniente que revises la descripción completa de cada elemento):

- <aside>: “representa una sección de una página que consiste en contenido que está indirectamente relacionado con el contenido principal del documento”.
- <section>: “representa una sección genérica independiente de un documento, que no tiene un elemento semántico más específico para representarla. Las secciones siempre deben tener un título, con muy pocas excepciones”.
- <article>: “representa una composición autocontenida en un documento, una página, una aplicación o en un sitio, que se quiere que sea distribuible y/o reutilizable de manera independiente, por ejemplo, en la redifusión”.

Ahora vamos a estructurar el contenido sustituyendo algunas etiquetas <div> por etiquetas semánticas.

Veamos primero el ejemplo de las tecnologías favoritas:

```
<aside>
  <header>
    <h3>Mis tecnologías favoritas</h3>
  </header>
  <ol>
    <li>HTML</li>
    <li>JavaScript</li>
  </ol>
</aside>
```

A continuación, veamos uno de los bloques de Proyectos (se debe completar agregando dos <article> más dentro de la misma <section>):

```
<h2 id="proyectos">Proyectos realizados</h2>
```

```
<section>
  <article>
    <header>
      <h4>Sitio Web Corporativo</h4>
    </header>
    <p>Desarrollo de un sitio Web corporativo con una interfaz amigable y
moderna</p>
    <footer>
      <small>Tecnologías utilizadas: HTML, CSS, JavaScript</small>
    </footer>
  </article>
</section>
```

El elemento main

“El elemento HTML <main> representa el contenido principal del <body> de un documento o aplicación. El área principal del contenido consiste en el contenido que está directamente relacionado, o se expande sobre el tema central de un documento o la funcionalidad central de una aplicación”.

Coloca la etiqueta <main> de modo que englobe a todo el contenido que hay desde después del <body>, hasta antes del </body>.

Elemento nav

Según aparece en MSDN HTML, “el elemento HTML <nav> representa una sección de una página cuyo propósito es proporcionar enlaces de navegación, ya sea dentro del documento actual o a otros documentos. Ejemplos comunes de secciones de navegación son menús, tablas de contenido e índices”.

Vamos a agregar un elemento <nav> para facilitar el acceso a diferentes secciones del documento: la experiencia personal, los proyectos en los que he participado y un enlace externo (por ejemplo, a un canal propio de YouTube o similar).

El elemento <nav> debe incluirse dentro de un elemento <header>.

Según MSDN HTML: “El elemento de HTML Header (<header>) representa un grupo de ayudas introductorias o de navegación. Puede contener algunos elementos de encabezado, así como también un logo, un formulario de búsqueda, un nombre de autor y otros componentes”.

Vamos a incluir 3 enlaces (<a>): dos a secciones de la propia página.

Los otros dos son a Proyectos y un enlace externo (en el tutorial midu utiliza un enlace a su canal de Twitch). Los enlaces los veremos en el siguiente apartado.

Veamos primero el código y luego la explicación del elemento <a> (este código lo incluimos justo después del <body>):

```

<header>
  <nav>
    <a href="#experiencia">Experiencia</a>
    <a href="#proyectos">Proyectos</a>
    <a href="https://twitch.tv/midudev" target="_blank"
rel="noreferrer">Twitch</a>
  </nav>
</header>

```

El elemento <a>

En MSDN HTML encontramos la definición de este elemento: “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”.

Veamos el enlace usado en el tutorial:

```
<a href="https://twitch.tv/midudev" target="_blank" rel="noreferrer">Twitch</a>
```

Veamos algunos atributos (véase la documentación para más información):

- href: contiene una URL o un fragmento de URL (“#proyectos”)
- target: especifica dónde se desea desplegar la URL, en este caso se indica que se carga la URL en nuevo contexto de navegación (“_blank”)
- rel: indica la relación del objeto destino con el objeto de enlace. En este caso se usa “noreferrer” con el que se indica que no se envíen encabezados del sitio de origen

La URL del atributo href no está restringida a documentos de internet basados en HTTP, sino que puede usar cualquier protocolo soportado por el navegador. Veamos algunos ejemplos:

- a href="mailto:xxxx@xxx.com"
- a href="tel:+3411111111"
- a href="whatsapp://send?text=hola JAG"

En el caso de una imagen, podemos usar el atributo download y nos permitiría descargar la imagen:

```
<a download href="jag.png"><img ....></a>
```

Esto solo funciona con recursos que tengamos en el propio dominio.

Elementos ul y ol

“El elemento ol permite definir listas o viñetas ordenadas (ordered list), bien con numeración o alfabéticamente”.

El elemento ul (unordered list), crea una lista no ordenada.

Veamos algunos atributos de ol:

- type="1" type="a" ...

- reversed
- start="10"

Un li puede tener un valor concreto, por ejemplo: `<li value="5">`.

Estos elementos ya los tenemos incluidos en nuestro documento. Modifica el elemento y comprueba el funcionamiento.

Formularios

Los formularios son elementos importantes, contienen un conjunto de elementos que sirven para que nuestros usuarios puedan enviar información al servidor web.

Los formularios se definen usando el elemento `<form>`.

Entre los diferentes atributos que tiene, destacamos los siguientes:

- action: la URI de una aplicación que procesará la información enviada
- method: define el método HTTP que el navegador usa para enviar el formulario. Hay dos opciones fundamentales:
 - o post: los datos del formulario se incluyen en el cuerpo de la petición
 - o get: los datos del formulario se incluyen como parte de la URL usando "?" como separador

Vamos a incluir en nuestra web un formulario de contacto:

```
<section>
  <h2>Formulario de contacto</h2>
  <form method="post" action="/">
    <fieldset>
      <legend>Información personal</legend>
      <label for="name">Nombre:</label>
      <input type="text" id="name" placeholder="JAG">
    </fieldset>
  </form>
</section>
```

El elemento `<label for="...">` sirve para que al hacer click en el label te lleve al input.

Al poner dos elementos input, se podrían agrupar cada uno con un div.

Agregamos otro input para el email:

```
<label for="em">email:</label>
<input type="email" id="em" placeholder="a@a.es">
```

Podemos usar diferentes valores en type: name, email, tel... Esto es útil porque facilita el autocompletado.

Otros atributos:

- required: el usuario no puede dejar en blanco este campo
- atributo pattern: por ejemplo, para para teléfonos, etc

Elemento select

El elemento select (<select>) de HTML representa un control que muestra un menú de opciones y permite seleccionar una opción.

Más adelante se incluye código para incluir en nuestro documento.

Elemento checkbox

El elemento HTML <input type="checkbox"> permite insertar un vector o array de valores.

En nuestro ejemplo incluiremos un select y un checkbox:

```
<fieldset>
  <legend>Otros</legend>
  <div>
    <label for="why">¿Sobre qué quieres contactar?:</label>
    <select name="why" id="why">
      <option value="1">Trabajo</option>
      <option value="2">Consulta</option>
      <option value="3">Otros</option>
    </select>
  </div>
  <div>
    <label for="terms">Acepta los términos</label>
    <input type="checkbox" id="terms" name="terms" required>
  </div>
</fieldset>
```

Etiqueta datalist

La etiqueta datalist sirve para definir un conjunto de datos estáticos.

Incluye el siguiente código dentro del <fieldset> del apartado anterior:

```
<div>
  <datalist id="lang">
    <option value="JavaScript"></option>
    <option value="Python"></option>
    <option value="C++"></option>
  </datalist>
  <label>Que lenguaje necesitas ayuda?</label>
  <input list="lang" name="lang">
```

</div>

Elementos details y summary

Obtenemos la definición de MSDN HTML: The <details> HTML element creates a disclosure widget in which information is visible only when the widget is toggled into an "open" state. A summary or label must be provided using the <summary> element.

```
<section>
  <h2>Preguntas frecuentes</h2>
  <details>
    <summary>¿Cómo contacto?</summary>
    <p>Enviando un correo a jag@jag.com</p>
  </details>
  <details>
    <summary>¿Cómo me llamo?</summary>
    <p>JAG</p>
  </details>
</section>
```

Diferencias entre submit y button

Tenemos varias maneras de mostrar un elemento botón:

```
<input type="submit" value="Enviar contacto">
<button type="submit">Enviar contacto</button>
<button>Borrar formulario</button>
```

Elementos multimedia (audio, video)

El elemento video se utiliza para insertar videos en un documento HTML.

- Localiza el video "sample_960x400_ocean_with_audio.mp4" y descárgalo en tu carpeta
- Incluye el código para mostrar el video (puedes ponerlo justo después de terminar la sección <header>...</header>)

```
<video
src="https://filesamples.com/samples/video/mp4/sample_960x400_ocean_with_
audio.mp4"></video>
```

Algunos atributos del elemento:

- controls: cambia según el navegador
- autoplay
- muted
- poster="...jpg"

Comprueba el funcionamiento de los atributos.

El elemento audio es similar a video:

```
<audio .... /audio>
```

Elemento iframe

Se usa para integrar otro contenido (otra página HTML) en nuestra página. Es una página sin el borde de la ventana del navegador.

Vamos a ilustrar su funcionamiento insertando un video de YouTube en nuestro documento:

- Abre un video de youtube
- Localiza la opción “compartir”, y selecciona la opción insertar <>
- Esto genera una etiqueta iframe que podemos copiar en nuestra página.

No todas las webs se pueden meter en un iframe (comprueba que Google no se puede).

Problema del iframe que no se ajusta al cambiar el tamaño de la pantalla. Hay que incluir css igual que hemos hecho con la imagen de tesla:

```
<iframe style="width:100%; aspect-ratio:16/9"...
```

Elemento dialog

Definición de MSDN HTML: The <dialog> HTML element represents a modal or non-modal dialog box or other interactive component, such as a dismissible alert, inspector, or subwindow.

```
<dialog>
  <h1>Se puede abrir un modal</h1>
  <p>Es totalmente nativo</p>
  <button>Cerrar</button>
</dialog>
<button>Ver más información</button>
```

Si incluyes el atributo open, el diálogo se abre al cargar la página.

También podemos controlar el momento de abrirlo y cerrarlo usando JavaScript:

```
<script>
  window.show.addEventListener('click',()=>{
    window.modal.showModal()
  });
  window.cerrar.addEventListener('click',()=>{
    window.modal.close()
  })
</script>
```

</script>

Conviene tener en cuenta que la expresión “window.show...” hace referencia a un elemento HTML con id “show” que existe en el documento actual (window). En nuestro caso, el botón que aparece justo después del diálogo debería tener un atributo id=”show”.

Algo similar ocurre con “window.cerrar...”. En este caso el atributo id=”cerrar” debería estar en el botón que hay dentro del diálogo.

Modifica el código de <dialog> para que funcione correctamente según lo indicado.

Referencias

Curso Completo de HTML (2024). Miguel Ángel Durán. Enlace al tutorial original:
<https://youtu.be/3nYLTiY5skU?si=jxt0lv1UWqDC4Ehk>

MSDN HTML <https://developer.mozilla.org/es/docs/Learn/HTML>