

Maquetación y marcado HTML

1. Conceptos básicos I

HTML se define como un lenguaje de marcado, no de programación, y utiliza etiquetas para estructurar el contenido de una página web. Estas etiquetas son elementos que contienen información sobre el contenido y su significado. HTML5 es la versión más reciente, que enfatiza una estructura semántica para mejorar la accesibilidad y el significado del contenido.

Las etiquetas

Una etiqueta permite **clasificar** cada unidad de contenido en el documento HTML según la naturaleza y significado del mismo.

Pueden ser de **dos tipos**:

- De solo apertura (simples): `<nombreEtiqueta>`
- De apertura y cierre (dobles): `<nombreEtiqueta></nombreEtiqueta>`

Internamente, una etiqueta se compone de:

- **Atributos:** proporcionan información adicional.
 - Atributos globales: aplicables a todas las etiquetas.
 - Atributos propios: específicos de cada etiqueta.
- **Contenido:** lo que está dentro de la etiqueta.

Los comentarios

Ayudan a entender el código y recordar su función y en HTML se escriben entre `<!--` y `-->`.

Estructura

- `doctype` - de solo apertura, indica al navegador el tipo de documento que debe interpretar.

- `html` - etiqueta raíz que contiene la totalidad del site.
- `head` - provee información general acerca del documento: título (`title`), metadatos (`meta`), recursos externos (`link`) y código Javascript (`script`).
- `body` - representa el contenido del documento HTML visible a través del viewport.

2. Conceptos básicos II

Contenido estructural y contenido semántico

En HTML4 o XHTML, la estructura se lograba con elementos como capas `div` y cabeceras `h1` a `h6`. Esto presentaba problemas en la semántica del documento, dependiendo excesivamente de los estilos CSS y dificultando la correcta estructuración.

HTML5 introduce nuevos elementos semánticos como `article`, `section`, `nav`, `aside`, `header` y `footer` para mejorar la estructuración del contenido.

Modelo semántico de HTML

La semántica en HTML y HTML5 se refiere al significado que los elementos proporcionan al contenido, permitiendo un mejor rastreo del documento. Dos tipos de elementos:

- **Elementos funcionales**

Tienen una función específica en la página web, como reproducir un vídeo, enviar datos o crear un enlace.

- `<video>`: Permite insertar un vídeo en la página web.
- `<audio>`: Permite insertar un audio en la página web.
- `<form>`: Permite crear un formulario.
- `<input>`: Permite crear un campo de entrada de datos dentro de un formulario.
- `<button>`: Permite crear un botón.
- `<a>`: Permite crear un hipervínculo.
- `<canvas>`: Permite crear gráficos y animaciones.

- **Elementos no funcionales**

Los elementos no funcionales no tienen una función específica y se utilizan para organizar el contenido de la página web.

- `<body>`: Define el cuerpo de la página web, donde se coloca todo el contenido.
- `<header>`: Define la cabecera de una sección o de la página web.
- `<footer>`: Define el pie de página de una sección o de la página web.
- `<article>`: Define un artículo independiente dentro de una página web.
- `<section>`: Define una sección dentro de una página web.
- `<nav>`: Define un área de navegación dentro de una página web.
- `<aside>`: Define un contenido secundario dentro de una página web.
- `<figure>`: Define un contenido independiente, a menudo con un título. Es muy utilizado para describir imágenes, video, ilustraciones, diagramas, fragmentos de código, etc.
- `<figcaption>`: Define un subtítulo o leyenda asociado al contenido del elemento padre `<figure>`.

¿Cómo se estructura semánticamente en HTML5?

El elemento `body` contiene el contenido visible de la página.

Otros elementos como `header`, `footer`, `main`, `section`, `article`, `nav` y `aside` se utilizan para organizar el contenido de manera semántica, proporcionando una mejor indexación.

Otras etiquetas de contenido semántico

Las etiquetas `strong`, `em`, `blockquote`, `cite`, `abbr`, `dfn`, `address`, `ins`, `del` y `s`, añaden información adicional o modifican el estilo del contenido de manera semántica.

3. Elementos: Listados en HTML5

Las listas HTML permiten a los desarrolladores web agrupar un conjunto de elementos relacionados.

Proporcionan una estructura visual para organizar información en un documento web.

Tipos de listas

Dos tipos:

- **Desordenadas**, utilizadas para elementos sin un orden específico (comienzan con la etiqueta ``).
- **Ordenadas** que son utilizadas para elementos que siguen un orden secuencial (comienzan con la etiqueta ``).

En ambos tipos de listas cada elemento de la lista se define con la etiqueta ``.

Atributo type en listas ordenadas

Define el tipo de marcador para los elementos de la lista.

Tipo	Descripción
<code>type="1"</code>	Los elementos de la lista estarán numerados con números (predeterminado)
<code>type="A"</code>	Los elementos de la lista estarán numerados con letras en mayúscula
<code>type="a"</code>	Los elementos de la lista estarán numerados con letras en minúscula
<code>type="I"</code>	Los elementos de la lista estarán numerados con letras romanas en mayúscula
<code>type="i"</code>	Los elementos de la lista estarán numerados con letras romanas en minúscula

Listas de descripción

Son listas de términos con una descripción para cada término.

Se definen con las etiquetas `<dl>` para la lista, `<dt>` para el término, y `<dd>` para la descripción.

Listas anidadas

Son listas que se encuentran dentro de otros elementos de lista. Pueden ser tanto listas ordenadas como desordenadas. Permiten una estructura jerárquica de la información.

4. Elementos: Tablas en HTML5

Las tablas son útiles para crear estructuras a modo de grid. Estas se basa en filas y columnas y para entenderlas no hay más que asimilar la funcionalidad de una tabla en Word.

Al igual que con los elementos de tipo lista, para crear una tabla primero es necesario incluir una etiqueta `<table>` para **declarar su creación**, y a continuación, indicar el **número de filas** que la conforman con la etiqueta `<tr>` (table row).

Atributos

- **Atributo “width”**: Ancho de la tabla, así como de las filas.
- **Atributo “cellpadding”**: Espaciado dentro de una celda.
- **Atributo “cellspacing”**: Espaciado entre celdas.
- **Atributo “border”**: Borde entre las celdas.
- **Atributo “colspan”**: Hacer que la información ocupe más de una columna.
- **Atributo “rowspan”**: Hacer que la información ocupe más de una fila.

Estructura de una tabla

Las siguientes etiquetas proporcionan una estructura semántica a las tablas:

- `<thead>`: Para la cabecera.
- `<tbody>`: Para el cuerpo.
- `<tfoot>`: Para el pie de la tabla.

5. Elementos: Enlaces en HTML5

La etiqueta `<a>` se utiliza para definir enlaces. Se define con el atributo href, que contiene la URL o la acción a realizar. Existen 5 tipos de links:

Links de una web a otra

Permiten la navegación entre diferentes páginas web y se generan con la etiqueta `<a>` y una URL absoluta.

Links de una sección a otra de una misma web

Se utilizan URLs relativas, que no requieren el dominio. Es necesario comprender y seguir una estructura de carpetas para definir las rutas.

Links para generar un email o teléfono

- Se usa el prefijo `mailto:` para abrir el gestor de correo.
- Para enlazar un número de teléfono, se utiliza el prefijo `tel:`.

Links que abren una nueva ventana/pestaña

- Se añade el atributo `target="_blank"` para abrir en una nueva ventana.
- Es una buena práctica advertir al usuario en el texto del enlace.
- Se utilizan anclas con el atributo `id` para enlazar diferentes partes de una misma sección.

Links de una sección a una parte de otra

Se combinan métodos anteriores para dirigir a una página distinta y a una sección específica dentro de ella.

6. Elementos: Imágenes

Organización de imágenes

1. Crear una carpeta para todas las imágenes.
2. Subdividirla si hay muchas imágenes.
3. Usar minúsculas para la nomenclatura y guión medio para espacios.

Para insertar una imagen usamos la etiqueta ``, que a su vez requiere de un atributo `src` para indicar la dirección de almacenamiento de la imagen, así:

```

```

Otros atributos:

- **alt** - Añade una descripción de la imagen.
- **title** - Añade información adicional de la imagen.

Trabajo con imágenes

Formatos de imagen

- **Formato JPEG:** Para imágenes fotográficas o aquellas imágenes que requieran de mucha información de color.
- **Formato PNG:** Para imágenes con colores planos o poca información de color. También para aquellas imágenes que requieran del uso de transparencias y semitransparencias.
- **Formato GIF:** Para imágenes con poca información de color o animaciones. También para imágenes que tengan píxeles o bien 100% opacos, o bien 100% transparentes.
- **El formato vectorial o svg** (Scalable Vector Graphic) permite hacer la imagen independiente de la resolución para así poder cubrir cualquier pantalla, independientemente de su resolución.

Subtítulos

Se usan las etiquetas `<figure>` y `<figcaption>` para subtítulos.

Elementos HTML

- **Bloque:** ocupan todo el ancho y generan saltos de línea.
- **En línea:** ocupan el mínimo espacio horizontal.

Los bloques pueden contener otros bloques o elementos en línea; lo contrario no es posible.

7. Creación y gestión de formularios

Un formulario web es un conjunto de controles que permiten a los usuarios introducir datos y enviarlos al servidor web para su procesamiento. Estos controles incluyen botones, cajas de texto, casillas de verificación, entre otros.

Elementos de un Formulario Web

Se dividen en cuatro grupos fundamentales:

- **Campos de búsqueda:** Permiten al usuario ingresar términos para buscar.
- **Escribir texto:** Incluye campos para escribir texto, tanto en una sola línea como en múltiples líneas.

- **Elegir entre varias opciones:** Permite al usuario seleccionar una opción entre varias, usando radio buttons, checkboxes, o dropdowns.
- **Enviar formularios:** Contiene elementos para enviar los datos introducidos en el formulario al servidor.

Etiqueta `<form>`

Delimita un formulario en HTML. Sus atributos más importantes son:

- **action:** Indica el agente que procesará los datos enviados al servidor.
- **method:** Define cómo se enviarán los datos al servidor, ya sea utilizando GET o POST.

Etiquetas para crear controles

`<input>`, `<button>`, `<select>`, `<optgroup>`, `<option>`, y `<textarea>` son las etiquetas básicas. Se pueden estructurar los controles con `<fieldset>` y `<legend>`.

Atributos Importantes

- **Atributo type:** Indica el tipo de control.
- **Atributo value:** Establece el valor inicial del control.
- **Atributo name:** Identifica al control y es necesario para enviar los datos del formulario.
- **Atributo required:** Hace que un control sea obligatorio.
- **Atributo placeholder:** Muestra un texto de ayuda en el control.
- **Atributos maxlength y minlength:** Establecen la longitud máxima y mínima de entrada para un control de texto.
- **Atributo disabled:** Deshabilita el control.
- **Atributo readonly:** Hace que el control sea de solo lectura.

Accesibilidad

La etiqueta `<label>` mejora la accesibilidad asociando un control con un texto.

- **Relación Implícita:** El control está dentro de la etiqueta `<label>`.
- **Relación Explícita:** Se usa el atributo `for` para asociar el control. El control debe tener un id.

Controles

Los controles permiten al usuario ingresar información y se crean principalmente con la etiqueta `<input>`. Se pueden usar varios tipos de controles como: Texto, Número, Botones, Contraseña, Email, URL, Rango, Color, Fecha, Hora, Semana, Archivos adjuntos, etc

Casillas de verificación

Se crean con `<input>` y el atributo `type` establecido como `checkbox`. Se envían sólo si están marcadas. El atributo `value` determina el valor enviado por el formulario. Útiles para estructuras de preguntas multirespuesta.

Botones Radio

- Se crean con `<input>` y `type` establecido como `radio`.
- Los botones con el mismo atributo `name` forman un grupo, permitiendo una única selección.
- Se envían sólo si uno está marcado.
- El atributo `value` define el valor enviado por el formulario.

Menú desplegable

- Se crea con la etiqueta `<select>`, y cada opción con `<option>`. El valor enviado es el texto de la opción, a menos que se especifique un valor con el atributo `value`.
- El atributo `selected` establece la opción predeterminada.
- Se aconseja incluir una opción en blanco para evitar confusiones.

Área de texto

Se usa para textos largos. Se crea con la etiqueta `<textarea>`, con atributos `rows` y `cols` para definir tamaño inicial. El atributo `placeholder` puede mostrar un texto de ayuda dentro del área de texto.

Control oculto `<input type="hidden">`

- Se crea con `<input>` y `type` establecido como `hidden`.
- No se muestra en pantalla pero es útil para enviar datos invisibles.
- Puede utilizarse para mantener información entre diferentes páginas de un formulario, como en formularios largos divididos en varias partes.