

Programación Web Desde Cero

HTML Semántico

¡Hola! 🖐️

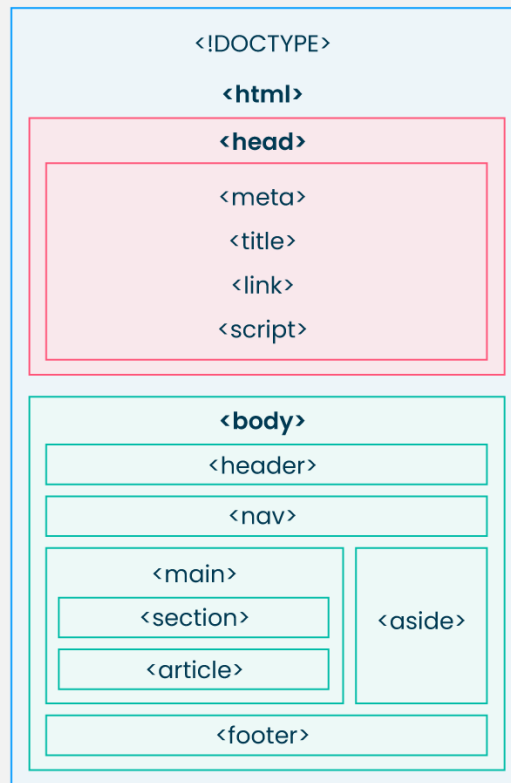
Ahora nos enfocaremos en cómo mejorar el contenido de tu web y su organización. Veremos la forma adecuada de crear y vincular páginas, agregar medios y aplicar buenas prácticas de accesibilidad que capten la atención de los usuarios.

¡Comencemos! 🚀

Etiquetas semánticas

Las **etiquetas semánticas** en el HTML que son elementos que proporcionan información adicional sobre la estructura y el significado del contenido dentro de un documento web.

Estructura página web



A diferencia de las etiquetas no semánticas, las etiquetas semánticas tienen un propósito y significado específicos.

A continuación, te recordamos la utilidad de algunas de estas etiquetas:

- **`<header>`**: Se utiliza para incluir contenido introductorio o de navegación en la parte superior de una página como el logo del sitio, el título de la página, el menú de navegación y otros elementos relacionados con su identidad.
- **`<nav>`**: Se utiliza para definir una sección de navegación dentro de un sitio web. Suele contener enlaces a otras páginas o secciones del mismo sitio.
- **`<main>`**: Se utiliza para envolver el contenido principal de la página excluyendo encabezados, pies de página, barras laterales u otras secciones. Ayuda a los usuarios a enfocarse en la información más relevante. Es importante tener en cuenta que solo debe haber un elemento `<main>` por documento HTML.

- **<article>**: Se utiliza para identificar una sección independiente o autónoma de contenido dentro de un documento. Por ejemplo, puede ser utilizado para un artículo de noticias, una publicación de blog o un comentario.
- **<section>**: Se utiliza para agrupar contenido relacionado dentro de un documento. Puede ser utilizado para dividir el contenido en diferentes secciones temáticas o para organizar el contenido en una página.
- **<aside>**: Se utiliza para marcar el contenido relacionado pero no esencial para el flujo principal del documento. Puede contener información adicional como publicidad, enlaces relacionados, citas o elementos destacados.
- **<footer>**: Se utiliza para incluir información de contacto, enlaces relacionados, derechos de autor y otros elementos que suelen ubicarse en la parte inferior de una página. Proporciona una ubicación específica para la información de cierre.

💡 Etiquetas `<div>` y ``: Aunque no son etiquetas semánticas en sí mismas, las etiquetas `<div>` y `` se utilizan comúnmente como contenedores genéricos para agrupar y estilizar elementos. Pueden ser útiles para organizar el contenido y aplicar estilos específicos, pero se recomienda utilizar las etiquetas semánticas cuando sea posible.

En resumen, **el uso adecuado de todas estas etiquetas proporciona una estructura clara y semántica, facilitando la navegación y mejorando la accesibilidad del sitio web**. De igual forma, permiten a los motores de búsqueda y tecnologías de asistencia comprender mejor el contenido y presentarlo de manera adecuada a los usuarios.

Sin HTML semántico



Con HTML semántico



💡 Otro recurso a tener en cuenta es el de los **comentarios** en el código HTML (`<!-- Comentario -->`). No afectan la visualización del contenido en el navegador, pero **son útiles para agregar notas y aclaraciones dentro del código**. Puedes utilizar comentarios para explicar la estructura de tu página, hacer anotaciones o proporcionar instrucciones para otros desarrolladores.

Etiquetas div y span

Recomendamos usar HTML semántico siempre que se pueda, ya que ayuda a las los navegadores a interpretar la estructura de su contenido y le brinda un vocabulario estandarizado para organizar sus páginas web. **Sin embargo**, a veces se necesita un elemento contenedor cuando ninguno de los elementos HTML semánticos que acabamos de cubrir tendría sentido. **No hay nada de malo en usar un <div>.**

Por ejemplo, si queremos agrupar contenido para eventualmente centrarlo luego, lo podemos englobar en un **<div>** sin problemas.

El punto es que no uses elementos semánticos solo por usarlo. Así que si alguna vez tienes dudas, utiliza un **<div>** en su lugar.

Etiqueta div

La etiqueta **<div>** en sí misma no tiene un significado semántico específico, lo que significa que no aporta ninguna información contextual sobre su contenido. Su propósito principal es permitir la estructuración y el diseño de la página web mediante la aplicación de estilos CSS y la manipulación con JavaScript. Esto permite a los desarrolladores web dividir y diseñar su contenido de manera más controlada.

Etiqueta span

La etiqueta **** en HTML es otro elemento fundamental que se usa para aplicar estilos o manipular partes específicas del contenido dentro de un elemento más grande. A diferencia de la etiqueta **<div>**, que es un **contenedor de bloque**, la etiqueta **** es un **contenedor en línea**, lo que significa que no genera un salto de línea y solo abarcará el ancho del contenido que contiene.

🔗 Veremos qué significa elementos en bloque y en línea a continuación

Elementos en bloque vs. elementos en línea

En HTML existen elementos en **bloque (block)** y elementos en **línea (inline)**. Los **elementos en bloque ocupan el total del ancho que se establece** (esto significa que los elementos en bloque comienzan en una nueva línea y empujan a otros elementos hacia abajo). Mientras que **los elementos en línea ocupan solo el espacio necesario dentro de una línea de texto**, permitiendo que otros elementos se sitúen junto a ellos en la misma línea.

Veamos un ejemplo:



Como mencionábamos, la etiqueta `<div>` es una etiqueta de bloque y se utiliza para agrupar varias etiquetas de HTML para que se puedan crear secciones y aplicarles estilo en conjunto.

Imágenes y Links

Otro punto clave al momento de diseñar una web son las **imágenes y los links**. Vamos a ver más en detalle estos dos elementos.

Imágenes

Las imágenes se incluyen en las páginas web con la etiqueta `` de la siguiente manera: ``

La etiqueta `` tiene dos atributos obligatorios:

- **src:** que especifica la ruta a la imagen
- **alt:** especifica un texto alternativo para la imagen. Es útil en el caso que por algún motivo la imagen no se puede visualizar en el navegador y también le

permite a buscadores como Google saber de qué se trata una imagen, dado que los mismos solo pueden comprender texto y no píxeles.

💡 *Para que la imagen se visualice correctamente se debe asociar a la ruta adecuada. Si la imagen la guardo en la carpeta imágenes, entonces la ruta quedará de la siguiente manera ****. Si la imagen se encuentra suelta en el mismo índice que mi archivo HTML, entonces solo tendrás que vincular el nombre.*

Atributo alt

Dentro de las comillas de este atributo colocaremos una breve descripción de la imagen. Esta etiqueta no es indispensable, pero presenta varias utilidades. La sintaxis quedaría de esta manera:

```

```

En primer lugar, puede ser utilizado para el posicionamiento de la página en buscadores. El buscador puede extraer las palabras clave determinadas en el atributo ALT y ayudar a comprender la función o contenido de la imagen, lo que le permite comprender la función o contenido de la página en sí mismo. Esta es una de las maneras en las que se trabaja el correcto posicionamiento web en buscadores (SEO, Search Engine Optimization).

Además, cuando la página no ha sido todavía cargada, el navegador podría mostrar esta descripción, lo que permitirá al usuario comprender lo que está sucediendo en ese sitio.

Links

Los **enlaces o links** en HTML son elementos que permiten a los usuarios **navegar de una página web a otra o acceder a diferentes recursos en la web**, como documentos, imágenes, videos, entre otros.

Los enlaces son fundamentales para la navegación en la web, ya que permiten crear una estructura interconectada de páginas y contenido.

Para crear un enlace en HTML, se utiliza la etiqueta **<a>** (ancla), que tiene el siguiente formato básico:

```
<a href="URL_del_destino">Texto o contenido del enlace</a>
```

Donde:

- **href:** es el atributo obligatorio que indica la dirección URL del destino al que apunta el enlace. Texto o contenido del enlace: es el contenido visible del enlace que el usuario puede hacer clic.

Por ejemplo, si quieres crear un enlace a la página de inicio de Google, el código HTML sería:

```
<a href="https://www.google.com">Ir a Google</a>
```

Los enlaces también pueden apuntar a recursos locales, como otros archivos HTML en el mismo sitio web. En ese caso, solo necesitas proporcionar la ruta relativa al archivo destino:

```
<a href="otra_pagina.html">Ir a otra página</a>
```


¿Qué son las rutas relativas o absolutas?

La diferencia principal entre una **ruta relativa** y una **ruta absoluta** radica en cómo se especifica la ubicación de un recurso, como un archivo, dentro de un sistema de archivos o una estructura web.

Ruta Relativa:

Una ruta relativa **especifica la ubicación de un recurso en relación con la ubicación del archivo HTML actual**. Es decir, se establece la ruta relativa al archivo actual. Esto es útil cuando se trabaja dentro de una estructura de carpetas y se quiere mantener la flexibilidad al mover o copiar el contenido a diferentes ubicaciones.

Por ejemplo, si tienes la siguiente estructura de carpetas:

```
sitio_web/  
├─ index.html  
├─ paginas/  
│   ├─ pagina1.html  
│   └─ pagina2.html  
└─ imagenes/  
    ├─ imagen1.jpg  
    └─ imagen2.jpg
```

Si estás dentro del archivo **index.html** y quieres enlazar a **pagina1.html**, usarías una ruta relativa como esta:

```
<a href="paginas/pagina1.html">Ir a Página 1</a>
```

Ruta Absoluta:

Una ruta absoluta **especifica la ubicación completa de un recurso desde la raíz del sistema de archivos o el dominio web**. Esto significa que se proporciona la ruta completa desde el inicio, independientemente de dónde se encuentre el archivo HTML actual. Las rutas absolutas son útiles cuando se necesita enlazar a recursos fuera de la estructura del sitio actual.

Por ejemplo, si estás en el archivo `index.html` y deseas enlazar a **`https://www.ejemplo.com/otrapagina.html`**, usarías una ruta absoluta:

```
<a href="https://www.ejemplo.com/otrapagina.html">Ir a Otra Página</a>
```

📌 **En resumen**, una ruta relativa se basa en la ubicación actual del archivo HTML y ofrece flexibilidad al mover el contenido, mientras que una ruta absoluta proporciona la ubicación completa desde el inicio y se utiliza para enlazar a recursos externos o cuando se necesita especificar la ubicación exacta en un sistema de archivos.

Resumen

¡Felicidades!

Esperamos que esta información te haya sido de utilidad y que te lleves para repasar:

- Qué son y por qué son útiles las etiquetas semánticas en html.
- Diferencias entre `div`, `span` y etiquetas semánticas.
- Qué son los elementos en bloque y los elementos en línea y cómo se comportan
- Cómo insertar imágenes y links en tu web