

# HTML 5 – Teoría

## Etiquetas básicas

1. `<header>`: Este elemento se utiliza para definir el encabezado de un sitio web o de una sección específica. Por lo general, contiene elementos como logotipos, títulos, y elementos de navegación que son relevantes para la parte superior de la página.
2. `<nav>`: El elemento `<nav>` se utiliza para agrupar enlaces de navegación, como menús o listas de enlaces que ayudan a los usuarios a moverse dentro del sitio web. Puede contener enlaces a diferentes páginas o secciones del sitio.
3. `<main>`: El elemento `<main>` define el contenido principal de una página. Debe haber solo un elemento `<main>` en todo el documento HTML, y este elemento contiene el contenido central y relevante de la página.
4. `<section>`: El elemento `<section>` se utiliza para dividir y organizar el contenido en secciones temáticas o relacionadas. Debe contener un encabezado (`<h1>` - `<h6>`) que describa el contenido de la sección. Ayuda a mejorar la estructura semántica del documento.
5. `<footer>`: El elemento `<footer>` se coloca al final de una página o sección y contiene información de pie de página. Esto puede incluir datos de contacto, enlaces a políticas, derechos de autor y otros elementos relacionados con la parte inferior del contenido.
6. `<aside>`: El elemento `<aside>` se usa para representar contenido que es secundario o complementario al contenido principal de la página. Puede contener información relacionada, anuncios, enlaces adicionales u otros contenidos relevantes pero no esenciales.
7. `<div>`: El elemento `<div>` es un contenedor genérico que se utiliza para agrupar y estructurar otros elementos sin ningún significado semántico específico. Se utiliza cuando no hay un elemento más apropiado disponible para cumplir con el propósito deseado.

Nota: el uso correcto de estos elementos no solo mejora la estructura semántica del código HTML, sino que también ayuda en la accesibilidad y el SEO, lo que beneficia tanto a los usuarios como a los motores de búsqueda al comprender mejor la jerarquía y organización del contenido en una página web.

```
<!DOCTYPE html>
<html>
<head>
  <title>Titulo de la Página</title>
</head>
<body>
  <header>
    <h1>Encabezado del Sitio</h1>
    <nav>
      <ul>
        <li><a href="#">Inicio</a></li>
        <li><a href="#">Acerca de</a></li>
        <li><a href="#">Servicios</a></li>
        <li><a href="#">Contacto</a></li>
      </ul>
    </nav>
  </header>

  <main>
    <section>
      <h2>Sección 1</h2>
      <p>Contenido principal de la sección 1.</p>
    </section>

    <section>
      <h2>Sección 2</h2>
      <p>Contenido principal de la sección 2.</p>
    </section>
  </main>

  <aside>
    <h3>Contenido Adicional</h3>
    <p>Este es un contenido adicional que acompaña al contenido principal.</p>
  </aside>

  <footer>
    <p>&copy; 2023 Nombre de la Compañía. Todos los derechos reservados.</p>
  </footer>
</body>
</html>
```

## <head>

En HTML, la sección ``<head>`` se utiliza para incluir metadatos, enlaces a recursos externos y scripts que son necesarios para el funcionamiento y la presentación del documento HTML. Aquí hay una lista de algunos elementos comunes que se encuentran en la sección ``<head>``:

1. ``<meta>``: Se utiliza para proporcionar metadatos sobre el documento, como el conjunto de caracteres utilizado, descripciones, palabras clave, autor, etc.

```
<meta charset="UTF-8">
<meta name="description" content="Descripción de la página">
<meta name="keywords" content="palabras clave, separadas por comas">
<meta name="author" content="Nombre del autor">
```

2. ``<title>``: Define el título del documento que se muestra en la barra de título del navegador o en la pestaña.

```
<title>Título de la página</title>
```

3. ``<link>``: Se utiliza para enlazar el documento HTML a hojas de estilo (CSS), iconos, fuentes, archivos de JavaScript y otros recursos externos.

```
<link rel="stylesheet" href="styles.css">
<link rel="icon" href="favicon.ico" type="image/x-icon">
```

4. ``<script>``: Se utiliza para incluir scripts de JavaScript que se ejecutarán en la página.

```
<script src="script.js"></script>
```

5. ``<style>``: Se puede utilizar para agregar estilos CSS directamente en el documento, aunque es más común colocar los estilos en un archivo separado y enlazarlo con la etiqueta ``<link>``.

```
<style>  
  /* Estilos CSS aquí */  
</style>
```

6. ``<base>``: Se utiliza para especificar la URL base para todos los enlaces relativos en el documento.

```
<base href="https://www.ejemplo.com/">
```

## `<body>`

En HTML, la sección ``<body>`` se utiliza para incluir el contenido visible de la página web. Aquí es donde se coloca todo el contenido que los usuarios verán en el navegador, como texto, imágenes, videos, enlaces, formularios y otros elementos interactivos. A continuación, se enumeran algunos de los elementos comunes que se encuentran dentro del ``<body>``:

1. Encabezados: Se utilizan para representar los diferentes niveles de títulos en la página.

```
<h1>Título de nivel 1</h1>  
<h2>Título de nivel 2</h2>  
<h3>Título de nivel 3</h3>  
<!-- y así sucesivamente -->
```

2. Párrafos: Se utilizan para estructurar el texto en párrafos.

```
<p>Este es un párrafo de texto.</p>  
<p>Este es otro párrafo de texto.</p>
```

3. Imágenes: Se utilizan para mostrar imágenes en la página.

```

```

4. Enlaces: Se utilizan para crear hipervínculos a otras páginas web o recursos.

```
<a href="https://www.ejemplo.com">Enlace a ejemplo.com</a>
```

5. Listas: Se utilizan para crear listas ordenadas o no ordenadas.

```
<ul>
  <li>Elemento 1</li>
  <li>Elemento 2</li>
  <li>Elemento 3</li>
</ul>

<ol>
  <li>Elemento 1</li>
  <li>Elemento 2</li>
  <li>Elemento 3</li>
</ol>
```

6. Formularios: Se utilizan para crear elementos interactivos, como campos de entrada, botones y selecciones.

```
<form>
  <label for="nombre">Nombre:</label>
  <input type="text" id="nombre" name="nombre" required>

  <label for="correo">Correo electrónico:</label>
  <input type="email" id="correo" name="correo" required>

  <input type="submit" value="Enviar">
</form>
```

7. Elementos multimedia: Se utilizan para incrustar contenido multimedia, como videos y audios.

```
<video src="video.mp4" controls>
  Tu navegador no soporta el elemento video.
</video>

<audio src="audio.mp3" controls>
  Tu navegador no soporta el elemento audio.
</audio>
```

8. Otros elementos: Además de los mencionados, hay muchos otros elementos en HTML que se utilizan para diversos propósitos, como tablas, divisiones de contenido, elementos de interacción, etc.

La estructura del contenido en el ``<body>`` debe ser coherente y semánticamente significativa para que los motores de búsqueda y los lectores de pantalla puedan interpretar el contenido correctamente y brindar una experiencia de usuario más accesible y comprensible.

## Etiquetas básicas

En HTML, hay varias etiquetas que puedes utilizar para agregar texto a tu página web. Aquí tienes algunas de las etiquetas más comunes para agregar contenido de texto:

1. ``<p>``: Define un párrafo de texto.

```
<p>Este es un párrafo de texto.</p>
```

2. ``<h1>`` a ``<h6>``: Define encabezados de diferentes niveles (h1 es el nivel más alto y h6 es el más bajo).

```
<h1>Título de nivel 1</h1>
<h2>Título de nivel 2</h2>
<h3>Título de nivel 3</h3>
<!-- y así sucesivamente -->
```

3. ``<span>``: Define una sección de texto en línea y se usa para aplicar estilos específicos o manipular parte del contenido.

```
<p>Este es un <span style="color: blue;">texto en azul</span>.</p>
```

4. ``<strong>`` y ``<em>``: Sirven para resaltar texto, donde ``<strong>`` generalmente se muestra en negrita y ``<em>`` en cursiva.

```
<p>Este es un texto <strong>importante</strong> y este es un texto
<em>énfasis</em>.</p>
```

5. ``<blockquote>``: Define un bloque de texto citado, generalmente indentado.

```
<blockquote>
  Este es un bloque de texto citado. Lorem ipsum dolor sit amet...
</blockquote>
```

6. ``<pre>``: Define un bloque de texto preformateado, donde el texto mantiene los espacios y saltos de línea originales.

```
<pre>
  Este es un bloque de texto preformateado.
  Línea 1
  Línea 2
</pre>
```

7. ``<br>``: Se utiliza para agregar un salto de línea dentro de un párrafo u otro contenido.

```
<p>Este es un texto<br>con un salto de línea.</p>
```

8. ``<ul>`` y ``<ol>``: Se utilizan para crear listas desordenadas y ordenadas, respectivamente.

```
<ul>
  <li>Elemento 1</li>
  <li>Elemento 2</li>
</ul>

<ol>
  <li>Elemento 1</li>
  <li>Elemento 2</li>
</ol>
```

9. ``<li>``: Define un elemento de lista dentro de una lista desordenada u ordenada.

10. ``<a>``: Se utiliza para crear enlaces a otras páginas o recursos.

```
<a href="https://www.ejemplo.com">Enlace a ejemplo.com</a>
```

## Imágenes `<img>`

Para agregar imágenes en HTML, se utiliza la etiqueta ``<img>``. Esta etiqueta es una etiqueta de autocompletado, lo que significa que no tiene una etiqueta de cierre y solo requiere atributos para funcionar correctamente. Aquí está la sintaxis básica de la etiqueta ``<img>``:

```

```

Donde:

- ``src``: Es el atributo requerido que especifica la ruta o URL de la imagen que se mostrará en la página. Puede ser una ruta relativa o una URL absoluta.

- ``alt``: Es el atributo requerido que proporciona un texto alternativo para la imagen. Este texto se mostrará si la imagen no se puede cargar o si el usuario utiliza un lector de pantalla. Es importante incluir un texto descriptivo para mejorar la accesibilidad de la página.

Aquí hay un ejemplo completo de cómo agregar una imagen en HTML:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Ejemplo de imagen en HTML</title>
  </head>

  <body>
    <h1>Mi página web con una imagen</h1>
    
    <p>Esta es una imagen de un hermoso paisaje.</p>
  </body>
</html>
```

La etiqueta ``<img>`` no es la única opción para agregar imágenes en HTML. Aunque es la forma más común y sencilla, también existen otras opciones para mostrar imágenes en una página web. Aquí algunas alternativas:

1. ``<picture>``: La etiqueta ``<picture>`` es una forma más avanzada de mostrar imágenes y permite proporcionar versiones diferentes de la imagen para diferentes tamaños de pantalla o resoluciones. Se utiliza en combinación con las etiquetas ``<source>`` y ``<img>``.

```
<picture>
  <source srcset="imagen-large.jpg" media="(min-width: 1200px)">
  <source srcset="imagen-medium.jpg" media="(min-width: 800px)">
  
</picture>
```

2. CSS de fondo (`background-image`): En lugar de utilizar la etiqueta ``<img>``, también puedes usar CSS para establecer una imagen de fondo en un elemento HTML.

```
<div style="background-image: url('imagen.jpg'); width: 800px; height: 600px;"></div>
```

3. SVG (`<svg>`): Si deseas mostrar gráficos vectoriales, puedes utilizar la etiqueta ``<svg>`` para incrustar imágenes vectoriales en tu página.

```
<svg width="100" height="100">
  <circle cx="50" cy="50" r="40" fill="blue" />
</svg>
```



4. Imágenes de fondo en CSS: También puedes establecer imágenes de fondo utilizando CSS para los elementos que desees.

## HTML

```
<div class="imagen-de-fondo"></div>
```

## CSS

```
.imagen-de-fondo {  
  background-image: url('imagen.jpg');  
  width: 800px;  
  height: 600px;  
}
```

## `<figure>`

La etiqueta ``<figure>`` es otra opción en HTML para mostrar imágenes y proporcionar una estructura semántica más significativa cuando se necesita asociar una imagen con su leyenda o descripción. La etiqueta ``<figure>`` se utiliza para representar contenido independiente, como imágenes, gráficos, ilustraciones, diagramas, etc., y puede combinarse con la etiqueta ``<figcaption>`` para agregar una leyenda o descripción a la imagen.

El uso de ``<figure>`` y ``<figcaption>`` proporciona una mejor estructura semántica en comparación con simplemente usar la etiqueta ``<img>`` y un párrafo ``<p>`` para la descripción. Esto es especialmente útil para mejorar la accesibilidad y la indexación por parte de motores de búsqueda.

## Ejemplo completo:

```
<!DOCTYPE html>  
<html>  
  <head>  
    <title>Ejemplo de figura en HTML</title>  
  </head>  
  
  <body>  
    <h1>Mi página web con una imagen y leyenda</h1>  
    <figure>  
        
      <figcaption>Un hermoso paisaje natural.</figcaption>  
    </figure>  
  </body>  
</html>
```

El uso de ``<figure>`` y ``<figcaption>`` es adecuado cuando la imagen y la leyenda son elementos relacionados y deben presentarse juntos. Si la descripción de la imagen es simplemente un

texto adicional o no está directamente relacionada con la imagen, entonces puedes usar otras etiquetas como párrafos `

` o encabezados `

# ` a `` para proporcionar el contenido descriptivo de manera más apropiada.

## Enlaces <a>

En HTML, los enlaces se crean utilizando la etiqueta ``, que significa "anchor" (ancla). Los enlaces permiten a los usuarios navegar entre páginas web o acceder a recursos externos al hacer clic en el texto o en una imagen que actúe como enlace.

La sintaxis básica para crear un enlace es la siguiente:

```
<a href="URL-del-enlace">Texto del enlace</a>
```

Donde:

- `href`: Es el atributo requerido que especifica la URL de destino del enlace. Puede ser una dirección web (URL absoluta) o una ruta relativa dentro del sitio web.

- `Texto del enlace`: Es el texto visible que se mostrará en el enlace y en el que los usuarios pueden hacer clic para acceder al destino.

Aquí hay algunos ejemplos de cómo usar la etiqueta `` para crear diferentes tipos de enlaces:

Enlace a otra página web:

```
<a href="https://www.ejemplo.com">Visitar ejemplo.com</a>
```

Enlace a una página dentro del mismo sitio web (ruta relativa)

```
<a href="/pagina2.html">Ir a Página 2</a>
```

Enlace a una dirección de correo electrónico (usando el protocolo "mailto"):

```
<a href="mailto:info@ejemplo.com">Contáctenos</a>
```

Enlace a un archivo para descargar:

```
<a href="ruta-del-archivo.pdf" download>Descargar PDF</a>
```

Enlace que se abre en una nueva pestaña o ventana del navegador (usando el atributo `target="\_blank"`):

```
<a href="https://www.otro-ejemplo.com" target="_blank">Abrir enlace en nueva ventana</a>
```

También puedes usar imágenes como enlaces. En este caso, simplemente coloca la etiqueta `` dentro de la etiqueta ``:

```
<a href="https://www.ejemplo.com">  
    
</a>
```

Los enlaces son una parte esencial de la navegación web y es importante asegurarse de que estén correctamente configurados y proporcionen una descripción significativa del destino para mejorar la accesibilidad y la experiencia del usuario.

## ¿Qué es “hero”?

En el contexto de desarrollo web utilizando HTML y CSS, el término "hero" se refiere a una sección o componente visualmente destacado en una página web. Una sección "hero" generalmente es la primera sección que los visitantes ven cuando cargan la página y se utiliza para captar su atención de manera impactante. Suele ocupar una parte significativa de la pantalla y se utiliza para presentar contenido importante, como un mensaje de bienvenida, una imagen atractiva, un llamado a la acción (CTA) u otro contenido relevante.

Las características típicas de una sección "hero" pueden incluir:

1. **Tamaño destacado:** La sección "hero" suele ocupar gran parte de la pantalla inicial, a menudo utilizando la altura completa (viewport height, o vh) para asegurarse de que el contenido sea impactante y esté bien visible.
2. **Imágenes o contenido visualmente llamativo:** Puede incluir imágenes de alta resolución, videos de fondo o elementos gráficos que atraigan la atención del espectador.
3. **Texto y llamados a la acción:** Suelen tener mensajes concisos y poderosos que transmiten la esencia del sitio web y alientan a los visitantes a realizar una acción específica, como explorar más, registrarse o comprar.
4. **Fondo y estilos atractivos:** Se pueden aplicar estilos de diseño, como superposiciones de color, efectos de paralaje, animaciones sutiles u otros efectos visuales que hagan que la sección "hero" sea visualmente atractiva.
5. **Responsividad:** La sección "hero" generalmente se diseña para ser receptiva, lo que significa que se adapta de manera efectiva a diferentes tamaños de pantalla, desde computadoras de escritorio hasta dispositivos móviles.

En HTML y CSS, se puede crear una sección "hero" utilizando elementos de marcado HTML, como `<div>`, `<header>`, `<section>`, etc., y luego aplicar estilos de diseño utilizando CSS para lograr el aspecto deseado. Dependiendo de la complejidad del diseño, se pueden usar propiedades CSS como `background-image`, `background-color`, `padding`, `margin`, `font-size`, `text-align`, entre otras, para controlar la apariencia de la sección "hero".