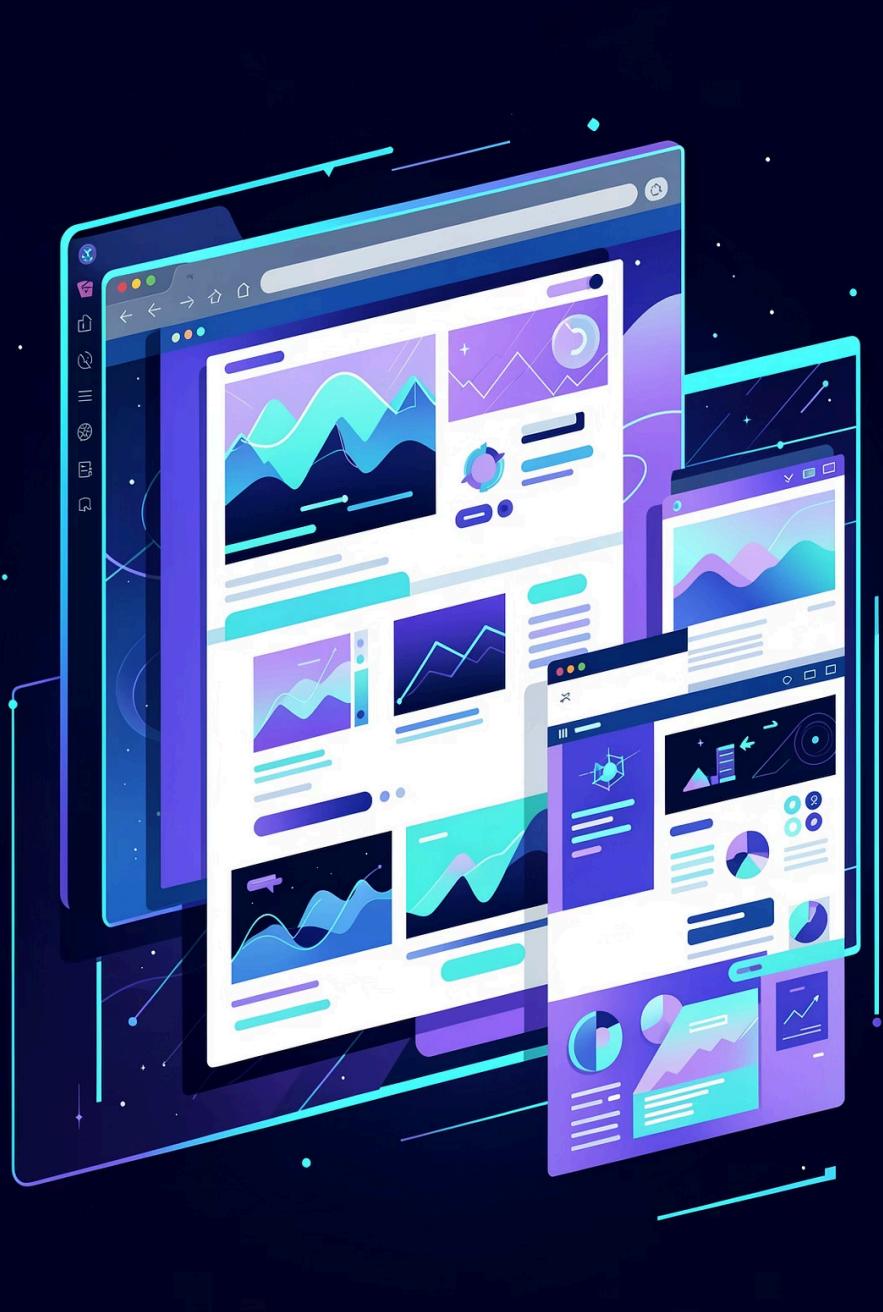




# HTML: HyperText Markup Language

Fundamentando el desarrollo web desde 1991



# ¿Qué es HTML?

## Lenguaje de marcado

Sistema de etiquetas que define la estructura y contenido de documentos web mediante elementos semánticos.

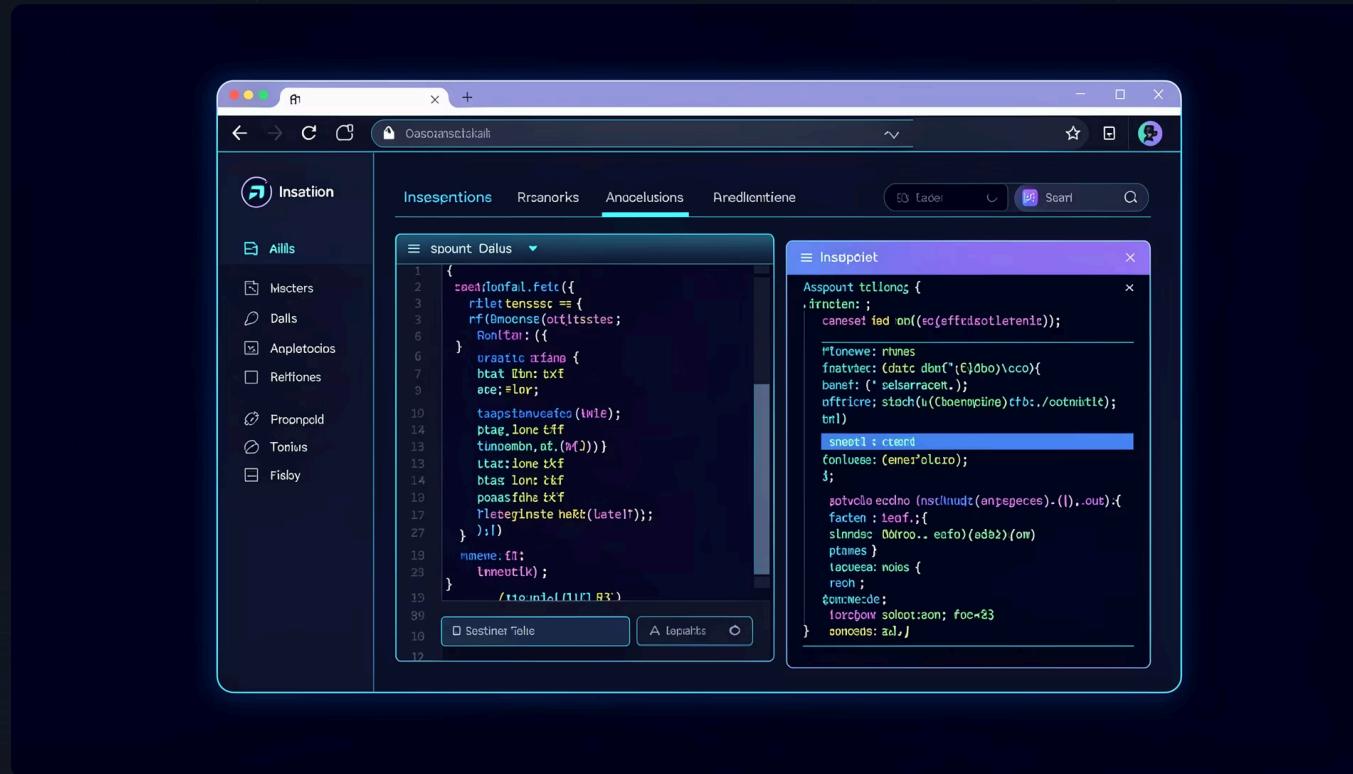
## Base del desarrollo web

Constituye el esqueleto de todas las páginas web, proporcionando el contenido que se renderiza en navegadores.

## Estándar W3C

Formato estandarizado por el World Wide Web Consortium para garantizar compatibilidad entre plataformas.

# ¿Cómo funciona HTML?



- El navegador lee el archivo .html
- Interpreta las etiquetas y estructura
- Renderiza el contenido visual
- Muestra la página al usuario

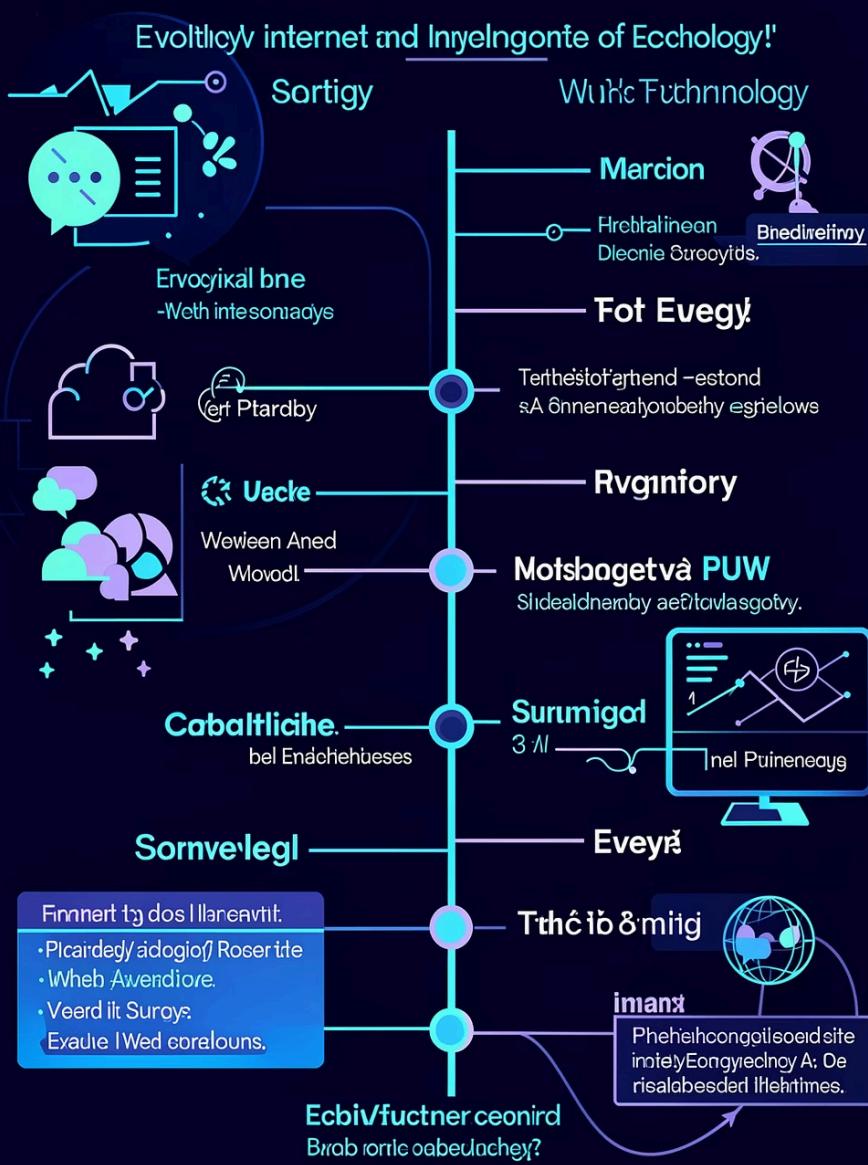
## Componentes de la relación

**HTML:** Define contenido y estructura.

**Navegador:** Procesa y visualiza.

**Usuario:** Interactúa con la interfaz.

# Tümeler Tie-leing!



# Evolución de HTML: Un viaje desde 1991

1

## **1991 - HTML Primera versión**

Tim Berners-Lee crea HTML para compartir documentos científicos en el CERN.

2

1997 - HTML 4.01

Establece estándares robustos y separa contenido de presentación mediante CSS.

3

2014 - HTML5 Hover

Introduce multimedia nativa, APIs modernas y soporte para aplicaciones web complejas.

- HTML5 representa el estándar actual con soporte para video, audio, canvas, almacenamiento local y nuevas etiquetas semánticas.

# ¿Para qué sirve HTML?



## Estructura del sitio

Define la jerarquía y organización del contenido mediante elementos semánticos como encabezados, párrafos y listas.



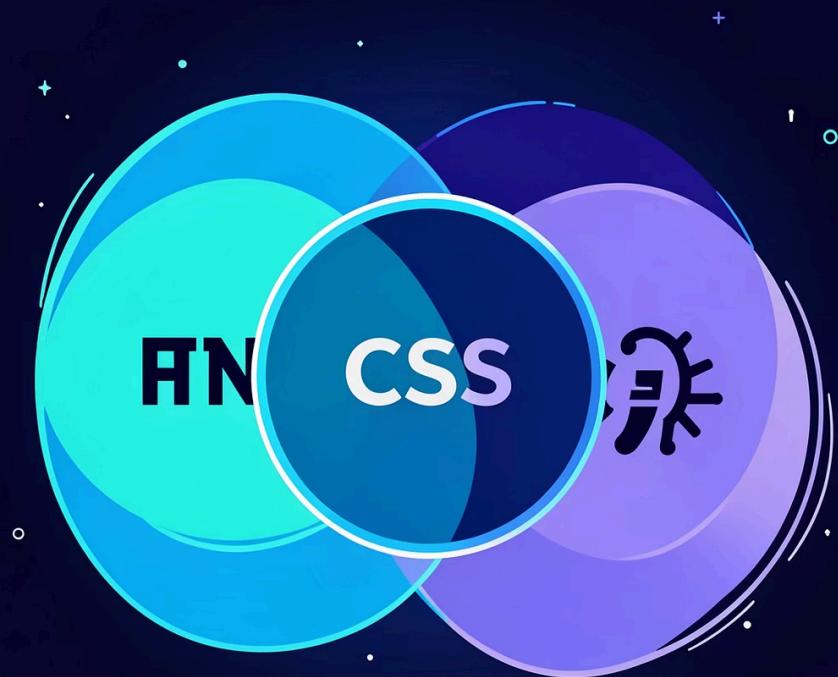
## Integración con CSS

Fundamenta la separación de contenido (HTML) y presentación (CSS), permitiendo diseños responsive y estilos reutilizables.



## Base para JavaScript

Proporciona el DOM que JavaScript manipula para crear interactividad dinámica y comportamiento funcional.



# Documentos HTML: Unidad y contenido



## <html>

Envuelve todo el documento



## <head>

Metadatos y recursos



## <body>

Contenido visible

## Buenas prácticas de estructura

- Siempre declara el tipo de documento
- Usa comentarios para documentar secciones
- Mantén consistencia en la indentación
- Valida tu HTML con herramientas como W3C Validator

## Elementos fundamentales del head

- <title>: Título de la página
- <meta>: Metadatos y encoding
- <link>: CSS y recursos externos
- <script>: Referencias JavaScript

# Etiquetas y atributos fundamentales



## Encabezados

`h1` hasta `h6` establecen jerarquía de contenido, con `h1` como título principal y `h6` como subsección menor.



## Párrafos

`<p>` define bloques de texto con espaciado automáticamente gestionado por el navegador.



## Enlaces

`<a href="#">` crea hipervínculos. El atributo `href` especifica la URL de destino.



## Imágenes

`<img src="" alt="">` incorpora imágenes. Los atributos `src` y `alt` son obligatorios para accesibilidad.



## Listas

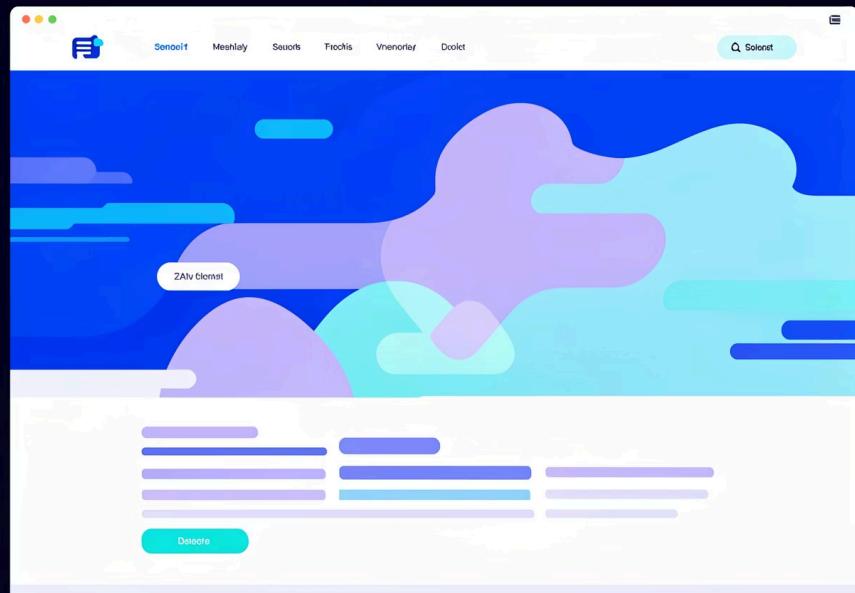
`<ul>` (no ordenadas) y `<ol>` (ordenadas) contienen elementos `<li>` individuales.



## Contenedores

`<div>` (bloque) y `<span>` (en línea) agrupan elementos para estilización y manipulación.

# Ejemplo práctico: Mi primera página



```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <title>Mi Página Web</title>
</head>
<body>
  <h1>Bienvenido</h1>
  <p>Este es mi primer documento HTML.</p>
  
  <ul>
    <li>Elemento 1</li>
    <li>Elemento 2</li>
  </ul>
</body>
</html>
```

01

**<!DOCTYPE html>**

Declara el tipo de documento como HTML5

03

**<head>**

Sección de metadatos y configuración

02

**<html>**

Elemento raíz con atributo de idioma

04

**<body>**

Contenido principal visible para usuarios



# Buenas prácticas y recomendaciones



## Cierre correcto de etiquetas

Las etiquetas deben abrirse y cerrarse correctamente con </>, o ser auto-cerradas como <img/> y <br/>.



## Indentación consistente

Usa 2 o 4 espacios para cada nivel de anidamiento. Facilita lectura y mantenimiento del código.



## Semántica apropiada

Elige etiquetas según su significado: <h1> para títulos principales, <p> para párrafos, <nav> para navegación.



## Atributos id y class

id identifica elementos únicos; class agrupa elementos para CSS y JavaScript. Usa nombres descriptivos y claros.

## Erros comunes a evitar

- Omitir etiquetas de cierre
- Anidar etiquetas incorrectamente
- Ignorar la accesibilidad
- No validar el código

**Consejo:** Usa el validador W3C para verificar que tu HTML cumpla las normas.

# Tu viaje con HTML comienza aquí

## Practica con ejemplos

Crea estructuras básicas y experimenta con diferentes etiquetas

## Únete a la comunidad

Miles de tutoriales, plataformas de código y foros de discusión

## Expándete gradualmente

Incorpora CSS para diseño y JavaScript para interactividad

**HTML es la puerta de entrada.** Aprende estas bases con dedicación y los conceptos avanzados surgirán naturalmente.

