

Construyendo la Web

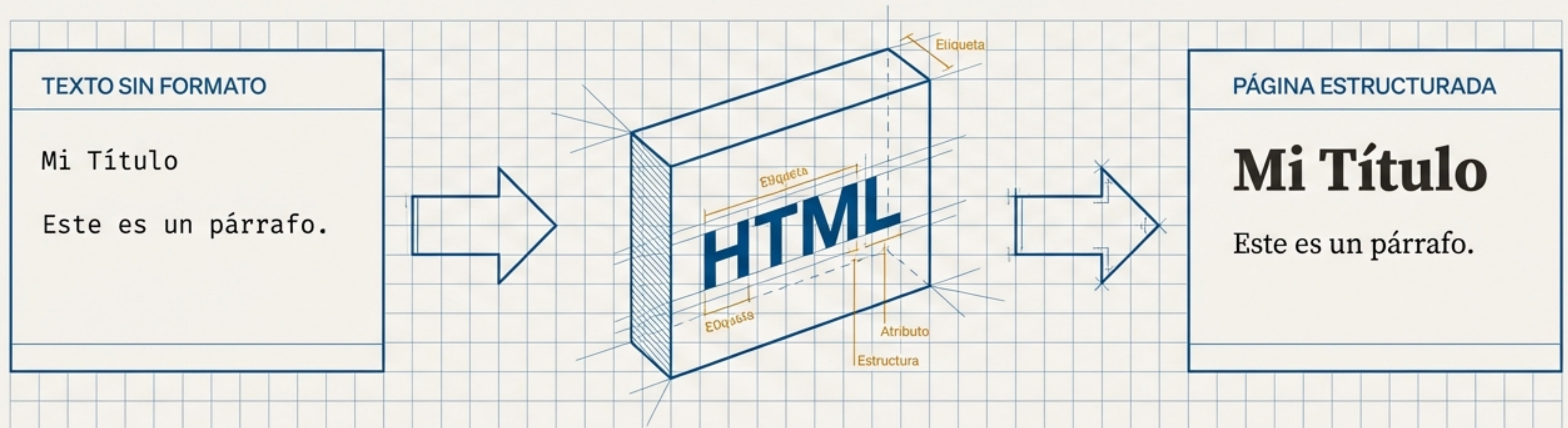
Una introducción a la estructura fundamental de HTML.



PRGJECT:	WEBPAGE STRUCTURE
DRAWING NO:	A-001
SCALE:	1:100
SCALE:	

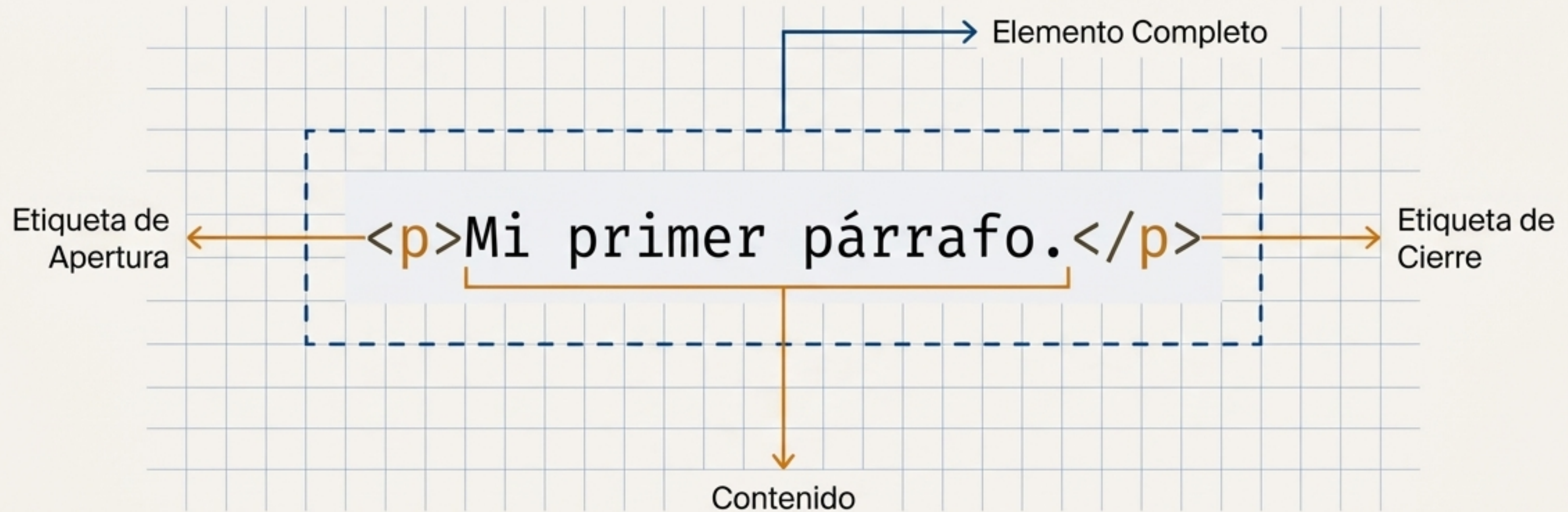
HTML no es un lenguaje de programación, es el plano de la web.

HTML son las siglas de *Hypertext Markup Language*. Su función no es crear algoritmos complejos, sino dar estructura y significado al contenido. Le dice al navegador qué es un título, qué es un párrafo o dónde va una imagen. Es el armazón sobre el que se construye toda página que visitas.



El ladrillo fundamental: Anatomía de un elemento HTML.

Toda la estructura de HTML se basa en 'elementos'. Un elemento 'marca' o 'envuelve' una porción de contenido para darle un significado específico. La mayoría de los elementos tienen una etiqueta de apertura y una de cierre.



De los ladrillos a la pared: Elementos en acción.

Así es como diferentes elementos estructuran el contenido. `<h1>` define el encabezado más importante, y `<p>` define un párrafo de texto.

CÓDIGO

```
```html
<h1>Mi primer encabezado</h1>
<p>Mi primer párrafo de texto. Sirve para
agrupar oraciones relacionadas.</p>
```
```



RESULTADO EN EL NAVEGADOR

Mi primer encabezado

Mi primer párrafo de texto. Sirve para agrupar oraciones relacionadas.

Una regla de estilo importante: Minúsculas para la coherencia.

Las etiquetas HTML no distinguen entre mayúsculas y minúsculas. <p>, <P>, y <p> funcionan igual. Sin embargo, la convención y buena práctica es escribir siempre las etiquetas en minúsculas. Esto mejora la legibilidad y la coherencia del código.



Correcto

```
```html
<title>Mi Página</title>
<h1>Encabezado</h1>
```
```



Incorrecto

```
```html
<TITLE>Mi Página</TITLE>
<H1>Encabezado</H1>
```
```


Componentes especializados:

Los elementos vacíos.

No todos los elementos envuelven contenido. Algunos, llamados 'elementos vacíos', consisten en una única etiqueta que se utiliza para insertar algo en el documento.

Para insertar un salto de línea:

```
<p>Mi perro es un gruñón.<br>
  Tiene muy malas pulgas.</p>
```

Inserta un salto de línea aquí.
No necesita cierre.

Para insertar una imagen (ejemplo conceptual):

```

```

Elemento para incrustar imágenes.
También es un elemento vacío.

Las especificaciones del plano: Añadiendo atributos a los elementos.

Los atributos son información adicional que se añade a la etiqueta de apertura de un elemento para modificar su comportamiento o proporcionar metadatos. Son como las notas en un plano que especifican el material o el color de un componente.

Ejemplo 1: Identificador de estilo

```
<p class="parrafo-destacado">Mi perro es un gruñón.</p>
```

El atributo `class` asigna un identificador para aplicar estilos (CSS).

Ejemplo 2: Destino de un hipervínculo

```
<a href="https://www.wikipedia.org">Wikipedia</a>
```

El atributo `href` en la etiqueta `<a>` especifica la URL de destino del enlace.

La sintaxis correcta de un atributo es clave.

Neue Haas Grotesk Display Pro

Para que los atributos funcionen correctamente, siempre deben seguir esta estructura:

```
<a href="https://www.wikipedia.org">
```

1

Un espacio entre el nombre del elemento y el atributo.

2

El nombre del atributo.

3

Un signo de igual.

4

El valor del atributo, envuelto entre comillas.

Nota: Si un elemento tiene múltiples atributos, estos también deben estar separados por espacios.

La regla de oro: Usa siempre comillas para los valores de los atributos.

Aunque en algunas situaciones muy simples el navegador puede interpretar atributos sin comillas, es una mala práctica que genera errores. Añadir un segundo atributo sin comillas romperá el código.

Código Incorrecto y Roto:



```
<!-- El navegador no puede interpretar dónde termina el valor de 'href' y empieza 'title' -->  
<a href=https://www.wikipedia.org/ title=La página de inicio de Wikipedia>Wikipedia</a>
```

Código Correcto:



```
<a href="https://www.wikipedia.org/" title="La página de inicio de Wikipedia">Wikipedia</a>
```

Usar comillas siempre evitará problemas y hará tu código más legible.

¿Comillas dobles o simples?

Tú eliges, pero sé consistente.

Puedes usar comillas simples (‘ ’) o dobles (“ ”) para los valores de los atributos. Es una cuestión de estilo. Lo importante es no mezclarlas en el mismo atributo.

Ambos son válidos:

```
<a href="http://www.ejemplo.com">Enlace de ejemplo.</a>  
<a href='http://www.ejemplo.com'>Enlace de ejemplo.</a>
```

Esto es incorrecto:

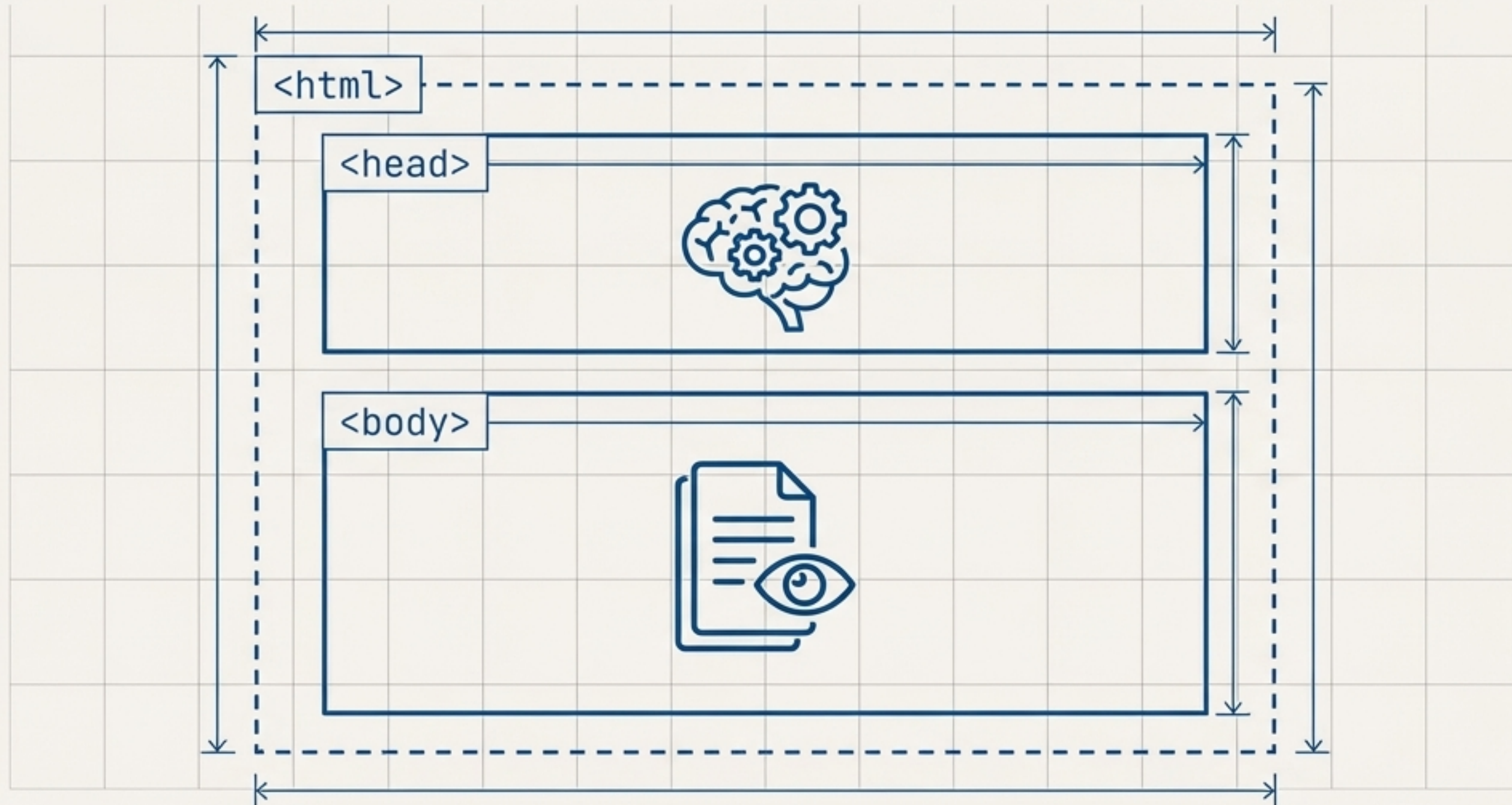
```
<a href="http://www.ejemplo.com'">Enlace de ejemplo.</a>
```

Para anidar comillas, usa el tipo opuesto o entidades HTML:

```
<!-- Correcto: comillas simples dentro de dobles -->  
<a href="http://ejemplo.com" title="¿A que es 'divertido'?">...</a>  
  
<!-- Correcto: usando una entidad HTML para la comilla -->  
<a href='http://ejemplo.com' title='¿A que es &apos;divertido&apos;?'>...</a>
```


El Plano Maestro: La estructura de todo documento HTML.

Ahora que conocemos los componentes básicos, vamos a ensamblarlos. Toda página HTML sigue una estructura estándar que actúa como el esqueleto principal.



La primera línea del plano: La declaración `<!DOCTYPE html>`.

Todo documento HTML moderno debe comenzar con esta declaración. Antiguamente era una línea larga y compleja que apuntaba a un conjunto de reglas. Hoy, es simplemente una señal para el navegador.

```
< !DOCTYPE html>
```

Esto le dice al navegador: “**Esta página está escrita en la última versión de HTML (HTML5)**”. Es un legado histórico que hay que incluir para que todo funcione correctamente.

El contenedor principal (<html>) y el centro de control (<head>).

Justo después del `DOCTYPE`, todo el contenido de la página se envuelve en el elemento `<html>`. Dentro de él, el primer elemento es `<head>`, que contiene toda la información de 'trastienda' que no es visible para el usuario.

Neue Haas Grotesk Display Pro
Source Serif 4

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <!-- Aquí va información importante para el navegador y los
    buscadores, pero no visible en la página. -->
  </head>

  <!-- El contenido visible irá aquí más tarde -->
</html>
```

El atributo `lang="es"` en `<html>` es importante para la accesibilidad y los motores de búsqueda, ya que declara el idioma principal de la página.
Source Serif 4, Annotation Ochre

Dentro del <head>: Los metadatos esenciales

El <head> es el hogar de los elementos <meta>, que proporcionan metadatos (datos sobre los datos) al navegador y a los motores de búsqueda.

Codificación de caracteres (esencial):

```
```html
<meta charset="utf-8">
```
```

----- Annotation Ochre (#D97904)

****Explicación**:** Asegura que caracteres como `á`, `ñ` o `€` se muestren correctamente en todos los idiomas.

Información para buscadores:

```
```html
<meta name="description" content="Una
descripción concisa de mi página.">
<meta name="keywords" content="HTML, CSS,
JavaScript">
<meta name="author" content="Fernando Ruiz
Rico">
```
```

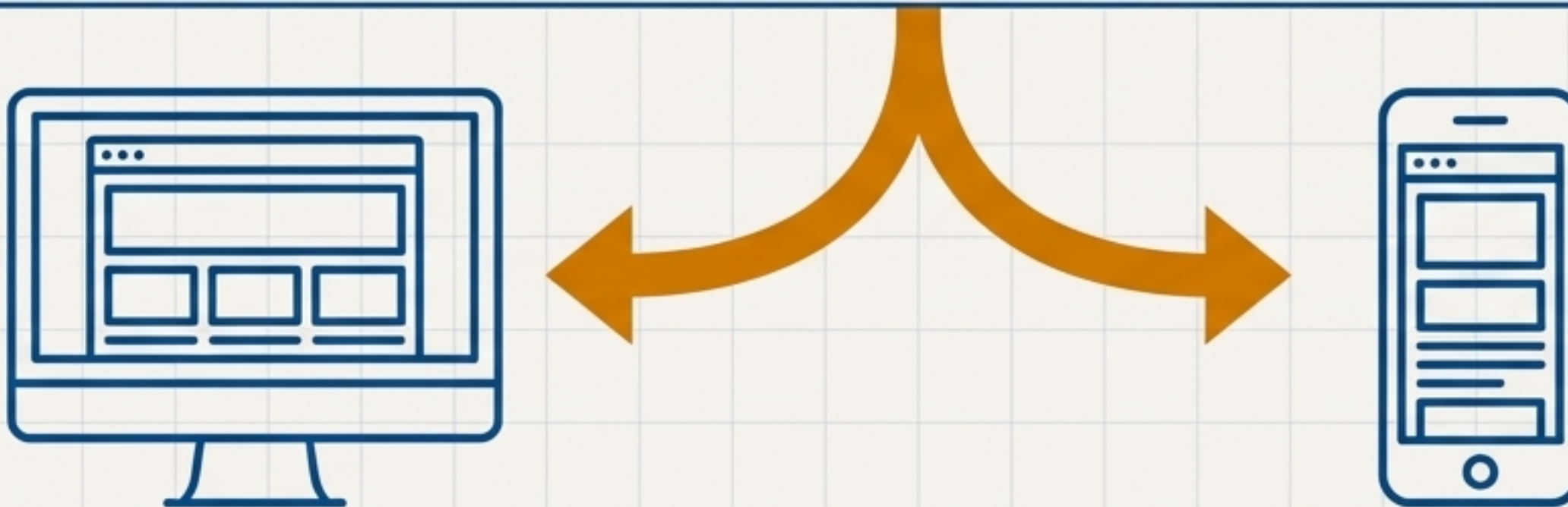
----- Annotation Ochre (#D97904)

****Explicación**:** Ayuda a Google y otros buscadores a entender de qué trata tu página.

Asegurando la vista en cualquier dispositivo: El `viewport`.

Para que una página web se vea bien tanto en un monitor de escritorio como en un teléfono móvil, necesitamos decirle al navegador cómo controlar el área visible de la página (el 'viewport').

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

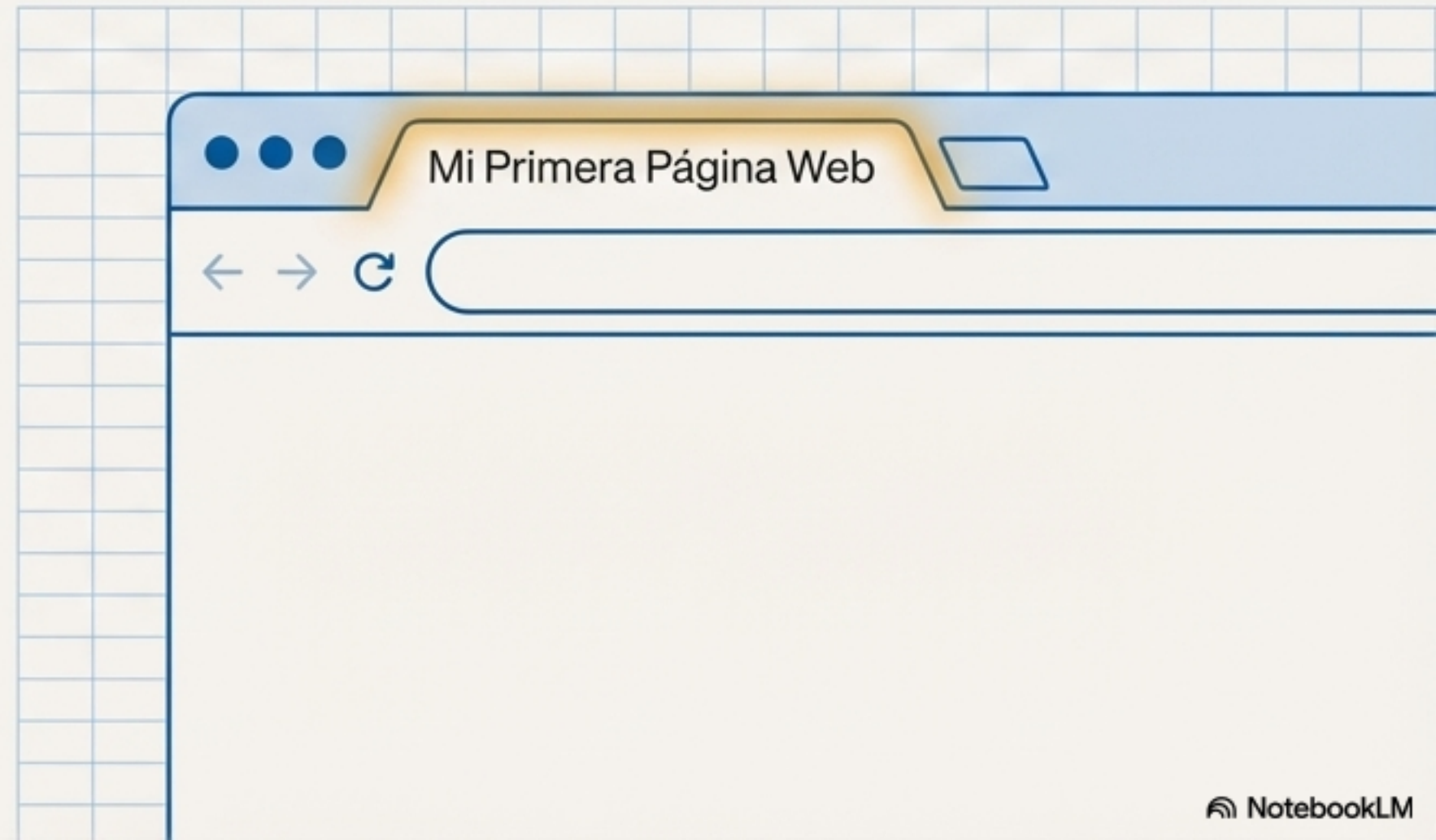


Nota: Esta línea es un estándar en el desarrollo web moderno para crear sitios web adaptables (*responsive*).

Dándole un nombre a tu página: El elemento <title>.

También dentro del <head>, el elemento <title> define el texto que aparece en la pestaña del navegador. También es el nombre que se utiliza cuando un usuario guarda tu página en sus favoritos.

```
<head>  
  <title>Mi Primera Página Web</title>  
</head>
```



Conectando recursos externos: <link> y <script>.

El <head> también es el lugar para enlazar archivos externos que añaden estilo y funcionalidad a tu página.

Enlazar una hoja de estilos (CSS):

```
<!-- El atributo 'rel' especifica la relación, 'href' la ubicación del archivo. -->  
<link rel="stylesheet" href="estilos.css">
```

Incluir un archivo de JavaScript:

```
<!-- El atributo 'src' especifica la ubicación del archivo de script. -->  
<script src="javascript.js"></script>
```

Nota: Estos elementos son la puerta de entrada para dar **diseño** (CSS) e **interactividad** (JavaScript) a tu estructura HTML.

La zona habitable: El elemento <body>.

Si el <head> es la sala de máquinas, el <body> es todo el espacio visible del edificio. Contiene todo lo que el usuario final ve y con lo que interactúa: texto, imágenes, vídeos, etc.

```
<!doctype html>
<html lang="es">
  <head>
    <meta charset="utf-8">
    <title>Página de ejemplo</title>
  </head>

  <body>
    <!-- ¡Todo el contenido visible va aquí! -->
    ¡Hola mundo!
  </body>
</html>
```



Cuidado con los caracteres especiales.

No puedes escribir los caracteres `<` o `>` directamente en tu texto, porque el navegador los interpretará como parte de una etiqueta HTML. Para mostrarlos, debes usar 'entidades HTML'.

Carácter a mostrar	Entidad HTML
<	<
>	>
&	&
"	"
©	©
€	€

Ejemplo: Para mostrar <p>, escribirías <p>.

Amueblando la página: Encabezados y párrafos.

Para que el contenido sea legible, es crucial estructurarlo. HTML proporciona encabezados de seis niveles (`<h1>` a `<h6>`) para los títulos y el elemento `<p>` para los párrafos.

Código

```
<h1>Encabezado de Nivel 1</h1>  
<p>Este es un párrafo bajo el encabezado principal. Introduce el tema.</p>
```

```
<h2>Encabezado de Nivel 2</h2>  
<p>Este es un párrafo que detalla una subsección del tema.</p>
```

Resultado Renderizado

Encabezado de Nivel 1

Este es un párrafo bajo el encabezado principal.
Introduce el tema.

Encabezado de Nivel 2

Este es un párrafo que detalla una subsección del tema.

Creando saltos de línea con `
`.

El elemento `
` (line break) fuerza un salto de línea. Es útil para contenido donde la división de líneas es importante, como en poemas o direcciones postales.

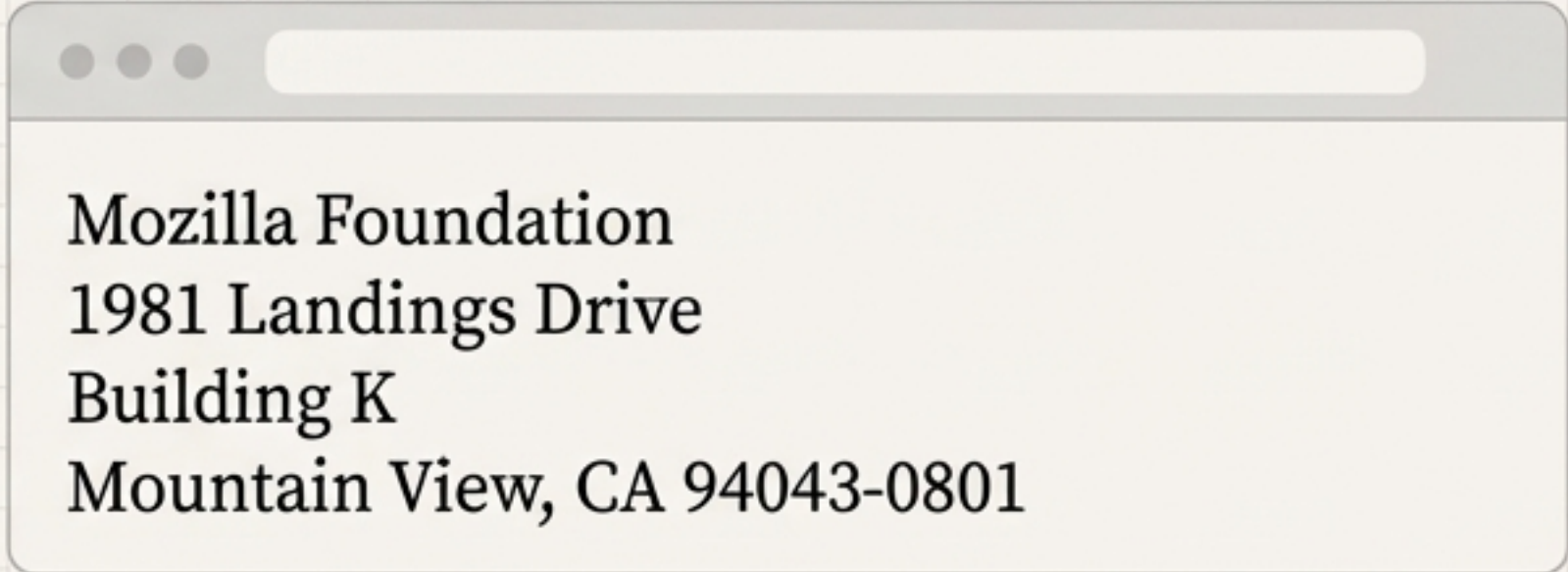
Importante:

**No uses `
` para crear espacio vertical entre elementos.** Para eso, se debe usar CSS (con la propiedad `margin`).

Ejemplo de Uso Correcto (Dirección)

```
<p>
  Mozilla Foundation<br>
  1981 Landings Drive<br>
  Building K<br>
  Mountain View, CA 94043-0801
</p>
```

Resultado



Mozilla Foundation
1981 Landings Drive
Building K
Mountain View, CA 94043-0801

Manteniendo el formato: El texto preformateado con <pre>.

El elemento <pre> le dice al navegador que muestre el texto exactamente como está escrito en el editor, conservando todos los espacios, sangrías y saltos de línea. Generalmente se muestra con una fuente monoespaciada.

Código

```
<pre>
-----
< Soy un experto en mi campo. >
-----
      ^  ^
      --
      (oo)\_____
      (__)\         )\/\
          ||-----w |
          ||         ||

</pre>
```

Resultado Renderizado

```
-----
< Soy un experto en mi campo. >
-----
      ^  ^
      --
      (oo)\_____
      (__)\         )\/\
          ||-----w |
          ||         ||
```


La anatomía completa de una página HTML.

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <!-- METADATOS PARA EL NAVEGADOR Y BUSCADORES -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta name="description" content="Página de ejemplo de la estructura HTML.">
    <title>Anatomía de una Página</title>
    <!-- ENLACES A RECURSOS EXTERNOS -->
    <link rel="stylesheet" href="estilos.css">
  </head>
  <body>
    <!-- CONTENIDO VISIBLE PARA EL USUARIO -->
    <h1>Título Principal de la Página</h1>
    <p>Un párrafo de introducción con un <a href="#">enlace</a>.</p>
    <h2>Un Subtítulo</h2>
    <p>Otro párrafo con más detalles.</p>
  </body>
</html>
```




Taller del Arquitecto: Tu primer plano.

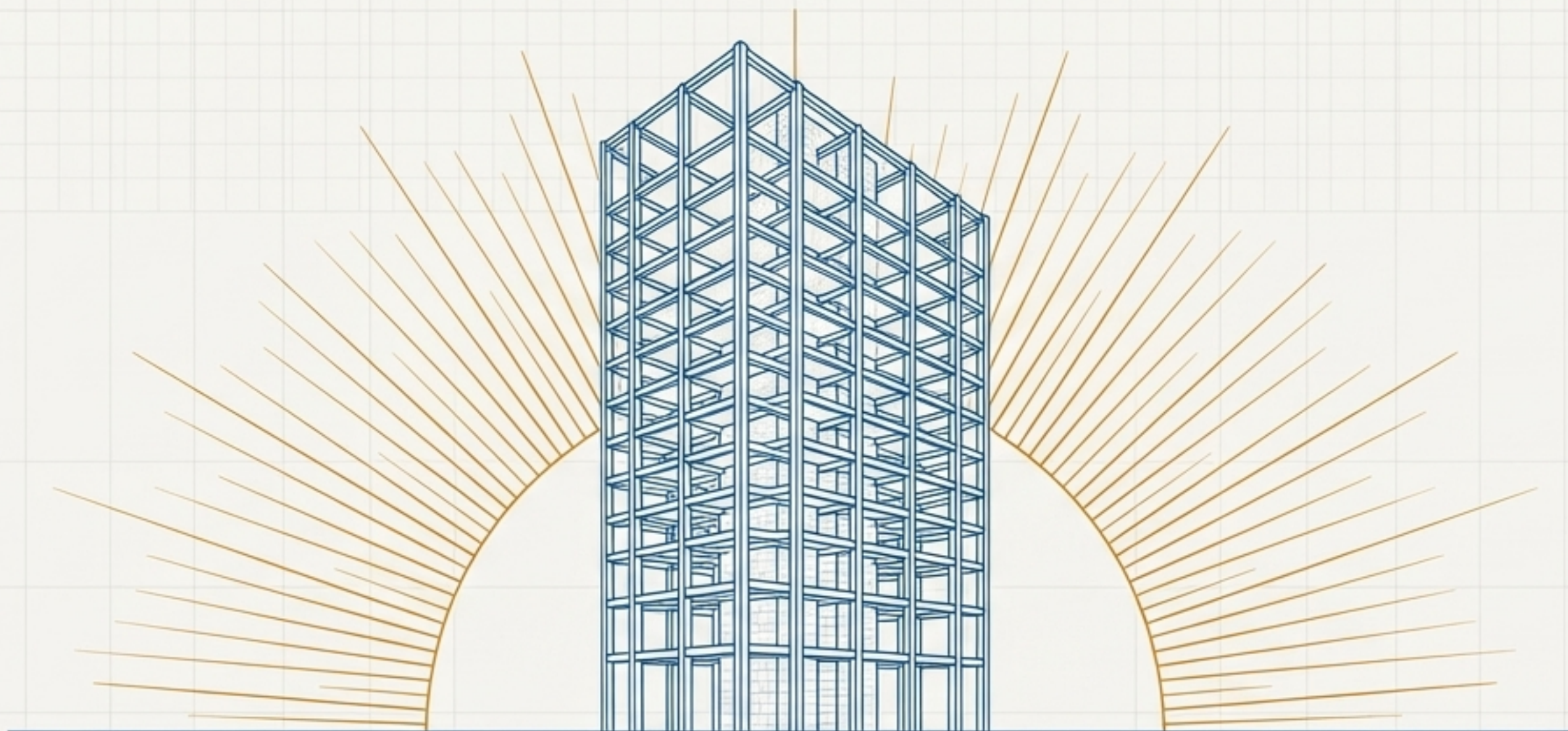
Es hora de construir. Utilizando un editor de texto simple (como Visual Studio Code, Notepad, etc.), crea un archivo `index.html` y reproduce la siguiente estructura. Rellena los detalles con tu propia información.

Desafío de Código

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta name="author" content=" [Tu Nombre Aquí] ">
    <meta name="description" content=" [Descripción de tu página] ">
    <title>Saludos</title>
  </head>
  <body>
    <h1>¡Hola mundo, mi nombre es [Tu Nombre Aquí] !</h1>

    <h2>Un Poema</h2>
    <p>
      Del salón en el ángulo oscuro,<br>
      de su dueña tal vez olvidada,<br>
      silenciosa y cubierta de polvo,<br>
      veíase el arpa.
    </p>
  </body>
</html>
```

Guarda el archivo y ábrelo en tu navegador para ver tu primera página web.



Has colocado la primera piedra.

Entender la estructura HTML es el cimiento indispensable sobre el que se construye toda la web. Cada etiqueta y atributo que has aprendido es una herramienta en tu cinturón de desarrollador. A partir de aquí, el siguiente paso es dar estilo y vida a esta estructura con CSS y añadir interactividad con JavaScript.