

Módulo 5.2: HTML, CSS, JS

MSc. Carlos Cedeño

1. HTML: La Estructura del Contenido Web

HTML, que significa **HyperText Markup Language**, no es un lenguaje de programación. Es un lenguaje de marcado que usamos para estructurar el contenido de una página web. Piensen en HTML como el esqueleto de un cuerpo; define qué elementos existen y cómo se organizan.

- **¿Qué hace?** Define el contenido y su estructura. Por ejemplo, un título, un párrafo, una imagen, una lista, etc.
- **Conceptos clave:**
 - **Elementos:** Son los bloques de construcción de HTML. La mayoría de los elementos tienen una etiqueta de apertura y una de cierre, como `<p>` para un párrafo y `</p>` para cerrarlo.
 - **Etiquetas:** Son las palabras clave encerradas en corchetes angulares (`<>`). Por ejemplo, `<h1>` para un título principal, `<p>` para un párrafo, `` para una imagen.
 - **Atributos:** Proporcionan información adicional sobre un elemento. Se colocan dentro de la etiqueta de apertura. Por ejemplo, `` tiene un atributo `href` que especifica la URL del enlace.

Estructura básica de un documento HTML

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Mi Primera Página Web</title>
</head>
<body>
  <h1>¡Hola Mundo!</h1>
  <p>Este es mi primer párrafo.</p>
</body>
</html>
```

2. CSS: El Estilo y Diseño

CSS, que significa **Cascading Style Sheets**, es el lenguaje que usamos para darle estilo a nuestro contenido HTML. Define cómo se ve la página.

- **¿Qué hace?** Controla la presentación de los documentos web, incluyendo colores, fuentes, espaciado, diseño, etc.
- **Conceptos clave:**
 - **Selectores:** Indican a qué elementos HTML se aplicarán los estilos. Pueden seleccionar elementos por su tipo (por ejemplo, `p`), por su clase (`.mi-clase`), o por su ID (`#mi-id`).
 - **Propiedades y valores:** Las propiedades son las características que queremos cambiar (por ejemplo, `color` , `font-size`), y los valores son lo que asignamos a esas propiedades (por ejemplo, `blue` , `16px`).

- **Sintaxis básica:**

```
selector {  
  propiedad: valor;  
  propiedad: valor;  
}
```

- **Formas de incluir CSS:**

- i. **En línea:** Directamente en la etiqueta HTML (no recomendado para grandes proyectos).

```
<p style="color: blue;">Texto azul</p>
```

- ii. **Interno:** Dentro de la sección `<head>` del HTML, usando la etiqueta `<style>` .

```
<head>
  <style>
    h1 {
      color: red;
    }
  </style>
</head>
```

1. Externo (más común y recomendado): En un archivo `.css` separado, enlazado desde el HTML.

```
<head>  
  <link rel="stylesheet" href="style.css">  
</head>
```


3. JavaScript: La Interactividad

JavaScript, a menudo abreviado como JS, es un **lenguaje de programación** que permite añadir interactividad y comportamiento dinámico a las páginas web. Si HTML es la estructura y CSS es la apariencia, JavaScript es la inteligencia y la acción.

- **¿Qué hace?** Permite que una página web haga cosas más allá de simplemente mostrar contenido estático. Por ejemplo, animaciones, validación de formularios, manipulación del DOM (Document Object Model), comunicación con servidores, etc.
- **Conceptos clave:**
 - **Variables:** Para almacenar datos.
 - **Funciones:** Bloques de código que se pueden ejecutar cuando se les llama.
 - **Eventos:** Acciones que ocurren en el navegador (clics, pasar el ratón por encima, cargar la página). JavaScript puede "escuchar" estos eventos y reaccionar a ellos.
 - **Manipulación del DOM:** JavaScript puede cambiar el contenido, los atributos y los estilos de los elementos HTML en tiempo real.

- Sintaxis básica:

```
// Esto es un comentario
let nombre = "Mundo"; // Declaración de una variable
function saludar() {
    alert("¡Hola, " + nombre + "!"); // Función que muestra un mensaje
}
```

- **Formas de incluir JavaScript:**

- i. **En línea:** Directamente en el atributo de un elemento HTML (no muy recomendado).

```
<button onclick="alert('¡Clicaste!')">Haz clic</button>
```

- ii. **Interno:** Dentro de la sección `<body>` o `<head>` del HTML, usando la etiqueta `<script>`. Se recomienda al final del `<body>` para asegurar que el HTML se cargue primero.

```
<body>
  <script>
    console.log("JavaScript interno");
  </script>
</body>
```

- **Externo (más común y recomendado):** En un archivo `.js` separado, enlazado desde el HTML.

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

Cómo Trabajan Juntos

La magia del desarrollo web ocurre cuando HTML, CSS y JavaScript colaboran:

- **HTML** proporciona la **estructura** y el **contenido**.
- **CSS** se encarga de la **presentación** y el **estilo**.
- **JavaScript** añade la **interactividad** y el **comportamiento dinámico**.

Recursos Adicionales

- **MDN Web Docs:** La mejor fuente de documentación sobre tecnologías web (HTML, CSS, JavaScript).
 - [HTML](#)
 - [CSS](#)
 - [JavaScript](#)
- **FreeCodeCamp:** Tutoriales interactivos y proyectos para aprender a programar.
- **W3Schools:** Tutoriales sencillos y ejemplos.