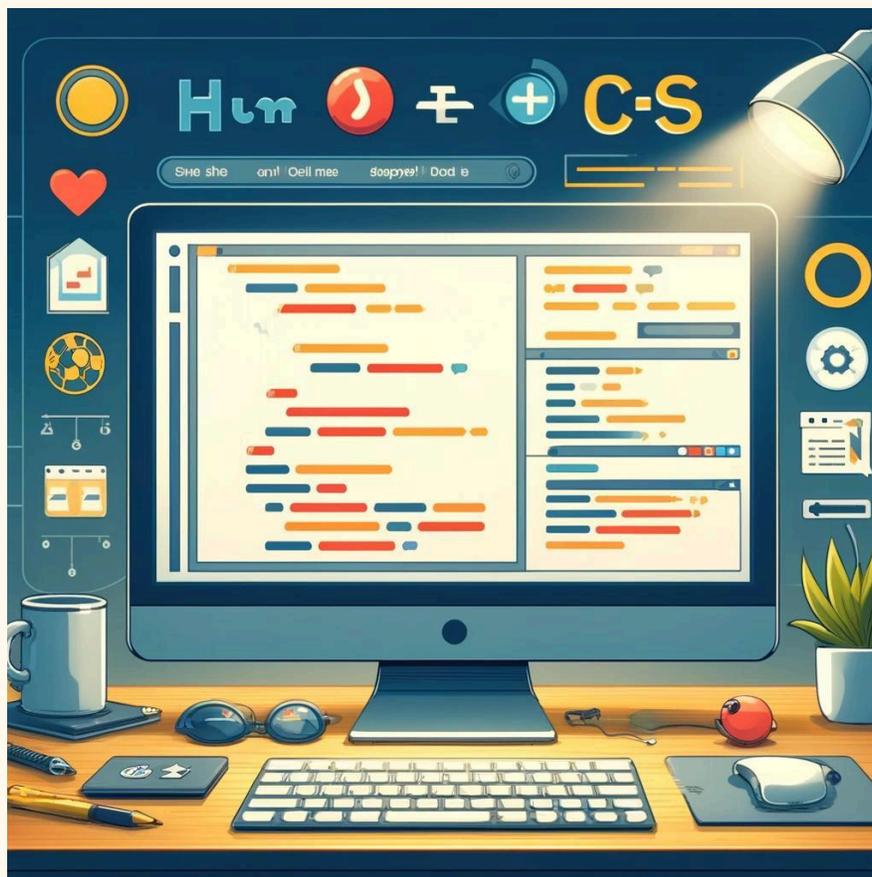


DESARROLLO WEB BÁSICO:

Cómo Construir una Página Web Usando HTML, CSS y JavaScript



INTRODUCCIÓN

El desarrollo web es una habilidad esencial en el mundo digital de hoy. Este tutorial te guiará a través de los fundamentos para construir una página web básica utilizando HTML, CSS y JavaScript. Al final de este tutorial, tendrás una comprensión clara de cómo crear y estilizar páginas web interactivas.

CAPÍTULO 1: FUNDAMENTOS DE HTML

Introducción

HTML (HyperText Markup Language) es el lenguaje estándar para crear y estructurar el contenido de una página web. Es el esqueleto de cualquier sitio web, proporcionando la estructura básica sobre la cual se pueden aplicar estilos y funcionalidades adicionales. Este capítulo cubrirá los conceptos esenciales de HTML, incluyendo la estructura básica de un documento HTML, las etiquetas más comunes y cómo crear enlaces e imágenes.

Estructura Básica de un Documento HTML

Un documento HTML se compone de una serie de elementos organizados de manera jerárquica. Cada elemento se define mediante etiquetas HTML, que se escriben entre corchetes angulares (< y >).

Estructura de un Documento HTML Típico:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Mi Primera Página Web</title>
  </head>
  <body>
    <h1>Bienvenido a Mi Página Web</h1>
    <p>Este es un párrafo de ejemplo.</p>
  </body>
</html>
```

- ★ <!DOCTYPE html>: Declara el tipo de documento y la versión de HTML.
- ★ <html>: Elemento raíz que contiene todo el contenido de la página.
- ★ <head>: Contiene metadatos como el título de la página y enlaces a estilos y scripts.
- ★ <title>: Define el título de la página que aparece en la pestaña del navegador.
- ★ <body>: Contiene todo el contenido visible de la página web.

Etiquetas HTML Comunes

Las etiquetas HTML son los componentes básicos de una página web. Aquí se presentan algunas de las etiquetas más utilizadas:

- Encabezados (<h1> a <h6>): Utilizados para definir títulos y subtítulos.

```
<h1>Título Principal</h1>
```

```
<h2>Subtítulo</h2>
```

- Párrafo (<p>): Define un párrafo de texto.

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

- Enlace (): Crea un hipervínculo a otra página o recurso.

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

- Imagen (): Inserta una imagen en la página.

```

```

- Listas:

- Lista no ordenada (y):

```
<ul>
```

```
<li>Elemento 1</li>
```

```
<li>Elemento 2</li>
```

```
<li>Elemento 3</li>
```

```
</ul>
```

- Lista ordenada (y):

```
<ol>
```

```
<li>Elemento 1</li>
```

```
<li>Elemento 2</li>
```

```
<li>Elemento 3</li>
```

```
</ol>
```

- División (<div>): Utilizada como contenedor para agrupar otros elementos.

```
<div>
```

```
<h2>Sección del sitio web</h2>
```

```
<p>Contenido de la sección.</p>
```

```
</div>
```

- Span (): Similar a <div>, pero se utiliza para agrupar contenido en línea.

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

Atributos de Etiquetas HTML

Las etiquetas HTML pueden tener atributos que proporcionan información adicional sobre el elemento. Los atributos se escriben dentro de la etiqueta de apertura.

Atributos Comunes:

- id: Identificador único para el elemento.

```
<p id="intro">Este es un párrafo con un ID único.</p>
```

- class: Clase que se puede compartir con otros elementos para aplicar estilos CSS

```
<p class="texto-destacado">Este es un texto destacado.</p>
```

- style: Define estilos en línea para un elemento.

```
<p style="color: blue; font-size: 14px;">Texto con estilo en línea.</p>
```

- href: Define la URL para un enlace.

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

- src y alt: Utilizados en imágenes para definir la fuente y el texto alternativo.

```

```

Formularios en HTML

Los formularios permiten a los usuarios enviar datos a un servidor. Un formulario HTML se define con la etiqueta `<form>` y puede contener varios tipos de campos de entrada.

Ejemplo de Formulario:

```
<form action="/submit" method="post">  
  
  <label for="nombre">Nombre:</label>  
  
  <input type="text" id="nombre" name="nombre">  
  
  
  <label for="email">Correo Electrónico:</label>  
  
  <input type="email" id="email" name="email">  
  
  
  <input type="submit" value="Enviar">  
  
</form>
```

- `<form>`: Define el formulario.
- `<label>`: Define una etiqueta para un campo de entrada.
- `<input>`: Define un campo de entrada (varios tipos disponibles).
- `<submit>`: Botón para enviar el formulario.

Enlaces y Navegación

Los enlaces (<a>) son esenciales para la navegación en la web, permitiendo a los usuarios moverse entre diferentes páginas o secciones.

Creación de Enlaces:

- Enlace Interno: Ir a Sección 1
- Enlace Externo: Visitar Google
- Enlace de Descarga: Descargar PDF

Buenas Prácticas de HTML

- Usar etiquetas semánticas: Como <header>, <footer>, <article>, y <section>.
- Validar el código HTML: Utilizar herramientas como el validador de W3C para asegurar que el código cumple con los estándares web.
- Mantener una estructura clara: Organizar el código en secciones y utilizar comentarios para describir partes del código.

Ejemplo de Etiquetas Semánticas:

```
<!DOCTYPE html>
<html>
<head>
  <title>Página Semántica</title>
</head>
<body>
  <header>
    <h1>Encabezado Principal</h1>
  </header>
  <nav>
    <ul>
      <li><a href="#inicio">Inicio</a></li>
      <li><a href="#sobre">Sobre Nosotros</a></li>
      <li><a href="#contacto">Contacto</a></li>
    </ul>
  </nav>
  <main>
    <article>
      <h2>Artículo Principal</h2>
      <p>Contenido del artículo.</p>
    </article>
  </main>
  <footer>
    <p>Pie de página</p>
  </footer>
</body>
</html>
```

CAPÍTULO 2: FUNDAMENTOS DE CSS

Introducción

CSS (Cascading Style Sheets) es el lenguaje utilizado para describir la presentación de documentos HTML. CSS permite separar el contenido de la presentación, proporcionando una mayor flexibilidad y control en el diseño de las páginas web. En este capítulo, exploraremos la sintaxis de CSS, los selectores, las propiedades de estilo, y cómo aplicar estilos a los elementos HTML para crear diseños atractivos y profesionales.



Sintaxis de CSS

La sintaxis de CSS está compuesta por selectores y declaraciones. Cada declaración incluye una propiedad y un valor, separados por dos puntos (:) y finalizados con un punto y coma (;).

Ejemplo de sintaxis de CSS

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

Ejemplo Aplicado

```
h1 {  
  color: blue;  
  font-size: 24px;  
}
```

- Selector (h1): Apunta a los elementos HTML que se van a estilizar.
- Propiedad (color): Indica la característica específica que se va a estilizar.
- Valor (blue): Define la apariencia de la propiedad.

Selectores CSS

Los selectores se utilizan para apuntar a los elementos HTML que se desean estilizar. Existen varios tipos de selectores en CSS.

Selectores comunes

- Selector de Elemento: Apunta a todos los elementos de un tipo específico.

```
p {  
  color: green;  
}
```

- Selector de Clase: Apunta a todos los elementos con una clase específica. Se denota con un punto (.) seguido del nombre de la clase.

```
.clase-ejemplo {  
  font-size: 18px;  
}
```

- Selector de ID: Apunta a un elemento con un ID específico. Se denota con un signo de almohadilla (#) seguido del nombre del ID.

```
#id-ejemplo {  
  background-color: yellow;  
}
```

- Selector Universal (*): Apunta a todos los elementos.

```
* {  
  margin: 0;  
  padding: 0;  
}
```

- Selector de Atributo: Apunta a elementos con un atributo específico.

```
input[type="text"] {  
  border: 1px solid #ccc;  
}
```

Propiedades de Estilo CSS

CSS permite estilizar diversos aspectos de los elementos HTML. A continuación se describen algunas de las propiedades de estilo más utilizadas.

- Propiedades de Texto:
 - color: Define el color del texto. `p { color: #333;}`
 - font-size: Define el tamaño de la fuente. `h1 { font-size: 32px;}`
 - text-align: Alinea el texto. `h2 { text-align: center;}`
 - text-decoration: Agrega decoraciones al texto. `a { text-decoration: none;}`
 - font-family: Define la familia de la fuente. `body { font-family: Arial, sans-serif;}`
- Propiedades de Fondo:
 - background-color: Define el color de fondo. `div { background-color: #f0f0f0;}`
 - background-image: Define una imagen de fondo. `body {background-image: url('fondo.jpg');}`
 - background-size: Define el tamaño de la imagen de fondo. `header { background-size: cover;}`
- Propiedades de Bordes y Espaciado:
 - border: Define el borde de un elemento. `img { border: 2px solid #000;}`
 - margin: Define el espacio exterior alrededor de un elemento. `.contenedor { margin: 20px;}`
 - padding: Define el espacio interior dentro de un elemento. `.contenido { padding: 15px;}`

Modelos de Diseño en CSS

CSS ofrece varios modelos de diseño para organizar y alinear elementos en una página web.

Modelos de caja CSS:

Cada elemento en CSS se representa como una caja rectangular que contiene propiedades como margin, border, padding y content.

```
.caja {
  width: 200px;
  padding: 10px;
  border: 1px solid #ccc;
  margin: 20px;
}
```

Modelo Flexbox:

Flexbox es un modelo de diseño que permite crear layouts flexibles y eficientes.

```
.contenedor-flex {
  display: flex;
  justify-content: center;
  align-items: center;
}
```

Modelo Grid:

Grid es un modelo de diseño que proporciona una manera de crear layouts complejos con filas y columnas.

```
.contenedor-grid {
  display: grid;
  grid-template-columns: 1fr 1fr 1fr;
  gap: 10px;
}
```

Media Queries y Diseño Responsivo

Las Media Queries permiten aplicar estilos CSS específicos basados en el tamaño de la pantalla, haciendo que el diseño sea responsivo.

- Ejemplo de Media Query:

```
@media (max-width: 600px) {
  .contenedor {
    flex-direction: column;
  }
}
```

Integración de CSS con HTML

CSS se puede integrar con HTML de tres maneras: en línea, interna y externa.

- CSS en Línea: Se utiliza el atributo style directamente en el elemento HTML.

```
<p style="color: red;">Texto en rojo</p>
```

- CSS Interno: Se utiliza dentro de una etiqueta `<style>` en el `<head>` del documento HTML.

```
<head>
  <style>
    body {
      background-color: #e0e0e0;
    }
  </style>
</head>
```

- CSS Externo: Se utiliza un archivo CSS separado, enlazado con la etiqueta `<link>`.

```
<head>
  <link rel="stylesheet" type="text/css" href="estilos.css">
</head>
```

Ejemplos Prácticos

1. Estilizar una Página Web Básica:

a. index.html:

```
<!DOCTYPE html>
<html>
<head>
  <title>Página Estilizada</title>
  <link rel="stylesheet" type="text/css" href="estilos.css">
</head>
<body>
  <header>
    <h1>Mi Página Web</h1>
  </header>
  <nav>
    <ul>
      <li><a href="#inicio">Inicio</a></li>
      <li><a href="#sobre">Sobre Nosotros</a></li>
      <li><a href="#contacto">Contacto</a></li>
    </ul>
  </nav>
```

```
<main>
  <section id="inicio">
    <h2>Inicio</h2>
    <p>Bienvenido a mi sitio web.</p>
  </section>
  <section id="sobre">
    <h2>Sobre Nosotros</h2>
    <p>Información sobre nosotros.</p>
  </section>
</main>
<footer>
  <p>Contacto: contacto@ejemplo.com</p>
</footer>
</body>
</html>
```

b. estilos.css:

```
body {
  font-family: Arial, sans-serif;
  background-color: #f4f4f4;
  margin: 0;
  padding: 0;
}

header {
  background-color: #333;
  color: #fff;
  padding: 10px 0;
  text-align: center;
}

nav ul {
  list-style: none;
  padding: 0;
}

nav ul li {
  display: inline;
  margin-right: 10px;
}
```

```
nav ul li a {
  text-decoration: none;
  color: #333;
}

main {
  padding: 20px;
}

footer {
  background-color: #333;
  color: #fff;
  text-align: center;
  padding: 10px 0;
  position: absolute;
  width: 100%;
  bottom: 0;
}
```