

Curso de Programación Desde Cero

Tema 2: HTML y CSS BÁSICO

Duración estimada: 12 horas

1. Introducción al Desarrollo Web

El desarrollo web consiste en crear sitios y aplicaciones que funcionan en navegadores. Componentes principales:

- 2 HTML: Define la estructura y contenido
- 2 CSS: Controla la presentación y diseño
- 2 JavaScript: Añade interactividad

2. ¿Qué es HTML?

HyperText Markup Language es el lenguaje estándar para crear páginas web. Utiliza "etiquetas" para marcar diferentes tipos de contenido y darle significado semántico.

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</title>
</head>
<body>
  <h1>Bienvenido a mi sitio web</h1>
  <p>Este es mi primer párrafo en HTML.</p>
</body>
</html>
```

3. Etiquetas HTML Esenciales

Encabezados (Headings):

<h1>Título Principal</h1>	<!-- MÁS importante -->
<h2>Subtítulo</h2>	<!-- Segundo nivel -->
<h3>Sección</h3>	<!-- Tercer nivel -->
<h4>, <h5>, <h6>	<!-- Hasta 6 niveles -->

Párrafos y Texto:

<p>Un párrafo de texto normal.</p>
Texto importante
Texto enfatizado
 <!-- Salto de línea -->

4. Listas en HTML

Listas No Ordenadas (viæetas):

```
<ul>
  <li>Primer elemento</li>
  <li>Segundo elemento</li>
  <li>Tercer elemento</li>
</ul>
```

Listas Ordenadas (numeradas):

```
<ol>
  <li>Primer paso</li>
  <li>Segundo paso</li>
  <li>Tercer paso</li>
</ol>
```

5. Enlaces e ImÆgenes

Enlaces (Links):

```
<a href="https://www.google.com">Ir a Google</a>
<a href="pagina2.html">Enlace interno</a>
<a href="mailto:correo@ejemplo.com">Enviar email</a>
<a href="tel:+34123456789">Llamar</a>
```

ImÆgenes:

```


```

6. Formularios HTML

Los formularios permiten recopilar información del usuario:

```
<form action="/procesar" method="POST">
  <label for="nombre">Nombre:</label>
  <input type="text" id="nombre" name="nombre" required>
  <label for="email">Email:</label>
  <input type="email" id="email" name="email" required>
  <label for="edad">Edad:</label>
  <input type="number" id="edad" name="edad" min="18" max="100">
  <label for="mensaje">Mensaje:</label>
  <textarea id="mensaje" name="mensaje" rows="4"></textarea>
  <input type="submit" value="Enviar">
</form>
```

Tipos de Input Importantes:

- 2 text: Texto libre
- 2 email: Validación automÆtica de email
- 2 password: Oculta el texto
- 2 number: Solo nœmeros
- 2 date: Selector de fecha
- 2 checkbox: Casillas de selección
- 2 radio: Opción œnica

7. Introducción a CSS

Cascading Style Sheets (CSS) controla la presentación visual de los documentos HTML. Permite separar contenido de diseño.

Formas de Incluir CSS:

1. CSS Inline (en la etiqueta):

```
<p style="color: blue; font-size: 18px;">Texto azul</p>
```

2. CSS Interno (en el <head>):

```
<style>
  p {
    color: blue;
    font-size: 18px;
  }
</style>
```

3. CSS Externo (archivo separado):

```
<link rel="stylesheet" href="estilos.css">
```

8. Selectores CSS

Selector de Elemento:

```
p {
  color: red;           /* Todos los párrafos */
}
```

Selector de Clase:

```
.destacado {
  background-color: yellow; /* class="destacado" */
}
```

Selector de ID:

```
#encabezado {
  font-size: 24px;        /* id="encabezado" */
}
```

Selector Descendiente:

```
div p {
  margin: 10px;          /* Párrafos dentro de divs */
}
```

9. Propiedades CSS Fundamentales

Colores y Tipografía:

```
.texto-principal {
  color: #333333;        /* Color del texto */
  background-color: #f0f0f0; /* Color de fondo */
  font-family: Arial, sans-serif;
  font-size: 16px;
  font-weight: bold;      /* Grosor: normal, bold */
  text-align: center;     /* Alineación del texto */
}
```

10. Box Model en CSS

Cada elemento HTML es una "caja" con cuatro Áreas:

- 2 Content: El contenido real (texto, imagen)
- 2 Padding: Espacio interno entre contenido y borde
- 2 Border: Borde alrededor del padding
- 2 Margin: Espacio externo alrededor del borde

Ejemplo Práctico del Box Model:

```
.caja {  
    width: 200px;          /* Ancho del contenido */  
    height: 100px;         /* Alto del contenido */  
    padding: 20px;          /* Espacio interno */  
    border: 2px solid #000;  /* Borde negro sólido */  
    margin: 10px;           /* Espacio externo */  
    background-color: lightblue;  
}  
/* Ancho total = 200 + 20*2 + 2*2 + 10*2 = 264px */
```

11. Layout con Flexbox

Flexbox es una herramienta poderosa para crear layouts flexibles y alinear elementos de forma eficiente.

Contenedor Flex:

```
.contenedor {  
    display: flex;  
    justify-content: center;  /* Alineación horizontal */  
    align-items: center;      /* Alineación vertical */  
    height: 100vh;           /* Altura completa */  
    gap: 20px;                /* Espacio entre elementos */  
}
```

Elementos Flex:

```
.elemento {  
    flex: 1;                  /* Crecer para llenar espacio */  
    min-width: 200px;          /* Ancho mínimo */  
    background-color: #e0e0e0;  
    padding: 20px;  
    border-radius: 8px;        /* Esquinas redondeadas */  
}
```

Direcciones de Flexbox:

- 2 flex-direction: row (horizontal, por defecto)
- 2 flex-direction: column (vertical)
- 2 flex-direction: row-reverse (horizontal invertido)
- 2 flex-direction: column-reverse (vertical invertido)

Justificación y Alineación:

- 2 justify-content: flex-start, center, flex-end, space-between
- 2 align-items: flex-start, center, flex-end, stretch

12. CSS Grid BÁSICO

CSS Grid permite crear layouts bidimensionales complejos con filas y columnas de manera muy intuitiva.

Grid Container BÁSICO:

```
.grid-container {  
    display: grid;  
    grid-template-columns: 1fr 2fr 1fr; /* 3 columnas */  
    grid-template-rows: auto auto auto; /* 3 filas */  
    gap: 20px; /* Espacio entre celdas */  
    padding: 20px;  
}
```

Grid Items:

```
.grid-item {  
    background-color: #f0f0f0;  
    padding: 20px;  
    border: 1px solid #ddd;  
    text-align: center;  
}  
  
.destacado {  
    grid-column: 1 / 3; /* Ocupa 2 columnas */  
    grid-row: 1 / 2; /* Primera fila */  
    background-color: #ffffcc;  
}
```

13. Responsive Design

El diseño responsive asegura que el sitio web se vea bien en todos los dispositivos: móviles, tablets y escritorio.

Media Queries:

```
/* Estilos para móviles (por defecto) */  
.contenedor {  
    width: 100%;  
    padding: 10px;  
}  
  
/* Estilos para tablets (768px y más) */  
@media screen and (min-width: 768px) {  
    .contenedor {  
        width: 750px;  
        margin: 0 auto;  
    }  
}  
  
/* Estilos para escritorio (1024px y más) */  
@media screen and (min-width: 1024px) {  
    .contenedor {  
        width: 1000px;  
    }  
}
```

Meta Viewport (Importante):

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

14. Proyecto PrÆctico: PÆgina Personal

Vamos a crear una pÆgina personal completa que combine HTML semÆntico con CSS moderno y diseæo responsivo.

Estructura HTML:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Juan PØrez - Desarrollador Web</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <header>
    <nav>
      <ul>
        <li><a href="#inicio">Inicio</a></li>
        <li><a href="#sobre-mi">Sobre Mí</a></li>
        <li><a href="#proyectos">Proyectos</a></li>
        <li><a href="#contacto">Contacto</a></li>
      </ul>
    </nav>
  </header>
  <main>
    <section id="inicio" class="hero">
      <h1>Juan PØrez</h1>
      <p>Desarrollador Web Full Stack</p>
      <button>Ver mi trabajo</button>
    </section>
    <section id="sobre-mi">
      <h2>Sobre Mí</h2>
      <p>Soy un desarrollador apasionado...</p>
    </section>
    <section id="proyectos">
      <h2>Mis Proyectos</h2>
      <div class="grid-proyectos">
        <article class="proyecto">
          <h3>E-commerce</h3>
          <p>Tienda online con React</p>
        </article>
        <!-- MÆs proyectos... -->
      </div>
    </section>
    <section id="contacto">
      <h2>Contacto</h2>
      <form>
        <input type="text" placeholder="Tu nombre">
        <input type="email" placeholder="Tu email">
        <textarea placeholder="Tu mensaje"></textarea>
        <button type="submit">Enviar</button>
      </form>
    </section>
  </main>
  <footer>
    <p>© 2024 Juan PØrez. Todos los derechos reservados.</p>
  </footer>
</body>
</html>
```

15. CSS para el Proyecto

```
/* Reset y estilos globales */
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

body {
  font-family: 'Arial', sans-serif;
  line-height: 1.6;
  color: #333;
}

/* Header y navegación */
header {
  background: #2c3e50;
  color: white;
  padding: 1rem 0;
  position: fixed;
  width: 100%;
  top: 0;
  z-index: 1000;
}

nav ul {
  display: flex;
  list-style: none;
  justify-content: center;
  gap: 2rem;
}

nav a {
  color: white;
  text-decoration: none;
  transition: color 0.3s;
}

nav a:hover {
  color: #3498db;
}

/* Sección hero */
.hero {
  background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
  color: white;
  height: 100vh;
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: center;
  text-align: center;
}

/* Grid de proyectos */
.grid-proyectos {
  display: grid;
  grid-template-columns: repeat(auto-fit, minmax(300px, 1fr));
  gap: 2rem;
  padding: 2rem;
}

/* Responsive */
@media (max-width: 768px) {
  nav ul {
    flex-direction: column;
    gap: 1rem;
  }
}
```