

## PRÁCTICA: Desarrollo de un sitio web con HTML5 y CSS3

### Objetivo

El objetivo de esta práctica es que los alumnos desarrollen un sitio web aplicando los conceptos de HTML5 y CSS3. Se espera que implementen una estructura organizada y semántica, con diseño responsivo, animaciones y efectos avanzados de CSS.

### Requisitos Generales

- **Editor de Código:** Se recomienda utilizar **Visual Studio Code** con los complementos:
  - *HTML CSS Support*: para mejorar la autocompleción y sugerencias.
  - *Live Server*: para visualizar el sitio en un servidor local de manera instantánea.
- **Estructura del Proyecto:**
  - Utilizar HTML5 para la estructura del contenido con etiquetas semánticas (*header, nav, footer, section, aside, article*).
  - Aplicar estilos desde un archivo **CSS externo** para mantener una organización limpia.
  - Utilizar el modelo de cajas de CSS (*box model*) para definir la disposición de los elementos.

### Estructura del Sitio Web: Gimnasio

#### 1. Página Principal (Home)

- **Encabezado (Header)** con un **mega-menú desplegable**, donde se incluyan opciones como Clases, Entrenadores y Precios.
- **Sección Hero** con un **video de fondo** (por ejemplo, de un entrenamiento motivacional) y un **overlay de texto animado**.
- **Clases destacadas:** Tarjetas (*cards*) con **efectos hover y transiciones CSS**.
- **Contador animado** con CSS Keyframes (Ejemplo: "500+ socios satisfechos").
- **Testimonios** en un **carrusel de solo CSS**.
- **Pie de página (Footer)** con:
  - Formulario para suscribirse a un **newsletter**.
  - Enlaces a **redes sociales**.
  - **Mapa interactivo embebido** (Google Maps o OpenStreetMap).

#### 2. Página de Clases (*Grid Avanzado + Filtros*)

- **Sistema de filtrado por categoría** (Musculación, CrossFit, Yoga, Spinning) utilizando solo CSS (:checked y combinadores).
- **Galería tipo "masonry"** con **CSS Grid**, mostrando imágenes de diferentes alturas.

- **Tarjetas con información de las clases**, con efectos de **zoom y overlay** al pasar el cursor.
- **Tabla responsiva de horarios semanales**, adaptada para vista en dispositivos móviles y escritorio.

### 3. Página de Entrenadores (*Perfiles Interactivos*)

- **Tarjetas de entrenadores con efecto flip** (rotación al hacer hover para ver detalles).
- **Barra de habilidades animada** (Ejemplo: Fuerza 90%, Flexibilidad 80%).
- **Sistema de rating** con estrellas animadas (CSS-only).

### 4. Formulario de Contacto Avanzado

- **Validación en tiempo real** con los selectores **:valid y :invalid**.
- **Efecto de carga con spinner CSS** al enviar el formulario.
- **Diseño responsivo usando Flexbox y Grid**.
- **Modal de confirmación**, que aparece al enviar el formulario correctamente.

### 5. Página de Precios (*Comparador de Planes*)

- **Tabla de precios** con un efecto de **resaltado al pasar el cursor**.
- **Toggle entre plan mensual y anual**, cambiando los precios con CSS.
- **Tooltips explicativos** (Ejemplo: "Incluye acceso ilimitado").

### 6. Blog de Fitness (*Estilo Revista*)

- **Diseño tipo "newspaper"** usando CSS Grid.
- **Sistema de tags y categorías**, permitiendo filtrado solo con CSS.
- **Sección de comentarios estilizada**, con avatares creados con CSS y pseudo-elementos.
- **Efecto Scroll Reveal**, donde los artículos aparecen al hacer scroll.

## Técnicas Avanzadas a Implementar

### CSS Avanzado

- Uso de **Custom Properties (Variables CSS)** para definir colores y temas.
- **Clip-path & Masking** para imágenes con formas creativas.
- **Scroll Snap** para secciones que encajan perfectamente al hacer scroll.
- **Modo Oscuro (Dark Mode)** con CSS Variables y un mínimo de JavaScript.

### Animaciones y Efectos

- **Micro-interacciones** en botones, enlaces y tarjetas.



- **Transiciones SVG** (animaciones en el logo e iconos interactivos).
- **Keyframes para animaciones avanzadas**, como el contador y el spinner de carga.

### Accesibilidad

- Contraste adecuado (cumpliendo los estándares AA/AAA).
- Estados de **focus visibles** para mejorar navegación con teclado.
- Uso de **ARIA labels** para mejorar la accesibilidad de elementos interactivos.

### Entregables

Los alumnos deben entregar:

1. **Código HTML y CSS bien estructurado.**
2. **Documentación técnica**, explicando las decisiones de diseño y código implementado.
3. **Demo publicada en GitHub Pages o Netlify**, permitiendo visualizar el sitio en línea.
4. **Video de presentación**, donde se muestren las funcionalidades implementadas.

Paginas de referencia:

<https://www.gymtop.com.ar/>