



# Ejercicios CSS Avanzados

## Fichero plantilla index.html

A partir de la siguiente plantilla web que se anexa que consta de:

### **Cabecera (Header):**

- Ubicada en la parte superior, contendrá un logotipo y nombre del alumno/a

### **Barra de navegación (Nav):**

- Puede estar dentro de la cabecera o en una sección separada.
- Incluye enlaces a las distintas páginas del sitio, en este caso los enlaces a cada uno los ejercicios que se plantean en el **apartado B**

### **Footer (pié de página)**

- Referencia a la fecha, ejercicio y autor.

INCLUYE EN ELLA LOS EJERCICIOS QUE SE EXPONEN A CONTINUACIÓN.

## A. Animaciones avanzadas con CSS

- Crea una **animación de carga** (*loading spinner*) con formas personalizadas.
- Diseña un menú de navegación con **animaciones de transición y transformación** (aparece/desaparece con efecto "*slide*").
- Implementa una animación de "*hover*" para botones con un efecto de rebote o transformación compleja.
- Realiza una animación en bucle que cambie gradualmente el color de fondo de una página de acuerdo con el tiempo.

## B. Diseños Responsivos con Grid y Flexbox

- Crea una cuadrícula de **productos de venta** con **CSS Grid**, que se reorganice según el tamaño de la pantalla, incluyendo áreas más grandes para productos destacados (4x4)
- Implementa un menú de navegación *responsive* con **Flexbox**, que colapse en un menú "hamburguesa" en pantallas pequeñas.
- Diseña una galería de imágenes que se ajuste automáticamente usando **CSS Grid** y que permita diferentes tamaños de imágenes en la cuadrícula.

## C. Efectos avanzados de transiciones y transformaciones

- Crea una **tarjeta interactiva** que gire en 3D al pasar el ratón por encima (efecto *flip card*), mostrando contenido en la parte trasera.
- Diseña un botón con un efecto "*neumorfismo*" que cambie de estilo al ser presionado.
- Implementa una **animación de un texto** que tenga un efecto de "**escritura**" o "**typing**", donde las letras aparezcan una por una.
- Crea un **efecto de parallax** usando solo CSS, en el que diferentes capas de contenido se muevan a diferentes velocidades al hacer *scroll*.

## D. Diseños de layout complejos

- Diseña un ***dashboard*** con múltiples paneles, estadísticas, y gráficos usando **CSS Grid**. Asegúrate de que el diseño sea completamente responsivo.
- Crea un diseño de **revista digital donde los artículos** se organizan en varias columnas con títulos superpuestos a imágenes usando **position: absolute** y **z-index**.
- Implementa un diseño tipo ***masonry*** (muro de ladrillos), como el de Pinterest, utilizando solo CSS.

## E. CSS Variables (Custom Properties)

- Crea un sistema de **temas dinámicos** para una página, donde al cambiar un color, toda la página se actualice (usando **CSS Variables** y *:root*).
- Diseña un formulario interactivo con estilos y validaciones (p.e. correo electrónico) que cambien según los estados de los inputs (usando variables para controlar los colores).

## F. Efectos avanzados de pseudo-elementos

- Crea un **"*tooltip*"** (cuadro de información emergente) personalizado utilizando pseudo-elementos ***::before*** y ***::after*** para las flechas y el contenido.
- Diseña una barra de progreso con contenido personalizado dentro, usando **pseudo-elementos** para estilizar el texto dentro de la barra.
- Utiliza pseudo-elementos para crear **formas geométricas complejas** (triángulos, trapecios, hexágonos) y úsalas para crear un logo o diseño decorativo.



## G. Filtros y Blend Modes

- Aplica **filtros CSS** (*blur, brightness*, etc.) en imágenes al hacer hover y crea un efecto de desenfoque que revele texto encima de la imagen.
- Utiliza **blend modes** para lograr efectos de mezcla de imágenes y texto que se combinen de formas creativas al superponerse.

## H. Interacciones avanzadas con formularios

- Diseña un **formulario** de registro que utilice **transiciones y animaciones CSS** para crear una experiencia suave y agradable. Implementa **efectos de "focus"** personalizados en los campos.
- Implementa **validaciones visuales en tiempo real con CSS puro**, cambiando el borde o fondo de los campos de entrada basados en el estado del formulario (usando pseudo-clases como ***:invalid*** o ***:valid***).

## I. Clip-path y formas personalizadas

- Crea un diseño donde las imágenes tengan bordes con formas no convencionales (como **círculos, hexágonos o formas personalizadas**) utilizando ***clip-path***.
- Diseña un encabezado de página con un **borde inferior curvado** o diagonal usando **clip-path**.

## J. Modo oscuro y accesibilidad

- Implementa un **modo oscuro** utilizando únicamente CSS, aprovechando ***prefers-color-scheme***, donde el esquema de colores cambia automáticamente en función de la configuración del usuario.
- Asegúrate de que la página sea accesible, utilizando **colores contrastantes y CSS** para mejorar la legibilidad y navegación para usuarios con discapacidades visuales.

## K. Botones personalizados

- Crea un **botones con estilos personalizables** (color de fondo, borde, tamaño, etc.).
- Debe tener diferentes estados (normal, hover, activo y deshabilitado).
- Usa variables CSS para permitir la personalización de colores y tamaños.
- Usa ***flexbox o CSS grid*** para presentarlos
- No deben diferenciarse solo en el color y/o tamaño.

## L. Uso avanzado de combinadores y selectores

- Realiza un diseño donde **solo el primer y el último elemento de una lista** tengan un estilo diferente, utilizando combinadores avanzados como *:first-child*, *:last-child*, *:nth-child()*.
- Crea un **menú de navegación con elementos de lista** que se iluminan dependiendo de su posición relativa a otros elementos (utilizando + y ~ para seleccionar elementos adyacentes o hermanos).

## M. Transformaciones SVG con CSS

- Diseña un **logo SVG interactivo** donde los elementos se animen con *@keyframes* y CSS.
- Crea **gráficos vectoriales** que cambien de **color y forma al hacer hover** o en respuesta a eventos usando solo CSS.