

Actividad Práctica 1 – HTML y CSS básicos

En esta actividad vamos a poner en práctica los conceptos vistos en clase para crear una pequeña estructura de sitio web utilizando únicamente las etiquetas y propiedades que ya conocemos. **No** utilizaremos herramientas más avanzadas como flexbox, grid o JavaScript.

Objetivo

Crear un sitio web con tres páginas HTML:

1. **index.html**
2. **inicio-sesion.html**
3. **registro.html**

Todas las páginas deberán tener **el mismo encabezado (header)** y **el mismo pie de página (footer)**.

Instrucciones

1. Página principal – **index.html**

- Debe incluir un **header** ubicado arriba del todo, sin margen y pegado al borde superior de la página.
 - El header tendrá dos opciones:
 - **Inicio de sesión**
 - **Registro**
 - Debajo del header, colocar una **imagen centrada** (ustedes eligen la temática).
 - Al final de la página, colocar un **footer** pegado abajo del todo.
 - Al hacer clic en:
 - **Inicio de sesión** → llevar a **inicio-sesion.html**
 - **Registro** → llevar a **registro.html**
-

2. Página de Inicio de sesión – `inicio-sesion.html`

- Incluir el **mismo header y footer** que en `index.html`.
 - En el contenido principal, colocar un formulario que pida:
 - **Email**
 - **Contraseña**
 - Ambos campos deben tener el atributo **required** para que sean obligatorios.
-

3. Página de Registro – `registro.html`

- Incluir el **mismo header y footer** que en `index.html`.
 - En el contenido principal, colocar un formulario que pida:
 - **Nombre**
 - **Apellido**
 - **Edad**
 - **Fecha de nacimiento**
 - **Email**
 - **Contraseña**
 - Todos los campos deben ser **requeridos**.
-

Notas importantes

- **No usar** flexbox ni grid. Pueden usar márgenes, paddings, text-align, etc.
- El header y el footer deben estar siempre en su posición: el header arriba, el footer abajo.
- Esta forma de tener varias páginas HTML distintas para cada vista es algo que **solo haremos por esta actividad**.
 - **Multi-page:** Cada vista es un archivo HTML diferente, y el navegador recarga la página al navegar.
 - **SPA (Single Page Application):** En este modelo, el contenido cambia dinámicamente sin recargar toda la página. Esto es más rápido y fluido, y lo veremos más adelante cuando trabajemos con JavaScript y frameworks.