

# Proyecto Integrador I

04-10-2025

## Ejercicios DOM

### Consideraciones Generales:

- Realiza cada ejercicio por separado.
- Cada ejercicio debe contener 3 archivos enlazados, un HTML, un CSS y un JavaScript.
- Realiza estos ejercicios de complejidad ascendente, en la próxima clase analizaremos las producciones y desarrollaremos alguno paso a paso, de manera grupal.

### Ejercicio 1: Cambiar texto y estilo

En una página web tenemos un título de bienvenida fijo que dice "Hola DOM". Queremos que la página sea más interactiva: al hacer clic en un botón, el título debe actualizarse con un nuevo mensaje y resaltarse visualmente para que el usuario note el cambio. Debes detectar el clic en el botón, cambiar el texto del título y alternar una clase CSS que modifique su apariencia.

### Ejercicio 2: Contador (+/-/reset)

Estás creando un mini contador para sumar productos a un carrito. Implementa un contador que comience en 0 y permita: sumar con "+", restar con "-" (sin permitir valores negativos) y reiniciar con "Reset". El botón de restar debe desactivarse automáticamente cuando el contador esté en 0.

### Ejercicio 3: Lista de tareas

Construye una lista de tareas (to-do). El usuario escribe una tarea en un input y al enviar el formulario la tarea aparece en una lista. Cada tarea debe tener un botón "Eliminar" para quitarla de la lista. Debes crear y eliminar nodos del DOM dinámicamente.

### Ejercicio 4: Filtro en vivo

Implementa un buscador en vivo para una lista de nombres. A medida que el usuario escribe en el campo de búsqueda, deben ocultarse los nombres que no coinciden (ignorando mayúsculas/minúsculas) y mostrarse los que sí.

### Ejercicio 5: Modal (ESC y clic fuera)

Crea una ventana modal que se abra al pulsar un botón. Debe poder cerrarse: con el botón "Cerrar" del modal, haciendo clic fuera del contenido (sobre el fondo oscurecido) y presionando la tecla ESC. Usa clases para mostrar/ocultar el modal.

### **Ejercicio 6: Pestañas con data-\***

Construye un sistema de pestañas (tabs) donde cada botón tenga un atributo data-panel que indica el id del panel a mostrar. Al hacer clic en una pestaña, debe activarse visualmente y mostrar su panel asociado, ocultando los demás.

### **Ejercicio 7: Acordeón dinámico**

Genera un acordeón de preguntas frecuentes a partir de un arreglo de datos. Cada sección debe tener un encabezado clickeable que muestre u oculte su contenido. El marcado del acordeón debe crearse desde JavaScript (no escrito a mano en el HTML).

### **Ejercicio 8: Validación de formulario en vivo**

Crea un formulario con campos Nombre y Email. Valida en vivo: Nombre (mínimo 3 caracteres) y Email (formato válido). Muestra mensajes de error y pinta el estado de los inputs. Al enviar, si hay errores, indícalo; si está correcto, muestra un mensaje de éxito.

### **Ejercicio 9: Tabla dinámica con orden**

Permite agregar filas (Nombre, Edad) a una tabla mediante un formulario. Los encabezados de la tabla deben permitir ordenar por columna, alternando entre ascendente/descendente.

### **Ejercicio 10: Mini Kanban (drag & drop + localStorage)**

Crea un tablero con columnas Pendiente, En progreso y Hecho. Permite crear tarjetas (se agregan a Pendiente), arrastrarlas entre columnas y guardar/restaurar el estado usando localStorage.