

Ejercicios Repaso – CSS Básico

Resultado de Aprendizaje

RA2 Planifica la creación de una interfaz web valorando y aplicando especificaciones de diseño.

Objetivo

- Elemento CSS: **Dropdowns**
- Dominar los patrones de dropdown sin JavaScript mediante CSS (`:hover`, `:focus-within`).
- Practicar posicionamiento (`position`, `top/left/right`, `z-index`) y el flujo de apilamiento.
- Integrar menús desplegables en navbars y sidebars, y diseñar variantes (alineación derecha, iconos, multi-nivel, megamenú).
- Asegurar accesibilidad mínima: foco visible, navegación por teclado, roles básicos ARIA y contraste
- Se proporcionan archivos **index.html** y **styles.css**

Referencia: https://www.w3schools.com/css/css_dropdowns.asp

Ejercicio 01 - Dropdown básico (hover)

- Construir un desplegable elemental con `position:absolute` y visualización mediante `:hover`.
- Pasos sugeridos:
 1. Estructura `.dropdown > .trigger + .menu` (trigger puede ser `<a>` o `<button>`).
 2. Oculta `.menu` por defecto (`display:none`).
 3. Posiciona `.menu` bajo el disparador (`top:100%`, `left:0`).
 4. Muestra `.menu` con `.dropdown:hover .menu { display:block; }`.
 5. Aplica bordes, sombra suave y radios para legibilidad.

Ejercicio 02 - Dropdown accesible (:focus-within)

- Habilitar apertura con teclado: el dropdown se muestra al enfocar el trigger o sus enlaces.
- Pasos sugeridos:
 1. Añade `:focus` visible al trigger.
 2. Usa `.dropdown:focus-within .menu { display:block; }`.
 3. Asegura orden lógico de tabulación (trigger → primer enlace...).

Ejercicio 03 - Dropdown en navbar

- Objetivo: Integrar el dropdown en una barra de navegación horizontal.
- Pasos sugeridos:
 1. Crea `
 - 2. Posiciona `.submenu` con `absolute` (top:100%).
 - 3. Muestra con `.has-sub:hover > .submenu` y `.has-sub:focus-within > .submenu`.
 - 4. Ajusta padding, gaps y radio; evita que la navbar salte al abrir.

Ejercicio 04 - Dropdown en sidebar

- Objetivo: Crear un menú vertical con un submenú que se despliega a continuación.
- Pasos sugeridos:
 1. Coloca los elementos de la sidebar en columna.
 2. Usa `.submenu`` estático (sin `absolute`) que aparece abajo del padre.
 3. Controla la apertura con ``:hover`` y ``:focus-within``.
 4. Cuida los espaciados internos y la lectura en estrecho.

Ejercicio 05 - Alineación a la derecha

- Objetivo: Hacer que el menú se ancle al borde derecho del trigger.
- Pasos sugeridos:
 1. Cambia a `.menu.right { right:0; left:auto; }``.
 2. Revisa que la sombra y bordes no se corten.
 3. Verifica en pantallas pequeñas (no debe salirse del viewport).

Ejercicio 06 - Iconos y separadores

- Mejorar la legibilidad con iconos (emoji o SVG) y separadores visuales.
- Pasos sugeridos:
 1. Usa ``display:flex; align-items:center; gap`` dentro de enlaces.
 2. Incluye ``<hr>`` estilizado como separador entre grupos.
 3. Asegura contraste adecuado y no dependas solo del color.

Ejercicio 07 - Multi-nivel (anidado)

- Objetivo: Crear un segundo nivel de menú que aparece a la derecha del primero.
- Pasos sugeridos:
 1. Estructura `.menu.nested .has-sub > a + .submenu``.
 2. Posiciona el submenú con ``left:100%; top:0``.
 3. Muestra el submenú con ``:hover`/`:focus-within`` en el `.has-sub`` padre.

Ejercicio 08 - Megamenú (grid)

- Objetivo: Diseñar un panel amplio con 2–3 columnas de enlaces organizados por categorías.
- Pasos sugeridos:
 1. Crea `.mega`` con ``display:grid`` y columnas repetidas.
 2. Muestra `.mega`` desde `.has-mega:hover`` y ``:focus-within``.
 3. Cuida tipografías, espaciado y jerarquía (títulos, grupos).
 4. Ajusta a 1 columna en `< 768px``.