

Actividades Unidad 4: CSS III

Actividades prácticas

1. Barra de botones horizontal (flex-direction + justify-content)

Simula una barra de botones horizontal similar a un menú de acciones de una aplicación web.

El HTML que hay que crear tiene:

- Un único contenedor `<div>` padre con una clase llamada `container`.
- Dentro del contenedor, cuatro elementos hijos.
- Cada hijo debe ser un `<div>` con la clase `item`.
- El texto de cada div será: Guardar, Editar, Eliminar y Cancelar

Los botones deben repartirse uniformemente a lo largo del contenedor.

Requisitos CSS:

- El contenedor debe usar Flex.
- Los ítems deben colocarse en fila.
- Usa `justify-content` para que el espacio entre los botones sea el mismo.
- Añade un color de fondo al contenedor y otro distinto a los ítems.
- Centra el texto dentro de cada botón.

2. Columna de tarjetas (flex-direction + align-items)

Diseña una columna de tarjetas como las que aparecen en un panel lateral de una web.

Todas las tarjetas deben estar centradas horizontalmente dentro del contenedor.

El HTML que hay que crear tiene:

- Un contenedor padre `<div>` con clase `container`.
- Dentro, tres `<div>` hijos, todos con la clase `item`.
- Cada div representará una tarjeta con texto: Tarjeta 1, Tarjeta 2 y Tarjeta 3

Requisitos CSS

- El contenedor debe ser flex y orientado en columna.

- Usa align-items para centrar las tarjetas en el eje secundario.
- Cada tarjeta debe tener un ancho fijo.
- Añade separación vertical entre tarjetas usando gap.

3. Galería de productos (flex-wrap + gap)

Crea una galería de productos que se adapte al ancho del contenedor. Cuando no quepan todos los elementos en una fila, deben pasar automáticamente a la siguiente.

El HTML que hay que crear tiene:

- Un contenedor `<div>` principal con clase container.
- Seis elementos hijos.
- Cada elemento será un `<div>` con clase item.
- El texto de cada elemento será: Elemento 1 hasta Elemento 6.

Requisitos CSS:

- El contenedor debe usar `display: flex`.
- Los ítems deben poder saltar de línea.
- Cada producto debe ocupar aproximadamente el 50% del ancho.
- Usa gap para separar filas y columnas.
- Asigna un ancho fijo al contenedor para forzar el salto de línea.

4. Cabecera web (justify-content + align-items)

Crea la cabecera de una página web con:

- Un logo a la izquierda
- Un menú a la derecha

Todo debe estar alineado verticalmente en el centro.

El HTML que hay que crear tiene:

- Un contenedor principal `<div>` que represente la cabecera.
- Dicho contenedor debe tener la clase container.
- Dentro del contenedor habrá dos elementos hijos, ambos `<div>`:
 - El primer div contendrá el texto LOGO.
 - El segundo div contendrá el texto Inicio | Servicios | Contacto.

Requisitos CSS:

- Usa Flex en el contenedor.
- Coloca los elementos en extremos opuestos del eje principal.
- Centra los elementos en el eje secundario.
- Asigna una altura fija a la cabecera.
- Añade colores para distinguir logo y menú.

5. Layout con orden de elementos (order)

Crea un layout de página con navegación lateral, contenido principal y un aside.

Visualmente, la navegación debe aparecer primero, el contenido en el centro y el aside al final, sin cambiar el orden del HTML.

El HTML que hay que crear tiene:

- Un contenedor principal `<div>` con identificador `main`.
- Dentro del contenedor deben existir tres elementos, en este orden:
 1. Un `<article>` con texto Contenido principal.
 2. Un `<nav>` con texto Menú de navegación.
 3. Un `<aside>` con texto Publicidad.

Requisitos CSS:

- El contenedor debe usar `display: flex`.
- Los elementos deben colocarse en la misma fila.
- Cambia el orden visual de los elementos usando `order`.
- El contenido principal debe ocupar el mayor espacio disponible.
- El menú y el aside deben tener un ancho fijo.