

TALLER N1 - CSS DINER, FLEX FROGGY, GRID GARDEN

Son juegos en línea educativos que ayudan a aprender y practicar CSS de una manera divertida. "CSS Diner" se enfoca en selectores de CSS, "Flex Froggy" en el modelo de Flexbox, y "Grid Garden" en el modelo de CSS Grid.

Partes de un navegador

- **Caché:** La caché del navegador es una memoria de almacenamiento temporal que guarda copias de archivos de sitios web que has visitado, como imágenes, archivos HTML, hojas de estilo CSS y scripts JavaScript. El propósito de la caché es acelerar el tiempo de carga de las páginas web en visitas posteriores, ya que el navegador puede cargar estos recursos directamente desde la caché en lugar de volver a descargarlos desde el servidor.

Ventajas de la caché:

Reducción de tiempo de carga: Los archivos almacenados en la caché se cargan más rápido.

Menor uso de datos: Menos necesidad de descargar archivos desde el servidor.

Reducción de la carga en el servidor: Al servir archivos desde la caché, se reduce la demanda en los servidores web.

- **Dom:** El **DOM** (Modelo de Objetos del Documento) es una representación estructurada de un documento HTML o XML en forma de árbol. Cada nodo en este árbol representa una parte del documento, como un elemento, un atributo o un texto. El DOM permite a los lenguajes de programación, como JavaScript, interactuar y manipular la estructura, el estilo y el contenido de las páginas web de manera dinámica.

El DOM es esencial para la manipulación dinámica de las páginas web. Permite a los desarrolladores:

- **Acceder y modificar el contenido:** Cambiar el texto, atributos, estilos, y añadir o eliminar elementos del documento.
- **Responder a eventos:** Asignar manejadores de eventos para que las páginas web puedan reaccionar a las acciones del usuario, como clics o entradas de teclado.
- **Navegar por la estructura del documento:** Recorrer el árbol del DOM para localizar y manipular elementos específicos.

- **Render:** El proceso de **renderizado** es la manera en que el navegador interpreta el código HTML, CSS y JavaScript para convertirlo en la página web visual que ves en la pantalla. Este proceso involucra varios pasos importantes:
 1. **Parseo del HTML:** El navegador lee el HTML y crea el árbol del DOM.
 2. **Parseo del CSS:** El navegador lee las hojas de estilo CSS y genera el árbol de estilo (CSSOM - CSS Object Model).
 3. **Construcción del árbol de renderizado:** El navegador combina el DOM y el CSSOM en un árbol de renderizado que representa el contenido visual de la página.
 4. **Cálculo del layout (diseño):** El navegador determina el tamaño y la posición de cada elemento en la página.
 5. **Pintado:** Finalmente, el navegador dibuja el contenido de cada nodo del árbol renderizado en la pantalla, produciendo la imagen visible de la página web.

Proceso detallado:

- **Parseo del HTML:** Convierte el HTML en el DOM.
- **Parseo del CSS:** Convierte el CSS en el CSSOM.
- **Construcción del árbol de renderizado:** Combina el DOM y el CSSOM.
- **Cálculo del layout:** Calcula el tamaño y la posición de los elementos.
- **Pintado:** Dibuja los píxeles en la pantalla.

Conceptos Fundamentales: XML y JSON

- **XML:** XML (eXtensible Markup Language) es un lenguaje de marcado que se utiliza para almacenar y transportar datos. Es similar a HTML, pero XML está diseñado para ser tanto legible por humanos como por máquinas. XML no tiene etiquetas predefinidas (como HTML); en cambio, se pueden definir etiquetas personalizadas para describir los datos. Su principal uso es la estructuración, el almacenamiento y el intercambio de datos entre diferentes sistemas.

Características de XML:

- **Extensible:** Permite definir etiquetas personalizadas.
 - **Estructurado:** Los datos se organizan en una estructura jerárquica (similar a un árbol).
 - **Independiente de la plataforma:** Se puede utilizar en cualquier sistema operativo.
 - **Legible por humanos:** El formato es claro y fácil de entender.
- **JSON:** JSON (JavaScript Object Notation) es un formato de texto ligero para el intercambio de datos. Es fácil de leer y escribir para los humanos y fácil de analizar y generar para las máquinas. JSON es un subconjunto de la sintaxis de objetos de

JavaScript, pero es independiente del lenguaje, lo que significa que puede ser utilizado con muchos lenguajes de programación diferentes.

Características de JSON:

- **Simplicidad:** Fácil de leer y escribir.
- **Ligero:** Tiene un tamaño reducido, lo que lo hace ideal para el intercambio de datos a través de la web.
- **Estructura de datos:** Utiliza pares clave-valor y arreglos para estructurar los datos.
- **Independiente del lenguaje:** Aunque se basa en la sintaxis de objetos de JavaScript, puede ser utilizado por muchos lenguajes de programación.