Flexbox

Flexbox, o "Flexible Box Layout," es un modelo de diseño en CSS (Cascading Style Sheets) que **se utiliza para diseñar y alinear elementos en un contenedor de una manera más eficiente y predecible** que los modelos de diseño tradicionales.

En Flexbox, los elementos dentro de un contenedor se colocan a lo largo de un solo eje (ya sea el eje horizontal o vertical) y se ajustan automáticamente para ocupar el espacio disponible de la mejor manera posible. Esto facilita la creación de diseños responsivos y la alineación de elementos en diseños complejos.

❶ Contenedor Flex (Flex Container): Es el elemento que envuelve a los elementos que deseas organizar utilizando Flexbox. Para convertir un elemento en un contenedor flex, debes aplicarle la propiedad CSS **display: flex**; o **display: inline-flex**; si deseas que se comporte como un elemento en línea.

❷ Elementos Flex (Flex Items): Son los elementos contenidos dentro del contenedor flex que serán organizados y alineados utilizando Flexbox.

❸ Eje Principal (Main Axis): En un contenedor flex, el eje principal es el eje a lo largo del cual se alinearán los elementos flex. Puede ser horizontal o vertical, y su dirección se define utilizando las propiedades **flex-direction** (por defecto, es horizontal).

❹ Eje Transversal (Cross Axis): El eje transversal es el eje perpendicular al eje principal. Si el eje principal es horizontal, el eje transversal será vertical, y viceversa.

❺ Propiedades clave:

justify-content: Controla la alineación de los elementos a lo largo del eje principal.

align-items: Controla la alineación de los elementos a lo largo del eje transversal.

flex-direction: Define la dirección del flujo de los elementos en el contenedor (fila, fila inversa, columna, columna inversa).

flex-wrap: Decide si los elementos flex deben envolverse en varias líneas si no caben en una sola.

**flex-grow, flex-shrink, flex-basis:** Controlan el crecimiento, la contracción y el tamaño inicial de los elementos flex.

❻ Ordenamiento: Puedes cambiar el orden de los elementos dentro del contenedor flex utilizando la propiedad **order.**

***Flexbox es especialmente útil para diseñar barras de navegación, encabezados, menús desplegables y otros componentes de diseño donde la alineación y la distribución flexible de elementos son importantes.***

Grid

Grid Layout es otro modelo de diseño en CSS que permite crear diseños de página complejos y alinear elementos en una cuadrícula bidimensional. A diferencia de Flexbox, que se enfoca en organizar elementos a lo largo de un solo eje, CSS Grid permite organizar elementos tanto en filas como en columnas, lo que proporciona un mayor control sobre la estructura del diseño.

❶ Contenedor de Cuadrícula (Grid Container): Al igual que en Flexbox, necesitas un contenedor para aplicar Grid Layout. Para convertir un elemento en un contenedor de cuadrícula, utiliza la propiedad CSS **display: grid**;.

❷ Elementos de Cuadrícula (Grid Items): Estos son los elementos hijos del contenedor de cuadrícula que serán colocados en la cuadrícula.

❸ Filas y Columnas: Con Grid, puedes definir explícitamente el número de filas y columnas en tu diseño. Utilizas las propiedades **grid-template-rows** y **grid-template-columns** para establecer el tamaño y la cantidad de filas y columnas, respectivamente.

❹ Celdas de la Cuadrícula: Los elementos de la cuadrícula se colocan en celdas específicas de la cuadrícula utilizando las propiedades **grid-row** y **grid-column**. También puedes utilizar la propiedad **grid-area** para especificar estas propiedades en una sola declaración.

❺ Alineación: **Grid Layout** proporciona propiedades como **justify-items**, **align-items**, **justify-content** y **align-content** para alinear los elementos y las filas o columnas de la cuadrícula según sea necesario.

❻ Áreas de la Cuadrícula: Puedes nombrar áreas de la cuadrícula y asignar elementos a estas áreas utilizando la propiedad **grid-template-areas**, lo que facilita la creación de diseños complejos y comprensibles.

***CSS Grid Layout es especialmente útil cuando necesitas un control preciso sobre el diseño de tu página web, especialmente para estructuras de página complejas y responsivas. Puedes combinar Grid Layout con Flexbox para lograr diseños aún más sofisticados y flexibles.***