**Tema 6: Flexbox**

**Contenedor flexible, en inglés: Flex container.**

Un contenedor con elementos flexibles o flex items, se pueden anidar los elementos, como un contenedor, dentro tres columnas(Horizontales¿?) y dentro de ellas un número de filas.



→ Contenedor en cuestión.

Es responsivo pasivamente, y funciona en vertical o en horizontal, o una u otra.

Horizontal →Eje principal (Donde se colocarán los elementos flexibles)

Vertical → Eje cruzado (Mover los elementos)



→ (Rows)



→(Columns)

→ Eje principal Vertical

→Flex Container (propiedades)

display:flex; (elemento flexible asignado y sus hijos también)

flex-direction: **row** o **column** (Cambia la posición en vertical o horizontal)

flex-direction: **row-reverse** o **column-reverse** (Cambia el orden ej: 123 a 321)

flex-wrap: **wrap** o **nowrap** o **wrap-reverse** (**wrap:** Mantiene los tamaños que le hemos puesto y los apila dentro del contenedor como float mas o meno, **wrap-reverse**: Alrevés pero como una figura entera)



→ Wrap-reverse

Diferentes combinaciones:

flex-direction: row-reverse;

flex-wrap: wrap-reverse;

flex-direction: column-reverse;

flex-wrap: wrap-reverse;

Propiedad shorthand:

flex-flow: “flex-direction” “flex-wrap”;

**Respecto al Eje principal**

justify-content (Aliena los elementos en el eje principal)

justify-content: flex-end; (el espacio lo mueve hacia la izquierda y el elemento a la derecha)

justify-content: center; (En el centro)

Modificar los espacios entre los elementos:

justify-content: space-between; (Pegados al borde y los demás esparcidos uniformemente)

justify-content: space-around; (Deja un espacio entre los mismos elementos pero más pequeño en los alrededores)

justify-content: space-evenly; (Más o menos que el around pero dejando el mismo espacio en todo)

**Respecto al Eje cruzado:**

align-items: flex-start; (alineación estándar)

align-items: flex-end; (alineación abajo)

align-items: center; (alineación central)

align-items: stretch; (alineación total, se estiran, para ello debemos quitar el height de los elementos)

Espacio sobrante en eje cruzado:

align-content (Cuando el contenido ocupa más de una línea)

align-content: flex-start;

align-content: flex-end;

align-content: center;

align-content: around;

align-content: between;

align-content: evenly;

**flex-items (Propiedades de los elementos flexibles)**

order: “Un número”; (Establece el orden en los que se presentan los elementos)



→Funciona como el z-index

flex-grow: “Un número” (se ensanchan más o menos, los redimensionan, puede ser que a veces tengamos que quitar el width)



Si los bloques son 1 3 1 , el número 3 crece el triple porque divide.

Si fuera 2 5 3 , el 2 crece un 20%, el 5 un 50% y el 3 30%.

flex-shrink: ”Un número”; (Lo contrario de flex-grow, se redimensionan al contrario, más pequeño)

flex-basis: 200px; (nos indica las dimensiones que intentará llegar sobre el eje principal, tamaño del width, podemos quitar el width, si están los dos se aplica el basis)

flex-basis: content; (Se ajusta automáticamente al contenido que haya dentro)

flex-basis: auto (se adapta)

align-self (El elemento tiene una alineación dependiendo el eje cruzado)

align-self: flex-start; etc

ShortHand:

flex: “flex-grow” “flex-shrink” “flex-basis”;

flex-grow obligatorio en shorthand