CSS

FRONTEND INICIAL

FRONTEND INICIAL
FRONTEND INICIAL

OBJETIVO DEL TEMA

- Qué es CSS
- Sintaxis de CSS (la forma correcta de escribirlo)
- Aprender las tres formas de incluir CSS en un documento HTML.
- Entender qué es un selector CSS y utilizarlo para cambiar el aspecto de elementos HTML
- Entender los conceptos de herencia y especificidad en CSS

¿Qué es Css?

Pilares

- Herencia
- Cascada
- Especificidad

Son los estilos de tu Web

- Selector
- Propiedad
- Valor

Cascading
Stheet
Style

Union

HTML + CSS

Selectores

Css

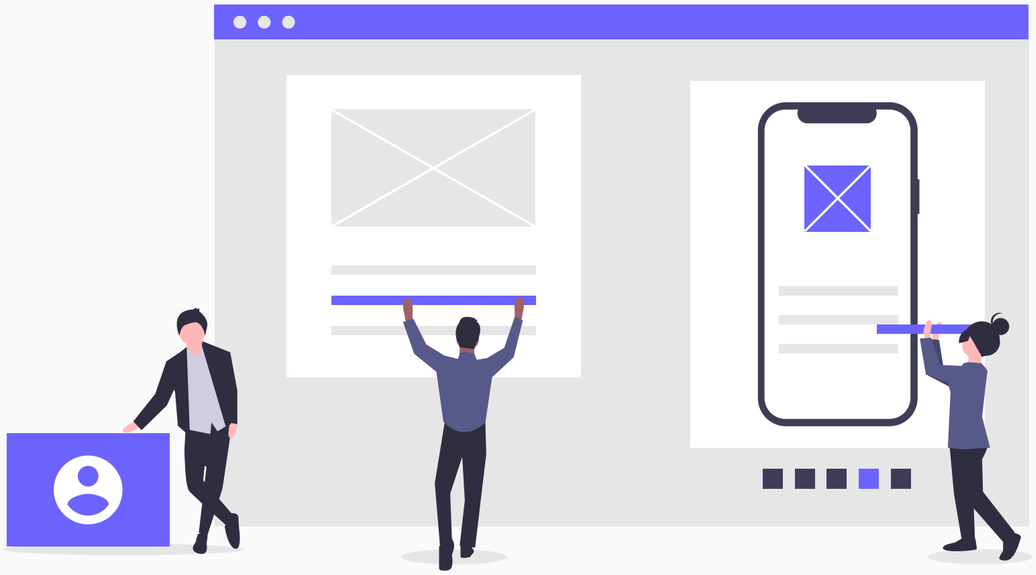
Las Hojas de Estilo en Cascada (del inglés Cascading Style Sheets)
o CSS es el lenguaje
de estilos utilizado para manipular la presentacion de los elementos de nuestra web

Estructura Basica

1. *Selector*
2. *Propiedad*
3. *Valor*

Unir Html y Css

Existen 3 formas para incluir CSS en documento web, la mejor practica es haciendo un llamado a nuestra hoja de estilos externa.



Sintaxis

Regla CSS

Selector

Un selector, es decir un comando que selecciona uno o más elementos del documento HTML, por ejemplo el <body>, los <p>, etc.

Propiedad

Dentro de las llaves se escribe una propiedad css que queremos modificar (el color, el tamaño de la letra, etc)

Valor

un valor para esa propiedad, separados por dos puntos (:), Pueden enumerarse muchas propiedades a modificar al mismo tiempo, separándolas con un punto y coma (;)

PILARES CSS

Cascada

Si se encuentra un estilo o regla css dos o mas veces en el código se aplicara el estilo la ultima vez que se declaro

Especificidad

El navegador evalua si existe algun conflicto de estilos y aplicara solo que tenga mayor especificidad

Herencia

Es la capacidad de heredar estilos de elementos padres y así no tener que asignarle estilos a cada elemento

Ejemplo body { color: blue; } - h1 { color: blue; } no es necesario hacer esto

SELECTORES - PARTE 1

Elemento o Tipo (tag)

Se refiere a una etiqueta HTML.

Clases (.)

Se refiere a un atributo de un elemento HTML (es lo recomendado para aplicar estilos a tu sitio web), no pueden comenzar con numero, puedes usar emoji estandar (caracter unicode) para clases.

ID (#)

Se refiere a un identificador unico que no se va a repetir nunca.

Universal (*)

Se refiere a seleccionar todos los elementos y aplicara los estilos.

```css
h1 {
    color: red;
}

.clase {
    width: 100px;
    height: 100px;
}

#id {
    background-color: yellow;
}

* {
    color: blue;
}
```

SELECTORES - PARTE 2

Agrupados (,)
Se refiere a selectores separados por comas.

Descendientes (espacio)
Se refieres o indican cuando un selector es hijo o nieto no importa el nivel en donde se encuentre el elemento dentro de otro. cuidado con los espacios (a b != a.b).

Hijo Directo (>)
Se refiere a seleccionar un selector que sea hijo directo y no a sus selectores de niveles mas a fondo.

```
.clase, .otra-clase {
    width: 100px;
    height: 100px;
}

div ul {
    list-style: none;
}

div > p {
    color: green;
}
```

SELECTORES PARTE 3

Hermano Siguiente (+)
Se refiere a seleccionar a un elemento que esta justo despues de otro.

Hermano Siguientes (~)
Se refiere a seleccionar a un elemento o varios que esta justo despues de otro que sean hermanos.

Atributos y Valor
Se refiere a seleccionar selectores distintos a un atributo de un elemento HTML como lo son los ID y las Class se escribre [attr] o [attr="valor"].

```
1  div + p {
2      color: red;
3  }
4
5  div ~ p {
6      color: blue;
7  }
8
9  [attr="valor"] {
10     color: pink;
11 }
```

Colores CSS

CSS nos permite escribir los colores de tres maneras diferentes indistintamente:

Con su nombre en inglés. Ej. red, green, blue, purple.

Con su código hexadecimal, una secuencia de 16 dígitos entre 0 y F precedidos por un signo '#'. Ej.#CD00CD (magenta).

Con RGB (red, green, blue), dándole un valor entre 0 y 255 a cada uno de los tres colores. Ej. rgb(210, 140, 176).

Entonces, escribir

"background-color: lime;"
"background-color: #00FF00;"
"background-color: rgb(0,255,0);"

daría el mismo resultado, un fondo color lima.

Esto es muy útil porque los diseñadores nunca les van a pasar el nombre de un color, sino que les van a dar un tono específico en hexa o en rgb.

Este recurso está muy bueno para buscar colores:

La página coolors.co

Unidades de medida

Absolutas:
Su valor no cambia

pc, cm, mm, in, Q
pt (1/72in)
px (1/96in)

Relativas:
Su valor es relativo a un escenario o contexto

em, rem, ex, ch - Tamaño de la fuente
em basada en la anchura de la letra "m" de la fuente del elemento
rem basada en la anchura de la letra "m" de la fuente del elemento raiz que es nuestro html
ex basada en la altura de la letra "x" de la fuente del elemento
ch basada en la anchura del "0" de la fuente del elemento

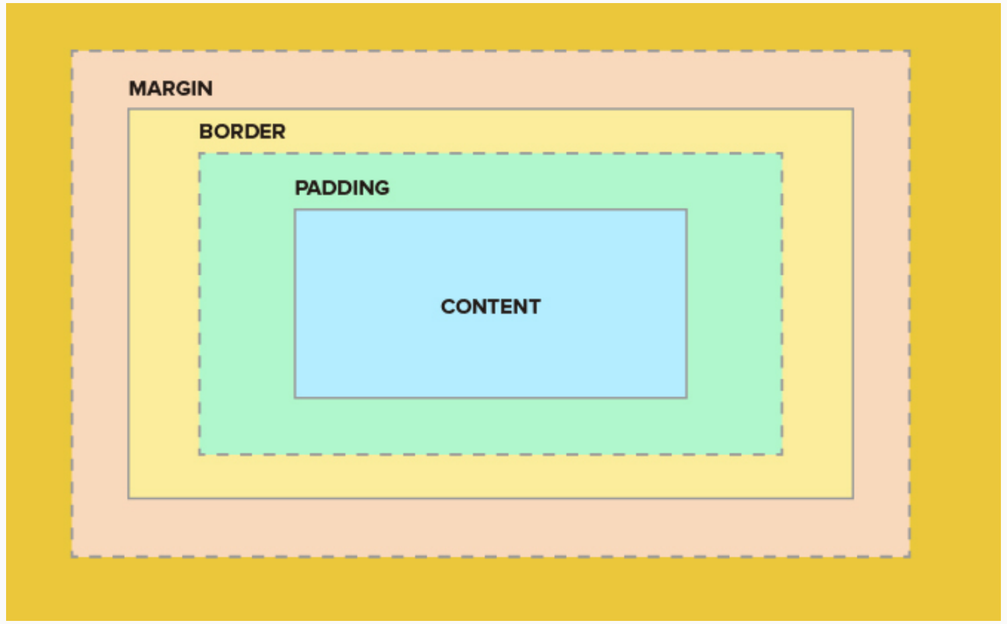
% - Tamaño del contenedor

vw, vh, vmin, vmax - Tamaño del viewport
vw ancho del viewport van del 1 a 100
vh alto del viewport van del 1 a 100
vmax entre vw y vh tomara el que tenga mayor valor
vmin entre vw y vh tomara el que tenga menor valor

Modelo de caja

En el desarrollo web, el modelo de cuadro CSS se refiere a cómo se modelan los elementos HTML en los navegador y cómo la dimensión de esos elementos HTML se deriva de las propiedades CSS

Todo en CSS tiene una caja alrededor, y comprender estas cajas es clave para poder crear diseños con CSS o para alinear elementos con otros elementos.



Modelo de caja

En CSS, en general, hay dos tipos de cajas: cajas en bloque y cajas en línea. Estas características se refieren al modo como se comporta la caja en términos de flujo de página y en relación con otras cajas de la página:

DE BLOQUE

- *La caja se extenderá para llenar todo el espacio disponible que haya en su contenedor.*
- *La caja fuerza un salto de línea al llegar al final de la línea.*

DE LINEA

- *La caja abarcara solo el contenido que tenga dentro sea texto o mas elementos html determinara su ancho y alto.*
- *La caja no fuerza ningún salto de línea al llegar al final de la línea.*

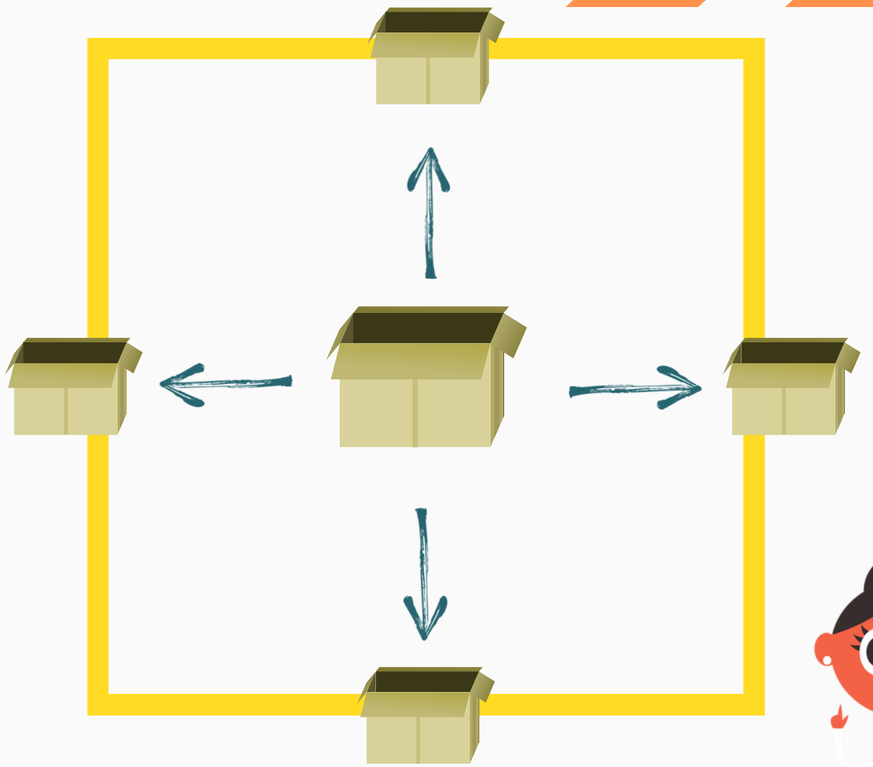
	Width	Height	Padding	Margin
Bloque	SI	SI	SI	SI
En línea	NO	NO	Solo costados	Solo costados
En línea y bloque	SI	SI	SI	SI

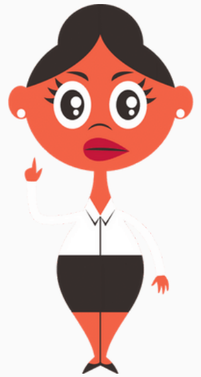
Margin

El margen es un espacio invisible que hay alrededor de la caja. Empuja otros elementos fuera de la caja. Los márgenes pueden tener valores positivos o negativos. el margen siempre se añade después de haber calculado el tamaño de la caja que se ve.

*Podemos controlar todos los márgenes de un elemento a la vez usando la propiedad **margin**, o cada lado individualmente usando las propiedades equivalentes sin abreviar:*

- *margin-top*
- *margin-right*
- *margin-bottom*
- *margin-left*





colapsado de margenes

Si tienes dos elementos cuyos márgenes se tocan, esos márgenes se combinan para convertirse en un solo margen, cuyo tamaño es el del margen más grande.

Padding

El área de relleno se encuentra entre el borde y el área de contenido. A diferencia de los márgenes, el área de relleno no puede tomar valores negativos, por lo que el valor debe ser 0 o positivo.

Cualquier fondo aplicado a tu elemento se mostrará detrás del área de relleno y, generalmente, se usa para mantener el contenido alejado del borde.

Podemos controlar el área de relleno para todos los lados de un mismo elemento usando la propiedad padding, o para cada lado uno de los lados usando las propiedades equivalentes:

- *padding-top*
- *padding-right*
- *padding-bottom*
- *padding-left*

border

padding

Border

El borde se dibuja entre el margen y el área de relleno de una caja.

Hay una gran cantidad de propiedades que sirven para aplicar estilo a los bordes: hay cuatro bordes y cada borde tiene un estilo, un ancho y un color que podemos modificar.

Puedes establecer el ancho, el estilo o el color de los cuatro bordes a la vez utilizando la propiedad border.

Para establecer las propiedades de cada lado de forma individual, puedes utilizar:

- *border-top*
- *border-right*
- *border-bottom*
- *border-left*

Para establecer el ancho, el estilo o el color de todos los lados, usa lo siguiente:

- *border-width*
- *border-style*
- *border-color*

BoxSizing

propiedad que es usada para modificar las propiedades del ***Box Model*** *(Modelo de caja) visto anteriormente, brindándonos aún más control sobre nuestra caja.*

box-sizing acepta principalmente ***3 valores:***

content-box

Es el valor por defecto que tiene el Box Model, su características es que es aditivo, es decir, si agregamos un tamaño al padding y border estos se sumaran al width y height establecidos, agrandando el tamaño real de nuestra caja.

padding-box

Incluye el tamaño del padding dentro del tamaño establecido para la caja, es decir que el tamaño de la caja se reducirá para que al agregarle el padding la caja mantenga el tamaño establecido inicialmente, dejando fuera a border y margin.

border-box

Al igual que el anterior disminuirá su tamaño agregando al padding y border para que el tamaño establecido de la caja se mantenga, dejando fuera al margin.

Background

La propiedad background es un atajo para definir los valores individuales del fondo en una única regla CSS. Se puede usar background para definir los valores de una o de todas las propiedades siguientes:

background-image

Nos permite agregar una o mas imagenes de fondo a nuestro elemento.

background-repeat

Podemos controlar si la imagen se va a repetir sea tanto en el eje X o Y.

background-position

Vamos a poder mover la imagen dentro del elemento y colocarla donde queramos puedo asignarle uno o dos valores (eje X o Y).

background-attachment

Esta propiedad establecera si la imagen de fondo se movera junto a la pagina al hacer scroll o si se quedara fija.

background-size

Nos permite asignar el tamaño de la imagen de fondo.

Position

Para poder tener estructuras mas complejas, tenemos que discutir la propiedad position (posición). que nos van a permitir mover un elemento desde su posición original o superponer elementos.

Esta propiedad acepta ***(relative, absolute, fixed, o sticky),*** *valor por defecto* ***static*** *(es decir sin posicionamiento).*

Las propiedades top y bottom especifican el desplazamiento vertical; las propiedades left y right especifican su desplazamiento horizontal.

relative

No pierde su espacio en el flujo, el punto de referencia son los costados.

absolute

Pierde su espacio en el flujo, el punto de referencia es el contenedor posicionado mas cercano si no el body sera su referencia.

fixed

Pierde su espacio en el flujo, el punto de referencia es la ventana del navegador y se mantendra fijo.

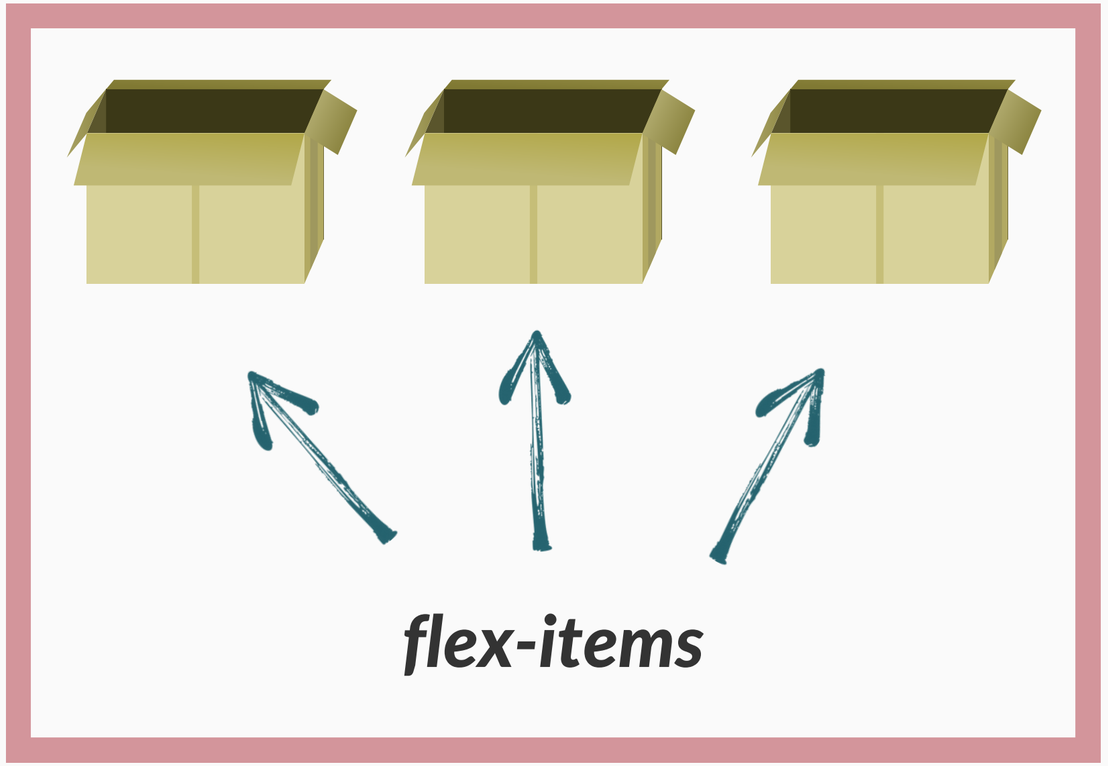
sticky

Es tratado como un elemento posicionado relativamente hasta que su bloque contenedor cruza un límite establecido.

Flexbox

fue diseñado como un modelo unidimensional de layout, y como un método que pueda ayudar a distribuir el espacio entre los ítems de una interfaz y mejorar las capacidades de alineación.

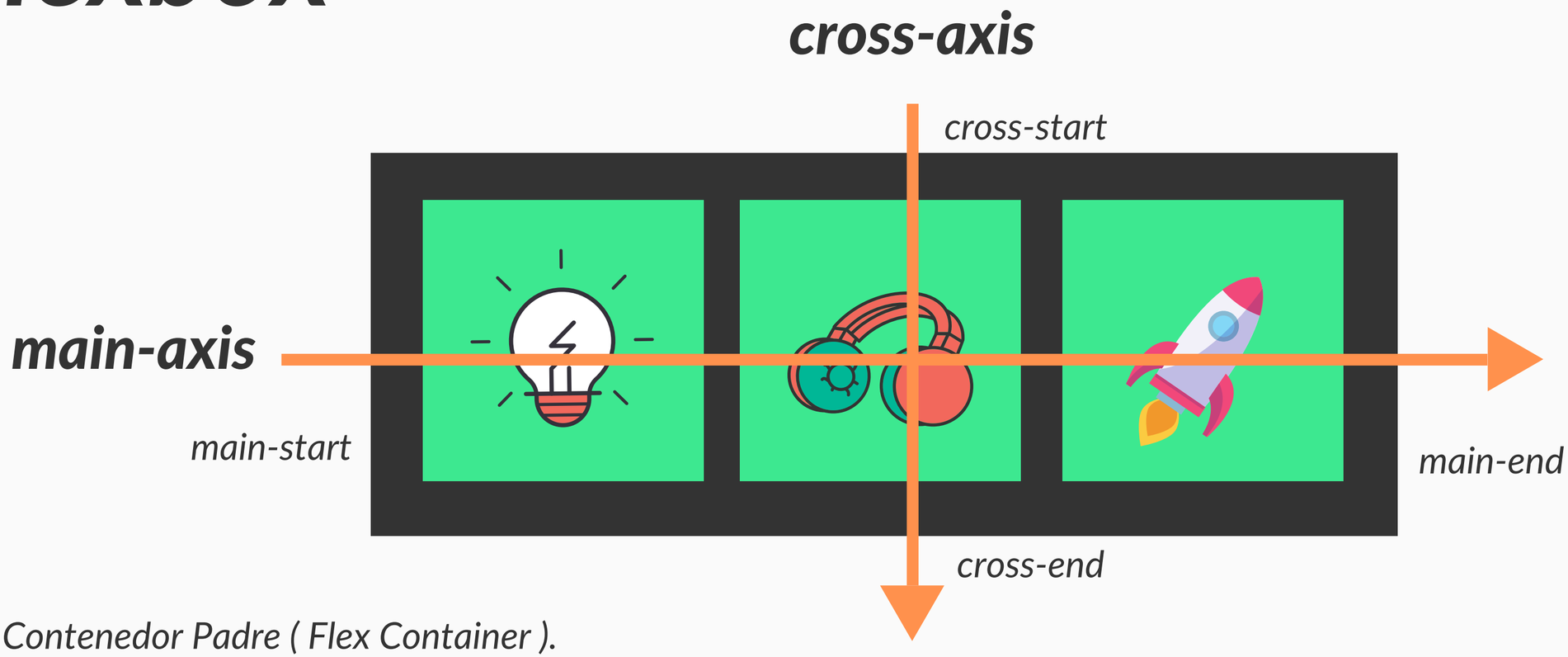
Cuando describimos a flexbox como unidimensional destacamos el hecho que flexbox maneja el layout en una sola dimensión a la vez — ya sea como fila o como columna.



flex-container

- *display: flex*
- *display: inline-flex*

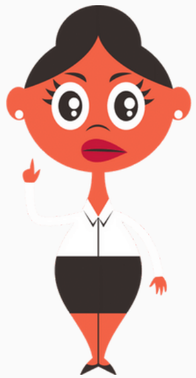
Flexbox



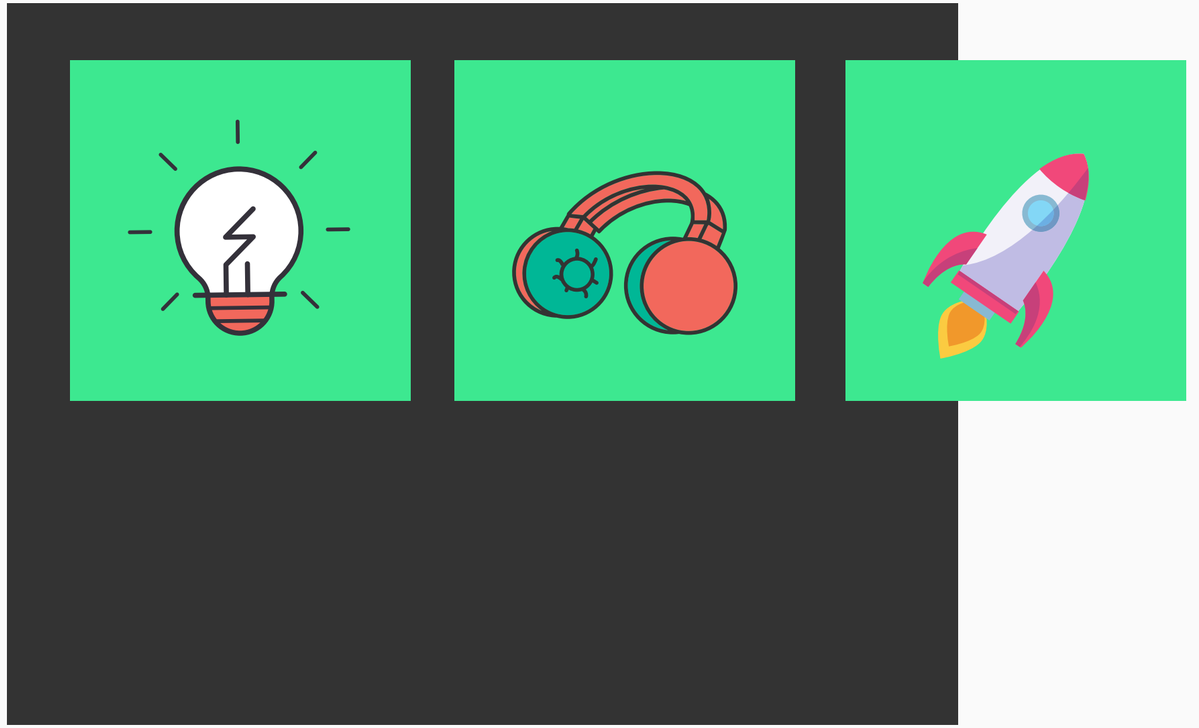
- *Contenedor Padre (Flex Container).*
- *Elementos Hijos (Flex Items).*
- *Eje Principal (Main Axis).*
- *Eje Transversal (Cross Axis).*

Flexbox

Primero especifica si los elementos ***"hijos"*** *son obligados a permanecer en una misma línea o pueden fluir en varias líneas.*

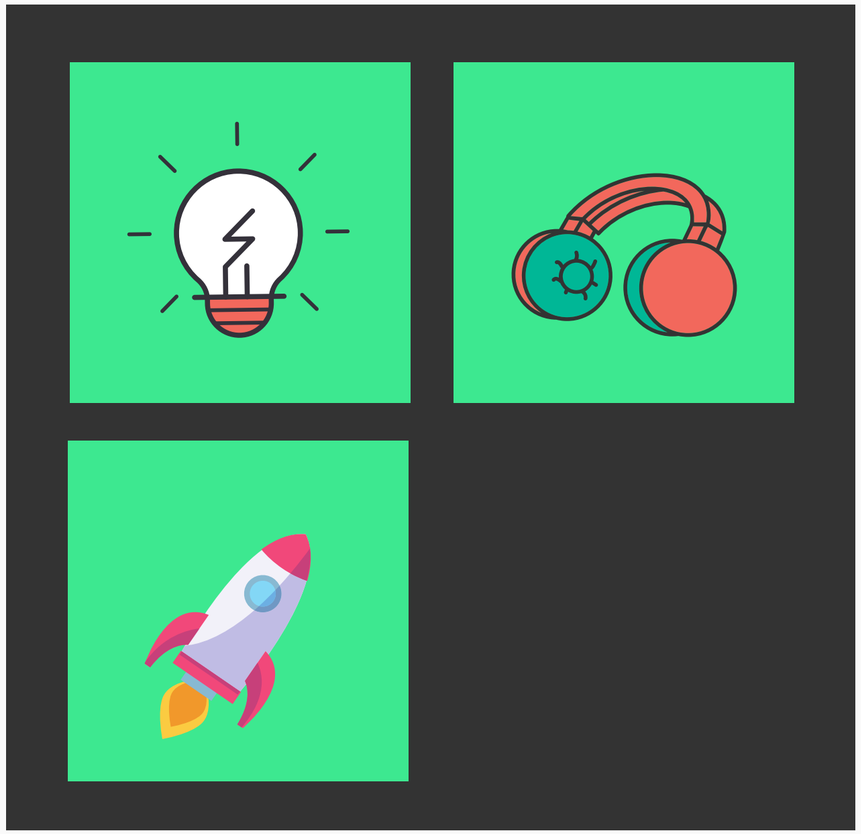


flex-wrap: nowrap *(por defecto)*

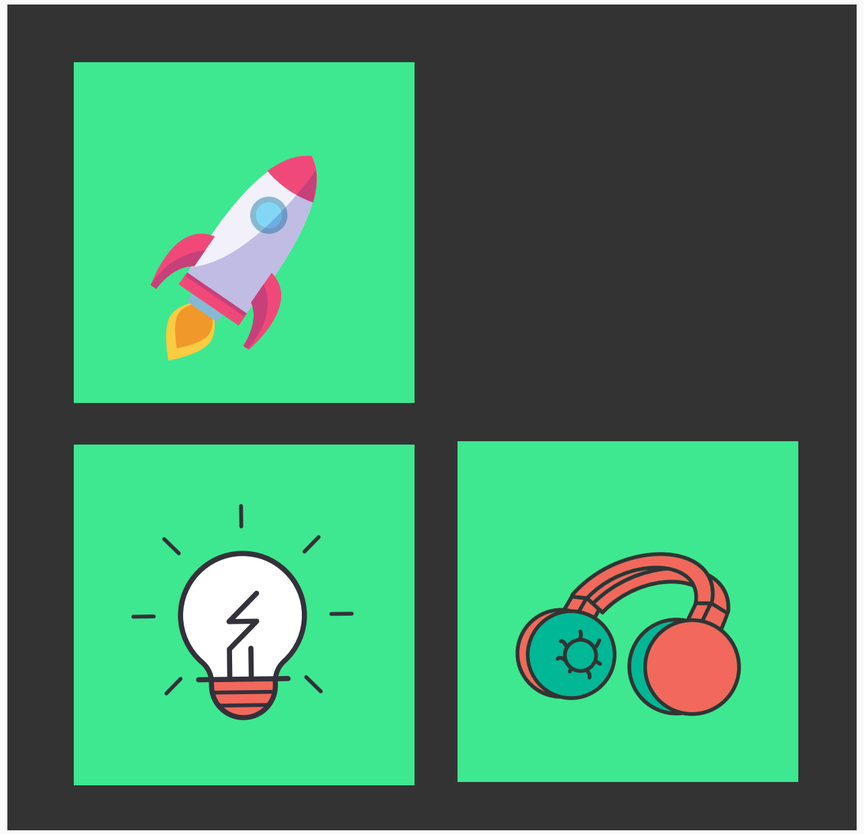


Flexbox

flex-wrap: wrap



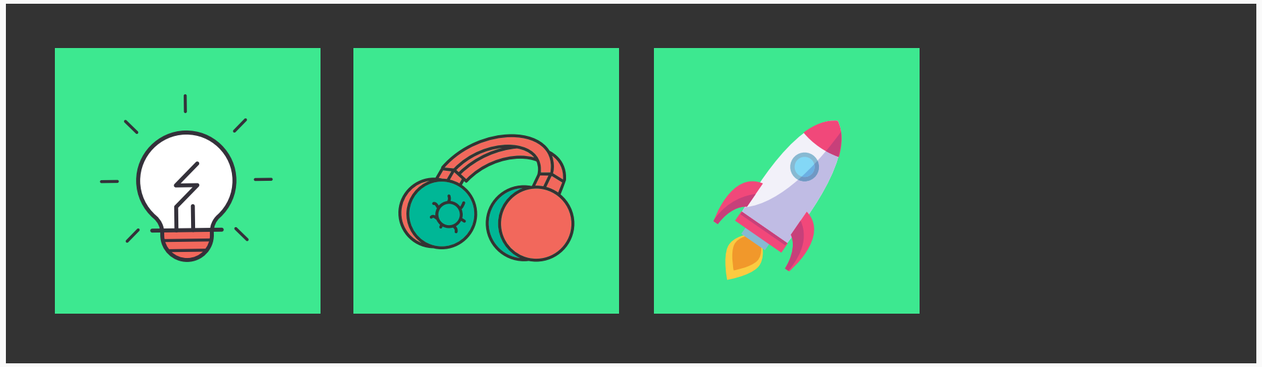
flex-wrap: wrap-reverse



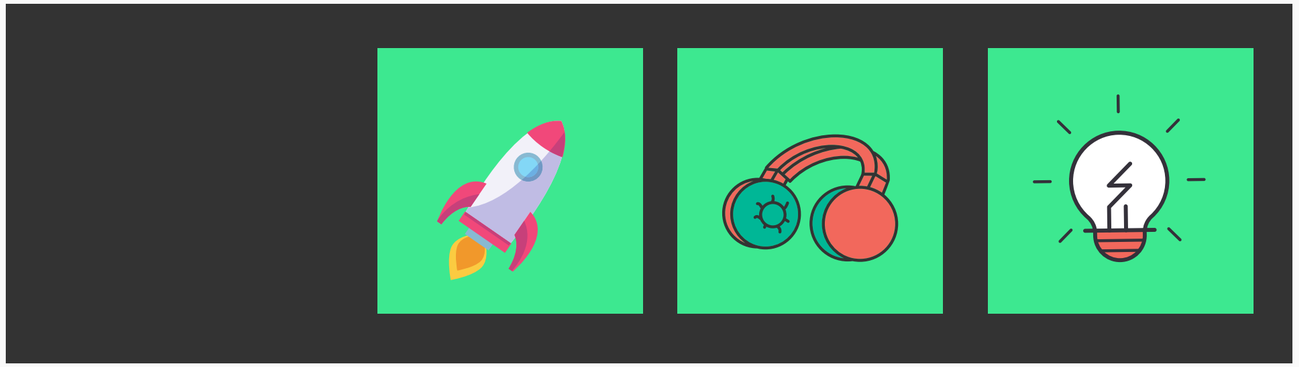
Flexbox

Podemos crear filas o columnas con flexbox
por defecto se creara una fila

flex-direction: row

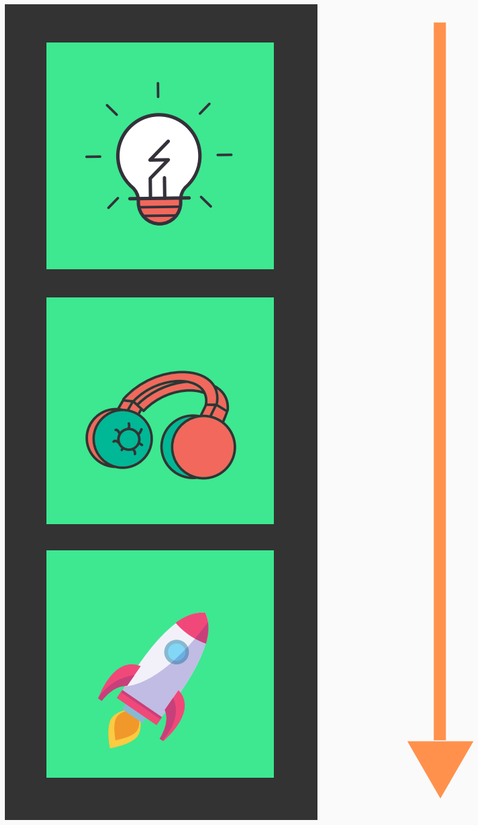


flex-direction: row-reverse

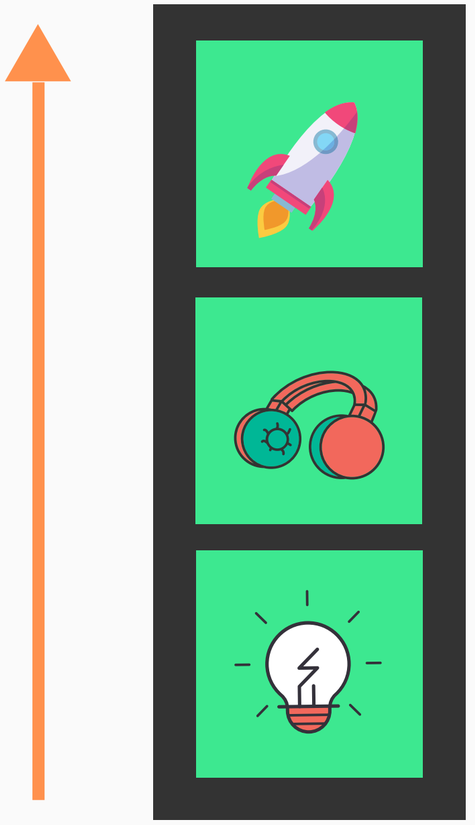


Flexbox

flex-direction: column



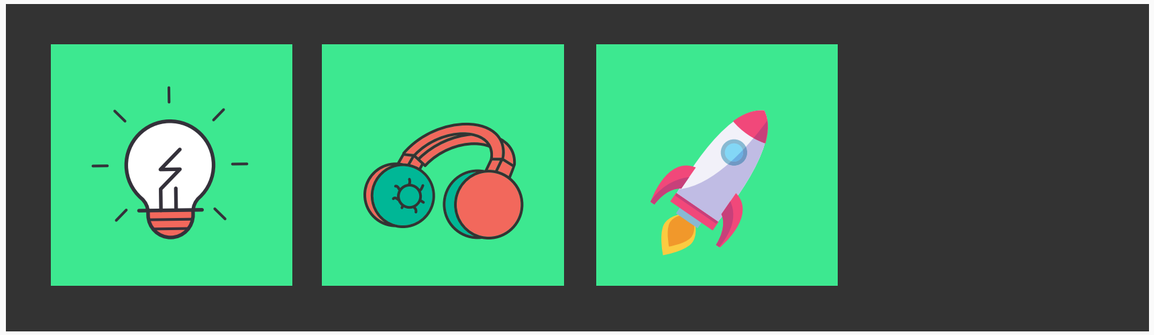
flex-direction: column-reverse



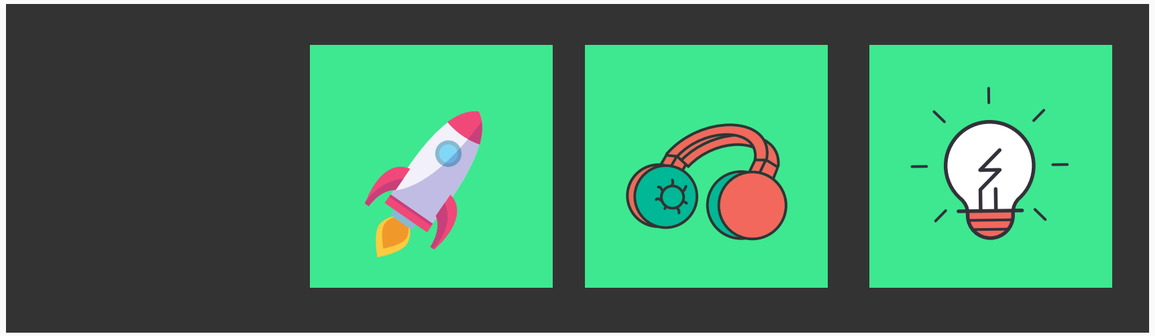
Flexbox

*Podemos alinear el contenido en el eje principal **(main-axis),** de diferentes formas*

justify-content: flex-start *(por defecto)*



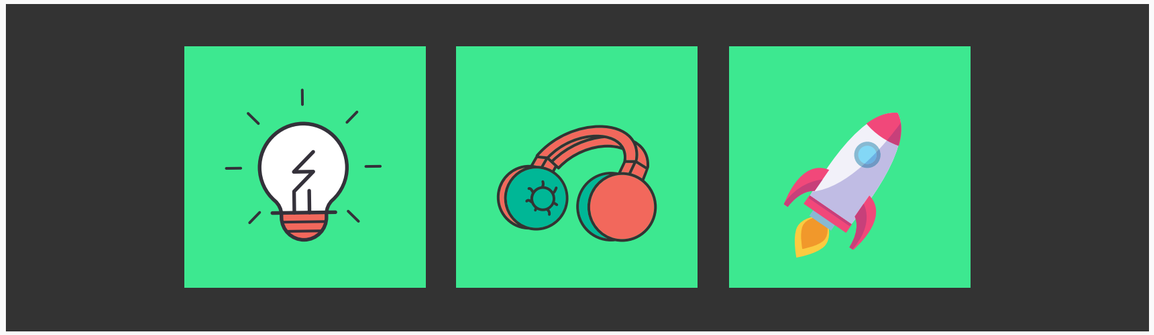
justify-content: flex-end



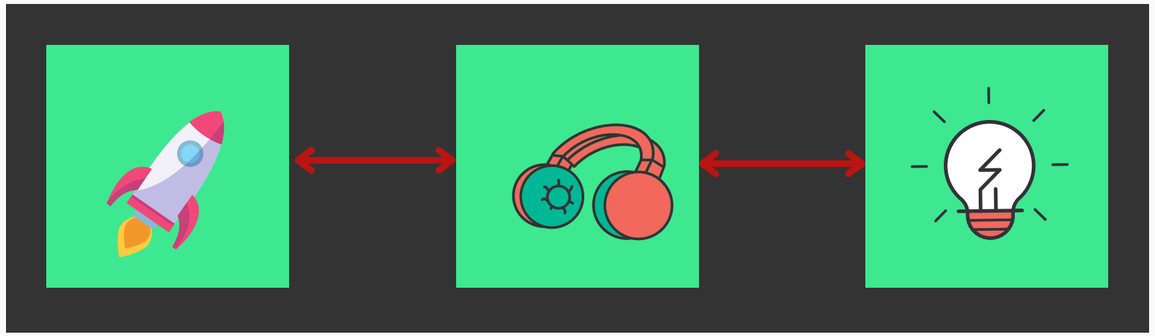
Flexbox

Podemos alinear el contenido en el eje principal ***(main-axis),*** *de diferentes formas*

justify-content: center



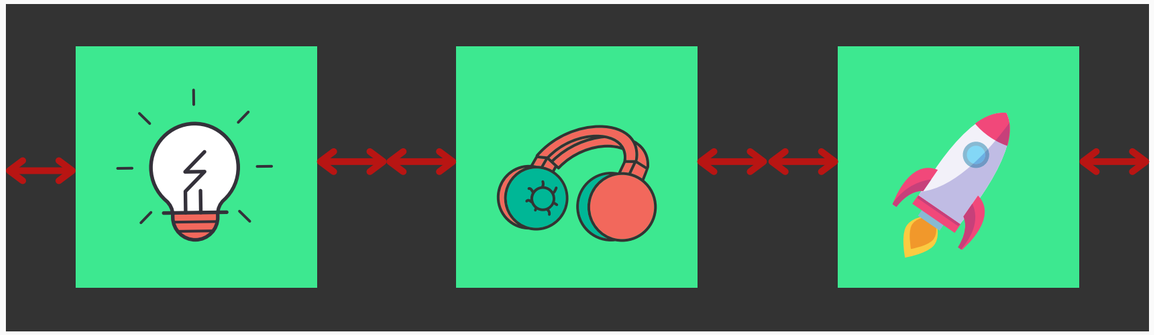
justify-content: space-between



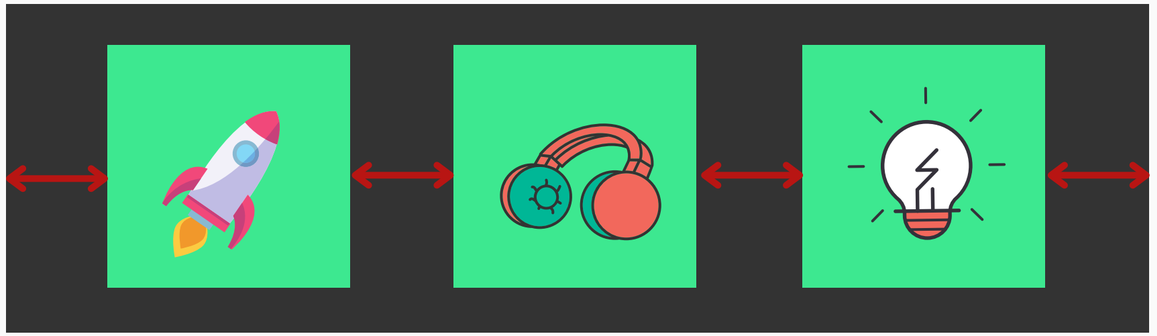
Flexbox

Podemos alinear el contenido en el eje principal ***(main-axis),*** *de diferentes formas*

justify-content: space-around



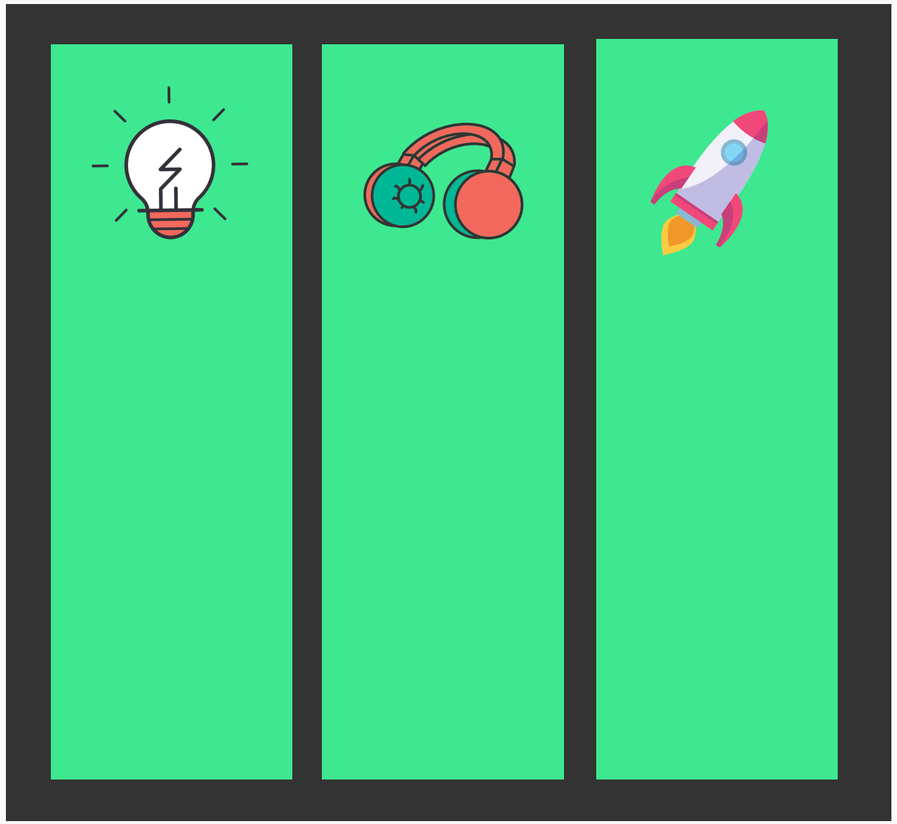
justify-content: space-evenly



Flexbox

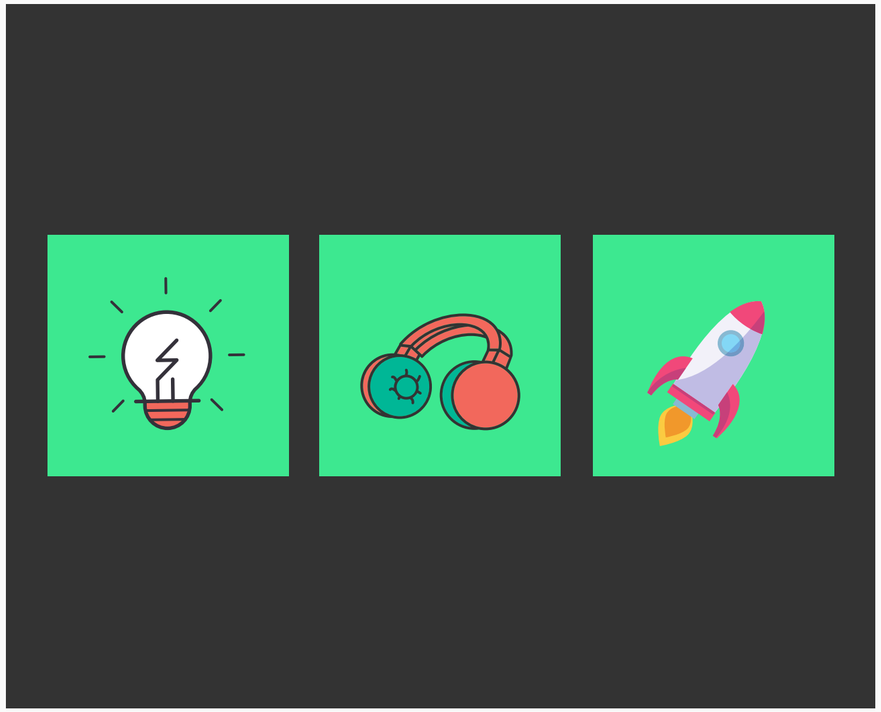
Podemos alinear el contenido en el eje transversal ***(cross-axis),*** *de diferentes formas*

align-items: strecth *(por defecto)*



Flexbox

align-items: center



align-items: baseline