

# Introducción a Flexbox

**DEV.F**  
DESARROLLAMOS(PERSONAS);

dev

# Contexto Histórico

Find It

Product Information

Customer Support

Technology & Research

Developer World

Groups & Interests

Resources Online

About Apple

Apple Sites Worldwide

Switzerland

Taiwan

Turkey

UK & Ireland

United States

JULY 14 1997



## Welcome to Apple



### Introducing CyberDrive

Register today for a free CD-ROM.

#### EMATE 300

Mobile, Affordable, & Smart



#### MOVIES FROM MARS

QuickTime VR Takes You Out of this World



What's Hot

#### Preorder Mac OS 8

Now you can [preorder Mac OS 8](#), described by Macworld as "the most comprehensive update to the Mac OS in years, sporting a bold new look, a speedier Finder, more

#### Be the First to Know

Learn about new Macintosh software releases the moment they become available. Check [Hot Mac Products](#) to hear about programs like Speed Demon, ReBirth RB-338 and

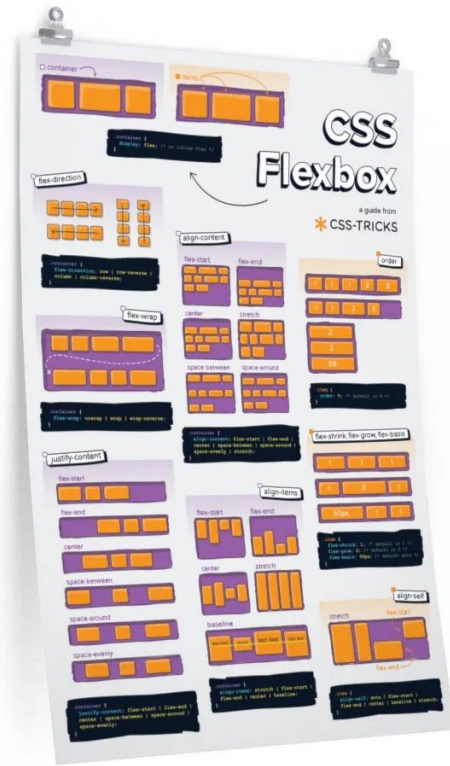
DEV.F

# Contexto Histórico



# ¿Qué es Flexbox?

Flexbox CSS es un modelo/método de diseño unidimensional utilizado para diseñar elementos (elementos HTML) de **forma de fila o columna** (horizontal o vertical).



# FLEX CONTAINER & FLEX ITEMS

## Flex Container & Flex Items

Flex Container (parent)

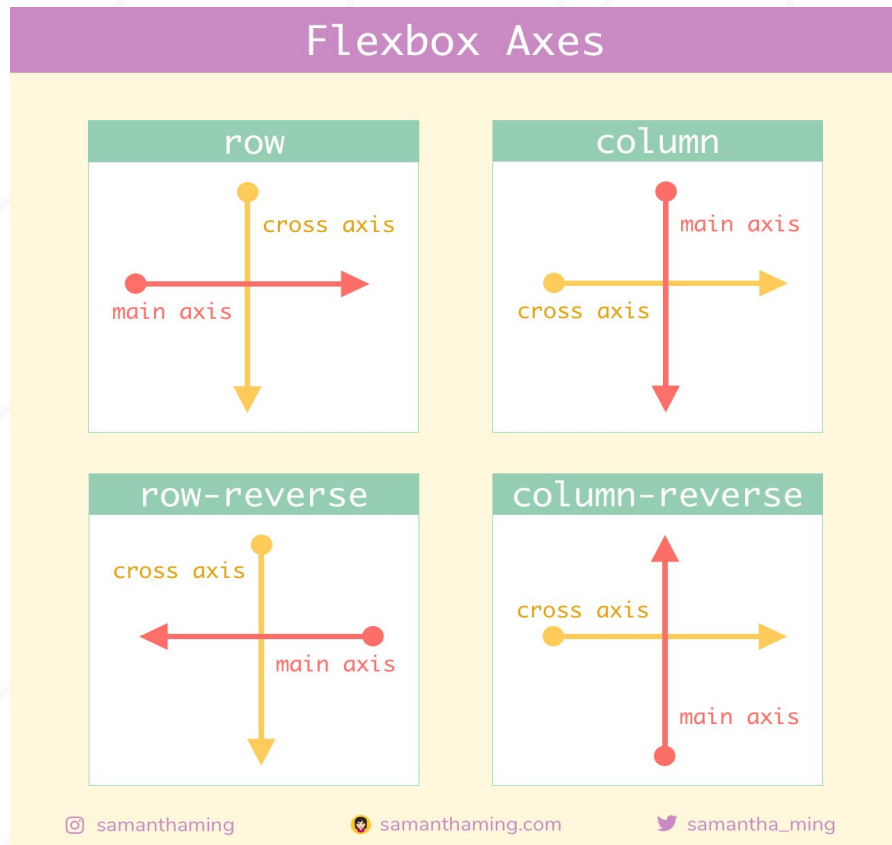
Flex Items (children)



Para que Flexbox funcione, debe configurar la relación **Padre-Hijo**. El padre es el contenedor flexible, y todo lo que hay dentro de él son los elementos secundarios o flexibles.

# EJES FLEXBOX

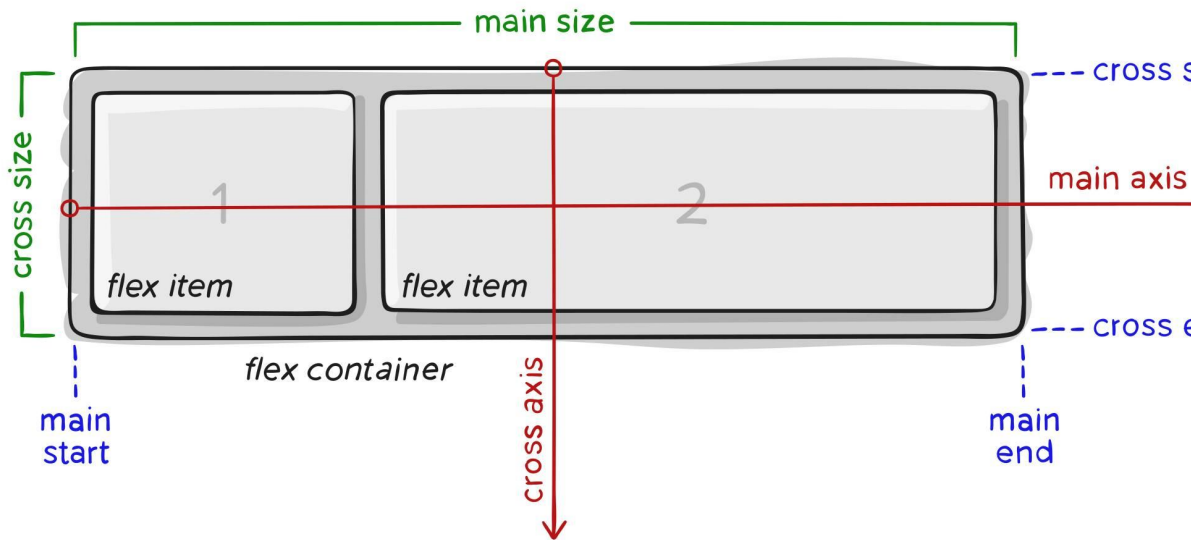
Recuerden en la clase de matemáticas, nos enseñaron **eje x e y**. Bueno, aquí no hacemos eso. Porque el **eje principal** puede ser horizontal o vertical.



# ¿Qué es Flexbox?

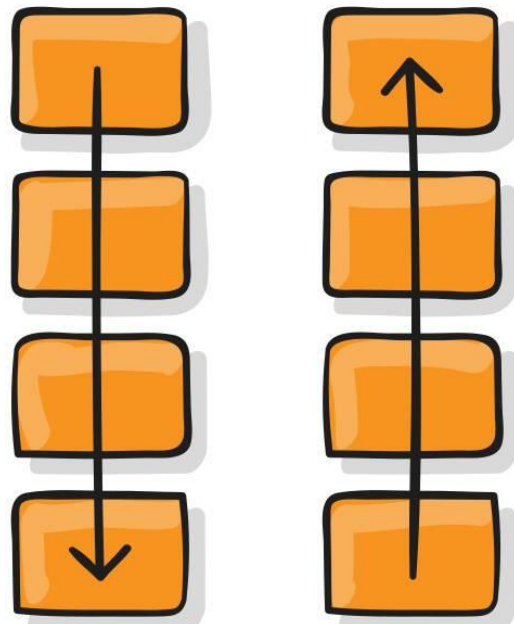
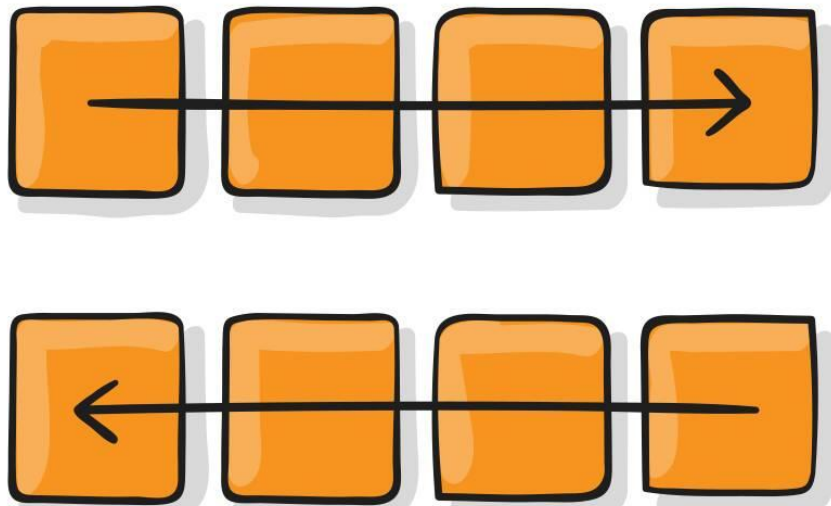
## *Flexible Boxes*

En cada eje, hay un inicio y un final. Si está en el **eje principal**, la posición inicial se denomina **inicio principal** y si la posición final se denomina **extremo principal**.





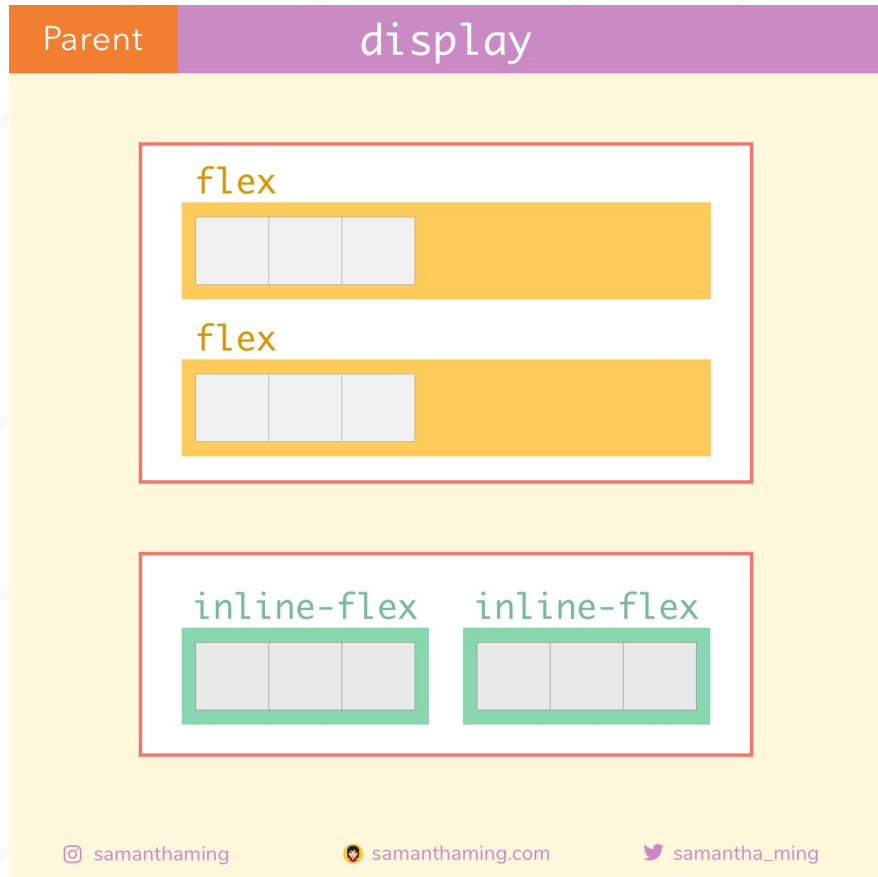
# Propiedad flex-direction





# DISPLAY / MONITOR

Para iniciar esta **fiesta flexbox**, primero necesitamos crear nuestro contenedor flexible. Esto se hace aplicando flex a la propiedad de visualización en el elemento primario. ¡Bam! Ahora todos sus hijos inmediatos se convertirán en artículos flexibles



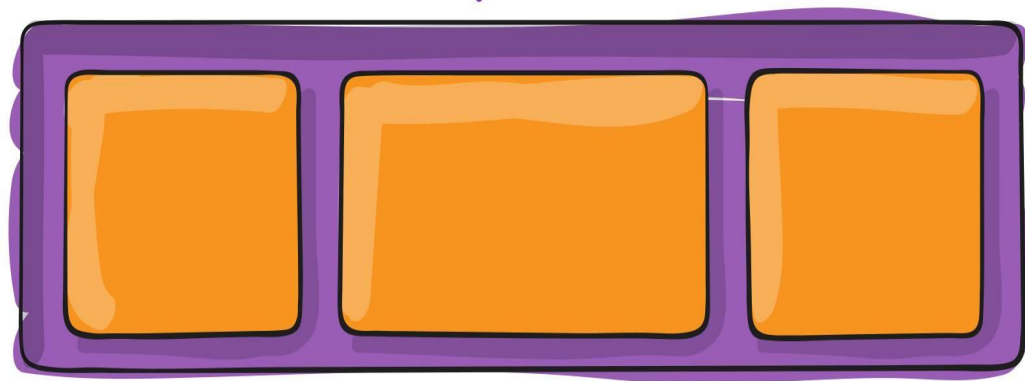
# Contenedor padre

Flexbox Properties

Parent	Child
<b>flex-direction</b>	order
<b>flex-wrap</b>	flex-grow
<b>flex-flow</b>	flex-shrink
<b>justify-content</b>	flex-basis
<b>align-items</b>	flex
<b>align-content</b>	align-self

[? samantha\\_ming](#) [? samanthaming](#) [? hi.samanthaming](#)

container



# Elementos hijos

## Flexbox Properties



Parent

flex-direction

flex-wrap

flex-flow

justify-content

align-items

align-content

Child

order

flex-grow

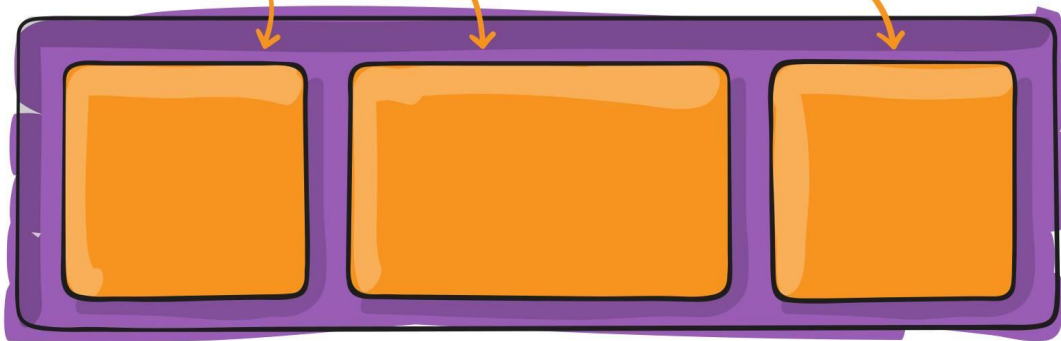
flex-shrink

flex-basis

flex

align-self

items



# FLEXBOX CHEATSHEET (HOJA DE TRUCOS)

## Flexbox Cheatsheet

### DIRECTION

**flex-direction**  
**flex-wrap**  
**flex-flow**  
**order**

### ALIGNMENT

**justify-content**  
**align-items**  
**align-content**  
**align-self**

### SIZE

**flex-grow**  
**flex-shrink**  
**flex-basis**  
**flex**

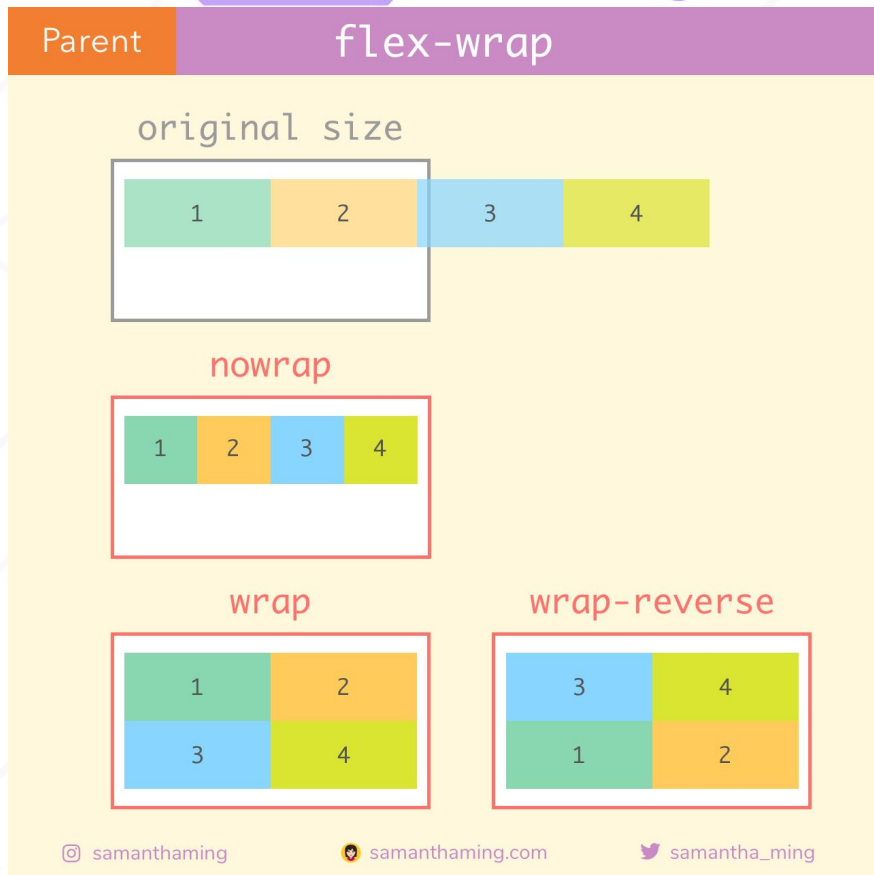


PARENT



CHILD

# Propiedad flex-wrap

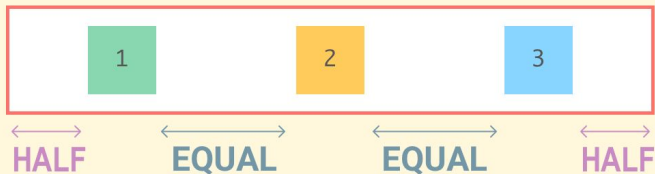


**Esta propiedad es lo que permitirá que los elementos flexibles del contenedor ocupen más de una línea.**

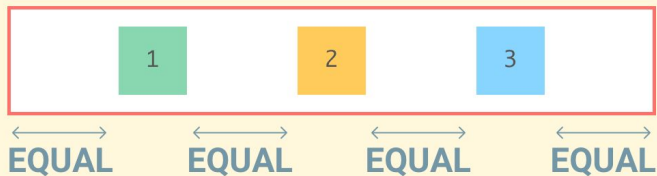
# Space-around vs space-evenly

## space-around vs space-evenly

### space-around



### space-evenly



# Order

Child

order

1	2	3	4
order:0	order:0	order:0	order:0

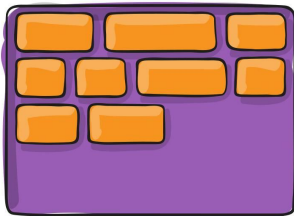
1	3	4	2
order:0	order:0	order:0	order:1

2	1	3	4
order:-1	order:0	order:0	order:0

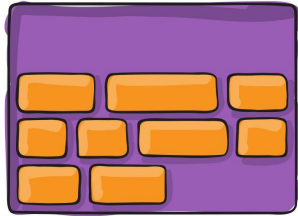


# Propiedad align-items

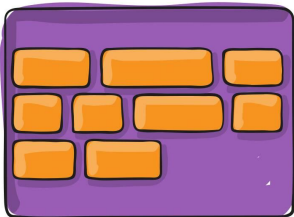
flex-start



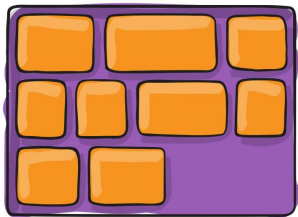
flex-end



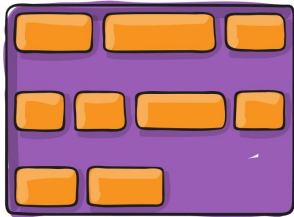
center



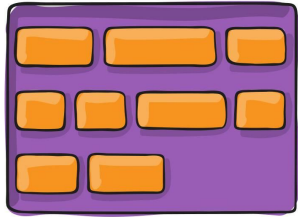
stretch



space-between



space-around

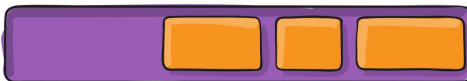


# Propiedad justify-content

flex-start



flex-end



center



space-between



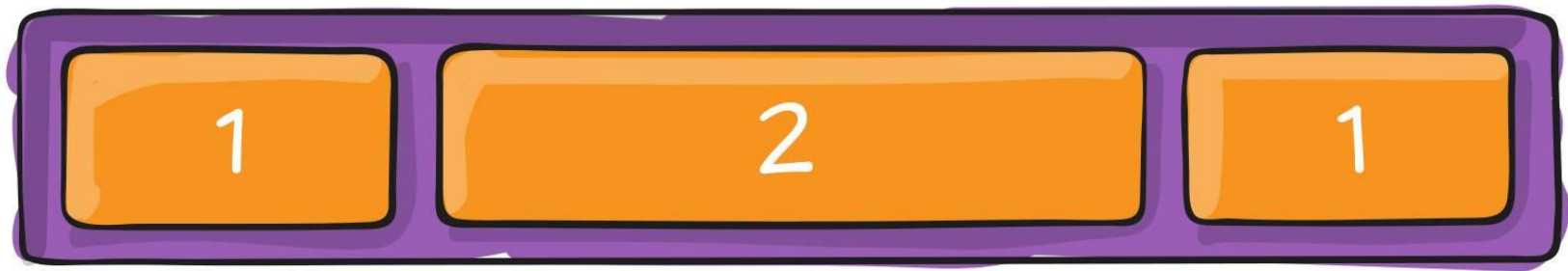
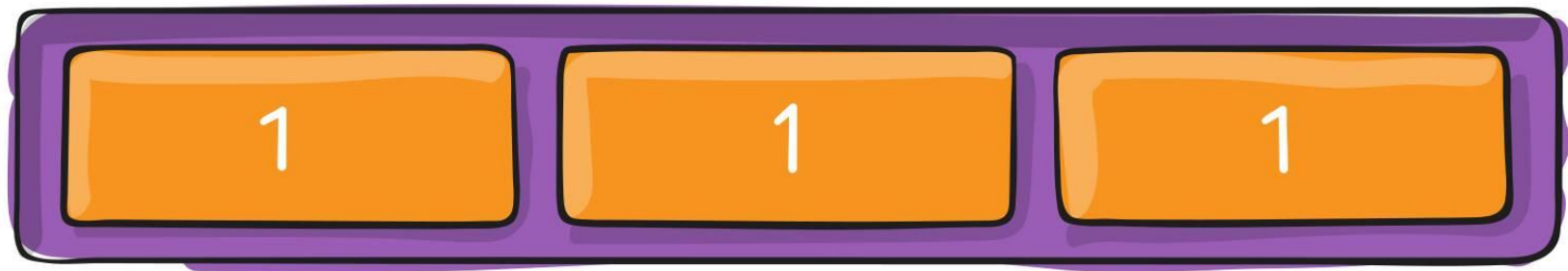
space-around



space-evenly



# Propiedad flex-grow



# Propiedad flex-shrink

Opuesta a flex-grow.

Mientras que la anterior indica un factor de crecimiento, flex-shrink aplica un factor de decrecimiento.

Los ítems que tengan un valor numérico más grande, serán más pequeños, mientras que los que tengan un valor numérico más pequeño serán más grandes.

# Actividad: Mi portafolio personal con Flexbox

**DEV.F**  
DESARROLLAMOS(PERSONAS);

dev