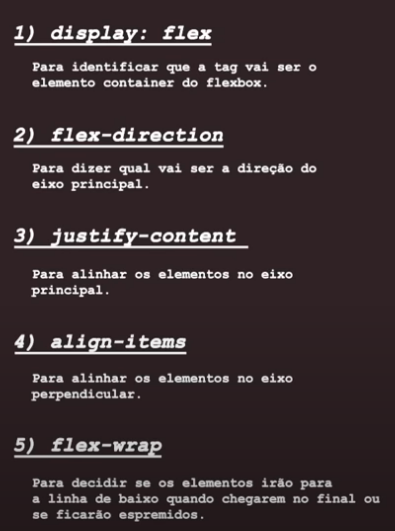
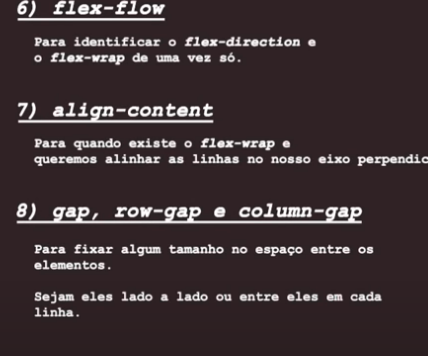
**FLEXBOX CSS! Como posicionar elementos na página web - parte 1**

Flexbox = flexible box – caixa flexivel

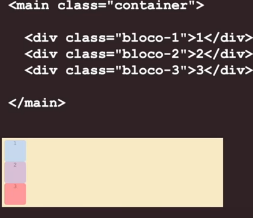
Previa ou resumo de tudo, caso queira mais detalhes veja o conteúdo de cada propriedade que eu expliquei.





.

Para comecarmos nos temos que ter uma tag pai que e utilizado como um container para guardar as tags filho.

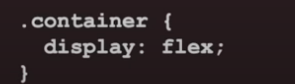


No exemplo acima nos temos um container e dentro dele temos 3 tags filhas.

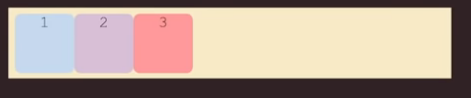
.

A primeira coisa que devemos fazer para identificar onde usaremos o flexbox e olha qual sera a tag container que vai armazenar as tags filhas.

E após isso colocamos o **display: flex;** flex de flexbox



So ai já ocorre a primeira mudança, as tags filhas vao se organizar lado a lado dentro do container, ficam alinhadas horizontalmente



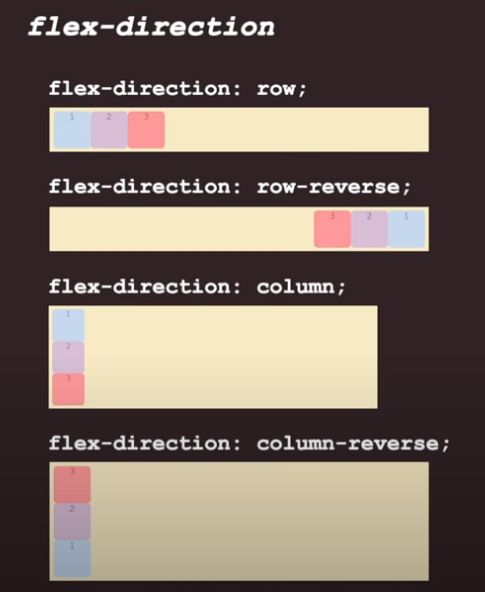
Antes nosso eixo principal era o vertical, agora se tornou o horizontal.

.

## **1 propriedade**

**flex-direction:** decide qual vai ser o eixo principal dos nossos elementos.

Por padrão quando colocamos o **display: flex;** o eixo padrão se torna o **row** (horizontal).



## **2 propriedade**

**justify-content:** serve para alinha o conteúdo ao longo do nosso eixo principal.

**Neste exemplos usaremos o flex-direction: row; que e o eixo horizontal.**



**Flex-start:** agrupa os elemtnos no inicio.

**Flex-end:** agrupa os elementos no final.

**Center:** agrupa os elementos no centro

**Space-between:** cria um epaco entre os elemtnos deixando um no inicio, um no meio, e outro no final dentro do container.

**Space-around:** deixa um espaço não so entre os elementos mas do lado de fora também

.

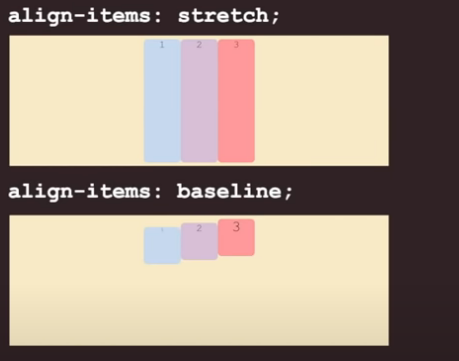
## **3 propriedade align -items**

**Neste exemplos usaremos o flex-direction: row; que e o eixo horizontal e também o justify-content:center; para criar nossos exemplos, mas podemos usar os outros de acordo com nossa preferencia.**

****

Como podemos ver o Align-items modifica verticalmente a posição dos objetos dentro de um container.

Temos o flex start que deixa em cima que e o padrão, center que deixa centralizado no meio, eo flex-end que deixa no final (embaixo).



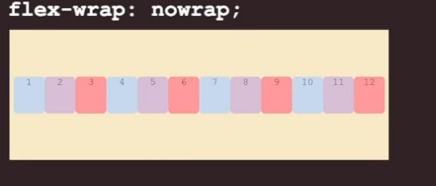


O streetck que e outra propriedade que ocupa todo o espaço vertical da tag container,

Eo baseline que podemos ver que os três números dos blocos tem tamanhos diferentes, o baseline não deixa os blocos na mesma altura mas sim o conteúdo dentro dela, entao a base do 1 2 e 3 estao na mesma altura.

## **4 propriedade flex-wrap**

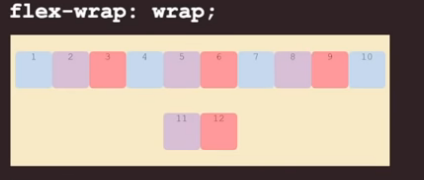
No padrão os elementos que estamos alinhando com flexbox, eles tentam caber em uma linha so, e ate mesmo se espremem para caber, essa propriedade se chama **nowrap**



.

.

Se mudarmos o valor para **wrap** daremos a possibilidade do flexbox descer uma linha caso o elemento já chegue ao final.



.

E também podemos fazer um **wrap-reverse** que invés de caso o elemento chegar ao final da linha quebrar para baixo ele ira quebrar para cima.



.

.

## **5 propriedade flex flow**

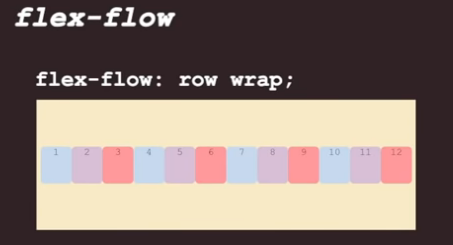
O flex-flow juntara o **flex-direction** e o **flex-wrap.**

**Por exemplo**

**flex-flow: row wrap;**

**ou**

**flex-flow: colum nowrap;**

****

.

## **6 propriedade align-content**

O align-content e para alinhar os elementos em mais de uma linha sempre que usarmos o wrap, isso e caso usemos o flex-wrap: wrap; e queiramos alinha as duas linhas.

.



O flex-start deixa as duas linhas no inicio (subir pra cima e ficarão agrupadas.

.



Alinha as duas linhas em baixo, (no final)

.



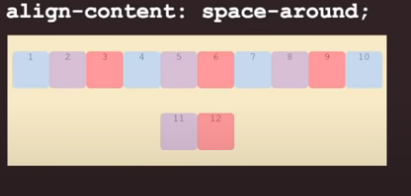
Vai agrupar as duas linhas no centro.

.



Vai colocar um espaço grande entre as linhas, com o maior espaço possível..

.



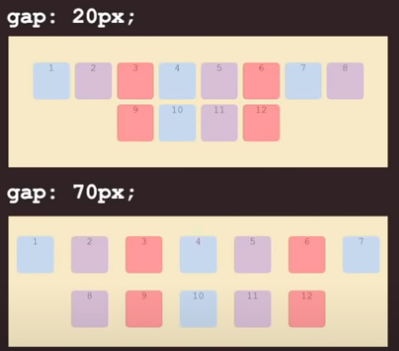
Cria um espaço entre as linhas e do lado de fora das linhas, como se fosse uma margem.

.

## **7 propriedades**

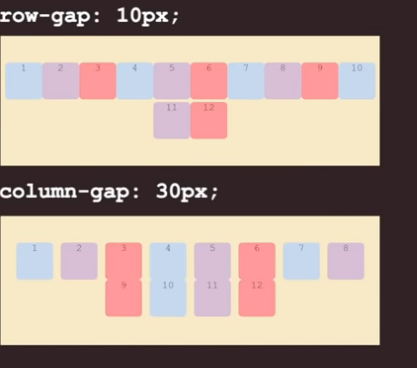
Caso você queira definir separar os elementos mais especificamente, especificar um tamanho para separalos invés de usar as propriedades que já vimos.

.



Cria um espaço entre eles apenas usando gap e a medida que você queira.

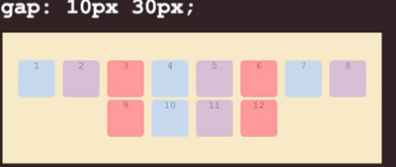
E caso você queira diferenciar a medida vertical e horizontal você pode colocar



row gap cria um espaco entre linhas (vertical), e column-gap cria um espaço entre as colunas (horizontal).

.

E caso você queira colocar os dois em uma so propriedade so usar



gap: (vertical) (horizontal)