

POSICIONANDO ELEMENTOS COM FLEXBOX EM CSS

O flex box foi projetado como um modelo de layout unidimensional e como um método que pode oferecer distribuição de espaço entre itens em uma interface e recursos de alinhamento.

Já o FLEX CONTAINER é a TAG que envolve os itens, será nela que iremos aplicar a propriedade "display: flex", que transforma todos os seus itens em flex itens.

ESTRUTURA BÁSICA DO FLEX DIRECTION

A flex-direction é basicamente a propriedade que estabelece o eixo principal do container, definindo assim, a direção que os flex itens serão colocados no flex container.

Os dois eixos principais que temos são: linha (horizontal) e coluna (vertical).

Os tipos de eixos:

- **row** (padrão): em direção ao texto, da esquerda para a direita.
- **row-reverse** (oposto): sentido oposto à direção do texto.
- **column**: ordenação de cima para baixo, em coluna única.
- **column-reverse**: ordenação inversa, de baixo para cima.

FLEX-WRAP

É a propriedade que define se os itens devem ou não quebrar a linha.

Por padrão eles não quebram linhas, isso faz com que os flex itens sejam compactados além do limite do conteúdo.

- **nowrap**: é o padrão, não permite a quebra de linha e deixa o conteúdo "vazado"
- **wrap**: surge para permitir a quebra de linha assim que um dos flex itens não puder mais ser compactado.

OBS: cuidado para tratar o espaçamento do contêiner.

- **wrap-reverse**: permite a quebra de linha assim que um dos flex itens não puder mais ser compactado, porém essa quebra ocorre para cima da linha.

FLEX-FLOW

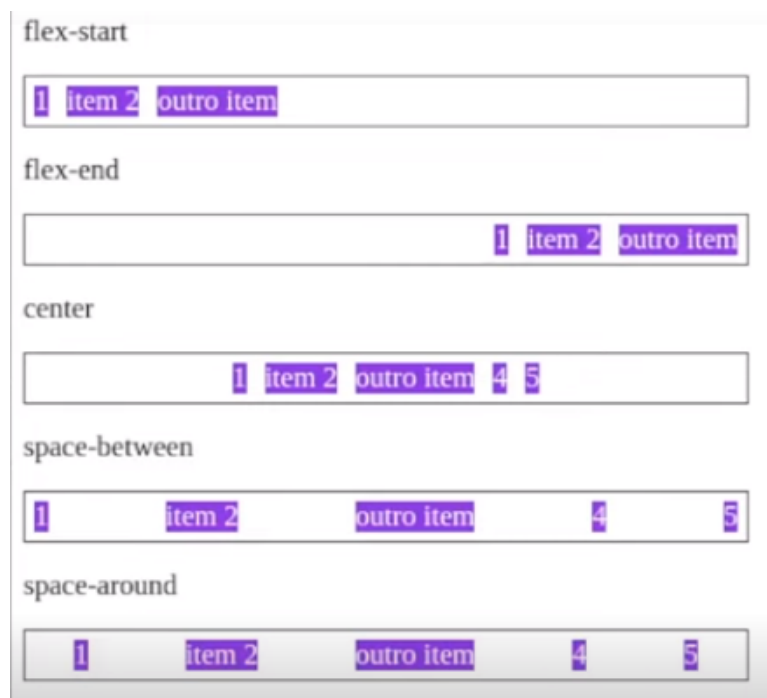
É um atalho para as propriedades flex-direction e flex-wrap.

JUSTIFY-CONTENT

Ela vai se encarregar de alinhar os itens dentro do container de acordo com a direção pretendida e tratar da distribuição de espaçamento entre eles.

As suas variações do JUSTIFY-CONTENT serão:

- **flex-start:** alinhamento no início do container.
- **flex-end:** final do container.
- **center:** elementos no centro do container
- **space-between:** cria um espaçamento igual entre os elementos.
- **space-around:** os espaçamentos do meio são duas vezes maiores que o início e final.



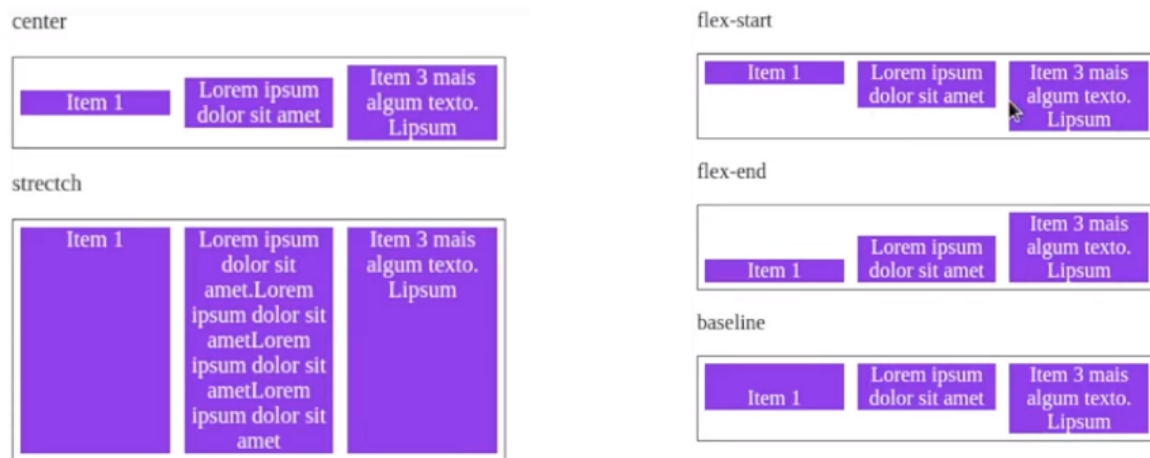
ALIGN ITENS

Trata do alinhamento dos flex itens de acordo com o eixo do container. Esse alinhamento é diferente para quando os itens estão em colunas ou linhas.

Permite o alinhamento central no eixo vertical. Além disso, ele possui algumas variações:

- **center:** alinhamento dos itens ao centro
- **stretch:** padrão, e os flex itens cresçam igualmente
- **flex-start:** alinhamento dos itens no início

- **flex-end**: alinhamento dos itens no final
- **baseline**: alinhamento de acordo com a linha base da tipografia dos itens



ALIGN CONTENT

É a propriedade responsável por tratar o alinhamento das linhas do container em relação ao eixo vertical do container.

Tipos de Alinhamento:

- **center**: alinhamento dos itens ao centro
- **stretch**: é o padrão e os flex itens crescem igualmente
- **flex-start**: alinhamento dos itens no início
- **flex-end**: alinhamento dos itens no final
- **space-between**: cria um espaçamento igual entre os elementos
- **space-around**: os espaçamentos do meio são duas vezes maiores que o inicial e final

FLEX-GROW

Define a proporcionalidade de crescimentos dos itens, respeitando o tamanho de seus conteúdos internos.

OBS: não irá funcionar caso tenhamos adicionado justify-content ao nosso flex container

FLEX-SHRINK

É a propriedade que estabelece a capacidade de redução ou compressão do tamanho de um item.

ALIGN-SELF

É a propriedade que estabelece o alinhamento de modo individual sobre um determinado item. Valores possíveis são:

- **auto**: valor padrão, irá respeitar a definição de align-items do container

- **flex-start:** ao início do container
- **flex-end:** ao final do container
- **center:** relativo ao centro de acordo com o eixo
- **stretch:** ocupa todo os espaço relativo
- **baseline:** utiliza a linha base da tipografia

FLEX-BASIS

É a propriedade que estabelece o tamanho inicial do item antes das distribuições de espaço restante dentro dele, usando como base o conteúdo interno disposto.

É como se ela pegasse o tamanho mínimo que esse item precisa ter e depois vai fazer a distribuição da sobra de espaço.

Portanto, se só uma das <div> de conteúdo tiver a class que inclua o flex-basis, apenas essa classe vai ficar do tamanho necessário para ocupar os espaços vazios do container.

FLEX-SHRINK

Essa propriedade estabelece a capacidade de redução ou compressão do tamanho de um item.

FLEX

É um atalho ou abreviação de escrita para as propriedades: grow, shrink e basis.

Exemplo:

```
.flex-1 {  
    flex: 1;  
  
    /* grow = 1, shrink = 1, basis = 0 */  
}
```