

## Tópicos especiais em Programação - Flex Box

Aluno: Augusto Hirt Farias (111180)

Projeto: <https://github.com/augusto-kuki/combustivel-inmetro>

FlexBox é uma ferramenta para criar o visual de páginas web/aplicativos, independente do dispositivo em que o código esteja sendo executado, sem necessidade de criar diferentes versões para os diferentes dispositivos, tamanhos de tela ou orientação (retrato ou paisagem), sendo assim, descartando a utilização de valores fixos ao posicionar elementos na tela, deixando tudo dinâmico. O FlexBox também permite criarmos heranças de estilos, passando para componentes filhos ou reutilizando os estilos e adicionando novos parâmetros.

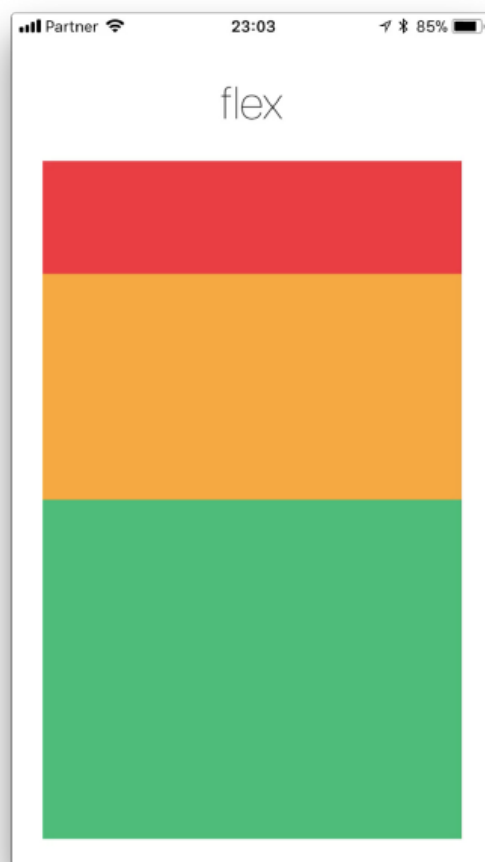
Um componente pode especificar o layout de seus filhos usando o algoritmo Flexbox. O Flexbox foi projetado para fornecer um layout consistente em diferentes tamanhos de tela.

Você normalmente usará uma combinação de `flexDirection`, `alignItemse` `justifyContent` para alcançar o layout certo.

## FLEX

Definirá como seus itens serão "preenchidos" sobre o espaço disponível ao longo do seu eixo principal. O espaço será dividido de acordo com a propriedade flex de cada elemento.

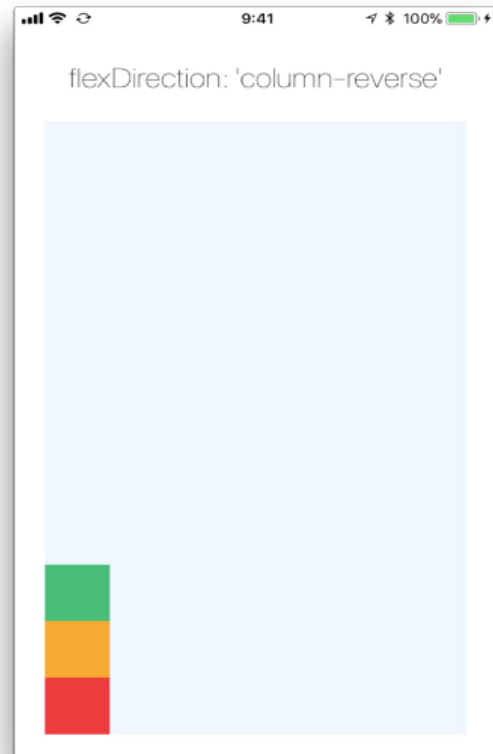
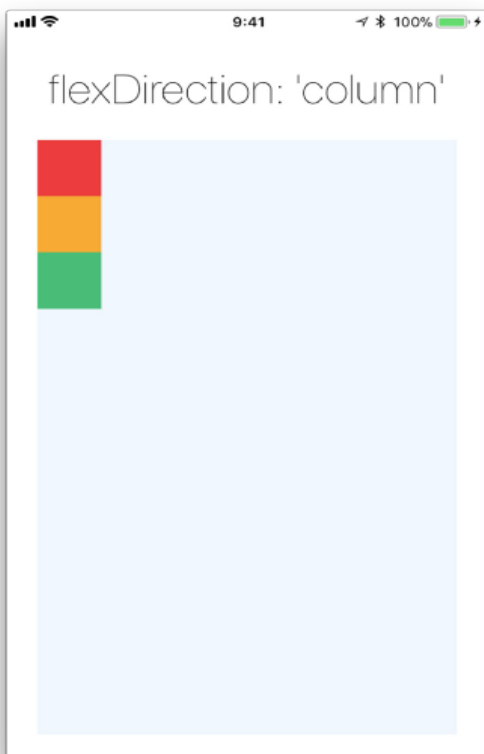
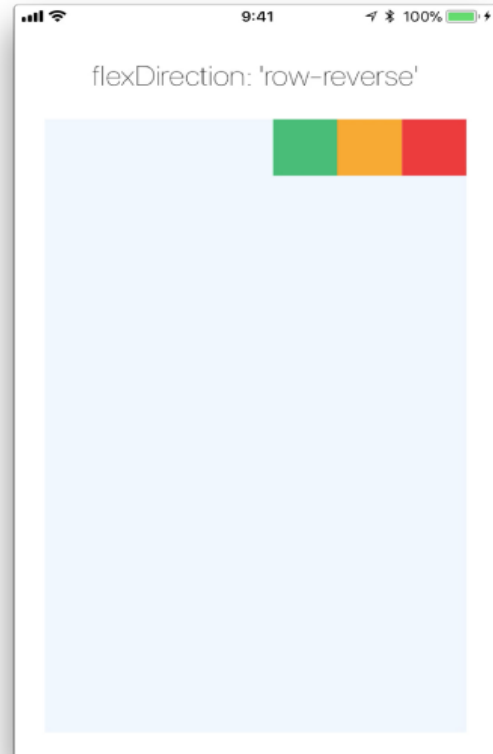
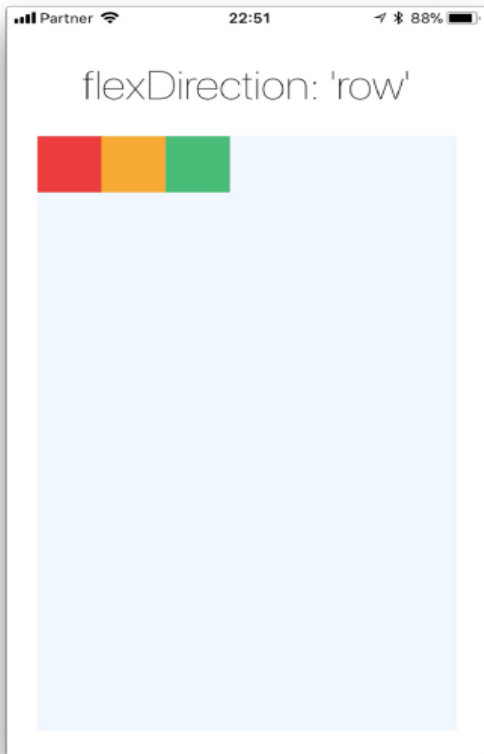
No exemplo a seguir, as visualizações em vermelho, amarelo e verde são todas filhas na visualização do contêiner flex: 1 configurada. A exibição em vermelho usa flex: 1, a exibição em amarelo flex: 2 e a exibição em verde flex: 3.  $1 + 2 + 3 = 6$ , o que significa que a visualização em vermelho ficará  $1/6$  do espaço, o amarelo  $2/6$  do espaço e o verde  $3/6$  do espaço.



## Direção Flex

`flexDirection` controla a direção na qual os filhos de um nó são dispostos. Isso também é chamado de *eixo principal*. O eixo transversal é o eixo perpendicular ao eixo principal ou o eixo no qual as linhas de disposição são dispostas.

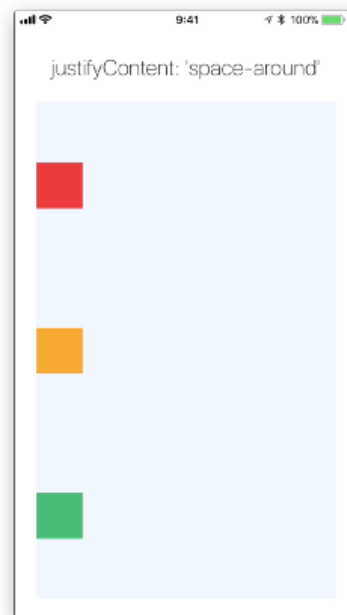
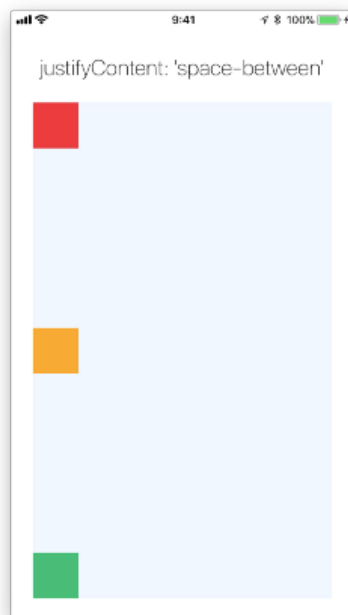
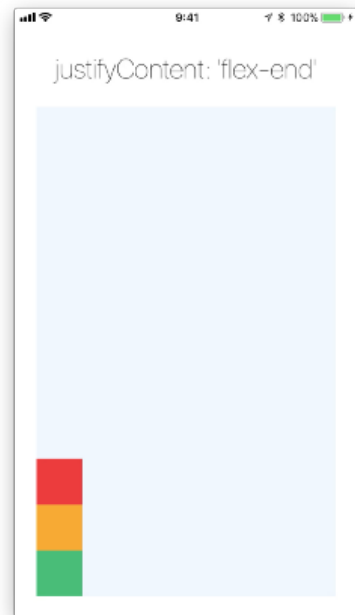
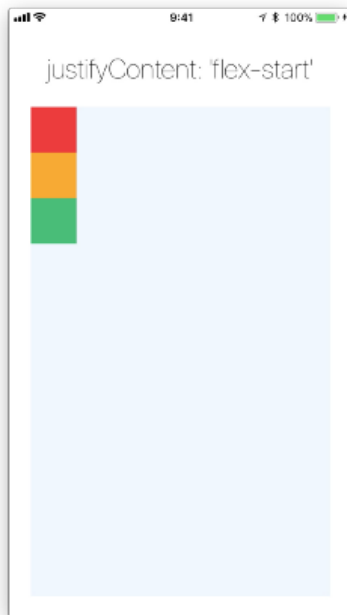
- `row` Alinhe os filhos da esquerda para a direita. Se a disposição estiver ativada, a próxima linha começará sob o primeiro item à esquerda do contêiner.
- `column`( valor padrão ) Alinhar filhos de cima para baixo. Se a disposição estiver ativada, a próxima linha começará à direita do primeiro item na parte superior do contêiner.
- `row-reverse` Alinhe os filhos da direita para a esquerda. Se a disposição estiver ativada, a próxima linha começará sob o primeiro item à direita do contêiner.
- `column-reverse` Alinhe os filhos de baixo para cima. Se a disposição estiver ativada, a próxima linha começará à direita do primeiro item na parte inferior do contêiner.



## Justificar conteúdo

`justifyContent` descreve como alinhar os filhos no eixo principal do contêiner. Por exemplo, você pode usar essa propriedade para centralizar um filho horizontalmente em um contêiner com `flexDirection` definido como `row` ou verticalmente em um contêiner com `flexDirection` definido como `column`.

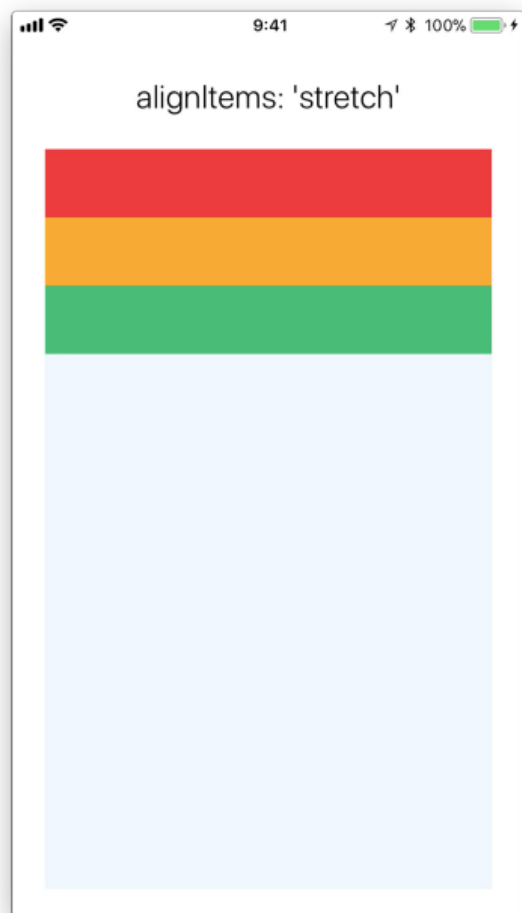
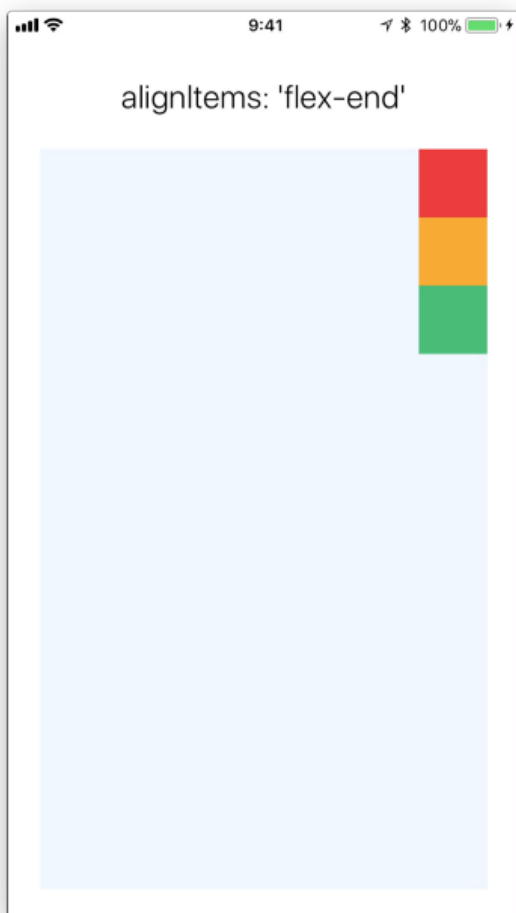
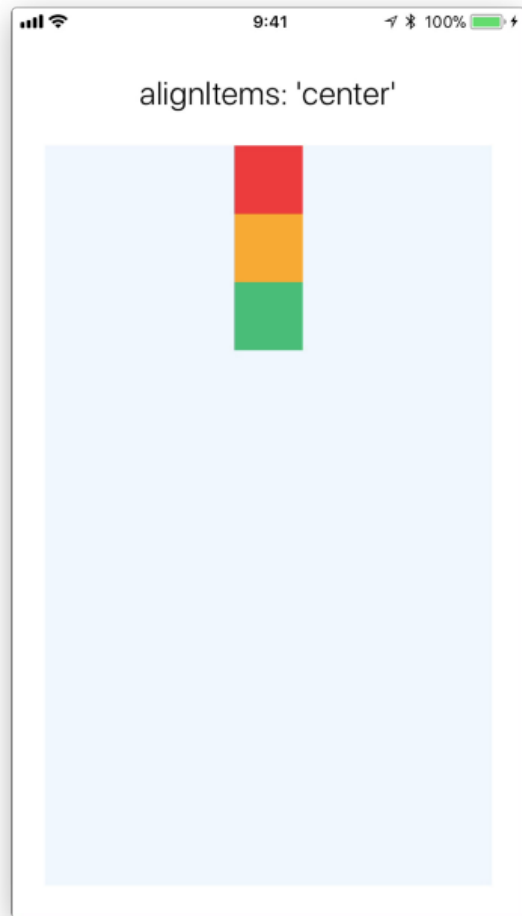
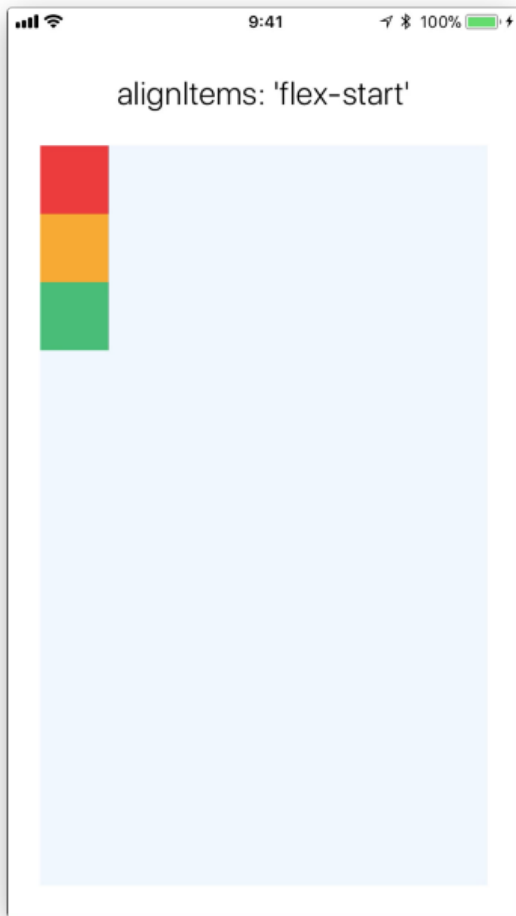
- `flex-start` (valor padrão) Alinhe os filhos de um contêiner ao início do eixo principal do contêiner.
- `flex-end` Alinhe os filhos de um contêiner no final do eixo principal do contêiner.
- `center` Alinhe os filhos de um contêiner no centro do eixo principal do contêiner.
- `space-between` Afastar uniformemente as crianças no eixo principal do contêiner, distribuindo o espaço restante entre as crianças.
- `space-around` Afastar uniformemente as crianças no eixo principal do contêiner, distribuindo o espaço restante ao redor das crianças. Comparado a `space-between`, o uso `space-around` resultará na distribuição do espaço no início do primeiro filho e no final do último filho.
- `space-evenly` Distribua uniformemente as crianças dentro do recipiente de alinhamento ao longo do eixo principal. O espaçamento entre cada par de itens adjacentes, a borda inicial principal e o primeiro item, e a borda final principal e o último item, são exatamente iguais.



## Alinhar itens

`alignItems` descreve como alinhar os filhos ao longo do eixo transversal de seu contêiner. Alinhar itens é muito semelhante, `justifyContent`, em vez de aplicar ao eixo principal, `alignItems` aplica-se ao eixo transversal.

- `stretch` ( valor padrão ) Estique os filhos de um contêiner para corresponder ao `height` transversal do contêiner.
- `flex-start` Alinhe os filhos de um contêiner ao início do eixo transversal do contêiner.
- `flex-end` Alinhe os filhos de um contêiner ao final do eixo transversal do contêiner.
- `center` Alinhe os filhos de um contêiner no centro do eixo transversal do contêiner.
- `baseline` Alinhe os filhos de um contêiner ao longo de uma linha de base comum. Os filhos individuais podem ser definidos como a linha de base de referência para seus pais.





## Alinhar conteúdo

`alignContent` define a distribuição de linhas ao longo do eixo transversal. Isso só tem efeito quando itens são agrupados em várias linhas usando `flexWrap`.

- `flex-start` (valor padrão) Alinhe as linhas quebradas ao início do eixo transversal do contêiner.
- `flex-end` Alinhe as linhas quebradas no final do eixo transversal do contêiner.
- `stretch` Estique as linhas quebradas para coincidir com a altura do eixo transversal do contêiner.
- `center` Alinhe as linhas quebradas no centro do eixo transversal do contêiner.
- `space-between` Espace as linhas uniformemente ao longo do eixo principal do contêiner, distribuindo o espaço restante entre as linhas.
- `space-around` Espace uniformemente as linhas ao longo do eixo principal do contêiner, distribuindo o espaço restante em torno das linhas. Comparado a `space-between`, o uso `space-around` resultará na distribuição do espaço no início da primeira linha e no final da última linha.

