

# Flexbox

Última atualização do doc. 06.Nov.2021

[Voltar para CSS \(Doc. principal\)](#).

- [Flex Container](#)
- [Flex Item](#)

CSS styles timeline

Flexbox: 2012 (ano do aparecimento das Media Queries)

O *Flexbox Layout (Flexible Box)* tem como objetivo a disponibilização de uma forma mais eficiente para dispor, alinhar e distribuir o espaço entre os itens de um *container*, mesmo quando o seu tamanho é desconhecido e/ou dinâmico (daí a palavra "flex").

A ideia principal na qual se baseia o *layout flexível* é fornecer ao *container* a capacidade de alterar a largura/altura (e ordem) de seus itens para melhor preencher o espaço disponível (principalmente para dar resposta a todos os tipos de dispositivos e respetivas áreas de visualização). Um *flex container* expande os itens para preencher o espaço livre disponível ou faz a sua redução por forma a evitar que ultrapasse o espaço disponível (evitar o *overflow*).

*Flexbox layout* é mais apropriado para as componentes/partes de uma aplicação/página (ou para páginas de pequenas dimensões), enquanto que o *Grid layout* destina-se a páginas de estrutura maior e, eventualmente, mais complexa.

## Flex Container

O Flex Container é a *tag* que envolve os itens flex, ao indicar `display: flex`, essa tag passa a ser um **Flex Container**.

### display: flex

Define o elemento como um flex container, tornando os seus filhos flex-itens.

### flex-direction

Define a direção dos flex itens. Por defeito é `row` (linha), por isso quando o `display: flex;` é adicionado, os elementos ficam em linha, ao lado uns dos outros.

### flex-wrap

Define se os itens devem quebrar ou não (mudança de linha). Por defeito, os `flex itens` são compactados, podendo ocorrer que seja excedido o limite da área disponível.

### flex-flow

A propriedade `flex-flow` serve para especificar, conjuntamente, as propriedades `flex-direction` e `flex-wrap`.

### justify-content

Alinha os itens flex no *container* de acordo com a direção. A propriedade só produz efeito se os itens atuais não ocuparem todo o *container*.

## align-items

Alinha os *flex itens* de acordo com o eixo do *container*. O alinhamento é distinto para quando os itens estão em colunas ou linhas.

## align-content

Alinha as linhas do *container* em relação ao eixo vertical. A propriedade só funciona se existir mais de uma linha de flex-itens. Para isso a propriedade `flex-wrap` precisa ser `wrap`.

Além disso, apenas produz efeito no caso do `container` for maior do que a soma das linhas dos itens. Isto significa que no caso de não estar definido `height` para o container, a propriedade não influencia o *layout*.

# Flex Item

Os *Flex Itens* são os filhos diretos de um *Flex Container*.

É possível que um *Flex Item* seja também um *Flex Container*, desde que, esse item, seja definido como `display: flex`. Assim, os filhos desse item também serão flex itens.

## flex-grow

Define que um *flex item* pode crescer. Por defeito o valor é zero, desta forma os *flex itens* ocupam um tamanho máximo relacionado de acordo com seu conteúdo ou como o `width` definido.

Ao definir 1 para todos os `Flex Itens`, normalmente ficam com a mesma largura e ocupam toda a largura (100%) do *container*. No caso de um determinado elemento ter um conteúdo muito *largo*, o item respeita o mesmo.

No caso de, por exemplo, de uma linha com quatro itens, onde três são `flex-grow: 1` e um `flex-grow: 2`, o `flex-grow: 2` ocupará duas vezes mais espaço do que os outros elementos.

A propriedade `justify-content` não funciona em itens com `flex-grow` definido.

## flex-basis

Indica o tamanho inicial do flex item antes da distribuição do espaço restante.

Quando existe a definição `flex-grow: 1` e `flex-basis: 1`, o valor restante para ocupar o *container* é distribuído ao redor do conteúdo do *flex-item*.

## flex-shrink

Define a capacidade de redução de tamanho do item.

## flex

Serve para especificar, conjuntamente, as propriedades `flex-grow`, `flex-shrink` e `flex-basis`.

## order

Modifica a ordem, ascendente, dos *flex itens*.

## **align-self**

Define o alinhamento específico de um único *flex item* dentro de um *container*. Caso um valor seja atribuído, terá mais *peso* do que for atribuído no `align-items` do *container*.