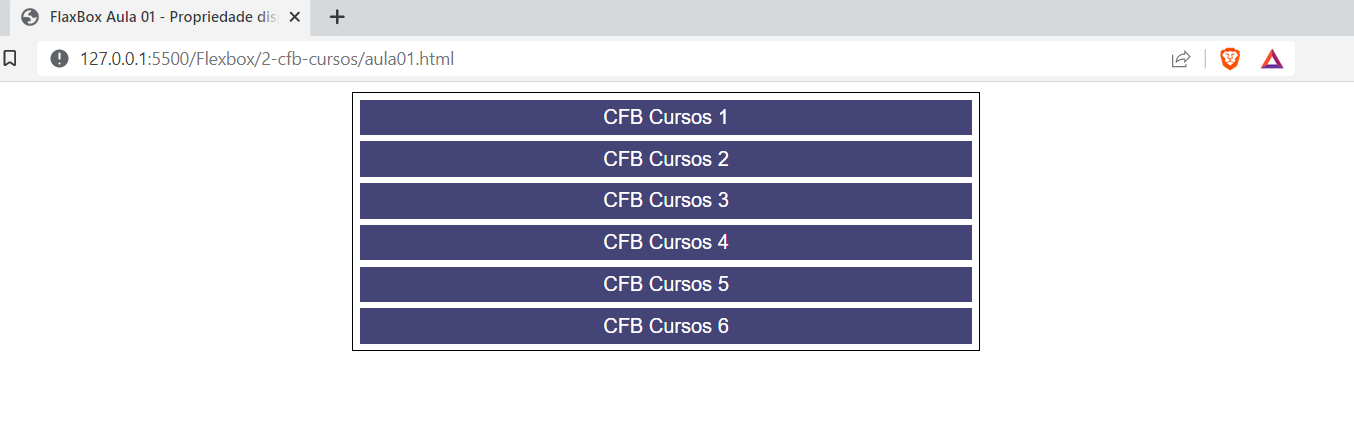
FlexBox

**Comandos de manipulação do Container:**

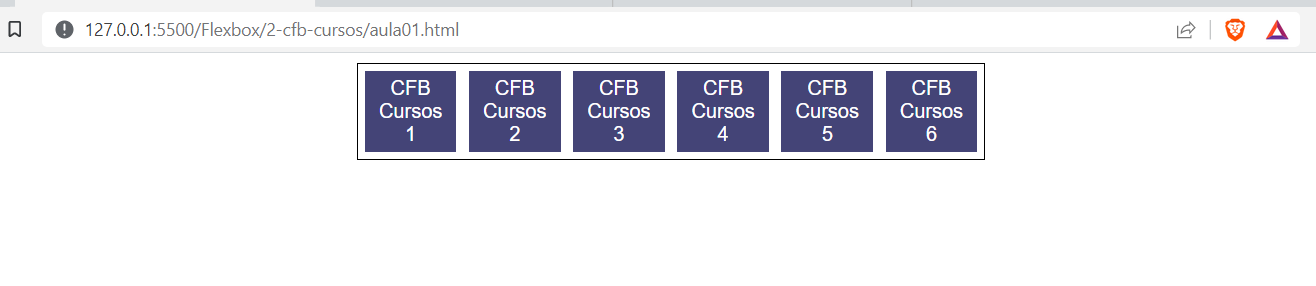
**display: block;** - É o que geralmente é o Padrão. Cada bloco ocupa toda a linha

Gráfico

Descrição gerada automaticamente com confiança média **Eixos:**

**Display: Flex;**

**display: flex;** - Coloca as Divs uma do lado da outra. Permite que tenhamos mais domínio do layout da página.



**Interface gráfica do usuário, Aplicativo, Teams

Descrição gerada automaticamenteCaracterística (1):** Se adicionarmos mais itens que não caibam dentro da largura definida para o container, os itens sairão do container.

**Característica (2):** Já para a altura, os itens acompanham a altura determinada para o container.

Gráfico

Descrição gerada automaticamente

**Flex-direction:**

**flex - direction;** - Manipula os eixos de orientação do container flexível. Por padrão, temos o **flex-direction: row;**

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

**Flex-wrap:**

**flex - wrap;** - Define a quebra de linha do container flexível e justa as Divs de acordo com o tamanho da Div-pai. Por padrão temos o **flex-wrap: nowrap;**

Teclado de computador

Descrição gerada automaticamente

do que um "display: inline-block"

flex-direction: row;

/\*Com o 'display: flex' podemos utilizar o 'flex-direction:' para determinar se as div

**Flex-flow:**

**flex - flow;** - Junta duas propriedades que são, flex-direction e flex-wrap em uma só propriedade. Por padrão temos o **flex-flow: row nowrap**;

**Comandos de manipulação dos Itens de um container:**

**POSICIONAMENTO HORIZONTAL:**

**Justify - Content: (de acordo com o main axis)**

**Justify-content;** - É uma propriedade de alinhamento dos elementos contidos em um container. Por padrão temos o **Justify-content: flex-start;**

* **justify-content: flex-start**; - (padrão) posiciona no início do main axis da Div-pai
* **justify-content: flex-end;** - Posiciona no fim do main axis da Div-pai
* **justify-content: center;** - Posiciona os itens ao centro da Div-pai
* **justify-content: space-between;** - Adapta um espaçamento uniforme entre as divs de acordo com o tamanho da Div-pai
* **justify-content: space-around;** - Cria um espaçamento uniforme em volta de cada div, respeitando o tamanho da Div-pai

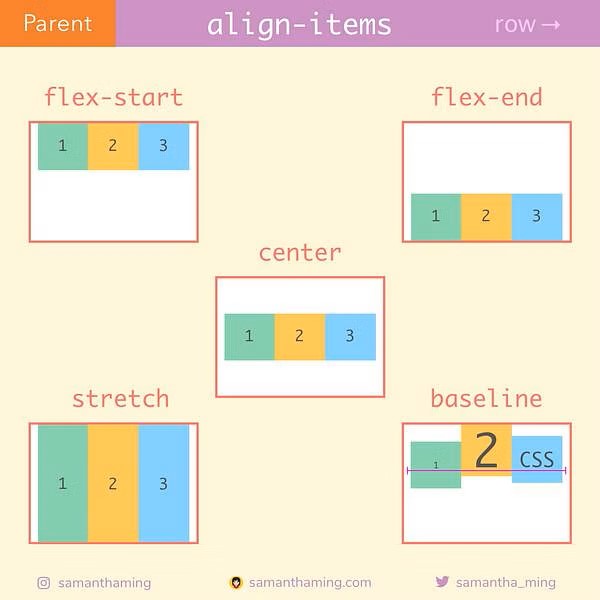


**POSICIONAMENTO VERTICAL:**

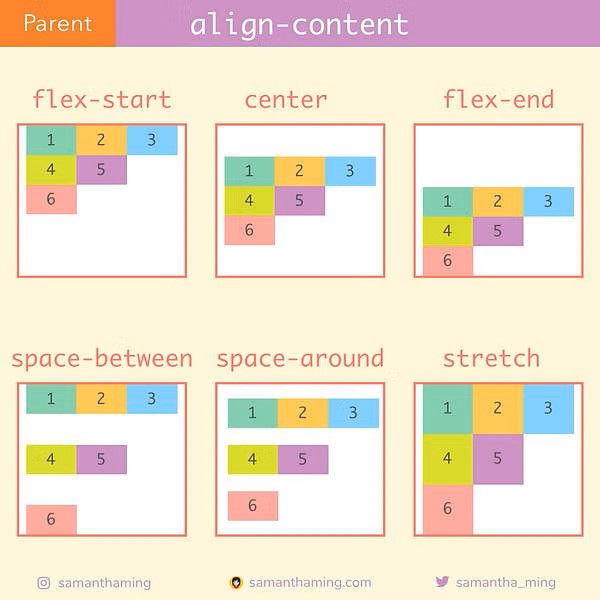
**Align - Items: (de acordo com o cross axis)**

**Align-items;** - Também é uma propriedade de alinhamento dos elementos contidos em um container, porém atua de acordo com o eixo secundário. Por padrão temos o **align-items: stretch;**

* **align-items: stretch;** - (Padrão) As divs acompanham a altura total da Div-pai
* **align-items: center;** - Centraliza de acordo com o eixo secundário
* **align-items: flex-start;** - Posiciona no início do eixo secundário
* **align-items: flex-end;** - Posiciona no final do eixo secundário



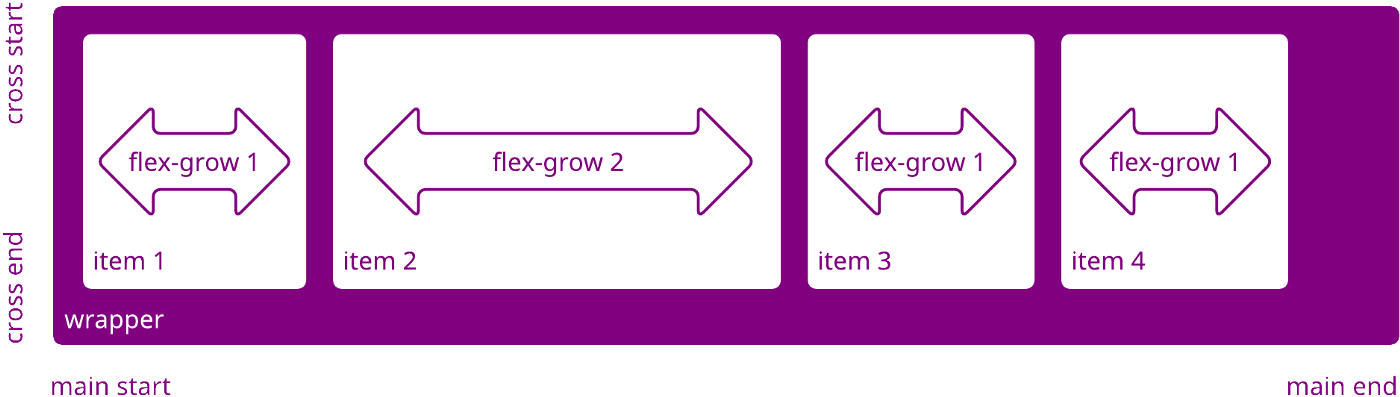
**Align-content;** - Também é uma propriedade de alinhamento, muito semelhante ao align-items (funcionando no eixo secundário), porém **sem aplicar espaçamento** entre os elementos. Por padrão temos também o **align-content: stretch;**

****

Diagrama

Descrição gerada automaticamente com confiança média**Diferença entre align-items e align-content:**

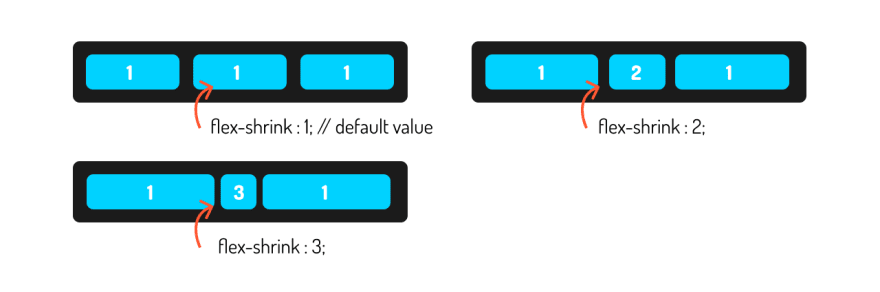
**CRESCIMENTO:**

**Flex - Grow;** - Os itens têm o seu tamanho adaptado automaticamente ao tamanho do container. **OBS**\*: Deve sempre ser usada na classe da Div-Filha. Como padrão temos **flex-grow: 0;**

**REDUÇÃO:**

**Flex - Shrink;** - Indica se os elementos poderão ser reduzidos ou não dentro de um container. Quanto maior o valor, maior a capacidade de redução do elemento. Como padrão temos **flex-shrink: 1;**

**OBS\*: flex-shrink: 0;** significa que o elemento não pode reduzir, e por isso os itens sairão do container ser for preciso.

****