



escola  
britânica de  
artes criativas  
& tecnologia

# Profissão: Engenheiro Front-End



**CSS**



## **GUIA DA AULA 5**



# Defina o display



Acompanhe aqui os temas que serão tratados na videoaula

- **Propriedade**
- **Flexbox**
- **Flexbox container**
- **Flexbox justify content**
- **Flexbox align items**
- **Flexbox direction**



## Propriedade

Com a propriedade **display** informamos ao navegador como e se um elemento deve ser exibido. Os principais valores que iremos explorar são:

### 1) **Display: none**

Com o valor none o elemento não será exibido.

### 2) **Display: block**

O elemento será exibido como um bloco, não tendo nenhum outro elemento a seu lado, como se houvesse uma quebra de linha antes e depois do elemento.

### 3) **Display: inline**

O elemento será exibido na mesma linha, como se fosse uma tag, span, por exemplo as listas possuem seus itens separados por linhas, com o display inline sendo a aplicado na tag li veremos os itens lado a lado.



## Flexbox

O flexbox não é uma propriedade CSS e sim um **módulo**.

É composto por algumas propriedades que exploraremos, como:

- flex-direction;
- flex-wrap;
- align-items;
- align-content.

Para começarmos a trabalhar com o Flexbox temos um requisito que é a construção de um *container* que possua o display configurado com o valor flex.



## Flexbox container

Para construir o *container* precisamos apenas de uma div com a propriedade display possuindo o valor flex. É através do container que iremos controlar o *layout*.

Vamos considerar o *layout*:



## Flexbox justify content

Com a propriedade `justify-content` informaremos ao navegador qual é a forma que os elementos dentro do container devem ser distribuídos.

Os dois principais valores são **Flex-Start e Flex-End**, mas também temos outros como o **center**, **space-around** e **space-between**. Acompanhe:

**1)** Com o **`justify-content: flex-start`**, informamos que os elementos devem estar concentrados ao lado esquerdo do container, onde ele se inicia:



2) Com o **justify-content: flex-end**, informamos que os elementos devem estar concentrados ao lado direito do container, onde é seu termino:



3) Com o **justify-content: center**, informamos que os elementos devem ser centralizados:





4) Com **justify-content: space-around**, informamos que os elementos devem ser separados e o espaço vago do container deverá ser distribuído de forma igual ao contorno dos elementos:



5) Com o **justify-content: space-between**, informamos que os elementos devem ser separados e o espaço vago do container deverá ser distribuído de forma igual, porém sem um contorno entre os elementos:



## Flexbox align items

O Flexbox também nos ajuda no alinhamento de itens na vertical.

Consideramos um container com os elementos H1, H3 e H6, por padrão seriam exibidos assim:

Texto em H1
 Texto em H3
 Texto em H6



Para alinharmos esses elementos considerando o canto inferior usamos a propriedade `align-items` com o valor **baseline**:

**Texto em H1**
 Texto em H3
 Texto em H6

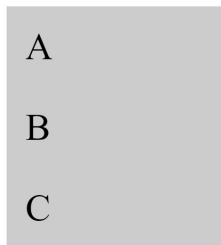
Para centralizarmos bastaria usar o valor **center**:

**Texto em H1**
 Texto em H3
 Texto em H6



## Flexbox – flex direction

Com essa propriedade podemos alterar a forma que os elementos dentro do container irão aparecer. O valor padrão é **row**, mas podemos mudar para **column** para que os elementos fiquem um embaixo do outro, se organizando em formato de colunas:



Note que com o **display: flex** os elementos buscam ocupar toda a área disponível.

