CSS FLEXBOX

**AULA01- DISPLAY: FLEX;**

Eixos de trabalho:

Main axis = eixo principal (da esquerda para direita)

Cross axis = eixo cruzado ou secundário (de cima para baixo)

Existem propriedades para containers e propriedades para itens

Da para adicionar duas classe para um mesmo código html

Ex:

<div class=”class1 class2”>

Exemplo

</div>

# AULA02- FLEX-DIRECTION

O padrão do flex-direction é o row (um do lado do outro)

flex-direction:

row: um do lado do outro;

column: um abaixo do outro;

row-reverse: ao contrario em linhas;

column-reverse: ao contrario em colunas

# AULA03- FLEX-WRAP

Define se os itens vão ou não poder passar para a linha de baixo

Padrão: nowrap

flex-wrap:

nowrap: não passar para alinha de baixo;

wrap: passar para linha de baixo;

wrap-reverse: quebra de linha invertida

# AULA04- FLEX-FLOW

Indica o valor para o flex direction e flex wrap

# AULA05- JUSTIFY-CONTENT

Define o alinhamento dos itens de acordo com o main axis (eixo principal)

Padrão: flex-start

justify-content:

* flex-start: alinha no inicio do eixo
* flex-end: alinha no final do eixo
* center: no centro
* space-between: espaça de forma igual entre eles, deixando um no flex start e outro do end
* spave-around: distribue igualmente os elementos

# AULA06 – ALIGN-ITENS

Controla o alinhamento vertical dos itens ou seja o eixo secundário (cross axis)

Padrão: stretch

* align-itens:
* stretch: esticado
* flex-start: inicio do eixo secundário
* flex-end: final do eixo secundário
* center: centro do eixo
* baseline: de acordo com a linha base do elemento

curiosidade sobre width: da para fazer cálculos dentro de width

width: calc(33.3333% - 10px - 10px);

## AULA07- ALIGN-CONTENT

Quase a mesma coisa que align itens

Mas ele não aplica o espaçamento vertical

No space around ele faz a mesma coisa que o center do align itens

E no between so n tem o espaçamento do inicio e do final

## AULA08- FLEX-GROW

Relação ao tamanho de um conteúdo

/\* o flex-grow ocupa a linha toda e a divisão de espaço varia com o tamanho ou seja um item com o grow 3 terá uma distribuição maior que um grow 2 \*/

Padrão: 0 (respeita ao conteúdo padrão do conteúdo)

flex-grow:(tamanho) ex:3

## AULA09- FLEX-BASIS

Ao aplicar o mesmo flex-grow em dois itens ou mais, o tamanho mesmo com o flex-grow muda pois ele vai respeita o conteúdo, ou seja se um item tiver mais conteúdo ele será maior

Para forçar todos os itens a ter o mesmo tamanho, o flex-basis vai entrar em ação.

Padrão: auto (não muda nada)

flex-basis: (tamanho) ex: 3

ao mudar os números, ele tem um propriedade parecida com o flex-grow mas e mais limitado em relação a distribuição de espaço sendo recomendado não usar para essa finalidade

# AULA10- ORDER

Server para redefinir a ordem dos elementos

Direcionado ao item

A ordem começa em 0

Ex:

order: 0;

# AULA11- FLEX-SHRINK

Propriedade da capacidade de um elemento diminuir para caber dentro de um container

Padrão: 1

Ao colocar em 0 ele perde a capacidade de reduzir

Quanto maior o item maior a redução

# AULA12- PROPRIEDADE FLEX

É um shorthand de (flex-grow flex-shrink flex-basis)

## AULA13- ALIGN-SELF

usado para o alinhamento de um único item