

**Posicionamento de Elementos com Flexbox em CSS**

Apresentar os fundamentos e aplicações da propriedade Flexbox na criação de layouts responsivos sem a necessidade de definição de valores fixos.

**Introdução ao Flexbox:**

Projetado como um modelo de layout unidimensional e como método que pode oferecer distribuição de espaço entre itens em uma interface e recursos limitados.

\* Flex Container

- Tag que envolve os itens.

- Faz uso da propriedade “display:flex”

- Transforma todos os seus itens filhos em flex itens.

- O container pode ser aplicado em qualquer tipo de tag que tenha itens filhos.

\* Propriedade do Flex Container

- display: é o iniciador do container.

- flex-direction: determina o direcionamento dos itens, em linha, coluna ...

- flex-wrap: se aplica para se ter quebra de linha ou não.

- flex-flow: abreviação para Direction e wrap.

- justify-content: alinha o itens do container de acordo com os itens.

- align-items: alinha o itens de acordo com o eixo do container.

- align-content: faz o alinhamento das linhas do container.

\* Flex Item

- São elementos filhos diretos de um flex container.

- Que também podem ser flex container de seus elementos filhos.

\* Propriedade dos Flex Itens

- flex-grow: define o fator de crescimento.

- flex-basis: define o tamanho inicial do item antes da distribuição do espaço restante.

- flex-shrink: define a capacidade de redução.

- flex: abreviação das propriedades grow, basis e shrink.

- order: define distribuição e listagem dos itens.

- align-self: define o alinhamento de um item específico do container.

**Propriedades do Flex Container**

As propriedades têm duas funções:

- Manipulação do container.

- Manipular os flex itens (filhos dos containers).

Propriedades de manipulação do container:

- **Display Flex**: os itens dentro de um container flex ficam em linha com cada um ocupando o tamanho proporcional a seu conteúdo.

Tem como padrão a orientação horizontal.

Se os itens não conseguirem se comportar dentro do container este começaram a vazar para fora deles.

- **Flex-direction**: propriedade que estabelece o eixo principal dentro do container.

O linha que é horizontal.

row – alinhamento em linha, por padrão, da esquerda para a direita.

row-reverse - em linha, com alinhamento da direita para esquerda.

O coluna que é o vertical.

column – alinhamento em coluna, alinhamento de cima para baixo a esquerda da tela.

colunm-reverse - em coluna, com alinhamento da baixo para cima a direita da tela.

- **Flex-wrap**: define se os itens devem fazer quebra de linha. Por padrão eles não quebram linha.

wrap – permite a quebra de linha.

nowrap – não permite a quebra de linha (padrão).

wrap-reverse – permite a quebra de linha, sendo a quebra para a linha superior.

Flex-direction – poder ser em row ou column.

- **Flex-flow**: é um atalho para se aplicar as propriedades flex-direction e flex-wrap.

**- Justify contente:**  alinha e distribui os itens dentro de um container.

Flex-start: alinha ao início do container.

Flex-end: alinha ao final do container.

Center: ao centro do container.

Space-between: cria um espaçamento igual entre os elementos, sendo que os espaçamento do início e final ficam as margens.

Space-around: os espaçamentos dos meios ficam duas vezes maior que os espaçamento do início e do final.

**- Align Items:**  Trata do alinhamento dos flex itens de acordo com o eixo do container. O alinhamento linha tem um comportamento diferente do alinhamento em coluna.

Center: alinhamento dos itens ao centro do container.

Atretch: os flex itens crescem igualmente (padrão).

Flex-start: alinhamento dos itens no início do container.

Flex-end: alinhamento dos itens no final do container.

Baseline: alinhamento de acordo com a linha de base da tipografia dos itens.

**- Align Content:**  responsável por tratar o alinhamento das linhas do container em relação ao seu eixo vertical.

Para o uso desta propriedade é necessário que:

- O container utilize quebra de linha (flex-wrap).

- A altura do container seja maior que a soma das linha dos itens.

Center: alinhamento dos itens ao centro.

Strech: os flex itens crescem igualmente (padrão).

Flex-start: alinhamento dos itens no início.

Flex-end: alinhamento dos itens ao final.

Space-between: cria um espaçamento igual entre os elementos.

Space-around: os espaçamentos dos itens do meio são duas vezes maiores que no início e no final do container.

**Propriedades dos Flex Items**

Flex-grow: define a proporcionalidade do crescimento dos itens, respeitando o tamanho dos seus conteúdos internos.

Obs: esta propriedade não irá funcionar se for aplicada a propriedade justify-content no flex container.

Flex Basis: é a propriedade que estabelece o tamanho inicial do item antes da distribuição do espaço dentro dele, usando como base o conteúdo interno disposto.

Tem uma relação direta com o que foi definido no flex-grow.

Pode ter como valores: auto, um valor fixo em px, % ou o valor 0(zero).

Flex-shrink: estabelece a capacidade de redução ou compressão do tamanho do item.

Flex: é um atalho ou abreviação para determinar as propriedades grow, shrink e basis, colocados os seus valores nesta ordem.

Order: especifica a ordem que os itens devem ser apresentados. Por padrão todos os itens tem order em 0. É possível trabalhar com valores negativos.

Align-self: é a propriedade que estabelece o alinhamento de modo individual sobre um determinado item. Tem os valores possíveis:

Auto: é o valor padrão.

Flex-start: ao início de container

Flex-end: alinhamento ao final do container.

Center: relativo ao centro de acordo com o eixo.

Stretch: ocupa todos os espaços relativos.

Baseline: utiliza a linha base da tipografia.

- Projeto: Trabalhando a Responsividade da Página

1 – Criar uma área para fazer os breakpoints:

2 – Adicionar o tamanho da tela máximo para as modificações:

@media(max-width: 992px){ }

3- Dentro do tag do tamanho da tela deve-se colocar as modificações nas classes que devem se adaptar melhor na tela com responsividade.