**Fundamentos do Flexbox PT1**

**Objetivo do curso:**

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

**Utilizado em aula:**

* VS Code
* HTML Snippets
* Live HTML Previewer

**Flexbox:**

* Foi pensado como um modelo de layout unidimensional e como um método que pode oferecer distribuição de espaço entre itens em uma interface e recursos de alinhamento

**Flex Container:**

* É a tag que envolve os itens. Será nela que iremos aplicar a propriedade **“display: flex”**. **Transforma** todos os seus **itens filhos em flex itens**.

Propriedades relacionadas:

* **display** – inicializador do container
* **flex-direction** – vai fazer o direcionamento dos itens, seja em linha ou coluna
* **flex-wrap** – vai se aplicar para quebra de linha ou não
* **flex-flow** – abreviação do Direction e do wrap
* **justify-content** – alinhar os itens do container de acordo com a sua direção
* **align-items** – vai alinhar esses itens de acordo com seu eixo do container
* **align-content** – vai alinhar as linhas desse container

**Flex item:**

* são os elementos filhos diretos do Flex Container. E também podem se tornar Flex Container

Propriedades relacionadas:

* **flex-grow** – vai definir o fator de crescimento
* **flex-basis** – vai definir o tamanho inicial desse item, antes da distribuição do espaço restante dentro do container
* **flex-shrink** – defini a capacidade de redução
* **flex** – abreviação para as 3 propriedades anteriores
* **order** – ordem de distribuição e listagem dos itens
* **align-self** – vai definir o alinhamento de um item específico desse container

**Fundamentos**

**Display: flex;**

* Torna a tag um elemento do tipo flex container, e assim automaticamente todos os seus filhos diretos desta tag tornam-se flex items.
* Ex tag: div, span, section, h1, h2, a...

**Ctrl + shift + p =** buscar alguma extensão no VS Code

**Flex-direction**

* é basicamente a propriedade que estabelece o eixo principal do container, definindo assim a direção que os flex items são colocados no flex container.
* eixos: linha (horizontal) e coluna (vertical)

Eixos:

* **row** (padrão): à direção do texto, **esquerda para direita**. 1 2 3 4
* **row-reverse**: sentido oposto à direção do texto. 4 3 2 1
* **column**: ordenação de **cima para baixo**, em coluna única
* **column-reverse**: ordenação inversa, de **baixo para cima**

**Flex-wrap**

* É a propriedade que define se os itens devem ou não quebrar a linha.
* Por padrão, eles não quebram linhas. Isso faz com que os flex items sejam compactados além do limite do conteúdo

**nowrap**

* **nowrap: é o padrão**. não permite a quebra de linha

**wrap**

* **wrap:** permite a quebra de linha assim que um dos flex items não puder mais ser compactado

**wrap reverse**

* permite a quebra de linha assim que um dos flex items não puder mais ser compactado, porém na direção contrária da linha acima. **O item que quebrou a linha vai para cima**

**flex-flow**

* é um atalho para as propriedades flex-direction e flex-wrap
* seu uso não é tão comum, pois quando mudamos o flex-direction para column, mantemos o **padrão do flex-wrap**, que é **nowrap**

**justify content**

* vai se encarregar de alinhar os itens dentro do container de acordo com a direção pretendida e tratar da distribuição de espaçamento entre eles
* Obs: caso os itens estejam ocupando 100% de todo o container, ela não se aplica

Variações:

* flex-start: início do container
* flex-end: final do container
* center: ao centro do container
* space-between: cria um espaçamento igual entre os elementos
* space-around: os espaçamentos do meio são duas vezes maiores que o inicial e final

**align-items**

* trata do alinhamento dos flex items de acordo com o eixo do container
* o alinhamento é diferente para quando os itens estão em colunas ou linhas
* permite o alinhamento central no eixo vertical

tipos de alinhamento:

* center: alinhamento dos itens ao centro
* stretch: padrão. faz com que os flex items cresçam igualmente
* flex-start: alinhamento dos itens no início
* flex-end: alinhamento dos itens no final
* baseline: alinhamento de acordo com a linha base da tipografia dos itens

**align-content**

* é a propriedade responsável por tratar o alinhamento das linhas do container em relação ao eixo vertical do container

requisitos:

* o container utilize quebra de linhas
* a altura do container seja maior que a soma das linhas dos itens

tipos de alinhamento:

* center: alinhamento dos itens ao centro
* stretch: é o **padrão**.Os flex items crescem igualmente
* flex-start: alinhamento dos itens no início
* flex-end: alinhamento dos itens no final
* space-between: cria um espaçamento igual entre os elementos
* space-around: os espaçamentos do meio são duas vezes maiores que o inicial e final