

Lembra do flex Grow?

Nele definimos o quanto o item vai crescer de acordo com o tamanho disponível dentro do container

Já com o flex basic vamos definir qual o menor tamanho que esse item vai ter, antes de ele mesmo se reajustar.

🧩 flex-grow

* Define **quanto o item pode crescer** em relação aos outros, **quando há espaço sobrando** no container.
* Exemplo: Se você tem três itens com flex-grow: 1, 2, e 1, o segundo vai ocupar o dobro do espaço extra em relação aos outros.

📏 flex-basis

* Define o **tamanho inicial do item** antes de qualquer crescimento ou encolhimento.
* É como dizer: “Esse é o tamanho ideal que eu quero para esse item.”
* Pode ser em px, %, em, etc. Exemplo: flex-basis: 200px.

🔄 Como eles interagem

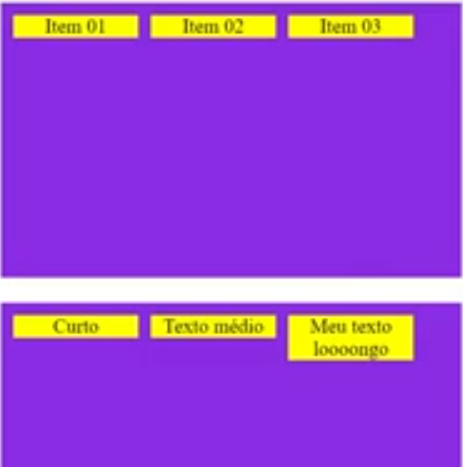
Quando você usa a shorthand flex: grow shrink basis, está dizendo:

flex: 1 1 200px;

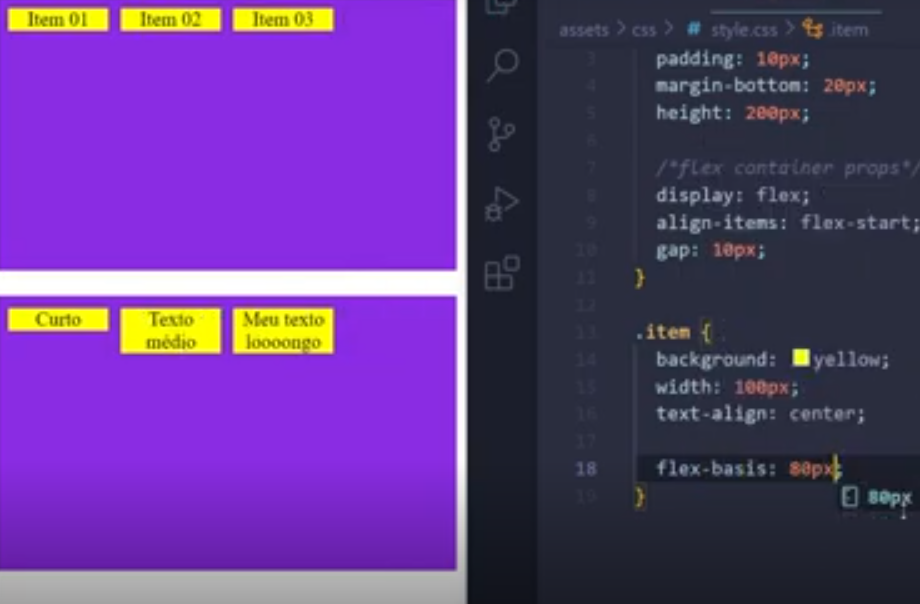
Isso significa:

* Cresça (grow) se tiver espaço.
* Encolha (shrink) se faltar espaço.
* Comece com 200px como base.

Pratica

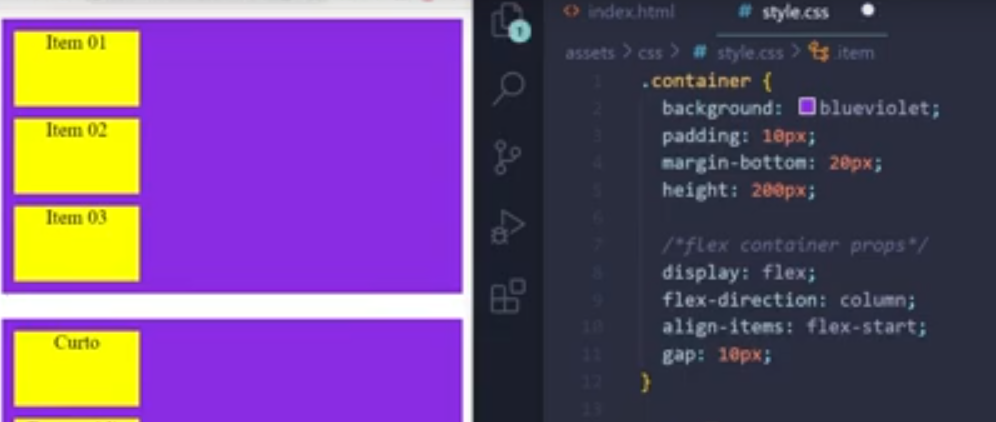


Antes sem nada

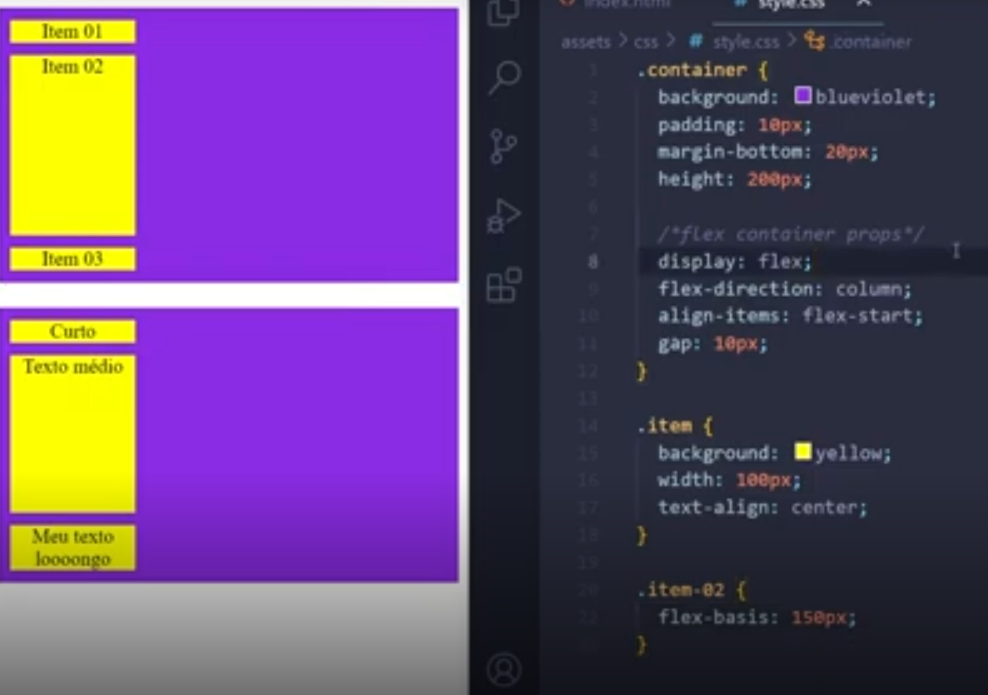


Aplicando que o item deve ter um tamanho base de 80px

Atenção se você reparar, você vai ver que estamos usando um row, que é o padrão do flex-direction

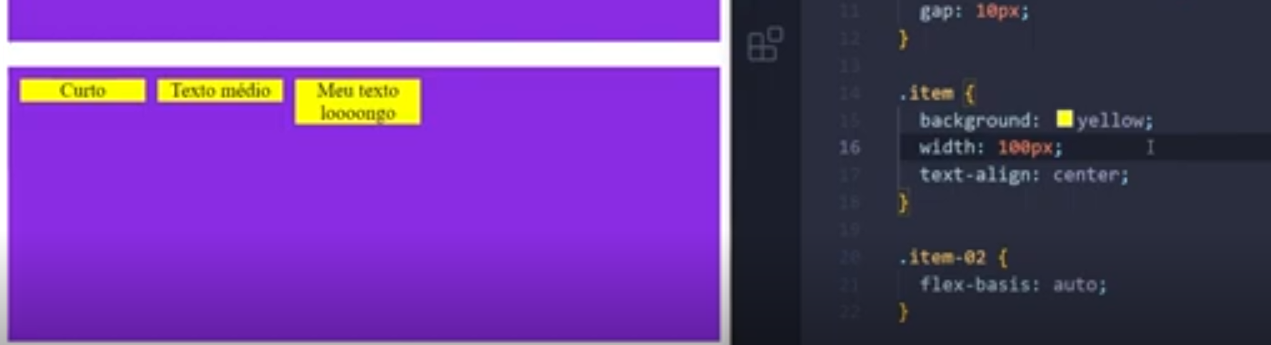


Mas aqui estamos usando um column então ele vai aplicar o basis na altura e não la largura.



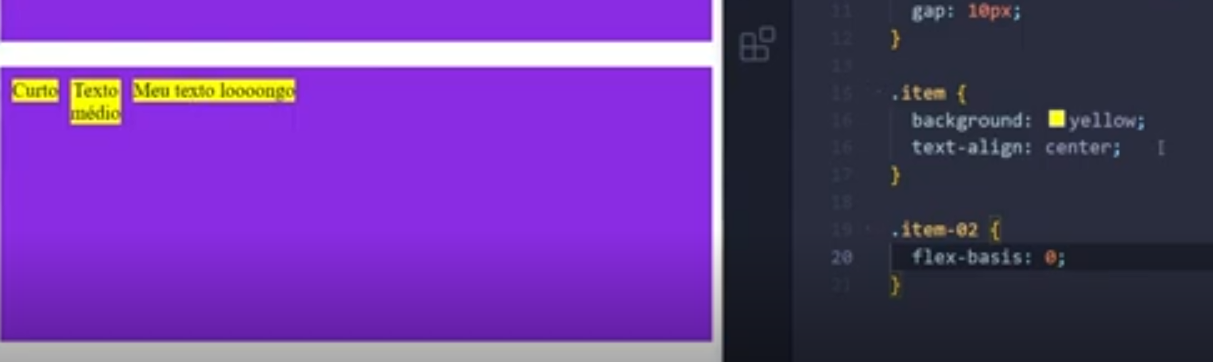
Agora aplicando um valor básico de 150px

Outros valores para essa propriedade

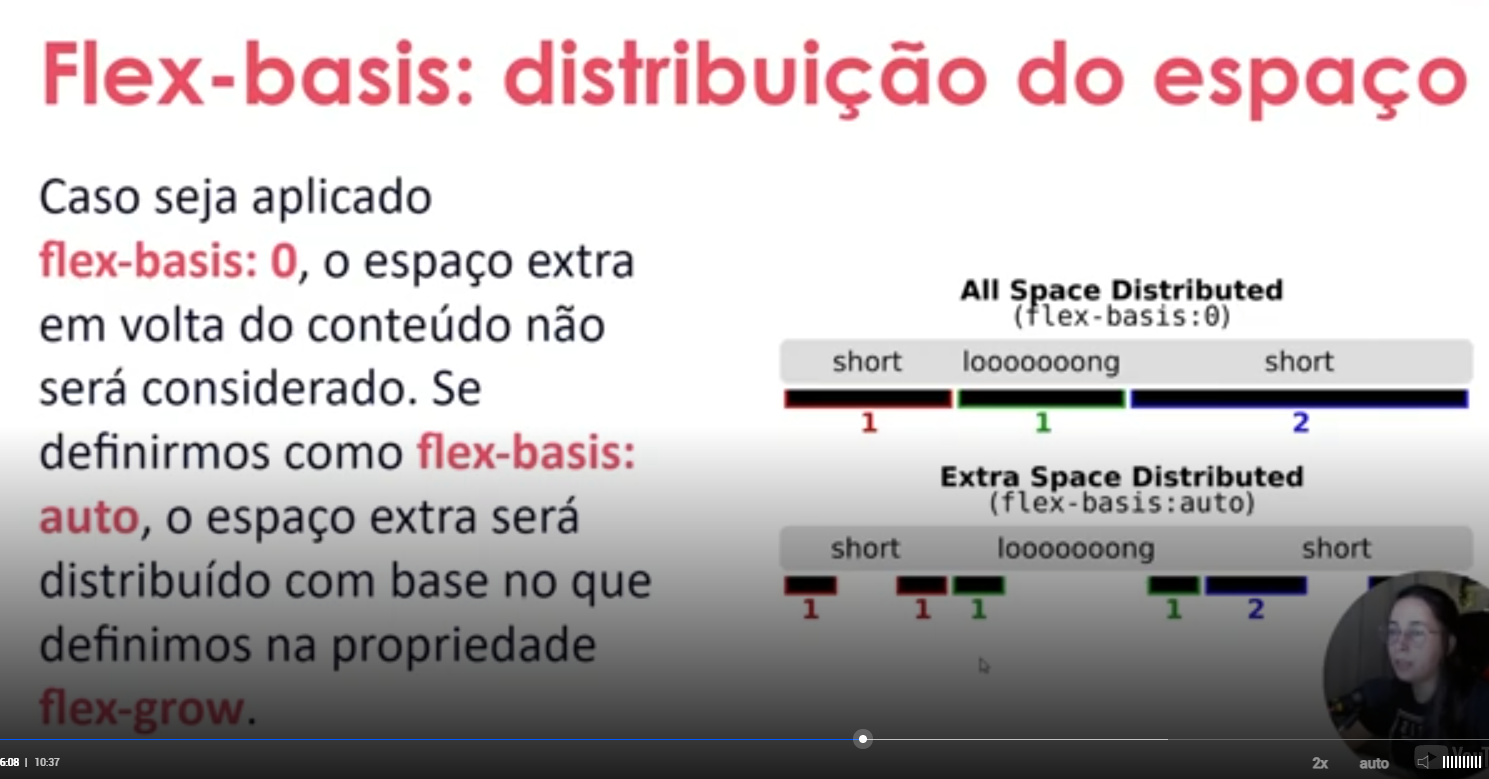


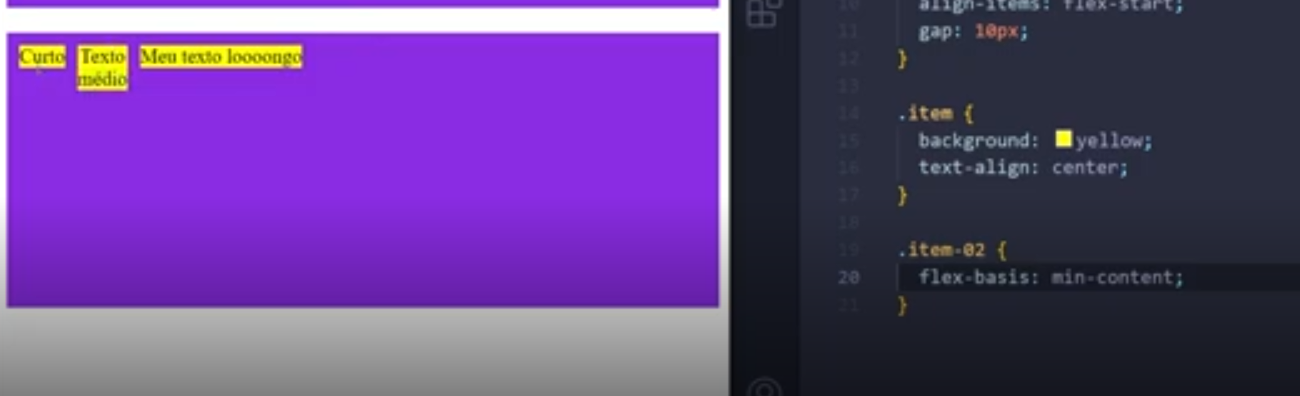
O valor auto, ele vai verificar se já foi definido alguma largura ou altura para o item, se já foi ele vai usar como base.

Se não tiver nada definido ele vai colocar somente o espaço que o conteúdo precisa para ser exibido.



No caso do base 0 ele vai retirar todo o espaço extra, ele vai usar somente o espaço que ele precisa para ser exibido.



O min contente ele vai pegar a menor palavra do nosso conteúdo e definir como o tamanho.

