

-----Pra Lembrar-----

criação do layout geral com grid - header, main, footer

rem(px) - função que transforma pixel em rem

elemento-pai__nome-proprio - pra melhorar a hierarquia e pro google identificar melhor os elementos. *quando for botar a classe em um elemento filho, botar primeiro o nome da classe do elemento pai, dois underline e o nome da classe que voce quiser dar

quando não tem underline-underline (seguindo a convenção) apesar de ser considerado filho direto, ele não necessariamente tem uma relação direta com o elemento pai

360 de tamanho é o maximo que devemos nos preocupar com a responsividade (porem caso queira algo a mais, da pra ir até 320)

o ponto (.) no grid area representa um espaço vazio, sem nada

flex-wrap: wrap; - quebra a tag inteira ao inves de quebrar no meio da palavra

vertical-align: text-bottom - alinha o texto com o icone na parte de baixo

viewBox=' ' -

fill: ;

currentColor - assumi a cor do elemento pai pra si mesmo

fill: currentColor;

cuidado com imagens, ela pode distorcer quando usar flex-box. Flex-box tenta fazer com que o flex-item ocupe toda a extensão

o flex-box ao inves de trabalhar com tamanhos fixos, ele trabalha com espaços e proporcoes, fazendo o calculo sozinho, *pega o que tem livre e distribui pra mim*

cascata
herança
especificidade

!important

css e html

atributo data- (no html)

[] - brackets (no css)

serve pra capturar atributos no css? atributos data- ou todos os tipo?

ex1: [atributo] { estilo: css; }

ex2: [atributo="valor"] { estilo: css; }

translate3d() - tem mais performance que o translate? como funciona?

offset (importante)

attr() - css

tabindex - atributo

```
<!-- css - clear, scroll / html - data-, -->  
<!-- depois de terminar dnv html - css ir pro ajax? -->
```

Quando for estudar ferramentas do css-html estudar CANVA DO HTML tambem e
acessibilidade*/

Tailwind CSS