

Extra - Flexbox

quinta-feira, 7 de setembro de 2023 09:58

Enfim, o flex

Organizar os elementos na tela, tendo em vista o cenário atual do web design, parece ser uma tarefa extremamente difícil (e, convenhamos, é). No entanto, algumas novidades do CSS vêm para amenizar as dores do desenvolvedor. O flexbox, assunto de hoje, vai nos dar dicas de como podemos organizar elementos em uma mesma direção.

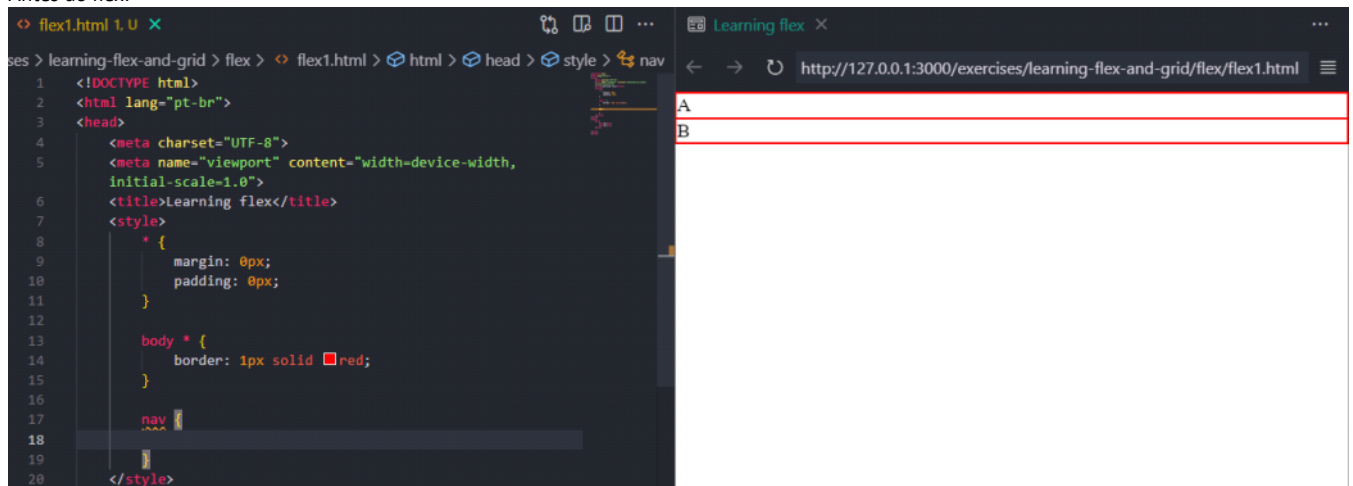
Tudo é caixa

Sabemos que todos os elementos do HTML seguem um modelo de caixas. Além disso, todas estas caixas possuem uma característica que revela a forma como elas se apresentam diante das demais (o seu comportamento), que é o `display`. Para relembrarmos, um **"`display: block`"** nos diz que a caixa ocupa todo o espaço horizontal da tela, impedindo que elementos fiquem ao lado dela, como é o caso das tags de título (`h1`, `h2`, etc.). O **"`display: inline`"**, por outro lado, nos diz que a caixa de um elemento ocupa o espaço de seu conteúdo, o que permite que outros elementos fiquem ao lado desta caixa (tags `<a>`, ``, etc.).

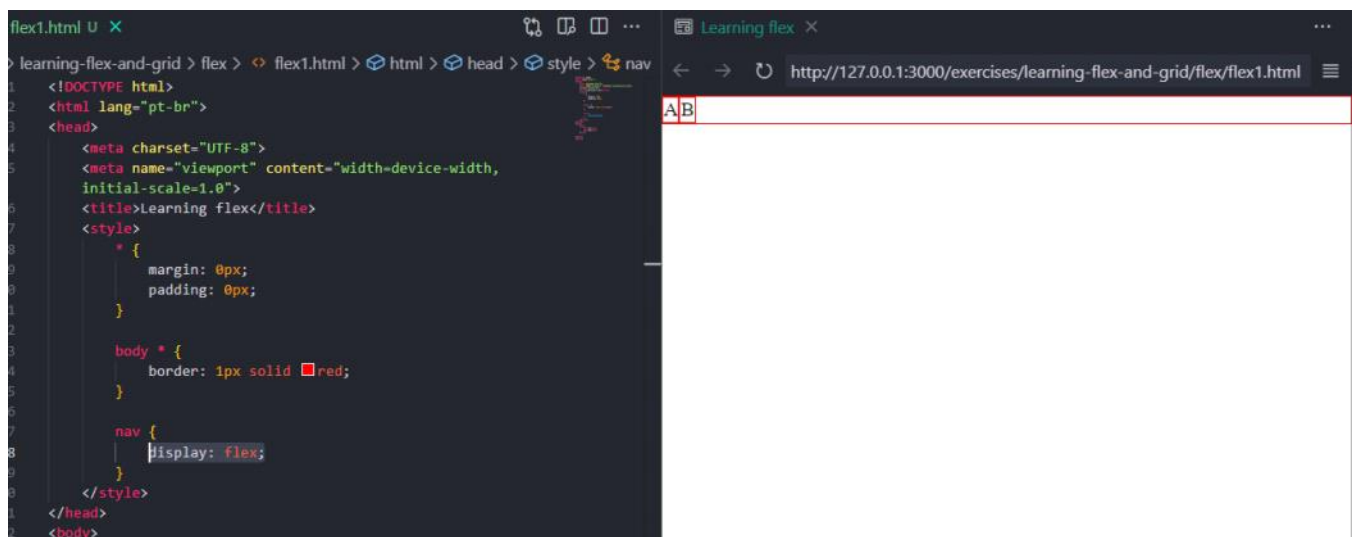
E onde entra o "display: flex" nisto?

O **"`display: flex`"** nos diz mais sobre os elementos que estão dentro de uma caixa do que a caixa em que aplicamos ele em si. Isto porque, durante o desenvolvimento, não vamos usar somente uma ou duas caixas para criar o conteúdo de um site, e sim diversas, muitas vezes aninhadas. Quando aplico o `display: flex`, os conteúdos de uma caixa (as caixas que estão dentro da caixa) são automaticamente organizadas horizontalmente, como se estivessem com `display: inline`.

Antes do flex:

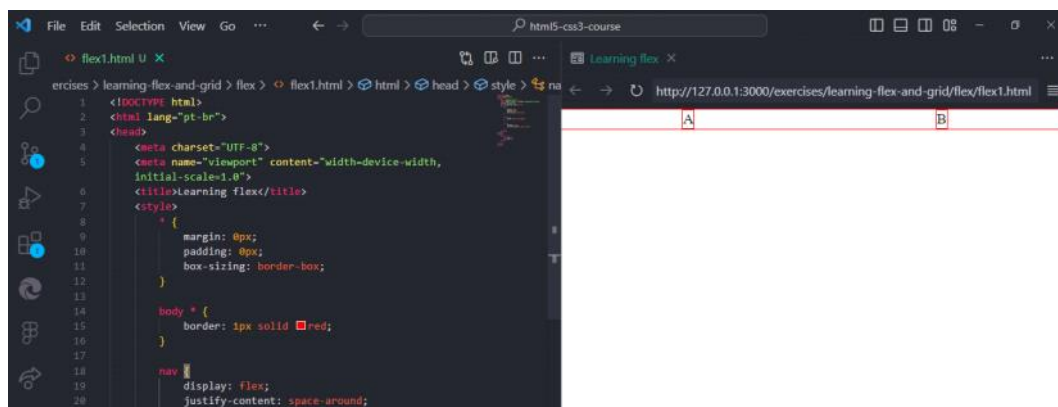


Depois do flex:



Usar o flex abre espaço para o surgimento de novas propriedades, como:

- Justify-content: coloca em posições como:
 - Left, center, right;
 - Flex-start, flex-end (trazem os elementos para o início e o fim do eixo principal da caixa, respectivamente);
 - Space-around (cria um espaço entre os elementos, em ambas as direções);
 - Space-between (cria um espaço interno entre os dois elementos);
 - Space-evenly (separa igualmente os elementos);

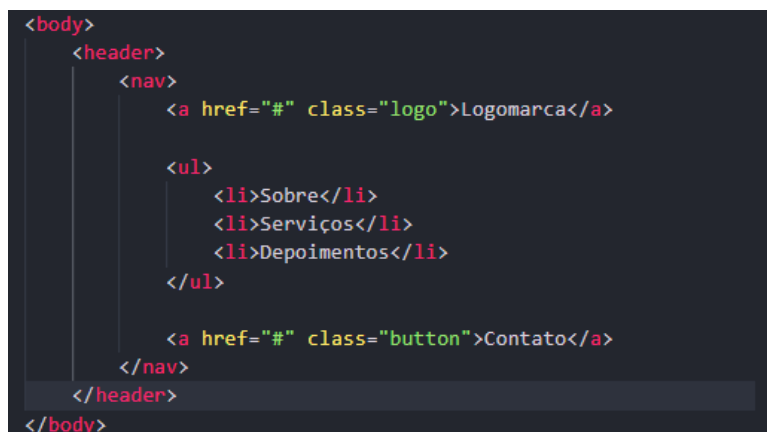


Criando menus

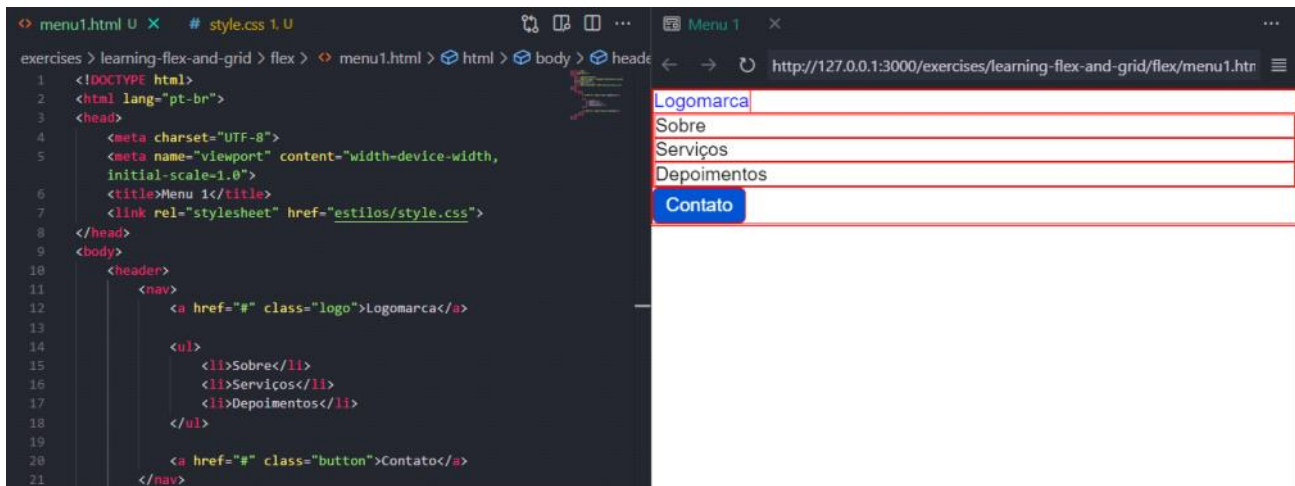


Tendo em vista os menus acima, já podemos imaginar que os elementos dispostos nele podem ser organizados usando o flex.

Podemos, inclusive, imaginar que cada parte deste menu representa uma coleção de elementos. Considerando o primeiro menu, vamos organizar o HTML do seguinte modo:

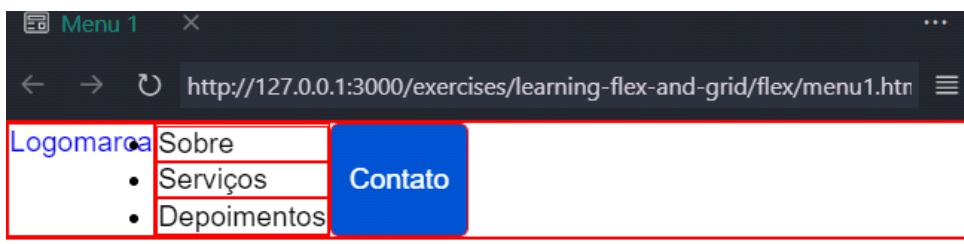


Vamos observar tudo como caixas:

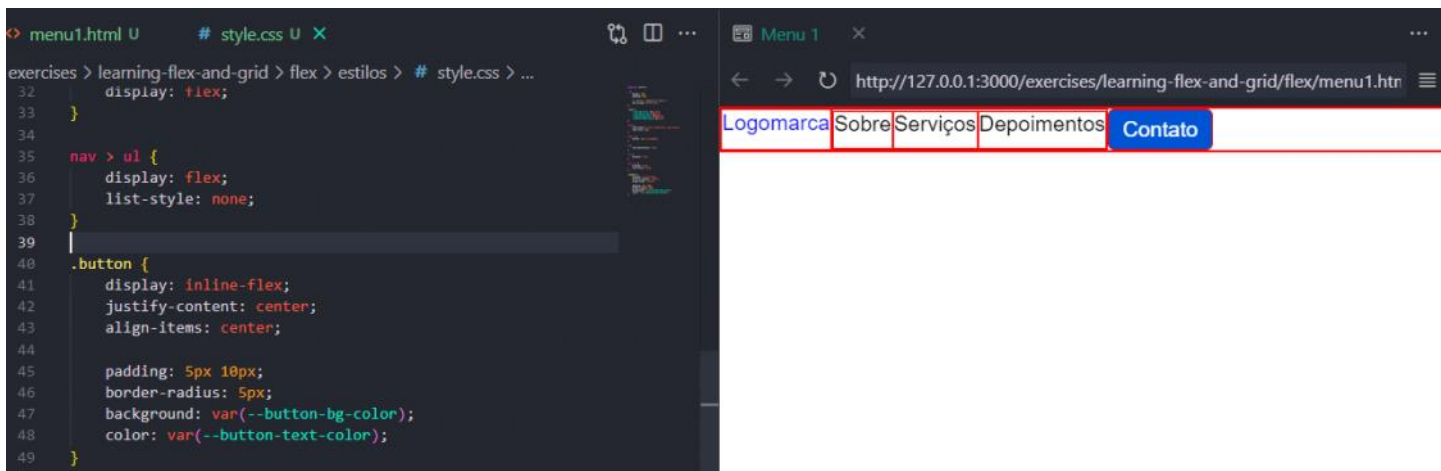


Perceba que já incluí um CSS para o botão e os links, mas não isto não vem ao caso agora.

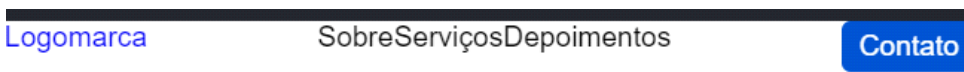
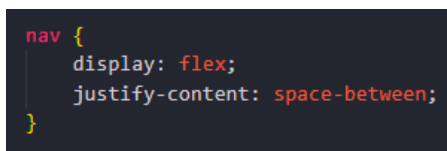
Se aplicar o *display: flex* na nossa `<nav>`, obteremos o seguinte resultado:



Vamos nos lembrar que o flex organiza em uma mesma direção os elementos de uma caixa. E é isto o que ocorre aqui. No entanto, as coisas estão feias. Vamos dar um retoques na nossa lista primeiro:



Todos os elementos estão bem juntinhos, certo? Isto porque não defini um espaço entre cada um deles. Vamos fazer isto com a `nav` primeiro.



Além disto, posso definir que os elementos fiquem centralizados de acordo com um eixo imaginário, que, por padrão é horizontal, usando a propriedade `align-items`. Se eu trocar o sentido do flex, usando `flex-direction` (de row para column), eu troco o sentido deste eixo também, o que muda a orientação dos elementos e a lógica da sua organização.

Voltando ao espaçamento, para distanciar os elementos da lista, basta usar um `gap`.

```

nav {
  display: flex;
  justify-content: space-between;
  align-items: center;

  background-color: var(--nav-bg-color);
  color: var(--nav-text-color);

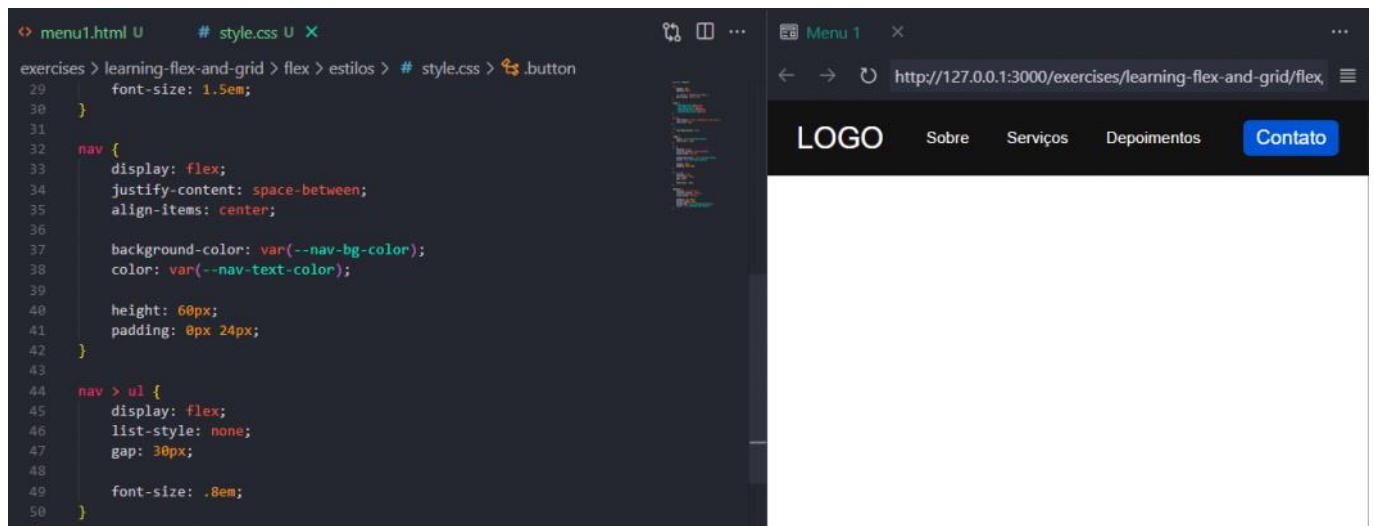
  height: 8vh;
}

nav > ul {
  display: flex;
  list-style: none;
  gap: 30px;
}

```



Bom, agora só falta adicionar os paddings, ajustar o tamanho das fontes, que o nosso menu está pronto.



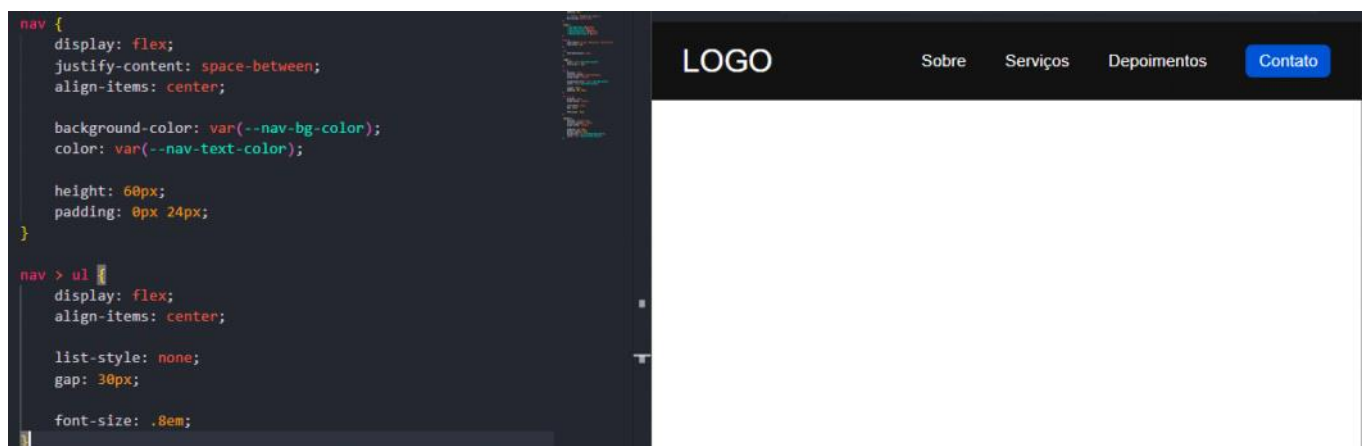
E sabe o que é melhor?

O flex é flexível. Ajustar a tela nos leva um espetáculo de responsividade nunca antes visto. É claro que, para versões mobile, as coisas vão ser diferentes e precisam ser tratadas com o devido cuidado. Mas o básico ele já faz.

Outros menus

Com apenas algumas mudanças no código eu consigo transformar este menu em outros. Vejamos.

Colocando o botão na lista e alinhando os itens dela ao centro do eixo:



Usando "*flex-direction: row-reverse*" eu consigo literalmente inverter a posição dos elementos.

```
nav {
  display: flex;
  justify-content: space-between;
  align-items: center;
  flex-direction: row-reverse;

  background-color: var(--nav-bg-color);
  color: var(--nav-text-color);

  height: 60px;
  padding: 0px 24px;
}
```

[Sobre](#)

[Serviços](#)

[Depoimentos](#)

[Contato](#)

LOGO