

Mão na Massa – CSS

Objetivo da Aula

Aplicar os conceitos de CSS no projeto.

Apresentação

Nesta aula, vamos explorar o `o.css` disponível no nosso **template**. A Folha de Estilos em Cascata (CSS) é uma linguagem baseada em regras utilizada para aplicar estilos (cor de fundo, borda, cor de texto, tamanho da fonte...) ao seu documento. As características visuais aplicadas aos elementos são representadas pelo que chamamos de **propriedades** e essas propriedades podem ser combinadas, trazendo possibilidades de personalização quase infinitas.

1. CSS

Na aula anterior, vinculamos o Bootstrap através de CDN's, lembra? Sendo assim, toda vez que a página for carregada, o servidor mais próximo da localidade da sua localidade se encarrega de fornecer o estilo da página. Por esse motivo, você poderá perceber uma certa lentidão no carregamento, dependendo da velocidade da sua internet. Esse modo de instalação é bem simples, pois não requer de nenhuma habilidade especial, porém o código é meio "fechado", não sendo possível fazer alteração de nenhuma variável ou classe do Bootstrap. Com isso, é uma prática comum termos, além do Bootstrap, um arquivo de `o.css` com formatações mais específicas. Veja abaixo o `o.css` disponível no nosso **template**:

```

1  body{
2    background-color: #eeeeee;
3  }
4
5  .footer-background{
6    background-color: rgb(204, 199, 199);
7  }
8
9  @media(max-width:991px){
10   #heading{
11     padding-left: 50px;
12   }
13   #prc{
14     margin-left: 70px;
15     padding-left: 110px;
16   }
17   #quantity{
18     padding-left: 48px;
19   }
20   #produc{
21     padding-left: 40px;
22   }
23   #total{
24     padding-left: 54px;
25   }
26 }

```

```

27 @media(max-width:767px){
28   .mobile{
29     font-size: 10px;
30   }
31   h5{
32     font-size: 14px;
33   }
34   h6{
35     font-size: 9px;
36   }
37   #mobile-font{
38     font-size: 11px;
39   }
40   #prc{
41     font-weight: 700;
42     margin-left: -45px;
43     padding-left: 105px;
44   }
45   #quantity{
46     font-weight: 700;
47     padding-left: 6px;
48   }
49   #produc{
50     font-weight: 700;
51     padding-left: 0px;
52   }

```

```

53     #total{
54         font-weight: 700;
55         padding-left: 9px;
56     }
57     #image{
58         width: 60px;
59         height: 60px;
60     }
61     .col{
62         width: 100%;
63     }
64     #zero-pad{
65         padding: 2%;
66         margin-left: 10px;
67     }
68     #footer-font{
69         font-size: 12px;
70     }
71     #heading{
72         padding-top: 15px;
73     }
74 }

```

Repare que este código usa diferentes propriedades. São elas:

- **width:** define a largura de um elemento que pode ser definido com valor fixo usando pixel (px) ou valor dinâmico usando porcentagem (%);
- **height:** define altura de um elemento que pode ser definido com valor fixo usando pixel (px) ou valor dinâmico usando porcentagem (%);
- **margin:** define o espaçamento externo de um elemento.
- **padding:** define o espaçamento interno, ou seja, espaço entre a borda e o conteúdo.
- **background-color:** define a cor de fundo do elemento.
- **font-size:** define o tamanho da fonte. Esta propriedade permite tais unidades: px (pixels), em (unidade relativa), pt (pontos), %, entre outras.
- **font-weight:** define a intensidade ou espessura da fonte. Esta propriedade pode ter os seguintes valores: **normal** | **bold** | **bolder** | **lighter**, entre outros.



Destaque

Quando as propriedades **padding** e **margin** recebem um único valor (padding: 2px;) alteram os quatro lados de um elemento: superior, direita, inferior e esquerda. Porém é possível especificar o valor de forma individual e separadamente (padding-top: 2px; padding-right: 4px; padding-bottom: 6px; padding-left: 8px;) ou em conjunto (padding: 2px 4px 6px 8px;), passando os 4 valores de uma vez.

Outra coisa que você deve ter percebido nesse código é que boa parte dele faz parte de **Medias Queries** (linhas 9 e 27).

Media query, nada mais é do que uma estrutura do CSS que permite aplicar estilizações específicas para uma página web de acordo com certas condições, fazendo assim com esta página possa se adequar ao layout de tela em diferentes tamanhos e tipos de mídia (MEDIA, 2022).

Sua sintaxe é basicamente composta pela expressão @media seguida de uma condição entre (), veja:

```
@media (condição) {
/* Estilizações CSS */
}
```

Esta condição resultará num resultado verdadeiro ou falso. Quando uma **media query** é verdadeira, as regras de estilos correspondentes são aplicadas, caso contrário as regras não serão aplicadas.

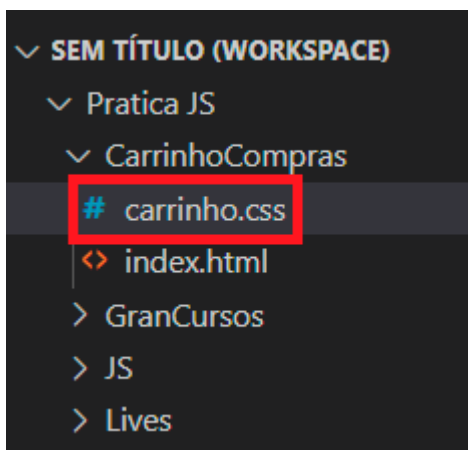


Destaque

Saiba mais sobre como criar expressões media query em: <https://www.treinaweb.com.br/blog/media-queries-o-que-sao-e-como-usar-no-css#:~:text=O%20que%20s%C3%A3o%20Media%20Queries,tamanhos%20e%20tipos%20de%20m%C3%ADdia.>

No nosso código estamos fazendo uso do que chamamos de **media features**, ela define a largura da janela. Usamos o max-width para definir a largura máxima de uma condição, definindo se as estilizações serão aplicadas conforme o tamanho da tela. Bom, vamos então vincular o css ao nosso código.

1º passo: crie um documento externo de.css.



2º passo: copie o código de css disponível no nosso template e cole neste documento que você criou.

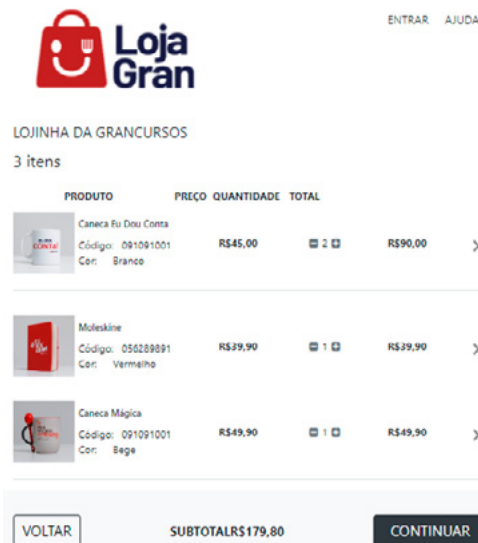


Fonte: <https://bbbootstrap.com/snippets/bootstrap-ecommerce-cart-page-template-91851305#>.

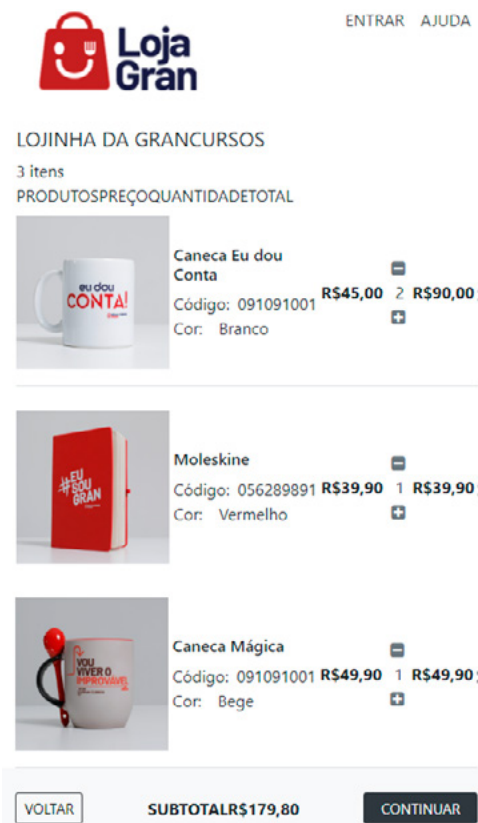
3º passo: Insira na linha 11 a tag que vinculará o HTML ao CSS.

```
11 <link rel="stylesheet" href="carrinho.css">
```

A princípio não teremos nenhuma mudança no visual da nossa página. Repare que vai continuar do mesmo jeito, mas ao redimensionar a tela você perceberá a diferença. Veja a janela redimensionada **com** o css:



Agora veja a janela redimensionada **sem** o css:



Percebeu a diferença? É nítido, não é mesmo? Logo, podemos concluir que o css do nosso template é basicamente voltado para a dimensão da janela.

Considerações Finais

Nesta aula, vimos que podemos trabalhar com CSS e Bootstrap ao mesmo tempo. Além disso, aprendemos um conceito novo chamado **media query**. Conhecimento nunca é demais, não é mesmo? Exploramos o código do nosso template e conhecemos muitas das propriedades utilizadas para estilizar o nosso carrinho. Legal, né? Com isso, temos certeza de que o conteúdo aprendido aqui é de fato utilizado na prática.

Materiais Complementares



Usando Medias queries: https://developer.mozilla.org/pt-BR/docs/Web/CSS/Media_Queries/Using_media_queries.

Referências

MEDIA. *Media queries, o que são e como usar no CSS?* 2022. Disponível em: <https://www.treinaweb.com.br/blog/media-queries-o-que-sao-e-como-usar-no-css#:~:text=O%20que%20s%C3%A3o%20Media%20Queries,tamanhos%20e%20tipos%20de%20m%C3%ADdia>. Acesso em: 30 nov. 2022.