

RETRO TUTORIAL

Este tutorial acompanha uma pasta. Ela está aí apenas para facilitar a organização do projeto. A estrutura da pasta deve ser mais ou menos assim:

```
//retro
|
|-----//img
|           |
|           |-----logo
|           |----//imagens para background
|           |           |
|           |           |----- typewriter, vehicle,
|           |           |         record-player, literature
|           |
|           |-----//ícones
|           |           |
|           |           |----- map-location
|           |           |-----//redes-sociais
|           |           |           |
|           |           |           |----- facebook, twitter
|           |           |           |instagram, pinterest
|
|-----//fonts
|
|-----//css
|           |
|           |----materialize.min.css
|
|-----//js
|           |
|           |----materialize.js
|
|-----retro.html
|-----retro.css
```

O arquivo **materialize.min.css** é o nosso materialize minificado, para usá-lo você deve referenciar o mesmo na head do **retro.html**. O mesmo deve ser feito com o **retro.css**, que é onde entrará o css da nossa página. (Não use CSS inline pfv).

A pasta **img** é onde vamos guardar todas as imagens do site. Em um projeto pequeno pode parecer besteira organizar tudo assim, bem como as próprias pastas, mas em um projeto grande isso faz total diferença, acredite.

Tudo aqui pode e deve ser acompanhado pela documentação do Materialize (<http://materializecss.com>), bem como pelos slides e material de apoio para o css, que você pode encontrar no www.w3schools.com.

Mas agora vamos ao que importa, a nossa página:



OBS.1: Não se esqueça de inicializar o Js e o JQuery (nesta ordem), se você estiver usando versão antiga. Para a versão mais atual (1.0.0-rc.2), não é necessário inicializar o JQuery.

Tudo isso você encontra no **getting started** do Materialize.

OBS.2: O que fica entre as tags `<script></script>` funciona mesmo quando colocamos em outros locais, mas, em geral, quando colocados ao final da tag body, o site tem melhor performance*. Lembre-se disso quando precisar usar alguma funcionalidade do Materialize que precise de inicialização especial em JavaScript

*<https://pt.stackoverflow.com/questions/1109/onde-devo-colocar-um-c%C3%B3digo-javascript-em-um-documento-html>

```
<html>
  <head>
    ...
  </head>
  <body>
    <nav>
      *navbar*
    </nav>
    <section>
      *conteúdo*
    </section>
    <footer>
      *footer*
    </footer>
    <script>
      ...
    </script>
  </body>
</html>
```

#BODY

No print dá para notar que temos uma imagem que acompanha a página inteira. Isso mesmo, essa imagem é background do body. Então precisamos escrever isso no nosso css.

Você não precisa usar esta imagem, coloquei até algumas outras na pasta como sugestão, mas se preferir pode escolher outra (desde que esteja no tamanho e qualidade certa para não ficar pixelizada).

Naturalmente, se ainda não há nada no body ela não irá aparecer, mas você pode pôr algo como “height: 100%” para ter certeza de que está tudo certo e testar a próxima parte.

#NAVBAR

Embora ela não esteja bem definida na imagem, nós temos sim uma navbar, mas ela é transparente. Lembra das classes de cores do Materialize? Bem, você vai encontrar essa opção lá.

Na navbar, além dos links, vai entrar a nossa logo, que está na pasta img. Lembre-se que o arquivo que você precisa está dentro de uma ou mais pastas, você deve escrever o “endereço” completo no html/css.

Nas seções da navbar (os links) usei a fonte Dosis, dá para achá-la aqui: <https://fonts.google.com/specimen/Dosis?selection.family=Dosis>. Pode ser que você não consiga mudar a fonte das seções só modificando a fonte geral usada na navbar (ok, você não vai conseguir indo por este caminho). Agora a tentação para usar um css inline pode ser grande, mas resista. Lembra daquilo do endereço completo dos arquivos? Pra usar CSS é parecido. Então, talvez não dê para mudar a fonte de toda a navbar como você estava pensando, mas pode funcionar dando um endereço mais detalhado do que quer modificar.

Se você fez tudo certinho e está vendo a nossa navbar deve ter se deparado com algo estranho: uma linha sombreada. Pois é, por default a navbar do Materialize tem essa sombrinha, mais precisamente um z-depth-1. Pode não aparecer no código, mas está lá. Para corrigir isso você precisa mudá-lo para 0.

#ROW WHITE

Row White é esta faixa branca onde está o nosso texto. Dentro dela você vai precisar usar um pouco da GRID do materialize. (Se conseguir, tente já deixar responsivo).

É provável que a row white “flutue” e não fique na posição do print, para corrigir isso você vai precisar acrescentar margem a ela (eu usei 75vh).

Esse texto é puro lero-lero, gerado no site <https://br.lipsum.com/>. Exceto pelo olho (essa citação aí no meio), ele está na fonte padrão do materialize (Roboto), a Dosis não é muito legível para textos assim. Além disso, (dica) sempre que tiver texto ou algum conteúdo que parece meio “jogado” no layout, veja se um container pode ajudar. Outra dica para organizar melhor o espaço em branco é usar padding (inclusive posso ou não ter usado no topo desta row). Também é bom saber a diferença e quando usar margin e padding.

#BACKGROUND ATTACHMENT

Bom, se você já chegou até aqui, preciso te contar um detalhe. Percebe que no print a imagem de fundo está estranha? Cortada? Na página real ela não está, mas esse efeito não pode ser captado no print pois requer movimento.

Lá no nosso css, na parte em que você colocou a sua background image você vai precisar registrar também o attachment. Neste caso: "background-attachment: fixed;". Essa função fixa a imagem de fundo enquanto o conteúdo da página fica "livre", como se percorresse a imagem. É uma boa saída para quando você tem uma imagem pequena para fundo de uma página muito longa, cheia de conteúdo, evitando que ela fique distorcida e estranha.

#FOOTER

Então já estamos no footer, não falta muito. Assim como aconteceu entre a navbar e a row white, você deve ter percebido que aqui também temos um espaço entre a row white e o footer. Então, se você já conseguiu resolver este problema lá, vai ser fácil. Se não conseguiu, dê uma olhada na parte de margens do w3schools.

O nosso footer não tem muita coisa (até porque essa página não tem conteúdo de verdade, né), mas tem o endereço físico e os ícones para os links das redes sociais ligadas à página (o que é bem comum em muitos sites). Estes eu peguei no **flaticon.com**, você não vai precisar pegar pois eles já estão organizadinhos na pasta ícones. Lembrando que são links, por mais que no nosso site não levem a lugar algum.

Pode ser que os ícones tenham dado um probleminha de tamanho na sua página. Neste caso você pode precisar regular a altura ou largura das imagens.

Por fim, se por acaso aparecer algum pedaço da imagem depois do footer tem algo errado aí. Verifique se não esqueceu do copyright ou se alguma tag ficou aberta

#O FIM É O COMEÇO

Se você conseguiu terminar esse tutorial em tempo record, pode aproveitar para fazer a página de contato ou a flea market e ou então já pode começar a tarefa desta semana.

Se não for o caso, tudo bem. E se tiver alguma dúvida, é só escrever para: **lobo.loise@ejcm.com.br** ;)

Este tutorial foi inspirado no tuto

django girls