



EBOOK

20 DICAS CSS PARA TURBINAR SEUS SKILLS

Turbine seu conhecimento com 20 dicas incríveis

SOBRE MIM

AUTOR

Olá, meu nome é Diogo Machado, desenvolvedor de software com mais de 7 anos de experiência prática na área, em minha carreira já desenvolvi muitos aplicativos, desenhei telas, modelei banco de dados, ministrei treinamentos e palestras.

Já fui destaque na revista Galileu nº 241 como um dos 10 jovens gênios do Brasil, no Panamá e África do Sul levei um projeto educacional que desenvolvi para o concurso Educadores Inovadores da Microsoft ganhando em 1º e 2º lugar.

Em 2018, publiquei o meu primeiro livro técnico pela editora Casa do Código, que aborda a construção de aplicativos híbridos avançados com Cordova e PhoneGap.



Diogo Souza Machado

Autor, Instrutor online, Analista de Sistemas na Unimed Sul Capixaba



CONTEÚDO

INTRODUÇÃO

Neste conteúdo exclusivo reuni várias dicas para você incrementar o seu conhecimento em CSS e de ferramentas que vão te auxiliar no seu trabalho.

O conteúdo apresentado neste ebook é fruto de muitas horas de desenvolvimento de interfaces, espero que aproveite e possa absorver algo que vá lhe acrescentar.

Se você curtir este material e quiser trocar ideia comigo, não deixe de me chamar no e-mail **diogo@diogomachado.com** ou em minhas redes sociais.

Obrigado e bons estudos!

DIREITOS AUTORAIS

Este guia está protegido por leis de direitos autorais. Todos os direitos sobre o guia são reservados. Você não tem permissão para vender este guia nem para copiar/reproduzir o conteúdo do guia em sites, blogs, jornais ou quaisquer outros veículos de distribuição e mídia. Qualquer tipo de violação dos direitos autorais estará sujeita a ações legais.

1. Aprenda a usar o seu navegador

Essa não é uma dica sobre CSS, mas é para quem trabalha com web, navegadores modernos como Google Chrome, Firefox e Safari possuem ferramentas para desenvolvedores.

Aprender a abrir e inspecionar elementos da página e qual a estilização está sendo renderizadas na página é uma poderosa ferramenta que você vai ter como aliada na construção de componentes CSS.

<https://developers.google.com/web/tools/chrome-devtools/>

2. Escolha bem o nome de suas classes

Assim como na escrita de um livro o nome de um personagem é importante, quando estamos escrevendo nome para classes CSS é importante que ele carregue sentido e seja único.

Nomes muito genéricos podem conflitar com outras folhas de estilo da sua página, portanto pense e analise qual a função que a classe que está criando terá para a página.

3. Entenda o que é especificidade

A linguagem de marcação CSS faz um calculo quando o navegador está renderizando os seletores, esse calculo atribui um peso para o seletor, esse peso determina a especificidade do seletor.

Quanto mais específico ele for, maior será a sua especificidade, o editor Visual Studio Code possui um suporte legal para o desenvolvedor visualizar o calculo da especificidade, basta posicionar o mouse no seletor CSS.

4. Alinhando elementos ao centro

Alinhamento sempre foi um terror para qualquer um que trabalha com web, o fato é que ele ainda é porque precisamos entender como funciona o sistema de display.

Com a vinda da especificação Flexbox, as coisas ficaram mais fáceis de se alinhar, existem um site que ajuda muito nesse trabalho de alinhar algo no CSS, o seu nome é <http://howtocenterincss.com>

5. Mantenha o código dos componentes próximos

Uma dica de organização, independente de como você lida com o CSS é manter o código próximo dos seletores de um determinado componente.

```
/* Componente cabecalho */  
.cabecalho{  
    background: red;  
}  
  
.cabecalho-body{  
    padding: 1rem;  
}  
/* Fim */
```

6. Use fontes web com cuidado

Fontes web são agradáveis e bonitas, mas é importante não exagerar para não atrapalhar o carregamento da sua página, o Google fornece um serviço de fontes web super fácil de usar, é o <http://google.com/fonts>, nele é possível escolher diversas fontes e *plugar* na sua página HTML.

7. Busque conteúdo em sites reconhecidos na área

Assim como qualquer outra coisa que fazemos é sempre bom buscar conteúdo novo (de qualidade), no caso do CSS o meu conselho é que você busque por sites que já são **reconhecidos** há muitos anos na área, vou deixar o link de alguns que recomendo:

Conteúdo fantástico

<http://tympanus.net/codrops/>

Os experimentos deles são incríveis

<http://codyhouse.co/>

Conteúdo super atual, site clássico de CSS

<https://css-tricks.com/>

Muito conteúdo PT-BR de CSS

<https://tableless.com.br/categories/css/>

Esse é histórico, Maujor, um dos maiores nomes do CSS no Brasil

<https://maujor.com/>

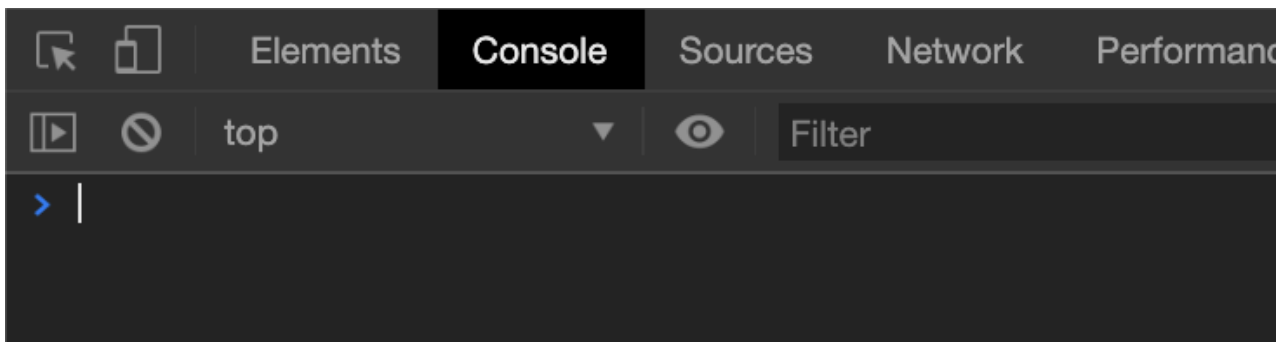
Ferramenta online de experimentos em HTML, CSS e JS

<https://codepen.io/>

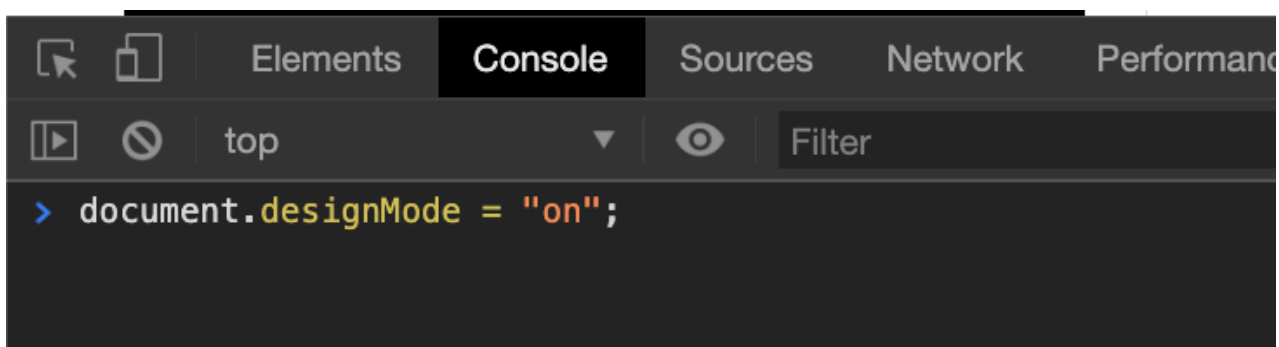
8. Use o modo designer do navegador

Existe um truque que podemos fazer em qualquer página para colocarmos todo o conteúdo de texto em modo de edição.

Para fazer isso você deve abrir o console do navegador, supondo que você está no Google Chrome, em uma página qualquer, abra a ferramenta de desenvolvedor na página que deseja editar, e vá em console.



Com o Console aberto, digite:



Agora você será capaz de editar qualquer texto da página.

9. Aprenda a diferença de REM e EM

Existe uma diferença nesses dois formatos de tamanho, o EM sofre influência do tamanho definido no elemento pai, exemplo:

```
.box{  
  font-size: 22px;  
}
```

```
.box h1{  
  font-size: 1em;  
}
```

Nesse caso, a fonte do elemento h1 terá o tamanho de 22px, porque o seu elemento pai é a classe .box, já com o elemento REM, que significa Root EM, estamos sempre fazendo referência ao elemento root da página, que será o seletor *html*

```
html{  
  font-size: 16px;  
}
```

```
.box{  
  font-size: 22px;  
}
```

```
.box h1{  
  font-size: 1rem;  
}
```

Nesse caso, 1REM terá o valor de 16px, porque o elemento root da página é o seletor *html*, e o valor de REM sempre será associado a ele, independente da influência de outros elementos.

10. Use sites para te ajudar com cores

Na prática é muito difícil a gente lembrar e decorar as cores em HEX ou RGB, eu diria que é loucura fazer isso, para isso existem diversos sites que facilitam a nossa vida na hora de encontrar cores bem bonitas para os elementos, aqui vão dois que uso muito:

Encontro a variação de uma determina cor em hex

<https://0to255.com>

Paletas de cores flat, super legal

<https://flatuicolors.com>

11. Cuidado com novas especificações CSS

O CSS é uma linguagem em evolução, existe o W3C e outras empresas grandes discutindo novas APIs e documentações, é importante usar esses recursos novos com cuidado, para isso existe um site que diz o nível de aceitação que determinado recurso está, o site é o <https://caniuse.com>

12. Como criar pontas em div com CSS



Um recurso bacana que é possível fazer no CSS é criar pontas em uma div, o conceito não é tão claro de aprender, vamos ao exemplo:

```
.bubble:after {  
  content: "";  
  position: absolute;  
  border-style: solid;  
  border-width: 15px 15px 0;  
  border-color: #FFFFFFF transparent;  
  display: block;  
  width: 0;  
  z-index: 1;  
  bottom: -15px;  
  left: 110px;  
}
```



Perceba que o que define o tamanho da seta é o **tamanho das bordas**. A cor também é definida pela borda.

Existe um gerador muito legal nesse link
<https://leaverou.github.io/bubbly/>

13. Criando bordas arredondadas com CSS



Um efeito muito interessante é o de **bordas redondas**, no CSS isso já é possível há algum tempo usando *border-radius*.

```
.card{  
  border-radius: 5px;  
}
```

14. RGBA e as cores com opacidade

O formato de cor RGBA é especial por aceitar um parâmetro que dita o nível de opacidade de uma cor, esse recurso é muito útil para tornar as cores menos chapadas na hora de fazer o layout.

A opacidade é o último parâmetro, e o valor maior é 1.0, você pode experimentar usar valores menores como 0.5, 0.35 e ver a diferença.

```
.conteudo{  
  color: rgba(0,0,0,0.5)  
}
```

15. Meta viewport básica

Uma forma de dizer ao viewport (área visível do navegador) de como se comportar é atribuir a meta viewport.

A forma mais comum de fazer seu site se adaptar é usando essa meta, é o primeiro passo.

A definição mais comum é:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

16. O mínimo para uma imagem se adaptar

Existem milhares de soluções para adaptar imagens em sites responsivos (ou sistemas), mas a forma mais simples e básica para quem está dando os primeiros passos, é atribuir a imagem com tamanho total a viewport:

```
.img-responsiva{  
  width: 100%;  
}
```

Lembrando que essa não é a melhor solução, mas é um passo que pode ser dado caso não possa implementar algo mais robusto.

17. Diferença de hidden para visibility

Por horas você irá precisar tirar da tela um elemento, e existe diferença entre usar **display: hidden** e **visibility: hidden**, a diferença está que, no caso do display, a área onde o elemento ocupa será liberada, portanto os elementos da página vão se reajustar.

Já no caso do visibility, a área onde o elemento está será mantida, e no lugar aparecerá uma área vazia.

18. Entenda de uma vez o que é DOM

Quando se programa um sistema para web, a todo momento vamos manipular o DOM (Modelo de Objeto de Documento).

DOM é a representação da árvore de elementos HTML em memória no navegador.

19. Fixed vs Absolute

O elemento **display: fixed** é fixo em relação a *viewport*, portando não importa aonde ele esteja disposto no HTML, ele estará sendo alinhado a parte visível do browser.

Já o elemento **display: absolute** é relativo ao elemento pai que ele está contido, nesse caso ele irá ficar fixo a um elemento aonde ele está dentro.

20. Aplique acessibilidade na web

As vezes os projetos esquecem que a web é para todos, e quando falamos todos incluímos as pessoas que possuem limitações para navegar em um site.

O fato é que existem vários recursos para tornar os sites com suporte mais adequado e acessível, aqui vai alguns links:

Orientações do W3C

<https://www.w3.org/TR/wai-aria-practices-1.1/>

Artigo muito legal escrito pelo desenvolvedor *Felipe Fialho*

<https://felipefialho.com/blog/2016/sobre-wai-aria-acessibilidade-e-semantica>

CONTATO

CONECTE-SE COMIGO

Minha vida, dicas curtas, desabafos

<https://www.instagram.com/diogom>

Dou dicas, falo de programação e interajo com developers

https://twitter.com/tec_diogo

Artigos sobre programação

<https://www.diogomachado.com>

Imersão CSS - Faça um curso avançado em CSS

<https://imersaocss.com>

Diogo Souza Machado

Autor, Instrutor online, Analista de Sistemas na
Unimed Sul Capixaba

