

Programação Web – FrontEnd – CSS

Índice

Algumas palavras.....	2
1 - Introdução ao CSS.....	3
2 - Seletor, propriedade e valor.....	4
3 - CSS como atributo style nos elementos.....	5
4 - CSS na tag style.....	6
5 - CSS em arquivo externo.....	7
6 - Tipos de Seletores.....	8
7 - Unidades de medida.....	10
8 - Cores.....	11
9 - Fontes.....	12
Comentários:.....	12
10 - Seletores Agrupadas.....	13
11 - Seletores Aninhados.....	14
12 - Pseudo Classes.....	15
13 - Primeiro filho.....	16
14 - Forma abreviada das propriedades.....	17
15 - Imagens de Fundo.....	18
16 - Resolvendo Conflitos de CSS.....	19
17 - Escrevendo Código CSS.....	20
18 - Exemplos para o background.....	21
19 - Exemplo mostrando o boxmodel.....	22
20 - Formulário de Contato com Estilo.....	25
21 - height e width.....	28
22 - Alinhamento na horizontal e na vertical.....	29
23 - Z-index (camadas).....	32
24 - Design Responsivo.....	33
25 - Exemplo de Tabela Elegante com Estilo.....	34
26 - Dicas de CSS.....	36
27 - Estilo para manter um rodapé fixo no fundo da página.....	38
28 - Dicas Extras sobre Seletores.....	39
29 - Setas em CSS.....	40
30 - Exemplos com Seletores.....	42
31 - Material para Download.....	44
32 - Referências.....	45

Algumas palavras

Este curso é destinado aos servidores do DNOCS, assim também como aos estagiários e terceirizados. Este curso faz parte do módulo 1, que é composto pelos conhecimentos básicos (HTML, CSS e JavaScript) e pelos Frameworks (Bootstrap e jQuery). A estes módulos se seguirão outros.

Sua realização acontece no setor de Suporte do DNOCS.

O objetivo principal é preparar o aluno para a construção de sites e aplicativos, que acontecerá em futuros cursos usando o CMS Joomla para a criação de sites e o framework CakePHP para a criação de aplicativos, assim como o OpenCart para a criação de lojas virtuais e também programação Mobile.

O aprendizado destas ferramentas tem como objetivo complementar os conhecimentos do aluno permitindo que trabalhe no órgão no setor de informática, assim como também preparar o mesmo para ter um trabalho extra por conta própria ou até venha a trabalhar na área de programação lá fora.

Seja sinceramente bem vindo(a). Aproveite e divirta-se.

Professor Ribamar FS – <http://ribafs.org>, abril de 2019

Citação do livro Eloquent JavaScript

Eu não esclareço os que não estão prontos para aprender, nem desperto aqueles que não estão ansiosos para dar uma explicação a si próprios. Se eu apresentei um canto da praça, e eles não podem voltar para mim com os outros três, eu não deveria passar por estes pontos novamente. (Confúcio)

Curso em cima de um Projeto

Este curso será ministrado em cima de um projeto. Trabalharemos com um site criado em HTML, com praticamente todos os recursos que vimos no curso de HTML e mais alguns. Em cima deste arquivo HTML faremos nossas intervenções com CSS para deixar a página mais agradável e com um layout como desejamos.

Começaremos analisando um desenho que contém o layout recebido de um cliente e em cima deste layout aplicaremos o CSS. Os arquivos para isto estão na pasta Aula1 do CSS no servidor. Depois trabalharemos com o arquivo index.html onde também aplicaremos o CSS.

1 - Introdução ao CSS

CSS significa Cascating Style Sheet, Folhas de Estilo em Cascata.

O CSS define o estilo, a formatação da página e também o layout e posicionamentos de elementos.

Por padrão o HTML desenha cada elemento um abaixo do outro e nunca do lado. O CSS veio para resolver isso e tem um forte controle sobre a posição dos elementos.

CSS é chamado de linguagem Cascading Style Sheet e é usado para estilizar elementos escritos em uma linguagem de marcação como HTML. O CSS separa o conteúdo da representação visual do site. Pense na decoração da sua página. Utilizando o CSS é possível alterar a cor do texto e do fundo, fonte e espaçamento entre parágrafos. Também pode criar tabelas, usar variações de layouts, ajustar imagens para suas respectivas telas e assim por diante.

É uma linguagem para estilos que define o layout de documentos . Por exemplo, controla fontes, cores, margens, linhas, alturas, larguras, imagens de fundo, posicionamentos e muito mais. Aguarde e você verá!

Vantagens:

- controle do layout de vários documentos a partir de uma simples folha de estilos;
- maior precisão no controle do layout;
- aplicação de diferentes layouts para servir diferentes mídias (tela, impressora, etc.);
- emprego de variadas, sofisticadas e avançadas técnicas de desenvolvimento.

CSS foi desenvolvido pelo W3C (World Wide Web Consortium) em 1996, por uma razão bem simples. O HTML não foi projetado para ter tags que ajudariam a formatar a página. Você deveria apenas escrever a marcação para o site.

Tags como foram introduzidas na versão 3.2 do HTML e causaram muitos problemas para os desenvolvedores. Como os sites tinham diferentes fontes, cores e estilos, era um processo longo, doloroso e caro para reescrever o código. Assim, o CSS foi criado pelo W3C para resolver este problema.

Se usado com moderação, CSS pode viabilizar uma ótima experiência ao desenvolvedor e usuários das páginas web.

Com cada regra é possível estilizar o conteúdo todo ou somente determinados elementos.

Teste

Vejamos como ficam alguns sites quando retiramos o CSS deles.

Faça o download de

<https://github.com/BlackrockDigital/startbootstrap-creative/archive/gh-pages.zip>

Descompacte em qualquer pasta e abra o index.html e comente os includes dos dois arquivos CSS, nas linhas 21 e 24.

2 - Seletor, propriedade e valor

O HTML tem tags e o CSS tem seletores.

Seletores são nomes usados para mudar o específico estilo de um elemento.

Cada seletor tem pelo menos uma propriedade e seu respectivo valor:

```
seletor {  
  propriedade: valor;  
}
```

Ver seletores no W3C

https://www.w3schools.com/cssref/css_selectors.asp

Os seletores são a alma do CSS e você precisa dominá-los.

```
body{  
  color: white;  
}
```

body - seletor

color - propriedade

white - valor da propriedade

Indica que em todo o <body> a cor da fonte será branca, a não ser que algum outro futuro seletor venha alterar a cor da fonte.

3 - CSS como atributo style nos elementos

Uma forma que deve ser evitada mas que use seu bom senso para usar, caso acredite que vale seu uso é o CSS em tags.

Veja este exemplo:

```
<!DOCTYPE html>
<html lang="pt">
  <head>
    <meta charset="utf-8"/>
    <title>Título da página</title>
  </head>
  <body style="background-color:black;">

<h1 style="font-size:28px;color:green;">Título do Conteúdo</h1>

<p style="color:red">Aqui é um texto dentro da TAG parágrafo do HTML</p>

</body>
</html>
```

Dica: Sempre tecle CTRL+U para visualizar o código da página. Assim verá todo o código HTML, CSS e JavaScript da página.

Algumas poucas páginas criptografam algo da página mas a maioria do código está aberta.

Rodando no WebMaker

Execute o exemplo acima

Tecle Ctrl+U para visualizar o código

Selecione apenas as 3 linhas abaixo:

```
<body style="background-color:black;">
<h1 style="font-size:28px;color:green;">Título do Conteúdo</h1>
<p style="color:red">Aqui é um texto dentro da TAG parágrafo do HTML</p>
```

Tecle Ctrl+C para copiar e abra o webmaker em:

<https://webmakerapp.com/app/>

Na região HTML cole as 3 linhas com Ctrl+V e veja no painel da direita o resultado imediatamente após colar.

4 - CSS na tag style

Uma forma que também deve ser evitada mas que use seu bom senso para usar, caso acredite que vale seu uso é o CSS na tag style.

Veja este exemplo:

```
<!DOCTYPE html>
<html lang="pt">
  <head>
    <meta charset="utf-8"/>
    <title>Título da página</title>
  <style>
    body{
      background-color: black;
    }

    h1{
      text-align: center;
      color: green;
      font-size: 32px;
    }

    p{
      color: red;
    }
  </style>
</head>
<body>

<h1>Título do Conteúdo</h1>

<p>Aqui é um texto dentro da TAG parágrafo do HTML</p>
```

Dica: Sempre tecle CTRL+U para visualizar o código da página. Assim verá todo o código HTML, CSS e JavaScript da página.

Algumas poucas páginas criptografam algo da página mas a maioria do código está aberta.

5 - CSS em arquivo externo

O uso mais importante do CSS e o mais frequente em sites profissionais é o arquivo externo contendo o código CSS e com extensão .css

```
<link rel="stylesheet" type="text/css" href="estilo.css">
```

Veja este exemplo:

```
<!DOCTYPE html>
<html lang="pt">
  <head>
    <meta charset="utf-8"/>
    <title>Título da página</title>
    <link rel="stylesheet" type="text/css" href="css_externo.css">
  </head>
</body>

<h1>Título do Conteúdo</h1>

<p>Aqui é um texto dentro da TAG parágrafo do HTML</p>
```

Arquivo **css_externo.css**:

```
body{
    background-color: black;
}

h1{
    text-align: center;
    color: green;
    font-size: 32px;
}

p{
    color: red;
}
```

Dica: Sempre tecle CTRL+U para visualizar o código da página. Assim verá todo o código HTML, CSS e JavaScript da página.

6 - Tipos de Seletores

- id (único por página)
- classe (um ou mais por página)
- elemento do HTML

Como cada navegador tem algumas propriedades default é interessante realizar um reset destas propriedades para que nossos seletores atuem como esperado.

Agirá sobre todo o documento.

Seletor universal - *

```
* {  
  margin: 0;  
  padding: 0;  
}
```

Com o CSS3, é possível elaborar cantos arredondados, sombras, efeitos gradientes, animações e efeitos de transição, dentre outras opções.

Criando classes

```
.minhaClasse {  
  color: red;  
}
```

Usando:

```
<p class = "minhaClasse">Este parágrafo é vermelho</p>
```

Criando ids

```
#meuID {  
  color: blue;  
}
```

Usando:

```
<h1 id = "meuID">Este título é azul</h1>
```

Existem alguns poucos recursos no HTML para formatar o conteúdo do site, como:

- b:

```
<b>Este parágrafo está em negrito por conta do elemento b, do HTML</b>
```

- font:

```
<font color="#FF0000" face="Verdana, Arial, Helvetica, sans-serif"><i>Texto formatado com o  
HTML</i></font>
```


Devemos evitar usar estes recursos do HTML e ao contrário usar CSS, assim:

```
<style type="text/css">
.meuEstilo {
  font-family: Verdana, Arial, Helvetica, sans-serif;
  font-weight: bold;
  color: #FF0000;
}
</style>

<p class="meuEstilo">Texto formatado com CSS</p>
```

7 - Unidades de medida

Absolutas:

cm, mm, in, px, pt, pc

Relativas:

em Relativo para o tamanho da fonte do elemento (2em significa 2 vezes o tamanho da fonte atual)

ex Relativo para o x-height da fonte atual (raramente usado)

ch Relativo para o width do "0" (zero)

rem Relativo para o tamanho da fonte do elemento raiz

vw Relativo para 1% do width do viewport*

vh Relativo para 1% da height do viewport*

vmin Relativo para 1% a menor dimensão do viewport's*

vmax Relativo para 1% da maior dimensão do viewport's*

% Relativo para o elemento pai

Dica: Quando o valor for zero não requer a unidade.

8 - Cores

O CSS tem 16.777.216 cores a sua disposição.

Podemos representar as cores de várias formas (todas as abaixo produzem a mesma cor, vermelho):

- red
- rgb(255,0,0)
- rgb(100%,0%,0%)
- #ff0000
- #F00

Alguns nomes de cores:

aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white e yellow

Transparent também é válido.

As cores podem ser aplicadas usando:

color

e

background-color

9 - Fontes

font-size

font-weight (negrito ou não). Valores:

normal ou bolder, lighter, 100, 200, 300, 400 (tipo normal), 500, 600, 700 (tipo bold), 800 ou 900.

font-style: italic

Comentários:

Os comentários do CSS são:

```
/*
```

```
Para
```

```
várias
```

```
linhas
```

```
*/
```

Ou

```
/* Assim */
```

10 - Seletores Agrupadas

Várias classes e um elemento agrupados com uma única propriedade:

No exemplo abaixo, h2, Classe1 e Classe2 terão a cor vermelha no texto

```
h2, .Classe1, .Classe2 {  
  color: red;  
}
```

11 - Seletores Aninhados

No exemplo abaixo criamos o id topo e em seguida tratamos os h1 dentro dele e os p

```
#topo {  
    background-color: #ccc;  
    padding: 1em  
}
```

```
#topo h1 {  
    color: blue;  
}
```

Todos os h1 dentro de #topo serão de cor blue

```
#topo p {  
    color: red;  
    font-weight: bold;  
}
```

Todos os p dentro de #topo serão de cor red e negrito

```
div p strong a {  
    color: red;  
}
```

Somente o a dentro de strong, que está dentro de p, que está dentro de div

```
#top p {  
    color: red;  
    font-weight: bold;  
}
```

Somente o p dentro de #top

12 - Pseudo Classes

```
selector:pseudo_class {  
  property: value;  
}
```

Exemplos:

```
a:link {  
  color: blue;  
}
```

Mudar a cor de todos os links para azul

```
a:visited {  
  color: purple;  
}
```

Mudar os links visitados para purple

```
a:active {  
  color: red;  
}
```

Mudar os links ativos para red

```
a:hover {  
  text-decoration: none;  
  color: blue;  
  background-color: yellow;  
}
```

Aplicar quando os links receberem o foco do ponteiro do mouse (hover)

```
input:focus, textarea:focus {  
  background: #eee;  
}
```

13 - Primeiro filho

```
p:first-child {  
  font-weight: bold;  
  font-size: 40px;  
}
```

Quando desejamos o estilo somente no primeiro parágrafo

```
p {  
  font-size: 12px;  
}
```

```
p:first-letter {  
  font-size: 24px;  
  float: left;  
}
```

Somente o primeiro será 24px e left e os demais com 12px

```
p:first-line {  
  font-weight: bold;  
}
```

CSS3 adicionou last-child, target, first-of-type e outros.

14 - Forma abreviada das propriedades

```
p {  
  margin-top: 1px;  
  margin-right: 5px;  
  margin-bottom: 10px;  
  margin-left: 20px;  
}
```

Pode ser escrita assim:

```
p {  
  margin: 1px 5px 10px 20px;  
}
```

padding e border podem também ser usado assim.

Dica: Podemos escrever apenas com dois valores:

```
padding: 1em 10em;
```

Neste caso o primeiro é top e bottom e o segundo é right e left.

```
p {  
  border-width: 1px;  
  border-color: red;  
  border-style: solid;  
}
```

Assim:

```
p {  
  border: 1px red solid;  
}
```

```
p {  
  font: italic bold 12px/2 courier;  
}
```

Combina font-style, font-weight, font-size, line-height e font-family.

```
p {  
  font: 14px/1.5 "Times New Roman", times, serif;  
  padding: 30px 10px;  
  border: 1px black solid;  
  border-width: 1px 5px 5px 1px;  
  border-color: red green blue yellow;  
  margin: 10px 50px;  
}
```

15 - Imagens de Fundo

```
body {  
  background: white url(http://www.html5dog.com/images/bg.gif) no-repeat top right;  
}
```

- background-color, which we have come across before.
- background-image, which is the location of the image itself.
- background-repeat, que repete a imagem. Seus valores são:
 - repeat - repete a imagem em grade
 - repeat-y - repete apenas no eixo y
 - repeat-x - repete no eixo-x
 - no-repeat - não repete
- background-position - que pode ser top, center, bottom, left, right, um comprimento ou porcentagem ou uma combinação como top right

16 - Resolvendo Conflitos de CSS

Se tivermos dois seletores para um mesmo elemento:

```
p { color: red }  
p { color: blue }
```

O elemento ficará blue, pois é a última/mais recente cor escolhida.

Se tivermos dois arquivos com seletores idênticos, também o do arquivo listado por último será o válido.

Display

Inline - mostra os elementos em linha

```
li { display: inline }
```

Block - mostra em blocos, cada um numa linha

```
#navigation a {  
  display: block;  
  padding: 20px 10px;  
}
```

```
#navigation, #related_links { display: none }
```

display: none e visibility: hidden

17 - Escrevendo Código CSS

O código CSS é escrito em 3 tipos de seletores, rescrevendo elementos do HTML e também criando classes e IDs.

```
seletor{  
    propriedade: valor;  
}
```

Reescrevendo elementos

```
body{  
    background-color: black;  
}
```

Criando classes

```
.classe1{  
    color: red;  
}
```

Criando ID

```
#meuid{  
    color: blue;  
}
```

Importante

Classes e IDs são seletores criados para serem usados junto ao HTML e então alterar sua formatação ou posições. Já os seletores com elementos estes já alteram automaticamente o comportamento dos respectivos elementos no HTML, sem precisar fazer mais nada, apenas incluir o arquivo CSS com os tais seletores. O ID e a classe precisam ser usados no HTML.

Podemos ter várias combinações entre estes:

```
h1, h2, p {  
    text-align: center;  
    color: red;  
}
```

```
h1, h2, .cls {  
    text-align: center;  
    color: red;  
}
```

18 - Exemplos para o background

```
body {  
  background-color: lightblue;  
}
```

```
body {  
  background-image: url("paper.gif");  
}
```

```
body {  
  background-image: url("gradient_bg.png");  
  background-repeat: repeat-x;  
}
```

```
body {  
  background-image: url("img_tree.png");  
  background-repeat: no-repeat;  
}
```

```
body {  
  background-image: url("img_tree.png");  
  background-repeat: no-repeat;  
  background-position: right top;  
}
```

```
body {  
  background-image: url("img_tree.png");  
  background-repeat: no-repeat;  
  background-position: right top;  
  background-attachment: fixed;  
}
```

```
body {  
  background: #ffffff url("img_tree.png") no-repeat right top;  
}
```

https://www.w3schools.com/css/css_background.asp

19 - Exemplo mostrando o boxmodel

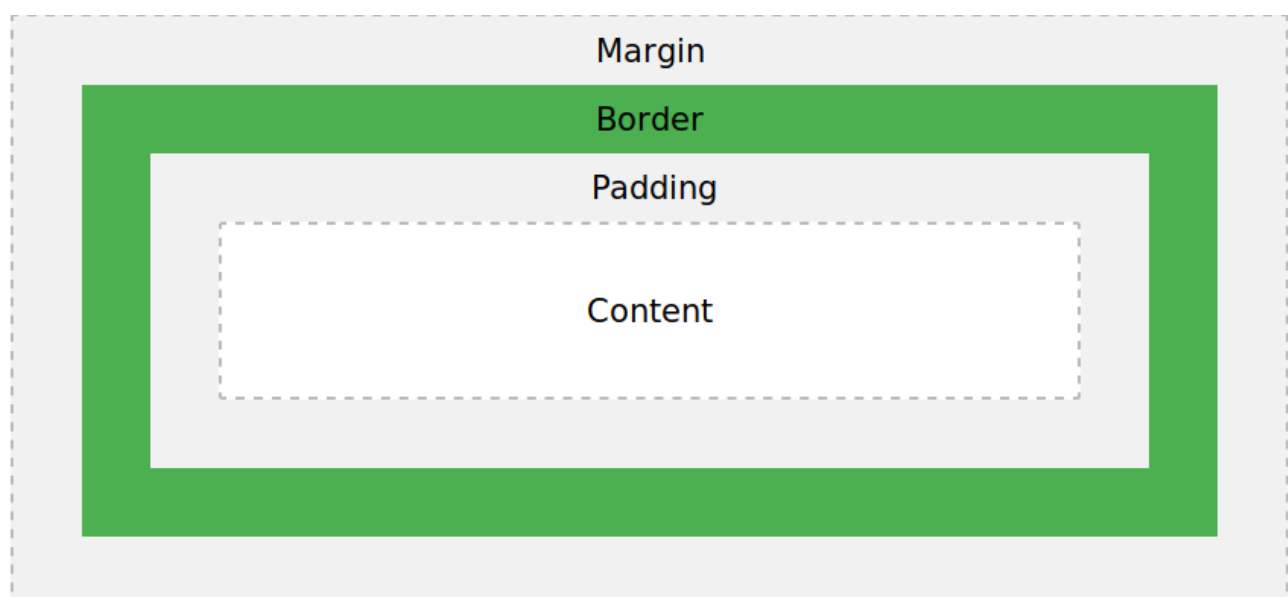
```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  background-color: lightgrey;
  width: 300px;
  border: 25px solid green;
  padding: 25px;
  margin: 25px;
}
</style>
</head>
<body>

<h2>Demonstrando o Box Model</h2>

<p>O CSS box model é essencialmente uma caixa que envolve todos os elementos HTML.
Consiste em: bordas, preenchimento, margens e o conteúdo real.</p>

<div>This text is the actual content of the box. We have added a 25px padding, 25px margin and a
25px green border. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut
aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse
cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa
qui officia deserunt mollit anim id est laborum.</div>

</body>
</html>
```

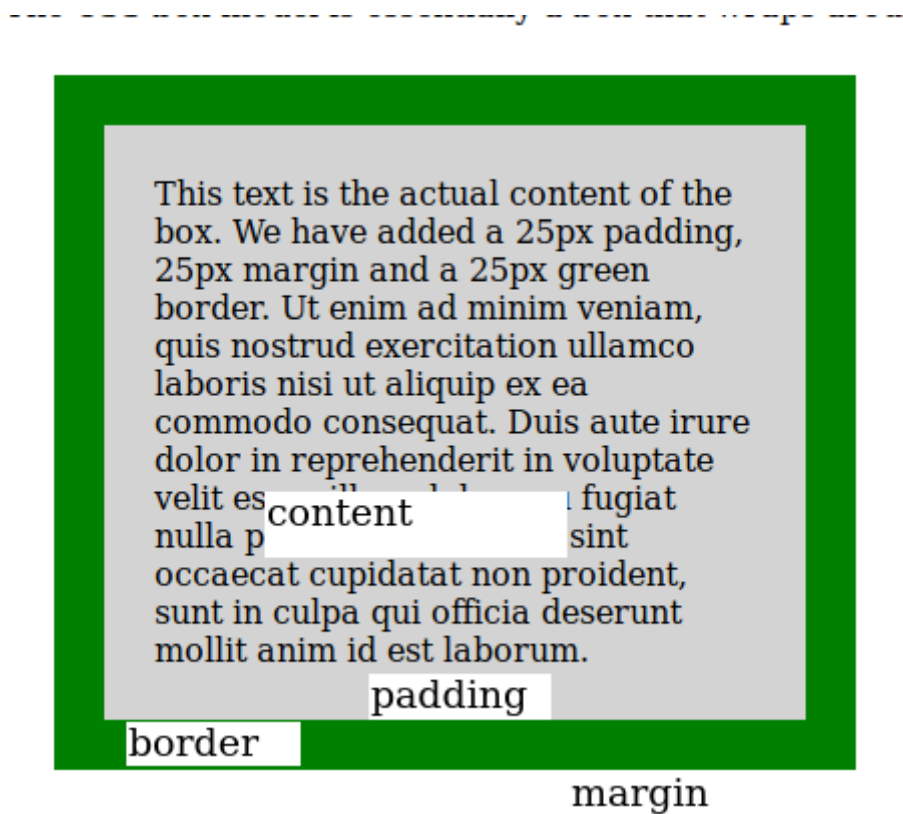


```
<style>
body{
    background-color: cyan;
}
div {
    background-color: lightgrey;
    width: 300px;
    border: 15px solid green;
    padding: 50px;
    margin: 20px;
    text-align: justify;
}
</style>
</head>
<body>
```

<div>Demonstrating the Box Model

This text is the content of the box. We have added a 50px padding, 20px margin and a 15px green border. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.


```
<pre>
background-color: cyan;
width: 300px;
border: 15px solid green;
padding: 50px;
margin: 20px;
</pre>
</div>
```



20 - Formulário de Contato com Estilo

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>--Formulário Contato--</title>
  <meta charset="utf-8" />
  <!-- <link href="ContatoEstilo.css" rel="stylesheet" media="all" />
  <script src="JavaScript1.js"></script>-->

<style>
body {
  padding: 0;
  margin: 0;
}

form {
  font-family: Verdana,Tahoma,sans-serif;
  font-size: 15px;
  background-color: #CC3300;
  max-width: 500px;
}

h1 {
  margin-bottom: 20px;
  margin-left: 20px;
  color: #66FFFF;
  text-shadow: 5px 5px 5px black;
}

input, textarea {
  width: 200px;
  padding: 10px;
  color: black;
  margin-left: 20px;
  -webkit-border-radius: 5px;
  -moz-border-radius: 5px;
  border: 1px solid;
}

input:hover, textarea:hover, input:focus, textarea:focus {
  border-color: #000066;
  box-shadow: 5px 0px 9px #FFFF00;
  background-color: #66FFFF;
  color: black;
  font-weight: bold;
}

.submit input {
```

```
background-color: #66FFFF;
height: 40px;
width: 100px;
font-weight: bold;
border-radius: 5px;
-webkit-border-radius: 5px;
-moz-border-radius: 5px;
border: 1px solid;
color: black;
}

textarea {
width: 400px;
height: 120px;
line-height: 20px;
max-width: 400px;
}

form label {
margin-left: 20px;
color: #FFFFFF;
font-weight: bold;
}
</style>
</head>
<body>
  <form name="meu_form">

    <h1>Entre em contato</h1>

    <p class="nome">
      <input type="text" id="nomeid" placeholder="Tiago Vale" required="required"
name="nome" />
      <label for="nome">Nome</label>
    </p>
    <p class="fone">
      <input type="text" id="foneid" placeholder="(xx)xx-xx-xx-xx" name="fone" />
      <label for="fone">Fone</label>

    </p>
    <p>
      <input type="email" id="emailid" placeholder="fulano@mail.com" name="email" />
      <label for="email">Email</label>
    </p>
    <p>
      <textarea placeholder="Deixe sua opinião"></textarea>
    </p>
    <p class="submit">
      <input type="submit" onclick="Enviar();" value="Enviar" />
    </p>
```

```
</form>

<script>
function Enviar() {

    var nome = document.getElementById("nomeid");

    if (nome.value != "") {
        alert('Obrigado sr(a) ' + nome.value + ' os seus dados foram encaminhados com sucesso');
    }

}
</script>
</body>
<!-- Créditos: https://www.devmedia.com.br/criando-form-de-contato-com-html5-css3-e-javascript/
29415#ixzz453cK6sZk -->
</html>
```

21 - height e width

Usados para setar a altura e a largura de elementos.

O default é auto.

Podemos setar com px, cm, %, etc

```
div {  
  height: 200px;  
  width: 50%;  
  background-color: powderblue;  
}
```

```
div {  
  height: 100px;  
  width: 500px;  
  background-color: powderblue;  
}
```

```
div {  
  max-width: 500px;  
  height: 100px;  
  background-color: powderblue;  
}
```

https://www.w3schools.com/css/css_dimension.asp

Exemplos de inclusão de CSS externo

```
<link rel="stylesheet" type="text/css" href="custom.css">  
<link href="//netdna.bootstrapcdn.com/bootstrap/3.0.0/css/bootstrap.min.css" rel="stylesheet">
```

22 - Alinhamento na horizontal e na vertical

Para alinhar texto o CSS traz o parâmetro text-align, mas para alinhar objetos como elementos do HTML, imagens e outros precisamos criar uma classe com alguns parâmetros, como os exemplos abaixo:

Para alinhar na horizontal use

```
margin auto;
```

Centralizar elemento

```
.center {  
  margin: auto;  
  width: 50%;  
  border: 3px solid green;  
  padding: 10px;  
}
```

Centralizar texto

```
.center {  
  text-align: center;  
  border: 3px solid green;  
}
```

Centralizar imagem

```
img {  
  display: block;  
  margin-left: auto;  
  margin-right: auto;  
  width: 40%;  
}
```

Alinhar pela esquerda e direita com position

```
.right {  
  position: absolute;  
  right: 0px;  
  width: 300px;  
  border: 3px solid #73AD21;  
  padding: 10px;  
}
```

Alinhar pela esquerda e direita usando float

```
.right {  
  float: right;  
  width: 300px;  
  border: 3px solid #73AD21;  
  padding: 10px;  
}
```

Centralizar na vertical usando padding

```
.center {  
  padding: 70px 0;  
  border: 3px solid green;  
}
```

```
.center {  
  padding: 70px 0;  
  border: 3px solid green;  
  text-align: center;  
}
```

Centralizar na vertical usando line-height

```
.center {  
  line-height: 200px;  
  height: 200px;  
  border: 3px solid green;  
  text-align: center;  
}
```

/* If the text has multiple lines, add the following: */

```
.center p {  
  line-height: 1.5;  
  display: inline-block;  
  vertical-align: middle;  
}
```

Centralizar na vertical usando position e transform

```
.center {  
  height: 200px;  
  position: relative;  
  border: 3px solid green;  
}
```

```
.center p {  
  margin: 0;  
  position: absolute;  
  top: 50%;  
  left: 50%;
```

```
transform: translate(-50%, -50%);  
}
```

https://www.w3schools.com/css/css_align.asp

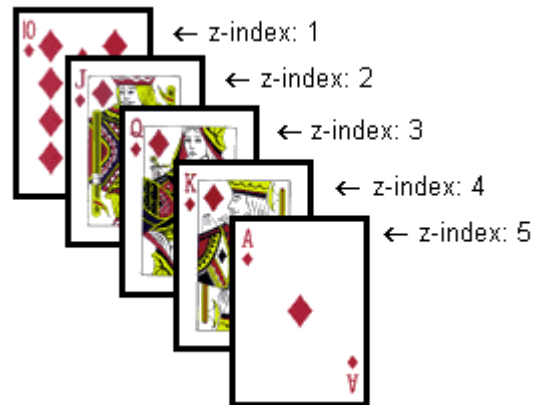
23 - Z-index (camadas)

Resumindo, camadas significam como os elementos se sobrepõem uns aos outros.

Para fazer isto definimos para cada elemento um número índice (z-index).

O comportamento é que elementos com número índice maior se

sobrepõem àqueles com menor número.



```
#ten_of_diamonds {  
    position: absolute;  
    left: 100px;  
    top: 100px;  
    z-index: 1;  
}  
  
#jack_of_diamonds {  
    position: absolute;  
    left: 115px;  
    top: 115px;  
    z-index: 2;  
}  
  
#queen_of_diamonds {  
    position: absolute;  
    left: 130px;  
    top: 130px;  
    z-index: 3;  
}  
  
#king_of_diamonds {  
    position: absolute;  
    left: 145px;  
    top: 145px;  
    z-index: 4;  
}  
  
#ace_of_diamonds {  
    position: absolute;  
    left: 160px;  
    top: 160px;  
    z-index: 5;  
}
```


24 - Design Responsivo

É muito importante atualmente criar sites responsivos, ou seja, que se ajustem automaticamente ao tamanho do dispositivo que acessa nosso site. Podemos ver isso redimensionando a janela do navegador. Sites responsivos se ajustam ao tamanho do dispositivo.

Como existem uma grande quantidade de usuários que atualmente acessa sites através de celulares, então se nosso site não for responsivo ele fica muito ruim de ser acessado através de um celular.

Existem muitas boas informações sobre como criar sites responsivos

<https://www.hostinger.com.br/tutoriais/criar-site-responsivo-css>

<http://www.carloshps.com.br/blog/criar-site-responsivo-com-html5-e-css3-parte-1-de-3/>

<https://marketingdeconteudo.com/como-criar-uma-pagina-web-responsiva/>

Uma dica interessante é sobre imagens: quando usar imagens em site responsivo use o tamanho com % e não especifique a altura, mas somente a largura com porcentagem. Assim o navegador calculará automaticamente a altura proporcionalmente a largura.

https://www.w3schools.com/css/css_rwd_intro.asp

Para um design responsivo adicionar entre as tags <head>:

```
<meta name="viewport" content="width=device-width,initial-scale=1">
```

Detalhes:

<http://blog.popupdesign.com.br/design-responsivo-i-o-que-e-e-por-que-usar/>

<http://blog.popupdesign.com.br/desenvolvimento-responsivo-e-viewport/>

<http://blog.popupdesign.com.br/palestra-design-e-desenvolvimento-responsivo/>

25 - Exemplo de Tabela Elegante com Estilo

```
<!DOCTYPE html>
<html lang="pt">
  <head>
    <meta charset="utf-8"/>
    <title>Tabela Elegante</title>
  </head>
</body>

  <h1 align="center">Tabela Elegante</h1>

<style>
#customers {
  font-family: "Trebuchet MS", Arial, Helvetica, sans-serif;
  border-collapse: collapse;
  width: 100%;
}

#customers td, #customers th {
  border: 1px solid #ddd;
  padding: 8px;
}

#customers tr:nth-child(even){background-color: #f2f2f2;}

#customers tr:hover {background-color: #ddd;}

#customers th {
  padding-top: 12px;
  padding-bottom: 12px;
  text-align: left;
  background-color: #4CAF50;
  color: white;
}
</style>
</head>
<body>

<table id="customers">
  <tr>
    <th>Company</th>
    <th>Contact</th>
    <th>Country</th>
  </tr>
  <tr>
    <td>Alfreds Futterkiste</td>
    <td>Maria Anders</td>
    <td>Germany</td>
  </tr>
  <tr>
    <td>Berglunds snabbköp</td>
    <td>Christina Berglund</td>
    <td>Sweden</td>
  </tr>
  <tr>
    <td>Centro comercial Moctezuma</td>
    <td>Francisco Chang</td>
    <td>Mexico</td>
  </tr>
</table>
</body>
</html>
```

```
</tr>
<tr>
  <td>Ernst Handel</td>
  <td>Roland Mendel</td>
  <td>Austria</td>
</tr>
<tr>
  <td>Island Trading</td>
  <td>Helen Bennett</td>
  <td>UK</td>
</tr>
<tr>
  <td>Königlich Essen</td>
  <td>Philip Cramer</td>
  <td>Germany</td>
</tr>
<tr>
  <td>Laughing Bacchus Winecellars</td>
  <td>Yoshi Tannamuri</td>
  <td>Canada</td>
</tr>
<tr>
  <td>Magazzini Alimentari Riuniti</td>
  <td>Giovanni Rovelli</td>
  <td>Italy</td>
</tr>
<tr>
  <td>North/South</td>
  <td>Simon Crowther</td>
  <td>UK</td>
</tr>
<tr>
  <td>Paris spécialités</td>
  <td>Marie Bertrand</td>
  <td>France</td>
</tr>
</table>

</body>
</html>
```

26 - Dicas de CSS

```
input:not([type])  
{  
  background:red;  
}
```

```
input[type="text"]  
{  
  background:red !important;  
}
```

Mudar a cor do texto do placeholder

```
::placeholder {  
  color: red;  
  opacity: 1; /* Firefox */  
}
```

Mudar a cor do texto da seleção

```
::selection {  
  color: red;  
  background: yellow;  
}
```

```
input[type="text"]  
input[type="button"]  
input[type="password"]  
input[type="number"]  
input[type="email"]
```

```
input[type=text]:invalid, input[type=tel]:invalid {  
  border: 2px dashed #ff0000;  
}
```

```
input[type=text]:focus {  
  width: 250px;  
}
```

```
a[target="_blank"] {  
  background-color: yellow;  
}
```

```
[class|="top"] {  
  background: yellow;  
}
```

```
[class^="top"] {  
  background: yellow;  
}
```

```
[class$="test"] {  
  background: yellow;  
}
```

Seleciona elemento cujo valor do atributo termina com um valor específico.

```
[class*="te"] {  
  background: yellow;  
}
```

Seleciona elemento cujo valor do atributo contém um valor específico.

Todos os que não são os da lista

`input:not([type=button]):not([type=password]):not([type=submit])...`

27 - Estilo para manter um rodapé fixo no fundo da página

```
<style>
footer{
  position: absolute;
  bottom: 0;
  left: 0;
  right: 0;
  height:30px;
  background-color:#E9ECEF;
  font-size:12px;
  text-align:center;
}
</style>
```

Transformation 2D e 3D no W3C

https://www.w3schools.com/css/css3_2dtransforms.asp

https://www.w3schools.com/css/css3_3dtransforms.asp

28 - Dicas Extras sobre Seletores

```
input:not([type])  
{  
  background:red;  
}
```

```
input[type="text"]  
{  
  background:red !important;  
}
```

```
input[type="text"]  
input[type="button"]  
input[type="password"]  
input[type="number"]  
input[type="email"]
```

```
input[type=text]:invalid, input[type=tel]:invalid {  
  border: 2px dashed #ff0000;  
}
```

```
input[type=text]:focus {  
  width: 250px;  
}
```

```
a[target="_blank"] {  
  background-color: yellow;  
}
```

```
[class|= "top"] {  
  background: yellow;  
}
```

```
[class^="top"] {  
  background: yellow;  
}
```

```
[class$="test"] {  
  background: yellow;  
}
```

Seleciona elemento cujo valor do atributo termina com um valor específico.

```
[class*="te"] {  
  background: yellow;  
}
```

Seleciona elemento cujo valor do atributo contém um valor específico.

Todos os que não estão na lista

```
input:not([type=button]):not([type=password]):not([type=submit])...
```

29 - Setas em CSS

```
<meta charset="UTF-8" >
<style>
#conteudo{
    padding: 10px;
    font-family: 'Arial', Helvetica, sans-serif;
    margin: 0 auto;
    width: 800px;
    padding-bottom: 50px;
    background-color: #fff;
    color: #000;
}
#rodape{
    margin-top: 400px;
}
</style>

<!-- Início de conteudo -->
<div id='conteudo'>
<h2 align="center" class="titulo">Dicas de CSS</h2>

<style>
i {
    border: solid black;
    border-width: 0 3px 3px 0;
    display: inline-block;
    padding: 3px;
}

.right {
    transform: rotate(-45deg);
    -webkit-transform: rotate(-45deg);
}

.left {
    transform: rotate(135deg);
    -webkit-transform: rotate(135deg);
}

.up {
    transform: rotate(-135deg);
    -webkit-transform: rotate(-135deg);
}

.down {
    transform: rotate(45deg);
    -webkit-transform: rotate(45deg);
}
```



```
</style>
</head>
<body>

<h2>Setas em CSS</h2>
<table>
<tr><td>Para a direita</td><td><i class="right"></i></td></tr>
<tr><td>Para a esquerda &nbsp;&nbsp;&nbsp;&nbsp;</td><td><i class="left"></i></td></tr>
<tr><td>Para cima</td><td><i class="up"></i></td></tr>
<tr><td>Para baixo</td><td><i class="down"></i></td></tr>
</table>

</body>
</html>
```

30 - Exemplos com Seletores

```
<!DOCTYPE html>
<html lang="pt">
  <head>
    <meta charset="utf-8"/>
    <title>Seletores CSS</title>
  </head>
<body>

  <h1 align="center">Seletores2 CSS</h1>

  <ul>
    <li><a href="#" rel="external">External Link 1</a></li>
    <li><a href="#" rel="internal">Internal Link 1</a></li>
    <li><a href="#" rel="internal">Internal Link 2</a></li>
    <li><a href="#" rel="external">External Link 2</a></li>
    <li><a href="#" rel="external">External Link 3</a></li>
  </ul>

  <style>
a[rel="external"]
{
  color: red;
}
a[rel="internal"]
{
  color: green;
}

/* Mudar a cor do link para verde e tirar o sublinhado
a
{
  text-decoration: none;
  color: green;
}
*/
</style>
```

```
<!DOCTYPE html>
<html lang="pt">
  <head>
    <meta charset="utf-8"/>
    <title>Seletores CSS</title>
  </head>
<body>

  <h1 align="center">Seletores CSS</h1>
```

```
<p>List 1</p>
<ul>
<li><a href="#">Link 1a</a></li>
<li><a href="#">Link 2a</a></li>
<li><a href="#">Link 3a</a></li>
<li><a href="#">Link 4a</a></li>
</ul>
<ul>
<li><a href="#">Link 1a</a></li>
<li><a href="#">Link 2a</a></li>
<li><a href="#">Link 3a</a></li>
<li><a href="#">Link 4a</a></li>
</ul>
```

```
<p>List 2</p>
<ul>
<li><a href="#">Link 1a</a></li>
<li><a href="#">Link 2a</a></li>
<li><a href="#">Link 3a</a></li>
<li><a href="#">Link 4a</a></li>
</ul>
<ul>
<li><a href="#">Link 1a</a></li>
<li><a href="#">Link 2a</a></li>
<li><a href="#">Link 3a</a></li>
<li><a href="#">Link 4a</a></li>
</ul>
```

```
<style>
/* Será aplicado o estilo somente aos primeiros ul após o p */
p + ul a
{
color: red;
}
</style>
```

Estes dois exemplos estão no livro <https://www.syncfusion.com/ebooks/css3>

Árvore de natal em CSS

<https://ihatetomatoes.net/svg-christmas/>

31 - Material para Download

Exemplos

form_contato.html
images.html
tabela_elegante.html

Exemplos avançados

flex_com.html
flex_sem.html
natal.html
required.txt
Rounded corners.html
setas_css.html

Outros bons tutoriais:

- HTML, CSS e JavaScript - <https://www.caelum.com.br/download/caelum-html-css-javascript.pdf>
- CSS3 - <https://www.syncfusion.com/ebooks/css3>

32 - Referências

<https://www.chiefdesign.com.br/o-que-e-css/>
<https://www.w3schools.com/css/default.asp>
<https://www.syncfusion.com/ebooks/css3>
<https://www.caelum.com.br/download/caelum-html-css-javascript.pdf>
https://www.w3schools.com/cssref/css_selectors.asp
https://www.w3schools.com/css/css3_2dtransforms.asp
https://www.w3schools.com/css/css3_3dtransforms.asp
https://www.w3schools.com/css/css_align.asp
https://www.w3schools.com/Css/css_boxmodel.asp