



FUNDAMENTOS DA PROGRAMAÇÃO WEB

AULA 3



Prof.^a Margarete Klamas



CONVERSA INICIAL

Vamos conhecer outras possibilidades de utilizar CSS.

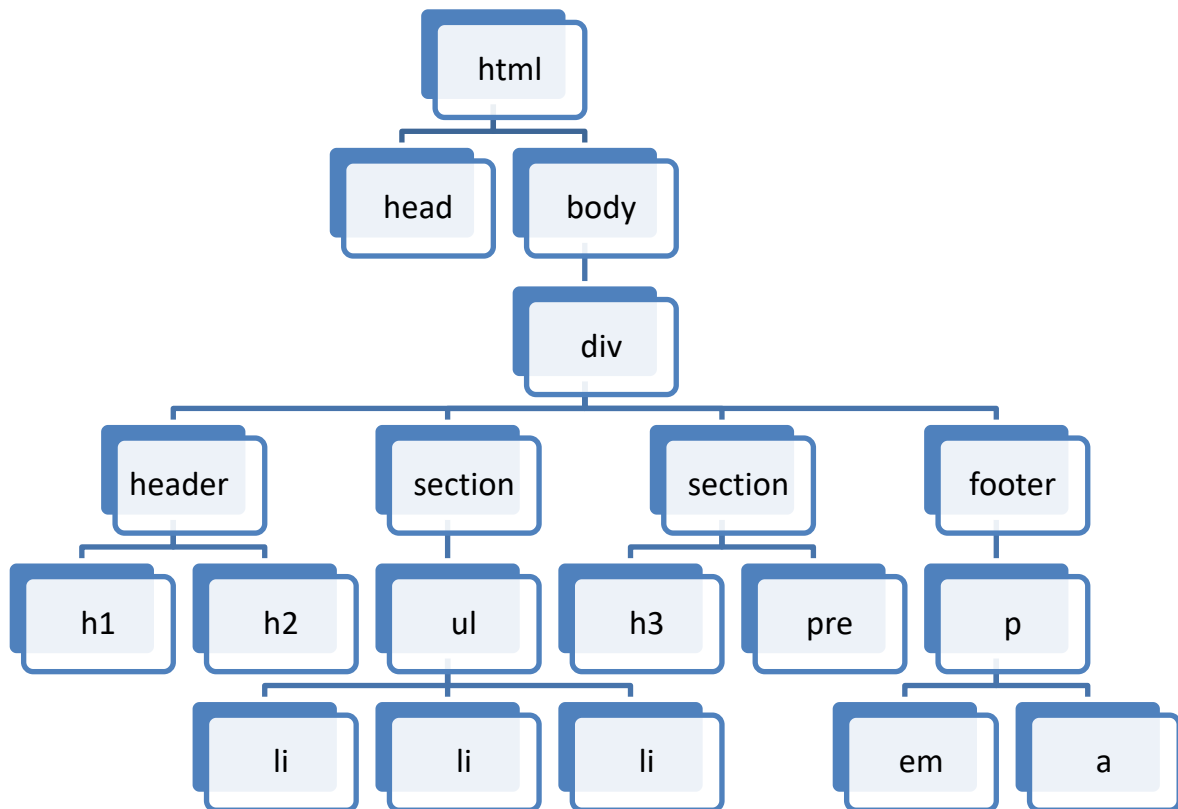
O objetivo desta abordagem é apresentar um aprofundamento em CSS, que serve de base para o desenvolvimento de aplicações web para os mais variados fins. Ao concluirmos os cinco tópicos, você será capaz de aplicar estilos (formatações) em páginas HTML, bem como poderá fazer alterações em páginas criadas por outros desenvolvedores.

No tópico 1, abordaremos a árvore do documento (DOM). No tópico 2, pseudoclasses e pseudoelementos. No tópico 3, veremos como criar menus (barras de navegação) e botões. No tópico 4 dimensionamento, variáveis e formatações para imagens. No tópico 5, abordaremos transformações 2D e transições.

TEMA 1 – ÁRVORE DO DOCUMENTO (DOM)

Ao manipular CSS, é necessário compreender o conceito de herança, porque algumas propriedades CSS por padrão herdam valores que foram definidos no elemento pai, outros não. Por exemplo: se declararmos uma cor da fonte para um elemento, todos os elementos dentro dele receberão essa cor por herança, salvo se for declarada uma cor diferente diretamente a eles. A propriedade width (largura) não faz parte da herança. Para compreendermos melhor a hierarquia (herança) dos elementos dentro de um documento HTML, usamos um modelo que representa os elementos de uma página HTML, o DOM. O diagrama representativo do DOM é semelhante a uma árvore. É uma analogia a árvore genealógica de uma família, na qual são representados os graus de parentesco, os ascendentes e os descendentes entre seus membros. Árvore do documento é uma estrutura que representa os nós do documento e a maneira como eles se relacionam.

Pode ser assim representada:



Fonte: elaborado com base em Silva, 20?, p. 105.

1.1 Elemento pai –filho

Um elemento é filho quando está contido em um elemento pai. Considerando a árvore do documento apresentada anteriormente, temos os seguintes elementos pai –filho(s):

ELEMENTO PAI	ELEMENTO FILHO
html	head, body
body	div
div	header, section, section, footer
header	h1, h2
ul	li,li,li
h3	Não tem elemento filho

Podemos identificar também os elementos ancestrais e descendentes.

ELEMENTO ANCESTRAL	ELEMENTO DESCENDENTE
div	header, section, section, footer
html	Todos os elementos dentro do html
body	Todos os elementos dentro do body



div	header, section, section, footer
header	h1, h2
footer	p, em, a
pre	-

Podemos identificar o elemento-irmão:

ELEMENTO PAI	ELEMENTO-IRMÃO
div	header, section, section, footer
section	h3 e pre
ul	li, li, li
p	em, a

Podemos estilizar em CSS utilizando nomes das tags, como já vimos, e até descrever o grau de parentesco, conforme a árvore do documento. A seguir, na tabela, a letra **E** é abreviatura de **elemento** e **F** é abreviatura de **filho**. Por exemplo:

SELETOR	SELECIONA	EXEMPLO
E F	Elemento F descendente do elemento E	
E>F	Elemento F filho do elemento E	nav > ul { margin-left: 30px; } Vai estilizar o ul filho do nav.
E + F	Elemento F imediatamente após E (irmãos)	
E ~F	Elemento F após elemento E (irmãos)	h1 ~ h2 { font-size: 27px; } Vai estilizar o h2, elemento-irmão do h1.
E:first-child E:last-child	Elemento E primeiro-filho do seu elemento-pai	nav > ul > li:first-child { font-size: 26px; } Seletor composto. Essa regra CSS estiliza o elemento li, último filho do elemento ul, que, por sua vez, é filho de nav.



Na tabela anterior, estão alguns exemplos de declarações que você pode vir a observar em CSS. Pode-se conhecer a tabela completa em: https://www.w3schools.com/cssref/css_selectors.asp.

TEMA 2 – PSEUDOCASSES E PSEUDOELEMENTOS

Conforme estudamos anteriormente, podemos definir estilos utilizando class, que se iniciam com um ponto (.). Outra maneira de aplicarmos estilos é a partir das **pseudoclasses**, que nos **permitem selecionar um elemento que não consta na árvore do documento**. A sintaxe utiliza um sinal de dois-pontos (:) seguido do nome da pseudoclasse.

Exemplo:

```
selector:pseudo-class{  
  property:value;  
}
```

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

2.1 Pseudoclasses

PSEUDOCASSE	APLICAÇÃO
:link	Link não visitado. O padrão de link não visitado é a cor azul. Podemos alterar essa cor, com essa pseudoclasse. CSS a:link {color: green; text-decoration:none;} HTML Site do Google Nesse exemplo, o texto do link, quando ainda não visitado, será na cor verde e sem o sublinhado padrão.
:visited	Link visitado. O padrão do link visitado é a cor roxa. Podemos alterar essa cor, com essa pseudoclasse. CSS a:visited {color: MistyRose;} HTML Site do Google Nesse exemplo, o texto do link depois que for visitado será na cor rosa (MistyRose).



:hover	<p>Sobre, quando o (apontador), por exemplo o mouse está sobre. Pode ser utilizado para qualquer elemento, não somente link.</p> <p>CSS</p> <pre>a:hover {color: grey;}</pre> <p>HTML</p> <pre>Site do Google</pre> <p>Com essa marcação, ao passar o mouse sobre, ficará na cor cinza.</p>
:active	<p>Elemento ativado, quando clicamos. Pode ser utilizado para qualquer elemento, não somente link.</p> <p>CSS</p> <pre>a:active {color: red;}</pre> <p>HTML</p> <pre>Site do Google</pre> <p>Com essa marcação, ao manter o clique, ficará na cor vermelha.</p>
:focus	<p>Pode-se estilizar um elemento, quando ele está em foco. Pode ser usado tanto em elementos a, para links, quanto em qualquer elemento da marcação.</p> <p>CSS</p> <pre>a:focus {color: black;}</pre> <p>HTML</p> <pre>Site do Google</pre> <p>Nesse exemplo, o texto do link receberá uma borda sólida de 2px na cor preta (black) quando o usuário o colocar em foco.</p>
:checked	<p>Quando utilizamos o radio-button ou checkbox, podemos estilizar o elemento “selecionado”.</p> <p>CSS</p> <pre>input:checked { ... }</pre> <p>HTML</p> <pre>< form > < input type = " radio" name = "" value = "" /> < input type = " checkbox" name = "" value = "" /> </ form ></pre>



	Nesse exemplo, os dois controles input serão estilizados, conforme a regra CSS.
:enabled :disabled	<p>Pode ser aplicado em um elemento habilitado (:enabled) ou desabilitado (:disabled).</p> <p>CSS</p> <pre>input:enabled { border: 1px solid gray; } input:disabled { border: 1px solid blue; }</pre> <p>HTML</p> <pre>< form > < input type =" text" /> < input type =" text" disabled =" disabled" /> </ form ></pre> <p>Nesse exemplo, o primeiro controle input terá uma borda na cor cinza (gray), e o segundo, na cor azul (blue).</p>

A ordem na estilização deve ser:

```
a:link { ... }  
a:visited { ... }  
a:hover { ... }  
a:active { ... }
```

Caso inverta a ordem, o efeito não será aplicado conforme o planejado.

2.1 Pseudoelementos

Quando desejar **estilizar parte de um elemento**, como, por exemplo, a primeira letra de uma palavra, temos o recurso chamado de **pseudoelemento**. É possível formatar antes ou depois do conteúdo de um elemento. A sintaxe dos pseudoelementos é dois-pontos duplos (::).

Sintaxe:

```
selector::pseudo-element{  
  property: value;  
}
```

PSEUDOELEMENTO	APLICAÇÃO
::after	<p>Pode-se formatar elemento depois.</p> <p>CSS</p> <pre><style> .teste::after {</pre>



	<pre>content: url('https://fvsch.com/articles/css- content-image/star.png'); }</pre> <p>HTML</p> <pre><p class="teste">Aqui um parágrafo</p></pre> <p>Aqui um parágrafo</p></p> <p>Irá aparecer um ícone após o parágrafo.</p>
::before	<p>Permite formatar elemento antes.</p> <p>CSS</p> <pre><style> .name::before { content: "Name: "; }</pre> <p>HTML</p> <pre><div class="name">Anna</div></pre> <p>No exemplo anterior, será renderizado no browser:</p> <p>Nome: Anna</p>
::first-letter	<p>Permite aplicar a formatação na primeira letra.</p> <p>Podemos formatar para ::first-letter:</p> <ul style="list-style-type: none">• propriedades da fonte• propriedades de cor• propriedades de fundo• propriedades de margem• propriedades de preenchimento• propriedades da borda• decoração de texto• vertical-align (somente se "float" for "none")• transformação de texto• altura da linha• float• clear <p>CSS</p> <pre><style> p::first-letter { color: #ff00ff; font-size: xx-large; }</pre> <p>HTML</p> <pre><p> Aqui o texto do parágrafo</p></pre> <p>No exemplo anterior, a primeira letra do parágrafo ficará na cor lilás.</p>
::first-line	<p>Permite aplicar formatação na primeira linha.</p> <p>CSS</p>

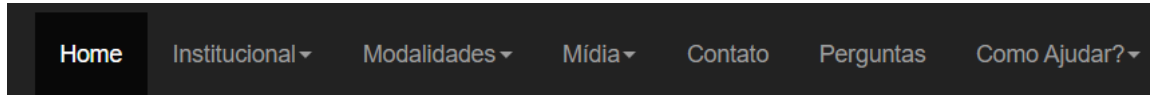


<p>Podemos aplicar ao <code>::first-line</code> :</p> <ul style="list-style-type: none">• propriedades da fonte• propriedades de cor• propriedades de fundo• espaçamento entre palavras• espaçamento entre letras• decoração de texto• alinhamento vertical• transformação de texto• altura da linha• clear	<pre><style> p::first-line { color: #ffff00; font-variant: small-caps; } </style></pre> <p>HTML</p> <pre><p> Aqui você digita o texto do parágrafo. </p></pre> <p>A primeira linha do parágrafo ficará na cor amarelo e em letra maiúscula pequena.</p>
<p><code>::marker</code></p>	<p>Seleciona os elementos de uma lista.</p> <p>CSS</p> <pre><style> ::marker { color: orange; font-size: 28px; } </style></pre> <p>HTML</p> <pre> HTML CSS JavaScript </pre> <p>O marcador da lista ficará com a cor laranja e tamanho de 28px.</p>
<p><code>::selection</code></p> <p>Pode-se alterar a cor de fonte, cor de fundo, cursor e outline.</p>	<p>Permite formatar a seleção.</p> <p>CSS</p> <pre><style> ::selection { color: blue; background: gray; } </style></pre> <p>HTML</p> <pre><h1> Titulo </h1> <p> Aqui um texto. </p> <div> Aqui um texto. </div></pre> <p>Ao selecionar o texto, este ficará com cor azul e cor de fundo cinza.</p>



TEMA 3 – BARRA DE NAVEGAÇÃO

A barra de navegação tem o objetivo de permitir que o usuário encontre os assuntos disponíveis no site ou aplicação.



Para criarmos uma barra de navegação (menus), precisamos de HTML para criar inicialmente uma lista.

3.1 Barra de navegação vertical

Exemplo:

<pre> Home Notícias Contato </pre>	<ul style="list-style-type: none">• Home• Notícias• Contato
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------

Descrição da imagem:

Na imagem ao lado, visualizamos a lista criada. Como utilização HTML, temos na lista o marcador padrão (bolinha) e os links: Home, Notícias e Contato estão sublinhados e na cor azul. Estes são os padrões de exibição do HTML

No quadro anterior, criamos uma lista com hiperlinks. Para estilizarmos o menu, iremos executar os passos a seguir:

	CSS	O QUE FAZ
1	<pre><style> ul { list-style-type: none; margin: 0; padding: 0; width: 200px; background- color: #f1f1f1; }</pre>	<p>remove o marcador: list-style-type: none;</p> <p>zeram as margens e os enchimento do browser: margin: 0; padding: 0;</p> <p>Largura 200px: width: 200px;</p>



		Cor de fundo: background-color: #1f1f1f;
2	<pre>li a { display: block; color: #000; padding: 8px 16px; text-decoration: none; }</pre>	Exibe os links como elementos de bloco e torna toda a área do link clicável: display: block; Colocar cor para o texto: color: #000; Especificar um padding: padding: 8px 16px; Retirar o sublinhado: text-decoration: none;
3	<pre>li a:hover { background-color: #555; color: white; } </style></pre>	Alterar a cor do :hover
4	<p>O código do exemplo anterior está disponível em:</p> <p>https://github.com/N-CPUninter/FUNDAMENTOS-DE-DESENVOLVIMENTO-WEB/blob/main/html/barra_navega%C3%A7%C3%A3o_vertical.html</p>	
	<p>Observação: caso queira modificar este exemplo para menu horizontal, altere o 2. Para:</p>	<pre>li{display:inline;} a { color: #000; padding: 8px 16px; text-decoration: none; }</pre>

3.2 Barra de navegação horizontal

Para a barra de navegação horizontal, podemos utilizar a propriedade `display` ou `float`.

Vamos utilizar a mesma lista para o exemplo:



<pre> Home Notícias Contato </pre>	<ul style="list-style-type: none">• Home• Notícias• Contato
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------

Descrição da imagem:

Na imagem ao lado, visualizamos a lista criada. Como utilização HTML, temos na lista o marcador padrão (bolinha) e os links: Home, Notícias e Contato estão sublinhados e na cor azul. Estes são os padrões de exibição do HTML

Para estilizarmos o **menu horizontal**, iremos executar os passos a seguir:

	CSS	O QUE FAZ
1.	<pre><style> ul { list-style-type: none; margin: 0; padding: 0; width: 200px; background-color: #f1f1f1; }</pre>	<p>Remove o marcador: list-style-type: none;</p> <p>zera as margens e os enchimento do browser: margin: 0; padding: 0;</p> <p>Largura 200px: width: 200px;</p> <p>Cor de fundo: background-color: #f1f1f1;</p>
2.	<pre>li { float: left; }</pre>	<p>Faz com que os elementos do bloco (li) flutuem à esquerda: float: left;</p>
3.	<pre>li a { color: black; display: block; text-align: center; padding: 14px 16px; text-decoration: none; }</pre>	<p>Para aplicar formatação como bloco: display: block; entralizar o texto: text-align: center;</p>
4.	<pre>li a: hover { background-color: #555; color: white; }</pre>	<p>Altera a cor da fonte e cor de fundo quando o mouse estiver sobre (hover).</p>



O exemplo anterior está disponível em:

<https://github.com/N-CPUninter/FUNDAMENTOS-DE-DESENVOLVIMENTO-WEB/blob/main/html/barra_navega%C3%A7%C3%A3o_horizontal.html>

3.3 Menu vertical com menu suspenso

O menu suspenso permite colocar uma lista suspensa como link. Vamos aqui também utilizar uma lista HTML.

CÓDIGO	EXPLICAÇÃO
<pre> Home Sobre nós Nossas Lojas Notícias Contato </pre>	<p>Aqui estamos inserindo uma lista dentro da outra para gerar o submenu.</p> <pre>Home</pre> <p>aqui não fechamos a </p> <pre></pre> <p>Inserimos a lista para o submenu</p> <pre>Sobre nós Nossas Lojas </pre> <p>Fechamos a li da 1ª linha</p> <pre></pre> <p>Fechamos a sublista</p>

Vamos ao CSS com a estilização do submenu:

CSS	O QUE FAZ
<pre>ul { margin: 0; padding: 0; list-style-type: none; width: 150px; }</pre>	Remove bullets e identificação das listas



<pre>ul li { position: relative; }</pre>	Agora vamos posicionar os itens da lista, definir o posicionamento com o valor relativo, porque depois iremos posicionar os submenus na posição absoluta em relação a eles.						
<pre>li ul { position: absolute; left: 149px; top: 0; display: none; }</pre>	Agora vamos estilizar os submenus. Queremos cada submenu ao lado direito do seu elemento-pai assim que passarmos o mouse sobre ele. Com as propriedades CSS Left e Top, podemos posicionar cada item do submenu ao lado de seu item-pai. Definimos o display: none, para que o submenu não fique visível por padrão.						
<pre>ul li a { display: block; text-decoration: none; color: #E2144A; background: #fff; padding: 5px; border: 1px solid #ccc; }</pre>	Agora vamos dar uma aparência melhor aos nossos links. É importante definir display:block, para que cada link ocupe toda a largura definida para ele na lista.						
<pre>ul li a:hover { text-decoration: underline; font-weight: bold; background: #ccc; }</pre>	Agora vamos dar uma aparência melhor ao nosso menu quando passarmos o mouse em cima dele.						
<pre>li:hover ul { display: block; }</pre>	Exibindo o submenu com: display: block;						
Na imagem ao lado, temos a renderização do menu. Ao colocar o cursor sobre Home, o submenu passa a ser exibido. Ao retirar o cursor, ele não será exibido.	<table><tr><td><u>Home</u></td><td>Sobre nós</td></tr><tr><td>Notícias</td><td>Nossas Lojas</td></tr><tr><td>Contato</td><td></td></tr></table>	<u>Home</u>	Sobre nós	Notícias	Nossas Lojas	Contato	
<u>Home</u>	Sobre nós						
Notícias	Nossas Lojas						
Contato							
O código do exemplo anterior está disponível em: < https://github.com/N-CPUninter/FUNDAMENTOS-DE-DESENVOLVIMENTO-WEB/blob/main/html/menu%20_lista.html >							



3.4 Botões CSS

Podemos inserir botões de três maneiras: utilizando o `<button>`, link e `<input type="button">`. A seguir, no HTML, temos aplicação com três exemplos. No CSS, a estilização para os botões.

HTML	CSS
<pre><h2>Botões</h2> <button>Botão Padrão</button> Link como botão <button class="button">Botão</button> <input type="button" class="button" value="Enviar"></pre>	<pre><style> .button { background-color: gray; border: none; color: white; padding: 15px 32px; text-align: center; text-decoration: none; display: inline- block; font-size: 16px; margin: 4px 2px; } </style></pre>
<p>O código do exemplo anterior está disponível em:</p> <pre><https://github.com/N-CPUninter/FUNDAMENTOS-DE- DESENVOLVIMENTO-WEB/blob/main/html/botoes.html></pre>	

TEMA 4 – OUTRAS FORMATAÇÕES

Neste tópico, vamos abordar algumas formatações interessantes, como o dimensionamento das caixas, variáveis e múltiplas colunas.

4.1 Dimensionamento do box model

Sabemos que a largura e a altura de um elemento são calculadas assim:

largura + preenchimento + borda = largura real de um elemento

altura + preenchimento + borda = altura real de um elemento

Quando você define a largura/altura de um elemento, o elemento geralmente aparece maior do que você definiu (porque a borda e o preenchimento do elemento são adicionados à largura/altura especificada do elemento).



Exemplo:

```
.div1 {  
  width: 400px;  
  height: 200px;  
  border: 1px solid red;  
}
```

```
.div2 {  
  width: 400px;  
  height: 200px;  
  padding: 30px;  
  border: 1px solid blue;  
}
```

No exemplo anterior, a largura da div1 é igual a div2, o mesmo ocorre com a altura. Porém, a div2 ocupará maior espaço, pois serão acrescentados os valores da borda e do *padding*.

A propriedade **box-sizing** permite incluir os valores da borda e preenchimento no valor total da div. Para que isso ocorra, devemos definir **box-sizing: border-box**:

```
.div1 {  
  width: 400px;  
  height: 200px;  
  border: 1px solid red;  
  box-sizing: border-box;  
}
```

```
.div2 {  
  width: 400px;  
  height: 200px;  
  padding: 30px;  
  border: 1px solid blue;  
  box-sizing: border-box;  
}
```

É aconselhável utilizar a propriedade em todos os elementos da página.

```
* {  
  box-sizing: border-box;  
}
```

4.2 Variáveis

Podemos utilizar variáveis com CSS, com a função **var()**. É possível utilizar variáveis para definir as cores, evitando a repetição de códigos.

Podemos declarar variáveis com escopo global ou local. As variáveis globais podem ser acessadas em todo o documento, enquanto as variáveis locais podem ser utilizadas apenas dentro do seletor onde está declarada.

Para declarar uma variável global, utiliza-se o seletor **:root**. Para declarar uma variável com escopo local, declare-a dentro do seletor que irá utilizá-la.

Exemplo de sintaxe:

```
:root {  
  --dourado: #DEB44B;  
  --branco: #ffffff;  
}  
body {  
  background-color: var(--
```

```
:root {  
  --top-color: #F5D95F;  
  --bottom-color: #ffffff;  
  --my-gradient: linear-  
gradient(var(--top-color), var(--  
bottom-color));
```




```
dourado);  
}  
h2 {  
  border-bottom: 2px solid  
var(--dourado);  
}
```

```
}  
body {  
  background-image: var(--my-  
gradient);  
}
```

4.3 Formatações para imagens

É possível aplicar estilos em imagens, como arredondamento de cantos e opacidade. Por exemplo:

Imagem com cantos arredondados:

```
img {  
  border-  
radius: 10px;  
}
```

Imagem circular se as medidas formarem um quadrado, ou oval, se as medidas formarem um retângulo.

```
img {  
  border-  
radius: 50%;  
}
```

Imagem com opacidade:

```
img {  
  opacity: 0.5;  
}
```

A opacidade padrão é 1. Teste números menores que 1.

Para centralizar imagens, utilize:

```
img {  
  display: block;  
  margin: auto;  
}
```

A respeito do redimensionamento de imagem, não é recomendável aumentar uma imagem além do seu tamanho normal. Diminuir é possível:

CSS

```
<style>  
img {  
  max-width:  
100%;  
  height: auto;  
}  
</style>
```

HTML

```

```

Obs.: o tamanho normal da imagem é 1000.



TEMA 5 – MOVIMENTOS EM CSS


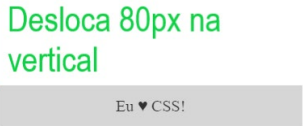
O sistema de coordenadas em CSS tem como origem o canto superior esquerdo do box. A coordenada x (horizontal) tem valores positivos (à direita) e valores negativos (à esquerda). A coordenada y (vertical) aumenta para baixo e diminui para cima.

5.1 Transformações CSS 2D

É possível mover, girar, dimensionar e inclinar elementos.

Métodos:

- translate()
- scale()
- rotate()
- skewX()
- skewY()
- matrix()

MÉTODO	DESCRIÇÃO	EXEMPLO
translateX(): Sintaxe: <code>seletor { transform: translateX(x); }</code>	Permite deslocar um box na horizontal.	No exemplo a seguir, o elemento div terá um deslocamento horizontal de 80px: <pre>div{ transform: translateX(80px); }</pre> 
translateY() <code>seletor { transform: translateY(y); }</code>	Permite deslocar um box na vertical.	No exemplo a seguir, o elemento div terá um deslocamento vertical de 80px: <pre>div{ transform: translateY(80px); }</pre> 



translate() seletor { transform: translate(x, y); }	Permite deslocar um box na horizontal e na vertical ao mesmo tempo.	No exemplo a seguir, o elemento div terá um deslocamento horizontal de 80px e vertical de 100px: <code>div {transform: translate(80px, 100px);}</code>
scaleX() seletor { transform: scaleX(n); }	Permite expandir ou contrair a largura de um box.	No exemplo a seguir, o elemento div aumentará três vezes o seu tamanho, na largura. <code>div {transform: scaleX(3);}</code> No exemplo a seguir, o elemento div diminuirá a metade de seu tamanho original, na largura. <code>div {transform: scaleX(.5);}</code>
scaleY() seletor { transform: scaleY(n); }	Permite expandir ou contrair a altura de um box.	No exemplo a seguir, o elemento div aumentará três vezes a sua altura. <code>div {transform: scaleY(3);}</code> No exemplo a seguir, o elemento div diminuirá a metade de sua altura. <code>div {transform: scaleY(.5);}</code>
scale() seletor { transform: scale(n,m); }	Permite expandir ou contrair a largura e altura de um box.	No exemplo a seguir, o elemento div aumentará três vezes a sua largura e duas vezes a sua altura. <code>div {transform: scale(3,2);}</code>
rotate() seletor { transform: rotate(a); }	Permite girar um box no sentido horário (valor positivo) e anti-horário (valor negativo).	No exemplo a seguir, o elemento div irá girar 20 graus no sentido horário. <code>div {transform: rotate(20deg);}</code> Medidas CSS para ângulos são: deg, grad, rad e turn.



skewX() seletor { transform: skewX(a); }	Permite inclinar um box, utilizando o eixo X.	No exemplo a seguir, a div será inclinada 20 graus ao longo do eixo X: <pre>div { transform: skewX(20deg); }</pre>
skewY() seletor { transform: skewY(a); }	Permite inclinar um box, utilizando o eixo Y.	No exemplo a seguir, a div será inclinada 10 graus ao longo do eixo Y: <pre>div { transform: skewY(10deg); }</pre>
skew() seletor { transform: skew(a,b); }	Permite inclinar um box, utilizando os eixos X e Y.	No exemplo a seguir, a div será inclinada 20 graus ao longo do eixo X e 10 graus ao longo do eixo Y: <pre>div { transform: skewY(20deg,10deg); }</pre>
matrix()	Combina todos os métodos de transformação 2D.	<u>Sintaxe:</u> matrix (scaleX(), skewY(), skewX(), scaleY(), translateX(), translateY())

5.2 Transições CSS

Permite criar transições ao definir uma mudança de valor de uma propriedade CSS, de modo que ela (a mudança de valor) ocorra de forma suave, em um espaço de tempo especificado (Silva, 2013).

Propriedades de transição:

- transition;
- transition-delay;
- transition-duration;
- transition-property;
- transition-timing-function.

Para aplicar a transição, é necessário especificar qual a propriedade CSS e qual o tempo da transição. É necessário especificar dois valores para a propriedade.



PROPRIEDADES	DESCRIÇÃO
<ul style="list-style-type: none">transition <pre>div { width: 100px; height: 100px; background: blue; transition: width 2s; }</pre>	<p>No exemplo ao lado, a transição será aplicada na largura (width) e levará 2 segundos.</p> <p>É necessário especificar o segundo valor:</p> <pre>div:hover { width: 300px; }</pre>
<p>Exemplo de transição:</p> <p><https://github.com/N-CPUninter/FUNDAMENTOS-DE-DESENVOLVIMENTO-WEB/blob/main/html/transi%C3%A7%C3%B5es.html></p>	
<ul style="list-style-type: none">transition-delay <pre>div { transition-delay: 1s; }</pre>	<p>Permite especificar um atraso (em segundos) para o efeito de transição.</p>
<ul style="list-style-type: none">transition-timing-function	<p>A propriedade de função de tempo de transição pode ter os seguintes valores:</p> <ul style="list-style-type: none">• ease- início lento, depois rápido, depois termina lentamente (isso é padrão);• linear- mesma velocidade do início ao fim;• ease-in- início lento;• ease-out- final lento;• ease-in-out- lento do início ao fim;• cubic-bezier(n,n,n,n)- permite definir seus próprios valores em uma função de bezier cúbico.
<p>Exemplo de transição:</p> <p><https://github.com/N-CPUninter/FUNDAMENTOS-DE-DESENVOLVIMENTO-WEB/blob/main/html/transi%C3%A7%C3%B5es1.html></p>	
<p>É possível associar transição com transformação</p> <pre>div { width: 100px;</pre>	



```
height: 100px;
background: rgb(8, 0, 255);
transition: width 2s, height 2s, transform 2s;
}

div:hover {
width: 300px;
height: 300px;
transform: rotate(360deg);
}
```

Observe que a última propriedade é o tempo para *transform*:

Exemplo de combinação das propriedades:

<<https://github.com/N-CPUninter/FUNDAMENTOS-DE-DESENVOLVIMENTO-WEB/blob/main/html/transi%C3%A7%C3%B5es2.html>>

FINALIZANDO

Nesta abordagem, você teve um embasamento mais aprofundado sobre CSS. Apresentamos a Árvore do Documento (DOM) e a hierarquia de elementos. Podemos fazer uso das pseudoclasses e pseudoelementos para acessar partes do DOM e criar formatações muito interessantes com esses elementos. As possibilidades de estilização foram expandidas.

O que é necessário para a aplicação das CSS? Pratique, estude os exemplos aqui apresentados e os refaça.

Bons estudos!



REFERÊNCIAS

ALVES, W. P. **HTML e CSS**: aprenda como construir páginas web. São Paulo: Expressa, 2021.

CSS selector reference. **W³ Schools**, [s. d.]. Disponível em: <https://www.w3schools.com/cssref/css_selectors.asp>. Acesso em: 10 jan. 2023.

FUNDAMENTOS de desenvolvimento web: barra de navegação horizontal. **Github**, 30 out. 2022. Disponível em: <https://github.com/N-CPUninter/FUNDAMENTOS-DE-DESENVOLVIMENTO-WEB/blob/main/html/barra_navega%C3%A7%C3%A3o_horizontal.html>. Acesso em: 10 jan. 2023.

FUNDAMENTOS de desenvolvimento web: barra de navegação vertical. **Github**, 30 out. 2022. Disponível em: <https://github.com/N-CPUninter/FUNDAMENTOS-DE-DESENVOLVIMENTO-WEB/blob/main/html/barra_navega%C3%A7%C3%A3o_vertical.html>. Acesso em: 10 jan. 2023.

FUNDAMENTOS de desenvolvimento web: botões. **Github**, 30 out. 2022. Disponível em: <<https://github.com/N-CPUninter/FUNDAMENTOS-DE-DESENVOLVIMENTO-WEB/blob/main/html/botoes.html>>. Acesso em: 10 jan. 2023.

FUNDAMENTOS de desenvolvimento web: menu lista.html. **Github**, 30 out. 2022. Disponível em: <<https://github.com/N-CPUninter/FUNDAMENTOS-DE-DESENVOLVIMENTO-WEB/blob/main/html/menu%20lista.html>>. Acesso em: 10 jan. 2023.

FUNDAMENTOS de desenvolvimento web: transições. **Github**, 5 nov. 2022. Disponível em: <<https://github.com/N-CPUninter/FUNDAMENTOS-DE-DESENVOLVIMENTO-WEB/blob/main/html/transi%C3%A7%C3%B5es.html>>. Acesso em: 10 jan. 2023.

FUNDAMENTOS de desenvolvimento web: transições 1. **Github**, 5 nov. 2022. Disponível em: <<https://github.com/N-CPUninter/FUNDAMENTOS-DE-DESENVOLVIMENTO-WEB/blob/main/html/transi%C3%A7%C3%B5es1.html>>. Acesso em: 10 jan. 2023.



FUNDAMENTOS de desenvolvimento web: transições 2. **Github**, 5 nov. 2022. Disponível em: <<https://github.com/N-CPUninter/FUNDAMENTOS-DE-DESENVOLVIMENTO-WEB/blob/main/html/transi%C3%A7%C3%B5es2.html>>. Acesso em: 10 jan. 2023.

SILVA, M. S. **CSS3**. São Paulo: Novatec Editora. 2013.

_____. **Fundamentos de HTML5 e CSS3**. São Paulo: Novatec Editora, 2015.

TERUEL, E. C. **HTML5**: guia prático. 2. ed. rev. atual. e ampl. São Paulo: Érica, 2014.