



## GUIA DE LABORATÓRIO 2.1 - CORES E TEXTO

---

### OBJECTIVOS

- Introduzir os conceitos básicos de CSS
- Aplicação de estilos de cores
- Propriedades do texto: tipo e dimensão das letras, espaçamentos, alinhamento, etc.

### INSTRUÇÕES

1. Crie uma pasta `Lab2` dentro da pasta `LabsCSS`. Nesta pasta irá guardar o material desenvolvido neste laboratório. (Sugestão: pode guardar aqui o enunciado deste mesmo laboratório)

#### PARTE I – Introdução às CSSs

2. Inicie o Notepad++ (NP).
3. Crie um novo ficheiro na pasta `Lab2` com o nome `Parte1.html`.
4. Crie o esqueleto base de um ficheiro html, ou seja:

```
<!DOCTYPE html>
<html>
  <head>
  </head>
  <body>
  </body>
</html>
```

*Uma folha de estilos consiste de um conjunto de definições que indicam como é que o browser deve exibir os elementos html da página.*

*Os estilos podem ser definidos na página html ou num ficheiro à parte a ser incluído pelo ficheiro html. Por simplicidade, vamos começar por utilizar a primeira forma, mas a segunda é mais interessante porque permite que a folha de estilos seja partilhada por vários ficheiros html. Quando definimos os estilos no próprio ficheiro html, estes devem ficar dentro da etiqueta **style**.*

5. Vamos criar uma folha de estilos no próprio ficheiro html. Para tal devemos colocar a seguinte etiqueta no cabeçalho (**head**):

```
<!-- Seguem-se os estilos da folha de estilos -->
<style media="all" title="estilosgerais" type="text/css">

</style>
```

Utilizamos três atributos com a etiqueta **style**:

- . **type="text/css"**: indica ao browser que vamos usar CSS para especificar os estilos
- . **title="estilosgerais"**: uma identificação para a folha de estilos
- . **media="all"**: podemos usar este atributo para especificar uma folha de estilos para ser exibida no ecrã (**screen**), para impressão (**print**), para modo projecção (**projection**), em braile (**braille**), etc.; aqui indicamos que esta folha deve ser utilizada para todos (**all**) os tipos de 'media'.

6. Vamos definir o estilo geral do corpo da página. Acrescente a seguinte definição a **style**:

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

Os estilos em CSS são especificados através de regras. Uma regra possui três partes:

1. selector: indica ao navegador que parte ou partes do documento são afectadas pela regra; no nosso exemplo, a regra é para o elemento **body** apenas, ou seja, tudo dentro do **body** (que não seja afectado por uma regra mais específica) será abrangido por esta regra.
2. propriedades: entre chavetas ({ e }) especificamos uma lista de propriedades separadas por ponto-e-vírgula (;), uma por linha; uma propriedade corresponde a um aspecto do selector que queremos modificar; no exemplo anterior, utilizámos a propriedade **background-color** para mudar a cor de fundo para vermelho.
3. valores: para propriedade devemos indicar o valor respectivo; valor e propriedade são separados por dois pontos (:).

Existem vários tipos de selectores, mas isso fica para laboratórios mais à frente.

7. Modifique a propriedade anterior para

```
background-color: #e8eae8;
```

Qual o significado de #e8eae8? Em vários programas e, em particular, em páginas Web as cores podem ser especificadas misturando 256 tons de vermelho com 256 de verde e 256 de azul. Obtemos, assim,  $256^3$  cores diferentes (16.777.216). Em computação é comum escrever determinadas grandezas em binário ou hexadecimal e o mesmo acontece aqui, onde utilizámos hexadecimal para especificar a cor com maior precisão.

A cor de fundo em cima utilizada corresponde à mistura dos tons **e8** (232) de vermelho, **ea** (234) de verde e **e8** (232) de azul. Podemos também especificar as cores utilizando um dos 140 nomes disponíveis (**red**, **green**, **blue**, **gray**, **lightred**, **darkgreen**, etc.). Voltando ao hexadecimal, a cor **#FFFFFF** é o branco, ao passo que a cor **#000000** corresponde ao preto.

8. Vamos recorrer à propriedade **color** para acrescentar uma cor para o texto. Podemos acrescentar esta propriedade à regra anterior ou definir uma nova regra, que será combinada com a anterior:

1) `body {background-color: #e8eae8; color: #5d665b;}`

(repare na utilização do ponto-e-vírgula para separar as propriedades)

ou

2) `body {background-color: #e8eae8;}`

`body {color: #5d665b;}`

9. Adicione um cabeçalho de nível 1 com o texto: Texto do cabeçalho

10. E agora um parágrafo com o texto: Algum texto para testarmos as cores.

11. Acrescente uma margem ao `body` em todas as direcções utilizando a propriedade `margin`:

`margin: 50px;`

*Em CSS dispomos de várias unidades de medição. Aqui especificamos uma distância em pixels. Um pixel é um ponto no ecrã e a sua dimensão depende da resolução utilizada. Outras unidades disponíveis:*

*. cm, mm: centímetros e milímetros*

*. em: 1em corresponde a 1x o tamanho da letra escolhida, 2em o dobro,...*

*. % : percentagem do elemento envolvente*

*. in : polegadas (inches)*

*. pt, pc: ponto (1/72 de 1 polegada) e pica (12 pontos)*

*. px: pixels*

## PARTE II – Manipular propriedades do texto

12. Vamos modificar o tipo de letra dos elementos do `body` para uma letra *serif* usando, para tal, a propriedade `font-family`:

`font-family: Georgia, "Times New Roman", Times, serif;`

*As páginas Web só podem utilizar tipos de letra (fontes, doravante) que estejam disponíveis nos sistemas onde os navegadores de internet executam. Assim, não podemos simplesmente escolher as fontes que nos agradam mais. Devemos utilizar fontes comuns a todos os sistemas (como as acima indicadas).*

*Historicamente, as fontes tendem a dividir-se em três tipos: serif, sans-serif e mono-espaco (monospace). As fontes serif possuem pequenos detalhes no final de alguns traços das letras. Se esses detalhes não estiverem presentes, a fonte é sans-serif. Uma fonte mono-espaco é aquela em que todos os caracteres ocupam o mesmo espaço. Tipos mono-espaco são muito utilizados para escrever código de programas. Dentro das primeiras, encontramos as fontes Georgia, Times, Garamond, etc. No grupo das segundas, temos as fontes Arial, Trebuchet, Verdana, etc. Entre as fontes mono-espaco, encontramos fontes como Courier, Courier New, Lucida Console, etc.*

*Quando desenhamos páginas, estamos, na prática, limitados a dois ou três tipos de letras de cada categoria. Assim, a propriedade `font-family` permite que especifiquemos mais do que uma letra sendo que a primeira, caso esteja disponível, será a escolhida. Se não estiver, então passamos para a segunda e assim sucessivamente.*

*Todas as partes de um nome de fonte composto de várias partes devem estar envolvidas em aspas ("").*

*Quando indicamos `serif`, o navegador utilize a fonte serif por omissão. Em cima, esta fonte será utilizada se as fontes Georgia, "Times New Roman" e Times não estiverem instaladas no sistema operativo do navegador.*



13. Experimente agora os tipos *sans-serif*:

```
font-family: Verdana, Geneva, Arial, sans-serif;
```

14. Acrescente os seguintes parágrafos ao documento:

```
<p style="font-size: xx-large;">Enorme</p>
<p style="font-size: x-large;">Muito Grande</p>
<p style="font-size: large;">Grande</p>
<p style="font-size: medium;">Médio</p>
<p style="font-size: small;">Pequeno</p>
<p style="font-size: x-small;">Muito Pequeno</p>
<p style="font-size: xx-small;">Minúsculo</p>
```

*Os estilos CSS podem ser aplicados, elemento a elemento, em linha (inline), através do atributo **style** . Este atributo pertence ao conjunto de atributos padrão que todos os elementos possuem. No entanto, é aconselhável centralizar os estilos em uma ou mais folhas de estilo que, depois, podem ser partilhadas por vários documentos html.*

15. A propriedade **font-size** (utilizada em cima) permite indicar a dimensão da fonte. Vamos especificar um tamanho de font relativo adicionando a seguinte propriedade ao estilo do **body**:

```
font-size: small;
```

*Em tipografia o tamanho de uma fonte é definido como sendo a distância do ponto mais alto de uma maiúscula ao ponto mais baixo das letras minúsculas 'p' ou 'y'.*

*Podemos especificar tamanhos de letras utilizando tamanhos absolutos ou relativos. No primeiro caso, utilizamos as unidades de medida px ou px. No entanto, quando contemplamos uma audiência mais alargada, em que os utilizadores podem aceder à nossa página a partir de PCs ou de smartphones, é melhor considerarmos o segundo caso.*

*Uma dimensão pode ser relativa ao tamanho de letra utilizada. Por exemplo, **font-size: 70%** corresponde a 70% do tamanho da letra padrão utilizada. Outra forma de especificar dimensões de fontes relativas consiste em utilizar as etiquetas **xx-large**, **x-large**, ..., **xx-small**. Este último método é, em geral, preferível porque os browsers nunca deixam uma fonte **xx-small** atingir um tamanho que torne a fonte ilegível, ao passo que 70% do tamanho da letra pode, num dispositivo móvel, passar a fronteira da legibilidade.*

16. Substitua o texto do parágrafo no início do documento pelo seguinte texto:

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi.

17. Divida o parágrafo anterior em três:

```
<p>Lorem ipsum dolor ... aliquam erat volutpat.</p>
<p>Ut wisi enim ... ea commodo consequat.</p>
<p>Duis autem vel ... nulla facilisi.</p>
```

18. Vamos alterar o tamanho por omissão de uma linha, acrescentando a seguinte propriedade ao **body** na folha de estilos:

```
line-height: 160%;
```

*Por omissão, a distância entre linhas é 120% do tamanho da letra. Tormamos o texto mais legível se acrescentarmos um pouco mais de espaço. Mais uma vez, podemos especificar o espaçamento entre linhas em termos absolutos ou relativos. Aqui, especificámos em termos relativos (160%).*

19. Acrescente a seguinte definição à folha de estilos:

```
p {
  text-indent: 3em;
}
```

*Até agora temos controlado o estilo do documento como um todo, acrescentando definições ao selector **body**. A partir daqui vamos começar a individualizar os estilos por etiquetas começando com a etiqueta **p**.*

20. Modifique o texto do cabeçalho de nível 1 para “Experiências com CSS”.

21. Antes do primeiro e do último parágrafos acrescente os seguintes cabeçalhos de nível 3:

```
<h3>Lorem ipsum dolor</h3>
<h3>Ut wisi enim ad minim veniam</h3>
```

22. Modifique o estilo dos cabeçalhos de nível 3 acrescentando a seguinte definição à folha de estilos:

```
h3 {
  color: #966b72;
  font-family: Georgia, "Times New Roman", Times, serif;
}
```

23. Para ganhar mais espaçamento entre o segundo parágrafo e o segundo **h3** acrescente um parágrafo em branco.

`<p></p>`

Verifique se produziu algum efeito. Experimente inserir alguns espaços entre as etiquetas do parágrafo.

24. Insira o caractere `&nbsp;` dentro das etiquetas do parágrafo. Ou seja, deverá ficar assim:

`<p>&nbsp;</p>`

*Um parágrafo sem nada entre `<p>` e `</p>` é como se não existisse para os navegadores. Quando queremos usar um parágrafo vazio para “espaçar” dois elementos, não adianta acrescentar espaços em branco pois todos os espaços em branco que não sejam utilizados para separar palavras são ignorados pelo HTML. Podemos especificar alguns caracteres especiais em HTML colocando-os entre `&` e `;`. Por exemplo o `&copy;` produz o símbolo ©, o ç é exibido com `&ccedil;`, ao passo que do `&sect;` resulta o §. Neste caso utilizamos o non-breaking space (espaço que não quebra) cujo código é `&nbsp;`.*

25. O espaçamento “dentro” de um parágrafo também pode ser controlado. Acrescente as seguintes definições ao último parágrafo:

`<p style="letter-spacing: 0.5em; word-spacing: 0.5em;">Duis autem vel ...`

26. Por outro lado, controle o espaçamento do primeiro h3 com o seguinte estilo:

`<h3 style="margin-top: 2.5em;">Lorem ipsum dolor</h3>`

27. Podemos controlar as margens com maior precisão, utilizando variações da regra `margin`. Por exemplo, no `body`, substitua a regra `margin` pelas seguintes regras:

```
margin-top: 70px;
margin-left: 120px;
margin-right: 50px;
margin-bottom: 70px;
```