

Prática Laboratorial 06 – CSS

Objectivos:

- Aplicar estilos numa página HTML através de uma *stylesheet* externa.
- Conhecer vários tipos de selectores CSS.
- Conhecer algumas propriedades CSS relevantes.

Exercícios:

A) Criação de projecto

1. Abra o Visual Studio Code.
2. Aceda ao menu “File” / “Ficheiro”, e clique em “Open folder” / “Abrir pasta”. Escolha a pasta onde pretende colocar o código da actividade (sugere-se a criação de uma nova pasta com o nome “05_web_pl06”).

B) Criação de página HTML

1. Utilizando o separador “Explorer” / “Explorador”, clique no botão “New File...” para criar um novo ficheiro. Atribua-lhe o nome “index.html”.
2. No ficheiro HTML criado, digite o símbolo “!”. Deverá ser aberto um *popup* com a indicação “Emmet Abbreviation”. Clique na primeira opção desse *popup* ou carregue na tecla Tab. O ficheiro HTML será pré-preenchido com alguma estrutura base.
3. Altere o título da página (no elemento “title” contido no “head”) para “Demonstração de estilos via CSS externo”.
4. Utilize a extensão Live Server para visualizar a página à medida que for fazendo alterações. Tente colocar as janelas do VS Code e do *browser* lado a lado para melhor visualizar o recarregamento da página após cada alteração. Se tiver dois monitores, poderá dispor o VS Code num e o *browser* noutro.
5. Comece por definir a estrutura semântica da página, criando elementos “header”, “main” e “footer” dentro do “body”.

C) Implementação de cabeçalho da página

1. Dentro do elemento “header”, insira um elemento de imagem, utilizando o logótipo do CESAE Digital (https://www.cesaedigital.pt/wp-content/uploads/2016/07/logo_cesae-cores_horizontal_header_site.png) e o texto alternativo “Logótipo do CESAE Digital”.
2. Como verificará, a imagem aparecerá em tamanho grande.

- Utilizando o separador “Explorer” / “Explorador” do VS Code, clique no botão “New File...” para criar um novo ficheiro. Atribua-lhe o nome “style.css”.
- Nesse ficheiro de estilos, declare o seguinte:

```
img.logo {  
  width: 300px;  
}
```

- De seguida, voltando ao ficheiro HTML, acrescente um atributo “class” com o valor “logo” ao elemento de imagem referente ao logótipo. O logótipo passará a aparecer no ecrã com uma largura de 300px (e altura calculada automaticamente pelo *browser*, de modo a manter o *ratio* da imagem). Tal sucede porque passou a ser um elemento imagem com a classe “logo”, o que corresponde ao selector declarado no ficheiro CSS.
- Ainda dentro do elemento HTML “header” da página, crie novo elemento “nav”, constituído por uma lista não ordenada de elementos: Início, Sobre nós, Serviços, Contactos.
- Envolva cada um desses 4 itens em hiperligações, de modo a que o texto dos itens seja o texto da hiperligação. O atributo “href” desses *links* poderá ser, respectivamente, “”, “about”, “services” e “contacts”.
- Por fim, atribua uma “class” de “navbar” ao elemento “nav”.
- O código HTML do “header” deverá agora ser algo semelhante a isto:

```
<header>  
    
  
    <nav class="navbar">  
      <ul>  
        <li><a href="">Início</a></li>  
        <li><a href="about">Sobre nós</a></li>  
        <li><a href="services">Serviços</a></li>  
        <li><a href="contacts">Contactos</a></li>  
      </ul>  
    </nav>  
</header>
```

- No ficheiro “style.css” deverá agora declarar os estilos seguintes.

```
.navbar {  
  float: right;  
}  
  
.navbar li {  
  list-style-type: none;  
  display: inline;  
  padding: 5px;  
}  
  
.navbar li a {  
  color: black;  
  text-decoration: none;  
  font-variant: small-caps;  
}  
  
.navbar li a:hover {  
  color: gray;  
  text-decoration: underline;  
}
```

Note que estamos a declarar estilos para elementos com a classe “navbar”, assim como para os seus descendentes (não necessariamente directos) “li” e para os elementos “a” descendentes dos “li”. Repare ainda que na última declaração, estamos a aplicar um selector de pseudoclassee, que indica que o estilo em causa deve ser aplicado apenas no momento de *hover* (rato por cima do texto) das hiperligações. Analise cada um dos estilos declarados e tente perceber a que corresponde cada uma das propriedades e respectivos valores.

11. O cabeçalho da sua página deverá apresentar-se assim:



INÍCIO SOBRE NÓS SERVIÇOS CONTACTOS

Ao passar o rato por cima de cada um dos elementos do menu criado, o texto do mesmo deverá aparecer sublinhado e numa cor ligeiramente diferente.

12. No ficheiro “style.css” deverá agora criar uma declaração utilizando o selector universal (*). Utilize-a para definir o valor da propriedade “font-family” para “Impact, Haettenschweiler, 'Arial Narrow Bold', sans-serif”. (Ao começar a escrever o valor no VS Code, é provável que este valor completo lhe apareça como opção para selecção.)
13. Verifique como a fonte do texto das opções do menu se alterou. Ao utilizar o selector universal, foi dada a indicação de que todos os elementos de texto da página deverão usar as regras de *font* definidas. Se a *font* Impact estiver disponível, será a utilizada pelo *browser*; caso contrário, deverá tentar aplicar a *font* seguinte; e assim sucessivamente, até que, no final, caso nenhuma das anteriores possa ser usada, o *browser* deverá escolher uma da família sans-serif.

C) Implementação do “corpo” da página

1. Vamos agora adicionar alguns elementos ao “corpo” da página HTML.
2. Dentro do elemento “main” comece por adicionar um elemento “article”.
3. Dentro dele crie um parágrafo com um texto à sua escolha.
4. No ficheiro “style.css” crie uma nova regra aplicável ao elemento “main”, que defina valores para as propriedades “width”, “padding”, “margin” e “background-color” nos moldes que se seguem.

```
main {  
  width: 80%;  
  padding: 20px 50px;  
  margin: 50px auto;  
  
  background-color: antiquewhite;  
}
```

Repare como o conteúdo do elemento “main” passou a aparecer centrado horizontalmente, ocupando apenas parte do ecrã, com uma margem de 50px acima e abaixo, com uma cor de fundo e com um espaçamento entre o conteúdo (para já, só um parágrafo de texto) e as fronteiras do elemento “main”.

5. Retorne agora ao ficheiro “index.html” e atribua ao elemento do parágrafo (“p”) uma classe “first-letter-big”.
6. No ficheiro “style.css” introduza a declaração seguinte.

```
article p.first-letter-big::first-letter {  
    font-size: 32px;  
    margin-right: 5px;  
}
```

Esta declaração algo complexa indica valores para as propriedades “font-size” e “margin-right” para a primeira letra dos parágrafos que tenham a classe “first-letter-big” e que sejam descendentes de um elemento “article”. Repare que foram usados selectores de elemento (“article” e “p”), de descendente (o espaço entre “article” e “p”), de classe (o ponto a seguir ao “p”, seguido do nome de classe “first-letter-big”) e de pseudoelemento (“::first-letter”).

7. Os elementos de estilo anteriormente declarados resultam numa alteração do texto do parágrafo para o que se segue (com a primeira letra com um tamanho maior, assim como um espaçamento mais largo entre ela e a letra seguinte).

Lorem ipsum dolor sit amet consectetur adipisicing elit. Eum aperiam
perferendis animi eligendi unde ea ipsa eius, molestiae iste possimus soluta
porro sed velit dolorem deleniti cupiditate praesentium harum? Fugiat.

8. Vamos agora criar uma tabela e estilizá-la.
9. Comece por declarar um elemento “table” com os seus subelementos semânticos “caption”, “thead” e “tbody”.
10. Defina uma legenda à sua escolha para incluir no elemento “caption”.
11. Dentro do “thead” crie uma linha (“tr”) e três células de cabeçalho (“th”). Atribua os textos “1.ª Coluna”, “2.ª Coluna” e “3.ª Coluna” (ou outros à sua escolha) a cada uma das células.
12. De seguida, crie quatro registos (cada um com três células) dentro do elemento “tbody”.
Sugere-se:
“A”, “B” e “C” para a primeira linha;
“α”, “β” e “γ” (códigos para as letras gregas alfa, beta e gama) para a segunda;
“1”, “2” e “3” para a terceira;
“I”, “II”, “III” para a última.
13. Vamos agora estilizar a tabela e respectivos elementos.
14. Comece por declarar o valor “0 auto” à propriedade “margin” do elemento “table”. Isto centrará a tabela já criada (e outras que eventualmente venham a ser criadas posteriormente).

```
table {  
  margin: 0 auto;  
}
```

15. De seguida, crie uma regra para o elemento “caption”.

```
table caption {  
  font-size: 12px;  
  color: #666;  
  caption-side: bottom;  
}
```

Neste caso estamos a utilizar o selector de descendente, mas poderia ser utilizado o de filho ou até apenas o selector de elemento “caption” (sem necessidade de ser descendente de um elemento “table”).

16. Repare como a legenda da tabela mudou de cor, de tamanho de texto e até de posição – face à tabela –, tal como definido na propriedade especial “caption-side”.
17. Crie agora regra de estilos para definir a cor de fundo “darkgrey” e um “padding” de 10px em todas as células de cabeçalho.

```
table th {  
  background-color: darkgrey;  
  padding: 10px;  
}
```

18. A cor das células alterou para a indicada e passaram a ter um espaçamento de 10px (em toda a volta) entre o conteúdo das mesmas e as suas fronteiras.
19. Vamos agora centrar o conteúdo de todas as células da tabela. Tal pode ser feito atribuindo o valor “center” à propriedade “text-align” de todos os elementos “th” e “td”. Por exemplo, assim:

```
table th,  
table td {  
  text-align: center;  
}
```

20. Note que utilizámos um selector de grupo, aplicando o mesmo estilo a dois tipos de elemento diferentes (“th” e “td”), em vez de criar duas declarações separadas. No caso dos elementos “th”, para os quais já existia a declaração do passo 17, passam a ser aplicados os estilos de cada uma das regras. Como não há sobreposição de regras relativamente a uma mesma propriedade, não há qualquer problema. Se houvesse sobreposição de regras numa qualquer propriedade, aplicar-se-ia a com maior prioridade.
21. O conteúdo de todas as células das tabelas está agora centrado horizontalmente.
22. Por fim, vamos definir cores alternadas para cada uma das linhas da tabela.
23. As pseudoclasses “first-child”, “last-child” ou “nth-child(n)” permitem seleccionar elementos de um determinado tipo que são – em relação ao seu elemento “pai”,

respectivamente – o primeiro filho, o último filho, ou o filho *n*. No caso do “*nth-child*”, o valor de *n* pode ser numérico (1, 2, 3, etc.) ou assumir os valores “*odd*” e “*even*” que correspondem aos elementos ímpares e pares.

24. Neste caso, vamos definir cores para linhas alternadas da seguinte forma:

```
table tr:nth-child(odd) {  
    background-color: #fefefe;  
}  
table tr:nth-child(even) {  
    background-color: rgba(0, 0, 230, 0.1);  
}
```

Para os elementos “*tr*” que são o primeiro, terceiro, quinto, etc. “filhos” do seu elemento “pai” (que será o elemento “*table*”), atribuímos a cor (no formato hexadecimal) “#fefefe”.

Para os que são segundo, quarto, etc. “filhos” do seu elemento “pai”, atribuímos uma cor usando a função *rgba()*, com um valor de 230 de azul e uma opacidade de 10%.

25. Posto isto, o “corpo” da nossa página deverá apresentar-se assim:

Lorem ipsum dolor sit amet consectetur adipiscing elit. Eum aperiam
perferendis animi eligendi unde ea ipsa eius, molestiae iste possimus soluta
porro sed velit dolorem deleniti cupiditate praesentium harum? Fugiat.

1.ª Coluna	2.ª Coluna	3.ª Coluna
A	B	C
α	β	γ
1	2	3
I	II	III

Legenda da tabela

D) Implementação do rodapé da página

26. Dentro do elemento “*footer*” do HTML, crie novo parágrafo com a indicação de *copyright* com o símbolo respectivo, o seu nome e o ano actual. Algo assim:

```
<footer>  
    <p>&copy; Bruno Leal, 2024</p>  
</footer>
```

27. Agora envolva o seu nome num elemento “*span*” com a classe “*text-green*”.

28. No ficheiro “*style.css*”, declare um estilo que atribua o valor “*green*” à propriedade “*color*” dos elementos com a classe “*text-green*”. Utilize apenas o selector de classe, sem indicar especificamente a que tipo de elemento se deve aplicar.

29. Repare que o seu nome passou a aparecer a verde.

30. No ficheiro HTML, remova o atributo “*class*” do “*span*” e coloque-o no elemento “*p*”.

31. Repare que todo o texto passou a verde, dado que a declaração feita no CSS não especificava o tipo de elemento, mas apenas o nome da classe.