



GUIA DE LABORATÓRIO 4.2 - CAIXAS - PARTE 2

OBJECTIVOS

- Aprofundar os conceitos da parte 1 (laboratório 4.1) e introduzir o *CSS Box-Model*.
- Aprender a trabalhar com outros tipos de posicionamento.

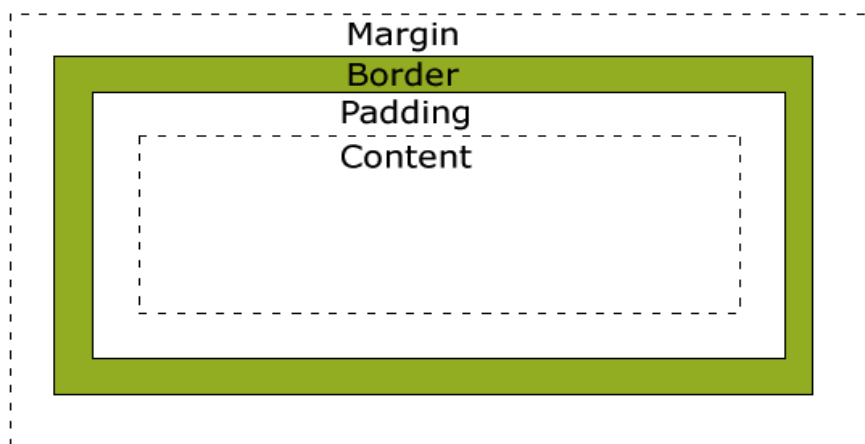
INSTRUÇÕES

1. Aceda à pasta `Lab4` dentro da pasta `LabsCSS`. Nesta pasta irá guardar o material desenvolvido neste laboratório. (Sugestão: pode guardar aqui o enunciado deste mesmo laboratório)

PARTE I – Dimensões

2. Inicie o Notepad++ (NP).
3. Crie o esqueleto base de um ficheiro html (ver guia de laboratório 2) e grave-o com nome `Lab42Parte1.html`.
4. Vamos por começar por acrescentar uma caixa com outra dentro, utilizando para tal a etiqueta `div`:

```
<div id="caixa1">  
  <div id="caixa2">  
  </div>  
</div>
```



5. Quer a `caixa1`, quer a `caixa2`, deverão começar por possuir as seguintes propriedades:

- . limite sólido, espessura de 3px e cor vermelha (`caixa1`) ou verde (`caixa2`)
- . 200 px de largura e de altura

Insira as declarações CSS correspondentes no atributo `style` da `div`. Que observa? Os limites estão sobrepostos? Porquê?

6. Modifique as dimensões da `caixa2` para 194x194 px. A `caixa2` deverá ficar completamente contida dentro de `caixa1`, com os respectivos limites encostados.

7. Acrescente uma margem interna (`padding`) de 3 px a esta caixa. Após testar no navegador, volte a modificar as dimensões de `caixa2` de modo a que esta volte a estar completamente dentro e encostada a `caixa1`.

8. Acrescente uma margem externa de 3px a `caixa2`.

Redimensione a área de conteúdo (propriedades `width` e `height`) de forma apropriada, ou seja, de modo a que `caixa2` fique centrada dentro de `caixa1` (note que, a partir do momento que adiciona uma margem externa, já não é possível que os limites fiquem encostados).

9. Modifique a dimensão de `caixa1` para 204x204 px.

10. Volte a “centrar” a `caixa2` dentro da `caixa1` utilizando para tal a propriedade `padding` na `caixa1`.

Quando indicamos a largura e altura de uma caixa através das propriedades `width` e `height`, apenas o estamos a fazer para a área reservada para o conteúdo (`content`) do elemento. Conforme dá para ver na imagem anterior, a estas dimensões acrescem ainda a dimensão das margens internas e externas e a espessura dos limites.

A largura total de um elemento deve ser calculada da seguinte forma:

```
Largura total = largura +  
                margem interna esquerda +  
                margem interna direita +  
                margem externa esquerda +  
                margem externa direita +  
                limite da esquerda      +  
                limite da direita
```

A altura total de um elemento deve ser calculada da seguinte forma:

```
Altura total = altura +  
                margem interna superior +  
                margem interna inferior +  
                margem externa superior +  
                margem externa inferior +  
                limite superior          +  
                limite inferior
```

PARTE II – Outros posicionamentos

11. Crie o esqueleto base de um ficheiro html (ver guia de laboratório 2) e grave-o com nome

`Lab42Parte2.html`.

12. Vamos por começar por acrescentar uma caixa:

```
<div id="caixa1">  
</div>
```

13. Coloque o seguinte parágrafo dentro de `caixa1`.

```
<p>Vamos explorar alguns tipos de posicionamento. Para tal, vamos relacionar  
esta caixa com o <em>&lt;body&gt;</em> e com outras caixas dentro desta.</p>
```

14. Acrescente as seguintes propriedades ao atributo `style` do `body`:

- . limite grosso (`thick`), sólido e vermelho
- . dimensão de 500x500 px

15. Agora, para `caixa1`, as propriedades são:

- . limite grosso, sólido e preto
- . margem interna de 10 px

16. Após testar no navegador, acrescente a seguinte propriedade a `caixa1`:

```
width: 30%;
```

17. Após testar no navegador, modifique a largura para 100% e teste. O que se passou e porquê?
Coloque 50%.

Relembrando aqui o que mencionámos no laboratório anterior sobre posicionamento, uma caixa pode ser posicionada em qualquer parte da página dando o valor `fixed` à propriedade `position`, ou seja:

```
position: fixed;
```

A propriedade `position` indica o tipo de posicionamento de um elemento html. Possui os seguintes valores:

-- `static`: posiciona os elementos por ordem, tal como surgem no documento html; valor por omissão

-- `absolute`: posiciona um elemento face ao primeiro ascendente posicionado; um elemento "posicionado" é um elemento cuja propriedade `position` possui um valor diferente de `static`.

-- `fixed`: posiciona o elemento face à janela do browser

-- `relative`: posiciona o elemento face à posição que ocuparia, ie, face à sua posição normal (conforme veremos num dos próximos laboratórios)

-- `inherit`: herda o posicionamento do ascendente

18. Acrescente a seguinte propriedade a `caixa1`:

```
position: absolute;
```

19. Após testar no navegador, adicione, também, a propriedade anterior ao `body`. Como explica o que se passou?

20. Remova a propriedade `position` de ambos os elementos, e adicione as seguintes a `caixa1`.

```
top: 100px; left: 100px;
```

Teste. Nota alguma diferença?

21. Adicione a propriedade `position` a ambos os elementos, e, nos dois casos, dê o valor `static` à propriedade. Teste. Nota alguma diferença agora?

22. Modifique o posicionamento de `caixa1` para `absolute`. Teste. Agora deverá notar a diferença...
23. Após testar, faça o mesmo ao `body` e volte a testar. Também aqui houve uma alteração. Repare como as propriedades `top` e `left` dependem do tipo de posicionamento.
24. Remova o parágrafo da `caixa1`.
25. Vamos modificar as dimensões da `caixa1` e reposicioná-la com as seguintes propriedades:

```
width: 50%; height: 50%;  
top: 25%; left: 25%;
```

26. Remova as propriedades referentes aos limites. Acrescente uma cor de fundo:

```
background-color: black;
```

27. Adicione a `caixa1` uma outra `div` com `id` `caixa2` e com as seguintes propriedades:

```
background-color: green;  
width: 100px; height: 100px;  
top: 20px; left: 20px;
```

28. Sem especificarmos o tipo de posicionamento, as propriedades `top` e `left` são inúteis. Adicione posicionamento absoluto a `caixa2`. Que aconteceria se tivesse adicionado posicionamento estático?
29. Modifique o valor de `position` para `fixed`.
30. Adicione uma margem externa ao `body` de 50 px. Teste. Que observa e porquê?

As CSS trabalham em 3 dimensões: largura, altura e profundidade. As primeiras duas dimensões são controladas pelas propriedades `width` e `height`, ao passo que trabalhamos com a 3a dimensão através da propriedade `z-index`. Um valor mais elevado significa maior “proximidade” de quem vê a página, ao passo que um valor mais pequeno indica maior “afastamento”. Por omissão, todos os elementos estão ao mesmo nível de afastamento, nível 0, e a sobreposição é resolvida através de algumas regras pré-definidas (é exibido o elemento posicionado em último lugar).

Ao darmos -1 ao `z-index` de um elemento, indicamos que ele está no patamar -1 que é um patamar mais afastado do que o patamar 0. A propriedade `z-index` apenas funciona para elementos posicionados, ie, aqueles cuja `position` é diferente de `static`.

A utilização da propriedade `z-index` é mais complicada do que parece e será aprofundada num dos próximos laboratórios.

31. Nesta altura a `caixa2` deverá estar por cima do limite do `body`. Adicione a seguinte propriedade a `caixa2`.

```
z-index: -1;
```

32. Nesta caixa, reponha o valor de `position` em `absolute`. Para onde foi a caixa? Como fazê-la reaparecer dentro de `caixa1`?

33. Remova a propriedade `z-index` (se a tiver adicionado para responder às perguntas anteriores) e `position`. Remova também as propriedades `top` e `left`.

34. Adicione agora a `caixa3`, também dentro de `caixa1` e a seguir a `caixa2`. Deverá possuir as seguinte propriedades:

```
background-color: yellow;  
width: 80px; height: 80px;
```

35. Acrescente o seguinte parágrafo a `caixa2`:

```
<p style="color: white;">Algum texto de teste.</p>
```

36. Após testar, modifique a `position` de `caixa2` e `caixa3` para `static` (note que é o mesmo que não ter a propriedade...). Notou diferenças nesta última transição?

37. Acrescente a seguinte propriedade a `caixa2` e teste.

```
top: -100 px;
```

38. De seguida modifique o posicionamento de `caixa2` para `relative`. Temporariamente mude-o para `absolute`. Quais as diferenças e porquê?

39. Experimente modificar o posicionamento das `caixas2` e 3, tentando perceber as diferenças.