

Aula 3 - CSS

Docupedia Export

Author:Gouveia Raissa (CtP/ETS)

Date:05-Mar-2024 17:01

Table of Contents

1	Div	3
2	Class	4
3	Id	5
4	Span	6
5	Elementos estruturais	7
5.1	Header	7
5.2	Main	7
5.3	Footer	8
5.4	Section	8
5.5	Article	8
5.6	Aside	8
5.7	Nav	8
6	CSS	12
6.1	Tipos de Seletores	13
6.2	Fundamentos Estruturais	14
6.2.1	Posicionamento	22

1 Div

É uma tag que diz respeito à divisões. Usa-se essa tag para separar conteúdo e manter o controle sobre a estrutura da página, existe a possibilidade de ter múltiplas "div" em uma página. É um container genérico para conteúdo de fluxo, que de certa forma não representa nada. Ele pode ser utilizado para agrupar elementos para fins de estilos (usando class ou id), ou porque eles compartilham valores de atributos.

2 Class

Normalmente utilizada como parametro de uma "div", para determinar qual classe aquela divisão e seu conteúdo pertencem, é fundamental para localizar o elemento na folha de estilo ou permitir manipulação de script.

```
1  <body>
2
3  <div class="cidade">
4    <h2>Londres</h2>
5    <p>Capital da Inglaterra.</p>
6  </div>
7
8  <div class="cidade">
9    <h2>Paris</h2>
10   <p>Capital da França.</p>
11 </div>
12
13 <div class="cidade">
14   <h2>Tokyo</h2>
15   <p>Capital do Japão.</p>
16 </div>
17
18 </body>
```

3 Id

É mais um atributo da "div", com uma configuração única, ou seja, só se pode utilizar um ID em apenas um único elemento. Já uma classe é uma configuração que pode ser utilizada em mais de um elemento.

```
1  <body>
2  <div id="cabecalho">
3    <h1>Capitais</h1>
4  </div>
5  <div id="corpo">
6    <div class="cidade">
7      <h2>Londres</h2>
8      <p>Capital da Inglaterra.</p>
9    </div>
10
11   <div class="cidade">
12     <h2>Paris</h2>
13     <p>Capital da França.</p>
14   </div>
15
16   <div class="cidade">
17     <h2>Tokyo</h2>
18     <p>Capital do Japão.</p>
19   </div>
20 </div>
21 </body>
```

Foi dividido com duas divs pais, cada uma contendo um ID única (cabecalho e corpo). Dentro da nossa div com a id corpo, recebemos duas divisões para o conteúdo. Observe que essas divisões receberam uma mesma classe: cidade. Portanto, podemos estilizar as divs através de suas ids, e ainda podemos estilizar todas as divs que contêm a mesma classe de uma só vez.

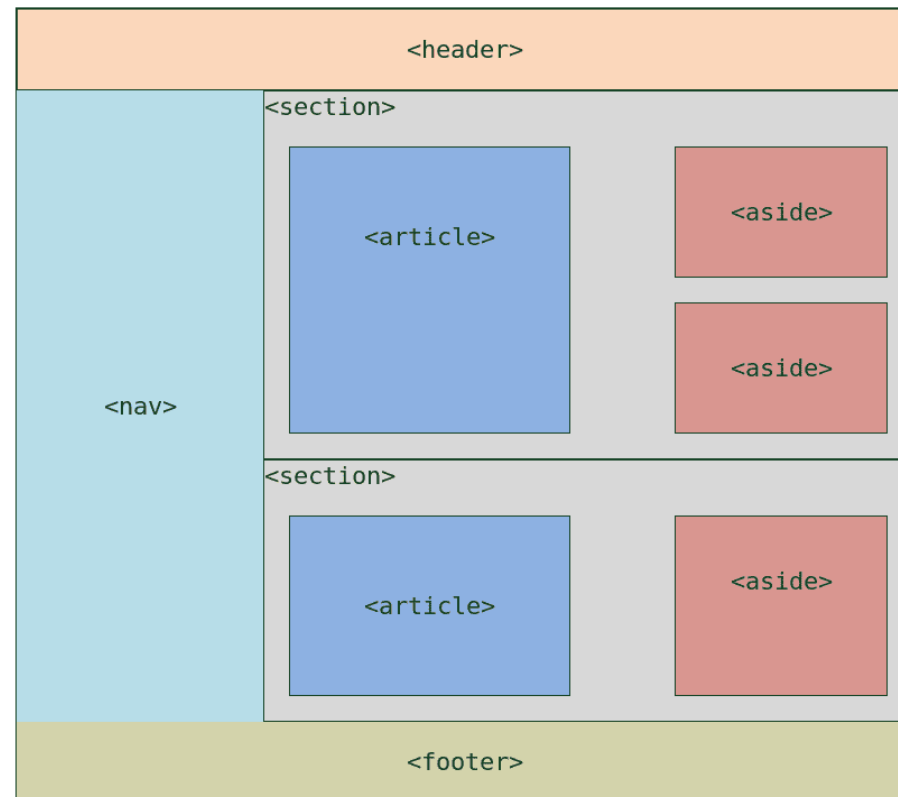
4 Span

É muito parecido com o elemento "div", entretando "div" é um elemento de nível de bloco enquanto "span" é um elemento em linha.

```
1  <body>
2  <div id="cabecalho">
3    <h1>Capitais</h1>
4  </div>
5  <div id="corpo">
6    <div class="cidade">
7      <h2>Londres</h2>
8      <span id="idspan">Capital da Inglaterra.</span>
9    </div>
10
11   <div class="cidade">
12     <h2>Paris</h2>
13     <span id="idspan">Capital da França.</span>
14   </div>
15
16   <div class="cidade">
17     <h2>Tokyo</h2>
18     <span id="idspan">Capital do Japão.</span>
19   </div>
20 </div>
21 </body>
```

5 Elementos estruturais

São elementos dentro da tag body definem a estrutura essencial de uma página. O principal objetivo é retirar essa responsabilidade das divs. Tendo como principal diferencial a navegação semântica com HTML 5.



5.1 Header

Configura o cabeçalho de uma página. Cuidado para não confundir com a tag "head", que serve para informar características da página. O que existe no "header" será, de fato, exibido no site. Geralmente é um o título de um artigo, slogan, logotipo de uma empresa/site, menu ou barra de navegação.

5.2 Main

Define o conteúdo principal dentro do "body" do documento ou aplicação. Entende-se como conteúdo principal aquele relacionado diretamente com o tópico central da página ou com a funcionalidade central da aplicação.

5.3 Footer

Representa um rodapé para o conteúdo, normalmente contém informações sobre o autor da seção de dados, direitos autorais ou links para documentos relacionados.

5.4 Section

É utilizada para marcar as seções de conteúdo de uma página. Com Esse elemento se agrupa de forma lógica nosso conteúdo, separando a informação em áreas diferentes.

5.5 Article

Utiliza-se quando existe a necessidade de declarar um conteúdo que não precisa de outro para fazer sentido em um documento HTML, por exemplo, um artigo em um blog. É recomendado identificar cada "article" com um título.

5.6 Aside

Define um bloco de conteúdo que faz referência ao conteúdo principal que o cerca, pode estar ou não em uma barra lateral (sidebar) exibindo informações contextuais, assuntos relacionados, conteúdo publicitário ou um grupo de navegação secundária. Quando usado dentro de um elemento article, seu conteúdo está diretamente relacionado com o conteúdo do artigo, quando usado fora de um elemento article, está relacionado com o conteúdo global do documento como, por exemplo, uma lista de links de um blog, blocos de navegação complementar e também publicidade relacionada com conteúdo do documento.

5.7 Nav

Representa uma seção de uma página que aponta para outras páginas ou para outras áreas da página, ou seja, uma seção com links de navegação. Consiste basicamente em um menu ou barra de navegação.

```
1  <!doctype html>
2  <html lang="pt-br">
3      <head>
4          <meta charset="UTF-8">
5          <title>Cafe Fontenebleu</title>
6      </head>
7
8      <body>
9          <main>
10
11              <header>
12                  <nav>
13                      <ul>
14                          <li><a href="home.html">Home</a></li>
15                          <li><a href="info.html">Info</a></li>
16                          <li><a href="special-events.html">Special Events</a></li>
17                          <li><a href="dvd.html">DVD</a></li>
18                          <li><a href="location.htm">Location</a></li>
```



```

19         <li><a href="menu.html">Menu</a></li>
20         <li><a href="review.html">Review</a></li>
21     </ul>
22 </nav>
23 </header>
24 <article>
25     <header>
26         <hgroup>
27             <h1>Restaurant Review</h1>
28             <h5>Leading experts talk about the restaurant</h5>
29         </hgroup>
30     </header>
31
32     <p>Francesco and Jane Bienni, founders and owners of San Francisco's world-class restaurant, have many reasons
to celebrate. Cafe Townsend has been a San Francisco favorite for more than 19 years, and their latest restaurant venture,
Cafe Isabella, will soon expand to more locations. Their biggest celebration is for their son Steven's recent graduation
from New York's highly acclaimed Culinary Institute of America. San Francisco diners should join the Bienni's in their
celebration because this graduation means an even better menu selection at Cafe Townsend, as Steven is lending his modern
and innovative culinary talents to his father's traditional, Old World kitchen.</p>
33
34     <aside>
35         <p>What I am doing is using seasonal Italian ingredients and combining them to create dishes with a modern
flair that will invigorate our customers' palates, Steven says.</p>
36     </aside>
37
38     <p>Those who have enjoyed Cafe Townsend and its world-class cuisine shouldn't worry - the stellar tableside
service, scrumptious regional specials and menu favorites remain. But Steven's latest menu additions are something you
don't want to miss.</p>
39     <p>"The utilization of regional seafood with the accompaniment of our fresh-grown herbs is something I'm avid
about."</p>
40 </article>
41
42 <div id="apoio">
43     <section class="fotos_apoio">
44         <h2>Mapa</h2>
45         
46     </section>
47
48     <section class="fotos_apoio">

```

```

49         <h2>Fotos</h2>
50         
51     </section>
52 </div>
53
54     <footer>Cafe Fontenebleu - Rua Ernesto de Paula Santos, 187 Sala 506 - Recife-PE</footer>
55
56 </main>
57 </body>
58
59 </html>

```

Resultado:

[Home](#)
[Info](#)
[Special Events](#)
[DVD](#)
[Location](#)
[Menu](#)
[Review](#)

Restaurant Review

Leading experts talk about the restaurant

Francesco and Jane Bienni, founders and owners of San Francisco's world-class restaurant, have many reasons to celebrate. Cafe Townsend has been a San Francisco favorite for more than 19 years, and their latest restaurant venture, Cafe Isabella, will soon expand to more locations. Their biggest celebration is for their son Steven's recent graduation from New York's highly acclaimed Culinary Institute of America. San Francisco diners should join the Bienni's in their celebration because this graduation means an even better menu selection at Cafe Townsend, as Steven is lending his modern and innovative culinary talents to his father's traditional, Old World kitchen.

“

What I am doing is using seasonal

Those who have enjoyed Cafe Townsend and its world-class cuisine shouldn't worry - the stellar tableside service, scrumptious regional specials

Mapa

Italian ingredients and combining them to create dishes with a modern flair that will invigorate our customers' palates, Steven says.

and menu favorites remain. But Steven's latest menu additions are something you don't want to miss.

"The utilization of regional seafood with the accompaniment of our fresh-grown herbs is something I'm avid about."

Fotos



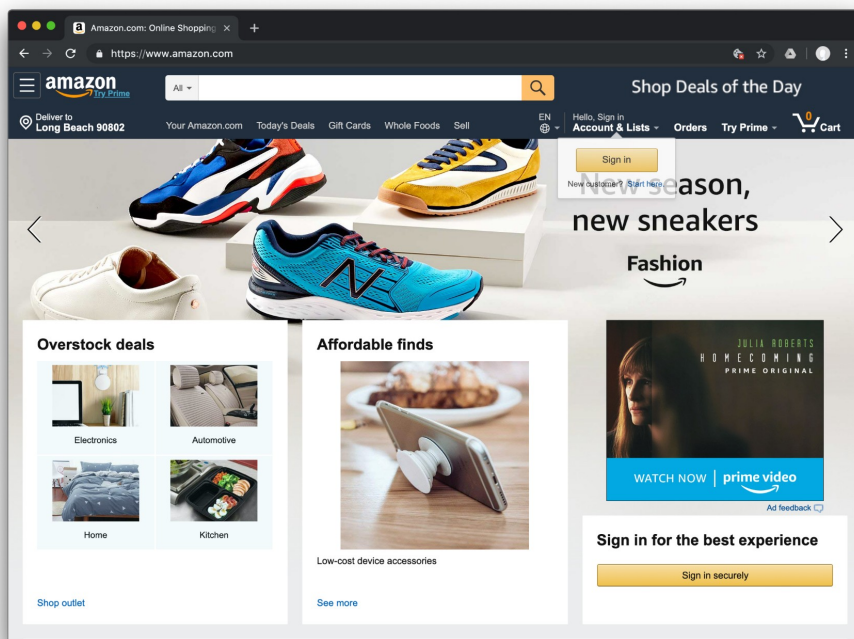
Cafe Fontenebleu - Rua Ernesto de Paula Santos, 187 Sala 506 - Recife-PE

6 CSS

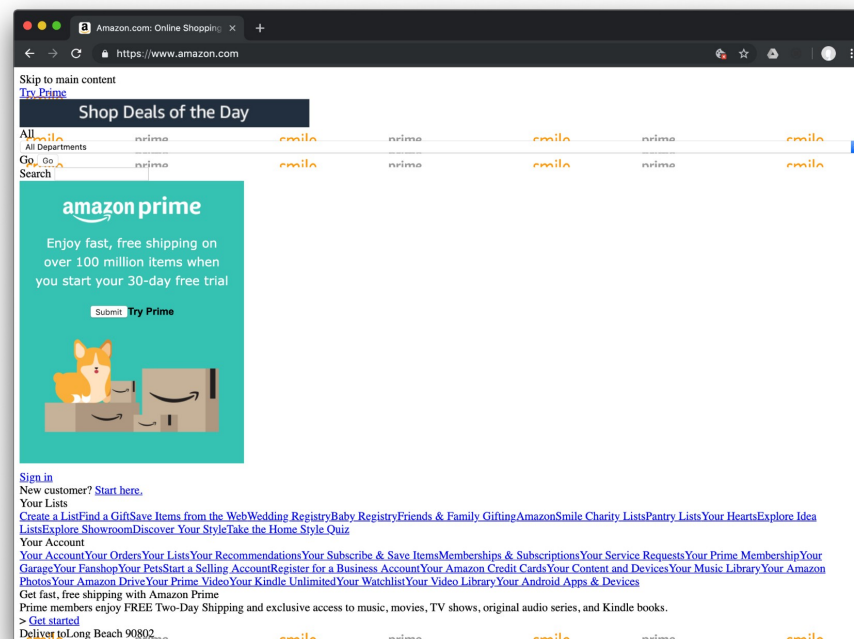
Assim como o HTML, o CSS não é realmente uma linguagem de programação. Também não é uma linguagem de marcação — é uma *linguagem de estilização ou seja, um folha de estilo*. Isso significa que o CSS permite aplicar estilos seletivamente a elementos em documentos HTML.

A imagem abaixo mostra o site da amazon com apenas codificação HTML, e a diferença quando a folha de estilo é acrescentada.

Html + CSS



Html



Na aplicação do estilo CSS, teremos 3 tipos de estilo a serem aplicados:

- Inline Style: Atribuído na própria tag HTML, com utilização do atributo "style".

1 `<p style="color:blue">Texto de Cor Azul</p>`

- Internal Style: Estará incorporado na própria folha HTML. Dentro do "head" do documento e com a tag "style".

```

1 <head>
2   <style>
3     p{color:blue}
4   </style>
5 </head>

```

- External Style: Quando existe-se um arquivo exclusivo para elaborar o estilo da página, ou seja, todo o conteúdo de estilo estará dentro de um documento separado chamado *style.css*. Para conectar a folha de estilo com a página principal deve-se usar o elemento "link" no head do documento. O atributo "rel" faz jus ao tipo de relação entre os arquivos, o atributo "type" refere-se ao tipo de documento que esta sendo linkado e por fim "href" define o diretório do arquivo css.
- Sendo essa a mais utilizada pelos desenvolvedores.

```

1 <head>
2   <link rel="stylesheet" type="text/css" href="style.css">
3 </head>

```

Regra de CSS

seletor	declaração
p	<div> <div>Propriedade</div> <div>font-family: Arial, Helvetica, sans-serif;</div> <div>color:#F00</div> <div>Valor</div> </div>

6.1 Tipos de Seletores

Em CSS a forma de declarar o seletor é diferente para tipo de elemento, seguindo o padrão das chaves e a declaração das propriedades.

- TAGS

```
1 p{
2   color:#00ff00;
3 }
```

- Classes

```
1 .cidade{
2   color:#00ff00;
3 }
```

- ID

```
1 #idspan{
2   color:#00ff00;
3 }
```

6.2 Fundamentos Estruturais

A estrutura de uma página nada mais é que uma serie de containers alinhados que separam o conteúdo. Vimos anteriormente qual seria a estrutura em HTML agora veremos como de fato construir essa estrutura com parâmetros estabelecidos com CSS.

Para exemplificar criaremos um container com três divisões de conteúdo:

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Aula 3</title>
5   <style type="text/css">
6     </style>
7 </head>
8 <body>
9   <div class="container">
10     Container
11     <div class="box1">BOX 1</div>
12     <div class="box2">BOX 2</div>
13     <div class="box3">BOX 3</div>
14   </div>
15
```

```
16 </body>
17 </html>
```

Note são exibidos apenas os textos sem separação nenhuma, pois foi dada dimensão as divisões, para isso utilizamos as funções:

- **Width:** que determina a largura do elemento;
- **Height:** que determina a altura do elemento;

Ambos podem ser definidos em px ou em % (considerando o tamanho do monitor). Inicie as classes de CSS dentro da tag style para cada elemento e altere da seguinte forma:

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Aula 3</title>
5   <style type="text/css">
6     .container{
7       width:100%;
8       height:400px;
9     }
10    .box1{
11      width:100px;
12      height:100px;
13    }
14    .box2{
15      width:100px;
16      height:100px;
17    }
18    .box3{
19      width:100px;
20      height:100px;
21    }
22  </style>
23 </head>
24 <body>
25   <div class="container">
26     Container
27     <div class="box1"></div>
28     <div class="box2"></div>
29     <div class="box3"></div>
30   </div>
```

```
31
32 </body>
33 </html>
```

Para melhor visualização utilizamos:

- **Background-color:** define a cor de plano de fundo do elemento, pode ser determinado pelo nome da cor ou códigos hexadecimais;
- **Border:** Cria um contorno na divisão, é composto por três parâmetros: espessura, tipo de borda e cor;

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Aula 3</title>
5   <style type="text/css">
6     .container{
7       width:100%;
8       height:400px;
9       background-color:#000000;
10    }
11    .box1{
12      width:100px;
13      height:100px;
14      background-color:#5d5d5d;
15      border: 1px solid white;
16    }
17    .box2{
18      width:100px;
19      height:100px;
20      background-color:#5d5d5d;
21      border: 1px solid white;
22    }
23    .box3{
24      width:100px;
25      height:100px;
26      background-color:#5d5d5d;
27      border: 1px solid white;
28    }
29  </style>
30 </head>
31 <body>
```



```
32     <div class="container">
33         Container
34     <div class="box1"></div>
35     <div class="box2"></div>
36     <div class="box3"></div>
37 </div>
38
39 </body>
40 </html>
```

Agora podemos visualizar a dimensão de cada divisão, a palavra "Container" já não é mais visível pois colocamos a mesma cor de fundo, para alterar utilize:

- Color: define a cor da fonte;

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Aula 3</title>
5      <style type="text/css">
6          .container{
7              width:100%;
8              height:400px;
9              background-color:#000000;
10             color:#ffffff;
11         }
12         .box1{
13             width:100px;
14             height:100px;
15             background-color:#5d5d5d;
16             border: 1px solid white;
17             color:#ffffff;
18         }
19         .box2{
20             width:100px;
21             height:100px;
22             background-color:#5d5d5d;
23             border: 1px solid white;
24             color:#ffffff;
25         }
26         .box3{
```

```

27         width:100px;
28         height:100px;
29         background-color:#5d5d5d;
30         border: 1px solid white;
31         color:#ffffff;
32     }
33 </style>
34 </head>
35 <body>
36     <div class="container">
37         Container
38         <div class="box1"></div>
39         <div class="box2"></div>
40         <div class="box3"></div>
41     </div>
42
43 </body>
44 </html>

```

Existe uma estrutura porém ainda não esta ordenada, as caixas por exemplo colocam-se uma em baixo da outra sendo que existe todo o espaço da tela por onde elas podem se distribuir, para alinha-las lado a lado, usa-se:

- **Float:** Determina a flutuação, onde um elemento deve ser retirado do seu fluxo normal e colocado ao longo do lado direito ou esquerdo do seu contêiner, onde textos e elementos em linha irão se posicionar ao seu redor. Para flutuação do lado esquerdo e direito usamos left e right respectivamente.
- **Clear:** limpa qualquer tipo de flutuação.

```

1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Aula 3</title>
5      <style type="text/css">
6          .container{
7              width:100%;
8              height:400px;
9              background-color:#000000;
10             color:#ffffff;
11         }
12         .box1{
13             width:100px;

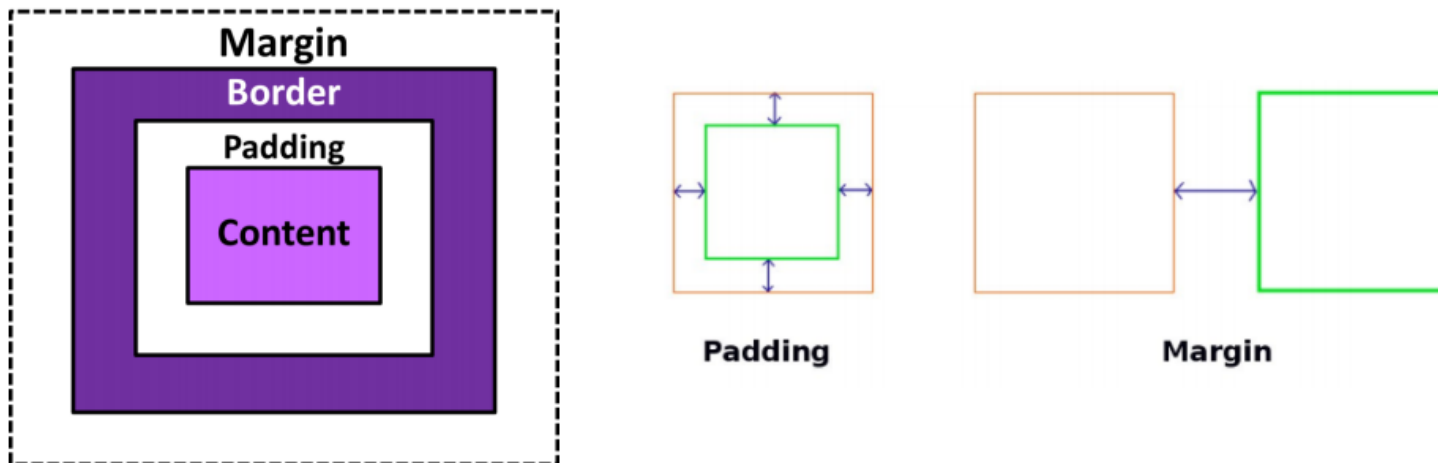
```

```
14         height:100px;
15         background-color:#5d5d5d;
16         border: 1px solid white;
17         color:#ffffff;
18         float:left;
19     }
20     .box2{
21         width:100px;
22         height:100px;
23         background-color:#5d5d5d;
24         border: 1px solid white;
25         color:#ffffff;
26         float:left;
27     }
28     .box3{
29         width:100px;
30         height:100px;
31         background-color:#5d5d5d;
32         border: 1px solid white;
33         color:#ffffff;
34     }
35 </style>
36 </head>
37 <body>
38     <div class="container">
39         Container
40         <div class="box1"></div>
41         <div class="box2"></div>
42         <div class="box3"></div>
43     </div>
44
45 </body>
46 </html>
```

Agora as caixas estão uma ao lado da outra, mas ainda parecem um único elemento pois não existe margem, no css para estruturarmos isso é necessário entender alguns conceitos.

Para deixar o layout da página mais limpo e organizado, existem atributos específicos em CSS, que são responsáveis por criar espaçamentos entre os conteúdos da página, Margin e Padding são dois desses atributos que iremos utilizar:

- **Margin** é utilizado para criar margens em torno dos elementos, **fora** de qualquer borda definida. Pode ser estipulado uma margem para todos os lados do elemento ou uma para cada lado, utilizando dos parâmetros: *margin-right*, *margin-left*, *margin-top*, *margin-bottom*;
- **Padding**: ou preenchimento, é usado para gerar espaço em torno do conteúdo de um elemento **dentro** de qualquer borda definida.



Ajuste:

```

1  <!DOCTYPE html>
2  <html>
3  <head>
4    <title>Aula 3</title>
5    <style type="text/css">
6      .container{
7        width:100%;
8        height:400px;
9        background-color:#000000;
10       color:#ffffff;
11     }
12     .box1{
13       width:100px;
14       height:100px;
15       background-color:#5d5d5d;
16       color:#ffffff;
17       float:left;
18       margin-right:100px;

```

```
19         padding: 10px;
20     }
21     .box2{
22         width:100px;
23         height:100px;
24         background-color:#5d5d5d;
25         color:#ffffff;
26         float:left;
27         margin-right:100px;
28         padding: 10px;
29     }
30     .box3{
31         width:100px;
32         height:100px;
33         background-color:#5d5d5d;
34         color:#ffffff;
35         float:left;
36         margin-right:100px;
37         padding: 10px;
38     }
39     .box4{
40         width:1000px;
41         height:300px;
42         background-color:#ff0000;
43         color:#ffffff;
44         padding: 10px;
45     }
46
47     </style>
48 </head>
49 <body>
50     <div class="container">
51         Container<br/>
52         <div class="box1">Caixa 1</div>
53         <div class="box2">Caixa 2</div>
54         <div class="box3">Caixa 3</div>
55     </div>
56
57 </body>
```

58

</html>

6.2.1 Posicionamento

Podemos dizer que esses estilos definem as “coordenadas” de posição dos elementos. Tendo em mente esse ponto importante, vamos analisar agora cada possível valor para o estilo **position**.

- O primeiro valor que iremos analisar é o estilo **position: static**. Esse é o valor padrão para a posição de qualquer elemento. Quando o aplicamos, definimos que o elemento irá ser posicionado de acordo com o fluxo normal da aplicação, ou seja, sua posição não será afetada.
- O valor **relative** define que o elemento poderá ter sua posição alterada. A posição dele será relativa a si mesmo, à sua posição inicial.
- O **position fixed**, como o próprio nome diz, define que o elemento ficará fixo na página. Ou seja, ele estará sempre visível. Mesmo que rolemos o scroll (barra de rolagem), ele continuará sendo exibido.
- **Absolute**, ou absoluto. Ela faz com que o elemento “saia” da hierarquia estabelecida pelo HTML – na prática, ele é filho da página e não de um elemento qualquer no HTML definido. Assim, é possível posicionar esse elemento em qualquer lugar da página, independentemente do que temos lá.

Assim vamos definir uma posição relativa para as caixas e alinha-las ao centro do layout:

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4    <title>Aula 3</title>
5    <style type="text/css">
6      .container{
7        width:100%;
8        height:400px;
9        background-color:#000000;
10       color:#ffffff;
11     }
12     .box1{
13       position: relative;
14       left:200px;
15       top:100px;
16       width:100px;
17       height:100px;
18       background-color:#5d5d5d;
19       color:#ffffff;
20       float:left;
21       margin-right:100px;
22       padding: 10px;
```

```
23     }
24     .box2{
25         position: relative;
26         left:400px;
27         top:100px;
28         width:100px;
29         height:100px;
30         background-color:#5d5d5d;
31         color:#ffffff;
32         float:left;
33         margin-right:100px;
34         padding: 10px;
35     }
36     .box3{
37         position: relative;
38         left:600px;
39         top:100px;
40         width:100px;
41         height:100px;
42         background-color:#5d5d5d;
43         color:#ffffff;
44         float:left;
45         margin-right:100px;
46         padding: 10px;
47     }
48     </style>
49 </head>
50 <body>
51     <div class="container">
52         Container
53         <div class="box1"></div>
54         <div class="box2"></div>
55         <div class="box3"></div>
56     </div>
57
58 </body>
59 </html>
```

Vamos criar mais uma caixa ao fundo, adicione:

```

1    ....box4{
2        position: fixed;
3        left:180px;
4        top:50px;
5        width:1000px;
6        height:300px;
7        background-color:#ff0000;
8        color:#ffffff;
9        padding: 10px;
10    }...
11    ... <div class="box4">Caixa 4</div>...

```

Note que a caixa vermelha sobrepõe as outras, mas a queremos no fundo, para isso utilizamos:

- **Z-index:** Esta funcionalidade permite a especificação da ordem de pilha de um elemento, o elemento com maior ordem de pilha fica a frente de outros elementos.

Adicione z-index:1 para a caixa vermelha, e z-index:2 para o restante.

► Observe que o Item de “z-index : 2” permanece visível, por seu index ser de valor mais prioritário. Caso alterasse o valor dele para um número menos prioritário do que o do outro elemento, este ficaria “abaixo”.

```

1    <!DOCTYPE html>
2    <html>
3    <head>
4        <title>Aula 3</title>
5        <style type="text/css">
6            .container{
7                width:100%;
8                height:400px;
9                background-color:#000000;
10               color:#ffffff;
11            }
12            .box1{
13                position: relative;
14                left:200px;
15                top:100px;
16                width:100px;
17                height:100px;
18                background-color:#5d5d5d;
19                color:#ffffff;

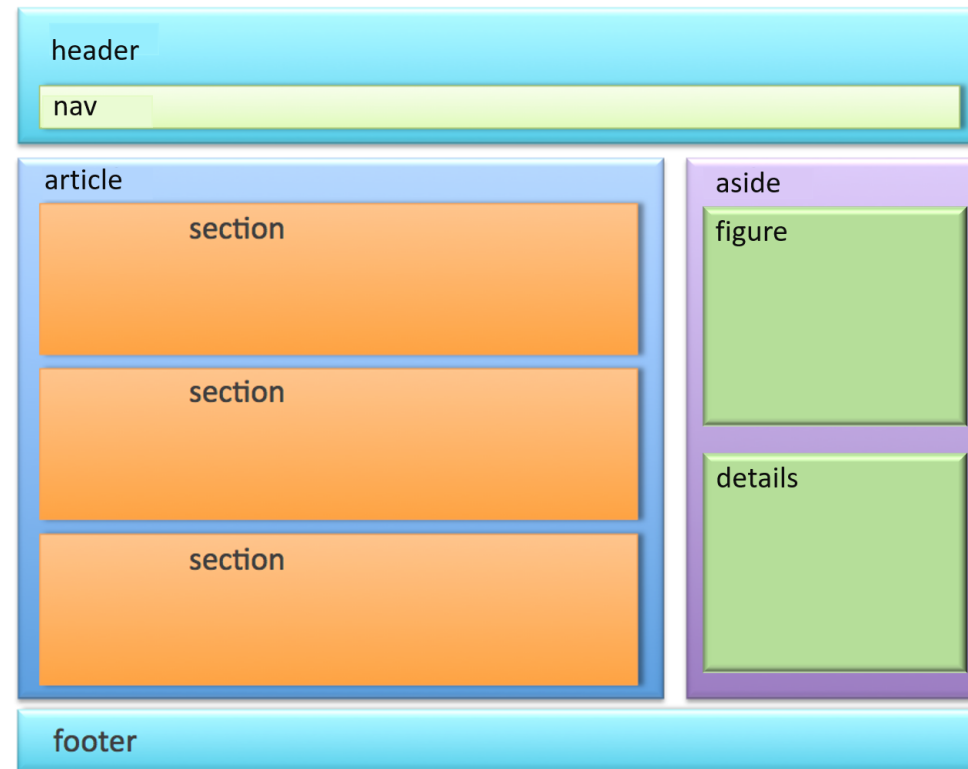
```



```
20     float:left;
21     margin-right:100px;
22     padding: 10px;
23     z-index: 2;
24 }
25 .box2{
26     position: relative;
27     left:400px;
28     top:100px;
29     width:100px;
30     height:100px;
31     background-color:#5d5d5d;
32     color:#ffffff;
33     float:left;
34     margin-right:100px;
35     padding: 10px;
36     z-index: 2;
37 }
38 .box3{
39     position: relative;
40     left:600px;
41     top:100px;
42     width:100px;
43     height:100px;
44     background-color:#5d5d5d;
45     color:#ffffff;
46     float:left;
47     margin-right:100px;
48     padding: 10px;
49     z-index: 2;
50 }
51 .box4{
52     position: fixed;
53     left:180px;
54     top:50px;
55     width:1000px;
56     height:300px;
57     background-color:#ff0000;
58     color:#ffffff;
```

```
59         padding: 10px;
60         z-index: 1;
61     }
62
63     </style>
64 </head>
65 <body>
66     <div class="container">
67         Container<br/>
68         <div class="box1">Caixa 1</div>
69         <div class="box2">Caixa 2</div>
70         <div class="box3">Caixa 3</div>
71         <div class="box4">Caixa 4</div>
72     </div>
73
74 </body>
75 </html>
```

Com os conhecimentos adquiridos até o momento, tente criar a seguinte estrutura:



Utilizando os elementos semânticos:

```
<body>
  <header>
    <nav></nav>
  </header>
  <main>
    <article>
      <section></section>
      <section></section>
      <section></section>
    </article>
    <aside>
      <figure></figure>
```

```
        <details></details>
      </aside>
</main>
<footer></footer>
</body>
```