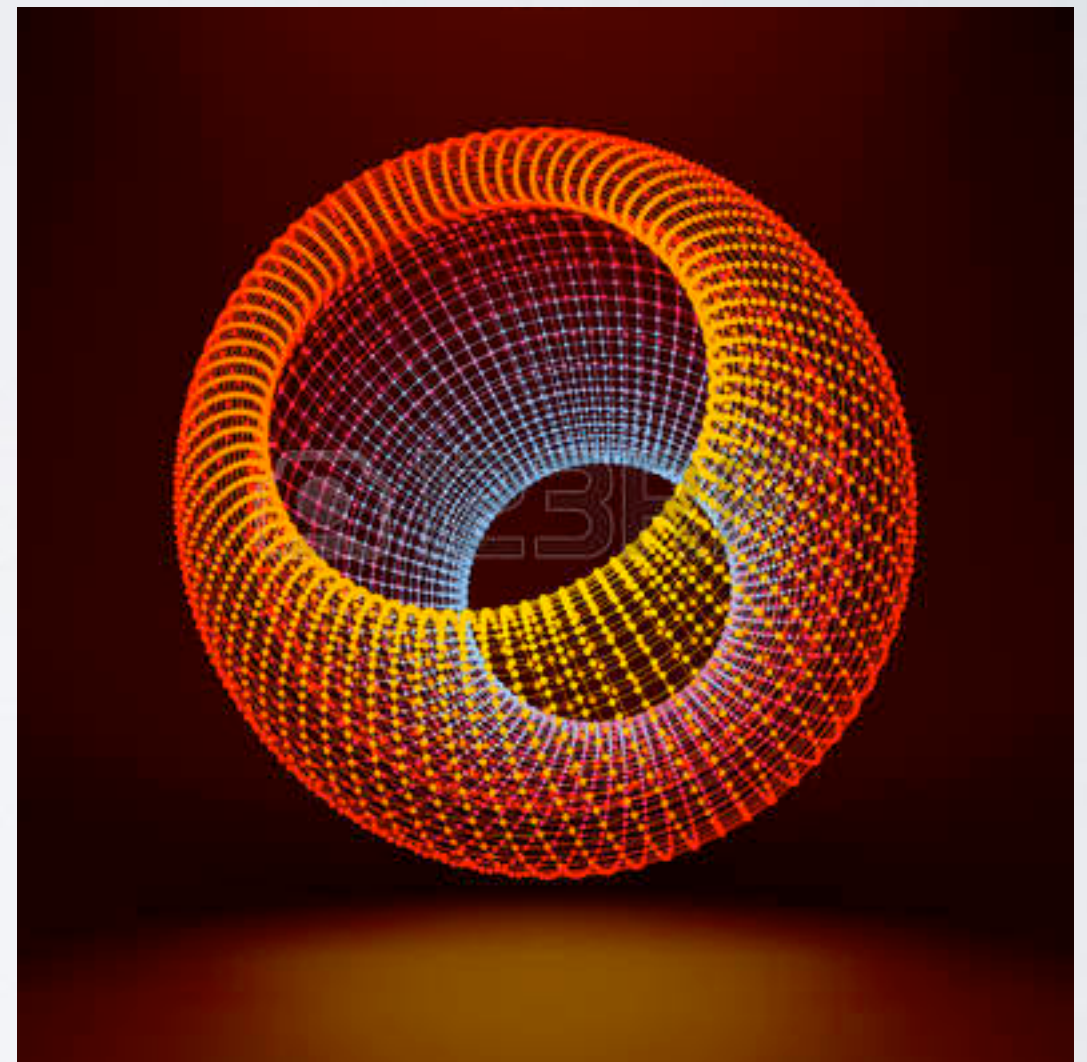


# CSS - SELETORES: CLASS X ID

Prof. André Caruso

# SELETOR CLASSE

- Classes são definidas com uma palavra iniciada por “.”
- Uma classe pode ser usada inúmeras vezes em um mesmo documento
- Usa-se o atributo “class=” para aplicar uma classe a um elemento / tag
- Exemplo CSS:  
`.negrito {font-weight: bold;}`



```
<!DOCTYPE html>
<html>
<head>
<title>IFSUL</title>
<meta charset="utf-8" />
<style>
    /* seletores por classe */
    .vermelho {color: red;}
    .verde {color: green;}
</style>
</head>
<body>
    <h1> Este é o <span class="vermelho">título</span> 1</h1>
    <h2 class="verde"> Este é o título 2 </h2>
</body>
</html>
```

exClass.html

# SELETOR ID

- IDs ou identificadores são definidos por uma palavra iniciada com “#”
- Um ID só pode ser usado uma vez e uma **única** vez em um documento HTML !!!
- Usa-se o atributo “id=” para aplicar um ID a um elemento / tag
- Exemplo CSS:  
`#principal {color: red;}`



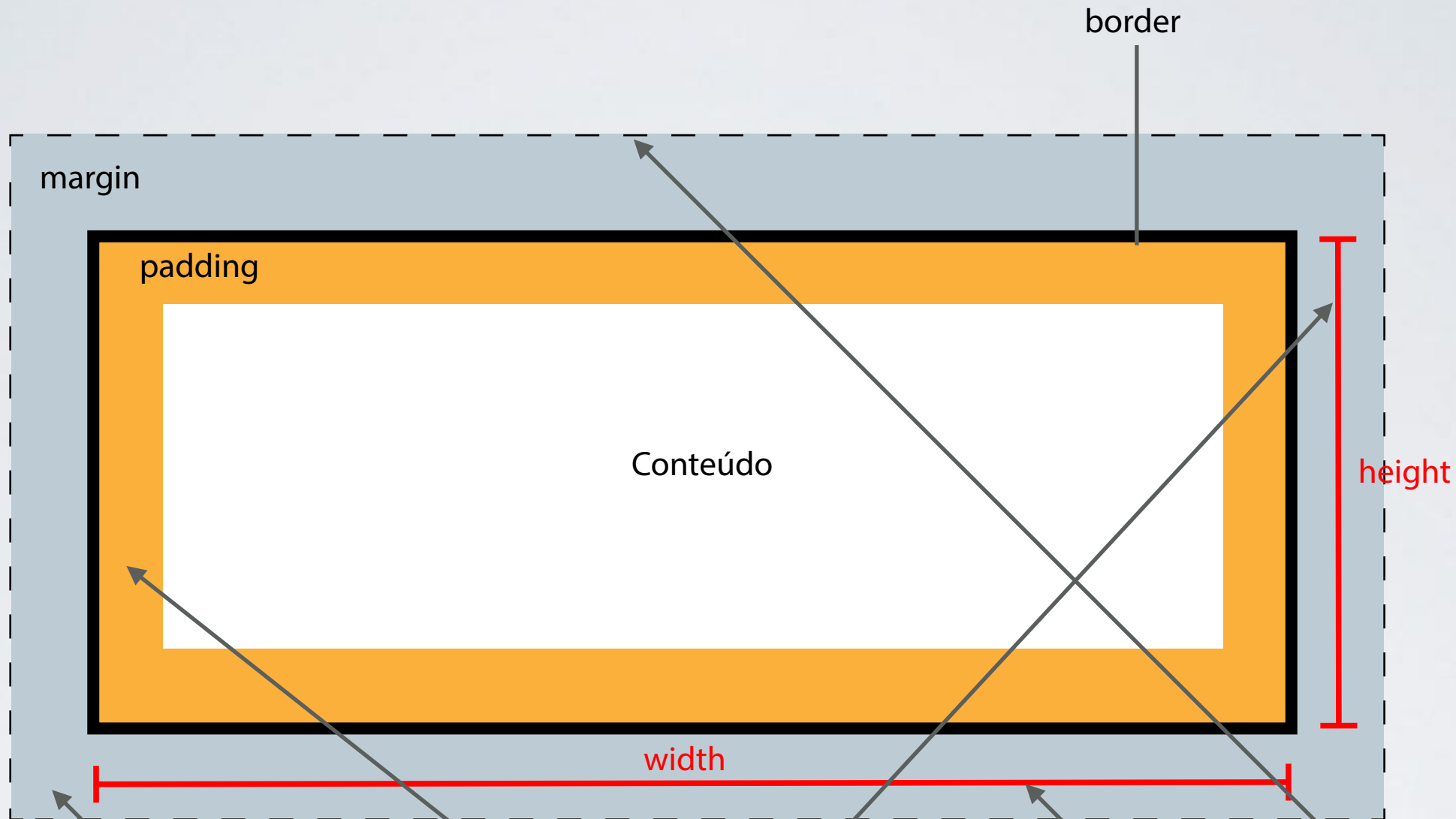


# USANDO “CAIXAS” (DIVS)

Organizando um documento XHTML como "gente grande" :)

# Box Model:

## tamanho, bordas e margens de uma div



Por padrão as medidas de padding (margem interna), border (borda externa) e margin (margem externa) se somam à width (largura) e a height (altura)!

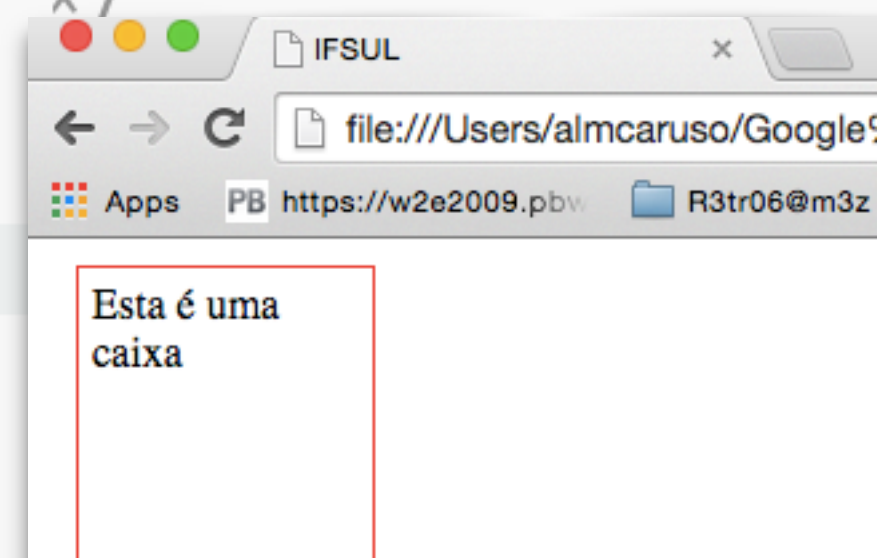


## Dica

- É possível mudar o comportamento do box model, através da propriedade `box-sizing`
- `box-sizing: border-box;`
  - fará que as propriedades `width` e `height` incluam as larguras do conteúdo, borda e padding, porém margem continua sendo somada à largura e altura da caixa.
- Exemplo interativo:
  - [https://www.w3schools.com/cssref/tryit.asp?filename=trycss3\\_box-sizing2](https://www.w3schools.com/cssref/tryit.asp?filename=trycss3_box-sizing2)

```
<!DOCTYPE html>
<html>
<head>
<title>IFSUL</title>
<meta charset="utf-8" />
<style>
    #dv01{
        border: 1px red solid; /* borda: largura cor tipo */
        padding: 5px; /* distancia interna à borda */
        margin: 10px; /* distancia externa à borda*/
        width: 100px; /* largura da caixa */
        height: 100px; /* altura da caixa */
    }
</style>
</head>
<body>
    <div id="dv01">Esta é uma caixa</div>
</body>
</html>
```

A tag DIV serve para criarmos uma "caixa"

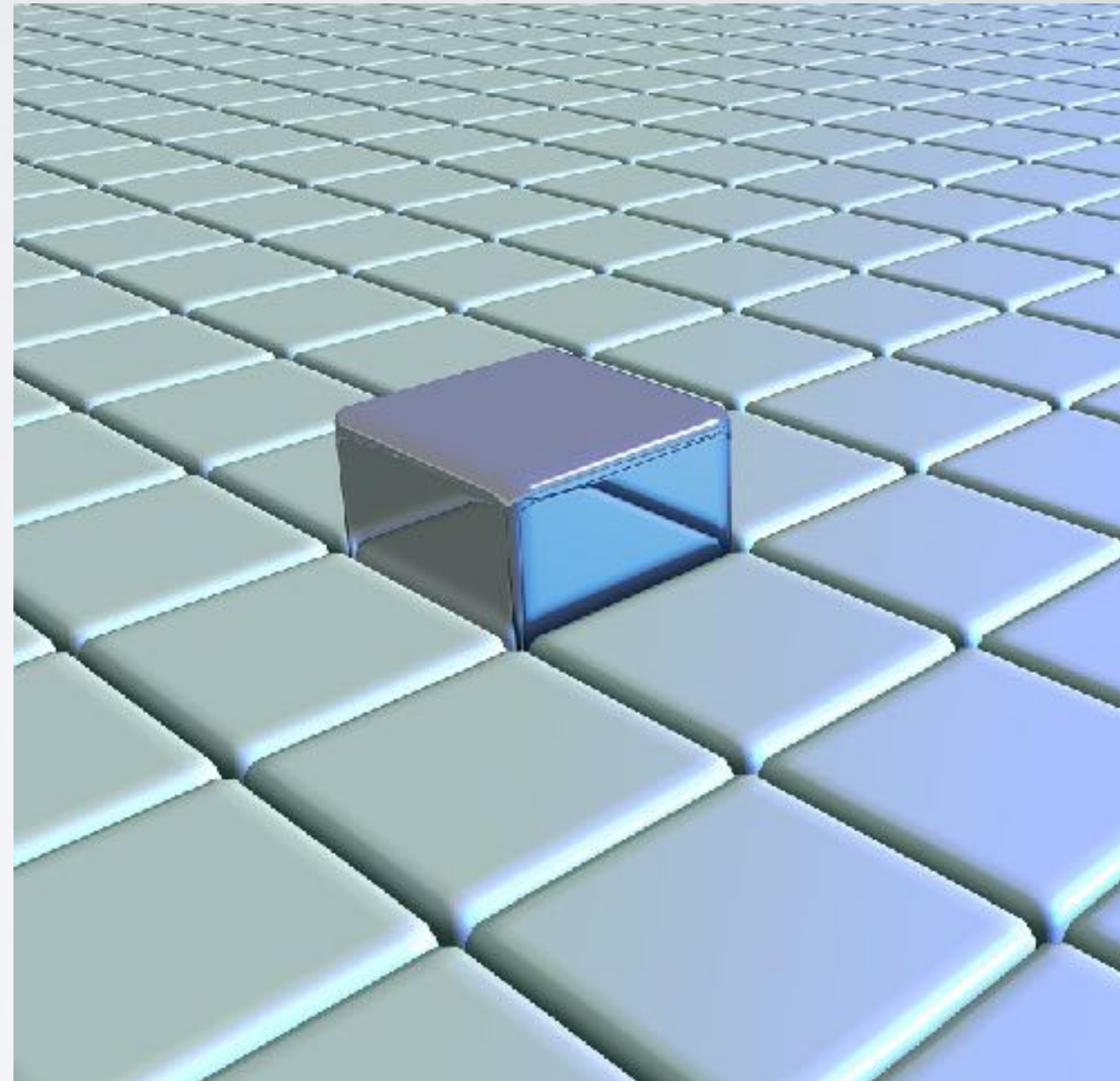


exemplo\_box01.html



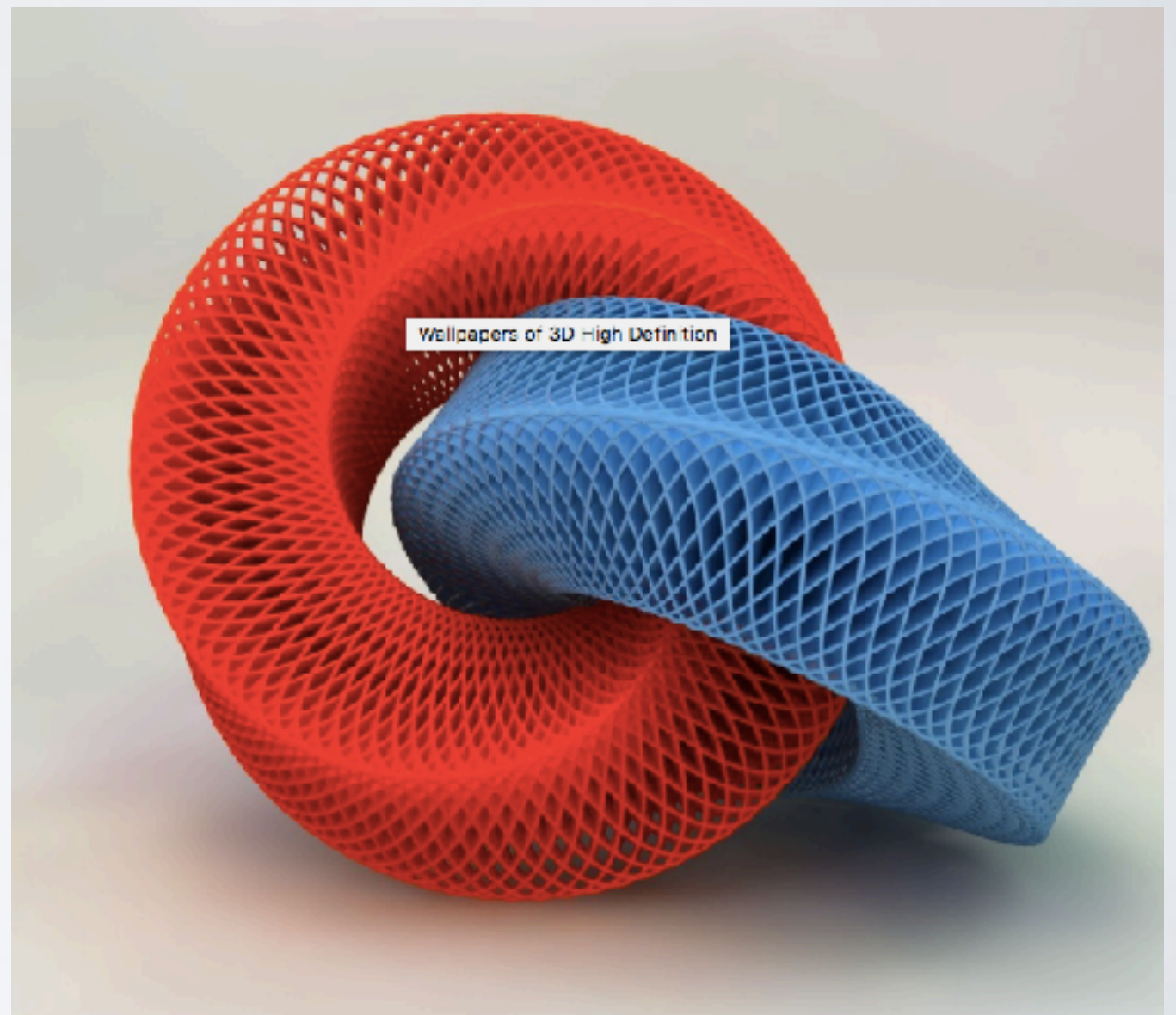
# UNIDADES DE MEDIDA

- **Comumente as propriedades width e height são medidas em:**
  - **px (pixels na tela)**
  - **% (percentual da área definida pela div pai)**
- **Há ainda duas unidades especiais que referem-se às medidas da janela do navegador:**
  - **vw (largura da janela)**
  - **vh (altura da janela)**
    - **Estas unidades foram adicionadas para corrigir incompatibilidades e algumas dores de cabeça causadas pela forma com que cada navegador (e o Internet Explorer) implementavam o comportamento dos valores percentuais!**
- **Mais sobre vw e vh:**
  - <https://brasa.art.br/2015/dicas-e-truques-htmlcssjs/vw-e-vh-unidades-de-medida-para-css-matadoras/>



# LARGURA E ALTURA MÁXIMAS

- Além de width e height é possível ainda definir:
  - min-width (largura mínima)
  - max-width (largura máxima)
  - min-height (largura mínima)
  - max-height (largura máxima)
- Estas propriedades são úteis quando trabalhamos com medidas percentuais





# FLOAT

Prof. André Caruso

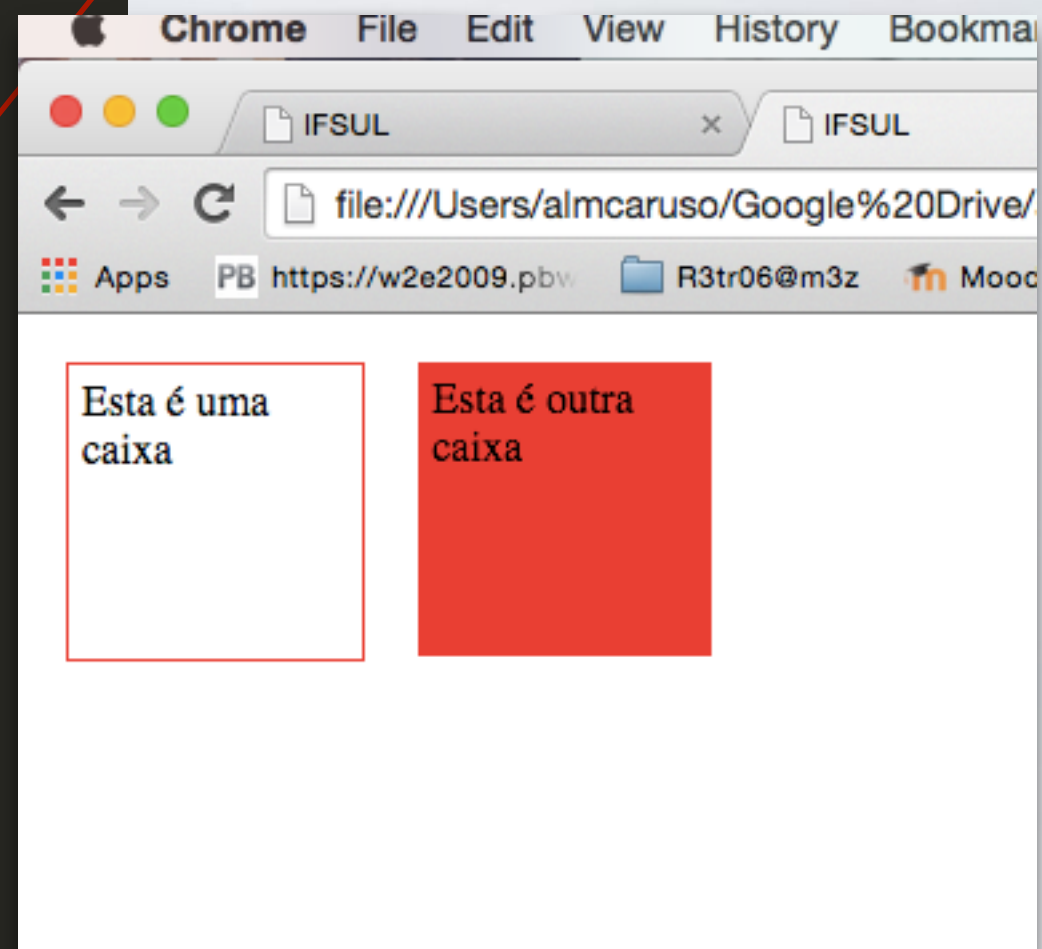


# FLUTUAÇÃO DOS ELEMENTOS

- A propriedade **float** serve para dizer como um elemento deve comportar-se em relação a seus vizinhos no HTML;
- Basicamente cada elemento pode flutuar à esquerda (**left**) ou à direita (**right**) em relação aos outros;
- É possível também definir a flutuação como **none** (nenhuma)

# FLOAT

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>IFSUL</title>
5 <meta charset="utf-8" />
6 <style>
7     #dv01{
8         border: 1px red solid;
9         padding: 5px;
10        margin: 10px;
11        width: 100px;
12        height: 100px;
13        float: left;
14    }
15    #dv02{
16        padding: 5px;
17        margin: 10px;
18        width: 100px;
19        height: 100px;
20        background-color: red;
21        float: left;
22    }
23
24 </style>
25 </head>
26 <body>
27     <div id="dv01">Esta é uma caixa</div>
28     <div id="dv02">Esta é outra caixa</div>
29
30 </body>
31 </html>
```



# FLUTUAÇÃO DOS ELEMENTOS

- A propriedade **clear** serve para dizer que o próximo elemento não flutuará em relação ao atual.
- Pense na propriedade **clear** como uma quebra de linha para as divs

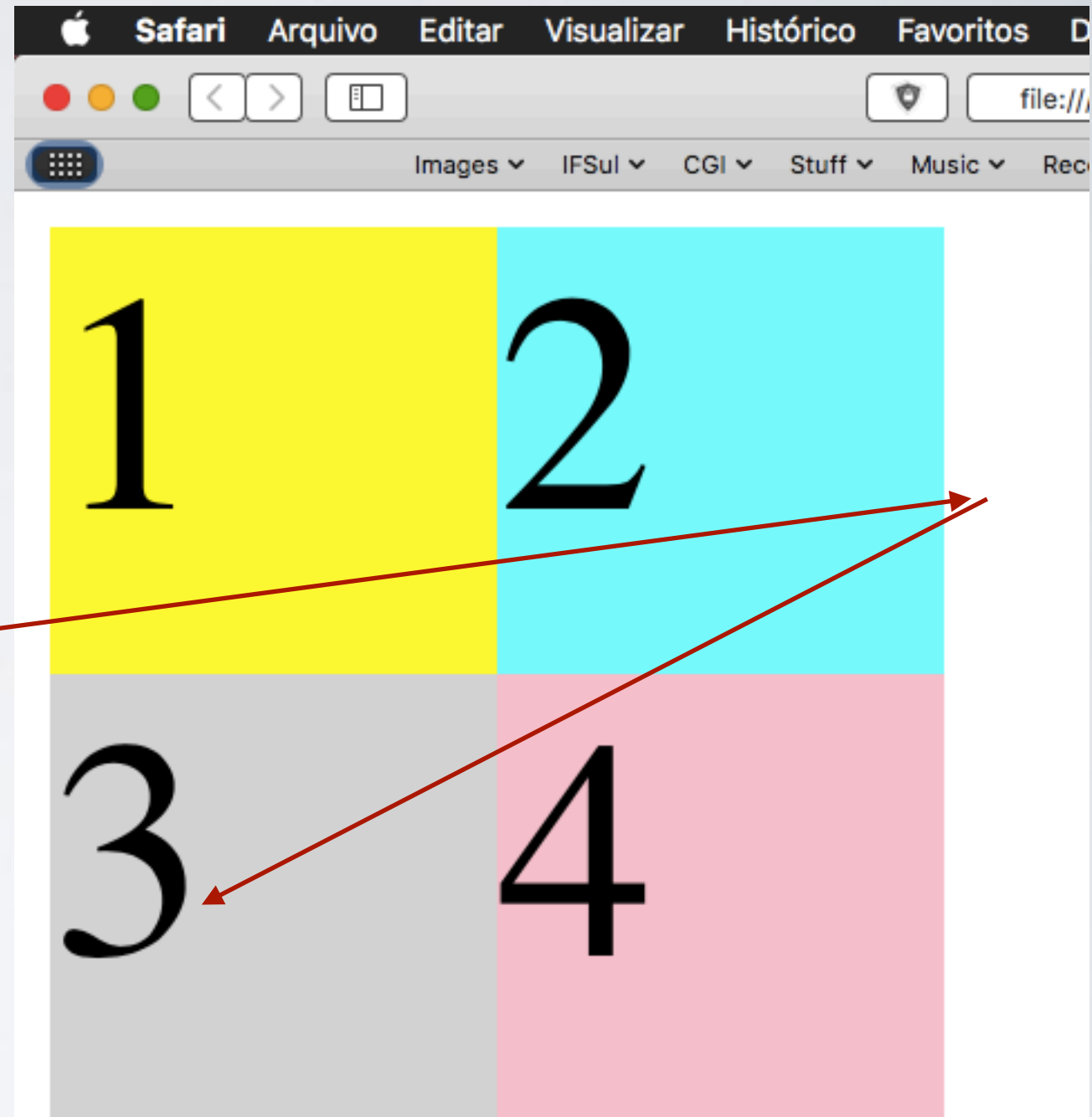


```

6 <style>
7   body { font-size:70px;}
8
9   #dv01{
10    width:100px;
11    height: 100px;
12    float:left;
13    background-color:yellow;
14  }
15  #dv02{
16    width:100px;
17    height: 100px;
18    background-color: cyan;
19    float:left;
20  }
21  #dv03{
22    width:100px;
23    height: 100px;
24    background-color: lightgray;
25    float:left;
26    clear:both;
27  }
28  #dv04{
29    width: 100px;
30    height: 100px;
31    background-color: pink;
32    float:left;
33  }
34 }
35 </style>
36 </head>
37 <body>
38   <div id="dv01">1</div>
39   <div id="dv02">2</div>
40   <div id="dv03">3</div>
41   <div id="dv04">4</div>

```

CLEAR



# EXERCÍCIO

- Crie um layout com 3 colunas, cabeçalho e rodapé
- Requisitos
  - Imagem no cabeçalho (logo)
  - Menu com links na coluna da esquerda
  - Banner de propaganda e caixa de busca na coluna da direita

