



# CSS

*Propriedades:*    *display*  
                          *visibility*  
                          *opacity*  
                          *position*

Tecnologias Web  
2020/2021

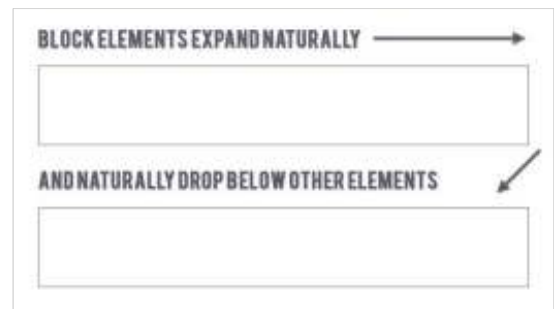
*Propriedade **display***

[exercício 1;2]

## Block-Level Element

3

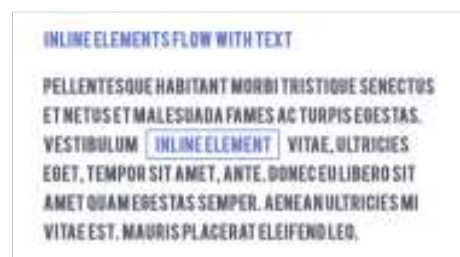
- Expande-se naturalmente para ocupar o seu *container*
  - Este comportamento é alterado se for especificada a largura do elemento (*width*)
  - Se não for especificada a altura (*height*) o elemento expande-se para englobar os seus elementos filho
- Permite a especificação das suas *margins* e/ou *padding*
- Provoca uma quebra de linha.
- A sua posição original é definida pelo fluxo do HTML
- Exemplos:
  - `<p>`, `<h1>`, `<div>`, `<form>`, `<ul>`, `<li>`



## Inline-Level Elements

4

- Aplica-se a conteúdo textual
  - Pode ser percepcionado como uma caixa que se comporta como texto.
  - Não se expande à totalidade do espaço disponível (container/viewport)
- Ignora as propriedades largura e altura (*width/height*)
- Ignora margens topo/fundo (*top/bottom*)
- Permite a definição de *left/right margin* assim como a propriedade *padding*.
  - Exemplos: `<a>`, `<span>`, `<strong>`, `<em>`, ...



## display

5

- A propriedade *display* especifica a forma como é mostrado o elemento, podendo assumir alguns valores (**apenas os mais comuns**):
  - **inline**
    - Valor por defeito: mostra o elemento em linha com o anterior.
  - **block**
    - Apresenta os elementos como um bloco (ex: <p>)
  - **inline-block**
    - Permite que os elementos sejam mostrados *inline*
    - Mantém as propriedades do bloco (width e height)
  - **none**
    - O elemento não é mostrado
    - Não tem qualquer efeito no layout
    - Pode ser aplicado a todo o tipo de elementos e não apenas a imagens

## display

6

```
.linha li {display:inline;
background-color:lightcoral;
color:white;
margin-left:2px;}
```

```
<ul class="linha">
  <li>Um</li>
  <li>Dois</li>
  <li>Três</li>
  <li>Quatro</li>
  <li>Cinco</li>
</ul>
```

Um Dois Três Quatro Cinco

## *display*

7

```
a { display:block;
    height:20px;
    width:70px;
    background-color:lightcoral;
    color:white;
    margin:2px;
    text-align:center;
    text-decoration: none;}
```

Um  
Dois  
Três  
Quatro  
Cinco

```
<body>
<a href="#">Um</a><a href="#">Dois</a><a href="#">Três</a><a href="#">Quatro</a><a href="#">Cinco</a>
</body>
```

Propriedade **visibility**

- A propriedade *visibility* especifica se um elemento é visível ou não
  - Mesmo os elementos invisíveis ocupam espaço
    - **visible**: valor por defeito. O elemento é visível
    - **hidden**: o elemento é invisível, no entanto continua a condicionar o *layout* (espaço reservado)

```
h1.visible {  
  visibility: visible  
}  
  
h1.hidden {  
  visibility: hidden  
}
```

```
<h1 class="visible">Título visível</h1>  
<h1 class="hidden">Título invisível</h1>  
<p>Reparar que o título invisível ocupa  
espaço.</p>
```

**Título visível**







Reparar que o título invisível ocupa espaço.

## Propriedade **opacity**

### *opacity*

12

- A propriedade *opacity* define o nível de opacidade para um elemento
- O nível de opacidade varia entre 0 e 1, onde
  - 1 é completamente opaco
  - 0 é completamente transparente

Property						
opacity	4.0	12.0	9.0	2.0	3.1	9.0

```
<style>
  p{background-color: orange;}
  p:first-child{opacity:1}
  p:nth-child(2){opacity:0.3}
</style>
</head>

<body>
  <p>Totalmente Opaco</p>
  <p>Transparência</p>
</body>
```

Totalmente Opaco

Transparência

## Propriedade **position**

[exercício 3]

## Posicionamento

15

- A propriedade ***position*** é utilizada para definir o posicionamento de um determinado elemento.
- Existem quatro tipos diferentes de posicionamento:
  - *static*
    - Caso a propriedade *position* não seja especificada, o posicionamento é *static* (valor por *default*).
  - *absolute*
  - *relative*
  - *fixed*

## Posicionamento Estático (*static*)

16

- Os elementos são apresentados segundo o fluxo normal do HTML
  - A ordem pela qual os elementos são definidos no código HTML.
  - Não provoca nenhuma alteração ao posicionamento dos elementos
  - As propriedades de posicionamento, top, left, right e bottom são ignoradas.

```
seletor {position: static;}
```



## Posicionamento Estático (*static*)

17

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {width:50px;height:50px}
    #box1 {background-color:lightgray;position:static}
    #box2 {background-color:orange;position:static}
    #box3 {background-color:darkgray;position:static}
  </style>
</head>
<body>
  <div id="box1"></div>
  <div id="box2"></div>
  <div id="box3"></div>
</body>
</html>
```



Fluxo normal: **static** não provoca nenhuma alteração ao posicionamento dos elementos

## Posicionamento Relativo (*relative*)

18

- Um elemento com *position relative*, posiciona-se em relação à sua **posição original** no fluxo HTML:
  - Admite as propriedades de posicionamento: *top; bottom; left; right*;
    - Estas propriedades permitem definir o deslocamento em relação à sua posição original, definida pelo fluxo normal do HTML.
      - **top**: deslocamento para baixo
      - **bottom**: deslocamento para cima
      - **left**: deslocamento para a direita
      - **right**: deslocamento para a esquerda

seletor {position: relative;}

## Posicionamento Relativo (*relative*)

19

- Os elementos com posicionamento relativo conservam o seu espaço original, assim:
  - Afetam o fluxo do HTML,
  - Os elementos colocados depois das camadas *relative*, terão em conta as dimensões dos elementos anteriores
  - Os valores de posicionamento (top, left, ...) não afectam os elementos seguintes.

## Posicionamento Relativo (*relative*)

20

```
<!DOCTYPE html>
<html>
<head>
  <title></title>
  <style>

    div.d1 {width:50px;height:50px}
    #box1 {background-color:lightgray;position:static}
    #box2 {background-color:orange;
           position:relative;
           top:10px;
           left:50px;}
    #box3 {background-color:darkgray;position:static}
  </style>
</head>
<body>
  <div class= "d1" id="box1"></div>
  <div class= "d1" id="box2"></div>
  <div class= "d1" id="box3"></div>
</body>
</html>
```



## Centrar um site com CSS (exercício 3; Folha 5)

21

- **margin:0 auto**

- Margens superior e inferior serão “0”
- Margens esquerda e direita são calculadas automaticamente, ou seja, o *container* fica centrado

```
<body>
  <div id="page-wrap">
    <!-- conteudo aqui -->
  </div>
</body>
```

```
# page-wrap {
  width:750px;
  margin:0 auto; }
```

## CSS (Exercício 3; Folha 5)

22

- **Imagem de Fundo**

- *seletor*{ **background-image:** url("file")}

# Propriedade **position**

[exercício 4]

## *Posicionamento Absoluto (absolute)*

24

- Elementos com posicionamento **absolute**:
  - **não são** afetados pelo lugar onde surgem no código HTML
  - **não afetam** outros elementos que façam parte do fluxo normal do HTML.
- O posicionamento absoluto, considera os deslocamentos relativamente a um ponto de referência:
  - *top, left, bottom e right*

*seletor {position: absolute;}*

## Posicionamento Absoluto (*absolute*)

25

- Um elemento com posicionamento absoluto posiciona-se em relação a um *container*
  - Assim, os valores *top*, *left*, *bottom* e *right* referem-se à distância referente ao **primeiro *container* que tenha um valor de *position* diferente de *static*:**
    - No limite o *container* para referência de deslocamento é o elemento `<body>`
    - diferente do posicionamento relativo o qual considera o posicionamento em relação à sua posição original
- Um elemento posicionado de forma absoluta **não cria** qualquer espaço no documento e **não deixa** nenhum espaço vazio após ser posicionado.

## Posicionamento Absoluto (*absolute*)

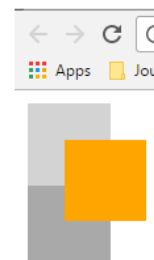
26

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title></title>
  <style>
    div.d1 {width:50px;height:50px}
    #box1 {background-color:lightgray;position:static}
    #box2 {background-color:orange;
           position:absolute;
           top:0px;
           left:0px;}
    #box3 {background-color:darkgray;position:static}
  </style>
</head>
<body>
  <div class= "d1" id="box1"></div>
  <div class= "d1" id="box2"></div>
  <div class= "d1" id="box3"></div>
</body>
</html>
```



A referência de posicionamento é o `<body>` (canto superior esquerdo da janela do browser)

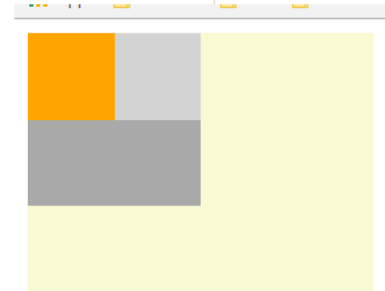
`top:30px;`  
`left:30px;`



## Posicionamento Absoluto (*absolute*)

27

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title></title>
  <style>
    #global {width:200px; height:150px; background-color:#FAFAD2; position:relative;}
    #box1 {background-color:lightgray;width:100px;height:50px;}
    #box2 {background-color:orange; width:50px;height:50px;
           position:absolute;
           top:0px;
           left:0px;}
    #box3 {background-color:darkgray; width:100px;height:50px;}
  </style>
</head>
<body>
  <div id="global">
    <div class="d1" id="box1"></div>
    <div class="d1" id="box2"></div>
    <div class="d1" id="box3"></div>
  </div>
</body>
</html>
```



A referência de posicionamento é o canto superior esquerdo do container com *position* diferente de *static*

## Sobreposição de Elementos

28

- Os elementos com *position absolute* são retirados do fluxo normal do documento HTML, podendo ficar sobrepostos a outros elementos:
  - A propriedade *z-index*, que pode ter valores positivos ou negativos, define a ordem dos elementos sobrepostos
  - O elemento com o valor mais alto de *z-index* irá surgir no topo

```
<style>
  #box1 {background-color:lightgray;width:100px;height:50px;
         position:absolute;
         top:0px;
         left:0px;
         z-index:2;}
  #box2 {background-color:orange; width:50px;height:100px;
         position:absolute;
         top:0px;
         left:0px;
         z-index:1;}
</style>
```

```
<body>
  <div id="box1"></div>
  <div id="box2"></div>
</body>
```



## Centrar um site com CSS (*position:absolute*)

29

```
#page-wrap{  
  position:absolute;  
  left:50%;  
  margin-left:-375px;  
  width:750px;  
}
```

