

Técnico em desenvolvimento de sistemas
SENAI SUL

FIESC SENAI

sc.senai.br

Programação web

Professor Rogério dos Santos

email:rogerio.santos@edu.sc.senai.br

Pós graduado com especialização em linguagem java

INTRODUÇÃO AO CSS

- ✓ Linguagem de folha estilo utilizada para apresentação de documentos em uma linguagem de marcação.
- ✓ Tem o objetivo de formatar visualmente documentos escritos.

INTRODUÇÃO AO CSS

- ✓ O desenvolvedor pode criar um link para uma página que contem os estilos precedendo de forma identica para toda a página;
- ✓ Uma folha de estilos contém uma lista de regras.

INTRODUÇÃO AO CSS

Por que usar CSS ?

- ✓ Alguns sites começaram a ser compostos por diversas páginas.
- ✓ Sites grandes são de difícil manutenção.

INTRODUÇÃO AO CSS

Vantagens de usar o CSS

- ✓ Centralizar;
- ✓ Padronizar;

INTRODUÇÃO AO CSS

- ✓ Css é apenas uma das tecnologias que compreendem a web, ou seja, ela por si só não é capaz de gerar resultados.
- ✓ Precisa do html para fazer a formatação de apresentação.
- ✓ Também é necessário Javascript e linguagens derivadas para outros fins como execução da lógica pelo lado do cliente.

INTRODUÇÃO AO CSS

- ✓ Css é uma linguagem do tipo **client side**.
- ✓ Pode ser escrito no próprio elemento, mas não é o melhor procedimento pois descentraliza páginas visuais, exceto se a página conter somente um arquivo.

INTRODUÇÃO AO CSS

Exemplo com CSS embutido

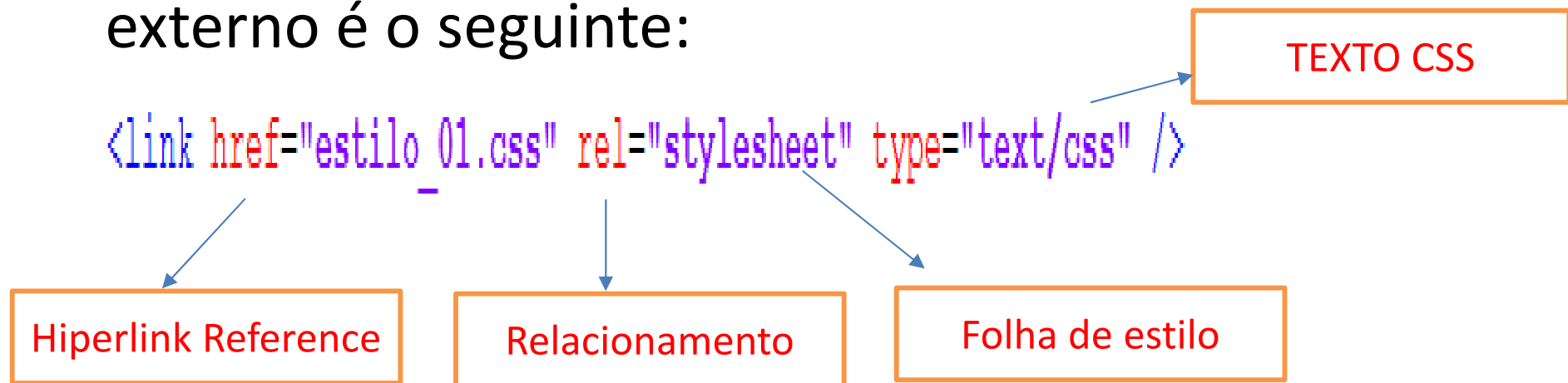
```
<style type="text/css">
    /* Instruções no corpo do próprio documento */

    body {
        font-size:20px;
    }
</style>
```

INTRODUÇÃO AO CSS

Uso de CSS em arquivo externo

- ✓ Pode-se usar arquivos externos para escrever CSS, esse é o procedimento correto e portanto mais recomendado. O exemplo para carregar o arquivo externo é o seguinte:



INTRODUÇÃO AO CSS

Exemplo de CSS em arquivo externo



```
body {  
    background:yellow;  
}
```

INTRODUÇÃO AO CSS

SELETORES

- ✓ Nada mais é do que as regras de aplicação de nossas instruções CSS.
- ✓ Como o nome diz são instruções que nos permitem selecionar objetos.

INTRODUÇÃO AO CSS

TIPOS DE SELETORES

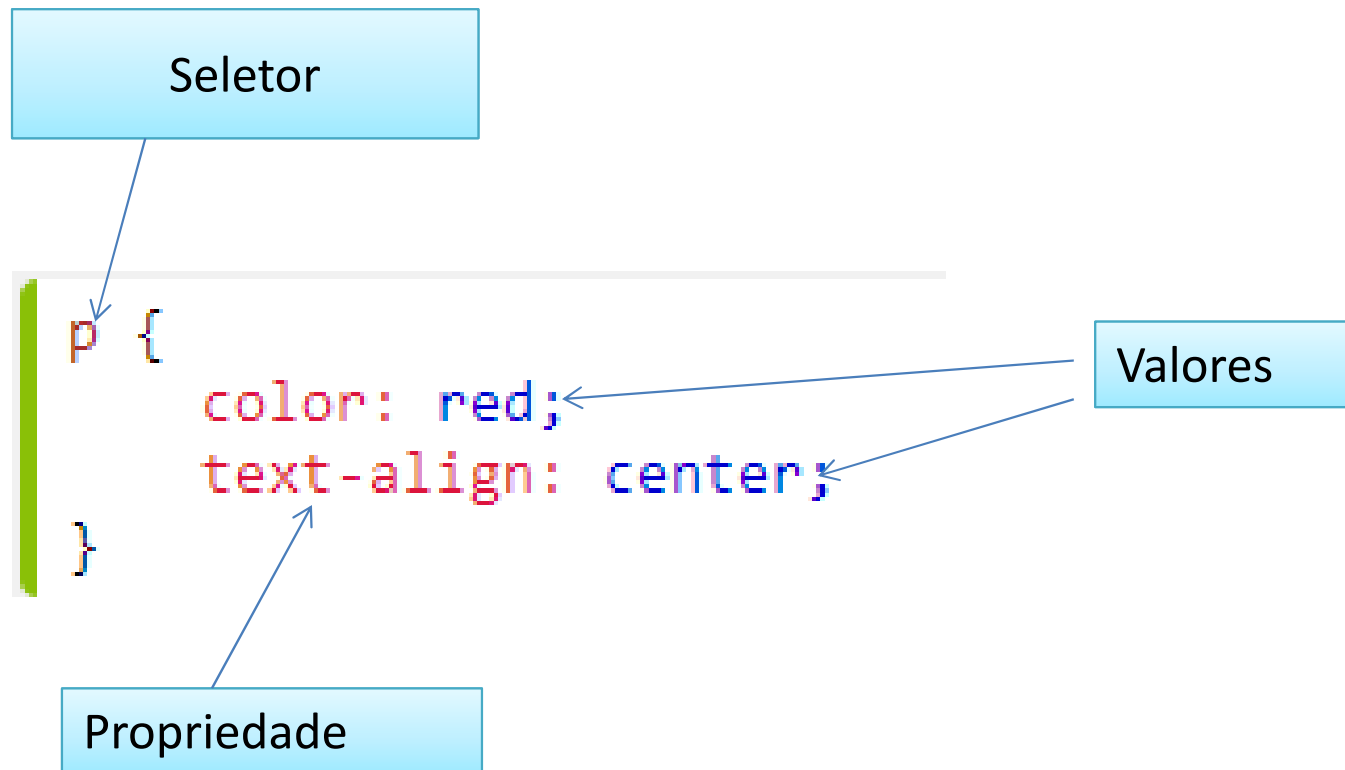
A seleção pode ser feita por 3 tipos:

- ✓ Seleção por tipo de elemento.
- ✓ Seleção por ID.
- ✓ Seleção por classe.

INTRODUÇÃO AO CSS

- A seleção por tipo de elemento nos possibilita aplicar uma sequência de regras a um determinado elemento HTML como por exemplo, todas as tags P do nosso elemento.

EXEMPLO COM CSS USANDO O SELETOR POR ELEMENTO



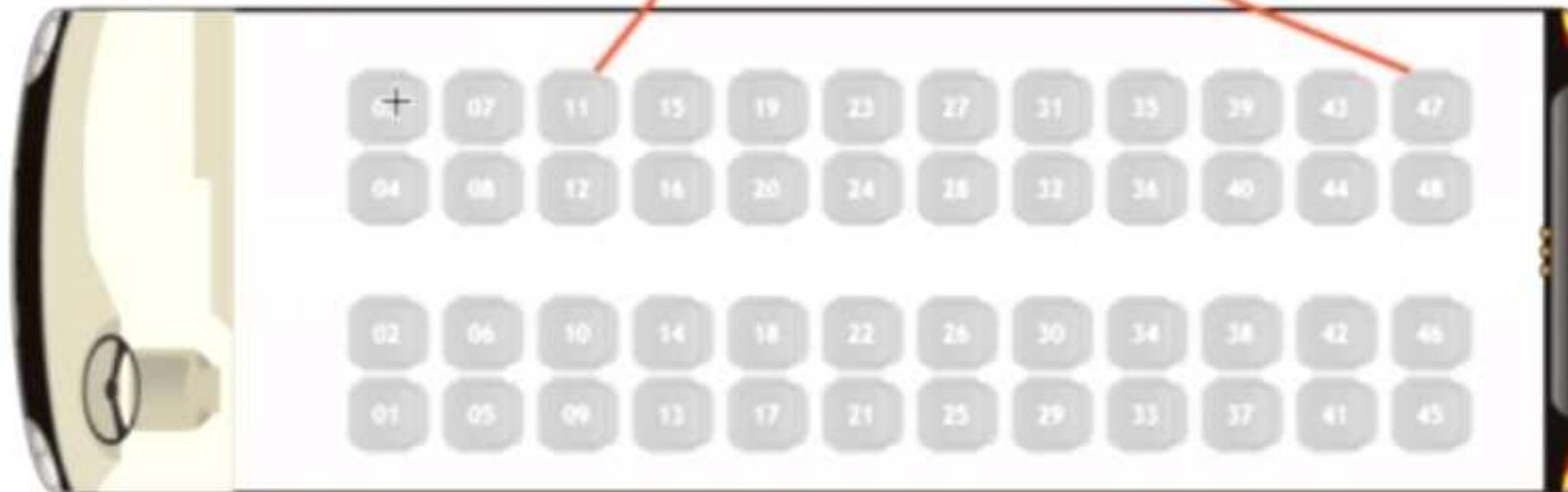
SELETORES POR ID

- A seleção por ID nos possibilita selecionar os objetos que possuam um mesmo identificador, esta é um atributo HTML disponível em todos os elementos.

SELETORES POR ID

Chamar passageiro pelo ID

id



SELETORES POR ID

- Em um ônibus cada passageiro está sentado em sua poltrona onde cada poltrona tem a numeração.

Ex: Um passageiro que está sentado na poltrona 11, é somente aquele passageiro com as suas características, ele é único

- Em uma página cada objeto identificado por id significa somente aquele elemento.

Ex: Quando é definido um cabeçalho, rodapé, somente esses serão únicos na página

SELETORES POR ID

Portando todo elemento em HTML que for único deverá ser estilizado por IDs



SELETORES POR CLASSE



SELETORES POR CLASSE

- Em um ônibus temos as classificações de masculino e feminino
Ex: Existem vários passageiros do sexo masculino como passageiros do sexo feminino.
- Em uma página cada objeto identificado por classe significa que pode ser repetido em várias partes da página
Ex: Quando é definido colunas em uma página, pode ter duas três ou mais colunas, então essas colunas são estilizadas por classe.

SELETORES COM SELETOR POR ID

```
<div id="cabecalho">  
    
  <ul>  
    <li>Home</li>  
    <li>Quem somos</li>  
    <li>Contato</li>  
  </ul>  
</div>
```

Neste exemplo a div com identificado ID referenciando o cabeçalho, sendo o único

```
#cabecalho {  
  background: green;  
}
```

SELETORES COM SELETOR POR CLASSE

Código html com tag P referenciando o IDENTIFICADOR PARÁGRAFO sendo ID

```
<p align = "Center" class="paragrafo02">  
Linha 01 Linha 02 Linha 03  
</p>
```

Código CSS com classe paragrafo02

```
.paragrafo02 {  
    color:red;  
}
```

A tag p sendo estilizada por classe, pois pode ter vários parágrafos em páginas HTML.

DIV & SPAN

- As **DIVs** e **SPANs** servem para estruturar melhor os sites.
- Praticamente todos os sites são estruturados por **DIVs** e muitas formatações são compostas por **SPANs**.
- As **DIVs** fazem a divisão do site em partes e também definem containers visuais para o site.
- A tag **SPAN** serve para fazer formatação de texto.

DIV & SPAN

```
<div id="topo">  
      
    <ul>  
        <li>Home</li>  
        <li>Produtos</li>  
        <li>Contato</li>  
    </ul>  
</div>
```

Acima o código com estrutura de DIV Para definir o top da página e a logo

DIV & SPAN

```
<div id="conteudo">  
  <div class="produto">  
    Produto 1  
  </div>  
  
  <div class="produto">  
    Produto 2  
  </div>  
  
  <div class="produto">  
    Produto 3  
  </div>  
</div>
```

A div com referencia de **conteúdo** através do ID

Existem três **divs** com a classe produto

DIV & SPAN

DIV para fazer a divisão do rodapé.

```
<div id="rodape">  
  Todos os <span class="azul"> direitos</span> reservados  
</div>
```

span serve para formatar texto.

BACKGROUND

- ✓ Praticamente todos os elementos possuem esta propriedade e a mesma nos permite definir como está o fundo de um determinado elemento.
- ✓ Pode-se utilizar cores, imagens e ainda cores mais elaboradas, tais como gradientes e degradês.

CORES DE FUNDO COM BACKGROUND

- ✓ As cores em css são definidas utilizando um esquema denominado RGB(**RED** **GREEN** **BLUE**) que são as cores primárias.
- ✓ Muitas cores podem ser criadas a partir destas três cores básicas.
- ✓ Podemos definir as cores usando a notação hexadecimal. Ex:
#000000 = preto
#ffffff = branco

CONVERSÃO DE HEXADECIMAL PARA RGB

<https://www.rgbtohex.net/hextorgb/>

Neste site é possível fazer conversão de hexadecimal para RGB

TONALIDADE DE CORES

Para ver exemplos com tonalidade de cores.

[https://www.w3schools.com/css/css_colors.
asp](https://www.w3schools.com/css/css_colors.asp)

CORES DE FUNDO COM BACKGROUND


Para ver exemplos com cores em hexadecimal visite o site:

www.w3schools.com/html/html_colors.asp
<https://html-color-codes.info/>

CORES DE FUNDO COM BACKGROUND

Exemplo usando background-color com hexadecimal

```
body {  
    font-size: 20px;  
    background-color : #000000;  
}
```

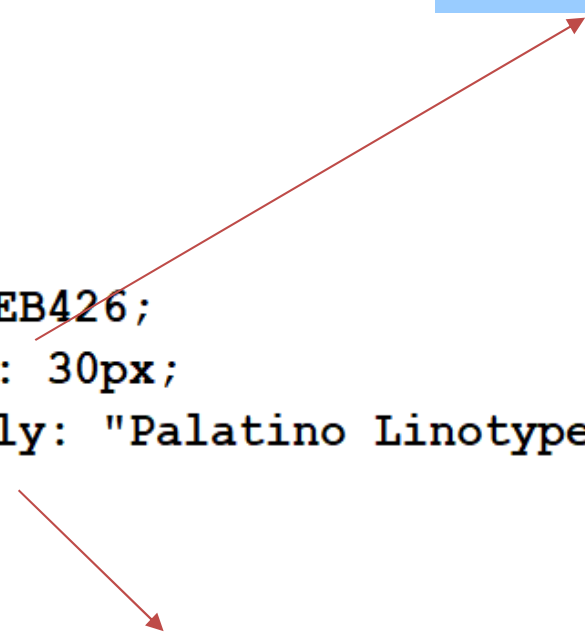


FONTS

Existem várias propriedades para FONTS

Font-size – Permite um tamanho fixo

```
.formatacao{  
  color: #0EB426;  
  font-size: 30px;  
  font-family: "Palatino Linotype", "Book Antiqua", Palatino, serif;  
}
```



Font-family– Permite definir uma família de fonts

Pode ser visto também em https://www.w3schools.com/css/css_font.asp

IMAGENS DE FUNDO COM BACKGROUND

Para utilização de imagens de fundo temos a seguinte sintaxe

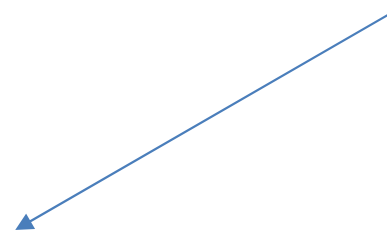
Background: url("Imagem") [Vertical horizontal repetição]

Exemplo para mostrar a imagem de fundo

```
.fundo_pagina {  
    background:url("htmlcss.png") ;  
}
```

IMAGENS DE FUNDO COM BACKGROUND

```
background-attachment: fixed;
```



Permite definir fixo uma imagem na tela de fundo, caso houver.

```
background-attachment: scroll;
```



Permite fazer um rolamento junto com tela.

IMAGENS DE FUNDO COM BACKGROUND

Propriedade Background-repeat

- ✓ Por padrão a imagem repete a página inteira.
- ✓ Mas se quiser que imagem se repita por todo o fundo da página, deve-se usar a propriedade **background-repeat**.

IMAGENS DE FUNDO COM BACKGROUND

Valor	Propósito
repeat	Faz com que a imagem se repita por toda a página.
repeat-x	A imagem será repetida somente na horizontal.
repeat-y	A imagem será repetida somente na vertical.
no-repeat	A imagem será exibida somente uma vez.

IMAGENS DE FUNDO COM BACKGROUND

Exemplo de imagem com repeat

```
.fundo_pagina {  
    background:url("htmlcss.png") repeat  
}
```

Exemplo de imagem com repeat-x

```
.fundo_pagina {  
    background:url("htmlcss.png") repeat-x  
}
```

IMAGENS DE FUNDO COM BACKGROUND

Exemplo de imagem com repeat-y

```
.fundo_pagina {  
    background:url("htmlcss.png") repeat-y  
}
```

Exemplo de imagem com no-repeat

```
.fundo_pagina {  
    background:url("htmlcss.png") no-repeat  
}
```


POSICIONAMENTO DE IMAGEM DE FUNDO

Pode-se definir uma posição para imagem de fundo.

Segue a tabela abaixo os valores:

Valor	Significado
X% Y%	Posicionamento nos eixos X(Horizontal) e Y(Vertical)
Left	É mostrado à esquerda da página que contiver o elemento.
Right	Mostrado à direita da página que contiver o elemento.
Top	Mostrado no topo da página que contiver o elemento.
Center	Mostrado no centro da página que contiver o elemento
Bottom	Mostrado na parte inferior da página que contiver o elemento

IMAGENS DE FUNDO COM BACKGROUND

Exemplos de posição de imagem de fundo com 50 % na vertical e 70 % na horizontal.

```
.fundo_pagina {  
    background:url("htmlcss.png") 50% 70% no-repeat  
}
```

Posicionamento de 50 %
do valor x (Horizontal)

Posicionamento de 70 % do valor y
(Vertical)

IMAGENS DE FUNDO COM BACKGROUND


Em relação ao código anterior, também pode ser resolvido dessa forma.

```
.fundo_pagina {  
    background:url("htmlcss.png") ;  
    background-repeat :no-repeat;  
    background-position: 50% 70%;  
}
```

IMAGENS DE FUNDO COM BACKGROUND

Exemplos de posição de imagem de fundo na posição esquerda e na parte superior.

```
.fundo_madeira {  
    background:url("htmlcss.png") left top no-repeat;  
}
```



Posicionamento na parte
esquerda na horizontal

Posicionamento na parte top na
vertical

TEXTOS

✓ Para formatação de textos é usado o elemento **TEXT**.

✓ As propriedades relacionadas ao elemento TEXT têm as seguintes operações:

Espaço entre letras, espaço entre palavras, recuo de texto, decoração de texto, alinhamento, maiúsculas e minúsculas.

PROPRIEDADE LETTER-SPACING

✓ Permite mostrar os caracteres separados.

Exemplo:

Em HTML

```
<p class="palavra_espacamento">Aprendendo CSS</p>
```

Em css

```
.palavra_espacamento {  
    letter-spacing: 80px;  
}
```

PROPRIEDADE WORD-SPACING

✓ Permite mostrar o espaço entre as palavras.

Exemplo:

Em HTML

```
<p class="palavra_wordSpacing">Aprendendo CSS</p>
```

Em css

```
.palavra_wordSpacing {  
    word-spacing: 90px;  
}
```

PROPRIEDADE TEXT-ALIGN

✓ A propriedade **text-align** tem quatro tipos de alinhamento: **left**, **right**, **center** e **justify**

Em HTML

```
<p class="palavra_textAlign">Aprendendo CSS</p>
```

Em css

```
.palavra_textAlign {  
    /* left, center, right, justify */  
    text-align:center;  
}
```


PROPRIEDADE TEXTDECORATION

✓ A propriedade **text-decoration** tem cinco valores: none, underline, overline, line-through, blink

Em HTML

```
<p class="palavra_textDecoration">Aprendendo CSS</p>
```

Em css

```
.palavra_textDecoration {  
    /* none, underline, overline, line-through, blink */  
    text-decoration:underline;  
}
```

Nesse exemplo irá mostrar a palavra sublinhada

PROPRIEDADE TEXTINDENT

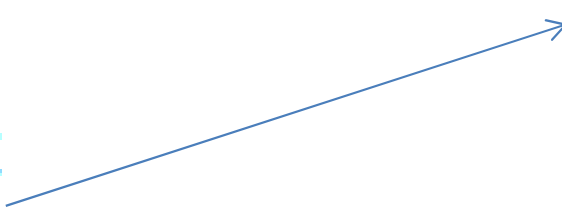
✓ A propriedade **text-indent** permite fazer indentação conforme a porcentagem.

Em HTML

```
<p class="palavra_textIndent">Aprendendo CSS</p>
```

Em css

```
.palavra_textIndent {  
    text-indent: 10%;  
}
```



Permite fazer o recuo de 10 % da esquerda para a direita.

PROPRIEDADE TEXTTRANSFORM


✓ A propriedade **text-transform** possui 4 valores: **none, capitalize, uppercase, lowercase**.

Em HTML

```
<p class="palavra textTransform">Aprendendo CSS</p>
```

Em css

```
.palavra_textTransform {  
    /* none, capitalize, uppercase, lowercase */  
    text-transform: capitalize;  
}
```



Capitalize permite colocar a primeira letra em maiúscula de cada palavra

TAMANHO DE TEXTOS

`px` -> tamanho fixo

`%` -> tamanho relativo

`em` -> tamanho relativo ao container pai

16 px – É o
tamanho padrão
para a maioria
dos browsers



Isso quer dizer que se colocarmos um parágrafo sem definir um tamanho, por padrão ele fica com 16px

TAMANHO DE TEXTOS

```
<div id="divisao">
  <p class="texto1">Texto 1 </p>
  <p class="texto2">Texto 2 </p>
  <p class="texto3">Texto 3</p>
</div>
```

No exemplo citado a tag **div** é o container pai das tags **parágrafos**. Isso quer dizer que quando as tags filhas são estilizadas com **em** o tamanho será relativo ao container pai.

No exemplo a tag que tem a classe **texto3** está estilizada com **px**, então o tamanho fica fixo na tag filha.

```
#divisao
{
  font-size: 60px;
}

.texto1{
  font-size: 1em;
}

.texto2 {
  font-size: 2em;
}

.texto3{
  font-size: 20px;
}
```

TAMANHO DE TEXTOS

```
.texto1{  
    font-size:100%;  
}
```

Se for estilizado
100% em uma
classe ele assume

Isso quer dizer que se colocarmos um parágrafo sem definir um tamanho, por padrão ele fica com 16px



TAMANHO DE TEXTOS

```
.texto2 {  
  font-size: 1em;  
}
```

Equivale a 16 px

```
.texto3{  
  font-size: 32px;  
}
```

Equivale a 2em

```
.texto4{  
  font-size: 64px;  
}
```

Equivale a 4em

TAMANHO DE TEXTOS

No exemplo abaixo, se for omitido a propriedade para a classe **texto2**, o tamanho ele assume o correspondente a tag pai que no caso é a **body** onde está definindo 100% da página que no caso é **16px** ou **1em**.

```
body {font: 100% verdana, arial, tahoma, sans-serif;}

.texto1{
    font-size: 1em;
}

.texto2 {
    font-size: 2em;
}

.texto3{
    font-size: 3em;
}
```

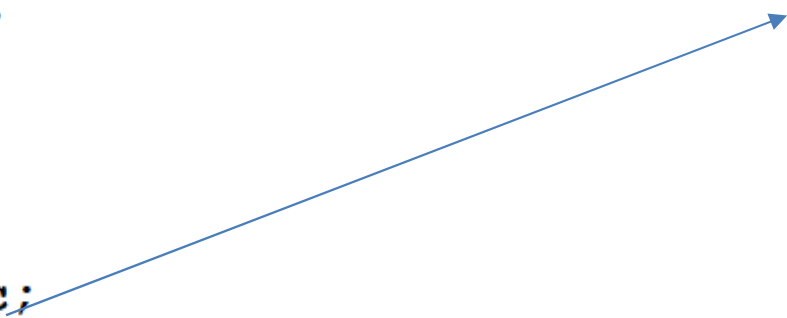

ESTILO DE TEXTO

```
body {  
    font-size: 40px;  
    font-family: "Times New Roman", Times, serif;  
}
```

```
.negrito {  
    /* normal, bold, 100 até 900 */  
    font-weight: bold;  
}
```

```
.italico {  
    font-style: italic;  
}
```

Permite definir o efeito itálico.



bold → **Negrito** **normal** → Permite deixar normal

Font-weight 100,200, 300... define a intensidade do negrito

ESTILO DE TEXTO

```
body {  
    font-size: 40px;  
    font-family: "Times New Roman", Times, serif;  
}
```

Font-family



Permite definir várias fontes, uma família de fontes.

```
.texto{  
    font: bold 40px "Lucida Sans Unicode", "Lucida Grande",  
    sans-serif ;  
}
```

Quando definimos **font** podemos definir uma família de fonts

ELEMENTO BORDER

- ✓ Como o próprio nome diz, é uma propriedade responsável por criar bordas em nossos elementos.
- ✓ Muito utilizado para delimitar um espaço ou uma caixa de textos ou ainda ressaltar algum elemento.

ELEMENTO BORDER.

✓ A propriedade **BORDER** possui os seguintes atributos:

-**Border Style, border-width, border-color.**

✓ E pode ser definida por partes:

-**Border-left, border-top, border-right e border-bottom**

ELEMENTO BORDER.

Resultado da implementação do borda.

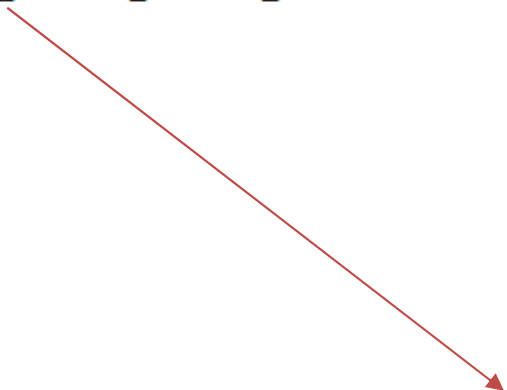


As bordas são aplicadas no sentido horário.

EXEMPLO COM BORDER-COLOR E BORDER-WIDTH

```
#conteudo {  
  
    border-color: red green yellow blue;  
    border-width: 15px 10px 8px 3px;  
    border-style: solid;  
}
```

```
<div id="conteudo">  
    Bordas<br>  
    Bordas<br>  
    Bordas<br>  
</div>
```



Os valores que estão em border-color e border-width vão da esquerda para a direita.

ELEMENTO BORDER.

✓ O atributo border-style pode possuir os seguintes valores:

None,dotted,dashed, solid,double, groove, ridge, inset e outset.

EXEMPLO COM ELEMENTO BORDER

Exemplo em html

```
<p class="borda_border01">Borda de 1 pixel solida de cor black</p>
```

Exemplo em css

```
.borda_border01 {  
    border:1px solid black;  
}
```


EXEMPLO COM ELEMENTO BORDER E SEUS ATRIBUTOS.

Exemplo em html

```
<p class="borda_border02">Borda vermelha, dotted, 3px</p>
```

Exemplo em css

```
.borda_border02 {  
    border-style:dotted;  
    border-width:3px;  
    border-color:red;  
}
```

EXEMPLO COM OS ATRIBUTOS BOTTOM E TOP

Exemplo em html

```
<h1 class="borda_bordaInferiorSuperior">Borda Superior e Inferior Diferenciada</h1>
```

Exemplo em css

```
.borda_bordaInferiorSuperior {  
    border-bottom: 3px dashed green;  
    border-top: 5px outset red;  
}
```

EXEMPLO COM OS ATRIBUTOS RIGHT E LEFT

Exemplo em html


```
<p class="borda_bordaEsquerdaDireita">Texto com borda na esquerda e na direita</p>
```

Exemplo em css

```
.borda_bordaEsquerdaDireita {  
    border-right:10px solid black;  
    border-left:5px inset red;  
}
```

SEM BORDA

```
.borda_semBorda {  
    border:none;  
}
```



A linha de código acima é para tirar a borda

PROPRIEDADE MARGIN

- ✓ Esta propriedade é responsável por criar espaçamento entre elementos, é criada após a borda do elemento.
- ✓ As margens não possuem cor de fundo, sendo totalmente transparentes.

PROPRIEDADE MARGIN

Esta propriedade pode ser escrita de diversas formas:

margin-top, margin-bottom, margin-right, margin-left;

```
.conteudo2{  
  margin-top: 10px;  
  width: 150px;  
  border: 1px solid red;  
}
```

→ Espaço no top da DIV

PROPRIEDADE MARGIN

```
.conteudo2{  
  margin-right: 10px; —————> Espaçamento à direita  
  width: 150px;  
  border: 1px solid red;  
}
```

```
.conteudo2{  
  margin-bottom: 10px; —————> Espaçamento abaixo  
  width: 150px;  
  border: 1px solid red;  
}
```

```
.conteudo2{  
  margin-left: 10px; —————> Espaçamento à esquerda  
  width: 150px;  
  border: 1px solid red;  
}
```

PROPRIEDADE MARGIN

Uma outra forma que pode ser resolvido configurando o espaçamento dos quatros lados.

```
.conteudo2{  
    margin-top:    20px;  
    margin-right:  15x;  
    margin-bottom:10px;  
    margin-left:   5px;  
}
```

```
.conteudo2{  
    margin:20px;  
    width: 150px;  
    border: 1px solid green;  
}
```


PROPRIEDADE PADDING

Padding

Esta propriedade é usada para criar um espaçamento interno no elemento, ou seja, é a forma invertida na margem.

Um exemplo com
padding-top

O resultado com o
uso padding-top

```
.conteudo {  
  border: 1px solid red;  
  padding-top: 20px;  
  width: 150px;  
}
```

Lorem ipsum dolor sit
amet, consectetur
adipiscing elit.

PROPRIEDADE PADDING

Um exemplo com
padding-right

```
.conteudo {  
  border: 1px solid red;  
  padding-right: 20px;  
  width: 150px;  
}
```

O resultado com o
uso padding-right

Lorem ipsum dolor sit
amet, consectetur
adipiscing elit.

PROPRIEDADE PADDING

Um exemplo com
padding-bottom

O resultado com o
uso padding-bottom

```
.conteudo {  
  border: 1px solid red;  
  padding-bottom: 20px;  
  width: 150px;  
}
```

Lorem ipsum dolor sit
amet, consectetur
adipiscing elit.

PROPRIEDADE PADDING

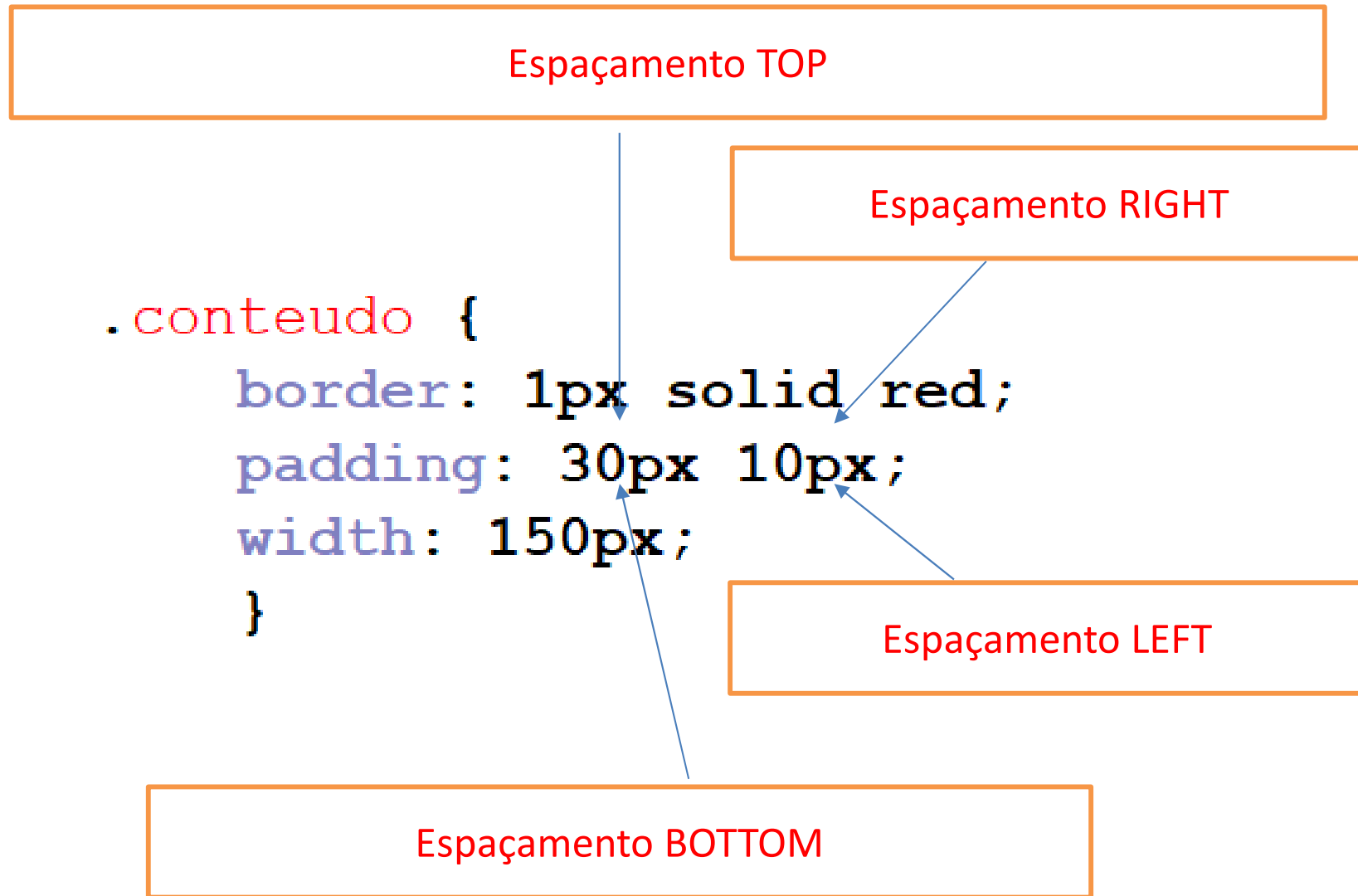
Um exemplo usando os quatros lados: top, right, bottom e left

```
.conteudo {  
  border: 1px solid red;  
  padding: 20px 15px 10px 5px;  
  width: 150px;  
}
```

```
.conteudo {  
  border: 1px solid red;  
  padding-top: 20px;  
  padding-right: 20px;  
  padding-left: 20px;  
  padding-bottom: 20px;  
  width: 150px;  
}
```

Um outro exemplo mas com mesmo propósito.

PROPRIEDADE PADDING



ELEMENTOS FLUTUANTES

Elementos flutuantes têm uma grande importância para montar layouts em páginas.

✓ float é a propriedade que permite flutuar os elementos. Recebe dois valores:

float:left

→ Permite flutuar para esquerda

float: right

→ Permite flutuar para direita

clear: right

→ Limpa o elemento de flutuação à direita

clear: left

→ Limpa o elemento de flutuação à esquerda

clear: both

→ Limpa o elemento de ambos os lados

ELEMENTOS FLUTUANTES

```
<div class="conteudo">  
  Conteúdo 1  
</div>
```

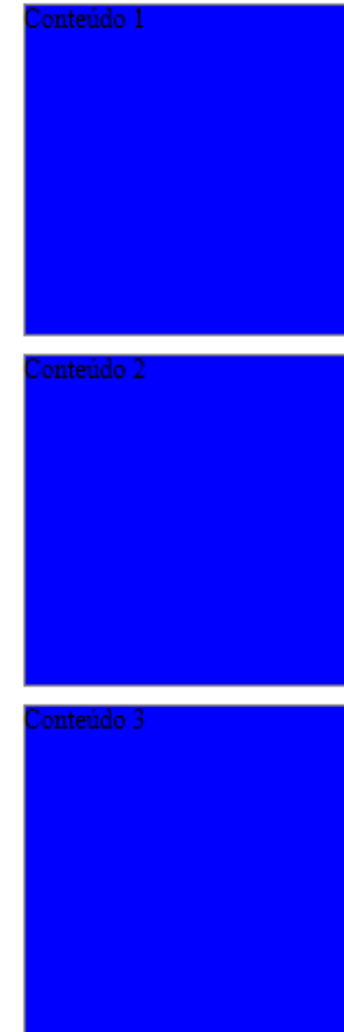
```
<div class="conteudo">  
  Conteúdo 2  
</div>
```

```
<div class="conteudo">  
  Conteúdo 3  
</div>
```

Estrutura de divs

```
.conteudo{  
  background: blue;  
  border: 1px solid gray;  
  width: 200px;  
  height: 200px;  
  margin: 10px;  
}
```

Estilizado em css



Neste exemplo temos três divs e é natural que segue o fluxo da página ou seja um abaixo do outro.

ELEMENTOS FLUTUANTES

```
.conteudo{  
  background: blue;  
  border: 1px solid gray;  
  width: 200px;  
  height: 200px;  
  margin: 10px;  
  float:right;  
}
```



Com a propriedade **float: right** os elementos da div posicionaram à direita.

ELEMENTOS FLUTUANTES



```
.conteudo{  
  background: blue;  
  border: 1px solid gray;  
  width: 200px;  
  height: 200px;  
  margin: 10px;  
  float:left; →  
}
```

Com a propriedade **float: left** os elementos da div posicionaram à esquerda.

ELEMENTOS FLUTUANTES

Para colocar um texto abaixo das divs, automaticamente irá ficar à direita ou à esquerda conforme o fluxo que está flutuando.

Conteúdo 1



Conteúdo 2



Conteúdo 3




Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur sodales sapien magna, quis congue augue consequat et. Praesent tristique blandit scelerisque. In hac habitas platea dictumst. Mauris non blandit neque. Maecenas facilisis pulvinar enim, sit amet dignissim quam. Sed sodales massa ligula, sit amet pellentesque ipsum pellentesque ege Quisque sem dolor, ornare tincidunt rutrum ac, imperdiet sed elit. Fusce pharetra magna sed ante placerat, nec faucibus dui viverra. Duis ligula dui, porttitor in neque ac, accumsan vestibulum nibh. Quisque ullamcorper lobortis sodales. Maecenas purus mauris, vulputate sed viverra sit amet, suscipit id turpis.

ELEMENTOS FLUTUANTES

Para o texto ficar posicionado abaixo das divs, deverá ser limpado o float

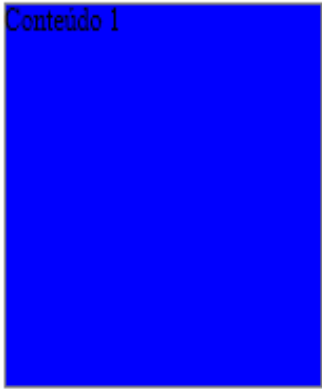
```
.clear{  
    clear: both;  
}
```



Permite limpar os elementos flutuantes em ambos os lados

ELEMENTOS FLUTUANTES

Conteúdo 1



Conteúdo 2



Conteúdo 3



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur sodales sapien magna, quis congue augue consequat et. Praesent tristique blandit scelerisque. In hac habitasse platea dictumst. Sed sodales massa ligula, sit amet pellentesque ipsum pellentesque eget. Quisque sem dolor, ornare tincidunt rutrum ac, imperdiet sed elit. Fusce pharetra magna sed ante placerat, nec faucibus ullamcorper lobortis sodales. Maecenas purus mauris, vulputate sed viverra sit amet, suscipit id turpis.



Neste caso para o texto aparecer abaixo das divs foi usado a propriedade **clear:both** ou então quando flutuando à esquerda ou a direita também pode ser usado as propriedades **clear:left** ou **clear:right**

ELEMENTOS INLINE, BLOCK e INLINE-BLOCK.

TAGS H1, TABLE, P, DIVS

São elementos do tipo bloco. Usam toda a largura da tela.

<a>, ,

São elementos **inline**. Usam conforme o espaço da tela.

Inline-block

Display:inline-block: Quando o conteúdo não cabe na mesma linha, ele automaticamente vai para outra linha.

POSICIONAMENTOS

RELATIVE, ABSOLUTE E FIXED

RELATIVE – Para posicionamentos relativos é usado as propriedades: top, right, bottom e left.

POSICIONAMENTO RELATIVO e ESTÁTICO

Posicionamento

Estudando cssEstudando cssEstudando cssEstudando
cssEstudando css Estudando cssEstudando cssEstudando
cssEstudando cssEstudando css Estudando cssEstudando
cssEstudando cssEstudando cssEstudando css Estudando
cssEstudando cssEstudando cssEstudando cssEstudando css
Estudando cssEstudando cssEstudando cssEstudando
cssEstudando css Estudando cssEstudando cssEstudando
cssEstudando cssEstudando css

Estudando cssEstudando cssEstudando cssEstudando
cssEstudando css Estudando cssEstudando cssEstudando
cssEstudando cssEstudando css Estudando cssEstudando
cssEstudando cssEstudando cssEstudando css Estudando
cssEstudando cssEstudando cssEstudando cssEstudando css
Estudando cssEstudando cssEstudando cssEstudando
cssEstudando css Estudando cssEstudando cssEstudando
cssEstudando cssEstudando css

Estudando cssEstudando cssEstudando cssEstudando
cssEstudando css Estudando cssEstudando cssEstudando
cssEstudando cssEstudando css Estudando cssEstudando
cssEstudando cssEstudando cssEstudando css Estudando
cssEstudando cssEstudando cssEstudando cssEstudando css
Estudando cssEstudando cssEstudando cssEstudando
cssEstudando css Estudando cssEstudando cssEstudando
cssEstudando cssEstudando css

Quando é usado posição relativa o parágrafo é deslocado conforme a posição inicial. Destaque para o parágrafo da borda vermelha.

Podemos dizer que agora os parágrafos estão posicionados no modo estático que é o posicionamento padrão dos elementos.

POSICIONAMENTO RELATIVO

Posicionamento

Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css

Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css

Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css

```
.relative{  
    border:1px solid red;  
    position:relative;  
    top:20px;  
    left:20px;  
}
```

O posicionamento é relativo. As propriedades **top** e **left** irão posicionar a partir do ponto inicial onde se encontra.

POSICIONAMENTO ABSOLUTO E FIXO

Posicionamento

Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css

Estudando css Estudando css Estudando css Estudando
Posicionamento absoluto Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css

```
.absolute{  
    border:1px solid red;  
    position:absolute;  
}
```

Quando o posicionamento é absoluto ele sai do fluxo normal do documento e não afeta a posição dos outros elementos da página.

POSICIONAMENTO ABSOLUTO E FIXO

css Estudando cssEstudando cssEstudando cssEstudando

cssEstudando css Estudando cssEstudando cssEstudando

cssEstudando cssEstudando cssEstudando cssEstudando

Estudando css Estudando

cssEstudando cssEstudando cssEstudando cssEstudando css
Estudando cssEstudando cssEstudando cssEstudando

Estudando C Estudando C Estudando C Estudando C Estudando C Estudando C

~~Estudando~~ ~~Estudando~~ ~~Estudando~~ ~~Estudando~~

cssEstudando cssEstudando css Estudando cssEstudando
 cssEstudando cssEstudando cssEstudando cssEstudando

cssEstudando cssEstudando cssEstudando cssEstudando
cssEstudando cssEstudando cssEstudando cssEstudando css

Estudando c#sEstudando c#sEstudando c#sEstudando c#s
Estudando c#sEstudando c#sEstudando c#sEstudando c#s

Estudando cssEstudando cssEstudando cssEstudando cssEstudando
cssEstudando cssEstudando cssEstudando cssEstudando

cssEstudando cssEstudando cssEstudando cssEstudando
cssEstudando cssEstudando css

Estudando cssEstudando cssEstudando cssEstudando

cssEstudando cssEstudando cssEstudando cssEstudando

cssEstudando cssEstudando cssEstudando cssEstudando

cssEstudando cssEstudando cssEstudando cssEstudando
cssEstudando cssEstudando cssEstudando cssEstudando css

Estudando esse Estudando esse Estudando esse Estudando esse
Estudando esse Estudando esse Estudando esse Estudando esse

Estudando cssEstudando cssEstudando cssEstudando cssEstudando
cssEstudando cssEstudando cssEstudando cssEstudando

cssEstudando cssEstudando css

```
.absolute{
  border:1px solid red;
  position:absolute;
  top:5px;
}
```

No posicionamento absoluto uma vez definido uma propriedade com um valor, exemplo **top: 5px** ele posiciona a distância da margem superior no documento.

POSICIONAMENTO ABSOLUTO E FIXO

Pós-Posicionamento

Estudando cssEstudando cssEstudando cssEstudando
cssEstudando css Estudando cssEstudando cssEstudando
cssEstudando cssEstudando css Estudando cssEstudando
cssEstudando cssEstudando cssEstudando css Estudando
cssEstudando cssEstudando cssEstudando cssEstudando css
Estudando cssEstudando cssEstudando cssEstudando
cssEstudando css Estudando cssEstudando cssEstudando
cssEstudando cssEstudando css

```
#container{
    background: #fff;
    width: 800px;
    margin: 50px auto;
    text-align: center;
    padding: 10px;
    position: relative
}
```

```
.absolute{
  border:1px solid red;
  position:absolute;
  top:5px;
}
```

Se a div estiver dentro de um container que no caso é a parte branca, para o posicionamento se basear pelo container, é necessário deixar o posicionamento do container como relativo.

POSICIONAMENTO ABSOLUTO E FIXO

Posicionamento

Estudando cssEstudando cssEstudando cssEstudando
cssEstudando css Estudando cssEstudando cssEstudando
cssEstudando cssEstudando css Estudando cssEstudando
cssEstudando cssEstudando cssEstudando css Estudando
cssEstudando cssEstudando cssEstudando cssEstudando css
Estudando cssEstudando cssEstudando cssEstudando
cssEstudando css Estudando cssEstudando cssEstudando
cssEstudando cssEstudando css

Estudando cssEstudando cssEstudando cssEstudando
cssEstudando css Estudando cssEstudando cssEstudando
cssEstudando cssEstudando css Estudando cssEstudando
cssEstudando cssEstudando cssEstudando css Estudando
cssEstudando cssEstudando cssEstudando cssEstudando css
Estudando cssEstudando cssEstudando cssEstudando
cssEstudando css Estudando cssEstudando cssEstudando
cssEstudando cssEstudando css

Posição Absoluta cssEstudando cssEstudando cssEstudando
css Estudando cssEstudando cssEstudando cssEstudando
cssEstudando css Estudando cssEstudando cssEstudando
cssEstudando cssEstudando css Estudando cssEstudando
cssEstudando cssEstudando cssEstudando css Estudando
cssEstudando cssEstudando cssEstudando cssEstudando css
Estudando cssEstudando cssEstudando cssEstudando
cssEstudando css

Conforme a imagem acima, o parágrafo que está com posição absoluta está ao lado direito

```
.absolute{  
    border:1px solid red;  
    position:absolute;  
    top:15px;  
    right:15px;  
}
```

POSICIONAMENTO ABSOLUTO E FIXO

Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css

Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css

Posição Absoluta css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css

Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css Estudando css Estudando
css Estudando css Estudando css

```
.fixed{  
    border:1px solid red;  
    position:fixed;  
}
```



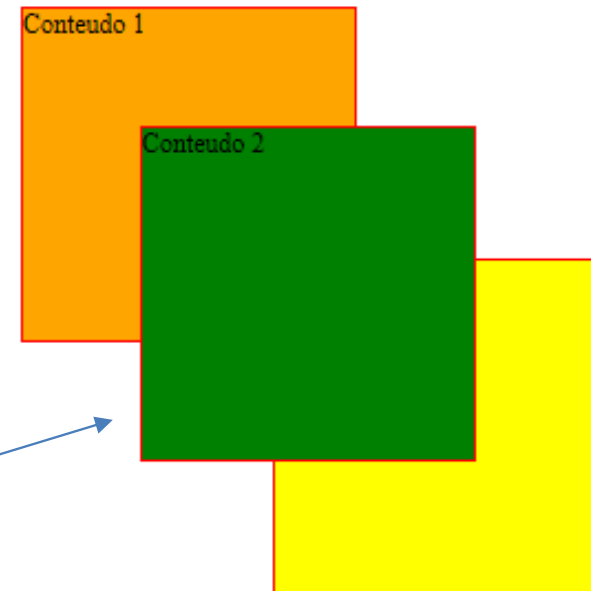
O parágrafo que está com borda vermelha está com posicionamento fixo. Ele fica fixo em relação a janela do navegador. Caso houver alguma barra de rolagem não se desloca deixando o elemento fixo.

SOBREPONDO ELEMENTOS.

```
.conteudo1 {  
  border: 1px solid red;  
  background: orange;  
  width: 200px;  
  height: 200px;  
  position: absolute;  
  z-index: 0;  
}
```

```
.conteudo2 {  
  border: 1px solid red;  
  background: green;  
  width: 200px;  
  height: 200px;  
  position: absolute;  
  top: 80px;  
  left: 80px;  
  z-index: 2;  
}
```

```
.conteudo3 {  
  border: 1px solid red;  
  background: yellow;  
  width: 200px;  
  height: 200px;  
  position: absolute;  
  top: 160px;  
  left: 160px;  
  z-index: 1;  
}
```



A propriedade z-index determina qual profundidade dos elementos. Como exemplo o conteúdo 2 é o elemento mais próximo

LISTAS

- ✓ Como o nome diz, são elementos HTML responsáveis por criar uma lista de elementos ordenados ou não.
- ✓ São elementos bastante utilizados atualmente, principalmente no que se refere a criação de menus de navegação.

LISTAS

Existem dois tipos de listas em HTML:

- ✓ Ordenados: Utilizam letras e números como marcadores.
- ✓ Não ordenados: Utilizam símbolos como marcadores.
- ✓ As listas possuem algumas predefinições de estilos:

List-style, list-style-image, list-style-position, list-style-type.

EXEMPLO COM LISTAS COM STYLE-IMAGE

Exemplo em HTML

```
<ul class="listas_lista">
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
</ul>
```

Exemplo em Css

```
.listas_lista{
  list-style-image:url("imagens/seta_lista.jpg");
}
```

EXEMPLO COM LISTAS COM STYLE-POSITION

Exemplo em HTML

```
<ul class="listas_lista">
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
</ul>
```

Exemplo em Css

```
.listas_lista{
  list-style-image:url("imagens/seta_lista.jpg");
  list-style-position:inside;
}

.listas_lista{
  list-style-image:url("imagens/seta_lista.jpg");
  list-style-position:outside;
}
```

EXEMPLO COM LISTAS COM STYLE-TYPE

Exemplo em HTML

```
<ul class="listas_lista">
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
  <li>Elemento li</li>
</ul>
```

Exemplo em Css

```
.listas_lista{
    list-style-type:circle;
}
```

WEBLINKS- FORMATAÇÃO DE LINKS

✓ Links possuem diversos tipos de formatação, sendo que é possível criar menus bastante elaborado reunindo listas, links e CSS.

✓ Os weblinks possuem alguns estados:

Hover – Quando o mouse está em cima do link.

Visited – Quando o link já foi visitado.

Link – Simplesmente sem ter visitado.

Active - Quando o link é clicado.

EXEMPLO COM WEBLINKS

Exemplo em HTML

```
<ul>
  <li><a href="#">weblink</a></li>
  <li><a href="#">weblink</a></li>
  <li><a href="#">weblink</a></li>
  <li><a href="#">weblink</a></li>
  <li><a href="#">weblink</a></li>
  <li><a href="#">weblink</a></li>
  <li><a href="#">weblink</a></li>
  <li><a href="#">weblink</a></li>
  <li><a href="#">weblink</a></li>
</ul>
```

EXEMPLO COM WEBLINKS FORMATADO

Exemplo em HTML

```
<ul class="weblink_link">
  <li><a href="#">weblink</a></li>
  <li><a href="#">weblink</a></li>
  <li><a href="#">weblink</a></li>
  <li><a href="#">weblink</a></li>
  <li><a href="#">weblink</a></li>
  <li><a href="#">weblink</a></li>
  <li><a href="#">weblink</a></li>
  <li><a href="#">weblink</a></li>
</ul>
```

Exemplo em CSS

```
.weblink_link{
  padding: 0px;
  margin: 0px;
  list-style-type: none
}
```

EXEMPLO COM WEBLINKS FORMATADO

Como no exemplo anterior vamos ter os links um abaixo do outro. Para colocar um do lado do outro faça o seguinte:

```
.weblink_link li{  
    display:inline;  
}
```

EXEMPLO COM WEBLINKS FORMATADO

Para fazer uma formatação de espaçamento e os itens, faça o seguinte:

```
.weblink_link li{  
    display:inline;  
    margin: 8px;  
}
```


EXEMPLO COM WEBLINKS FORMATADO

Para colocar os itens centralizado, faça o seguinte:

```
.weblink_link{  
    padding: 0px;  
    margin:0px;  
    list-style-type:none;  
    width: 940px;  
    margin-left:auto;  
    margin-right:auto;  
}
```

EXEMPLO COM WEBLINKS FORMATADO

Para formatar as cores dos itens, faça o seguinte:

```
.weblink_link a{  
    color:white;  
}
```

Para tirar o sublinhado, faça o seguinte:

```
.weblink_link a{  
    color:white;  
    text-decoration:none;  
}
```

EXEMPLO COM WEBLINKS FORMATADO

Quando passar o mouse sobre o link, trocar a cor:

```
.weblink_link a:hover{  
    background: black;  
}
```

Quando passar o mouse sobre o link, colocar um sublinhado:

```
.weblink_link a:hover{  
    background: black;  
    text-decoration:underline;  
}
```

EXEMPLO COM WEBLINKS FORMATADO

Quando passar o mouse sobre o link e pressionar o botão do mouse, aparecer uma cor:

```
.weblink_link a:active{  
    background: red;  
}
```

EXEMPLO COM WEBLINKS FORMATADO

Uma outra formatação também definir quando um link foi visitado, para tanto faça o seguinte:

```
.weblink_link a:visited{  
    color: yellow;  
}
```

TABELAS

- ✓ São elementos que tem por finalidade exibir **dados tabulares**.
- ✓ Não é recomendável usar **tabelas** para montar uma estrutura de formulário, mas podem ser utilizadas para outras finalidades.

TABELAS

Tabelas são compostas por:

<table></table> - Objeto tabela propriamente dito.

<tr></tr> - São as linhas de uma tabela.

<td></td> - São as colunas de uma tabela.

TABELAS

As tabelas possuem muitas propriedades e tem uma formatação bastante flexível.

Uma propriedade pouco conhecida é a propriedade `border-collapse: collapse` que tem por finalidade transformar a borda tabela em uma linha simples.

EXEMPLO DE IMPLEMENTAÇÃO DE TABELA

```
<table>
  <thead>
    <tr>
      <th>Cabeçalho01</th>
      <th>Cabeçalho01</th>
      <th>Cabeçalho01</th>
      <th>Cabeçalho01</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Valor1</td>
      <td>Valor1</td>
      <td>Valor1</td>
      <td>Valor1</td>
    </tr>
    <tr>
      <td>Valor2</td>
      <td>Valor2</td>
      <td>Valor2</td>
      <td>Valor2</td>
    </tr>
  </tbody>
</table>
```

O thead permite que o cabeçalho seja formatado em negrito

EXEMPLO DE IMPLEMENTAÇÃO DE TABELA

Exemplo no HTML

```
<table class="Tabelas_tabela01">
  <thead>
    <tr>
      <th>Cabeçalho01</th>
      <th>Cabeçalho01</th>
      <th>Cabeçalho01</th>
      <th>Cabeçalho01</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Valor1</td>
      <td>Valor1</td>
      <td>Valor1</td>
      <td>Valor1</td>
    </tr>
    <tr>
      <td>Valor2</td>
      <td>Valor2</td>
      <td>Valor2</td>
      <td>Valor2</td>
    </tr>
  </tbody>
</table>
```

Exemplo no CSS

```
.Tabelas_tabela01 {
  width:50%;
  border-collapse:collapse;
  border: 1px solid black;
  margin-left:auto;
  margin-right:auto;
}
```

EXEMPLO DE IMPLEMENTAÇÃO DE TABELA

Para colocar bordas nas células

```
.Tabelas_tabela01 td {  
    border: 1px solid black;  
    padding: 10px;  
}
```

EXEMPLO DE IMPLEMENTAÇÃO DE TABELA

Para alternar linhas com cores

```
<tr class="trPar">
  <td>Valor1</td>
  <td>Valor1</td>
  <td>Valor1</td>
  <td>Valor1</td>
</tr>
<tr>
  <td>Valor2</td>
  <td>Valor2</td>
  <td>Valor2</td>
  <td>Valor2</td>
</tr>
<tr class="trPar">
  <td>Valor2</td>
  <td>Valor2</td>
  <td>Valor2</td>
  <td>Valor2</td>
</tr>
<tr>
  <td>Valor2</td>
  <td>Valor2</td>
  <td>Valor2</td>
  <td>Valor2</td>
</tr>
```

```
.tabelas_tabela01 .trPar{
    background-color: green;
}
```

EXEMPLO DE IMPLEMENTAÇÃO DE TABELA

Quando passar o mouse pela célula, atribuir uma cor.

```
.Tabelas_tabela01 tr:hover {  
    background:blue;  
    color:white;  
}
```

EXEMPLO DE IMPLEMENTAÇÃO DE TABELA

Para não colocar cor no cabeçalho.

```
.Tabelas_tabela01 thead tr:header {  
    background:none;  
    color:black;  
}
```

EXEMPLO DE IMPLEMENTAÇÃO DE TABELA

Para colocar uma cor de fundo no cabeçalho.

```
.Tabelas_tabela01 thead tr {  
    background:red;  
    color:white;  
}
```

EXEMPLO DE IMPLEMENTAÇÃO DE TABELA

Propriedade **COLLAPSE**

```
table{  
    width:100%;  
    border-collapse: separate;  
}
```

Uma vez que aplicado as bordas, elas ficam separadas

```
th,td{  
    border:1px solid red;  
}
```

Código	Descrição	Valor Unit
001	Notebook	2000,00
002	Bastão de Selfie	35,00
003	JBL	170,00

EXEMPLO DE IMPLEMENTAÇÃO DE TABELA

Propriedade COLLAPSE

```
table{  
    width:100%;  
    border-collapse: collapse;  
}
```

O valor **collapse** na propriedade **border-collapse** mantém as colunas juntas

```
th,td{  
    border:1px solid red;  
}
```

Código	Descrição	Valor Unit
001	Notebook	2000,00
002	Bastão de Selfie	35,00
003	JBL	170,00

FORMULÁRIOS

Formulários são bastante utilizados para que usuários de sites e sistemas faça a entrada de dados. Os campos de entrada no formulário são:

✓ Campos de texto, área de texto, listas, campos de checkbox, radio buttons e labels.

FORMULÁRIOS

```
<form id="frmContato">
  <label for="campoNome">
    Nome: <input type="text" name="campoNome" id="campoNome" />
  </label>
  <label for="campoEmail">
    E-mail: <input type="text" name="campoEmail" id="campoEmail" />
  </label>
  <textarea id="mensagem"></textarea>
  <input type="submit" value="Enviar formulário" class="aula14_botao" />
</form>
```

FORMATANDO FORMULÁRIOS

Exemplo

```
#frmContato {  
    width:500px;  
    margin-left:auto;  
    margin-right:auto;  
    border:1px solid gray;  
    padding:20px;  
}
```

FORMATANDO LABEL

Exemplo

```
#frmContato label {  
    display: block;  
    margin-bottom: 20px;  
}
```

FORMATANDO TEXTAREA

Exemplo

```
#frmContato textarea {  
    width:100%;  
    margin-bottom:15px;  
    height:180px;  
    border:1px solid gray;  
    border-radius:10px;  
    padding:7px;  
}
```

FORMATANDO O TEXTAREA QUANDO RECEBE O FOCO

Exemplo

```
#frmContato textarea:focus {  
    border: 1px dashed blue;  
}
```

FORMATANDO UMA LABEL QUANDO É PASSADO O MOUSE SOBRE O CAMPO

Exemplo

```
#frmContato label input:hover {  
    border: 1px dashed blue;  
}
```


FORMATANDO UMA ENTRADA NA LABEL

Exemplo

```
#frmContato label input {  
    width:350px;  
    border:1px solid gray;  
    border-radius:10px;  
    padding:7px;  
}
```

FORMATANDO UM BOTÃO

Exemplo de formatação do botão

```
.formulario_botao {  
    border-radius:10px;  
    border:1px solid gray;  
    color:white;  
    background:blue;  
    padding:10px;  
}
```

Exemplo de formatação do botão
quando é passado sobre o mouse

```
.formulario_botao:hover {  
    background:black;  
}
```

AGRUPANDO DADOS SEMELHANTES EM FORMULÁRIOS

FIELDSET: Permite fazer agrupamento de formulários

Detalhes do Contato

Nome:

Email:

Url:

LEGEND: Permite descrever o que vai no FIELDSET

Exemplo de implementação com **FILDSET** e **LEGEND**

```
<fieldset>
  <legend>Detalhes do Contato</legend>
  <div>
    <label for="nome">Nome:</label>
    <input type="text" name="nome" id="nome">
  </div>
  <div>
    <label for="email">Email:</label>
    <input type="text" name="email" id="email">
  </div>
  <div>
    <label for="url">Url:</label>
    <input type="text" name="url" id="url">
  </div>
</fieldset>
```

Implementação de um atributo


```
input [type="text"] {  
    width:20em;  
}
```

No formulário quando tem vários componentes, para estilizar somente atributo de um determinado tipo, como no exemplo input do tipo **TEXT**, é colocado entre **colchetes** como no exemplo acima.

Neste caso somente do tipo **text** é que vai ter uma largura de 20em

Implementação de um atributo

```
input.radio, input.checkbox, input.submit{  
    width: auto;  
}
```



Irá definir a largura automática para os componentes do formulário.