

LPW-CSS

Eduardo Vasconcelos

eduardo.vasconcelos@garanhuns.ifp
e.edu.br



Cascading Styles Sheets

- CSS – Folha de Estilo:
 - Permite configurar a apresentação da página
 - Editar tamanho, cor, cor de fundo e assim por diante;
 - Pode ser aplicado a qualquer elemento do HTML;
 - Não é uma linguagem de programação;



Cascading Styles Sheets

- CSS – Folha de Estilo:

- Vantagens:

- É possível organizar a apresentação da página de forma separada do código HTML;
 - Diminui a quantidade de código na página;
 - Pode-se aplicar um estilo de apresentação a mais de um elemento;
 - Pode-se aplicar um conjunto de estilos a várias páginas;



Cascading Styles Sheets

- CSS – Folha de Estilo:
 - É uma boa prática utilizar CSS para definir a apresentação de uma página;
 - A maioria dos atributos HTML para definição de estilo das tags, são consideradas depreciadas pela maioria das IDEs.



Cascading Styles Sheets

- Comandos inline.
 - São inseridos na própria tag através do atributo style.
 - `<p style="color:red">LPW</p>`
 - Neste caso, o parágrafo é definido na cor vermelha;
 - `<p style="text-align:center">Olá</p>`
 - Alinha ao centro.



Cascading Styles Sheets

- Comandos na tag style:
 - É possível definir o estilo da pagina usando a tag <style>.
 - Esta tag deve estar localizada no cabeçalho da página:

<style>

p{color:blue;}

</style>



Cascading Styles Sheets

- Usando arquivo externo.
 - É possível definir o estilo em um arquivo separado.
 - O arquivo deve possuir a extensão .css;
 - Para incluir um arquivo .css à página usa-se
 - `<link rel="stylesheet" href="meucss.css"/>`
 - A tag `<link>` deve ser colocada no cabeçalho da página.



Cascading Styles Sheets

- Exemplo de CSS:

```
<html>
  <head>
    <title>Exemplo CSS</title>
    <style>
      p{
        text-align: center;
        color:green;
        font-size: 100pt;
      }
    </style>
  </head>
  <body>
    <p>Exemplo CSS</p>
  </body>
</html>
```




Cascading Styles Sheets

- Estrutura do CSS:
 - Os estilos CSS são construídos por meio de declarações;
 - Estrutura básica de uma declaração;
 - Atributo:valor;
 - Os atributos são elementos padrão do CSS;
 - Color; font-size, font-style; text-align...;
 - Os valores correspondem ao estado que determinado atributo deve conter;
 - Em color os valores podem ser descritos com palavras reservadas ou combinações hexadecimais.



Cascading Styles Sheets

- Seletores:
 - Permite selecionar e manipular elementos HTML;
 - Estrutura básica:
Seletor{
 declarações
}

```
p{  
    text-align: center;  
    color:green;  
    font-size: 100pt;  
}
```

Seletores

- Seletor de Elemento:
 - O seletor é o conjunto de caracteres que formam a tag HTML;
H1{}
 - No seletor acima, todas as declarações que estiverem dentro da chave serão aplicados a todos as tags “H1” da página.

```
img{  
    width: 40%;  
    height: 100%;  
}
```

Seletores

- Seletor de Id:
 - Aplica um conjunto de declarações a um elemento html com id especificado;
 - Para usar este seletor usa-se o simbolo “#”;

```
#id1{  
    color:purple;  
    font-style: italic;  
}  
  
</style>  
</head>  
<body>  
<p id="id1">Exemplo seletor id</p>
```

Seletores

- Seletor de classe:
 - É possível definir classes que correspondem a blocos de estilos que podem ser aplicados a qualquer elemento;
 - As classes são inicializadas com o símbolo “.”;

```
.Minhaclasse{  
    color:grey;  
    width: 40%;  
}
```

```
  
<table class="Minhaclasse" border="2">
```

Seletores

- Seletor de classe para determinados elementos:
 - É possível definir classes que só serão aplicadas a alguns elementos;
 - Estrutura;
 - Elemento.classe{
 - No caso da figura ao lado,
Minha classe só pode ser aplicada à tag .

```
img.MinhaClasse{  
    color:grey;  
    width: 40%;  
}
```

Seletores

- Agrupando seletores;
 - É possível aplicar um estilo a vários elementos.

```
img, table, div, button{  
    color:grey;  
    width: 40%;  
}
```



Seletores

- Múltiplos Estilos;
 - É possível definir estilos diferentes para o mesmo elemento em locais diferentes;
 - Exemplo – é possível definir um estilo para o elemento h1 em um arquivo .css, e fazer uma definição diferente para o mesmo elemento na tag <style> da página;
 - Neste caso, a prioridade é do estilo que está na página;
 - Os elementos definidos no arquivo e não na tag <style> permanecem ativos.

Seletores

- Múltiplos Estilos (cont);

```
newCSS.css  ⓘ  
  
p{  
    color: blueviolet;  
    text-align: center;  
    font-size: 10pt;  
}
```

```
<style>  
    p{  
        font-size: 40pt;  
    }  
</style>
```

- No caso acima, o elemento <p> da página terá:
 - Color: blueviolet;
 - Text-align: center;
 - **Font-size: 40pt;**

Seletores

- Múltiplos Estilos (cont);
 - O estilo colocado na tag tem prioridade sobre os demais;
 - O estilo definido depois também tem prioridade;

```
p{
    text-align: center;
    color:green;
    font-size: 100pt;
}
p{
    color: blue;
    font-size:60pt;
}

</style>
</head>
<body>
    <p style="font-size: 30pt">Qual o estilo aplicado aqui?</p>
    <p>e aqui?</p>
```



Background

- background-color:
 - Usado para definir cor de fundo a um elemento;
- background-image:
 - Permite colocar uma imagem de fundo;
 - Uso: background-image: url("paper.gif");
- background-repeat:
 - Quando uma imagem é inserida como plano de fundo, ela é repetida tanto verticalmente quanto horizontalmente;
 - Uso; background-repeat: repeat-x;
 - » A imagem é repetida apenas na horizontal.

Background

- background-position:

- Quando a imagem não é repetida no plano de fundo é possível posiciona-la;

- Uso: background-repeat: no-repeat;
background-position: right top;

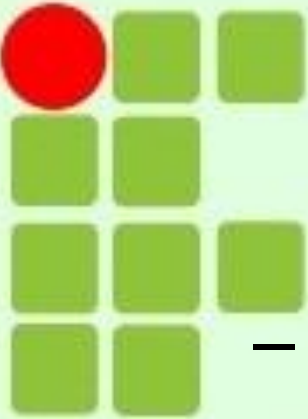
- É possível também aplicar vários valores ao plano de fundo:

- background: #ffffff url("img_tree.png") no-repeat right top;



Formatando texto

- Color:
 - Define a cor do texto;
- text-align:
 - Define a forma como o texto será alinhado;
 - Center, right e Justify;
- text-decoration:
 - Utilizado para decorar um texto;
 - Overline, line-through e underline;
- text-transform:
 - Usado para mudar a forma como o texto é apresentado
 - » Uppercase, lowercase and capitalize;



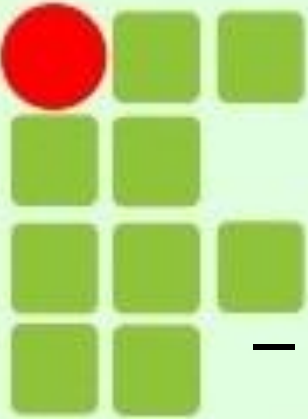
Formatando texto

- text-transform:

- Usado para mudar a forma como o texto é apresentado
 - Uppercase, lowercase and capitalize;

- text-indent:

- Define tabulação de um texto;
 - Uso: text-indent: 30px;
 - Define tabulação de 30 pixels.



Formatando texto

- text-transform:

- Usado para mudar a forma como o texto é apresentado
 - Uppercase, lowercase and capitalize;

- text-indent:

- Define tabulação de um texto;
 - Uso: text-indent: 30px;
 - Define tabulação de 30 pixels.

Font

– font-family:

- Permite definir a família da font;
 - "Times New Roman", Times, serif;

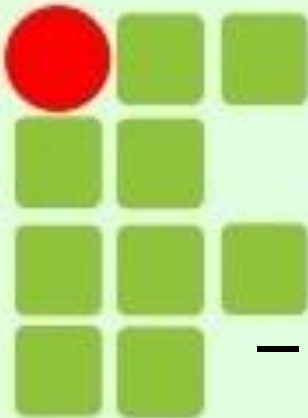
– font-style:

- Define o estilo da fonte;
 - Normal, italic, oblique

– font-size:

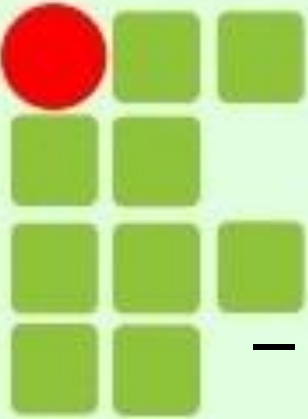
- Define o tamanho da fonte;
 - Pode ser definida em px, pt, em...





Propriedades dos links

- É possível com CSS mudar as propriedades do link:
 - Ex: mudar a cor de um link visitado;
- a:link:
 - Mudar as propriedades do link que não foram visitados;
- a:visited:
 - Mudar as propriedades do link visitado;



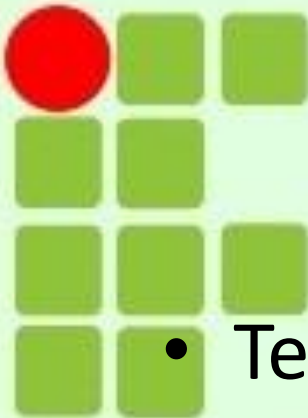
Propriedades dos links

– a:hover:

- Permite aplicar um estilo ao link quando o usuário passa o cursor do mouse sobre este;

– a:active:

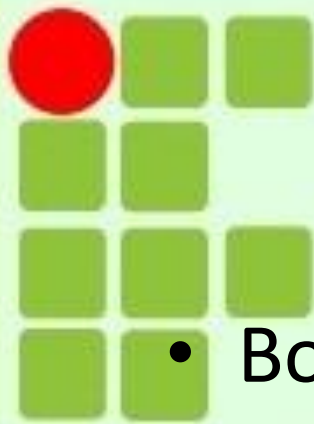
- Utilizado para definir um estilo para quando o link é clicado.



Propriedades dos links

- Teste os estilos em uma página contendo um link:

```
a:link{
    color:green;
    font-size:10pt;
}
a:visited{
    color:brown;
    font-size: 20pt;
}
a:hover{
    color: red;
    font-size:30pt;
}
a:active{
    color: yellow;
    font-size: 40pt;
}
```



Tabelas

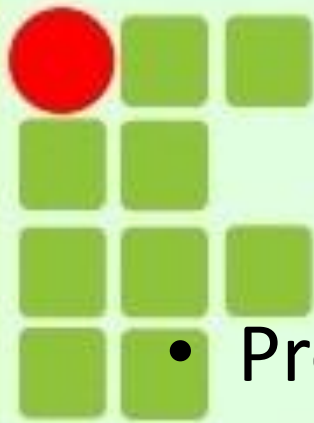
- Bordas de uma tabela:
 - É possível definir bordas usando a propriedade border do CSS;
 - O estilo a seguir cria uma borda em uma tabela;
 - `table, td, th { border: 1px solid black; }`
 - Note que a propriedade border tem 3 valores;

testel	teste2
mude	os valores
para ver	como fica

Tabelas

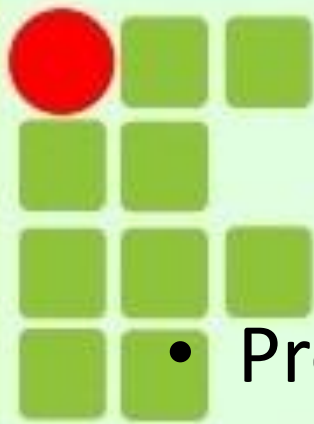
- Bordas de uma tabela:
 - Para definir a borda de uma tabela com uma única linha usa-se a propriedade:
 - ```
table {
 border-collapse: collapse;
}
```

| teste1   | teste2     |
|----------|------------|
| mude     | os valores |
| para ver | como fica  |



# Tabelas

- Propriedades de uma tabela:
  - Podem ser aplicadas aos elementos <table>, <td>, <th>
  - Width:
    - Define a largura da tabela;
  - Height:
    - Define altura.
  - Text-align:
    - Define a posição do texto na tabela (left, center, right...)



# Tabelas

- Propriedades de uma tabela:
  - Podem ser aplicadas ao elemento `<td>`
    - vertical-align:
      - Permite alinhar o texto de forma vertical;
        - » vertical-align: bottom;
    - padding:
      - Insere um espaço dentro da célula.
      - Valor definido em pixels.

| teste1   | teste2     |
|----------|------------|
| mude     | os valores |
| para ver | como fica  |

# Tabelas

- Exemplo de tabela:

```
table {
 border-collapse: collapse;
}

table, td, th {
 border: 1px solid greenyellow;
}

th{
 background-color: graytext;
 color: whitesmoke;
}

td{
 vertical-align: bottom;
 padding: 30px;
 background-color: cyan;
}
```

| teste1   | teste2     |
|----------|------------|
| mude     | os valores |
| para ver | como fica  |



# Tabelas

- Exemplo de tabela:

```
table {
 border-collapse: collapse;
}

table, td, th {
 border: 1px solid greenyellow;
}

th{
 background-color: graytext;
 color: whitesmoke;
}

td{
 vertical-align: bottom;
 padding: 30px;
 background-color: cyan;
}
```

| teste1   | teste2     |
|----------|------------|
| mude     | os valores |
| para ver | como fica  |



# Elemento <div>

- Em HTML, todos os elementos são considerados como uma “caixa”;
  - Geralmente estas caixas contém algum tipo de conteúdo:
    - Texto ou imagem;
  - Elementos como tabelas e formulários, possuem definições específicas;
  - O elemento <div> é uma caixa vazia.



# Elemento <div>

– Pra que serve um <div> então?

- Serve para agrupar elementos.

– <div>

    <h1>este elemento esta dentro da div</h1>

    <h2>este também</h2>

  </div>

- No exemplos, os elementos h1 e h2 estão agrupados pelo div.
- Qualquer estilo aplicado ao div também será aplicado aos elementos.



# Elemento <div>

– Pra que serve um <div> então?

```
div {

 font-style: italic;
 color: indianred;
 font-size: 20pt;
}
```

```
<div>
 <h1>este elemento esta dentro da div</h1>
 <h2>este também</h2>
</div>
```

*este elemento esta dentro da div*  
*este também*



# Elemento <div>

– É possível também posicionar um div na página.

- margin-top, margin-left, margin-right, margin-bottom;
- Os valores destas propriedades podem ser definidos em “%” ou “px” por exemplo;
- Os valores são referentes a distancia entre o elemento e a margem referida;
  - Margin-top: 200px;
    - » A margem do topo do elemento fica a 200 pixels de distância do elemento anteriormente colocado no HTML.



# Elemento <div>

- Crie um elemento <div> na página e em seguida aplique o seguinte estilo.

```
div {
 margin-top: 10%;
 margin-left: 20%;
 margin-bottom: 30%;
 margin-right: 10%;
 border-style: dashed;
}
```

- Modifique o valor de margin-right para 20% e veja o que acontece;
- Adicione uma tabela com 4 linhas e 2 colunas na página e veja o que acontece.

# Elemento <div>

– Aplique os códigos abaixo e veja o que acontece;

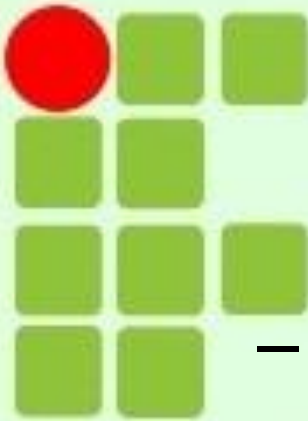
- explique o motivo que faz um div ficar do lado do outro.

```
div.classeMain{
 margin-top: 0%;
 margin-left: 0%;
}
div.classe1{
 margin-top: 0%;
 margin-left: 0%;
 border-style: dashed;
 padding: 100px;
}
div.classe2{
 margin-top: 0%;
 margin-left: 0%;
 margin-right: 80%;
 margin-bottom: 0%;
 border-style: dashed;
}
div.classe3{
 margin-top: -6.35%;
 margin-left: 20%;
 margin-bottom: 0%;
 border-style: dashed;
}
```

```
<div class="classe1">
 <p>primeira parte</p>
</div>

<div class="classeMain">
 <div class="classe2">
 <h1>Parte 2</h1>
 </div>

 <div class="classe3">
 <h1>Parte 3</h1>
 </div>
</div>
```



# Posicionando elementos

- É possível posicionar um elemento utilizando as propriedades “margin” como mostrado para o <div>;
- É possível também utilizar outras propriedades para posicionar elementos;
  - Propriedade “position”:
    - Permite definir como um elemento será posicionado na página;
    - Se o valor for “relative”, a posição do elemento dependerá dos outros elementos da página;
    - Se for “absolute”, a posição do elemento será relativo apenas a página a qual ele está inserido.



# Posicionando elementos

- Aplique os seguintes códigos a uma página HTML;

```
<style>
 p.posicao{
 position: absolute;
 top: 200px;
 left: 200px;
 }

 table, td, th{
 border: 1px solid grey
 }
</style>

<table>
 <tr><th colspan="2" >cabeçalho</th></tr>
 <tr><td>coisas dentro da tabela</td><td>outra coisa</td></tr>
</table>

<p class="posicao">oi tudo bem?</p>
```

- Mude o valor do position para “relative” e veja o que acontece com o parágrafo.

# Referências

– <http://www.w3schools.com/css/default.asp>