

The page features a decorative graphic on the right side consisting of three blue circles of varying sizes, each with a lighter blue ring around its center. Two thin blue lines intersect at the top left, forming a large 'V' shape that frames the circles. The largest circle is at the top, a smaller one is in the middle, and a very large one is at the bottom right, partially cut off by the edge of the page.

HTML e CSS

Comandos Básicos de Introdução ao HTML
e CSS

Prof. Vitor Hugo Migoto de Gouvêa
Colégio IDESA – Minicursos 2010

Conteúdo

Sintaxe do HTML	3
Tags de Formatação de Texto e Parágrafo	4
A tag de Parágrafo	4
A tag de Formatação de fontes	5
Criando Hyperlinks	6
Hiperlinks de Ligação de páginas	6
Hiperlinks para criação de âncoras.....	6
Inserindo imagem no HTML	8
Sintaxe do CSS	9
CSS Interno.....	10
Identificando características pelo nome da TAG.....	10
Identificando características por um ID.....	12
Identificando características de Classes	13
CSS Externo	14
Uso de Div`s nos Layouts.....	16
Links indicados	18

Sintaxe do HTML

A linguagem HTML possui uma sintaxe simples, podemos entender que dentro de uma página HTML temos um cabeçalho, com um título, e o corpo da nossa página.

Vejamos a sintaxe:

```
<html>
  <head>
    <title> Titulo que irá aparecer na barra de titulo do browser</title>
  </head>
  <body>
    Aqui entram todas as tags que irão formar o corpo da página
  </body>
</html>
```

Na linguagem HTML utilizamos tags que dizem ao computador que algo deverá ser produzido, as tags são formadas por palavras entre o sinal de menor (<) e o sinal de maior(>), por exemplo: <html> <head> <title>.

As maiorias das tags possuem abertura e fechamento indicando o inicio daquele comando e o final dele, para indicar a finalização de uma tag usamos a mesma tag só que com uma barra (/) depois do sinal de menor, vejamos um exemplo:

<HTML> ← inicio da tag HTML

</HTML> ← Finalização da tag HTML

A tag <html> indica ao browser o inicio de uma nova página de internet e seu fechamento </html> indica o final da mesma. Entre estas tags temos a tag <head> </head> que indica o inicio e fim do cabeçalho da página de internet, entre estas temos o <title> e o </title> que indica o título da página que irá aparecer na barra do navegador.

Após o cabeçalho temos a tag <body> e </body> que indica o começo e o fim do corpo da página, é entre essas tags que iremos inserir varias outras tags que iremos aprender para poder montar nossas páginas de internet.

Tags de Formatação de Texto e Parágrafo

A tag de Parágrafo

A partir de agora iremos aprender algumas tags para podermos começar a montar nossas paginas. Inicialmente vamos aprender como construir um parágrafo de texto para começarmos a digitar o conteúdo de nossas paginas.

Para iniciarmos um parágrafo em uma pagina de internet, usaremos a tag <p> e para finalizarmos usaremos a tag </p>. Veja o exemplo:

```
<html>
  <head>
    <title> Minha primeira página</title>
  </head>
  <body>
    <p>Meu primeiro texto</p>
  </body>
</html>
```

Nos parágrafos temos como utilizar também de parâmetros de alinhamento, algumas tags podem conter dentro delas parâmetros que irão configurar algumas propriedades do conteúdo delas. Na tag de parágrafo temos um parâmetro de alinhamento chamado *align*. Usamos este parâmetro com as seguintes propriedades que deverão estar entre aspas duplas:

- left
- right
- center
- justify

Consecutivamente estes parâmetros irão alinhar o texto que estão dentro das tags <p> a esquerda, ou a direita, ou centralizado ou então justificado. Para saber como usamos esse parâmetro vejamos o exemplo:

```
<html>
  <head>
    <title> Titulo que irá aparecer na barra de titulo do browser</title>
  </head>
  <body>
    <p align="left">Texto alinhado a esquerda</p>
    <p align="right">Texto alinhado a direita</p>
    <p align="center">Texto centralizado</p>
    <p align="justify">Texto justificado</p>
  </body>
</html>
```

A tag de Formatação de fontes

Ainda podemos fazer algumas configurações com o texto como por exemplo, trocar o tipo de letra, o tamanho da fonte e até alterar a cor do nosso texto. No HTML para realizarmos estas configurações utilizamos a tag `` e para finalizar a tag usamos o ``.

Os parâmetros dessa tag são os seguintes:

- `face` – altera o tipo da fonte que deverá ser incluída entre aspas duplas
- `size` – altera o tamanho da fonte que é colocada entre aspas duplas
- `color` – altera a cor do texto que deverá ser colocada entre aspas duplas e pode ser digitada em inglês ou então em código hexadecimal.

Vejamos um exemplo de uso desta tag:

```
<html>
  <head>
    <title> Titulo que irá aparecer na barra de titulo do browser</title>
  </head>
  <body>
    <p>Neste meu texto <font face="Arial" > essa linha será escrita em arial</font></p>
    <p>Neste meu texto <font size="12pt" > essa linha será escrita em tamanho 12 points</font></p>
    <p>Neste meu texto <font color="red" > essa linha será escrita em vermelho</font></p>
    <p>Neste meu texto <font face="Arial" size="12pt" color="red" > essa linha será escrita em arial no tamanho 12 points e na cor vermelha</font></p>
  </body>
</html>
```

Ainda temos algumas tags de destaque como temos num editor de texto padrão. Temos a opção de deixar um texto em negrito, itálico ou sublinhado usando as tags consecutivamente ` `, `<i></i>` e `<u> </u>`. Estas tags tem o significado em inglês de Bold `` (Negrito), Italic `<i>` (Itálico) e Underline `<u>` (Sublinhado). Usamos estas tags normalmente em nosso texto no trecho que você deseja destacar, por exemplo:

```
<html>
  <head>
    <title> Titulo que irá aparecer na barra de titulo do browser</title>
  </head>
  <body>
    <p>Neste meu texto <font face="Arial" > essa linha será escrita em arial <b> e este trecho sairá em negrito</b></font></p>
  </body>
</html>
```

Criando Hyperlinks

Hiperlinks de Ligação de páginas

Podemos usar uma tag para criar uma ligação com outro arquivo ou outra página de internet, para isso iremos usar a tag de hyperlink que é a tag <a>.

Para abrir uma página ou arquivo, usaremos dentro da tag <a> o parâmetro *href* este parâmetro trará entre aspas o caminho do arquivo ou página que desejar abrir.

Vejamos um exemplo

```
<html>
  <head>
    <title> Titulo que irá aparecer na barra de titulo do browser</title>
  </head>
  <body>
    <a href="http://www.idesa.com.br"> Clique aqui para abrir a pagina do IDESA
    </a>
    <a href="pagina.html"> Clique aqui para abrir a pagina.html</a>
    <a href="arquivo.doc"> Clique aqui para abrir o arquivo do Word
    arquivo.doc</a>

  </body>
</html>
```

Ainda temos a opção de indicar para que esta página seja aberta em outra janela, para isso usamos o parâmetro *target="_blank"*. *Veja o exemplo:*

```
<a href="pagina.html" target="_blank"> A pagina será aberta em outra janela
do navegador</a>
```

Hyperlinks para criação de âncoras

O que é âncora? Na verdade muitas pessoas se questionam sobre o que é isso. Âncora é uma ligação a um espaço ou texto localizado na mesma página, como por exemplo aqueles links que você encontra no final de uma página que te remete ao topo da página novamente.

Com o HTML podemos criar âncoras também, para isso usaremos no link uma nova propriedade que é o parâmetro *name*. O parâmetro *name* identificará como âncora o nome de um determinado lugar para o qual você quer remeter o link, feito isso no lugar determinado você indicará no link o nome indicado na âncora iniciado com um # (Lê-se suspenso) e então o link remeterá ao lugar que você determinou. Você irá notar também que a âncora não fica sublinhada como um link pois ela só servirá para identificar um local. Vejamos o exemplo de uso na próxima página:

Inserindo imagem no HTML

No HTML podemos também inserir uma imagem através da tag esta tag diferente de algumas outras tags, não precisa ser finalizada. Vejamos agora seus parâmetros.

- Src – indicará o caminho da imagem a ser mostrada
- Width – indicará a largura da imagem
- Height – indicará a altura da imagem
- Alt – indicará uma legenda da imagem que será mostrada quando o mouse for posicionado sobre ela

Visto seus parâmetros vejamos agora um exemplo de uso:

```
<html>
  <head>
    <title> Titulo que irá aparecer na barra de titulo do browser</title>
  </head>
  <body>
    <p> Veja minha foto:</p>
    <p align="center"></p>
  </body>
</html>
```

É bom saber que no tamanho de width e height você poderá indicar em porcentagem ou em pixels o tamanho a ser exibido, sendo que se não definidos ela ficará por padrão com o tamanho original da imagem que esta no arquivo. Podemos usar uma imagem também dentro da tag de link fazendo assim uma imagem que irá ligar algum arquivo ou pagina. Isso é muito utilizado em sites para fazer um link mais atraente parecendo um botão ou ícone. Vejamos um exemplo disso:

```
<html>
  <head>
    <title> Titulo que irá aparecer na barra de titulo do browser</title>
  </head>
  <body>
    <p> Clique no botão abaixo para ir ao site do idesa:</p>
    <p align="center"><a href="http://www.idesa.com.br "></a></p>
  </body>
</html>
```

É preciso saber que os arquivos de imagens que forem inseridas na pagina sempre deverão estar juntas do arquivo HTML, em caso de mover os arquivos as imagens devem acompanhar não saindo do caminho definido dentro das tags, pois a pagina HTML somente exibe a imagem sabendo que ela está no local indicado.

Sintaxe do CSS

Para dar início ao CSS é bom entendermos para que serve essa linguagem. A linguagem CSS ou Cascading Style Sheets é uma linguagem de estilo que definirá ou personalizará somente os estilos que serão empregados em uma página HTML. Para criarmos o CSS é preciso que tenha uma página HTML para receber as características declaradas nos arquivos CSS.

O CSS utiliza diversas características que podem ser declaradas às tags HTML, desde propriedades de dimensão como altura e largura até declarações de alguns eventos como a definição de cor de um link quando o mouse está sobre ele.

A sintaxe do CSS é simples, como o HTML ele possui uma tag de abertura e uma de fechamento também, a tag usada na folha de estilo, ou também chamado de CSS, é a tag `<style>` e seu fechamento é `</style>`. Sabendo assim sua sintaxe, podemos utilizá-la de duas maneiras, internamente em um arquivo HTML ou externamente tendo um arquivo único somente para a codificação do CSS, também chamado de uso externo. Vejamos um exemplo do uso interno e a seguir vamos explicar cada um dos usos.

```
<html>
  <head>
    <style>
      Corpo da codificação CSS
    </style>
    <title> Título da Página HTML </title>
  </head>
  <body>
    Corpo da página HTML
  </body>
</html>
```

É importante saber que o uso do CSS interno na página HTML deve ser feito entre a tag HEAD e seu fechamento, o porquê vamos saber um pouco mais a frente.

CSS Interno

Como falado anteriormente o uso do CSS interno na pagina HTML deve ser feito dentro da tag HEAD simplesmente porque é uma das primeiras partes a serem carregadas pelo browser e colocadas em cache (arquivos temporários salvos localmente no PC de quem acessa a página).

O uso do CSS é simplesmente dizer quais as características que cada tag html irá ter na hora da execução da pagina. Mas para isso temos 3 maneiras diferentes de definir as características das tags. Essas maneiras são: através do próprio nome da TAG; através de uma ID definida em uma tag; ou através de CLASSES definidas num conjunto de tags. Vamos ver agora como funciona cada tipo de identificação.

Identificando características pelo nome da TAG

Podemos usar o nome da tag para definir características a ela, o legal disso é que podemos definir em algumas linhas todas as características desta tag e em todo lugar da minha pagina que minha tag for usada estas características irão se repetir, ou seja, alterando depois apenas estas características toda a minha pagina irá mudar automaticamente onde aquela tag estiver. Antes de explicarmos melhor isso vamos conhecer como definimos no CSS a identificação de uma TAG pelo seu nome.

No CSS identificamos uma tag usando o nome dela sem os sinais de < (menor) e > (maior) e a frente abrimos uma chave ({) depois descrevemos as características e fechamos a chave (}). As chaves servem para delimitar o inicio e o fim das características daquela tag identificada. Vejamos um exemplo para entender melhor:

```
<html>
  <head>
    <style>
      p{
        text-align:center;
      }
    </style>
    <title> Titulo da Pagina HTML </title>
  </head>
  <body>
    <p>Este parágrafo ficará alinhado de forma centralizada por causa da
    programação CSS</p>
    <p>Este parágrafo ficará alinhado de forma centralizada por causa da
    programação CSS</p>
    <p>Este parágrafo ficará alinhado de forma centralizada por causa da
    programação CSS</p>
  </body>
</html>
```

Vimos aqui um exemplo de caracterização do CSS, usando a característica **text-align** definimos que toda tag <p> nessa pagina terá alinhamento centralizado, após definir isso no CSS vemos que na pagina HTML eu omito agora a propriedade *align* da tag <p> que normalmente era utilizada.

Vemos através do exemplo também que as características são feitas por algumas propriedades um pouco diferentes das aprendidas na tag HTML, e que as definições são feitas após digitarmos “ :” (Lê-se dois pontos) e ainda finalizamos uma linha de característica usando um “;” (Lê-se ponto-e-vírgula) . Assim é que funciona a definição de características no CSS.

Para sabermos operar bem algumas características fica indicado o uso do site da Organização Mundial da Linguagem CSS que é o W3C, lá encontraremos todas as características CSS existentes na linguagem juntamente com seus parâmetros e exemplos, segue link para pesquisa <http://www.w3schools.com/css/> , mas como iremos utilizar algumas propriedades vamos descrever aqui abaixo algumas principais características CSS que podemos usar, com suas seguintes explicações:

- **Width:** valor em pixels(px) ou porcentagem(%);
- **Height:** valor em pixels(px) ou porcentagem(%);
- **Border:** valor em pixels(px) / tipo da borda / cor da borda;
- **Top:** valor em pixels(px) ou porcentagem(%);
- **Left:** valor em pixels(px) ou porcentagem(%);
- **Margin:** valor em pixels(px) ou porcentagem(%);
- **Padding:** valor em pixels(px) ou porcentagem(%);
- **Text-align:** tipo do alinhamento;
- **Font-size:** valor em pixels(px) ou points(pt);
- **Font-family:** tipo da fonte;
- **Color:** cor da fonte em inglês ou hexadecimal;
- **Background:** url(“endereço da imagem”);
- **Background-color:** cor do fundo em inglês ou hexadecimal;

Os parâmetros têm as determinadas funções:

- **Width:**define largura
- **Height:**define altura
- **Border:**define a espessura da borda/tipo borda/ cor da borda
- **Top:**define distancia apartir do topo
- **Left:**define distancia apartir do lado esquerdo
- **Margin:**define margem externa ao objeto
 - **Margin-top** – define somente margem do topo
 - **Margin-left** - define somente margem esquerda
 - **Margin-right** - define somente margem direita
 - **Margin-bottom** - define somente margem de baixo
- **Padding:** define margem interna ao objeto
 - **Padding-top** – define somente margem do topo
 - **Padding-left** - define somente margem esquerda
 - **Padding-right** - define somente margem direita
 - **Padding-bottom** - define somente margem de baixo
- **Text-align:** define o alinhamento do texto
- **Font-size:** define o tamanho da fonte
- **Font-family:** define o tipo da fonte
- **Color:** define a cor do texto
- **Background:** define imagem de fundo
- **Background-color:** define a cor de fundo

Identificando características por um ID

Observamos que ao identificar uma característica de uma tag pelo nome dela podemos definir esta característica a todas as tags da pagina. Mas e se eu quiser definir características para somente uma tag especifica através de CSS? Bem respondendo esta pergunta é que iremos aprender a identificar uma tag com um ID.

O ID definirá um codinome a uma tag, fazendo com que no CSS eu declare as informações a este codinome e não mais ao nome da tag, fazendo assim que somente as tags com o codinome especificado irão receber tais características.

Para definir um codinome a uma tag, na hora da definição dessa tag no HTML acrescentamos dentro da tag o parametro "**id**" e atribuímos a ele um nome que iremos usar depois na área do CSS para caracterizar o estilo da tag. Na área do CSS para caracterizar aquele ID usaremos o símbolo do sustenido (#) antes do nome do ID definido na tag. Veja como ficará a definição no exemplo abaixo:

```
<html>
  <head>
    <style>
      #id_usado_no_html{
        text-align:center;
      }
    </style>
    <title> Titulo da Pagina HTML </title>
  </head>
  <body>
    <p>Este parágrafo não ficará centralizado</p>
    <p id="id_usado_no_html">Este parágrafo ficará alinhado de forma
    centralizada por causa da programação CSS através do ID</p>
    <p>Este parágrafo não ficará centralizado</p>
  </body>
</html>
```

Observe no exemplo acima que ao invés de usar o nome da tag, utilizamos um ID fazendo com que somente a tag que continha aquele ID recebesse as definições especificadas no CSS. Dessa forma todas as tags, independentemente de serem ou não parágrafos, que conterem nelas o ID definido terão o alinhamento do texto centralizado.

Porem sintaticamente falando, a definição de ID serve para explicitarmos características referidas para apenas o caso de **UMA** tag, ou seja, se quisermos definir características iguais para um conjunto de tags diferentes então iremos utilizar a definição de classes, que iremos aprender a seguir.

Identificando características de Classes

Como finalizado no tópico anterior se quisermos acrescentar um conjunto de características para um conjunto de tags específicas, iremos utilizar as definições de classes que no CSS indicaremos com a propriedade **CLASS**. As class como o próprio nome diz define uma classe de tags que irão receber um mesmo conjunto de características. Para usarmos isso, vamos definir as class da mesma maneira como definimos os ID's porem com uma diferença, no CSS para atribuímos as características não usaremos o # susinado e sim um "." (ponto) antes do nome da class definida. Vamos ver um exemplo e explicá-lo para entendermos melhor:

```
<html>
  <head>
    <style>
      #id_usado_no_html{
        text-align:center;
      }
      .classe_de_borda_preta{
        border:1px solid Black;
      }
    </style>
    <title> Titulo da Pagina HTML </title>
  </head>
  <body>
    <p class="classe_de_borda_preta">Este parágrafo ficará com uma borda
    preta</p>
    <p id="id_usado_no_html">Este parágrafo ficará alinhado de forma
    centralizada por causa da programação CSS através do ID</p>
    <p class="classe_de_borda_preta">Este parágrafo ficará com uma borda
    preta</p>
    
  </body>
</html>
```

Analisando o exemplo acima vemos que os dois parágrafos e a imagem, ou seja, esse conjunto de tags terão agora uma borda preta de um pixel. Podemos dizer também que aquelas tags pertencem a uma mesma classe por conterem uma mesma característica igual.

Podemos concluir então que os ID's são usados para exceções de tags que irão conter uma determinada característica, e a CLASS serve para definir um conjunto de características iguais a diferentes tipos de tags.

CSS Externo

Vimos as três formas de identificarmos uma tag para usar o CSS, e podemos usar estas mesmas formas de identificação para uma definição de CSS externo também. A diferença básica entre o uso interno e externo é que no interno as características são definidas internamente no arquivo HTML em que são usadas, já no externo nós teremos um arquivo único para as definições do CSS que será salvo com a extensão “.css” e será ligado através de uma tag no arquivo HTML para que as características sejam aplicadas.

Temos uma segunda importância para preferirmos o uso do CSS Externo a o interno. Esta importância está na leitura da página pelo browser, que ao ler o arquivo CSS externo uma vez guardará no cache do computador utilizando assim o processamento local da máquina (que fará a leitura muito mais rápida dos dados) para definir as características para todas as outras páginas que usarem este mesmo arquivo “.css”.No arquivo interno cada vez que a página for acessada ela irá fazer o download de todas as informações CSS e utilizará da banda de internet para buscar as características definidas dentro do arquivo fazendo com que a todo momento que outra página que utilizará as mesmas características, já baixadas num arquivo anterior, sejam baixadas novamente.

Para a criação do arquivo CSS externo, não é preciso definir as tags <style></style> é preciso apenas definir as características como vimos anteriormente através das ID`s, do nome das tags ou através das classes. Este arquivo conterá apenas estas informações não tendo nele nenhum conteúdo HTML. Após definido tudo será salvo com um nome como por exemplo *estilo.css*.

Feito isso agora no arquivo HTML chamaremos este arquivo de estilo colocando entre a tag <head> </head> a seguinte tag:

```
<head>
    <link rel="stylesheet" type="text/css" href="estilo.css" />
</head>
```

A tag anterior definirá que será ligado a este arquivo HTML o arquivo relacionado à folha de estilo do tipo texto CSS que está no caminho definido dentro do parâmetro href e que tem como nome *estilo.css* .

Depois disso você pode salvar normalmente o arquivo HTML e então as características usadas no arquivo CSS serão atribuídas as tags da página HTML. Vamos ver criar um exemplo:

Conteúdo do arquivo que deverá ser salvo com o nome ***estilo.css***:

```
#id_usado_no_html{
    text-align:center;
}
.classe_de_borda_preta{
    border:1px solid Black;
}
```

Conteúdo do arquivo que deverá ser salvo com o nome **índex.html**:

```
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="estilo.css" />
    <title> Titulo da Pagina HTML </title>
  </head>
  <body>
    <p class="classe_de_borda_preta">Este parágrafo ficará com uma borda
    preta</p>
    <p id="id_usado_no_html">Este parágrafo ficará alinhado de forma
    centralizada por causa da programação CSS através do ID</p>
    <p class="classe_de_borda_preta">Este parágrafo ficará com uma borda
    preta</p>
    
  </body>
</html>
```

Ao concluir as instruções e abrir a pagina índex.html no browser você verá que as características definidas no arquivo estilo.css foram aplicadas na pagina mesmo não estando inclusas no arquivo índex.html, mas somente ligadas através da tag link.

Uso de Div`s nos Layouts

Usualmente os programadores de paginas de internet utilizam tabelas para a construção de layouts de paginas de internet, porem não é o uso correto da tag, para isso utilizamos uma tag chamada DIV que é a tag feita especificamente para utilização de blocos de paginas.

Esta tag não possui nenhum parâmetro no HTML porem ela é toda caracterizada através de CSS. Sua sintaxe é simples: <div> </div> e ela funciona como se fosse um quadrado dentro da sua página.

Vamos ver um exemplo de como centralizar uma pagina no browser através de CSS:

```
<html>
  <head>
    <style>
      #geral{
        position:absolute;
        width:400px;
        height:400px;
        margin-left:-200px;
        left:50%;
        border:1px solid black;
      }
    </style>
  </head>
  <body>
    <div id="geral">
    </div>
  </body>
</html>
```

Podemos montar uma pagina com duas colunas, para isso usamos a propriedade float que fará com que uma determinada div ou objeto flutue para alguma direção em uma pagina. Vamos ver o código abaixo que montará um layout com 2 colunas dentro de uma div geral.

```
<html>
  <head>
    <style>
      #geral{
        position:absolute;
        width:400px;
        height:400px;
        margin-left:-200px;
        left:50%;
        border:1px solid black;
      }
      #col1{
        width:200px;
        height:400px;
        background-color:red;
        float:left;
      }
      #col2{
        width:195px;
        height:400px;
        background-color:blue;
      }
    </style>
  </head>
  <body>
    <div id="geral">
      <div id="col1">
      </div>
      <div id="col2">
      </div>
    </div>
  </body>
</html>
```

Com isso finalizamos esta apostila de conceitos básicos de HTML e CSS. Para aprofundar seus conhecimentos pratique bastante e em caso de duvidas você achará diversos modelos na internet.

Links indicados

Tabless - <http://www.tableless.com.br/> - Site desenvolvido com intuito de informar programadores a se educarem ao correto da sintaxe HTML.

W3C School - <http://www.w3schools.com/> - Site do Padronizador oficial das linguagens de programação para internet.

Zen Garden - <http://www.csszengarden.com/> - Site desenvolvido com o mesmo conteúdo, mas com folhas de estilos CSS externas alteráveis e variáveis, alterando toda sua forma apenas com a folha de estilo.

Maujor - <http://maujor.com/> - Site desenvolvido com o intuito de envolver o conhecimento de diversos conceitos e efeitos de CSS