

# **TPI I**

## **Técnicas de Programação para Internet I (HTML/CSS)**



**Professor: LUCIANO CRECENTE**

## Sumário

1. Introdução ao HTML .....	5
O que é HTML? .....	5
Requisitos para o desenvolvimento de uma página WEB .....	6
TAGs / Elementos .....	6
O NotePad++ .....	6
O Ambiente do NotePad++ .....	7
Criar um Documento Novo .....	7
Salvar Documento .....	7
As primeiras Tags .....	8
Criando a Primeira Página .....	8
Exibindo a Página .....	9
Tags de Título e de Paragrafo .....	9
Comentários em HTML .....	11
TAG marquee .....	11
HTML5 .....	12
Diferenças entre HTML5 e HTML4 .....	12
Sintaxe HTML4 .....	12
Sintaxe HTML5 .....	13
CSS – Folhas de Estilo .....	14
Tipos de CSS .....	15
Estilo inline .....	15
Folha de Estilo Incorporada .....	16
A propriedade font .....	16
Color .....	17
Font-family .....	17
Font-size: .....	17
font-style: .....	17
font-variant: .....	17
A propriedade text .....	18
Color: .....	18
Text-align: .....	19
Text-decoration: .....	19
Text-indent: .....	19

text-transform: .....	19
A propriedade background.....	19
background-color .....	20
background-image.....	20
background-repeat .....	20
background-attachment.....	21
background-position .....	21
Inserindo Imagens.....	22
A propriedade Float.....	23
Folha de Estilo Externa .....	24
Import .....	25
Listas .....	26
Listas não ordenadas.....	26
Listas ordenadas ou numerada.....	26
Listas de definição.....	27
Estilos para Listas .....	28
Propriedade Border .....	29
color .....	30
style.....	30
width .....	31
Margin e Padding .....	33
As propriedades para margens.....	33
As Propriedades Padding .....	35
Hyperlinks HTML (Links) .....	37
Sintaxe HTML Link .....	37
O Atributo Target .....	37
Classes e ID .....	39
O Selector ID .....	39
O Seletor de Class .....	40
Quando usar Classe e quando usar ID.....	40
Pseudo-elementos CSS: .....	41
Before e after.....	42
Pseudo-classe para links .....	42
Menus com Listas.....	43

Div e Span .....	45
Bordas Arredondadas com CSS .....	50
Sombras com CSS .....	52
Trabalhando com Tabelas .....	55
Mesclando Células com ColSpan .....	57
Mesclando Celulas com RowSpan .....	58

## 1. Introdução ao HTML

### O que é HTML?

HTML significa **HyperTextMarkupLanguage** e é a linguagem de marcação e hipertexto, ou seja, é a linguagem na qual são escritas as páginas da Web, interpretada pelo navegador.

As informações estão ligadas na forma de páginas ligadas entre si. A página é transferida de um computador remoto para o usuário, onde o **browser** (navegador) faz o trabalho de interpretar os códigos naquele documento e mostra a página que o usuário vê. A Web está estruturada em dois princípios básicos: Hiper Text Transfer Protocol (HTTP) e Hiper Text Markup Language (HTML).

O protocolo de transferência de hipertexto (HTTP), ou seja, é o protocolo que permite a navegação na Web, com o simples clicar do mouse sobre o texto (ou imagem) que esteja associado a outro link.

**A web foi inventada em 1992 Sir Tim Berners-Lee** com a criação do HTML, usado para a comunicação e disseminação de pesquisas entre ele e seu grupo de colegas, e assim estavam criadas as noções *web* e *links* como são conhecidos atualmente. O HTML ficou bastante conhecido quando começou a ser utilizado para formar a rede pública daquela época, o que se tornaria mais tarde a internet que conhecemos hoje.

**O HTML é uma linguagem fácil de aprender**, o que permite que qualquer pessoa, mesmo que nunca tenha programado possa enfrentar a tarefa de criar um website. O HTML é fácil e em pouco tempo poderemos dominar sua linguagem. Mais adiante, conseguiremos os resultados profissionais graças as nossas capacidades para o desenho e a nossa veia artística.

Esta linguagem se escreve em um documento de texto, por isso necessitamos de um editor de textos para escrever uma página web. Sendo assim, o arquivo onde está contido o código HTML é um arquivo de texto com uma peculiaridade, que tem uma **extensão .html ou .htm** (é indiferente qual deles utilizar). De modo que quando programarmos em HTML, o faremos com um editor de texto, o mais simples possível e salvaremos nossos trabalhos com extensão .html, por exemplo: minhapagina.html.

**Requisitos para o desenvolvimento de uma página WEB**

- Conhecer a linguagem HTML para escrever o código fonte de sua página;
- Editor de texto para gerar o seu código fonte (bloco de notas, notepad++, Dremweaver, entre outros);
- Um navegador de internet (browser) para visualizar as suas páginas (Chrome, Firefox, Safari, etc).

**TAGs / Elementos**

Os comandos HTML são chamados de TAGs, compreendem de marcas padrões que são utilizadas para fazer indicações a um browser.

Assim como em outras linguagens, os comandos têm uma sintaxe própria, e seguem algumas regras:

- As TAGs aparecem sempre entre sinais de “menor que” (<) e “maior que”(>);
- Geralmente são utilizadas aos pares, sendo que a TAG de finalização de um comando qualquer é finalizada com a precedência de uma barra (/).

Exemplos:

TAG único:

```
<br>
```

TAG duplo:

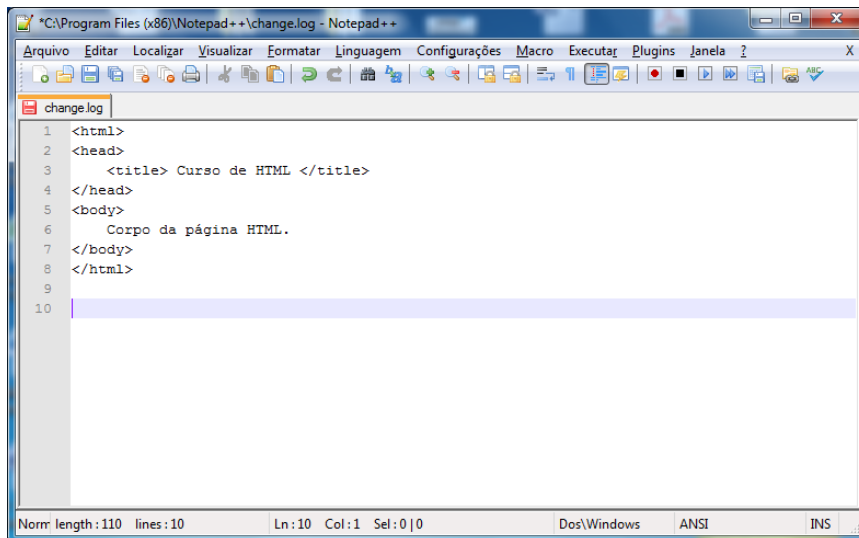
```
<center>Texto </center>
```

**O NotePad++**

Notepad++ é um editor de texto e códigos fonte completo que suporta as mais diversas linguagens de programação e, sendo assim, uma ótima alternativa ao bloco de notas. Com suporte às linguagens C, C++, Java, C#, XML, HTML, PHP, Javascript, etc.

Possuindo muitas características favoráveis para facilitar o trabalho, como auto completar, sistema de busca e substituição, interface funcional e navegação por abas.

## O Ambiente do NotePad++



## Criar um Documento Novo

Arquivo -> Novo Ou Clique no Botão Novo

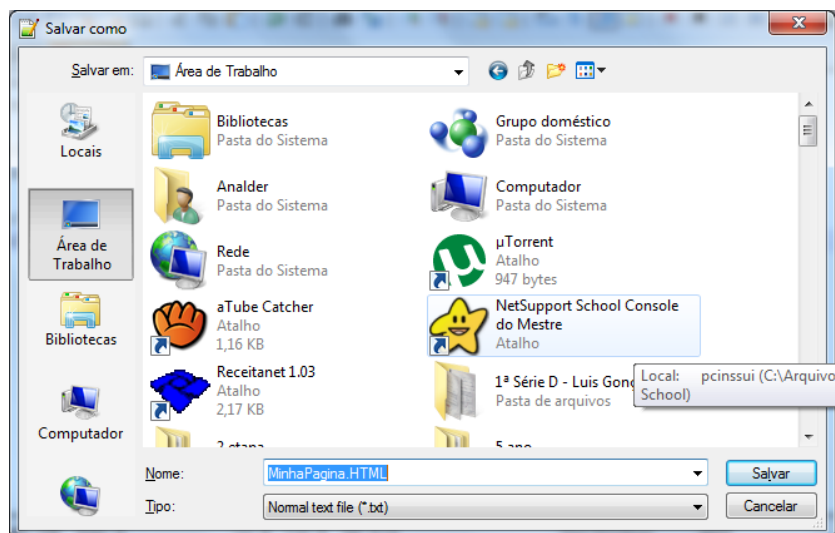


## Salvar Documento


Arquivo-> Salvar ou clique no Botão



Depois selecione onde deseja salvar e no campo nome você deve inserir o nome seguido da extensão .html, depois clique no botão salvar.



Repare que ao salvar o documento como html as tags ficarão na cor azul.

Para cada tag aberta e fechada também surgirá um símbolo(  ), ele deverá ser usado para ocultar ou expandir o conteúdo da tag, como no exemplo abaixo:

Expandido:

```
<head>
  <title> Curso de HTML </title>
</head>
<body>
  Corpo da página HTML.
</body>
</html>
```

Oculto:

```
<html>
<head>
<body>
  Corpo da página HTML.
</body>
</html>
```

Este recurso facilita a visualização do código em determinados momentos.

## As primeiras Tags

Todo o documento HTML fica contido entre os TAGS: <HTML> e </HTML>. Uma página HTML possui três partes básicas: estrutura principal, um cabeçalho e um corpo de página.

Possui uma estrutura dividida em 4TAGs básicas:

**<HTML> ...</HTML>**São usados para delimitar os comandos HTML, indicam o início e o fim de um documento.

**<HEAD> ...</HEAD>**Usado para indicar parâmetros de configuração do documento, também utilizado para exibir o título na barra de títulos do browser.

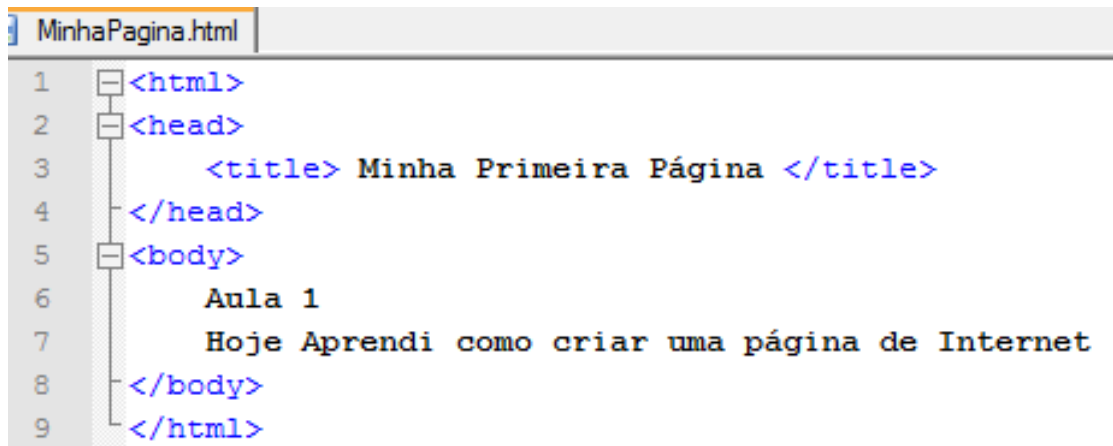
**<TITLE> ...</TITLE>**Indica o título do documento para o browser. Esta TAG deve estar sempre dentro das TAGs<HEAD></HEAD>.

**<BODY> ...</BODY>**Envolvem a seção de corpo do documento. Aqui fica o conteúdo principal da Home Page.

## Criando a Primeira Página

Para compreender o que aprendemos até aqui, vamos criar a sua primeira página de Internet, para isso abra o notepad++ e crie o documento a seguir, crie uma pasta chamada Aula1 e salve-o como MinhaPagina.html.

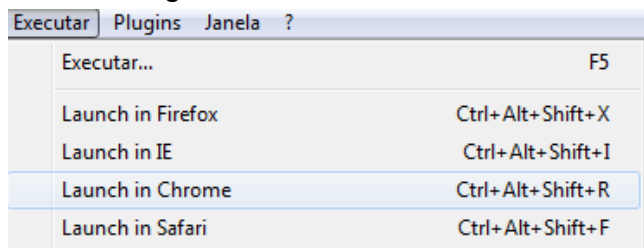




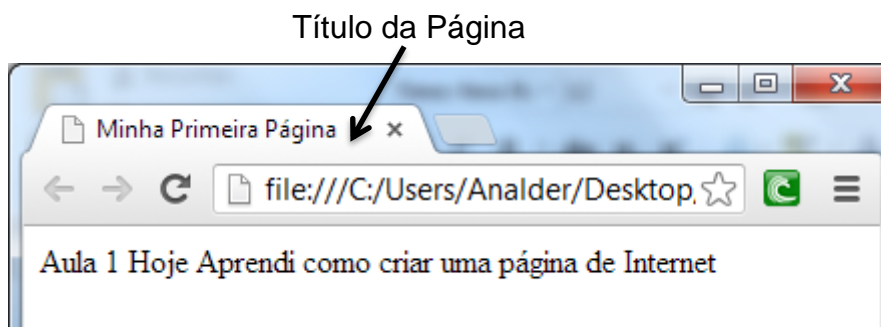
```
1 <html>
2 <head>
3     <title> Minha Primeira Página </title>
4 </head>
5 <body>
6     Aula 1
7     Hoje Aprendi como criar uma página de Internet
8 </body>
9 </html>
```

### Exibindo a Página

Para visualizar a sua página no navegador, vá até o menu executar e escolha em qual navegador você quer visualizar a sua página, você também usar as teclas de atalho conforme ilustração ao lado.



### Resultado



Você deve ter percebido que apesar de ter digitado em duas linhas, tudo ficou apenas em uma linha, para resolver este problema teremos que trabalhar com títulos e parágrafos, para garantir que cada item ocupe seu espaço.

### Tags de Título e de Paragrafo

<P>a TAG responsável pela quebra de parágrafos é a <P>.

#### Sintaxe do comando:

<P>Este é um texto no paragrafo.</P>

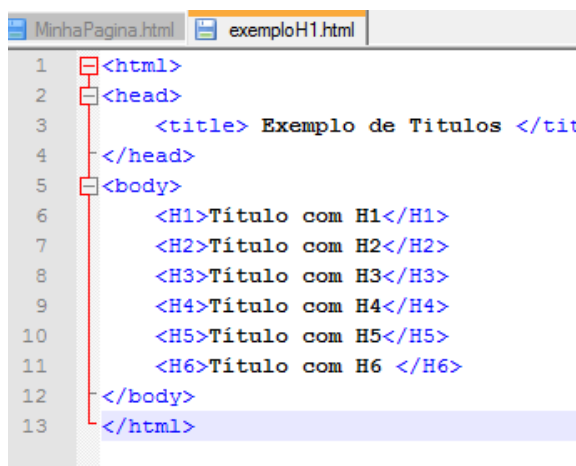
**<BR>** Com esta tag você pode inserir uma quebra de linha dentro de um texto.

**<H1>** É utilizado para demarcar títulos, além do <H1> temos outros níveis de subtítulos, sendo: <H2><H3><H4><H5><H6>

**Sintaxe do comando:**

< H1>Este é um Título com H1.</H1>

Para visualizar melhor como funcionam os títulos crie um novo arquivo e faça conforme exemplo abaixo, salve como exemplo H1:

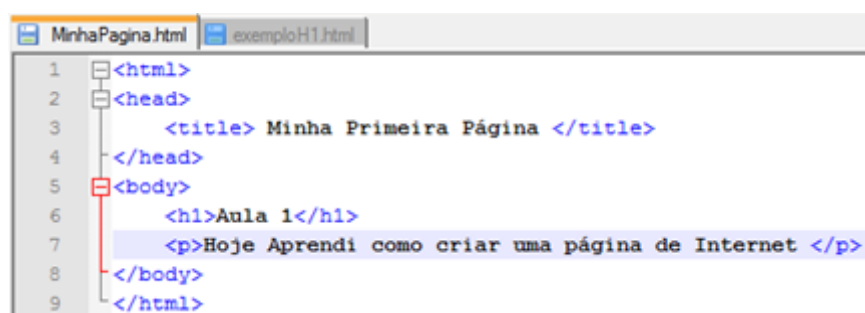


```
1 <html>
2 <head>
3   <title> Exemplo de Titulos </tit
4 </head>
5 <body>
6   <H1>Título com H1</H1>
7   <H2>Título com H2</H2>
8   <H3>Título com H3</H3>
9   <H4>Título com H4</H4>
10  <H5>Título com H5</H5>
11  <H6>Título com H6 </H6>
12 </body>
13 </html>
```

**Resultado**

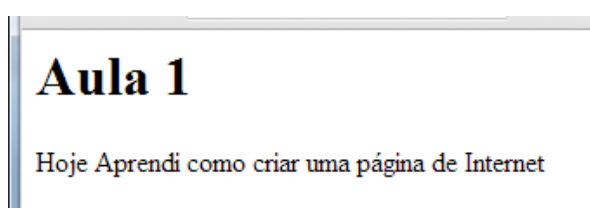


Agora vamos voltar ao nosso arquivo MinhaPagina.html e criar títulos e parágrafos na página:



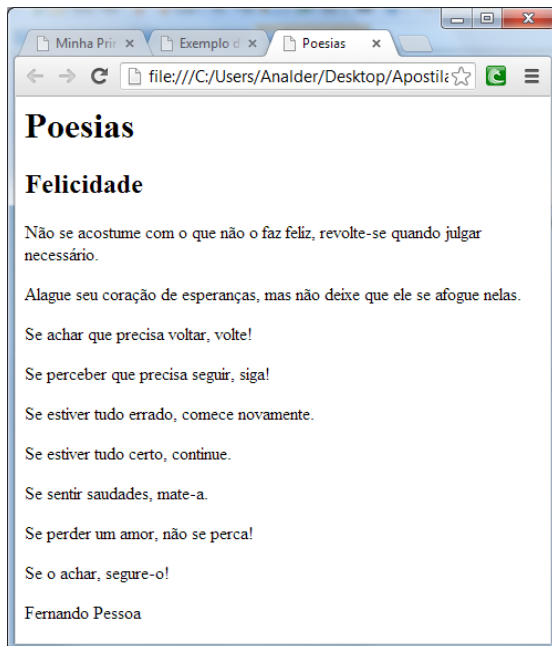
```
1 <html>
2 <head>
3   <title> Minha Primeira Página </title>
4 </head>
5 <body>
6   <h1>Aula 1</h1>
7   <p>Hoje Aprendi como criar uma página de Internet </p>
8 </body>
9 </html>
```

**Resultado**



## Exercício

1. Usando os recursos aprendidos até aqui, crie a página abaixo:



## Comentários em HTML

Comentários são partes do código que deverão ser ignoradas pelo navegador na hora de criar a página web.

```
<p>Se o achar, segure-o!</p>
<p>Fernando Pessoa</p>
<!-- Insira seu comentário Aqui!-->
</body>
```

No notePad++ os comentários ficarão na cor verde.

## TAG marquee

O texto fica passando pela tela, no sentido horizontal.

Ex: <marquee>texto</marquee>

## HTML5

Com a criação da HTML versão 5 (HTML5) o Grupo de Trabalho da HTML, pertencente ao *World Wide Web Consortium* (W3C), divulgaram em novembro de 2007 um conjunto de diretrizes que visam orientar o desenvolvimento da HTML5 nas áreas:

**Compatibilidade:** ser compatível com conteúdos existentes de outras versões da HTML.

**Utilidade:** visa garantir que a HTML possa ser usada para todos os fins que ela se destina.

**Interoperabilidade:** visa aumentar as chances de uma aplicação HTML funcionar nos mais variados dispositivos e sistemas.

**Acesso Universal:** garantir o acesso às funcionalidades da HTML5 pelo maior número possível de dispositivos e sistema

## Diferenças entre HTML5 e HTML4

As principais diferenças têm origem no propósito da HTML5 vir para reformular tanto a HTML dos anos 90 e quanto o XHTML, que foi uma tentativa frustrada de reformular a HTML4 como uma aplicação XML.

## Sintaxe HTML4

Observe a seguir um exemplo de marcação HTML:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html lang="pt-br">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <title>Título da página</title>
    <link rel="stylesheet" type="text/css" href="/estilos/main.css">
  </head>
  <body>
    <h1>Minha página HTML4</h1>
  </body>
</html>
```

## Sintaxe HTML5

A seguir, a marcação HTML5 compatível com a sintaxe HTML:

```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <meta charset="utf-8">
    <title> Título da página </title>
    <link rel="stylesheet" href="/estilos/main.css">
  </head>
  <body>
    <h1>Minha página HTML5</h1>
  </body>
</html>
```

**DOCTYPE** – em HTML5 foi simplificado, não havendo mais necessidade do link. A sintaxe para DOCTYPE é **case insensitive** (não distingue minúsculas de maiúsculas).

**Codificação de caracteres** – em HTML5 a codificação de caracteres com uso do elemento *meta* foi simplificada, não necessitando dos atributos *http-equiv* e *content*.

**Elemento link** – o atributo *type* para links a folha de estilos e para scripts é dispensável, podendo ser usada de forma opcional.

## CSS – Folhas de Estilo

CSS é a sigla para CascadingStyleSheets que em português foi traduzido para folhas de estilo em cascata e nada mais é, que um documento onde são definidas regras de formatação ou de estilos, a serem aplicadas aos elementos da marcação de um documento HTML.

A finalidade das CSS é a de retirar da marcação toda e qualquer declaração que vise a apresentação do documento. Isto significa dizer que tags do tipo <font>, <b>, <i> bem como uso de colunas e linhas de tabelas para obtenção de espaçamentos não são admitidos ou admitidos com restrições em um projeto Web com CSS. Vale dizer simplificando: HTML para estruturação e CSS para apresentação.

CSS permite que você retire da marcação HTML toda a formatação (apresentação) do documento WEB. Quem vai determinar cores, formas, tipos e tamanhos, posicionamento e em fim todo o "visual" da página são as CSS.

Entre as Principais vantagens do uso das CSS estão:

- controle total sobre a apresentação do site a partir de um arquivo central;
- agilização da manutenção e redesign do site;
- saída para diferentes tipos de mídia a partir de uma versão única de HTML;
- redução do tempo de carga dos documentos Web;
- adequação simplificada aos critérios de acessibilidade e usabilidade;
- elaboração de documentos consistentes com as aplicações de usuários futuras;
- aumento considerável na portabilidade dos documentos Web.

As CSS estão entre as práticas vivamente recomendadas pelo W3C para projetos Web. Os esforços dos órgãos normatizadores apontam no momento atual para elaboração de documentos Web acessíveis, usáveis e portáteis com grande ênfase. E, CSS facilita e simplifica a obtenção destas três variáveis.

Uma regra CSS é uma unidade mínima de programação de estilos, que segue uma sintaxe própria e destina-se a estilizar uma ou mais propriedades

CSS. A sintaxe de uma regra CSS compreende um seletor uma propriedade e um valor escritos como mostrado abaixo:

**seletor {propriedade: valor;}** Exemplo: `p {text-indent:12px;}`

**p** é o seletor CSS.

**{text-indent: 12px}** é a declaração CSS.

**text-indent-** é a propriedade CSS.

**12px** - é o valor CSS.

Todas as propriedades CSS têm um valor inicial (default), ou valor padrão, quase nenhuma regra CSS estabelece o valor para esta propriedade, este valor será aplicado. Existe uma folha de estilos nativa no navegador para ser aplicada quando não for definida pelo desenvolvedor, para o documento renderizado, uma folha de estilo.

Para que você tenha uma ideia melhor de como funciona as CSS na prática, vá até o site <http://www.csszengarden.com/>, neste site, você poderá alterar a folha de estilo da página e perceber assim a mudança que ocorre na página apenas com a troca de folha de estilo, embora pareça que a página foi alterada, na verdade a página é a mesma, o que mudou foi a folha de estilo do site.

## Tipos de CSS

As CSS podem ser salvas em três lugares diferentes no seu site, dentro da própria tag HTML a estilizar, dentro da tag `<style>` colocada na seção head do documento HTML a estilizar ou em um arquivo independente salvo com a extensão `.css`, a elas damos o nome respectivamente de estilo **inline**, folha de estilo **incorporada** e folha de estilo **externa**, veremos agora cada uma delas.

### Estilo inline

Chamamos de estilo é inline quando as regras de estilo são declaradas dentro da própria tag HTML, com o uso do atributo `style`. As regras de estilo são aplicadas exclusivamente naquele elemento HTML. Este tipo de declaração retira toda a flexibilidade das CSS uma vez que você terá que agir diretamente na tag HTML se desejar fazer uma mudança de estilos, seu uso deve ser evitado ou no máximo restrito a um ou poucos documentos.

Exemplo:

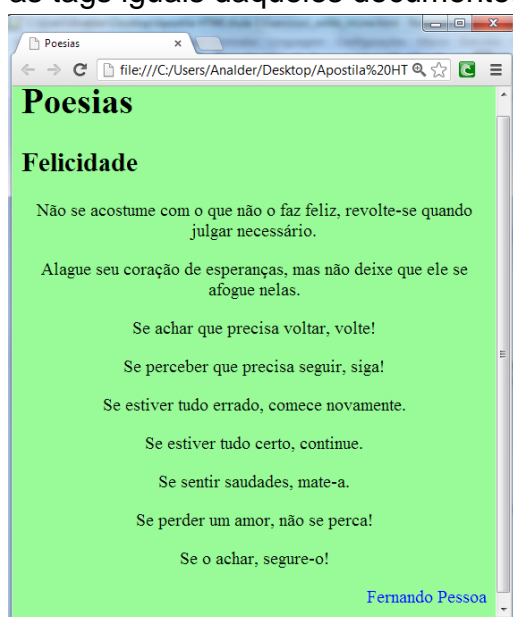
```
21 <p style="text-align:right; color:blue;" >Fernando Pessoa</p>
```

## Folha de Estilo Incorporada

Folha de estilo incorporada é uma folha de estilo anexada a um só documento HTML. As regras de estilo são válidas tão somente para aquele documento específico. A folha de estilo é especificada dentro da seção head do documento e declarada como conteúdo da tag <style>.

```
4 <style>
5   body{
6     background-color:#98FB98;
7   }
8
9   p{text-align:center}
10
11 </style>
```

Ao contrário do inline, ao criar uma estilo para uma determinada tag, todas as tags iguais daqueles documentos serão afetadas pelo estilo, veja o exemplo:



## A propriedade font

As propriedades para as fontes, definem as características (os valores na regra CSS) das letras que constituem os textos dentro dos elementos HTML.

As propriedades básicas para fontes são:

- color:.....cor da fonte
- font-family:.....tipo de fonte
- font-size:.....tamanho de fonte
- font-style:.....estilo de fonte
- font-variant:.....fontes maiúsculas de menor altura
- font-weight:.....quanto mais escura a fonte é (negrito)



- font:.....maneira abreviada para todas as propriedades

## Valores válidos para as propriedades da fonte

### Color

- código hexadecimal: #FFFFFF
- código rgb: rgb(255,235,0)
- nome da cor: red, blue, green...etc

Você encontra facilmente tabelas de cores com código hexadecimal, basta fazer uma pesquisa na internet, se preferir acesse diretamente o site: <http://www.mxstudio.com.br/Conteudos/Dreamweaver/Cores.htm>, lá você encontrará um formato de tabela de cores, mas há uma infinidade de páginas com o mesmo propósito.

Quando uma cor hexadecimal for os 6 dígitos iguais basta você repetir 3 vezes o dígito, por exemplo #FFFFFF é igual a #FFF.

### Font-family

- nome da família de fontes : define-se pelo nome da fonte,
- p.ex: "verdana", "helvetica", "arial", etc.
- nome da família genérica: define-se pelo nome genérico,
- p.ex: "serif", "sans-serif", "cursive", etc.

A propriedade font-family é usada para definir uma lista de tipos de fontes, podendo portanto usar mais de uma família, o browser assume o primeiro nome que ele reconhece na lista. Deve-se separar cada nome por uma vírgula e sempre prever um nome genérico, se o nome da fonte for composto (mais de uma palavra, p. ex: ComicSans MS), usar aspas duplas no nome.

### Font-size:

- xx-small, x-small, small, medium, large, x-large, xx-large, smaller, larger
- Uma medida reconhecida pelas CSS (px, pt, em, cm, ...)
- %

### font-style:

- normal: fonte normal na vertical
- italic: fonte inclinada
- oblique: fonte oblíqua

### font-variant:

normal: fonte normal

small-caps: transforma em maiúsculas de menor altura

Vejamos a aplicação destas propriedades no elemento h1 da nossa página de poesia:

```
h1{
  text-align:center;
  color:#9400D3;
  font-family:"Comic Sans MS", "sans-serif", "cursive";
  font-style:italic;
  Font-size: 30px;
  font-variant: small-caps;
}
```

Resultado:



### Exercícios:

1. Aplique as propriedades de fonte nos outros elementos da página de poesias (h2, p)

### 2. ATIVIDADE PARA ENTREGA:

Organizar uma página com estrutura **HTML5** e CSS no estilo **INCORPORADA** `<style></style>`. Na seção `<BODY>` constar as formatações para `<p>` e `<h1>`.

Atividade em dupla e entrega manuscrito em folha de caderno.

## A propriedade text

As propriedades para textos, definem as características (os valores na regra CSS) dos textos inseridos dentro dos elementos HTML.

**As propriedades para textos são as listadas abaixo:**

- color.....cor do texto;
- letter-spacing.....espaçamento entre letras;
- word-spacing.....espaçamento entre palavras;
- text-align.....alinhamento do texto;
- text-decoration.....decoração do texto;
- text-indent.....recuo do texto;
- text-transform.....forma das letras;
- direction.....direção do texto;
- white-space.....como o browser trata os espaços em branco;

## Valores válidos para as propriedades do texto

### Color:

- código hexadecimal: #FFFFFF
- código rgb: rgb(255,235,0)

- nome da cor: red, blue, green...etc

**Text-align:**

- left: alinha o texto a esquerda
- right: alinha o texto a direita
- center: alinha o texto no centro
- justify: força o texto a ocupar toda a extensão da linha da esquerda a direita

**Text-decoration:**

- none: nenhuma decoração
- underline: coloca sublinhado no texto
- overline: coloca um sobrelinhado no texto
- line-through: coloca uma linha em cima do texto
- blink: faz o texto piscar

**Text-indent:**

- Uma medida reconhecida pelas CSS (px, pt, em, cm, ...)
- % : porcentagem da largura do elemento pai

**text-transform:**

- none: texto normal
- capitalize: todas as primeiras letras do texto em maiúsculas
- uppercase: todas as letras do texto em maiúsculas
- lowercase: todas as letras do texto em minúsculas

Vamos aplicar algumas propriedades de Texto ao elemento h2:

```
18 h2{
19   color:#228B22;
20   text-align:center;
21   text-decoration:overline;
22   text-transform: uppercase;
23   font-family:"Comic Sans MS", "sans-serif", "cursive";
24   font-style:italic;
25   Font-size: 25px;
26 }
```

Resultado:



## A propriedade background

A propriedade background define as características (os valores na regra CSS) do fundo dos elementos HTML.

**As propriedades background são as listadas abaixo**

- background-color..... cor do fundo;
- background-image..... imagem de fundo;
- background-repeat..... maneira como a imagem de fundo é posicionada;
- background-attachment.....se a imagem de fundo "rola" ou não com a tela;
- background-position.....como e onde a imagem de fundo é posicionada;
- background.....maneira abreviada para todas as propriedades;

**Valores válidos para as propriedades do fundo****background-color**

- código hexadecimal: #FFFFFF
- código rgb: rgb(255,235,0)
- nome da cor: red, blue, green...etc
- transparente: transparent

**Exemplo:** background-color:#66CDAA;

**background-image**

- URL: url(caminho/imagem.gif)

**Exemplo:** background-image:url("imagens/fundo.png");

**background-repeat**

- não repete: no-repeat
- repete vertical e horizontal: repeat
- repete vertical: repeat-y
- repete horizontal: repeat-x

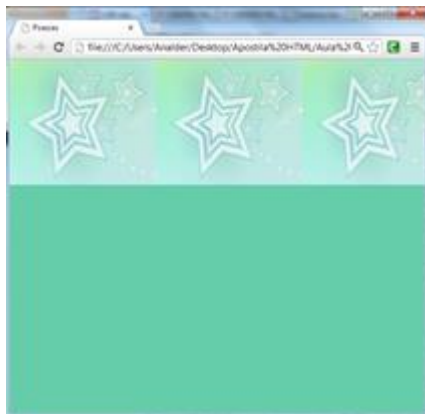
**Exemplo:**



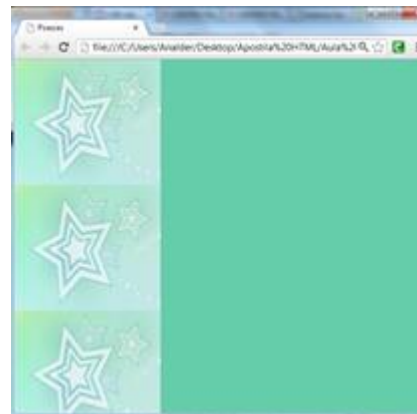
background-repeat: repeat;



background-repeat: no-repeat;



background-repeat: repeat-x;



background-repeat: repeat-y;

### background-attachment

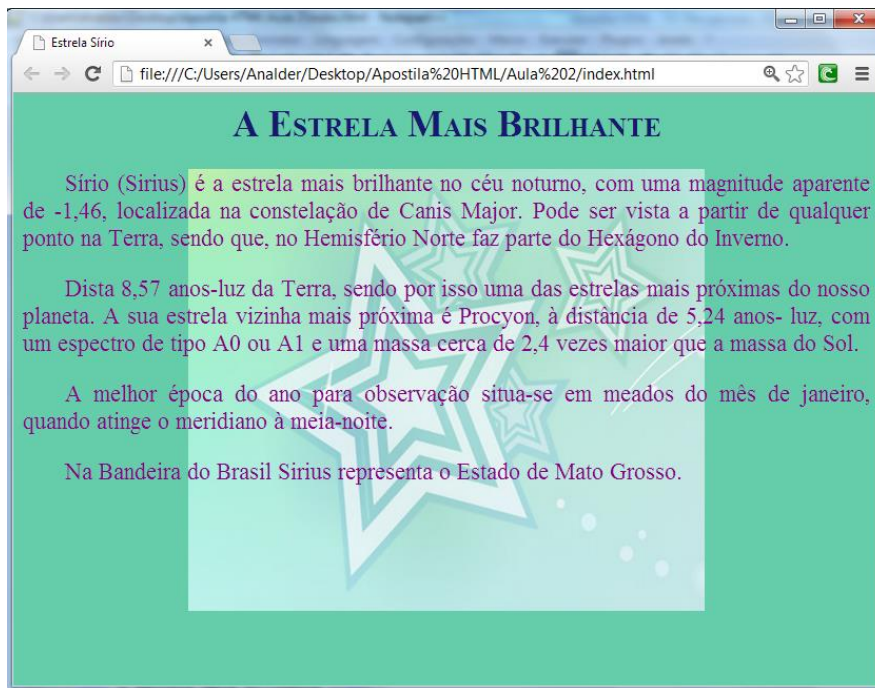
- imagem fixa na tela: fixed
- imagem "rola" com a tela: scroll

### background-position

- x-pos y-pos
- x-% y-%
- top left
- top center
- top right
- center left
- center center
- center right
- bottom left
- bottom center
- bottomright

### Exercício:

1. Usando o que foi aprendido até aqui crie a página abaixo:

**Parâmetros:**

Fundo: (Cor: #66CDAA; Imagem: fundo.png, Sem Repetir, Alinhada no Centro, Fixa)

H1:( Color:#191970; Alinhamento do Texto: Centro, Variant: Small-Caps)

P:(Cor: :#8B008B, Alinhamento do Texto: Justificado, Tamanho: 20px, Tabulação: 5%)

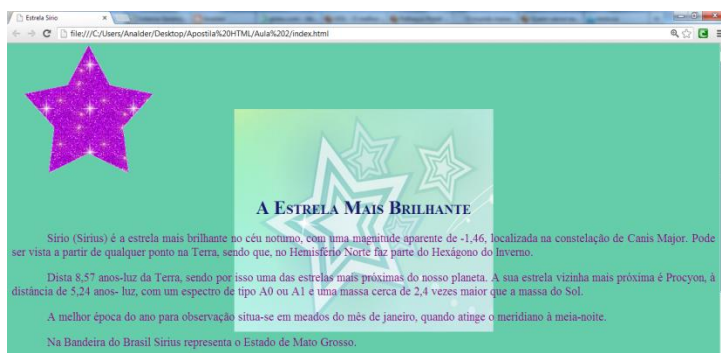
**Inserindo Imagens**

Para inserir uma imagem em uma página web utilizamos o comando, tag<img> e seus atributos. Os formatos mais usados são o GIF e o JPG, ambos com compactação de pixels.

**Sintaxe:**



Exemplo:



## A propriedade Float

Com float, um elemento pode ser empurrado para a esquerda ou para a direita, permitindo que outros elementos possam ficar torno dele. O float é muito usado para as imagens, mas também é útil quando se trabalha com layouts.

Elementos são flutuados na horizontal, isto significa que um elemento só pode ser lançado para a esquerda ou para a direita, não para cima ou para baixo. Desta forma os valores válidos para a propriedade Float é left ou right.

Veja os resultados:

`img{ float: left; }`



`img{ float: right; }`



## Exercício

1. Crie com o que você aprendeu até agora a página abaixo:



Salve o arquivo em uma pasta chamada EXE\_ETEC com o nome index.html.



## Folha de Estilo Externa

Até aqui trabalhamos apenas com a folha de estilo interna, ou seja, a formatação da página, embora realizada com CSS estava incluída no próprio arquivo HTML, porém o ideal é que ao separarmos estrutura (HTML) e formatação (CSS) também façamos a separação dos arquivos, isso fará com que os códigos possam ser reaproveitados, de forma que eu possa usar um mesmo CSS para várias páginas HTML ou alterar facilmente a formatação de uma página HTML apenas alterando o arquivo de CSS.

Vamos usar exemplos usando o exercício anterior, para isso você deverá criar dentro da pasta EXE\_ETEC criado no exercício anterior um arquivo chamado estilo.css, não esqueça da extensão do arquivo (.CSS) é esta extensão que irá garantir que meu arquivo seja uma folha de estilo.

Neste arquivo digite:

```
1 body{text-indent:1.25cm}
2
3 h1, h3{ color: #B22222; }
4
5 img{float: left;}
6
```

Neste estilo estamos na primeira linha criando uma tabulação de 1,25cm em todos os textos da página, na segunda linha estamos alterando a cor dos títulos h1 e h3 e na terceira linha estamos deixando o float da imagem a esquerda.

Se você executar a sua página neste momento você não conseguirá visualizar as alterações, isto porque ainda não fizemos o link entre o html e o CSS.

Quando trabalhamos com folha de estilo externa, como o nome diz, trabalhamos com um arquivo externo, ou seja, teremos a estrutura HTML e o estilo CSS em arquivos independentes, sendo assim é preciso no arquivo HTML fazer o link com o arquivo CSS para que a formatação seja executada, para isso, dentro do cabeçalho devemos inserir a seguinte linha:

```
3 <head>
4   <title> ETEC Matão </title>
5
6   <link rel="stylesheet" type="text/css" href="estilo.css">
7
8 </head>
```

Desta forma estamos linkando o arquivo estilo.css com o nosso arquivo html, neste momento, se executarmos novamente a nossa página já veremos os efeitos do estilo que acabamos de criar.



É importante lembrar que no atributo href é necessário inserir o endereço do arquivo css, no exemplo acima o arquivo “estilo.css” encontra-se no mesmo diretório do arquivo html, e por isso não é necessário fornecer o caminho completo do arquivo.

## Import

Outra forma de usar estilo externo é fazendo o import deste estilo, para isso usaremos a tag <style> dentro do cabeçalho e dentro dela faremos o import, como no exemplo:

```
3 <head>
4   <title> ETEC Matão </title>
5
6   <style type="text/css">
7       @import url("estilo.css");
8   </style>
9
10 </head>
```

As duas formas produzirão o mesmo resultado, ou seja, o estilo será aplicado na pagina html da mesma forma, mas usando o import o processamento da informação poderá ser um pouco mais lento, pois quando se usa o import apenas um arquivo é carregado por vez, já no caso do link, serão carregados paralelamente, o que permite um carregamento mais rápido.

Agora que o estilo já está linkado a página html, sua página deverá ficar assim:



## Listas

Lista é um importante recurso de HTML, pois permite criarmos tópicos de textos para uma melhor exemplificação de um determinado assunto. São recursos extremamente usados, inclusive quando nem imaginamos que ele está sendo usado, como no caso de menus. Hoje, boa parte dos menus em HTML é feito com listas.

### Há três tipos de listas:

- Não ordenada
- Ordenada ou numerada
- De definição - usada para fazer comentários sobre os itens expostos

### Listas não ordenadas

As listas não numeradas são usadas para listar itens, sem se preocupar com sua sequência. Chamamos de lista de marcadores apenas.

As tags usadas para criar uma lista não ordenada são `<ul>` e `<li>`. A tag `<ul>` é usada para definir a lista e a tag `<li>` é usada para cada item da lista.

### Exemplo

```
<ul>  
<li>Internet Explorer</li>  
<li>Opera</li>  
<li>Firefox</li>  
<li>Safari</li>  
</ul>
```

- Internet Explorer
- Opera
- Firefox
- Safari

### Listas ordenadas ou numerada

As listas ordenadas ou numeradas são usadas para indicar alguma sequência ou numeração.

As tags usadas para criar uma lista ordenada são `<ol>` e `<li>`. A tag `<ol>` é usada para definir a lista e a tag `<li>` é usada para cada item da lista.

### Exemplo

```

<ol>
<li>São Paulo</li>
<li>Rio de Janeiro</li>
<li>Belo Horizonte</li>
<li>Brasília</li>
</ol>

```

### Listas de definição

Listas de definição são usadas em assuntos onde há um termo a em sequência a sua definição, estilo perguntas e respostas.

As tags usadas são <dl>, <dt> e <dd>.

- <dl>definitionlist - cria a lista de definição
- <dt>definitionterm - onde o termo é proposto
- <dd>definitiondefinition - onde o termo é definido

### Exemplo

```

<dl>
<dt>HTML</dt>
<dd>Linguagem de marcação de texto</dd>
<dt>CSS</dt>
<dd>Usado para formatação de documentos</dd>
</dl>

```

HTML  
Linguagem de marcação de texto  
CSS  
Usado para formatação de documentos

### Exercício

1. Ainda na página da ETEC em que estamos trabalhando, crie a seguinte lista não ordenada:

Etec tem contribuído para o desenvolvimento de Matão e região, forman

#### Cursos Oferecidos

- Administração (EAD)
- Automação Industrial
- Eletrônica
- Enfermagem
- Finanças
- Informática
- Informática para Internet
- Manutenção e Suporte em Informática
- Mecânica
- Mecatrônica

## Estilos para Listas

A propriedade list define as características (valores) das listas HTML.

### As propriedades list são as listadas abaixo:

- list-style-image..... imagem como marcador da lista;
- list-style-position.....onde o marcador da lista é posicionado;
- list-style-type.....tipo do marcador da lista;
- list-style.....maneira abreviada para todas as propriedades;

### Valores válidos para as propriedades do lista

#### list-style-image

- none
- URL: url(caminho/marcador.gif)

#### list-style-position

- outside: marcador fora do alinhamento do texto
- inside: marcador alinhado com texto

#### list-style-type

O list-style-type pode ser usado quando não for usado imagens nos marcadores, pode conter o valor none para não exibir nenhum marcador, ou um dos marcadores disponíveis, sendo :disc (bolinha cheia), circle(bolinha vazia), square(quadrado cheio), decimal (números 1, 2, 3, 4), decimal-leading-zero, lower-roman ( romano minúsculo i, ii, iii, iv), upper-roman( romano maiúsculo I, II, III, IV), lower-alpha( letra minúscula a, b, c, d), upper-alpha(letra maiúscula A, B, C, D), lower-greek, lower-latin, upper-latin, Hebrew, Armenian, Georgian, cjk-ideographic, hiragana, katakana, hiragana-iroha, katakana-iroha.

### Exemplo: Regra criado no CSS.

```
ul {  
  list-style-image: url("seta.png");  
  list-style-position:inside;  
  list-style-type:circle;  
}
```

## Exercício

1. Formate a Lista criada no exercício anterior para que ela fique como o exemplo:

### Cursos Oferecidos

- Administração (EAD)
- Automação Industrial
- Eletrônica
- Enfermagem
- Finanças
- Informática
- Informática para Internet
- Manutenção e Suporte em Informática
- Mecânica
- Mecatrônica

2. Coloque a figura muro.jpg como fundo da página, deixe ela se repetir por toda a página.

## Propriedade Border

As propriedades para as bordas, definem as características (os valores na regra CSS) das quatro bordas de um elemento HTML.

As propriedades para as bordas são as listadas abaixo:

- border-width:.....espessura da borda
- border-style:.....estilo da borda
- border-color:.....cor da borda

## Exemplo:

```
p {  
  border-width: 2px;  
  border-style: solid;  
  border-color: red; }
```

Usando estas propriedades os valores irão alterar as **4 bordas do elemento**, se quiser alterar apenas um dos lados você poderá usar as propriedades abaixo:

- border-top-width:.....espessura da borda superior
- border-top-style:.....estilo da borda superior
- border-top-color:.....cor da borda superior

Estas propriedades correspondem ao top, ou seja, as margens de cima, mas o mesmo pode ser usado para os lados e a borda de baixo usando respectivamente right, left e bottom.

Podemos ainda usar a forma resumida:

- border:.....maneira abreviada para todas as quatro bordas
- border-top:...maneira abreviada para todas as propriedades da borda superior
- border-right:...maneira abreviada para todas as propriedades da borda direita
- border-bottom:...maneira abreviada para todas as propriedades da borda inferior
- border-left:...maneira abreviada para todas as propriedades da borda esquerda

Neste caso basta você informar os valores desejados para os 3 atributos, exemplo:

**Border: 1px solid #000;**

Valores válidos para as propriedades das bordas

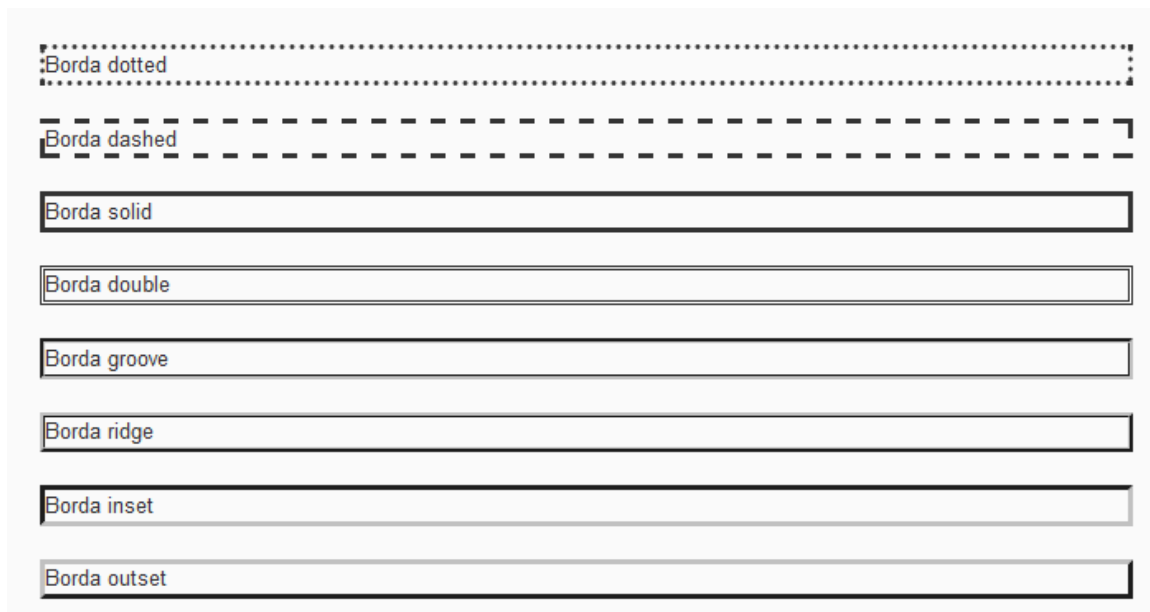
#### **color**

- código hexadecimal: #FFFFFF
- código rgb: rgb(255,235,0)
- nome da cor: red, blue, green...etc

#### **style**

- none: nenhuma borda
- hidden: equivalente a none
- dotted: borda pontilhada
- dashed: borda tracejada
- solid: borda contínua
- double: borda dupla
- groove: borda entalhada
- ridge: borda em relevo
- inset: borda em baixo relevo
- outset: borda em alto relevo

Veja na figura a seguir como é cada estilo de borda:



Fonte: <http://maujor.com/tutorial/bordertut.php>

### width

- thin: borda fina
- medium: borda média
- thick: borda grossa
- length: uma medida reconhecida pelas CSS (px, pt, em, cm, ...)

### Exemplo:

Ainda na página da ETEC criada nos exercícios anteriores, vamos colocar uma borda na página, de forma que o body no css fique conforme código abaixo:

```
body{  
    text-indent:1.25cm;  
    background-image: url("imagens/muro.jpg");  
    border-width: 5px;  
    border-style:inset;  
    border-color: red;  
}
```

Repare que foi utilizado as formatações para todas as bordas ao mesmo tempo, agora vamos alterar apenas dois lados usando a forma reduzida da propriedade border.

Ainda no CSS altere o img conforme exemplo abaixo:

```
img{  
    float: left;  
    border-right:5px groove #778899;  
    border-bottom:5px groove #778899;
```

}

O resultado deverá ser parecido com este:



Podemos ver que temos agora a borda na página toda e apenas em dois lados da imagem.



## ATIVIDADE AVALIATIVA

- Criar um código HTML5 e CSS EXTERNO aplicando BORDAS no elemento <body>, os quatro lados.
- Atividade em DUPLA e entrega MANUSCRITO em folha de caderno.

### Margin e Padding

Os atributos de CSS margin e padding, embora em algumas ocasiões possam se comportar de maneira parecida na prática, são muito diferentes.

**Margin** é a margem que há desde um elemento até os que tiverem ao lado.

**Padding** é o espaço que há em um container entre o conteúdo e as bordas do container.

Ou seja, ao especificar uma margin estamos definindo um espaço entre uma camada e as camadas texto, imagens ou outros conteúdos que tiver aos lados.

Ao especificar um padding em uma camada, por exemplo, estamos definindo o espaço que há entre a borda da camada e o que tiver dentro.

### As propriedades para margens

- margin-top.....define a margem superior;
- margin-right.....define a margem direita;
- margin-bottom.....define a margem inferior;
- margin-left.....define a margem esquerda;
- margin.....maneira abreviada para todas as margens

### Valores válidos para as propriedades de margem

- auto: valor default da margem
- length: uma medida reconhecida pelas CSS (px, pt, em, cm, ...)
- %: porcentagem da largura do elemento pai

Usando a página da ETEC criada nos exercícios anteriores vamos adicionar uma margem no body, exemplo:

```
margin: 30px;
```



Agora teremos um espaço de 30px entre o início da página e o início da margem.

Também poderíamos adicionar valores diferentes para cada uma das bordas podendo ser feito de duas formas:

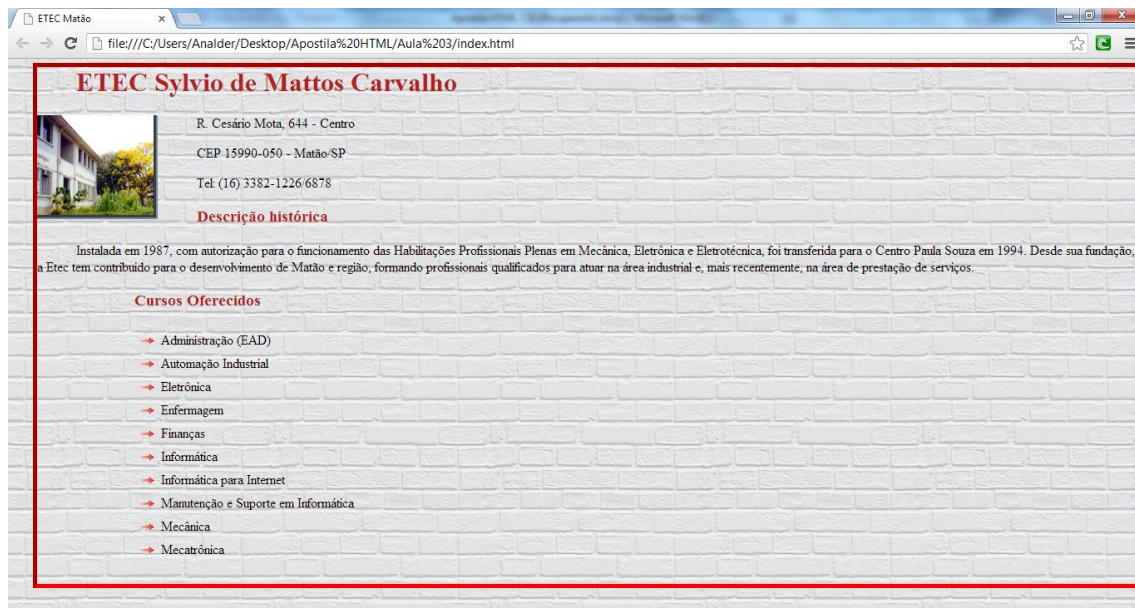
```
P { margin: 5px 5px 30px 30px; }
```

Neste caso os valores são respectivamente (superior, direita, inferior, esquerda).

Ou ainda poderíamos definir cada valor com uma propriedade:

```
Body {  
margin-left: 30px;  
margin-bottom: 30px;  
margin-top: 5px;  
margin-right: 5px; }
```

Em ambos os casos o resultado alcançado será o mesmo conforme imagem a seguir:



## As Propriedades Padding

São semelhantes as propriedades margin, podendo ser utilizada de forma reduzida apenas com o padding ou uma para cada um dos lados.

- padding-top.....define a espaçamento superior;
- padding-right.....define a espaçamento direita;
- padding-bottom.....define a espaçamento inferior;
- padding-left.....define a espaçamento esquerda;
- padding.....maneira abreviada para todas os espaçamentos

## Valores válidos para as propriedades de espaçamento

- auto: valor default da margem
- length: uma medida reconhecida pelas CSS (px, pt, em, cm, ...)
- %: porcentagem da largura do elemento pai

No exemplo da página da ETEC vamos definir o padding de 50px para todos os lados, exemplo:

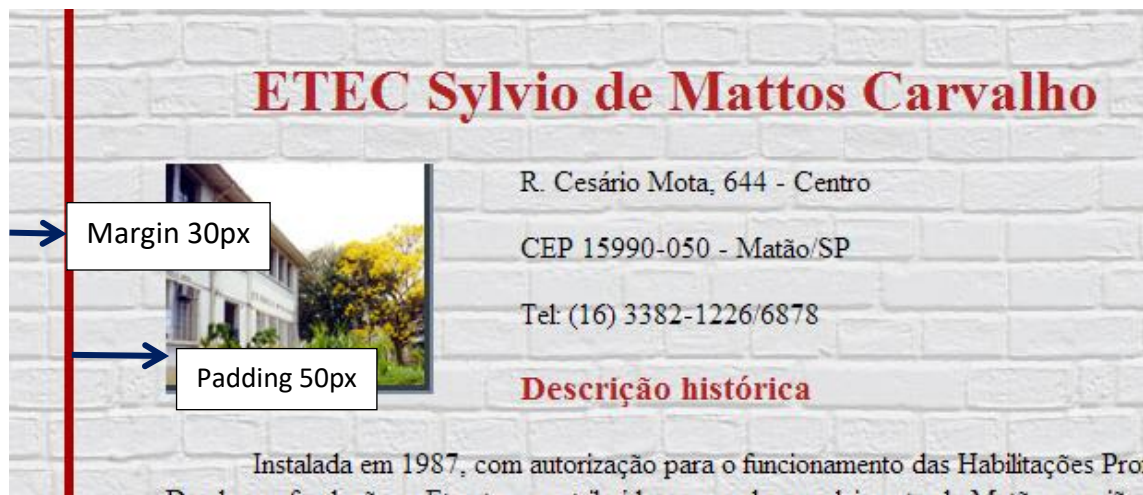
```
body {  
    padding: 50px;  
}
```

Obs: Quando definimos um único valor, ele será aplicado aos quatro lados.

Resultado:



### Entendendo melhor



As propriedades Margin, Border e Padding podem ser usada para os outros itens do html também, não apenas para o body.

### Exercícios:

1. Explique a propriedade MARGIN. Dê exemplos de código.
2. Explique a propriedade BORDER. Dê exemplos de código.
3. Explique a propriedade PADDING. Dê exemplos de código.

## Hyperlinks HTML (Links)

A tag <a> HTML define um hyperlink.

Um hiperlink (ou link) é uma palavra, grupo de palavras, ou a imagem que você pode clicar para abrir um outro documento ou arquivo, quando você move o cursor sobre um link em uma página da Web, o ponteiro do mouse se transformar em uma mãozinha, indicado portanto que aquele objeto é um link.

O atributo mais importante do elemento <a> é o atributo href, que indica o destino do link, ou seja, qual outra página ou mesmo arquivo será aberto quando clicado.

Por padrão, os links vão aparecer da seguinte forma em todos os navegadores:

- Um link não visitado é sublinhado e azul
- Um link visitado é sublinhado e roxo
- Um link ativo é sublinhado e vermelho

Mas podemos modificar este padrão através do CSS.

### Sintaxe HTML Link

O código HTML para um link é simples. Veja o Exemplo:

```
<a href="url ">Texto</a>
```

Em URL deve ficar o endereço do que se deseja abrir, podendo ser por exemplo um arquivo ou mesmo uma outra página de internet, no caso de arquivo deverá ser fornecido o caminho completo do arquivo, assim como fazemos com as imagens que inserimos nas páginas.

### O Atributo Target

Nos Links podemos definir se queremos que a página seja aberta em uma nova aba ou se deve sobrepor o conteúdo atual, por padrão, quando criamos um link haverá a sobreposição, ou seja, a página será aberta na mesma aba, se quiser que o conteúdo do link seja aberto em uma nova aba você deverá utilizar o atributo target indicando como valor “\_blank”

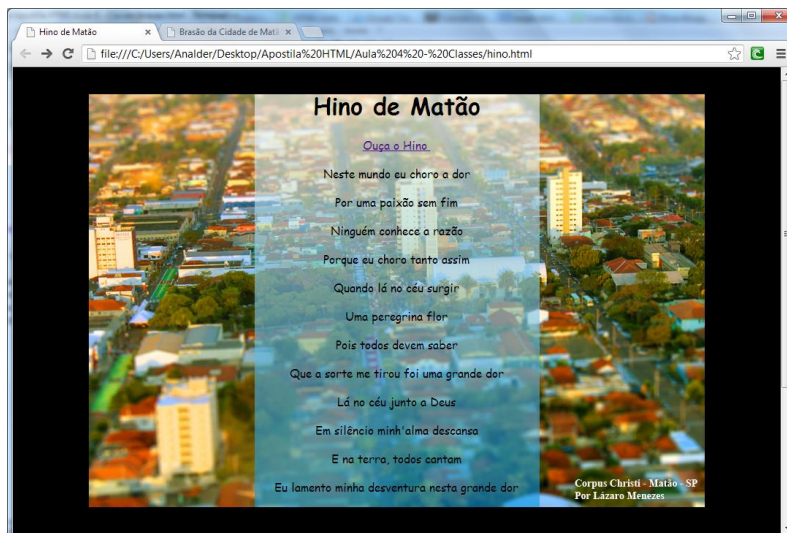
Exemplo:

```
<a href="pagina.html" target="_blank"> Abrir em Nova Aba </a>
```

Exercício:

1. Crie a Página abaixo e salve-a como hino.html





- O hino de Matão você encontra em <http://letras.mus.br/hinos-de-cidades/945331/> ou em uma busca pela internet
- A página deve ter fundo preto e a imagem que será fornecida pelo professor, a imagem de background deverá ficar centralizada e fixa.
- O link deverá abrir o arquivo Hino\_Matao.mp3 fornecido pelo professor
- A fonte deverá ser ComicSans.

2. Crie a página abaixo e salve como brasao.html



- Não é necessário criar um novo estilo, deve ser usado o mesmo do exercício anterior.
- O texto centralizado será conseguido com o uso do alinhamento e do Padding (este valor poderá variar de acordo com a resolução, no exemplo foi utilizado 32%)

3. Crie a página abaixo e salve como bandeira.html



#### 4. Crie a página abaixo e salve como index.html



#### 5. Faça os links para as páginas criadas nos exercícios 1, 2 e 3.

## Classes e ID

Até aqui aplicamos os mesmos estilos para todos os elementos iguais, mas acontece que nem sempre é assim, muitas vezes precisamos ter por exemplo dois elementos h1 com formatações diferentes, porém da forma que vimos até aqui, sempre que fizer qualquer alteração no elemento h1 ela será aplicada para todos os elementos h1 da página.

Para resolver este problema temos as Classes e o ID dos elementos

### O Selector ID

O seletor id é usado para especificar um estilo para um elemento único e exclusivo, ele usa o atributo id do elemento HTML, e é definido com um "#".

Para declarar o atributo no HTML basta adicionar o atributo ao elemento, exemplo:

```
<h1 Id="titulo1"> Meu Título Principal </h1>
```

No CSS usaremos

```
#titulo1{  
text-align:center;  
font-size:100px;  
color:red;  
}
```

Agora vamos fazer mais um título

```
<h1 Id="titulo2"> Meu Título Secundário </h1>
```

No CSS usaremos

```
#titulo2{
text-align:left;
font-size:50px;
color:blue;
}
```

O Resultado será este:



### O Seletor de Class

O seletor de classe é usado para especificar um modelo para um grupo de elementos. Ao contrário do seletor id, o seletor de classe é mais frequentemente usado em vários elementos.

Isso permite que você defina um estilo particular de muitos elementos HTML com a mesma classe.

O seletor de classe usa o atributo da classe HTML, e é definido com um "."

Exemplo:

```
<p class="azul"> Paragrafo 1 com texto Azul </p>
<p class="verde"> Paragrafo 2 com texto Verde </p>
<p class="azul"> Paragrafo 3 com texto Azul </p>
<p class="verde"> Paragrafo 4 com texto Verde </p>
```

No CSS usaremos

```
.azul{color:blue;}
.verde{color:GreenYellow;}
```

### Quando usar Classe e quando usar ID

Se o objeto que você deseja formatar for único dentro da sua página você usará o ID, caso contrário, se houver repetição do elemento com a mesma formatação, você usará classe.

Você pode ainda determinar formatações diferentes para elementos diferentes com a mesma classe, veja o exemplo:

No HTML:

```
<h2 class="azul"> Título</h2>
```



```
<p class="azul"> Paragrafo 1 com texto Azul </p>
```

No CSS

```
.azul{color:blue;}  
p.azul{font-size:16px;}
```

Neste caso ao colocar o elemento antes do nome da classe, estou dizendo que apenas aquele tipo de elemento daquela classe deverá ter essa formatação, no exemplo, os parágrafos também manterão a cor azul, no entanto o tamanho da fonte será diferente do tamanho do elemento h2.

## Pseudo-elementos CSS:

São usados em CSS, para adicionar efeitos a um seletor, ou a parte de um seletor.

### A sintaxe dos pseudo-elementos:

```
seletor:pseudo-elemento {propriedade: valor;}
```

As classes em CSS podem também ser usadas com pseudo-elementos.

Esta regra permite que você defina diferentes efeitos para pseudo-elementos localizados em diferentes lugares em uma mesma página.

```
seletor.classe:pseudo-elemento {propriedade: valor;}
```

O pseudo-elemento `first-letter` é usado para obter um efeito especial na primeira letra de um texto.

O pseudo-elemento `first-line` é usado para obter um efeito especial na **primeira linha** de um texto.

Algumas propriedades aplicáveis ao pseudo-elemento `first-letter` e `first-line`

- font - propriedades de letras
- color - propriedades de cores
- background - propriedades de fundo
- margin - propriedades de margens
- padding - propriedades de espaçamentos
- border - propriedades de bordas
- text-decoration
- text-transform

Exemplo:

```
p:first-letter{  
font-size:40px;  
color:GreenYellow;
```

```
}
```

Resultado:

**P**aragrafo 1 com texto Azul

### Before e after

Os pseudo elementos `before` (antes) e `after` (depois) são usados em conjunto com a propriedade `content` pra colocar conteúdo de qualquer lado de um elemento sem tocar no HTML.

Exemplos:

```
H1:before{ content:" Capítulo: "}
```

```
H1:after{ content:url(imagem.jpg)}
```

Exercício:

1. Usando a pagina criada no exercício dos símbolos de Matão, use o `before` e o `after` para inserir uma imagem antes do link e o texto “ – Acesse” após o link.



### Pseudo-classe para links

Além das pseudo-classes vistas anteriormente, temos algumas para link, um link possui quatro estados, e para cada estado temos uma pseudo-classe que nos permite fazer formatações diferentes para cada um destes estados, são eles:

- `a:link`.....define o estilo do link no estado inicial;
- `a:visited`....define o estilo do link visitado;
- `a:hover`.....define o estilo do link quando passa-se o mouse sobre ele;
- `a:active`....define o estilo do link ativo (o que foi "clicado").

Vamos usar o exemplo a página dos Símbolos de Matão que utilizamos no exercício anterior, crie as formatações CSS abaixo no estilo do referido exercício:

```
a:link, :visited{
font-size:20px;
color: #1874CD;
text-decoration:none;
}
a:hover, :active{
font-size:24px;
color: #473C8B;
```

```
text-decoration:underline;  
}
```

Neste Exemplo foi colocadas formatações diferentes para os estados link (o natural) e visitado (quando ele já foi clicado) de forma que não haverá distinção entre os links já visitados como é o padrão do HTML, também foi retirado o sublinhado padrão de links, e outra formatação foi criada para o hover (quando o cursor está sobre o link) e active (quando o link é clicado).

O efeito disto é que quando você mover o mouse haverá uma pequena alteração no tamanho e na cor do link retornando ao normal assim que o mouse deixar a área do link.

### Exercício:

1. Crie um link para voltar a página index nas 3 páginas de símbolos.

## Menus com Listas

Uma forma muito comum de se fazer menus é utilizando o elemento `<ul>` listas, basta criar uma lista como vimos anteriormente e depois formatá-la usando CSS, vamos a um exemplo.

Primeiro vamos criar um menu horizontal, para isso crie a lista abaixo:

```
11 <ul class="menu_horizontal">  
12   <li><a href="#">Home</a></li>  
13   <li><a href="#">História</a></li>  
14   <li><a href="#">Filosofia</a></li>  
15   <li><a href="#">Produtos</a></li>  
16   <li><a href="#">Contato</a></li>  
17 </ul>
```

Agora no arquivo CSS você formatará da seguinte forma:

```
4 ul.menu_horizontal {  
5   margin: 0;  
6   padding: 0;  
7   list-style: none;  
8   font-size: 20px;  
9 }
```

A principal alteração que fizemos aqui foi definir o `list-style` como `none`, já que o que queremos não é uma lista e sim apenas a estrutura da lista, desta forma não precisamos de nenhum símbolo na lista.

Continuando:

```
11 ul.menu_horizontal li a{
12     float:left;
13     display: inline;
14     padding: 10px;
15     background: #228B22;
16     border-right: 1px solid #fff;
17     text-decoration: none;
18     border-left:10px solid #FFD700;
19     color:#fff;
20 }
```

Agora as alterações são maiores, a começar pelo left, já que queremos um menu horizontal, isso quer dizer que vamos precisar que os itens da lista flutuem um do lado do outro, caso contrario eles ficariam um abaixo do outro.

Para efeito de decoração foi definida a borda direita como branca de 1px e a esquerda de 10px na cor amarela, o fundo foi definido como verde e a cor da fonte branca.

Essa formatação fará efeito para o link no estado natural, agora vamos formatar o hover para que a formatação mude quando passar o mouse em cima.

```
22 ul.menu_horizontal li a:hover {
23     background: #FFD700;
24     border-left:10px solid #228B22;
25     color:#228B22;
26 }
```

Apenas para termos um pequeno efeito foi alterado no hover a cor de fundo, a borda esquerda e a cor do texto.

O resultado será este:



No exemplo o mouse está sobre o item Produtos, por isso ele apresenta cor diferenciada dos demais.

Agora Vamos ao link Vertical.

Criaremos uma cópia da lista anterior porém agora iremos alterar o nome da classe para Menu\_Vertical.

No CSS faremos uma formatação parecida com a anterior:

```

29  /* Menu Vertical */
30  ul.menu_vertical {
31      margin: 0;
32      padding: 0;
33      list-style: none;
34      width:140px;
35      font-size:20px;
36
37  }
38
39  ul.menu_vertical li a{
40      display: block;
41      padding: 10px;
42      background: #228B22;
43      border-bottom: 1px solid #fff;
44      text-decoration: none;
45      border-left:10px solid #FFD700;
46      color:#fff;
47  }
48  ul.menu_vertical li a:hover {
49      background: #FFD700;
50      border-left:10px solid #228B22;
51      color:#228B22;
52  }

```

O que mudou com relação ao anterior foi o width de 140px no ul.menu\_vertical, isso se faz necessário para delimitar o tamanho do menu, se não alterarmos esta propriedade o menu ocupará a tela toda.

Outra alteração importante foi na formatação do a, onde alterou-se o display para block e retirou-se o float, já que neste menu os itens não irão flutuar.

O resultado será este:



Assim como no anterior, neste exemplo o cursor do mouse também está sobre a opção Produtos.

## Div e Span

Os elementos div e span foram criados com o HTML 4 com a finalidade de fornecer um mecanismo genérico para agrupar e prover estrutura aos documentos. O elemento div é um container nível de bloco e span é um elemento inline. Uma id e/ou class é em geral usada para fornecer uma identidade a uma div ou span com o propósito de fornecer uma referência para estilização por CSS

ou captura por um script. O elemento `div` é semanticamente neutro: ele não tem qualquer valor semântico, é simplesmente um elemento container. O elemento `div` deverá ser usado quando não estiver disponível um outro elemento container e semântico, capaz de substituí-lo tal como são os cabeçalhos, parágrafos, listas etc.

Vamos a um exemplo do elemento `span`, digite o código abaixo em uma página html:

```
<p> Minha mãe tem <span class="azul"> olhos azuis </span> e meu pai tem  
<span class="verde"> olhos verdes</span></p>
```

Repare que no código foi criado dois spans, cada um com uma classe diferente, repare também que o span foi colocado no meio de um parágrafo, é exatamente esta a sua função, fazer formatação diferente em um elemento dentro de outro elemento.

Agora no CSS vamos fazer a formatação de cada uma destas duas classes, conforme exemplo abaixo:

```
p{font-size:30px}  
.azul{color:blue;}  
.verde{color:green;}
```

O resultado será este:

Minha mãe tem olhos azuis e meu pai tem olhos verdes

Agora vamos ao elemento `div`, que será muito utilizado em suas páginas html. Como dito ele é um elemento de nível de bloco, usaremos ele para agrupar outros elementos. A princípio vamos criar 3 divs para que fique mais fácil sua compreensão. Em um arquivo html faça:

```
10 <body>  
11 <div class="Div1">  
12 </div>  
13  
14 <div class="Div2">  
15 </div>  
16  
17 <div class="Div3">  
18 </div>  
19 </body>
```

No CSS faça:

```
1  .div1{
2    background-color:red;
3    width:800px;
4    height:200px;
5    border: 2px solid black;
6  }
7
8  .div2{
9    background-color:blue;
10   width:800px;
11   height:200px;
12   border: 2px dotted black;
13 }
14
15 .div3{
16   background-color:yellow;
17   width:800px;
18   height:200px;
19   border: 2px dashed black;
20 }
```

O resultado será este:



Cada div foi colocada de uma cor e com uma borda diferente para que você pudesse compreender o que é cada div. Você deve ter percebido que foram criadas 3 caixas, e dentro destas 3 caixas poderíamos inserir novos elementos. Vamos, por exemplo, inserir uma frase e uma imagem do autor da frase em cada um destes nossos 3 blocos.

### EXERCÍCIOS

O que é classe e id em CSS?

Qual a diferença entre classe e id?

O que é a tag DIV?

O que é a tag SPAN?

Baseado neste primeiro faça os outros dois:

```
<body>
  <div class="Div1">
    
    <p> A mente que se abre a uma nova idéia jamais voltará ao seu tamanho original.</p>
    <p>Albert Einstein</p>
  </div>
```

Faça também as devidas formatações css.

Seu site ao final deverá ficar assim:



Agora altere o css das divs mantendo as configurações já existentes e inserindo a formatação de margin conforme exemplo:

```
.div1{
...
margin-left:50px;
}

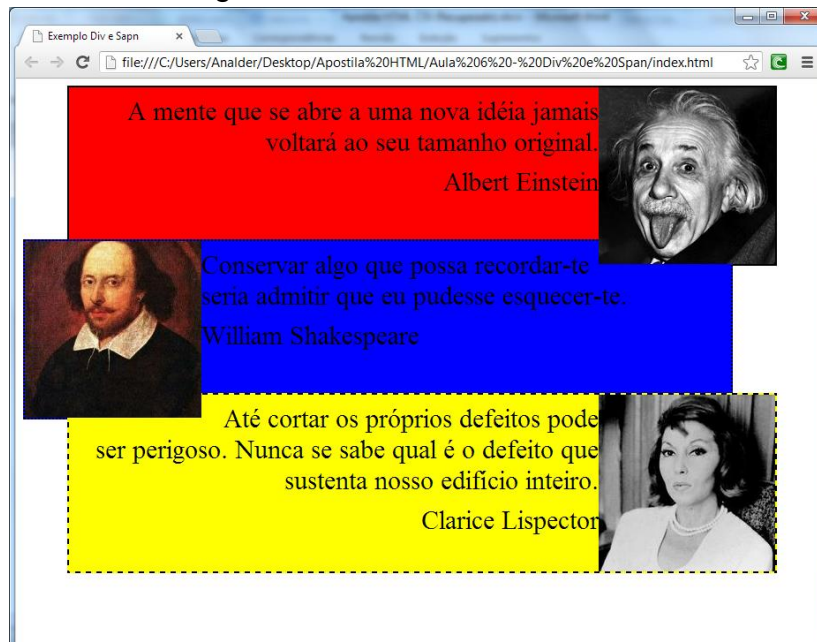
.div2{
...
margin-top:-30px;
}

.div3{
...
margin-top:-30px;
```



```
margin-left:50px;  
}
```

O Resultado agora será este:



Podemos ainda alterar as posições dos divs, altere o seu CSS conforme abaixo:

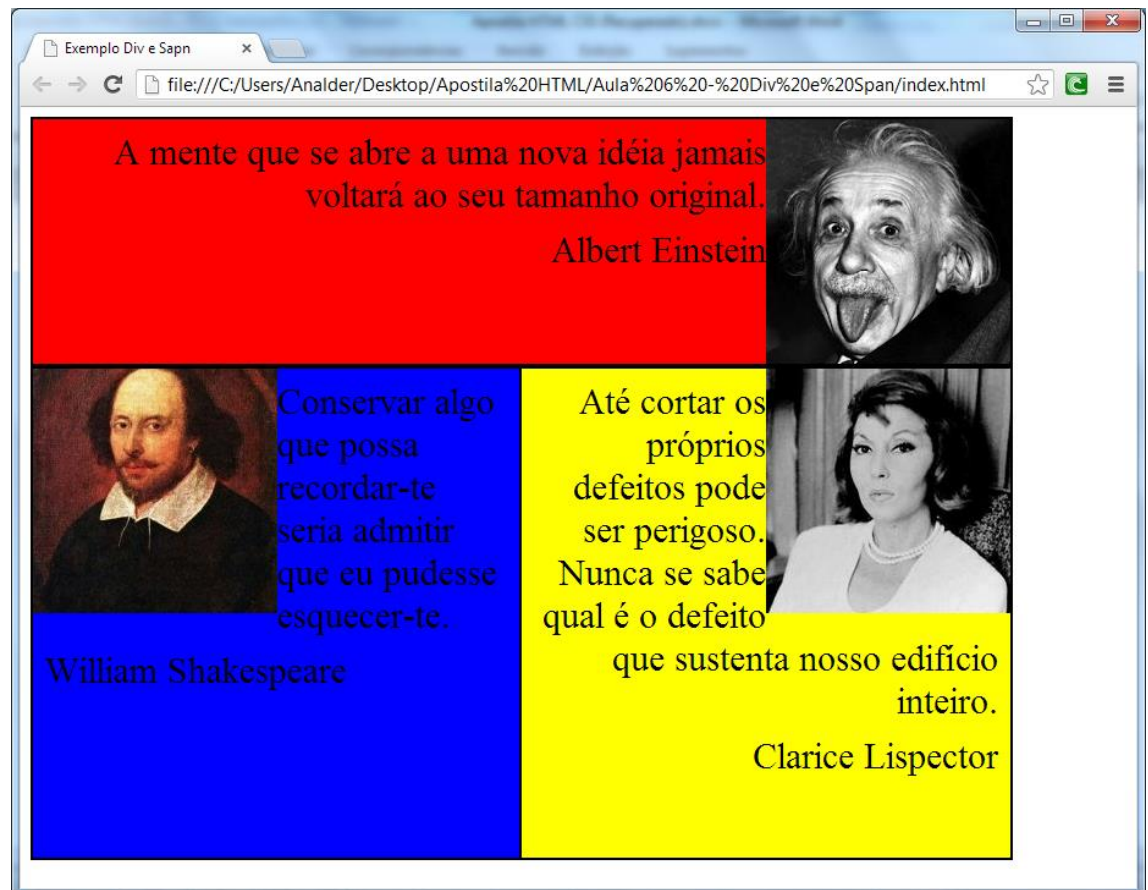
```
.div1{  
background-color:red;  
width:800px;  
height:200px;  
border: 2px solid black;  
text-align:right;  
font-size:30px;  
}
```

```
.div2{  
background-color:blue;  
width:400px;  
height:400px;  
border: 2px solid black;  
text-align:left;  
font-size:30px;  
}
```

```
.div3{  
background-color:yellow;  
width:400px;
```

```
height:400px;  
border: 2px solid black;  
text-align:right;  
font-size:30px;  
margin-top:-404px;  
margin-left:400px;  
}
```

O resultado será este:



Repare que modificamos a página sem alterar o html, toda modificação ocorreu no css.

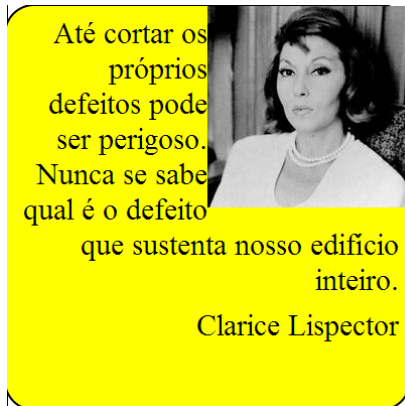
### Bordas Arredondadas com CSS

Um efeito interessante que pode deixar nossos sites mais bonitos são as bordas arredondadas, para que consigamos este efeito teremos que usar o border-radius, veja o exemplo:

Insira no css do div3 do exemplo das frases a seguinte linha:

```
border-radius:30px;
```

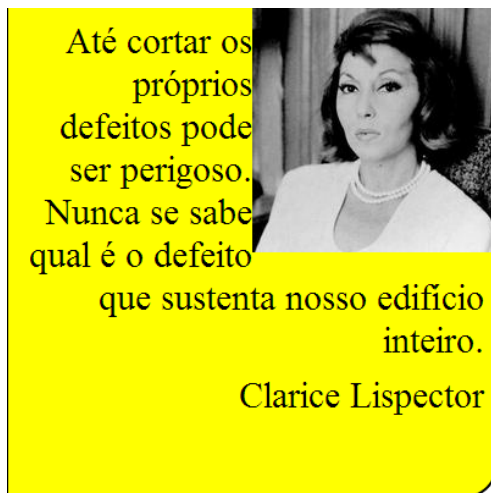
O div com a frase da Clarice Lispector aparecerá da seguinte forma agora:



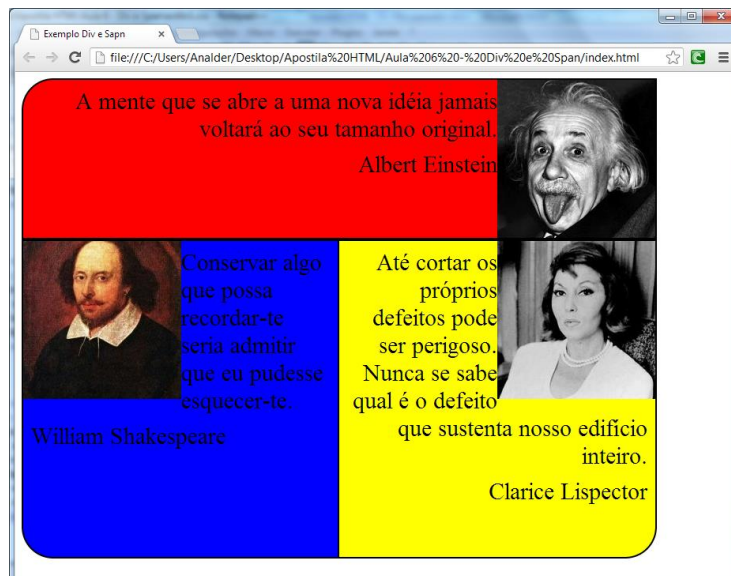
Repare que as quatro bordas tem os cantos arredondados, quanto maior o px mais arredondado ficará a figura. Neste exemplo a imagem está escondendo a borda arredondada na parte de cima, mas ela está arredondada também. Agora altere o valor do border-radius da seguinte forma:

```
border-radius: 0 0 30px 0;
```

Repare que agora estamos fornecendo quatro valores, isso quer dizer que estamos fornecendo um para cada lado, a sequencia dos números segue a sequencia dos lados, de forma que como está apenas uma borda ficará arredondada. Veja como ficará:



Agora faça com que sua página fique conforme a imagem abaixo, arredondando apenas um canto de cada uma das 3divs:



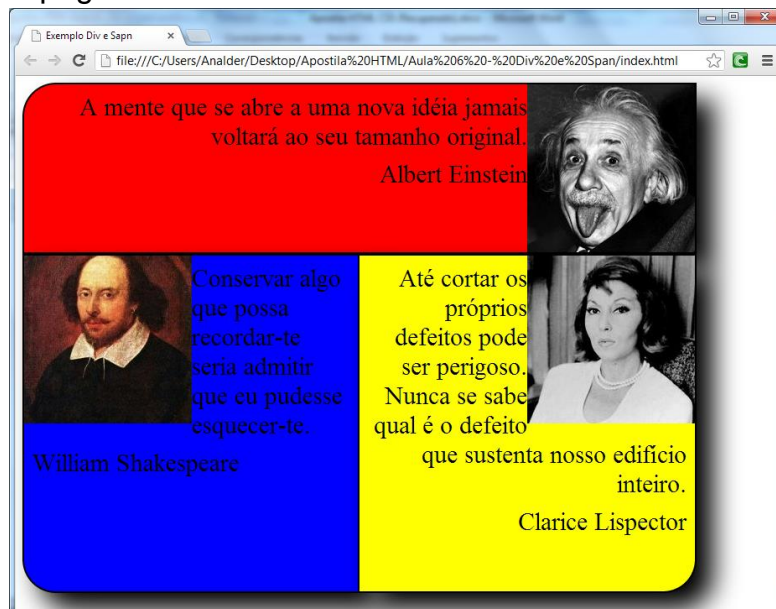
## Sombras com CSS

Outro efeito interessante que podemos obter com CSS são as sombras utilizando o shadow, como estamos trabalhando com o elemento div que é uma caixa podemos colocar sombra nele adicionando a propriedade box-shadow.

Adicione a linha abaixo as 3divs:

```
box-shadow: 20px 20px 50px black;
```

A página ficará assim:



Ou seja, foi adicionado uma sombra sob as 3 caixas.

Note que o box-shadow possui 4 valores, veja:

box-shadow: X Y Z cor;

Sendo:

- X a posição da sombra com relação a horizontal, ou seja, direita ou esquerda, sendo 0 o centro, valores negativos para a esquerda e valores positivos para a direita.
- Y a posição com relação a vertical, ou seja, acima e abaixo, sendo 0 o centro, valores negativos acima e valores positivos abaixo.
- Z a solidez da sombra, quanto menor o valor mais solida será a sombra, quanto maior mais dispersa ela será, veremos um exemplo a seguir.
- Cor indica qual a cor da sombra, nos exemplos estamos nos limitando ao preto mas poderia ser usada qualquer outra cor da tabela de cores hexadecimal aceita no CSS.

Para compreender melhor os valores vamos ver alguns exemplos:

#### **Alterando os valores de X**

box-shadow: **10px**10px10pxblack;}

box-shadow: **-10px**10px10px black;}

#### **Alterando os valores de Y**

box-shadow: 10px **10px**10pxblack;}

box-shadow: 10px **-10px** 10px black;}

#### **Alterando os valores de Z**

box-shadow: 10px 10px**10px**black;

box-shadow: 10px 10px**50px** black;

Teste com 10px

Teste com -10px

Teste com 10px

Teste com -10px

Teste com 10px

Teste com 50px

#### **Exercício:**

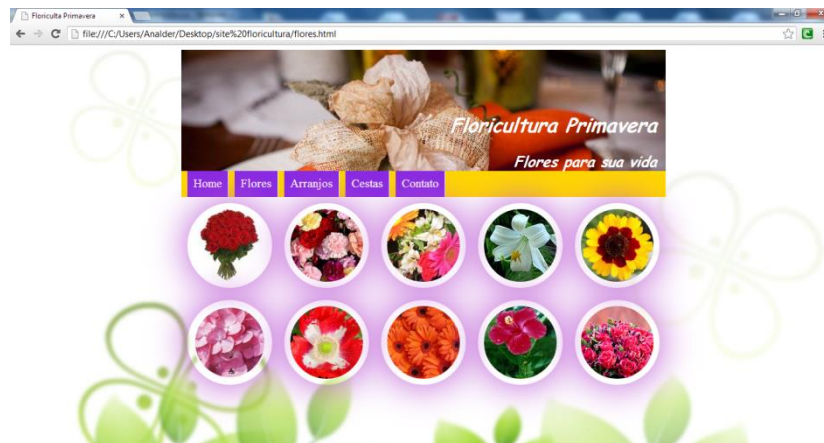


## 1. Crie as páginas abaixo e faça os devidos links

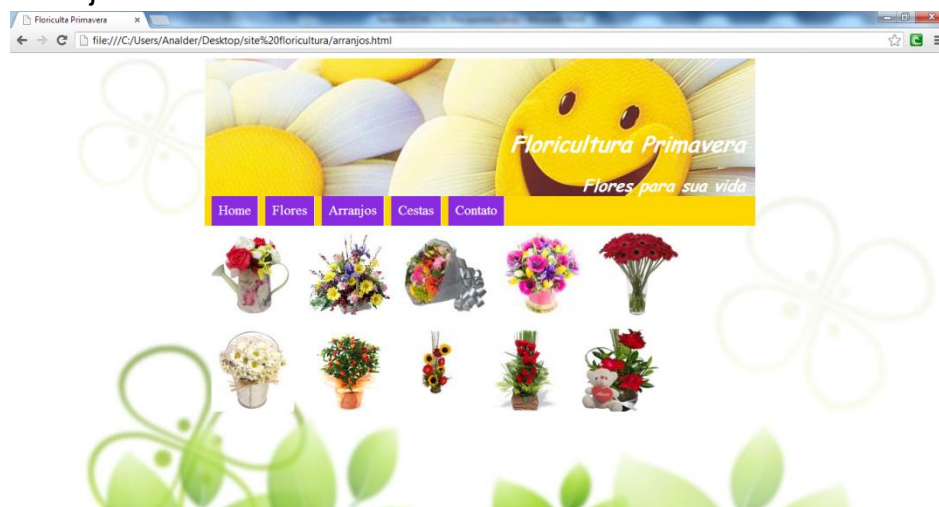
### index.html



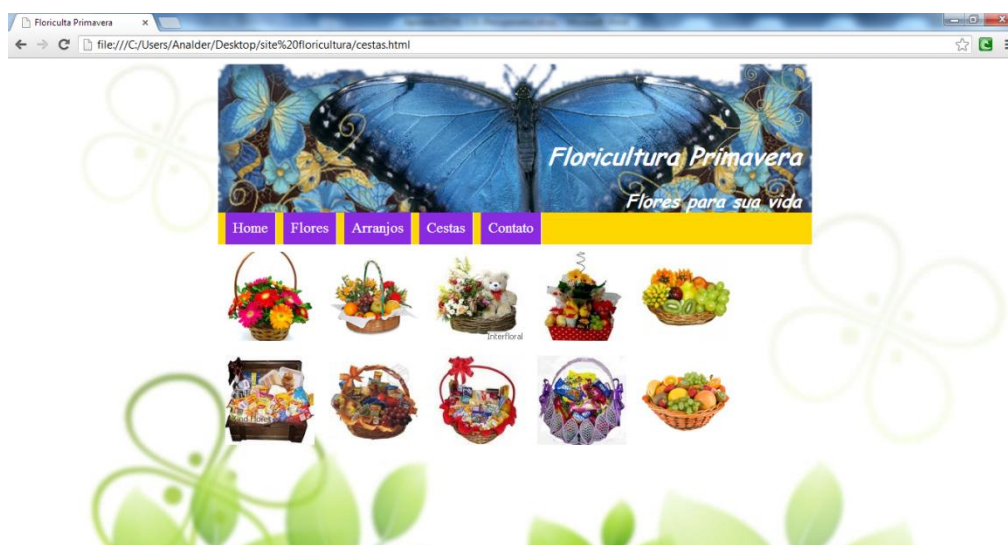
### Flores.html



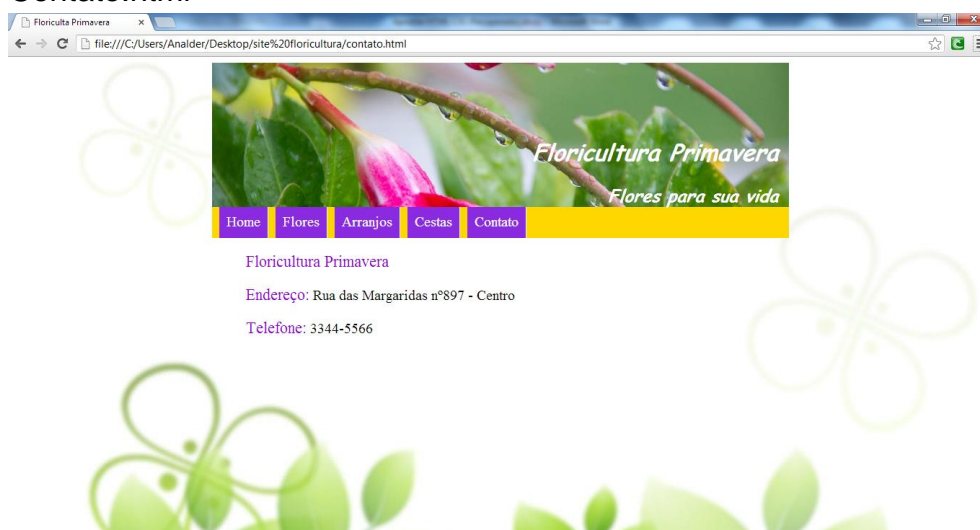
### Arranjos.html



### Cestas.html



## Contato.html



## Trabalhando com Tabelas

As tabelas já foram muito utilizadas para construir páginas HTML, principalmente para construir layout de páginas, porém com o surgimento do elemento `div` essa sua função deixou de existir, e tabelas passaram a ser utilizadas apenas para construção de estruturas de tabelas, ou seja, estruturas com linhas e colunas preenchidas com dados, tal qual uma tabela no word ou mesmo uma planilha no excel.

Uma tabela é criada através do elemento `<table>`, além do elemento `<table>` teremos ainda um ou mais `<tr>`, `<th>`, e elementos `<td>`.

A tag `<tr>` define uma linha em uma tabela HTML. Um elemento `<tr>` contém um ou mais elementos ou `<th>``<td>`.

A tag `<th>` define uma célula de cabeçalho em uma tabela HTML, uma tabela HTML tem dois tipos de células:

- As células de cabeçalho - contém informações de cabeçalho (criado com o elemento <th>)
- Células Normais - contém dados (criado com o elemento <td>)

O texto em elementos <th> estão em negrito e centrado por padrão.

O texto em elementos <td> são regulares e alinhado à esquerda por padrão.

Para Compreender crie a tabela abaixo:

```

9
10 <body>
11 <table>
12 <tr>
13 <th> Titulo Coluna 1</th>
14 <th> Titulo Coluna 2</th>
15 <th> Titulo Coluna 2</th>
16 </tr>
17 <tr>
18 <td> Celula 1 - Linha 1</td>
19 <td> Celula 2 - Linha 1</td>
20 <td> Celula 2 - Linha 1</td>
21 </tr>
22 <tr>
23 <td> Celula 1 - Linha 2</td>
24 <td> Celula 2 - Linha 2</td>
25 <td> Celula 2 - Linha 2</td>
26 </tr>
27 <tr>
28 <td> Celula 1 - Linha 3</td>
29 <td> Celula 2 - Linha 3</td>
30 <td> Celula 2 - Linha 3</td>
31 </tr>
32 </body>

```

O Resultado deste código será este:

<b>Titulo Coluna 1</b>	<b>Titulo Coluna 2</b>	<b>Titulo Coluna 2</b>
Celula 1 - Linha 1	Celula 2 - Linha 1	Celula 2 - Linha 1
Celula 1 - Linha 2	Celula 2 - Linha 2	Celula 2 - Linha 2
Celula 1 - Linha 3	Celula 2 - Linha 3	Celula 2 - Linha 3

Como ainda não formatamos ainda não é possível ver as bordas da tabela, para isso será necessário utilizarmos a formatação CSS, coloque a formatação abaixo no arquivo CSS:

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

Agora a tabela ficará assim:

<b>Titulo Coluna 1</b>	<b>Titulo Coluna 2</b>	<b>Titulo Coluna 2</b>
Celula 1 - Linha 1	Celula 2 - Linha 1	Celula 2 - Linha 1
Celula 1 - Linha 2	Celula 2 - Linha 2	Celula 2 - Linha 2
Celula 1 - Linha 3	Celula 2 - Linha 3	Celula 2 - Linha 3

Para que não fiquem com a borda dupla podemos usar:



border-collapse: collapse;

Desta forma a tabela ficará assim:

Titulo Coluna 1	Titulo Coluna 2	Titulo Coluna 2
Celula 1 - Linha 1	Celula 2 - Linha 1	Celula 2 - Linha 1
Celula 1 - Linha 2	Celula 2 - Linha 2	Celula 2 - Linha 2
Celula 1 - Linha 3	Celula 2 - Linha 3	Celula 2 - Linha 3

Com relação as bordas das tabelas, conseguimos formata-las como as bordas de qualquer elemento html, desta forma tudo aquilo que vimos de bordas até o momento se aplica as tabelas também.

Para colorir uma célula de uma tabela basta alterarmos o background da td ou tr desejada. Para ilustrar isso faremos uma tabela zebrada, ou seja, com as linhas de cores intercaladas, para isso precisaremos definir duas classes para as tr, uma chamada impar outra chamada par, a começar pela tr onde está a th que será impar, como no exemplo abaixo:

```
<tr class="impar">
```

Feito isso com as 4 linhas da nossa tabela, agora basta formatar no css essas classes impar e par, conforme abaixo:

```
.impar{ background-color: #D3D3D3;}
```

```
.par{ background-color: #2F4F4F;}
```

Neste exemplo foi usada duas tonalidades de cinza, e a tabela ficará assim:

Titulo Coluna 1	Titulo Coluna 2	Titulo Coluna 2
Celula 1 - Linha 1	Celula 2 - Linha 1	Celula 2 - Linha 1
Celula 1 - Linha 2	Celula 2 - Linha 2	Celula 2 - Linha 2
Celula 1 - Linha 3	Celula 2 - Linha 3	Celula 2 - Linha 3

### Mesclando Células com ColSpan

Se você precisar mesclar células dentro de uma linha em tabelas você precisará utilizar o colspan, bastando informar quantas células você vai mesclar, veja o exemplo a seguir:

```
<table>
<tr>
  <th>TituloColuna 1</th>
  <th>TituloColuna 2</th>
  <th>TituloColuna 2</th>
</tr>
```

```

<tr>
  <td colspan="3"> 3 Celulas Mesclada com Colspan</td>
</tr>
<tr>
  <td colspan="2"> 2 Celulas Mesclada com Colspan</td>
  <td></td>
</tr>
<table>

```

Titulo Coluna 1	Titulo Coluna 2	Titulo Coluna 2
3 Celulas Mesclada com Colspan		
2 Celulas Mesclada com Colspan		

### Mesclando Celulas com RowSpan

Com o ColSpan conseguimos mesclar células da mesma linha, ou seja, transformamos duas ou mais colunas em uma única coluna, por isso **ColSpan**, mas se a ideia é unir células em uma coluna teremos que usar o RowSpan, já que transformaremos duas ou mais linhas em uma única, veja o exemplo abaixo:

```

<table>
<tr>
  <th>TituloColuna 1</th>
  <th>TituloColuna 2</th>
  <th>TituloColuna 2</th>
</tr>
<tr>
  <td rowspan="3"> 3 Celulas Mesclada <br>com RowSpan</td>
  <td> linha 1 - coluna 2 </td>
  <td> linha 1 - coluna 3 </td>
</tr>
<tr>
  <td> linha 2 - coluna 2 </td>
  <td> linha 2 - coluna 3</td>
</tr>
<tr>
  <td> linha 3 - coluna 2 </td>
  <td> linha 3 - coluna 3 </td>
</tr>
<table>

```

Esse código irá gerar essa tabela:

Titulo Coluna 1	Titulo Coluna 2	Titulo Coluna 2
3 Células Mesclada com RowSpan	linha 1 - coluna 2	linha 1 - coluna 3
	linha 2 - coluna 2	linha 2 - coluna 3
	linha 3 - coluna 2	linha 3 - coluna 3

O importante quando se usa o RowSpan, é lembrar que uma coluna será usada para a mesclagem e portanto não deverá conter uma td, no exemplo acima, reparem que apenas a primeira <tr> contém 3 <td>s, já que as demais terão apenas duas células.

### Exercícios:

1. Crie a tabela abaixo:

Imposto de Renda na fonte 2013			
De	Até	Alíquota (%)	Parcela a deduzir (em R\$)
—	1.710,78	—	—
1.710,79	2.563,91	7,50	128,31
2.563,92	3.418,59	15,00	320,60
3.418,60	4.271,59	22,50	577,00
Acima de 4.271,59		27,50	790,57

2. Crie a página abaixo:

Clubes Brasileiros com Libertadores da América		
Posição	Clube	Quantidade
1º	 São Paulo	3
	 Santos	
	 Cruzeiro	
3º	 Grêmio	2
	 Internacional	
6º	 Corinthians	1
	 Flamengo	
	 Palmeiras	
	 Vasco	



## REFERÊNCIAS

<http://codando.wordpress.com/2008/07/20/tutorial-avanado-de-css-pseudo-elementos/>

<http://dabuttonfactory.com/>

<http://maujor.com>

<http://tableless.com.br/o-que-html-basico/#.Ubi1DIbgaup>>

<http://www.criarweb.com>

<http://www.devmedia.com.br>

<http://www.marcioprado.eti.br/ensino.html?download=6%3Aapostilahtml> >

<http://www.mxstudio.com.br/Conteudos/Dreamweaver/Cores.htm>

<http://www.w3schools.com>

SILVA, Maurício Samy. **HTML5 A linguagem de marcação que revolucionou a web**. São Paulo. Editora Novatec, 2011

**Apostila Elaborada pelo Professor Analder Magalhães Honório**

**Versão 1.0 Jun/2013**