

1.3 A LINGUAGEM HTML

1.3.3 TAGS ESTRUTURAIS

Um documento html refere-se a um arquivo texto que deve ser salvo no sistema de arquivos com a extensão “.html” ou “.htm”.

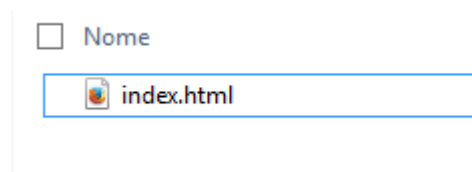


Figura 1 - Exemplo de arquivo html

Para que este arquivo possa ser interpretado corretamente pelo navegador web é preciso que o mesmo possua um conjunto de tags de marcação específicas.

Uma Tag é uma palavra reservada da linguagem disposta entre os sinais de maior “>” e menor “<”. Abaixo segue um exemplo de tag HTML.

```
<html>
```

Estas tags em sua grande maioria são compostas por uma tag de abertura, um conteúdo utilizado para ser estruturado pela tag, seguido de uma tag de fechamento. Esta tag de fechamento é igual a tag de abertura, porém precedida do caracter barra “/”. Somente as tags que não possuem conteúdo não necessitam de fechamento Exemplo:

```
<title>Este é um título</title>
```

```
<br>
```

Para que uma página html possa ser criada, é preciso que a página contenha um conjunto de tags básicas, conhecidas como tags estruturais, conforme pode ser observado no exemplo abaixo.

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

Cada uma das tags acima citadas possui um significado, o qual será detalhado abaixo:

`<!DOCTYPE html>`: Esta tag não faz parte do documento html, mas tem por objetivo informar o navegador que o documento descrito na página está construído com o uso de html versão 5;

`<html>` e `</html>`: São as tags que contém em seu interior todo o documento escrito na linguagem html. Qualquer página html deve conter estas tags;

`<head>` e `</head>`: Estas tags representam a estrutura de cabeçalho do documento. Dentro deste bloco de tags devem estar contidas apenas tags que possuem instruções que devem ser seguidas pelo navegador, como inclusão de arquivos complementares, descrição de scripts, regras, entre outros. Não devem ser inseridas tags de estruturação e formatação de texto neste local;

`<title>` e `</title>`: Esta tag tem por objetivo definir o título do documento html que será exibido na guia do navegador;

`<body>` e `</body>`: Estas tags representam o corpo do documento html. É dentro deste bloco de tags que serão adicionados todos os elementos e conteúdo que compõem a página html.

Observação: O documento HTML deve ser estruturado conforme apresentado, porém, se algumas destas tags não forem adicionadas ou adicionadas em ordem diferente da apresentada, o resultado visualizado no navegador possivelmente será o mesmo, visto que o motor de renderização¹ do navegador interpreta as tags conforme elas aparecem no documento, entretanto, o posicionamento incorreto das tags tornará a renderização da página mais lenta e também ocasionará erros de execução de rotinas Java Script.

Agora, implemente o exemplo acima em um arquivo html e execute este arquivo no navegador de sua preferência, analisando o resultado. Se tudo foi implementado de forma correta o resultado deverá ser uma página em branco.

A tag `<body>` permite a inclusão de propriedades. Estas propriedades são características que podem ou não ser adicionadas a tag, as quais complementam a funcionalidade original da tag. Determinadas tags somente funcionam de forma correta, quando tiverem as suas propriedades adicionadas. Estas propriedades sempre são um conjunto formado por chave e valor, onde o valor deve ser adicionado sempre entre aspas duplas. Exemplo:

`Background="imagem.png"`

Algumas das propriedades que podem ser adicionadas a tag `<body>` são:

Backbround: Esta propriedade permite vincular uma imagem de fundo para a página html;

¹ Motor de Renderização: É o software que transforma conteúdo em linguagem de marcação como o html e informações de formatação como o CSS em um conteúdo formatado para ser exibido no navegador. Atualmente existem diversos motores de renderização, de forma que alguns navegadores utilizam o mesmo motor de renderização. Um exemplo é o Gecko, utilizado pelo navegador Firefox.

Bgcolor: Esta propriedade permite definir uma cor de fundo para a página html. Esta cor pode ser definida como uma cor sólida através do seu nome em inglês, ou então através de um valor hexadecimal;

















Text: Esta propriedade permite definir a cor das letras da página html, sendo preenchida da mesma forma que a propriedade bgcolor.

Exemplo:

```
<!DOCTYPE html>
<html>
  <head>
    <title></title>
  </head>
  <body background="imagem.png" text="red">
  </body>
</html>
```

Observação: Estas propriedades da tag `<body>` funcionarão nos testes, entretanto, as mesmas não são mais suportadas pelo html 5.

Tabela 1 - Tabela de cores básicas

Cor	Nome da Cor	Representação Hexadecimal
	<i>black</i>	#000000
	<i>silver</i>	#C0C0C0
	<i>gray</i>	#808080
	<i>white</i>	#FFFFFF
	<i>maroon</i>	#800000
	<i>red</i>	#FF0000
	<i>purple</i>	#800080
	<i>fuchsia</i>	#FF00FF
	<i>green</i>	#008000
	<i>lime</i>	#00FF00
	<i>olive</i>	#808000
	<i>yellow</i>	#FFFF00
	<i>navy</i>	#000080
	<i>blue</i>	#0000FF
	<i>teal</i>	#008080
	<i>aqua</i>	#00FFFF

Uma tabela de cores completa pode ser encontrada neste [link](#).

1.3.4 FORMATAÇÃO DE TEXTO E IMAGENS

Agora que já conseguimos definir a estrutura básica de uma página html, pode-se iniciar a inserir conteúdo na mesma. Para isso, utiliza-se um conjunto de tags que tem por objetivo estruturar e organizar os textos inseridos na página web.

Nota 1: Como o objetivo primordial do HTML é estruturar o conteúdo da página, não existem muitas tags que apliquem formatação de texto. Esta tarefa de formatação de texto foi amplamente descontinuada com o HTML 5, deixando a mesma para o CSS, o qual será visto no próximo capítulo.

Como estamos dando os primeiros passos com a programação web, neste momento utilizaremos algumas tags de formatação de texto em HTML para termos a possibilidade de adicionar algum conteúdo da página.

Nota 2: Todos os exemplos de código que serão vistos a partir de agora não apresentam acentuação propositalmente, pois em momento oportuno serão apresentadas maneiras de adicionar acentos e caracteres especiais no hml.

- **Tag para inclusão de títulos**

Sempre que um conteúdo textual, como um documento ou um artigo for criado, sempre existe a necessidade de definir-se um título para o mesmo. Estes títulos podem ser adicionados em diversos pontos do documento, de forma que em alguns locais o título apresenta uma maior prioridade (destaque) do que outros.

Para representar estes títulos em um documento descrito na linguagem html, devem ser empregadas as tags **<h1>**, **<h2>**, **<h3>**, **<h4>**, **<h5>** ou **<h6>**.

O texto inserido entre estas tags será interpretado pelo html como sendo um título, recebendo maior ou menor destaque de acordo com a tag empregada. Exemplo: A tag **<h1>** refere-se a um título de maior prioridade que um título adicionado com a tag **<h2>** e assim sucessivamente. Exemplo de código:

```
<html>
  <head>
  </head>
  <body>
    <h1>Titulo de primeiro nivel</h1>
    <h2>Titulo de segundo nivel</h2>
    <h3>Titulo de terceiro nivel</h3>
    <h4>Titulo de quarto nivel</h4>
    <h5>Titulo de quinto nivel</h5>
    <h6>Titulo de sexto nivel</h6>
  </body>
</html>
```

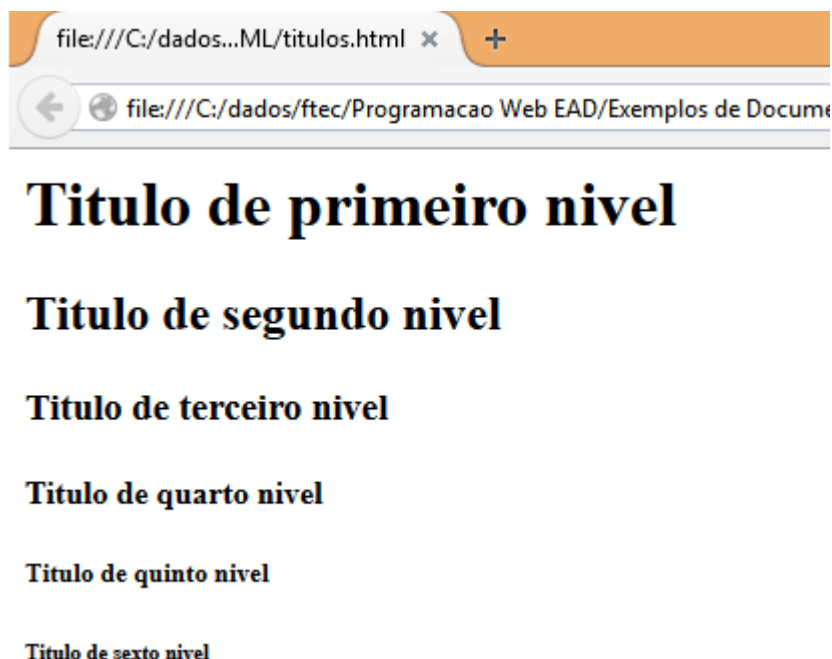


Figura 2 - Visualização no Browser do código acima

- Tag para inclusão de parágrafos e quebra de linhas

Durante a elaboração de documentos, existe a necessidade de adicionarmos parágrafos para o documento. Ainda nos documentos também há a necessidade de adicionar uma quebra de linha que não seja um parágrafo.

As tags para inclusão de parágrafos e quebra de linha forçada são respectivamente **<p>** e **
**. Exemplo de código html com parágrafos:

```
<html>
  <head>
  </head>
  <body>
    <h1>Residuo eletronico</h1>
    <p>
      Residuo computacional tambem conhecido como Residuo eletronico
      ou lixo eletronico, conhecidos pelo acronimo de REEE
      (Residuo de Equipamentos Eletronicos) 1 e o termo utilizado para
      qualificar equipamentos eletroeletronicos descartados ou obsoletos2.
      A definicao inclui computadores, televisores,telemoveis/celulares,
      entre outros dispositivos.
      A classificacao dos produtos por categoria pode ser encontrada no
      site da Comunidade Europeia 3
    </p>
    <p>
      Origem: Wikipedia, a enciclopedia livre.
    </p>
  </body>
</html>
```

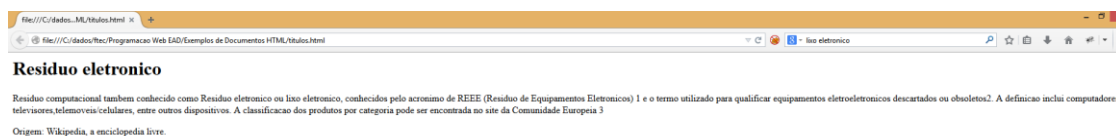


Figura 3 - Visualização no browser do código acima

Percebe-se pelo exemplo que o browser interpretou apenas as tags `<p>` no código html para fazer a quebra de linha ignorando as quebras de linhas adicionadas no código HTML. Esta situação é um padrão no HTML, de forma que qualquer que seja a formatação, espaçamento ou quebra de linha que seja adicionado ao texto, sem o uso de uma tag html ou css será sempre ignorado pelo mesmo.

Caso houver a necessidade de forçar uma quebra de linha e esta quebra não é um parágrafo, deve-se utilizar a tag `
`, conforme segue o exemplo:

```
<html>
  <head>
  </head>
  <body>
    <h1>Residuo eletronic</h1>
    <p>
      Residuo computacional tambem conhecido como Residuo eletronic
      ou lixo eletronic, conhecidos pelo acronimo de REEE
      (Residuo de Equipamentos Eletronicos) 1 e o termo utilizado
      para qualificar equipamentos eletroeletronicos descartados ou
      obsoletos2.<br>
      A definicao inclui computadores,
      televisores,telemoveis/celulares, entre outros
      dispositivos.<br>
      A classificacao dos produtos por categoria pode ser encontrada
      no site da Comunidade Europeia 3
    </p>
    <p>
      Origem: Wikipedia, a enciclopedia livre.
    </p>
  </body>
</html>
```



Figura 4 - Visualização no browser do código acima

Pela figura 4 pode-se perceber que quando adicionada a tag `
` ocorre de forma forçada a quebra da linha no texto. A maior diferença do uso da tag `
` em relação a tag `<p>` é a sua função semântica para o navegador e não apenas o espaçamento diferenciado entre as linhas conforme pode ser observado.

A identificação semântica permite ao navegador saber que aquele conteúdo que está envolvido em uma tag do tipo `<p>` é um parágrafo, enquanto que se o texto apenas for inserido e

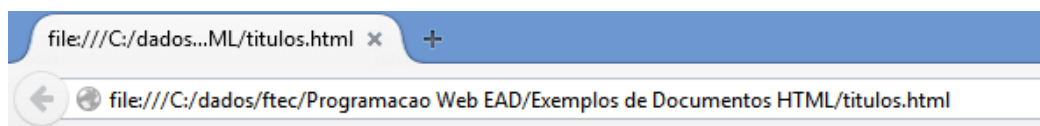
adicionadas tags do tipo **
** para fazer a sua quebra, apesar de visualmente entendermos que tratam-se de parágrafos, o navegador não terá este entendimento, provocando com isso possíveis divergências de renderização e dificuldade da página ser reconhecida da forma correta pelos mecanismos de busca.

- **Tags para destacar textos**

Com o html, pode-se destacar um texto de quatro formas diferentes: Pode-se tornar um bloco de texto em negrita, itálico, sublinhado ou ainda riscado. As tags para aplicar este tipo de formatação são as seguintes:

Texto em Negrito: Tag . Exemplo:

```
<html>
  <head>
  </head>
  <body>
    <h1>Residuo eletronico</h1>
    <p>
      <b>Residuo computacional</b> tambem conhecido como
      Residuo eletronico ou lixo eletronico....
    </p>
    <p>
      Origem: Wikipedia, a enciclopedia livre.
    </p>
  </body>
</html>
```



Residuo eletronico

Residuo computacional tambem conhecido como Residuo eletronico ou lixo eletronico....

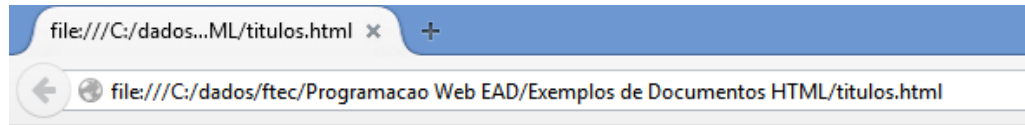
Origem: Wikipedia, a enciclopedia livre.

Figura 5 - Visualização de conteúdo em negrito

Texto em Itálico: Tag <i>. Exemplo:

```
<html>
  <head>
  </head>
  <body>
    <h1>Residuo eletronico</h1>
    <p>
```

```
<i>Residuo computacional</i> tambem conhecido como
Residuo eletronicou lixo eletronicou....
</p>
<p>
Origem: Wikipedia, a enciclopedia livre.
</p>
</body>
</html>
```



Residuo eletronicou

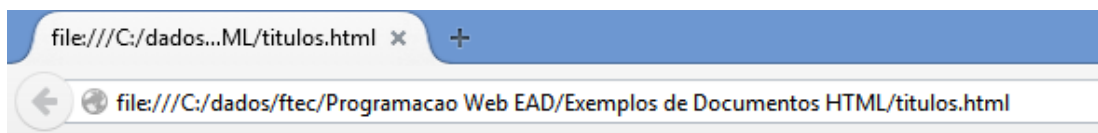
Residuo computacional tambem conhecido como Residuo eletronicou lixo eletronicou....

Origem: Wikipedia, a enciclopedia livre.

Figura 6 - Visualização de conteúdo em itálico

Texto Sublinhado: Tag <u>. Exemplo:

```
<html>
<head>
</head>
<body>
<h1>Residuo eletronicou</h1>
<p>
<u>Residuo computacional</u> tambem conhecido como
Residuo eletronicou lixo eletronicou....
</p>
<p>
Origem: Wikipedia, a enciclopedia livre.
</p>
</body>
</html>
```



Residuo eletronicou

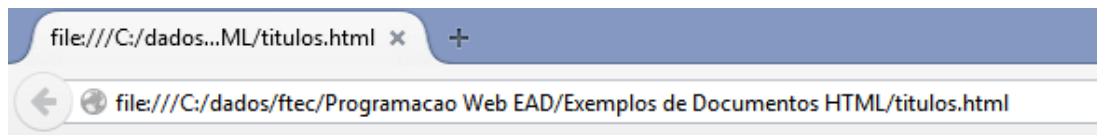
Residuo computacional tambem conhecido como Residuo eletronicou lixo eletronicou....

Origem: Wikipedia, a enciclopedia livre.

Figura 7 - Visualização de conteúdo sublinhado

Texto Riscado: Tag <s>. Exemplo:


```
<html>
  <head>
  </head>
  <body>
    <h1>Residuo eletronico</h1>
    <p>
      <s>Residuo computacional</s> tambem conhecido como
      Residuo eletronico ou lixo eletronico....
    </p>
    <p>
      Origem: Wikipedia, a enciclopedia livre.
    </p>
  </body>
</html>
```



Residuo eletronico

~~Residuo computacional~~ tambem conhecido como Residuo eletronico ou lixo eletronico....

Origem: Wikipedia, a enciclopedia livre.

Figura 8 - Visualização de conteúdo riscado

Deve-se observar que estes elementos de formatação também podem ser aplicados de forma combinada conforme segue exemplo:

```
<html>
  <head>
  </head>
  <body>
    <h1>Residuo eletronico</h1>
    <p>
      <b><i>Residuo computacional</i></b> tambem conhecido como
      <u>Residuo eletronico
      ou <s>lixo eletronico</s></u>....
    </p>
    <p>
      Origem: Wikipedia, a enciclopedia livre.
    </p>
  </body>
</html>
```



Figura 9 - Visualização com múltiplas tags de formatação

- Tag para alterar a fonte de um texto

Para alterar características da fonte de um determinado bloco de texto em uma página deve-se aplicar a tag . Esta tag permite através do emprego de suas propriedades alterar a família da fonte, bem como o seu tamanho e cor conforme segue no exemplo:

```
<html>
  <head>
  </head>
  <body>
    <h1>Residuo eletronico</h1>
    <p>
      Residuo computacional tambem conhecido como Residuo
      eletronico
      ou lixo eletronico....
    </p>
    <p>
      <font size="1" color="red" face="verdana">
        Origem: Wikipedia, a enciclopedia livre.
      </font>
    </p>
  </body>
</html>
```

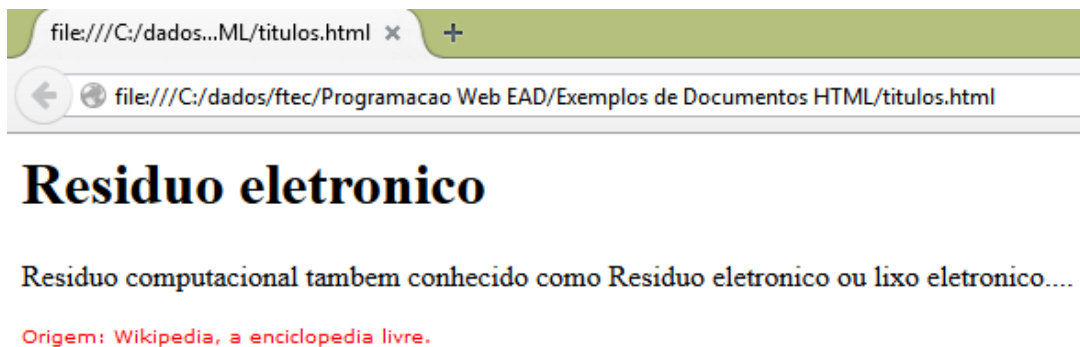


Figura 10 - Visualização do emprego da tag font

Conforme pode-se observar a tag `font` somente funciona se utilizada em conjunto com as suas propriedades. Não existe ordem específica para inclusão destas propriedades e também não há necessidade de inclusão de todas as propriedades de uma única vez. Abaixo segue o detalhamento de cada uma das propriedades.

Face: Permite definir as fontes que podem ser aplicadas ao elemento. Podem ser aplicadas mais de uma fonte desde que o nome delas seja separado por vírgula;

Size: Permite definir o tamanho da fonte. Nesta propriedade é aceito um valor de um a sete, sendo que o valor padrão é três;

Color: Permite aplicar uma cor ao texto. Pode ser aplicada uma cor sólida ou uma cor em formato hexadecimal conforme tabela 1 deste documento.