

*Identifique sua apostila*

Nome: \_\_\_\_\_

Turma: \_\_\_\_\_

# **LP II – Ap HTML5**

**Apostila HTML 5 – Disciplina LP II (PHP).**

**Conteúdo: HTML5. Curso: Informática –**

**Turmas: 72A, 72B, 72C, 72D**

**Profª Ariane Scarelli e Profª Kátia Zambon**

**2017**

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>1</b>
<b>2</b>	<b>HTML</b>	<b>1</b>
<b>2.1</b>	<b>Tags</b>	<b>1</b>
<b>2.2</b>	<b>Editor padrão TXT</b>	<b>1</b>
<b>2.3</b>	<b>Conceitos Importantes</b>	<b>2</b>
<b>2.3.1</b>	<b>W3C</b>	<b>2</b>
<b>2.3.2</b>	<b>XHTML</b>	<b>2</b>
<b>2.3.3</b>	<b>XML</b>	<b>2</b>
<b>2.3.4</b>	<b>CSS</b>	<b>3</b>
<b>2.3.5</b>	<b>JavaScript</b>	<b>3</b>
<b>2.4</b>	<b>Desenvolvimento em Camadas</b>	<b>3</b>
<b>3</b>	<b>HTML5</b>	<b>4</b>
<b>3.1</b>	<b>Elementos em Bloco (block) e em Linha (inline)</b>	<b>7</b>
<b>3.2</b>	<b>Cabeçalho &lt;head&gt; - metadados</b>	<b>7</b>
<b>3.3</b>	<b>Corpo &lt;body&gt; - dados</b>	<b>15</b>
<b>3.3.1</b>	<b>Títulos</b>	<b>15</b>
<b>3.3.2</b>	<b>Quebra de textos, espaços em branco e outros elementos</b>	<b>16</b>
<b>3.3.3</b>	<b>Listas</b>	<b>19</b>
<b>3.3.4</b>	<b>Links</b>	<b>22</b>
<b>3.3.5</b>	<b>Tabelas</b>	<b>25</b>
<b>3.3.6</b>	<b>Imagens e Objetos</b>	<b>29</b>
<b>3.3.7</b>	<b>Frames</b>	<b>34</b>
<b>3.3.8</b>	<b>Formulários</b>	<b>36</b>
<b>4</b>	<b>REFERÊNCIAS BIBLIOGRÁFICAS</b>	<b>61</b>
<b>5</b>	<b>SITES ÚTEIS</b>	<b>62</b>
<b>6</b>	<b>APÊNDICES</b>	<b>63</b>
	APÊNDICE A – Exemplos de formatação de tabelas	63
	APÊNDICE B – Lista de <i>tags</i> da linguagem HTML5	67
	APÊNDICE C – Lista de <i>tags</i> da linguagem HTML5 ordenadas por função	70
	APÊNDICE D – Lista de acentos e caracteres especiais (entities) em HTML	75
	APÊNDICE E – XML	79
	APÊNDICE F – PATTERN	84

Todos os direitos reservados e protegidos por Lei de Direito Autoral, de 19/02/1998, artigo 7º. Nenhuma parte desse material poderá ser reproduzida ou transmitida sem citação de autoria.

## 1 INTRODUÇÃO

Nosso estudo é voltado à programação Web, para isso trabalharemos os conceitos da linguagem de marcação HTML5 e da linguagem de programação PHP na criação de páginas estáticas e dinâmicas e a interação com banco de dados. Aprenderemos a acessar um servidor remoto e a transferir arquivos de forma segura.

## 2 HTML

Iniciaremos o nosso estudo com a linguagem HTML (**HyperText Markup Language** – Linguagem de Marcação de Hipertexto) que é a principal linguagem para a criação de páginas Web, e foi criada em 1990. Com ela é possível construir documentos com títulos, parágrafos, listas, links, tabelas, formulários, incorporar vídeos e imagens. Nos códigos escritos em HTML é possível incluir outras linguagens, como PHP e JavaScript, por exemplo.

O HTML foi desenvolvido para ser entendido de forma universal, para fazer com que a informação publicada seja acessível em diversos tipos de dispositivos, com características diferentes, independentemente do tamanho da tela ou resolução, móveis ou não.

A marcação da informação, ou seja, as *tags* é o que dá significado (semântica) ao conteúdo de uma página, tanto para os navegadores (*browsers*) que irão executá-lo, quanto para os robôs que irão armazenar as informações para busca, por exemplo, no Google.

### 2.1 Tags

A linguagem HTML é escrita na forma de *tags* delimitadas pelos sinais `< >` (início da *tag*) e `< />` (fim da *tag*), que identificam a função e o conteúdo de cada elemento da linguagem. Os nomes das *tags* foram escolhidos levando-se em conta o seu objetivo.

Exemplos:

```
<p> Estas são tags delimitadoras de parágrafo </p>
<title> Agora, as tags delimitadoras de título do documento HTML </title>
<html> Tags que identificam o começo e o fim do documento HTML </html>
```

### 2.2 Editor padrão TXT

Para construirmos nossos códigos em HTML utilizaremos qualquer editor que armazene o conteúdo exclusivamente no padrão texto (TXT). Isto significa que editores do tipo MS-Word ou Writer (do pacote LibreOffice) não são adequados para esse fim porque incluem caracteres de formatação de texto. Você pode utilizar o Notepad, por exemplo, ou outro editor de sua preferência desde que salve os dados no padrão texto. A extensão dos arquivos será:

.html ou .htm para documento em HTML puro;

.php quando criarmos códigos misturando as linguagens PHP e HTML.

## 2.3 Conceitos Importantes

Alguns conceitos são importantes e necessários para a nossa compreensão:

### 2.3.1 W3C

Significa World Wide Web Consortium<sup>1</sup>, uma organização internacional que estabelece as normas para a Web (Web Standards). Desenvolve protocolos e diretrizes para levar a Web ao seu potencial máximo, sempre crescente. Na construção de *sites* recomenda usar marcação semântica, códigos estruturados e separar apresentação de marcação e comportamento (veja resumo na Figura 2).

### 2.3.2 XHTML

Significa Extensible HyperText Markup Language, uma linguagem de marcação com os mesmos elementos da linguagem HTML, mas com formatação mais rigorosa e estruturada. Por exemplo, as *tags* devem ser em letras minúsculas, devem estar todas aninhadas, *tags* de fechamento devem sempre ser usadas, mesmo para elementos vazios (exemplo: `<br />`). A sintaxe deve estar correta. Em HTML um navegador (*browser*) procura resolver os erros de marcação (sintaxe), mas diferentes navegadores podem exibir as páginas de diferentes modos. Em XHTML não são admitidos esses erros. O padrão tem que estar de acordo com as recomendações para XML 1.0.

### 2.3.3 XML

Significa Extensible Markup Language, uma linguagem de marcação em que é possível definir ou criar os próprios elementos de marcação (*tags*), por isso, é expansível. Por dar preferência ao uso de “elementos” em vez de “atributos” sua codificação é mais clara, organizada e legível e os dados tornam-se mais fáceis para separação e armazenamento.

Um documento XML é apresentado de forma simples, com raiz (elemento principal) e elementos filho codificados em cores, um sinal de mais (+) ou menos (-) no canto superior esquerdo, ou ainda, um triângulo para baixo (▼) ou um triângulo para o lado (►) no canto superior esquerdo, dependendo do navegador utilizado. Não há formatação, a menos que se utilize estilo CSS (que será descrito no item 2.3.4). Por que um documento XML é apresentado desta forma? Porque *tags* podem ser criadas ou “inventadas” e os

<sup>1</sup> W3C: <http://www.w3.org/>

navegadores não sabem distinguir se uma *tag* como `<table>`, por exemplo, descreve uma tabela HTML ou uma mesa de jantar! Sendo assim, um documento XML não carrega informações sobre como apresentar os dados e a maioria dos navegadores os apresenta do jeito que foram inseridos.

Você encontrará mais informações no APÊNDICE E – XML.

### 2.3.4 CSS

Significa **C**ascading **S**tyle **S**heets – Folha de Estilo em Cascata, uma linguagem de estilo utilizada para a construção de leiautes com definições de cores e fontes para formatação e apresentação dos dados de uma página web.

Exemplo 3:

```
1 body{
2   color: #000000;
3   font: arial;
4   font-size: 10pt;
5   background: FFFFFFFF;
6 }
```

### 2.3.5 JavaScript

É uma linguagem de programação interpretada que funciona no navegador (*client side* ou *front-end*), sem passar pelo servidor (*server side* ou *back-end*), e é utilizada para dar ao usuário interatividade (Figura 1).

Tipo de linguagem	Onde funciona	Exemplo
<i>Client Side</i>	Dispositivo do usuário	JavaScript
<i>Server Side</i>	Servidor	PHP

Figura 1 – Tipos de linguagem (fonte: Flatschart (2011), p. 12)

## 2.4 Desenvolvimento em Camadas

O desenvolvimento *Client Side* em camadas possibilita modularização e independência na programação web. As três camadas são: informação (ou marcação ou conteúdo), formatação (ou estilo) e comportamento. A Figura 2 mostra um resumo das camadas.

Camada	Linguagem	Finalidade
Informação (marcação)	HTML	Permite a correta visualização e compreensão do conteúdo e interfere diretamente na ação dos mecanismos de busca.
Formatação (estilo)	CSS	Responsável pela formatação da apresentação do conteúdo (leiaute, cores, fontes).
Comportamento	JavaScript	Adiciona interatividade e controle aos elementos do HTML.

Figura 2 – Camadas de desenvolvimento *clients side* (fonte: adaptado de Flatschart (2011), p. 13)

### 3 HTML5

Esta é a versão da linguagem de marcação que utilizaremos em nosso curso!

Recomendada como padrão em 28 de outubro de 2014 pelo W3C, foi desenvolvida a partir de 2004 pelo grupo WHATWG (Web Hypertext Application Technology Working Group) formado pelas empresas Opera, Mozilla e Apple. Seu reconhecimento pelo W3C só ocorreu posteriormente, em 2006.

A linguagem HTML5 traz novas *tags*, modifica algumas e retira totalmente outras. Uma de suas principais propostas é o de enriquecer o conteúdo semântico dos documentos da web e, além disso, eliminar o uso de *plug-ins* (como *Flash*, *SilverLight*). No entanto, a estrutura básica da linguagem HTML5 continua sendo a mesma das versões anteriores, com exceção na escrita do *Doctype* que foi simplificada.

A estrutura básica de um documento HTML5 é composta por:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="utf-8">
  <title>Título ou nome do documento que será mostrado na aba do navegador</title>
</head>
<body>
  conteúdo do documento...
</body>
</html>
```

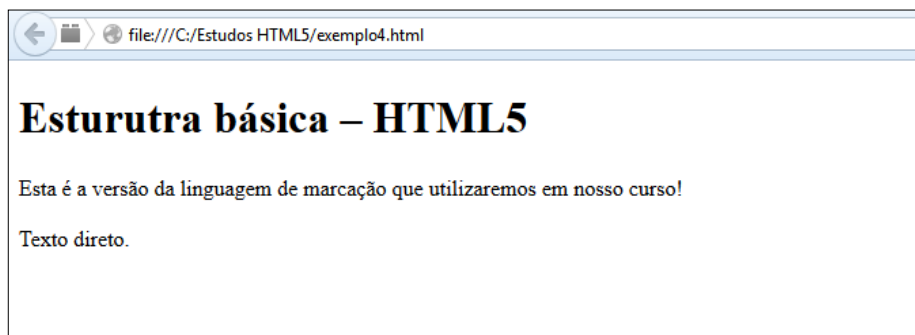
Exemplo 4 – uso de marcação de comentários `<!--` e `-->`:

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8">
5   <!-- observe o fechamento da tag acima com o sinal de maior -->
6   <meta name="author" content="Profa Ariane e Profa Kátia" />
7   <!-- agora observe o fechamento da tag acima com o caractere barra e o sinal de maior.
8   O HTML5 aceita ambas as formas. O XHTML, apenas a segunda forma -->
```

```

9 <title>Estrutura básica de HTML5</title>
10 </head>
11 <body>
12 <h1>Esturutra básica – HTML5</h1>
13 <p> Esta é a versão da linguagem de marcação que utilizaremos em nosso curso!</p>
14 Texto direto.
15 </body>
16 </html>

```

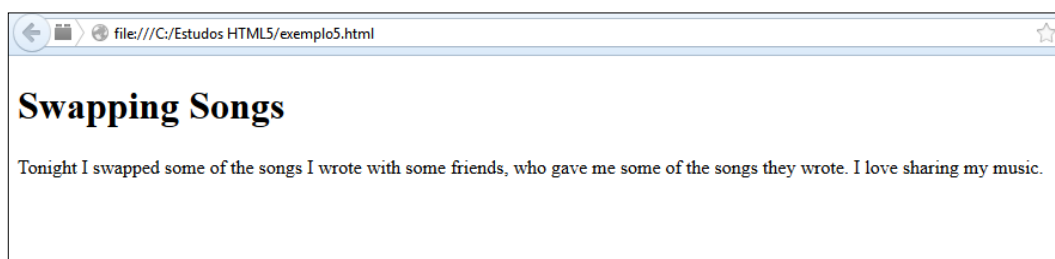


Exemplo 5<sup>2</sup> - setar o idioma para inglês:

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <title>Swapping Songs</title>
5 </head>
6 <body>
7 <h1>Swapping Songs</h1>
8 <p>Tonight I swapped some of the songs I wrote with some friends, who
9 gave me some of the songs they wrote. I love sharing my music.</p>
10 </body>
11 </html>

```



## EXERCÍCIOS:

- ☞ Para os exercícios utilize o editor Notepad (padrão do Windows), Notepad2 ou Notepad++ (ambos gratuitos, com sintaxe destacada para html), ou ainda um editor de sua preferência, desde que TXT.
- ☞ Salve o arquivo com a extensão .html (ou .htm).

<sup>2</sup> Fonte: <<http://www.w3.org/TR/html5/semantics.html#semantics>>, acesso em 23 dez. 2013.

- ☞ Para testar a validade de um código HTML5 acesse <http://validator.w3.org/> e escolha a guia 'Validate by Direct Input'. Cole seu código na área indicada e faça a validação, ou...
- ☞ Acesse <https://html5.validator.nu/>. No select do 'Validator Input' escolha a opção 'Text Field'. Cole seu código na área indicada e faça a validação.

1) Digite o código abaixo, salve como exerc1.html e execute-o no navegador.

```
<!DOCTYPE html>
<html>
<head>
  <title>Meu Título</title>
</head>
<body>
  <!--Comentário -->
  <h1>Meu Cabeçalho</h1>
  <p>Meu Primeiro Parágrafo</p>
</body>
</html>
```

2) Faça um doc em html5 cujo título da página seja "Aula de HTML", coloque um comentário com a data de hoje e no corpo da página apareça seus dados pessoais como nome, número, turma.

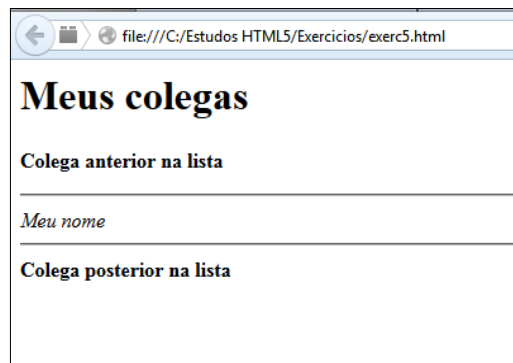
3) Escreva um doc em html5 para:

- a) Setar o idioma em português (exemplo 5 está em inglês)
- b) Utilizar o padrão de caracteres Unicode UTF-8 (exemplo 4)
- c) Inserir o título: Minha Primeira Página Web
- d) No corpo do código insira a mensagem "Hello World!" como um texto direto.

4) Nos Apêndices B e C há uma lista de *tags* da linguagem html5 que nos fornece recursos para serem implementados no nosso código. Escreva um doc em html com o título da página como "Um animal de estimação", no cabeçalho coloque a categoria do animal (Gato, Cachorro, Pássaro ou Outro a sua escolha). No corpo do texto, insira uma linha para cada característica do animal, como cor, pelagem, idade, raça e outros. Para cada característica insira uma linha de separação.

5) Utilize os recursos de negrito e itálico para escrever o nome do colega anterior ao seu número, seu nome e o nome do colega posterior a você na lista de chamada, todos com uma linha horizontal separando os nomes. Exemplo:





- 6) Faça um doc em html5 e apresente um jogo de sua preferência. No título da página coloque uma referência adequada ao conteúdo da página a ser exibida. No cabeçalho, o nome do jogo. No corpo, coloque dados como ano de criação, autor ou companhia de distribuição, síntese do jogo e quantas horas por semana você joga. Inclua formatações como negrito, itálico, linhas de separação, mudança de linha e outros recursos (Apêndices B e C).

### 3.1 Elementos em Bloco (block) e em Linha (inline)

Um navegador renderiza os elementos HTML em dois tipos de conteúdos: em blocos e em linha.

Um elemento em bloco ocupa uma linha inteira ou mais da página e descreve um agrupamento que pode ser visível ou não. Pode conter outros elementos em bloco dentro dele e quaisquer elementos em linha. Exemplos de elementos de bloco: `<p>`, `<ul>`, `<blockquote>`, `<table>`, `<div>`, `<header>`, `<nav>`, `<article>`, `<footer>`, `<section>`, `<aside>`, `<hgroup>`.

Um elemento em linha geralmente marca apenas o espaço que o elemento necessita e aparece em pequenos trechos de texto dentro de blocos, ou não, e pode incluir outros elementos em linha. Exemplos de elementos em linha: `<a>`, `<strong>`, `<em>`, `<span>`, `<img>`, `<b>`, `<abbr>`.

### 3.2 Cabeçalho `<head>` - metadados

O elemento *head* oferece informações ao navegador e aos mecanismos de busca. Se pensarmos que publicar conteúdos na rede é um processo relativamente simples, temos que nos conscientizar que controlar e acessar esta informação não seja assim tão fácil. Utilizar elementos que proporcionem recursos para a busca otimiza este processo. No escopo deste elemento podemos inserir outros elementos, que são: *title*, *meta*, *script*, *style*, *link* e *base*.

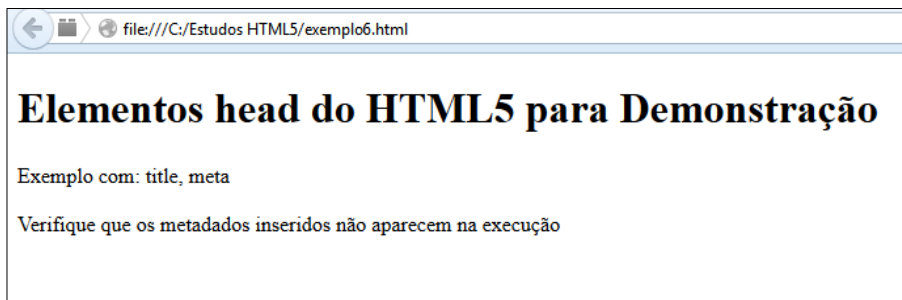
Elemento	Uso	Finalidade
<title>	Necessário	Define o título do documento que será mostrado na aba ( <i>tab</i> ) do navegador; pode conter todo tipo de caractere; importante para a indexação de conteúdos pelos mecanismos de busca.
<meta>	Recomendado	Conjunto de metadados de um documento; informações que o navegador utiliza, mas que não são, necessariamente, apresentados ao usuário; Exemplos: keywords, description, author, robots; definição de código padrão de caracteres (charset).
<script>	Opcional	Adiciona códigos geralmente em JavaScript (camada de comportamento) no documento para gerar interatividade; a inserção direta não é recomendada pelas boas práticas.
<style>	Opcional	Adiciona códigos CSS (camada de formatação ou estilo) aplicáveis aos elementos do corpo do documento, como cor, fonte, alinhamento; a inserção direta não é recomendada pelas boas práticas.
<link>	Opcional	Informa ao navegador que há outros documentos relacionados ao documento em uso; não cria um link visível para o usuário, mas gera metadados para o navegador interpretar; muito utilizado para ligar o doc atual a documentos externos CSS e JavaScript.
<base>	Padrão	Define o endereço base de todos os links relativos à página; recursos externos que serão incorporados no documento; quando não especificado assume o valor da página atual como padrão.

#### Exemplo 6 – meta e title:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <meta name="author" content="Profa Ariane e Profa Kátia" />
6   <meta name="description" content="Exemplo - uso de elementos head" />
7   <meta name="keywords" content="Exemplo, HTML5, CTI" />
8   <title>Elementos head do HTML5</title>
9 </head>
10 <body>
11   <h1>Elementos head do HTML5 para demonstração </h1>
12   <p> Exemplo com: title, meta</p>
13   <p> Verifique que os metadados inseridos não aparecem na execução </p>
14 </body>
15 </html>

```



Algumas considerações:

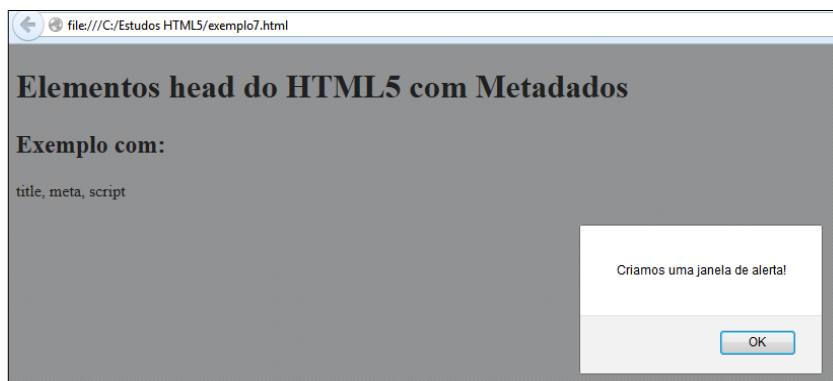
- A etiqueta "author" apresenta a identificação de responsabilidade intelectual pelo conteúdo.
- A etiqueta "description" fornece uma indicação do conteúdo da página, quando do resultado de pesquisa realizada por um mecanismo de busca. Pode ser uma palavra, frase ou parágrafo breve sobre o documento.
- A etiqueta "keywords" apresenta relação de palavras-chaves apropriadas à descrição do conteúdo da página. Recomenda-se, também, a inclusão de sinônimos. Seu uso é importante por ser uma fonte utilizada pelos mecanismos de busca para indexação dos documentos digitais.

Exemplo 7 – meta, title e script javascript interno:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <meta name="author" content="Profa Ariane e Profa Kátia" />
6   <meta name="description" content="Exemplo - uso de elementos head" />
7   <meta name="keywords" content="Exemplo, HTML5, CTI" />
8   <title>Elementos head do HTML5</title>
9   <script type="text/javascript">
10     window.onload = function() {
11       alert("Criamos uma janela de alerta!");
12     }
13   </script>
14   <!-- a inserção direta acima do código javascript não é recomendada nas melhores práticas,
15       melhor seria uma inserção de código externo como:
16       <script src="DocJavaScript.js" type="text/javascript"> </script>
17   -->
18 </head>
19 <body>
20   <h1>Elementos head do HTML5</h1>
21   <h2>Exemplo com:</h2>
22   <p>title, meta, script</p>
23 </body>
24 </html>

```



Exemplo 8 – meta, title, script javascript externo e noscript:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <meta name="author" content="Profa Ariane e Profa Kátia" />
6   <meta name="description" content="Exemplo - uso de elementos head" />
7   <meta name="keywords" content="Exemplo, HTML5, CTI" />
8   <title>Elementos head do HTML5</title>
9   <script src="DocJavaScript.js" type="text/javascript">
10  </script>
11 </head>
12 <body>
13
14   <h1>Elementos head do HTML5 e javascript </h1>
15   <h2>Exemplo com:</h2>
16
17   <p>title, meta, script, noscript</p>
18   <noscript>Seu navegador não suporta JavaScript ou está desabilitado.</noscript>
19 </body>
20 </html>

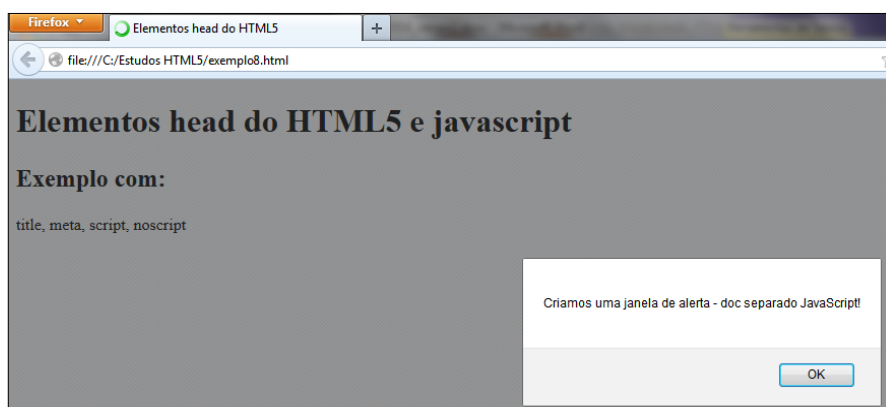
```

#### Conteúdo do arquivo DocJavaScript.js:

```

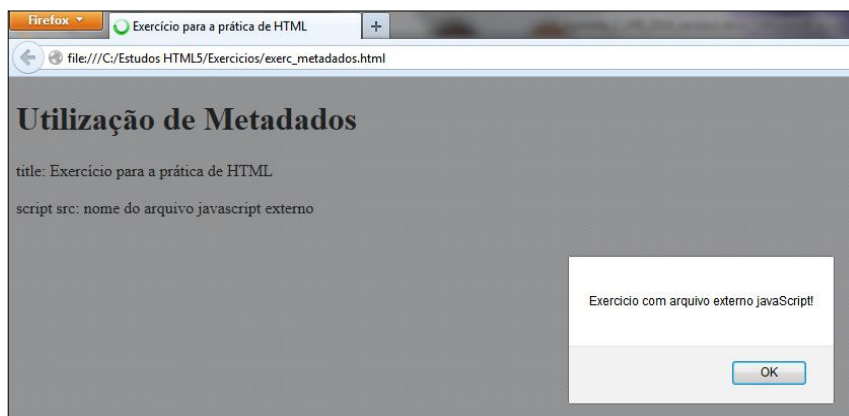
1 /* javascript */
2 window.onload = function() {
3   alert("Criamos uma janela de alerta - doc separado JavaScript!");
4 }

```



## EXERCÍCIOS:

- 7) Acesse <http://validator.w3.org/> ou <https://html5.validator.nu/> e digite o código do exemplo 6 para testar.
- 8) Copie e cole o exemplo 6 validado em um editor TXT, novo documento. Após salvar, execute-o no navegador.
- 9) Digite os exemplos 7 e 8, separadamente, e execute-os no navegador (se necessário, faça a validação).
- 10) Faça um doc em html5 que apresente no seu navegador a tela abaixo. Utilize, como no exemplo 8, metadados que definam o nome do autor (identificando você), uma breve descrição e palavras-chaves como metadados. Para a execução da janela utilize um arquivo javascript externo.

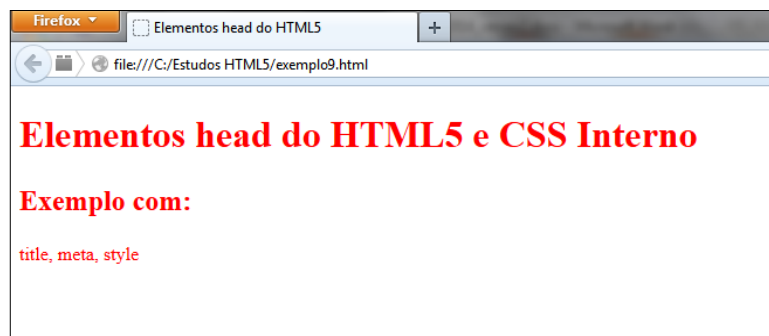


### Exemplo 9 – meta, title e style css interno:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <meta name="author" content="Profa Ariane e Profa Kátia" />
6   <meta name="description" content="Exemplo - uso de elementos head" />
7   <meta name="keywords" content="Exemplo, HTML5, CTI" />
8   <title>Elementos head do HTML5</title>
9   <style type="text/css">
10     body {color:red}
11   </style>
12   <!-- a inserção direta do código css no script acima não é recomendada nas melhores práticas,
13     melhor seria uma inserção de código externo css
14   -->
15 </head>
16 <body>
17   <h1>Elementos head do HTML5 e CSS Interno</h1>
18   <h2>Exemplo com:</h2>
19   <p>title, meta, style</p>
20 </body>
21 </html>

```



Exemplo 10 – meta, title e style css interno e externo:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <meta name="author" content="Profa Ariane e Profa Kátia" />
6   <meta name="description" content="Exemplo - uso de elementos head" />
7   <meta name="keywords" content="Exemplo, HTML5, CTI" />
8   <title>Elementos head do HTML5</title>
9   <style type="text/css">
10     body {color:white}
11   </style>
12   <!-- a inserção direta do código css no script acima não é recomendada nas melhores práticas,
13     melhor seria uma inserção de código externo css, como abaixo:
14   -->
15   <link rel="stylesheet" type="text/css" href="arqCSS.css" />
16 </head>
17 <body>
18   <h1>Elementos head do HTML5 com CSS interno e externo</h1>
19   <h2>Exemplo com:</h2>
20   <hr>
21   <p>title, meta, style, link</p>
22   <hr /> <!-- fechamento de tag padrão XHTML -->
23 </body>
24 </html>

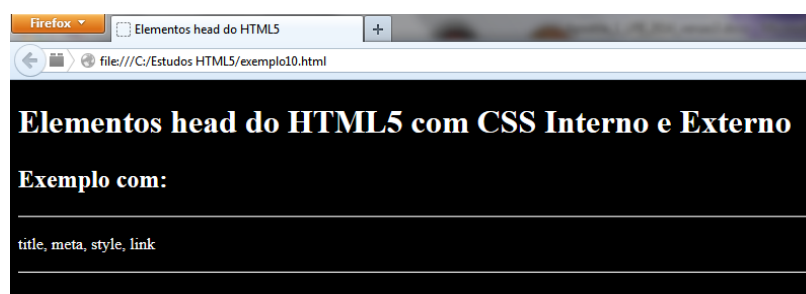
```

Conteúdo do arquivo arqCSS.css:

```

1 body {
2     background-color:black
3 }

```



Observe que a tag `<hr />` foi finalizada no padrão XHTML. No html5 funciona sem a barra de fechamento.

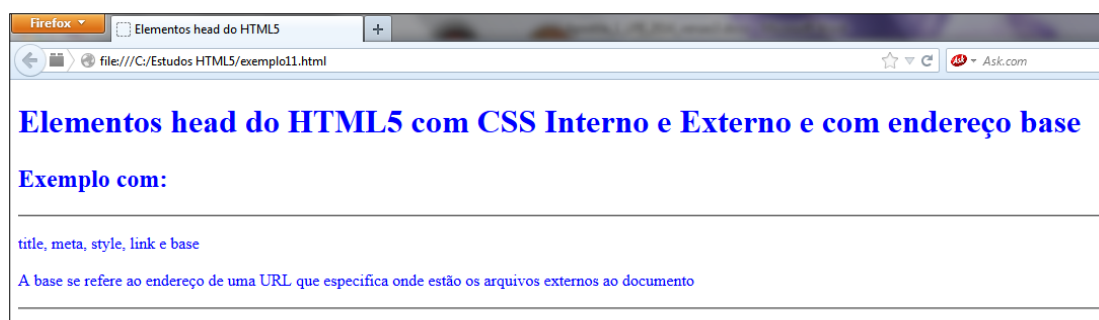
`<hr />` - fechamento de tag no padrão XHTML

Exemplo 11 – meta, title, style css interno e externo e base:

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <meta name="author" content="Profa Ariane e Profa Kátia" />
6   <meta name="description" content="Exemplo - uso de elementos head" />
7   <meta name="keywords" content="Exemplo, HTML5, CTI" />
8   <title>Elementos head do HTML5</title>
9   <style type="text/css">
10     body {color:blue}
11   </style>
12   <!-- a inserção direta do código css no script acima não é recomendada -->
13   <base href="www.cti.feb.unesp.br/" /> <!-- a partir deste ponto o endereço relativo de
14     busca de arquivos externos passa a ser essa URL -->
15   <link rel="stylesheet" type="text/css" href="arqCSS.css" /> <!-- como arqCSS.css não
16     se encontra em www.cti.feb.unesp.br o estilo não será aplicado, será ignorado. -->
17 </head>
18 <body>
19   <h1>Elementos head do HTML5 com CSS Interno e Externo e com endereço base </h1>
20   <h2>Exemplo com:</h2>
21   <hr>
22   <p>title, meta, style, link e base</p>
23   <p>A base se refere ao endereço de uma URL que especifica onde estão os arquivos externos ao
24 documento</p>
25   <hr>
26 </body>
27 </html>
```

Conteúdo do arquivo arqCSS.css:

```
1 body {background-color:black}
```





## EXERCÍCIOS:

11) Digite os exemplos 9, 10 e 11, separadamente, e execute-os no navegador (se necessário, faça a validação em <http://validator.w3.org/>).

12) Escreva um doc em html5 para:

- a) Setar o idioma em português
- b) Utilizar o padrão de caracteres Unicode UTF-8
- c) Inserir o título: Minha Segunda Página Web – uso de elementos head
- d) Defina seu nome na autoria da página
- e) Adicione uma descrição da finalidade desta página
- f) Acrescente palavras-chaves que identifiquem bem a página para os buscadores
- g) Crie um script externo em javascript, como segue:

-Nome do arquivo: MeuJS.js

-Conteúdo a ser digitado:

```
document.write("<h1>Isto é um cabeçalho</h1>");
document.write("<p>Isto é um parágrafo.</p>");
```

- h) Crie um estilo (style) externo em CSS, como segue:

-Nome do arquivo: MeuCSS.css

-Conteúdo a ser digitado:

```
h1{ /*todo cabeçalho tipo h1*/
    color:white;
    background-color:green;
}
p{ /*todo parágrafo*/
    color:#ff0000; /*vermelho*/
    font-style:italic;
    background-color:#ccc; /*cinza*/
}
```

- i) No corpo do doc insira a mensagem "Hello World in CTI!" como um parágrafo.



### 3.3 Corpo <body> - dados

O elemento *body* contém as informações que serão exibidas ao usuário pelo navegador. Essas informações podem aparecer na forma de textos, listas, links, tabelas, imagens e objetos incorporados.

#### 3.3.1 Títulos

Identificam as seções de um documento. Existem 6 tipos que variam de 1 a 6.

<h1> – título primário de um documento; utilizado por mecanismos de busca; recomenda-se que apenas um título h1 deva ser inserido por página.

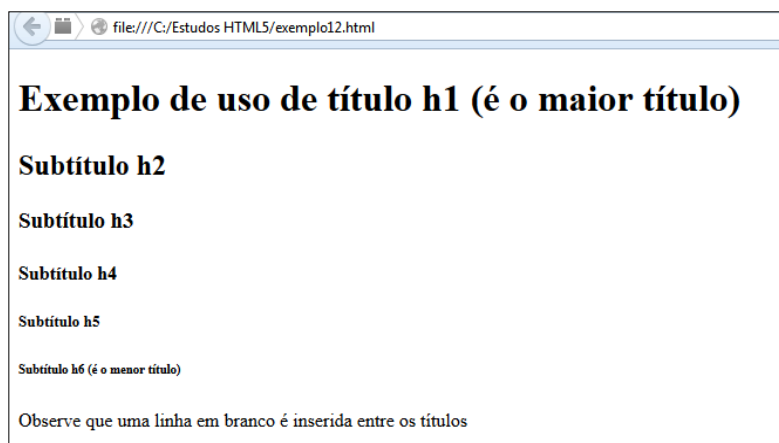
<h2> - <h6> – subtítulos

Exemplo 12 – título e subtítulos:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Elementos head do HTML5</title>
6 </head>
7 <body>
8   <h1>Exemplo de uso de título h1 (é o maior título) </h1>
9   <h2>Subtítulo h2</h2>
10  <h3>Subtítulo h3</h3>
11  <h4>Subtítulo h4</h4>
12  <h5>Subtítulo h5</h5>
13  <h6>Subtítulo h6 (é o menor título) </h6>
14  <p> Observe que uma linha em branco é inserida entre os títulos </p>
15 </body>
    </html>

```



### 3.3.2 Quebra de textos, espaços em branco e outros elementos

Espaços em branco geralmente são ignorados nos elementos em linha e em blocos e são substituídos por um único caractere branco. Se for necessário que se conserve os espaços em branco na quantidade assinalada, o elemento `<pre>` deverá ser utilizado. Para uma quebra de linha explícita, usaremos o elemento `<br>`. Para a inserção de uma linha horizontal, usaremos `<hr>`.

Os caracteres de entidade (*entities*) são conjuntos de caracteres que dizem ao navegador que caractere especial deverá ser renderizado. No APÊNDICE D – Lista de acentos e caracteres especiais (*entities*) em HTML – é possível visualizar esse conjunto de caracteres para utilizar quando necessário.

Outros elementos que alteram a forma como um texto será mostrado numa página e suas funções são descritas na tabela abaixo:

Elemento	Finalidade
<code>&lt;sup&gt;</code> e <code>&lt;sub&gt;</code>	Caracteres sobrescrito e subscrito, respectivamente; utilizados em equações matemáticas e outros.
<code>&lt;cite&gt;</code>	Utilizado para fazer citações de trabalhos de terceiros num documento; citar a origem do trabalho.
<code>&lt;blockquote&gt;</code>	Utilizado para definir uma seção citada de outra fonte; um bloco extraído do trabalho de terceiro e inserido no documento atual.
<code>&lt;q&gt;</code>	Utilizado para definir uma citação curta em um documento.
<code>&lt;abbr&gt;</code>	Utilizado para abreviações e acronismos.
<code>&lt;em&gt;</code>	Utilizado para enfatizar um texto.
<code>&lt;strong&gt;</code>	Utilizado para definir um texto como importante.
<code>&lt;code&gt;</code> <code>&lt;sample&gt;</code>	Alteração de fonte utilizada para indicar bloco de códigos de programação e saídas de computadores.
<code>&lt;kbd&gt;</code> <code>&lt;var&gt;</code>	Alteração de fonte utilizada para indicar entrada via teclado e variáveis de códigos de programação.

Exemplo 13 – quebras, espaço e outros elementos de texto:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Quebras, espaços e outros elementos de texto</title>
6 </head>
7 <body>
8   <p>
9     <acronym title="Universidade Estadual Paulista">Unesp</acronym> é uma
10      das maiores universidades do Estado de <abbr title="São Paulo">SP</abbr>.
11   </p>
12   <p>
13     <cite>Tim Berners Lee</cite> foi o criador da World Wide Web e do HTML.
14     <blockquote cite="http://pt.wikipedia.org/wiki/Html">
15       As primeiras versões do HTML foram definidas com regras sintáticas flexíveis, o que
16       ajudou aqueles sem familiaridade com a publicação na Web. Através do tempo,<br>
17       a utilização de ferramentas para autoria de HTML aumentou, assim como a
18       tendência em tornar a sintaxe cada vez mais rígida. Apesar disso, por questões
19       históricas (retrocompatibilidade), os navegadores ainda hoje conseguem
20       interpretar páginas web que estão longe de ser um código HTML válido<sup>1,2</sup>.
21     </blockquote>
22   </p>
23   <p>
24     <em>Texto enfatizado</em><br>
25     <strong>Texto importante</strong><br>
26     <code>Um pedaço de código de programação: <br>
27       <pre>
28         int calculo(int a1, int b1)
29         {
30             int c1;
31             c1 = a1 * b1;
32             return c1;
33         } </pre>
34     </code><br>
35     <samp>Exemplo ou amostra de código de programação</samp><br>
36     <kbd>Entrada pelo teclado</kbd><br>
37     <var>Variável</var>
38   </p>
39 </body>
40 </html>

```

Unesp é uma das maiores universidades do Estado de SP.

Tim Berners Lee foi o criador da World Wide Web e do HTML.

As primeiras versões do HTML foram definidas com regras sintáticas flexíveis, o que ajudou aqueles sem familiaridade com a publicação na Web. Através do tempo, a utilização de ferramentas para autoria de HTML aumentou, assim como a tendência em tornar a sintaxe cada vez mais rígida. Apesar disso, por questões históricas (retrocompatibilidade), os navegadores ainda hoje conseguem interpretar páginas web que estão longe de ser um código HTML válido<sup>1,2</sup>.

*Texto enfatizado*

**Texto importante**

Um pedaço de código de programação:

```
int calculo(int a1, int b1)
{
    int c1;
    c1 = a1 * b1;
    return c1;
}
```

Exemplo ou amostra de código de programação

Entrada pelo teclado

Variável

## EXERCÍCIOS:

**13)** Faça um doc que apresente as seguintes estruturas no formato html5 utilizando os elementos apresentados acima:

- a) H<sub>2</sub>O
- b) Prof<sup>a</sup> Ariane e Prof<sup>a</sup> Kátia
- c) Uma área pode ser medida em m<sup>2</sup> e seu volume em m<sup>3</sup>

**14)** Escreva um doc em html5 que contenha as referências abaixo:

- a) Título da página: Informações importantes do CTI
- b) Cabeçalho: Escola de Ensino Profissionalizante
- c) Insira a sigla CTI com a descrição "Colégio Técnico Industrial – Prof. Isaac Portal Roldán"
- d) Coloque o texto como citação: "O CTI é uma das três unidades de ensino médio e profissionalizante da Universidade Estadual Paulista. Criado no dia 7 de abril de 1967, o Colégio foi a primeira unidade da Unesp no câmpus de Bauru. Seu primeiro diretor foi Isaac Portal Roldán, que hoje dá nome ao Colégio"
- e) Coloque a fonte de onde foi retirado o texto acima: [www.cti.feb.unesp.br](http://www.cti.feb.unesp.br)
- f) Apresente como um texto importante os três cursos que atualmente são oferecidos no CTI: Eletrônica, Informática e Mecânica (um em cada linha) e enfatize o curso de vocês

### 3.3.3 Listas

Há basicamente três tipos de listas em HTML (a descrição das *tags* está nos Apêndices B e C):

- Listas ordenadas: exibição de uma lista com itens ordenados numérica ou alfabeticamente; `<ol> <li> </ol>`
- Listas não ordenadas: exibição de itens de maneira sequencial; separação de itens através de caractere em forma de bola, quadrado etc.; `<ul> <li> </ul>`
- Listas de descrição: utilizadas para descrever termos; `<dl> <dt> <dd>`.

Exemplo 14 – listas ordenadas, não ordenadas e de descrição:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Listas ordenadas, não ordenadas e de descrição</title>
6 </head>
7 <body>
8   <h1>Listas</h1>
9   <h2>Lista ordenada</h2>
10  <ol>
11    <li>Abacaxi</li>
12    <li>Limão</li>
13    <li>Laranja
14      <ol>
15        <li>Pera</li>
16        <li>Bahia</li>
17      </ol>
18    </li>
19  </ol>
20  <h2>Lista não ordenada</h2>
21  <ul>
22    <li>Abacaxi</li>
23    <li>Limão</li>
24    <li>Laranja
25      <ul>
26        <li>Pera</li>
27        <li>Bahia</li>
28      </ul>
29    </li>
30  </ul>
31  <h2>Lista ordenada</h2>
32  <ol type="A">
33    <li>Morango</li>
34    <li>Maçã
35      <ol type="i">
36        <li>vermelha</li>
37        <li>verde</li>
38      </ol>
39    </li>
40    <li>Cereja</li>
41  </ol>

```

```

42 <h2>Lista não ordenada</h2>
43 <ul type="circle"> <!-- Obsoleto; alterar em CSS -->
44 <li>Morango</li>
45 <li>Maçã
46 <ul type="square"> <!-- Obsoleto; alterar em CSS -->
47 <li>vermelha</li>
48 <li>verde</li>
49 </ul>
50 </li>
51 <li>Cereja</li>
52 </ul>
53 <h2>Lista de descrição</h2>
54 <dl>
55 <dt>Abacaxi</dt>
56 <dd>fruta ácida de tom amarelado e casca dura</dd>
57 <dt>Limão</dt>
58 <dd>fruta ácida e azeda de tom verde</dd>
59 <dt>Laranja</dt>
60 <dd>fruta ácida de cor laranja e doce</dd>
61 </dl>
62
63 </body>
64 </html>

```

Firefox

file:///C:/Estudos HTML5/exemplo14.html

## Listas

### Lista ordenada

1. Abacaxi
2. Limão
3. Laranja
  1. Pera
  2. Bahia

### Lista não ordenada

- Abacaxi
- Limão
- Laranja
  - Pera
  - Bahia

### Lista ordenada

- A. Morango
- B. Maçã
  - i. vermelha
  - ii. verde
- C. Cereja

### Lista não ordenada

- Morango
- Maçã
  - vermelha
  - verde
- Cereja

### Lista de descrição

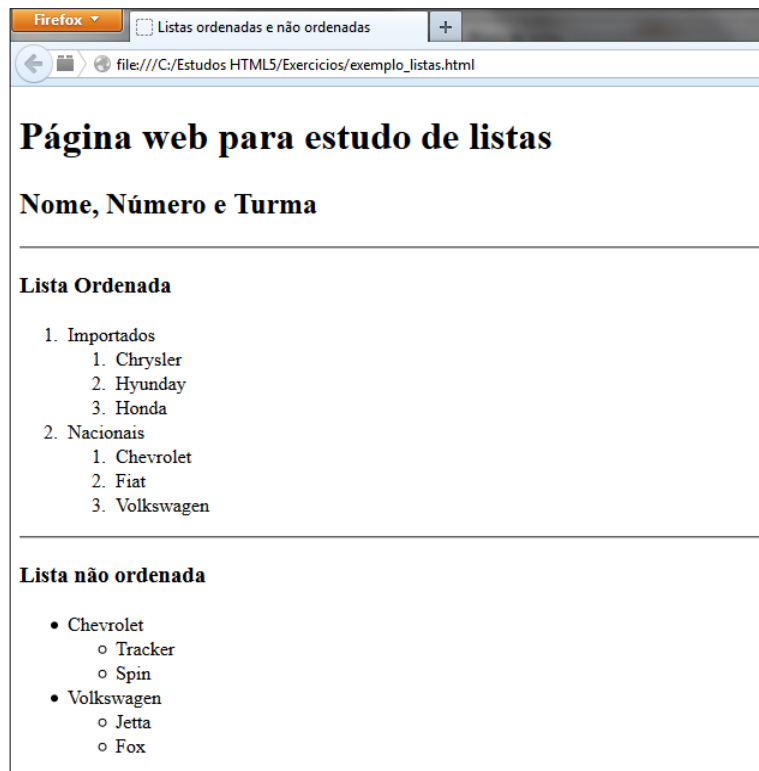
Abacaxi  
fruta ácida de tom amarelado e casca dura

Limão  
fruta ácida e azeda de tom verde

Laranja  
fruta ácida de cor laranja e doce

## EXERCÍCIOS:

- 15) Escreva um doc em html5 que apresente uma página web com as listas abaixo representadas. Inclua para cada lista mais um item de sua preferência. Atente para os nomes de página e dos cabeçalhos, completando o que foi solicitado.



- 16) Faça uma página web que apresente uma lista ordenada contendo 3 *tags* de funções básicas, 3 *tags* de formatação e 3 *tags* de metadados. Suas descrições podem ser encontradas nos Apêndices B e C. Exemplo:

### 1. Tags Básicas

1. <!--...--> define um comentário
2. <!DOCTYPE> define um document type

### 2. Tags de formatação

1. <b> define texto em negrito
2. <i> define texto em itálico

- 17) Faça outra página web que apresente listas não ordenadas contendo 4 *tags* de funções de listas e 4 *tags* de formulários. Suas descrições podem ser encontradas nos Apêndices B e C.

- 18) Escreva um doc em html5 que apresente uma lista de descrição contendo 3 *tags* de funções de frames e 3 *tags* de funções de formulários. Suas descrições podem ser encontradas nos Apêndices B e C.

### 3.3.4 Links

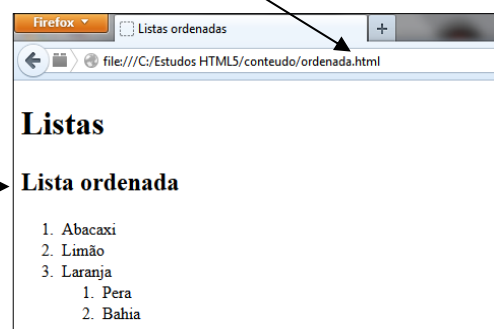
Os links ou ligações entre conteúdos podem ser de dois tipos: relativos ou absolutos. O link relativo remete a uma página dentro do próprio servidor da página em uso, podendo ser na mesma página, na mesma pasta ou em diferentes pastas. O link absoluto remete a um local em outro servidor e requer a especificação completa da URL. O link pode aparecer na forma de texto, imagem etc.

Exemplo 15 – links relativos e absolutos:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Links</title>
6 </head>
7 <body>
8   <h1> Estudo para Uso de Links</h1>
9   <h2>Listas do Exemplo Anterior</h2>
10  <ul>
11    <!-- 3 links relativos para a pasta chamada conteudo-->
12    <li><a href=".\\conteudo\\ordenada.html">Lista ordenada</a></li>
13    <li><a href=".\\conteudo\\nao_ordenada.html">Lista não ordenada</a></li>
14    <li><a href=".\\conteudo\\descricao.html">Lista de descrição</a></li>
15    <!-- 1 link absoluto -->
16    <li><a href="http://www.w3schools.com/html/default.asp">Para saber mais</a></li>
17  </ul>
18 </body>
19 </html>

```



Exemplo 16 – links relativos para a mesma página:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Links</title>
6 </head>
7 <body>

```



```

8 <h2>Extraído da Wikipedia</h2>
9 <h2>Definindo alguns termos técnicos</h2>
10 <ul>
11     <!-- observe o uso do caractere # -->
12     <li><a href="#Info1">Informática</a></li>
13     <li><a href="#Computacao">Ciência da computação</a></li>
14 </ul>
15
16
17 <a name="Info1" title="Definição de informática">
18     <h2>Informática</h2>
19 </a>
20 <p>
21 Informática é um termo usado para descrever o conjunto das ciências da informação,
22 estando incluídas neste grupo: a ciência da computação, a teoria da informação, o
23 processo de cálculo, a análise numérica e os métodos teóricos da representação dos
24 conhecimentos e da modelagem dos problemas.
25 </p>
26
27 <p> <a href="#">Voltar ao topo</a> </p>
28
29 <a name="Computacao" title="Definição de ciência da computação">
30     <h2>Ciência da Computação</h2>
31 </a>
32 <p>
33 Ciência da computação é a ciência que estuda as técnicas, metodologias e instrumentos
... computacionais, que automatiza [...] e computação móvel que têm sido utilizadas por bilhões de
pessoas ao redor do globo.
    </p>
    <p> <a href="#">Voltar ao topo</a> </p>
</body>
</html>

```

Outros tipos de links:

- Link de e-mail:

<p><a href="mailto:ariane.scarelli@gmail.com">Fale com um autor</a></p>

- Link com atributo *target* para abrir em uma nova janela:

<p><a href="http://google.com.br" target="\_blank">Google</a></p>

Obs.: Se target="\_self" o link será aberto na mesma janela; por ser padrão, não é necessário especificar.

## Extraído da Wikipedia

### Definindo alguns termos técnicos

- [Informática](#)
- [Ciência da computação](#)

### Informática

Informática é um termo usado para descrever o conjunto das ciências da informação, estando incluídas neste grupo: a ciência da computação, a teoria da informação, o processo de cálculo, a análise numérica e os métodos teóricos da representação dos conhecimentos e da modelagem dos problemas.

O termo informática, sendo dicionarizado com o mesmo significado amplo nos dois lados do Atlântico, assume em Portugal o sentido sinônimo da ciência da computação, enquanto que no Brasil é habitualmente usado para refer especificamente o processo de tratamento da informação por meio de máquinas eletrônicas definidas como computadores.

O estudo da informação começou na matemática quando nomes como Alan Turing, Kurt Gödel, Alonzo Church e Pitágoras, começaram a estudar que tipos de problemas poderiam ser resolvidos, ou computados, por elementos humanos que seguissem uma série de instruções simples de forma, independente do tempo requerido para isso. A motivação por trás destas pesquisas era o avanço durante a revolução industrial e da promessa que máquinas poderiam futuramente conseguir resolver os mesmos problemas de forma mais rápida e mais eficaz. Do mesmo jeito que as indústrias manuseiam matéria-prima para transformá-la em um produto final, os algoritmos foram desenhados para que um dia uma máquina pudesse tratar informações. Assim nasceu a informática.

### Etimologia

A palavra surge, em meados do século XX, da junção de parte de duas outras palavras, processamento da informação por meios automáticos em simples impulsos

formada pela junção das palavras informação + automática. Pode dizer-se que informática é a informática que estuda o processamento automático de informações por meio do computador. Há uma linha de pensamento que conduz o termo "informática" à junção dos conceitos "informação" e "matemática".

[Voltar ao topo](#)

### Ciência da Computação

Ciência da computação é a ciência que estuda as técnicas, metodologias e instrumentos computacionais, que automatiza processos e desenvolve soluções baseadas no uso do processamento digital. Não se restringe apenas ao estudo dos algoritmos, suas aplicações e implementação na forma de software, extrapolando para todo e qualquer conhecimento pautado no computador, que envolve também a telecomunicação, o banco de dados e as aplicações tecnológicas que possibilitam atingir o tratamento de dados de entrada e saída, de forma que se transforme em informação. Assim, a Ciência da Computação também abrange as técnicas de modelagem de dados e os protocolos de comunicação, além de princípios que abrangem outras especializações da área. Enquanto ciência, classifica-se como ciência exata, apesar de herdar elementos da lógica filosófica aristotélica, tendo por isto um papel importante na formalização matemática de algoritmos, como forma de representar problemas decíveis, i.e., os que são susceptíveis de redução a operações elementares básicas, capazes de serem reproduzidas através de um qualquer dispositivo mecânico/eletrônico capaz de armazenar e manipular dados. Um destes dispositivos é o computador digital, de uso generalizado, nos dias de hoje. Também de fundamental importância para a área de ciência da computação são as metodologias e técnicas ligadas à implementação de software que abordam a especificação, modelagem, codificação, teste e avaliação de sistemas de software. Os estudos oriundos da ciência da computação podem ser aplicados em qualquer área do conhecimento humano em que seja possível definir métodos de resolução de problemas baseados em repetições previamente observadas. Avanços recentes na ciência da computação tem impactado fortemente a sociedade contemporânea, em particular as aplicações relacionadas às áreas de redes de computadores, Internet, Web e computação móvel que têm sido utilizadas por bilhões de pessoas ao redor do globo.

### Algoritmos

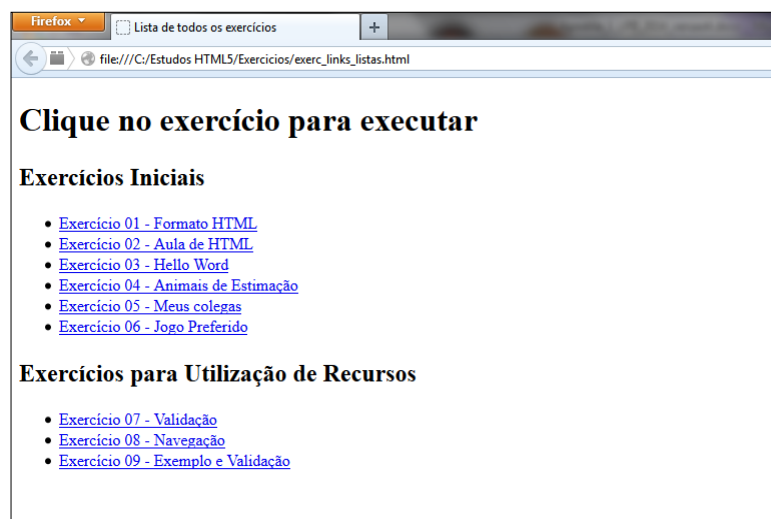
No século VII, o matemático indiano Brahmagupta explicou pela primeira vez o sistema de numeração hindu-arábico e o uso do 0. Aproximadamente em 825, o matemático persa al-Khwarizmi escreveu o livro Calculando com numerais hindus, responsável pela difusão do sistema de numeração hindu-arábico no Oriente Médio, e posteriormente na Europa. Por volta do século XII houve uma tradução do mesmo livro para o latim: Algoritmi de numero Indorum. Tais livros apresentaram novos conceitos para definir sequências de passos para completar tarefas, como aplicações de aritmética e álgebra. Por derivação do nome do matemático, atualmente usa-se o termo algoritmo.

## EXERCÍCIOS:

19) Faça uma página web que chame, através de links, os exemplos 15 e 16 relativos ao uso de links, como apresentado abaixo (verifique nome da página, cabeçalho etc):



20) Faça uma página que contenha um link para cada um dos exercícios realizados anteriormente, ou seja, serão 19 links que deverão ser agrupados como listas referentes ao assunto, como no exemplo, exercícios iniciais, exercícios para a utilização de recursos, exercícios com uso de elementos de texto, com listas:



### 3.3.5 Tabelas

Uma tabela é uma representação tabular e organizada de determinados dados.

Elementos que compõem uma tabela (<table>) são descritos abaixo:

Elemento	Finalidade
<caption>	Título da tabela.
<thead>	Indica cabeçalho da tabela
<tr>	Insere uma linha na tabela; necessário em <thead>, <tfoot> e <tbody>
<th>	Adiciona um valor à coluna do cabeçalho
<td>	Adiciona um dado à coluna da tabela em <tbody> e <tfoot>
<tfoot>	Indica rodapé da tabela
<tbody>	Indica múltiplas linhas com dados

#### Exemplo 17 – tabelas simples<sup>3</sup>:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Tabelas simples</title>
6 </head>
7 <body>
8
9 <p>
10 Cada tabela começa com uma tag. <br>
11 Cada linha da tabela começa com uma tag tr.<br>
12 Cada dado da tabela começa com uma tag td.
13 </p>
14
15 <h4>Uma coluna:</h4>
16 <table border="1">
17 <tr>
18   <td>100</td>
19 </tr>
20 </table>
21
22 <h4>Uma linha e três colunas:</h4>
23 <table border="1">
24 <tr>
25   <td>100</td>
26   <td>200</td>
27   <td>300</td>
28 </tr>
29 </table>
30
31 <h4>Duas linhas e três colunas:</h4>
32 <table border="1">
33 <tr>

```

<sup>3</sup> Fonte: adaptado de <[http://www.w3schools.com/html/html\\_tables.asp](http://www.w3schools.com/html/html_tables.asp)>, acesso em 06 jan. 2014.

```

34 <td>100</td>
35 <td>200</td>
36 <td>300</td>
37 </tr>
38 <tr>
39 <td>400</td>
40 <td>500</td>
41 <td>600</td>
42 </tr>
43 </table>
44
45 <h4>Duas linhas e três colunas sem bordas:</h4>
46 <table border="0">
47 <tr>
48 <td>100</td>
49 <td>200</td>
50 <td>300</td>
51 </tr>
52 <tr>
53 <td>400</td>
54 <td>500</td>
55 <td>600</td>
56 </tr>
57 </table>
58
59 <h4>Sem preenchimento da célula:</h4>
60 <table border="1">
61 <tr>
62 <td>Primeira</td>
63 <td>Linha</td>
64 </tr>
65 <tr>
66 <td>Segunda</td>
67 <td>Linha</td>
68 </tr>
69 </table>
70
71 <h4>Com preenchimento da célula:</h4>
72 <table border="1" cellpadding="10">
73 <tr>
74 <td>Primeira</td>
75 <td>Linha</td>
76 </tr>
77 <tr>
78 <td>Segunda</td>
79 <td>Linha</td>
80 </tr>
81 </table>
82 </body>
83 </html>

```

Firefox Tabelas simples

file:///C:/Estudos HTML5/exemplo17.html

Cada tabela começa com uma tag.  
Cada linha da tabela começa com uma tag tr.  
Cada dado da tabela começa com uma tag td.

**Uma coluna:**

Tabela  
de  
uma  
coluna

100
-----

**Uma linha e três colunas:**

100	200	300
-----	-----	-----

**Duas linhas e três colunas:**

100	200	300
400	500	600

**Duas linhas e três colunas sem bordas:**

100	200	300
400	500	600

**Sem preenchimento da célula:**

Primeira	Linha
Segunda	Linha

**Com preenchimento da célula:**

Primeira	Linha
Segunda	Linha

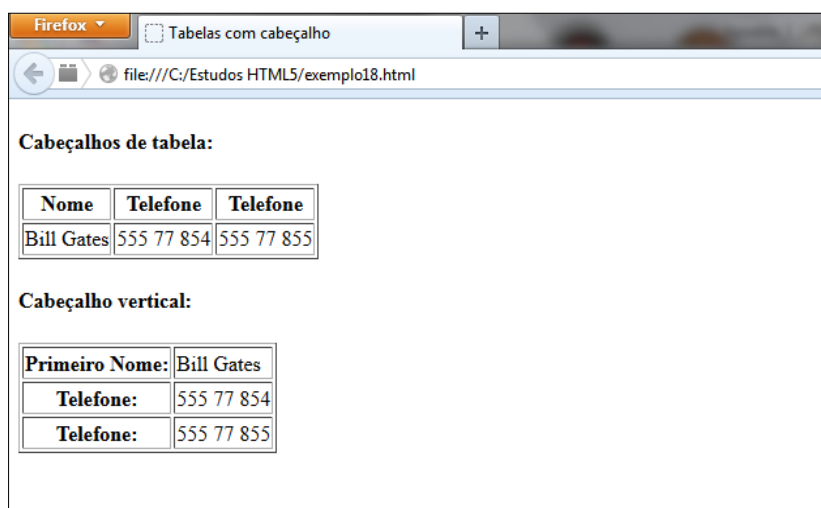
### Exemplo 18 – tabelas com cabeçalho:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Tabelas com cabeçalho</title>
6 </head>
7 <body>
8 <h4>Cabeçalhos de tabela:</h4>
9
10 <table border="1">
11 <tr>
12   <th>Nome</th>
13   <th>Telefone</th>
14   <th>Telefone</th>
15 </tr>
16 <tr>
17   <td>Bill Gates</td>
18   <td>555 77 854</td>
19   <td>555 77 855</td>
20 </tr>
21 </table>
22
23 <h4>Cabeçalho vertical:</h4>
24 <table border="1">
25 <tr>
26   <th>Primeiro Nome:</th>
27   <td>Bill Gates</td>
28 </tr>
29 <tr>
30   <th>Telefone:</th>
31   <td>555 77 854</td>

```

```
32 </tr>
33 <tr>
34   <th>Telefone:</th>
35   <td>555 77 855</td>
36 </tr>
37 </table>
38 </body>
39 </html>
```



Mais exemplos de tabelas no APÊNDICE A – Exemplos de formatação de tabelas.

## EXERCÍCIOS:

- 21) Faça uma página web que contenha uma tabela que seja relativa a um boletim com as disciplinas que você tem neste ano.

Disciplina	Med – 1ºbi	Med – 2ºbi	Med – 3ºbi	Med – 4ºbi	Med Final	Faltas

### 3.3.6 Imagens e Objetos

- <img>

O elemento <img> permite exibir imagens em uma página web. Os formatos normalmente utilizados são GIF, JPEG, PNG. A inserção de uma imagem pode ser feita com endereço relativo ou absoluto (ver item 3.3.4).

Atributos que compõem uma imagem <img> são descritos abaixo:

Atributo	Finalidade
src	Endereço da imagem ( <i>source</i> ).
alt	Texto alternativo ( <i>alternate</i> ) para ser apresentado caso a imagem não possa ser renderizada pelo navegador. É um importante atributo que deve ser incluído sempre por questões de acessibilidade.
height	Altura da imagem.
width	Largura da imagem.

Exemplo 19 – exibição simples de imagem e link de imagem:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Imagens</title>
6 </head>
7 <body>
8   <h1>Trabalhando com imagens</h1>
9   <div id="areaimagem">
10     
11     
12     <p>
13       <a href="https://www.facebook.com/cti.unesp.bauru" title="CTI no Facebook"
14 target="_blank">
15         
16       </a>
17     </p>
18   </div>
19 </body>
</html>

```





## EXERCÍCIOS:

22) Faça uma página web para exibir como imagem a sua fotografia, com as seguintes especificações:

- Título da página: Uso de Imagens
- Cabeçalho: Essa é a fotografia de: <seu nome completo>
- Insira como imagem uma foto sua no tamanho 150 x 170
- Insira no texto alternativo: foto tamanho 150x170
- Repita o item 'c' no tamanho 400 x 420
- Insira no texto alternativo: foto tamanho 400x420

23) Faça uma página web para exibir como imagem a sua fotografia, com as seguintes especificações:

- Título da página: Uso de Imagens
- Cabeçalho: "Este sou eu, <seu nome completo>"
- Insira como imagem uma foto sua no tamanho que você preferir
- Insira no texto alternativo: <seu nome completo>
- Insira um cabeçalho em tamanho menor com o texto: "Para saber mais sobre mim, acesse:"
- Insira uma imagem pequena do facebook
- Crie um link com a imagem do facebook que remeta ao seu perfil na rede
- Ao clicar na imagem do facebook o perfil deve ser aberto em uma nova janela em branco
- Insira uma imagem pequena do twitter
- Crie um link com a imagem do twitter que remeta ao seu perfil na rede (se não tiver twitter utilize o de outra rede social)
- Ao clicar na imagem do twitter o perfil deve ser aberto em uma nova janela em branco

- <object>

O elemento <object> permite incluir objetos como imagens, áudio, vídeo, componentes ActiveX<sup>4</sup>, PDF, Flash etc. em uma página web. Geralmente é utilizado para exibir conteúdos em Flash que precisam de plugin externo. Embora possa incluir objetos do tipo imagem, recomenda-se nesse caso o uso do elemento <img>, assim como, para objetos de áudio, o elemento <audio>, vídeos, o elemento <video>.

Atributos que compõem um <object> são descritos abaixo:

Atributo	Finalidade
data	Indica o arquivo; é obrigatório se o atributo <i>type</i> não for utilizado.
type	Indica o tipo de aplicação a ser carregada; é obrigatório se atributo <i>data</i> não for utilizado.
height	Altura do objeto.
width	Largura do objeto.
param	Enviar parâmetros ao objeto inserido.

<sup>4</sup> ActiveX complemento da Microsoft que controla conteúdo dinâmico

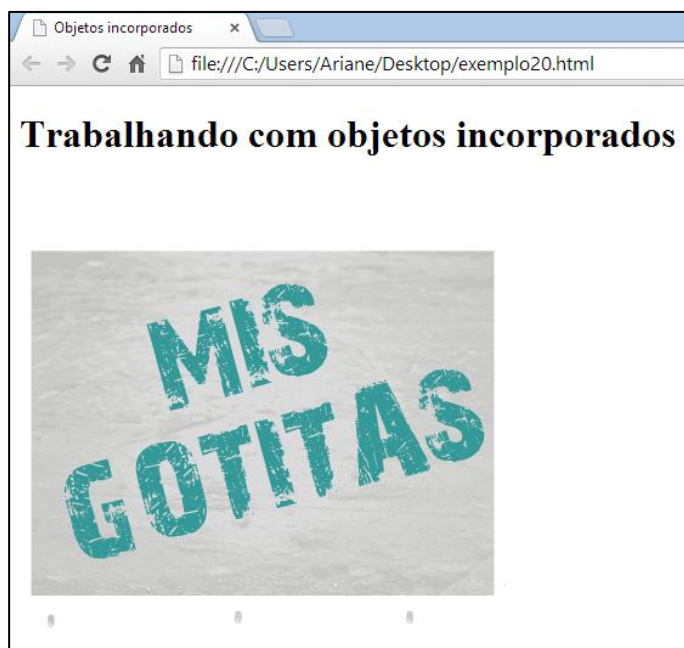
Exemplo 20 – exibição de animação Flash com <object> e mensagem caso o navegador não suporte:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Objetos incorporados</title>
6 </head>
7 <body>
8   <h1>Trabalhando com objetos incorporados</h1>
9   <div id="objetos">
10     <p>
11       <object width="400" height="400" data=".\\objetos\\MisGotas.swf">
12         <span>Seu navegador não pode exibir arquivo Flash.</span>
13       </object>
14     </p>
15   </div>
16 </body>
17 </html>

```

Se o navegador suportar plugin Flash, você verá em movimento:



Se o navegador não suportar plugin Flash, você verá:



Exemplo 21 – exibição de animação Flash com <object>, mensagem caso o navegador não suporte e com link para obter o Flash Player:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Objetos incorporados</title>
6 </head>
7 <body>
8   <h1>Trabalhando com objetos incorporados</h1>
9   <div id="objetos">
10     <p>
11       <object width="400" height="400" data=".\\objetos\\MisGotas.swf">
12         <span>Seu navegador não pode exibir arquivo Flash, obtenha clicando na
13           imagem.</span>
14         <a href="http://www.adobe.com/go/getflash/">
15           
17         </a>
18       </object>
19     </p>
20   </div>
21 </body>
22 </html>

```



- <embed>

O elemento <embed> tem a mesma finalidade do elemento <object>, permite incluir ou incorporar objetos em uma página web.

Atributos que compõem um <embed> são descritos abaixo:

Atributo	Finalidade
src	Especifica o endereço do arquivo externo para incorporar.
type	Indica o tipo de mídia do objeto incorporado.
height	Altura do objeto.

width	Largura do objeto.
-------	--------------------

Exemplo 22 – exibição de animação Flash com <embed> e mensagem caso o navegador não suporte:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Objetos incorporados</title>
6 </head>
7 <body>
8   <h1>Trabalhando com objetos incorporados</h1>
9   <div id="objetos">
10     <p>
11       <embed src=".\\objetos\\MisGotas.swf" width="400" height="400">
12       Seu navegador não pode exibir arquivo Flash.
13     </p>
14   </div>
15 </body>
16 </html>

```

Se o navegador suportar plugin Flash, você verá em movimento:



### 3.3.7 Frames

O elemento <iframe> é utilizado para mostrar *frames* (quadros) que são páginas web dentro de outra página web.

Atributos que compõem um <iframe> são descritos abaixo:

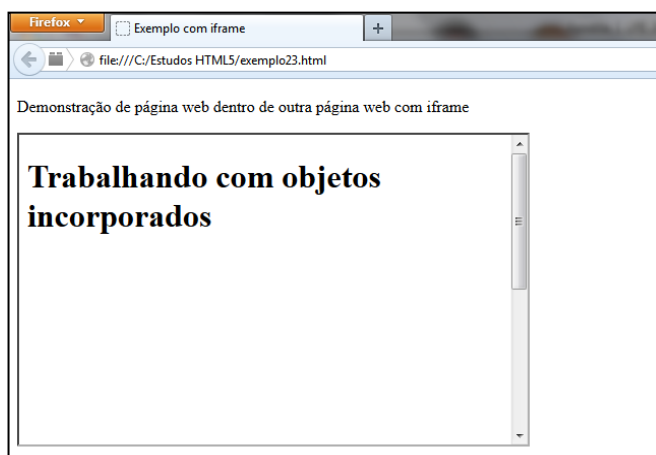
Atributo	Finalidade
src	Especifica o endereço relativo (pasta) ou absoluto (URL) da página que será exibida dentro da página atual.
value	Especificado em pixels ou porcentagem (p.ex. value="50%")
frameborder	Se igual a zero, exibe o frame sem bordas.
height	Altura do iframe.
width	Largura do iframe.

Exemplo 23 – exibição de página web (exemplo22.html) dentro de outra com <iframe>:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Exemplo com iframe</title>
6   </head>
7   <body>
8     <p>Demonstração de página web dentro de outra página web com iframe</p>
9     <iframe src="exemplo22.html" width="500" height="500"></iframe>
10  </body>
11 </html>

```



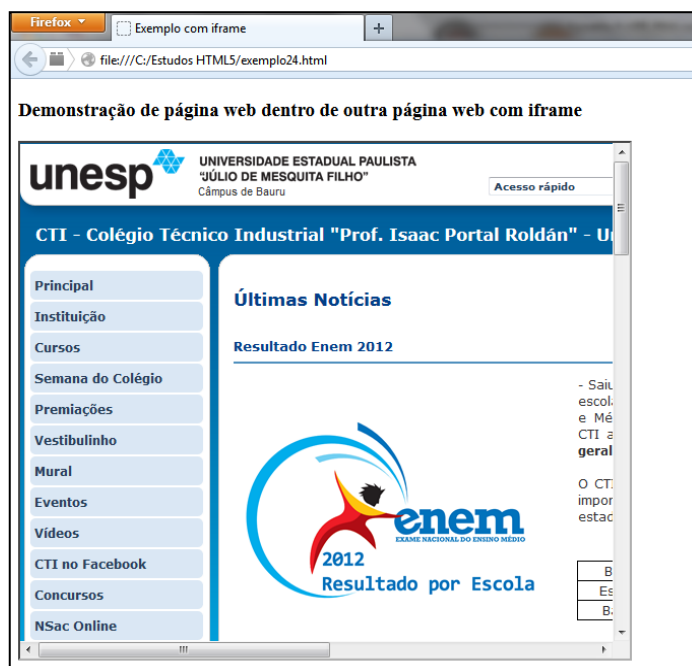
Exemplo 24 – exibição de página web (URL) dentro de outra com <iframe>:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Exemplo com iframe</title>
6   </head>
7   <body>
8     <h3>Demonstração de página web dentro de outra página web com iframe</h3>
9     <iframe src="http://www.cti.feb.unesp.br/" width="600" height="500"></iframe>

```

10 </body>  
11 </html>



## EXERCÍCIOS:

24) Faça uma página web sobre o tema relativo a um livro ou filme de sua preferência. A página deve conter pelo menos um frame, uma imagem e um objeto que vimos acima.

### 3.3.8 Formulários

Os formulários <form> são utilizados para selecionar diferentes tipos de entradas do usuário e para enviar dados ao servidor. O elemento <form> pode conter outros elementos de entrada, como descritos abaixo:

Elemento	Finalidade
<input>	O elemento mais importante de um formulário, usado para selecionar uma informação de entrada.
<textarea>	Define uma entrada com múltiplas linhas (área de texto).
<button>	Define um botão que pode ser clicado.
<select>	Define uma lista do tipo <i>drop-down</i> .

<option>	Define uma opção numa lista <i>drop-down</i> .
<optgroup>	Define um grupo de opções relacionadas em uma lista <i>drop-down</i> .
<fieldset>	Grupo de elementos relacionados em um formulário.
<legend>	Define um título para os elementos de um <fieldset>
<label>	Define um rótulo (ou etiqueta) para um elemento <input>
<datalist>	Define uma lista de opções pré-definidas para uma entrada de dados; quando utilizada ativa a função de autocompletar do <input>.
<keygen>	Especifica um campo gerador de "pares de chaves" ( <i>key-par</i> ) utilizadas pelos formulários. Quando o formulário é enviado, a chave privada ( <i>private key</i> ) é armazenada localmente e a chave pública ( <i>public key</i> ) é enviada ao servidor.
<output>	Mostra o resultado de um cálculo.

O elemento <form> utiliza diversos atributos que modificam e expandem seu funcionamento, que são:

Atributo	Finalidade
action	Especifica para onde enviar os dados quando o formulário for submetido. Pode ser um endereço absoluto ou relativo (ver item 3.3.4).
name	Especifica o nome de um formulário.
accept-charset	Especifica a codificação de caracteres a ser usada quando o formulário for submetido. Geralmente é omitido indicando a utilização da codificação padrão da página.
autocomplete	Especifica se o formulário deve ter ou não a função autocompletar ativada; as opções são <i>on</i> e <i>off</i> .
method	Especifica como os dados do formulário enviados em <i>action</i> serão submetidos. São dois os métodos HTTP: <i>get</i> e <i>post</i> .  <i>get</i> → é o método padrão. É ideal para a transação de dados que não exige segurança, como os utilizados nas pesquisas em buscadores (Google) e na solicitação de endereços de páginas. Os dados não são inseridos no banco de dados e são visíveis na barra de endereços do navegador.

	<i>post</i> → ideal para a transação de dados que exige segurança, pois os dados são enviados de forma oculta para um programa ou banco de dados.
enctype	<p>Especifica como os dados do formulário devem ser codificados quando forem submetidos ao servidor. Somente para o método <i>post</i>. Valores possíveis:</p> <p><i>application/x-www-form-urlencoded</i> → é o valor padrão; todos os caracteres são codificados antes de serem enviados.</p> <p><i>multipart/form-data</i> → nenhum caractere é codificado; utilizado quando o formulário tem um controle para <i>upload</i> de arquivos.</p> <p><i>text/plain</i> → espaços em branco são convertidos em "+", mas nenhum outro caractere é codificado.</p>
novalidate	Especifica que o formulário não deve ser validado quando submetido.
target	<p>Especifica onde apresentar o retorno dos dados do formulário submetido. Opções disponíveis:</p> <p><i>_blank</i> → retorno exibido em uma nova janela ou aba.</p> <p><i>_self</i> → retorno exibido na mesma janela (é o padrão).</p> <p><i>_parent</i> → retorno exibido em uma janela pai (janela anterior à chamada da atual).</p> <p><i>_top</i> → retorno exibido ocupa toda a área da janela atual, sobrepondo outras informações.</p>

- O elemento <input>

Esse elemento tem uma grande variedade de atributos, um dos mais importantes é o *type* (ou tipo). O atributo *type* determinará a forma e a funcionalidade do elemento <input> dentro de um formulário. Os valores que podem ser atribuídos ao *type* são chamados de *state* (estado). Os tipos mais comuns de *type* são: text, password, hidden, radio-button, checkbox, submit, button, image, e-mail, reset, file.



Exemplos de entradas com a utilização de *type* para o elemento `<input>`:

a) Campo texto

`<input type="text">` → define um campo de entrada do tipo texto onde o usuário pode digitar utilizando o teclado.

b) Campo senha

`<input type="password">` → define um campo de entrada do tipo senha. Os caracteres da senha não aparecerão no formulário.

c) Campo oculto

`<input type="hidden">` → define um campo de entrada oculto, ideal para enviar ao servidor informações que não devem ser mostradas no formulário.

d) Botões de radio

`<input type="radio">` → define um campo de entrada do tipo radio-button com um número limitado de opções onde o usuário pode escolher somente uma.

e) Caixas de check

`<input type="checkbox">` → define um campo de entrada do tipo checkbox com um número limitado de opções onde o usuário pode escolher mais de uma opção.

f) Campo e-mail

`<input type="email">` → define um campo de entrada do tipo e-mail. A validação é feita pelo próprio navegador (verifica se há @ e ponto). Não suportado no navegador Safari.

g) Botão submit

`<input type="submit">` → define um botão de envio (submit) utilizado para enviar dados de um formulário a um servidor. Os dados são enviados à página especificada no atributo *action* do formulário.

h) Imagem como botão submit

`<input type="image">` → define uma imagem como sendo um botão submit.

i) Botão button

`<input type="button">` → define um botão que pode ser utilizado para qualquer fim, mas geralmente é utilizado para executar scripts JavaScript.

j) Botão reset

`<input type="reset">` → define um botão que ao ser clicado limpa os conteúdos de todos os campos do formulário, restaurando os padrões.

k) Botão file

`<input type="file">` → define um botão que ao ser clicado abre uma janela para escolha de um arquivo para *upload*.

Exemplo 25 – Formulário simples com dois `<input>` tipo text:

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Exemplo com form</title>
6   </head>
7   <body>
8     <h3>Demonstração de página web usando elemento form</h3>
9     <form>
10      Primeiro nome: <input type="text" name="primeiro_nome"> <br>
11      Último nome: <input type="text" name="ultimo_nome">
12    </form>
13  </body>
14 </html>
```



Exemplo 26 – Formulário com `<label>` e `<input>`, com botões de radio (radio-button) e submit:

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Exemplo com form</title>
6   </head>
7   <body>
8     <h2>Demonstração de página web usando elemento form</h2>
9     <p>Clique nos labels para ver alternar a opção entre os radio-buttons:</p>
10    <form>
11      <label for="masc">Masculino</label>
12      <input type="radio" name="sexo" id="masc" value="masculino"><br>
13      <label for="femi">Feminino</label>
```

```

14     <input type="radio" name="sexo" id="femi" value="feminino"><br><br>
15     <input type="submit" value="Enviar">
16 </form>
17 </body>
18 </html>

```



No exemplo 26, a inserção do elemento `<label>` para dar nome a um campo de entrada, possibilita ao usuário clicar sobre o nome do campo para selecioná-lo. Sem o `<label>` esta ação não é possível.

Para que o elemento `<label>` ligue-se ao elemento `<input>` correto, na propriedade `id` do `<input>` especifica-se uma identidade única para ele. Em `<label>`, a propriedade `for` fará referência a identidade (`id`) do elemento `<input>` a que se deseja ligar. No exemplo 26, temos um `id="masc"` e um `for="masc"`, um `id="femi"` e um `for="femi"`.

Há outra maneira de fazer a associação entre um `<label>` e um `<input>` que é inserir o elemento `<input>` dentro do elemento `<label>`.

Veja o exemplo:

```
<label>Masculino <input type="radio" name="sexo" id="masc" value="masculino"> </label>
```

Exemplo 27 – Formulário com action para e-mail, método post, <textarea> e diversos tipos de <input>:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Exemplo com form</title>
6   </head>
7   <body>
8     <h2>Demonstração de página web usando elemento form</h2>
9     <h3>Envie e-mail para:</h3>
10    <form action="MAILTO:alguem@example.com" method="post" enctype="text/plain">
11      Nome:<br>
12      <input type="text" name="nome" value="seu nome"><br>
13      E-mail:<br>
14      <input type="email" name="mail" value="seu email"><br>
15      Comentário:<br>
16      <textarea rows="4" cols="50" placeholder="seu comentário" ></textarea><br><br>
17      <input type="submit" value="Enviar">
18      <input type="reset" value="Limpar">
19    </form>
20  </body>
21 </html>

```

No exemplo 27, o atributo *action* especifica para onde os dados serão enviados quando o formulário for submetido, neste caso, um e-mail.

Exemplo 28 – Formulário com action para arquivo HTML, método post, atributo enctype, <input> com file para upload, botões submit e reset:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Exemplo com form</title>
6   </head>

```

```

7 <body>
8   <h2>Demonstração de upload de arquivo</h2>
9   <form action="HTMLPage.html" method="post" enctype="multipart/form-data">
10     <!-- enctype="multipart/form-data" deve ser inserido quando se deseja
11       fazer upload de arquivos -->
12     Escolha o arquivo a ser carregado clicando no botão abaixo:<br>
13     <input type="file" name="arquivo"><br><br>
14     <input type="submit" value="Enviar">
15     <input type="reset" value="Limpar">
16   </form>
17 </body>
18 </html>

```



No exemplo 28:

- O atributo *action* envia os dados do formulário para um arquivo HTML localizado na mesma pasta do site.
- O método *post* faz com que os dados sejam enviados de maneira segura.
- O atributo *enctype* especifica o tipo de codificação dos dados utilizado para um formulário a ser submetido. Foi definido (a) *enctype="multipart/form-data"* porque há um *upload* a ser feito. Se nada for especificado, o padrão para o atributo é (b) *enctype="application/x-www-form-urlencoded"*. A diferença entre eles é que na forma (a) todos os dados do formulário serão enviados, inclusive o conteúdo do arquivo escolhido para *upload*; na forma (b) todos os dados do formulário serão enviados, exceto o conteúdo do arquivo escolhido para *upload*, somente o nome do arquivo será enviado. A forma (a) quebra o arquivo em múltiplas partes garantindo o seu envio e por isso deve ser utilizado em caso de *uploads*.

Além dos tipos mencionados anteriormente, há outros tipos que podem ser utilizados no elemento `<input>` com o atributo *type*, que são: *color*, *date*, *datetime*, *datetime-local*, *month*, *number*, *range*, *search*, *tel*, *time*, *url*, *week*.

Vejamos a lista com a definição desses estados (*states*) para o atributo *type*, com os respectivos navegadores que os suportam:

Estado	Finalidade
color	Controle para escolha de cor. Somente para Chrome e Opera.
date	Controle para definição de data. Somente para Chrome, Safari e Opera.
datetime	Controle para definição de data e hora no padrão UTC (Universal Time Coordinated). Somente para Safari e Opera.
datetime-local	Controle para definição de data e hora local. Somente para Chrome, Safari e Opera.
month	Controle para definição de mês e ano. Somente para Chrome, Safari e Opera.
number	<p>Define um campo para a entrada de um número. Também é possível estabelecer restrições para os números que serão aceitos. Não suportado no Firefox.</p> <p>Os seguintes atributos são utilizados para as restrições:</p> <ul style="list-style-type: none"> <li>• max – especifica o máximo valor permitido</li> <li>• min – especifica o mínimo valor permitido</li> <li>• step – especifica o salto (positivo) entre um número e outro no intervalo válido</li> <li>• value – especifica o valor inicial padrão</li> </ul>
range	<p>Exibe um controle deslizante com valores mínimos e máximos definidos.</p> <p>Os seguintes atributos são utilizados:</p> <ul style="list-style-type: none"> <li>• max – especifica o máximo valor permitido</li> <li>• min – especifica o mínimo valor permitido</li> <li>• step – especifica o salto (positivo) entre um número e outro no intervalo válido</li> <li>• value – especifica o valor inicial padrão</li> </ul>
search	Define um campo de pesquisa (como um site de busca). Somente para Chrome e Safari.
tel	Define um campo para a entrada de um número de telefone. Por enquanto, nenhum navegador suporta esse estado.
time	Controle para definição de hora local. Somente para Chrome, Safari e Opera.
url	Define um campo para a entrada de uma URL. Exige a entrada de "http://". Não suportado no Safari.
week	Controle para escolha de número da semana e ano. Somente para Chrome, Safari e Opera.

Há também atributos utilizados no elemento `<input>`. Vejamos os atributos e sua finalidade (nem todos os atributos aparecem nessa lista):

Atributo	Finalidade
accept	Especifica os tipos de arquivos que serão aceitos pelo servidor (somente para o <code>type="file"</code> ).
alt	Especifica um texto alternativo para uma imagem quando ela não for renderizada corretamente (somente para o <code>type="image"</code> ).
autocomplete	Especifica se um elemento <code>&lt;input&gt;</code> deve ter a função autocompletar habilitada.
autofocus	Especifica se um elemento <code>&lt;input&gt;</code> deve automaticamente ter o foco quando a página é carregada.
checked	Especifica se um elemento <code>&lt;input&gt;</code> deve estar pré-selecionado quando a página é carregada (para <code>type="checkbox"</code> ou <code>type="radio"</code> ).
disabled	Especifica que um elemento <code>&lt;input&gt;</code> deve estar desabilitado. Se um elemento estiver desabilitado não será enviado ao servidor.
form	Especifica um ou mais formulários onde o elemento <code>&lt;input&gt;</code> deve pertencer.
formaction	Especifica a URL do arquivo que processará a entrada a ser submetida (para <code>type="submit"</code> e <code>type="image"</code> ).
formenctype	Especifica como os dados do formulário devem ser codificados quando forem submetidos a um servidor (para <code>type="submit"</code> e <code>type="image"</code> ).
formmethod	Define o método HTTP ( <i>get</i> ou <i>post</i> ) para o envio dos dados em uma <i>action</i> (para <code>type="submit"</code> e <code>type="image"</code> ).
formnovalidate	Define que elementos do formulário não devem ser validados quando submetidos.
formtarget	Especifica onde mostrar o retorno dos dados do formulário que foram submetidos (para <code>type="submit"</code> e <code>type="image"</code> ).
height	Especifica a altura de um elemento <code>&lt;input&gt;</code> (somente para o <code>type="image"</code> ).
list	Refere-se ao elemento <code>&lt;datalist&gt;</code> que contém opções pré-definidas para um elemento <code>&lt;input&gt;</code> .
max	Define o valor máximo para um elemento <code>&lt;input&gt;</code> .
maxlength	Define o número máximo de caracteres permitido em um elemento <code>&lt;input&gt;</code> .

min	Define o valor mínimo para um elemento <input>.
multiple	Especifica que o usuário pode entrar com mais de um valor em um elemento <input>.
name	Especifica o nome de um elemento <input>.
pattern	Especifica uma expressão regular que é verificada em um elemento <input>.
placeholder	Especifica um pequeno texto (uma dica) que descreve o valor esperado para um elemento <input>, como uma dica.
readonly	Especifica que um elemento <input> é somente para leitura.
required	Especifica que um campo de entrada deve ser preenchido antes de o formulário ser submetido.
size	Especifica a largura, em caracteres, de um elemento <input>. Nas boas práticas, este atributo deve ser definido na camada de formatação (estilo) CSS.
src	Especifica a URL da imagem para um botão (para type="image").
step	Especifica os intervalos de números válidos para um campo de entrada.
value	Especifica o valor de um elemento <input>.
width	Especifica a largura de um elemento <input> (somente para o type="image").

Exemplo 29 – Formulário com método post, action para arquivo php, <select>/<option>, <optgroup>, radio-button, checkbox, password, hidden, atributos name e value:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Exemplo com form</title>
6   </head>
7   <body>
8     <h2>Demonstração de página web usando elemento form</h2>
9     <h3>Dados pessoais:</h3>
10    <form method="post" action="processa.php">
11      Primeiro nome:<input type="text" name="prim_nome"><br>
12      Último nome:<input type="text" name="ult_nome"><br><br>
13
14      <!-- campo hidden (oculto) para enviar o nome do país=Brasil como padrão para todos -->
15      <input type="hidden" name="pais" value="Brasil">
16
17      Gênero:<br>
18      Masculino:<input type="radio" name="genero" id="masc" value="masculino"><br>

```



```

19 Feminino:<input type="radio" name="genero" id="femi" value="feminino"><br><br>
20
21 E-mail:<input type="email" name="email"><br>
22 Senha:<input type="password" name="senha"><br><br>
23
24 Ocupação:<br>
25 <select id="ocupacao" name="ocupacao">
26   <optgroup label="Técnico">
27     <option label="Desenvolvedor" value="desenvolvedor"></option>
28     <option label="Web Designer" value="designer"></option>
29   </optgroup>
30   <optgroup label="Não Técnico">
31     <option label="Consultor" value="consultor" selected></option>
32     <option label="Gerente" value="gerente"></option>
33   </optgroup>
34   <option label="Outro" value="outro"></option>
35 </select>
36
37 <br><br>
38 Preferências:<br>
39 <input type="checkbox" id="pref1" name="prefs" value="newsletter">Envie-me
40 newsletter<br>
41 <input type="checkbox" id="pref2" name="prefs" value="parceiros">Envie-me ofertas de
42 parceiros<br>
43 <input type="checkbox" id="pref3" name="prefs" value="marketing">Envie-me material de
44 propaganda<br>
45
46 <br><br>
47 <input type="submit" value="Enviar">
48 <input type="reset" value="Limpar">
49 </form>
50 </body>
</html>

```

Exemplo com form

file:///C:/Users/Ariane/Desktop/exemplo29.html

### Demonstração de página web usando elemento form

**Dados pessoais:**

Primeiro nome:

Último nome:

Gênero:

Masculino: ☒

Feminino: ☐

E-mail:

Senha:

Ocupação:

Preferências:

☒ Envie-me newsletter

☒ Envie-me ofertas de parceiros

☐ Envie-me material de propaganda

Ocupação:

- Consultor
- Técnico
  - Desenvolvedor
  - Web Designer
- Não Técnico
  - Gerente
  - Outro

No exemplo 29, o campo oculto (*hidden*) não será mostrado no formulário, mas, quando este for submetido, será enviado ao servidor.

No elemento `<select>` foi definido que a opção “consultor” será a padrão para a ocupação do usuário. Observe na linha 31 como isso é feito com o uso da propriedade *selected* em destaque:

```
<option label="Consultor" value="consultor" selected></option>
```

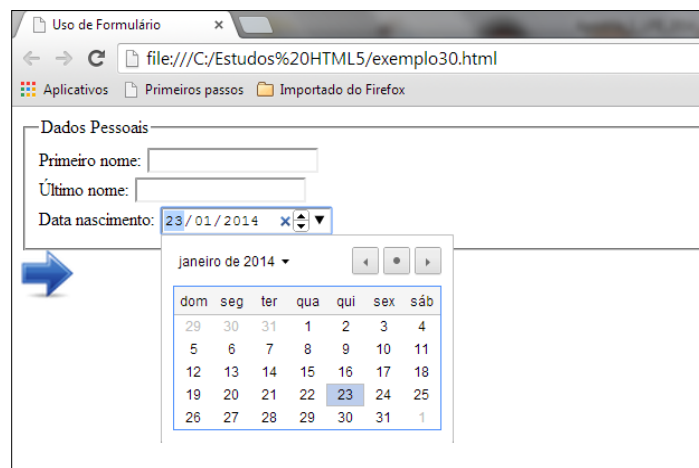
Exemplo 30 – Formulário com `<fieldset>/<legend>`, `<input>` tipo text, date, image:

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title> Uso de Formulário</title>
6   </head>
7   <body>
8     <form>
9       <fieldset>
10        <legend>Dados Pessoais</legend>
11        Primeiro nome: <input type="text" name="pnome"><br>
12        Último nome: <input type="text" name="unome"><br>
13        Data nascimento: <input type="date" name="dtnasc"><br>
14      </fieldset>
15      <input type="image" src="seta_direita_32x32.png" alt="Enviar" width="48" height="48">
16    </form>
17  </body>
18 </html>
```

No exemplo 30, o tipo *date* foi utilizado para *type*. Este recurso apresenta um formato pré-definido para uma data, geralmente dd/mm/aaaa no padrão brasileiro, e ainda exibe um calendário para facilitar a escolha. No entanto, somente os navegadores Chrome, Safari e Opera suportam o tipo *date*, os navegadores Internet Explorer e Firefox mostrarão uma entrada tipo *text* simples.

A imagem da seta é o botão *submit* para este formulário.

A imagem abaixo mostra o recurso *date* no navegador Chrome:



Exemplo 31 – Formulário com método get, action para arquivo php, <datalist>/<option>, <input>:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title> Uso de Formulário</title>
6   </head>
7   <body>
8     <form action="demo_form.php" method="get"> <!-- método get por ser padrão
9       não era necessário especificar -->
10    <input list="navegadores" name="navegador">
11    <datalist id="navegadores"> <!-- ativa o autocompletar com as opções abaixo -->
12      <option value="Internet Explorer">
13      <option value="Firefox">
14      <option value="Chrome">
15      <option value="Opera">
16      <option value="Safari">
17    </datalist><br>
18    <input type="submit"> <!-- value não especificado, assume valor padrão do browser -->
19    <input type="reset"> <!-- value não especificado, assume valor padrão do browser -->
20  </form>
21 </body>
22 </html>

```

No exemplo 31:

- foi especificado o método *get* para o envio do formulário, porém, como é o método padrão, não era necessário especificá-lo:

```
<form action="demo_form.php" method="get">
```

- a inserção do elemento <datalist> ativa o autocompletar no momento da digitação pelo teclado, com as opções dadas em <option>, se o usuário iniciar a digitação com um caractere igual à primeira letra de uma das opções:

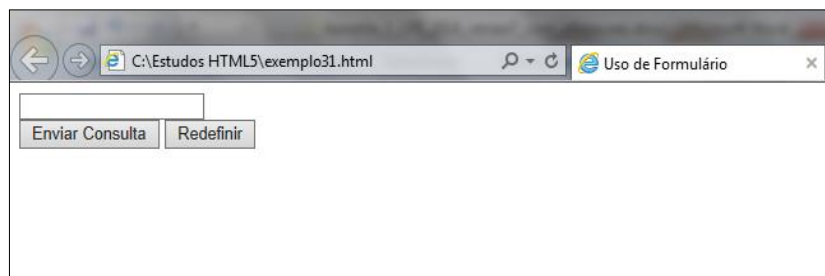
```

<datalist id="navegadores"> <!-- ativa o autocompletar com as opções abaixo -->
  <option value="Internet Explorer">
  <option value="Firefox">
  <option value="Chrome">
  <option value="Opera">
  <option value="Safari">
</datalist><br>

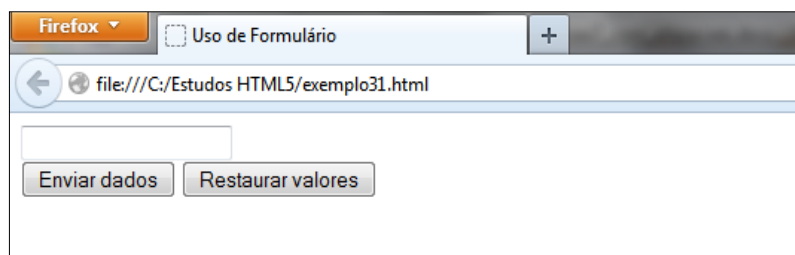
```

- Os botões “submit” e “reset” do <input>, quando não recebem valores pré-definidos em “value” para serem exibidos nos botões, assumirão os valores de texto padrão de cada um dos navegadores utilizados (Redefinir, Restaurar Valores e outros). Abaixo, veja as telas de exemplo em três navegadores:

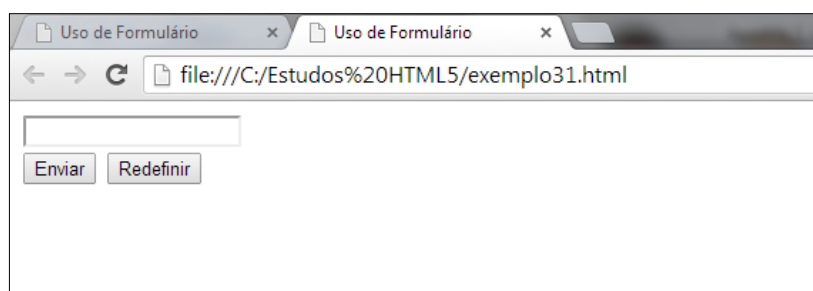
No Internet Explorer versão 11:



No Firefox versão 26.0:



No Chrome versão 32.0.1700.76:



Exemplo 32 – Formulário com diversos exemplos do uso de estados (*states*) do atributo *type*:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title> Uso de Formulário</title>
6   </head>
7   <body>
8     <h2>Alguns tipos de estado para o atributo <i>type</i> do HTML5</h2>
9     <form>
10      <label for="cor">Cor:</label>
11      <input type="color" id="cor" name="define_cor"> <br>
12      <label for="data">Data:</label>
13      <input type="date" id="data" name="data" min="2000-01-01"> <br>
14      <label for="datahora">Data e hora local:</label>
15      <input type="datetime-local" id="datahora" name="dtime"> <br>
16      <label for="mes">Mês:</label>
17      <input type="month" id="mes" name="define_mes"> <br>
18      <label for="numero">Número:</label>
19      <input type="number" id="numero" name="nume" min="2" max="15"> <br>
20      <label for="intervalo">Intervalo:</label>
21      <input type="range" id="intervalo" name="inter"> <br>
22      <label for="pesq">Pesquisar:</label>
23      <input type="search" id="pesq" name="pesquisa"> <br>
24      <label for="tel">Telefone:</label>
25      <input type="tel" id="tel" name="tel"> <br>
26      <label for="hora">Hora local:</label>
27      <input type="time" id="hora" name="horalocal"> <br>
28      <label for="site">Endereço do site:</label>
29      <input type="url" id="site" name="url"> <br>
30      <label for="semana">Número da semana do ano:</label>
31      <input type="week" id="semana" name="sema"> <br>
32    </form>
33  </body>
34 </html>

```

Exemplo 32 no Chrome:

Exemplo 32 no Firefox (somente o estado *range* é suportado, os demais estados são exibidos como *text*):

Exemplo 32 no Internet Explorer (somente o estado *range* é suportado, os demais estados são exibidos como *text*):

Exemplo 33 – Formulário com atributos *disabled* e *readonly* do elemento `<input>`:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title> Uso de Formulário</title>
6   </head>
7   <body>
8     <h2>Atributos <i>disabled</i> e <i>readonly</i> do elemento <i>&lt;input></i></h2>
9     <!-- observe o uso das entities &lt; e &gt; para exibir os símbolos < e > com a
10      palavra input entre eles, em itálico -->
11     <form action="HTMLPage.html">
12       Nome: <input type="text" id="nome" name="define_nome"
13         value="Tim Berners Lee" readonly> <br>
14       Observações: <input type="text" id="obs" name="observa"
15         value="Criou a www e o HTML" disabled> <br>
16       <input type="submit" value="Envie">
17     </form>
18   </body>
19 </html>

```

Os atributos *disabled* e *readonly* do elemento `<input>` apresentados no exemplo 33 tem comportamentos diferentes um do outro, embora exibições semelhantes no navegador.

O *readonly* torna a entrada somente para leitura, o usuário pode clicar, mas não pode alterar seu conteúdo.

O *disabled* torna a entrada desabilitada e o usuário não pode clicar. O formulário ao ser submetido vai ignorar qualquer entrada que esteja desabilitada, não enviando ao servidor.

Os exemplos a seguir mostram como utilizar o elemento *pattern* com expressões regulares para validação de diferentes tipos de entradas de dados. Leia o APÊNDICE F – PATTERN, para saber mais a respeito.

Exemplo 34a – Formulário com atributo *pattern* do elemento `<input>` para entrada de sigla de estado:

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Exemplo input com pattern</title>
6   </head>
7   <body>
8     <form action="script_exemplo34a.php">
9       Sigla do estado (2 letras):
10      <input type="text" name="sigla" pattern="[A-Za-z]{2}"
11        title="Duas letras para a sigla do estado">
12      <input type="submit">
13    </form>
14  </body>
15</html>
```

Conteúdo do *script\_exemplo34a.php* chamado pelo formulário do exemplo 34a:

```
1 <?php
2     $sigla=$_GET["sigla"];
3     echo "<p>A sigla do estado enviada foi: <b>$sigla</b></p>";
4 ?>
```

O atributo *pattern* é muito útil por fornecer uma checagem ou validação dos dados imediatamente na entrada do formulário, isto é, no navegador (*client side*), sem ter que enviar ao servidor php (*server side*). No exemplo 34a a sigla do estado deve ser digitada com duas letras, sem números ou outros caracteres, podendo ser maiúsculas ou minúsculas.

Exemplo 34b – Formulário com atributo *pattern* do elemento `<input>` para entrada de senhas:

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Exemplo input com pattern</title>
```



```

6 </head>
7 <body>
8   <form action="script_exemplo34b.asp" method="post">
9     Senha1:
10    <input type="password" name="senha1"
11      pattern="{6,}"
12      title="Seis ou mais caracteres">
13    <br><br>
14    Senha2:
15    <input type="password" name="senha2"
16      pattern="(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).{8,}"
17      title="Debe conter pelo menos um número, uma letra maiúscula, uma letra
18 minúscula e oito ou mais caracteres">
19    <br><br>
20    <input type="submit">
21  </form>
22 </body>
23 </html>

```

No exemplo 34b as senhas têm formatos específicos que devem ser respeitados durante a digitação no formulário.

Exemplo 34c – Formulário com atributo *pattern* do elemento `<input>` para diversas entradas:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Exemplos input com pattern</title>
6   </head>
7   <body>
8     <form action="script_exemplo34c.php">
9       <p>Apenas letras
10      <input type="text" required name="letras"
11        pattern="[a-z\s]+$"
12        title="Letras e espaços em branco"></p>
13      <p>Apenas números
14      <input type="text" required name="numeros"
15        pattern="[0-9]+$"
16        title="Somente números"></p>
17      <p>Data1
18      <input type="date" required maxlength="10" name="data1"
19        pattern="[0-9]{2}\V[0-9]{2}\V[0-9]{4}$"

```

```

20         min="2000-01-01" max="2025-02-18"
21         title="formato: dd/mm/aaaa"></p>
22     <p>Data2
23     <input type="date" required maxlength="10" name="data2"
24         pattern="\d{2}\d{2}\d{4}"
25         min="2000-01-01" max="2025-02-18"
26         title="formato: dd/mm/aaaa"></p>
27     <p>Hora
28     <input type="time" required maxlength="8" name="hora"
29         pattern="[0-9]{2}:[0-9]{2}:[0-9]{2}$"
30         title="formato: hh:mm"></p>
31     <p>Telefone
32     <input type="tel" required maxlength="15" name="telefone"
33         pattern="\([0-9]{2}\) [0-9]{4,6}-[0-9]{3,4}$"
34         title="formato: (99) 99999-9999"></p>
35     <p>Email
36     <input type="email" required class="input-text" name="email"
37         pattern="[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,4}$"></p>
38     <p>Moeda
39     <input type="tel" required maxlength="15" name="valor_monetario"
40         pattern="([0-9]{1,3}\.)?[0-9]{1,3},[0-9]{2}$"></p>
41     <input type="submit">
42 </form>
43 </body>
44 </html>

```

Conteúdo do *script\_exemplo34c.php* chamado pelo formulário do exemplo 34c:

```

1 <?php
2     $letras=$_GET["letras"];
3     $numeros=$_GET["numeros"];
4     $data1=$_GET["data1"];
5     $data2=$_GET["data2"];
6     $hora=$_GET["hora"];
7     $telefone=$_GET["telefone"];
8     $email=$_GET["email"];
9     $valor_monetario=$_GET["valor_monetario"];
10    echo "<p>Só letras: <b> $letras </b></p>";
11    echo "<p>Só números: <b> $numeros </b></p>";
12    echo "<p>Data1: <b> $data1 </b></p>";
13    echo "<p>Data2: <b> $data2 </b></p>";
14    echo "<p>Hora: <b> $hora </b></p>";
15    echo "<p>Telefone: <b> $telefone </b></p>";
16    echo "<p>e-mail: <b> $email </b></p>";
17    echo "<p>Valor monetário: <b> $valor_monetario </b></p>";
18
19 ?>

```

Exemplo 34d – Formulário com atributo *pattern* do elemento `<input>` para entrada de CPF e CEP:

```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Exemplos Formato CPF, CEP</title>
6   </head>
7   <body>
8     <form action="script_exemplo34d.php">
9       CPF
10      <input type="text" name="cpf" required
11        pattern="[0-9]{3}.[0-9]{3}.[0-9]{3}-[0-9]{2}"
12        placeholder="000.000.000-00"
13        title="Digite um CPF no formato: xxx.xxx.xxx-xx">
14      <!-- ou ainda para o cpf... pattern="\d{3}\.\d{3}\.\d{3}-\d{2}" -->
15      <br><br>
16      CEP
17      <input type="text" name="cep" required
18        pattern="\d{2}\.\d{3}\-\d{3}"
19        placeholder="99.999-999"
20        title="Digite um CEP no formato: xx.xxx-xxx">
21      <br><br>
22      <input type="submit" value="Verificar">
23    </form>
24  </body>
25 </html>

```

Conteúdo do *script\_exemplo34d.php* chamado pelo formulário do exemplo 34d:

```

1 <?php
2   $cpf=$_GET["cpf"];
3   $cep=$_GET["cep"];
4   echo "<p>CPF: <b> $cpf </b></p>";
5   echo "<p>CEP: <b> $cep </b></p>";
6 ?>

```

The diagram illustrates the interaction between a web form and a PHP script. On the left, a form contains two input fields: 'CPF' with the placeholder '000.000.000-00' and 'CEP' with the placeholder '99.999-999', followed by a 'Verificar' button. Arrows indicate the flow of data from these inputs to a central box representing the script. From the script, arrows point back to the form, where validation messages are displayed. The CPF field shows a message: 'É preciso que o formato corresponda ao exigido. Digite um CPF no formato: xxx.xxx.xxx-xx'. The CEP field shows a message: 'É preciso que o formato corresponda ao exigido. Digite um CEP no formato: xx.xxx-xxx'.

☞ O atributo *pattern* trabalha em conjunto com os seguintes tipos de entrada (*input type*): *text*, *date*, *search*, *url*, *tel*, *email* e *password*.

## EXERCÍCIOS:

25) Faça um formulário que contenha uma página web que apresente para o usuário uma tela para que ele informe seus dados como a seguir:

Na senha utilize *pattern* solicitando que os caracteres sejam pelo menos um número, uma letra maiúscula, uma letra minúscula e oito ou mais caracteres.

Informações Pessoais

Usuário:

Senha:

Data nascimento: dd/mm/aaaa

Gênero:

Masculino:

Feminino:

26) No mesmo arquivo anterior, inclua agora as informações Profissionais do usuário de forma que seja obtida uma página semelhante à apresentada a seguir. Neste exercício está sendo utilizada a tag `<textarea>` para a inserção de informações em múltiplas linhas com a configuração de 10 linhas e 100 colunas. Verifique como inseri-la no documento.

Informações Pessoais

Usuário: aula\_PHP

Senha: ....

Data nascimento: 06/02/2014

Gênero:

Masculino:

Feminino:

Informações Profissionais

Conhecimentos:

☒ Java

☒ HTML

☒ CSS

☒ PHP

Curriculo:

Escolher arquivo
Curriculo Lat...a Zambon.pdf

Enviar
Limpar

Inclua aqui sua pretensão salarial e profissional:

Aprender PHP no CTI!

- 27) A partir das figuras dadas a seguir, construa um formulário no qual você utilize a sua imaginação e as tags apresentadas para criar uma página sobre animais vertebrados e invertebrados.

### ANIMAIS VERTEBRADOS

Os animais vertebrados possuem ossos. Alguns desses ossos são chamados vértebras. São as vértebras que formam a coluna vertebral.

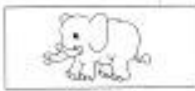
Alguns animais vertebrados:



galinha



cachorro



elefante



gato



boi



peixe

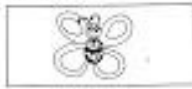
### ANIMAIS INVERTEBRADOS

Os animais invertebrados não possuem coluna vertebral.

Alguns animais invertebrados:



abelha



borboleta



minhoca



aranha



escorpião



siriri

28) Para exercitar os conceitos vistos sobre formulários, construa o form abaixo contendo estas informações básicas apresentadas. Outras que você achar interessante podem ser inseridas, bem como "mais informações pessoais" e "mais detalhes do histórico médico".

**INFORMAÇÕES PESSOAIS**

Nome :   
Endereço :   
Cidade :   
...mais informações pessoais...



**HISTÓRICO MÉDICO**

Varíola : ☐  
Caxumba : ☐  
Tonturas : ☐  
Espirros : ☐  
...mais detalhes do histórico médico...



**OPÇÕES**

Nenhum

**MEDICAÇÃO ATUAL**

...atualmente faz uso de algum medicamento?  
Sim : ☐  
Não : ☐  
...se faz uso de algum medicamento, informe-o no espaço abaixo :



Enviar

Reset

#### 4 REFERÊNCIAS BIBLIOGRÁFICAS

BRUNO, O. M.; ESTROZI, L. F.; BATISTA NETO, J. E. S. **Programando para a internet com PHP**, Rio de Janeiro: Brasport, 2010.

FLATSCHART, F. **HTML5: embarque imediato**, Rio de Janeiro: Brasport, 2011.

MAZZA, L. **HTML5 E CSS3 – domine a web do futuro**. 1. ed: Casa do Código, 2013.

SAVOIA, H. R. **HTML e CSS + PHP e MySQL**, Ribeirão Preto, SP: IELD, 2013.

**The PHP Group**. Disponível em <<http://www.php.net/>>. Acesso em: 18 dez. 2013.

**W3C**. Disponível em <<http://www.w3.org/>>. Acesso em: 18 dez. 2013.

## 5 SITES ÚTEIS

<http://www.w3c.br/Home/WebHome>

[http://www.w3schools.com/html/html5\\_intro.asp](http://www.w3schools.com/html/html5_intro.asp)

[http://validator.w3.org/#validate\\_by\\_input](http://validator.w3.org/#validate_by_input)

<http://jigsaw.w3.org/css-validator/>

<http://www.php.net/>

[http://www.php.net/manual/pt\\_BR/](http://www.php.net/manual/pt_BR/)

<http://codex.wiki.br/Html/Html>

<http://tableless.com.br/html5/>

<http://pt.wikipedia.org/wiki/HTML5>

Lista de Entities: <http://dev.w3.org/html5/html-author/charref>



## 6 APÊNDICES

### APÊNDICE A – Exemplos de formatação de tabelas

Fonte: adaptado de <[http://www.w3schools.com/html/html\\_tables.asp](http://www.w3schools.com/html/html_tables.asp)>, acesso em: 06 jan. 2014.

Apêndice A\_1 – tabela com título e cabeçalhos:

```

1 <!DOCTYPE html>
2 <html>
3 <html lang="pt-br">
4 <head>
5   <meta charset="utf-8"/>
6   <title>Tabelas com título e cabeçalhos</title>
7 </head>
8 <body>
9 <table border="1">
10  <caption>Poupança mensal</caption>
11  <tr>
12    <th>Mês</th>
13    <th>Poupança</th>
14  </tr>
15  <tr>
16    <td>Janeiro</td>
17    <td>$100</td>
18  </tr>
19  <tr>
20    <td>Fevereiro</td>
21    <td>$50</td>
22  </tr>
23 </table>
24 </body>
25 </html>

```

Apêndice A\_2 – tabela com cabeçalhos expandidos:

```

1 <!DOCTYPE html>
2 <html>
3 <html lang="pt-br">
4 <head>
5   <meta charset="utf-8"/>
6   <title>Tabelas com cabeçalhos expandidos</title>
7 </head>
8 <body>
9 <h4>Célula com cabeçalho que ocupa duas colunas:</h4>
10
11 <table border="1">
12 <tr>
13   <th>Nome</th>
14   <th colspan="2">Telefone</th>
15 </tr>
16 <tr>

```

```

17 <td>Bill Gates</td>
18 <td>555 77 854</td>
19 <td>555 77 855</td>
20 </tr>
21 </table>
22
23 <h4>Célula com cabeçalho que ocupa duas linhas:</h4>
24 <table border="1">
25 <tr>
26 <th>Primeiro Nome:</th>
27 <td>Bill Gates</td>
28 </tr>
29 <tr>
30 <th rowspan="2">Telefone:</th>
31 <td>555 77 854</td>
32 </tr>
33 <tr>
34 <td>555 77 855</td>
35 </tr>
36 </table>
37
38 </body>
39 </html>

```

Apêndice A\_3 – tabela com lista, parágrafos e outra tabela inserida:

```

1 <!DOCTYPE html>
2 <html>
3 <html lang="pt-br">
4 <head>
5 <meta charset="utf-8"/>
6 <title>Tabelas</title>
7 </head>
8 <body>
9
10 <table border="1">
11 <tr>
12 <td>
13 <p>Este é um parágrafo</p>
14 <p>Este é outro parágrafo</p>
15 </td>
16 <td>Esta célula contém uma tabela:
17 <table border="1">
18 <tr>
19 <td>A</td>
20 <td>B</td>
21 </tr>
22 <tr>
23 <td>C</td>
24 <td>D</td>
25 </tr>
26 </table>
27 </td>
28 </tr>

```

```
29 <tr>
30 <td>Esta célula contém uma lista
31 <ul>
32 <li>apples</li>
33 <li>bananas</li>
34 <li>pineapples</li>
35 </ul>
36 </td>
37 <td>HELLO</td>
38 </tr>
39 </table>
40 </body>
41 </html>
```

#### Apêndice A\_4 – tabelas com espaçamento de célula:

```
1 <!DOCTYPE html>
2 <html>
3 <html lang="pt-br">
4 <head>
5 <meta charset="utf-8"/>
6 <title>Tabelas</title>
7 </head>
8 <body>
9
10 <h4>Sem espaçamento de células:</h4>
11 <table border="1">
12 <tr>
13 <td>Primeira</td>
14 <td>Linha</td>
15 </tr>
16 <tr>
17 <td>Segunda</td>
18 <td>Linha</td>
19 </tr>
20 </table>
21
22 <h4>Com espaçamento de células="0":</h4>
23 <table border="1" cellspacing="0">
24 <tr>
25 <td>Primeira</td>
26 <td>Linha</td>
27 </tr>
28 <tr>
29 <td>Segunda</td>
30 <td>Linha</td>
31 </tr>
32 </table>
33
34 <h4>Com espaçamento de células="10":</h4>
35 <table border="1" cellspacing="10">
36 <tr>
37 <td>Primeira</td>
38 <td>Linha</td>
```

```
39 </tr>
40 <tr>
41   <td>Segunda</td>
42   <td>Linha</td>
43 </tr>
44 </table>
45 </body>
46 </html>
```

## APÊNDICE B – Lista de tags da linguagem HTML5

Fonte: adaptado de <<http://codex.wiki.br/Html/ListaDeTags>>, acesso em: 18 dez. 2013.

Tag	Função	Descrição
<!--...-->	básica	Define um comentário
<!DOCTYPE>	básica	Define o document type
<a>	link	Define um link
<abbr>	formatação	Define uma abreviação
<acronym>	formatação	Define um acrônimo
<address>	formatação	Define informações de contato do autor do documento
<area>	imagem	Define uma área dentro de um mapa de imagem
<b>	formatação	Define texto em negrito
<base>	meta dados	Especifica a URL base para todas as urls relativas de um documento
<bdo>	formatação	Substitui a atual direção de texto
<big>	formatação	Define texto grande
<blockquote>	formatação	Define uma citação longa
<body>	básica	Define o corpo do documento
 	básica	Define a quebra de linha
<button>	formulários	Define um botão clicável
<caption>	tabela	Define o título de uma tabela
<cite>	formatação	Define uma citação
<code>	formatação	Define um trecho de código de computador
<col>	tabela	Define valores de atributo para uma ou mais colunas em uma tabela
<colgroup>	tabela	Especifica um grupo de uma ou mais colunas em uma tabela para formatação
<dd>	listas	Define a descrição de um item em uma lista de definições
<del>	formatação	Define texto que foi retirado de um documento
<dfn>	formatação	Define um termo de definição
<div>	estilo	Define uma seção em um documento
<dl>	listas	Define uma lista de definição
<dt>	listas	Define um termo em uma lista de definição
<em>	formatação	Define texto enfatizado
<fieldset>	formulários	Agrupa elementos relacionados em um formulário

<form>	formulários	Define e envolve um formulário
<frame>	frames	Define uma janela/frame em um frameset
<frameset>	frames	Define uma seção de frames
<h1> ... <h6>	básica	Define títulos
<head>	meta dados	Define informações sobre um documento
<hr>	básica	Define uma linha horizontal
<html>	básica	Define a raiz do documento HTML
<i>	formatação	Define texto em itálico
<iframe>	frames	Define um frame embutido
<img>	imagem	Define uma imagem
<input>	formulários	Define controles de entrada de dados
<ins>	formatação	Define texto que foi inserido em um documento
<kbd>	formatação	Define entrada de dados de teclado
<label>	formulários	Define uma etiqueta para um elemento <input>
<legend>	formulários	Define o título para um elemento <fieldset>
<li>	listas	Define um item de lista
<link>	link	Define a relação entre o documento e um recurso externo
<map>	imagem	Define um mapa de imagem
<meta>	meta dados	Define metadados sobre um documento html
<noframes>	frames	Define texto alternativo para aqueles que não podem ver frames
<noscript>	programação	Define conteúdo alternativo para aqueles que não rodam scripts no browser
<object>	programação	Define um objeto embutido
<ol>	listas	Define uma lista ordenada
<optgroup>	formulários	Define um grupo de opções relacionadas em uma lista drop-down
<option>	formulários	Define uma opção em uma lista drop-down
<p>	básica	Define um parágrafo
<param>	programação	Define um parâmetro para um objeto
<pre>	formatação	Define texto pré formatado
<q>	formatação	Define uma citação curta
<samp>	formatação	Define amostra de saída de um computador
<script>	programação	Define um script client-side
<select>	formulários	Define uma lista drop-down
<small>	formatação	Define texto menor
<span>	estilo	Define seção de um documento

<strong>	formatação	Defines strong text
<style>	estilo	Define informações de estilo para um documento
<sub>	formatação	Define texto subscrito
<sup>	formatação	Define texto sobrescrito
<table>	tabela	Define uma tabela
<tbody>	tabela	Agrupar o conteúdo do corpo de uma tabela
<td>	tabela	Define uma célula de uma tabela
<textarea>	formulários	Define um controle de entrada de dados multi linha (text area)
<tfoot>	tabela	Agrupar os conteúdos do rodapé de uma tabela
<th>	tabela	Define uma célula do cabeçalho de uma tabela
<thead>	tabela	Agrupar os conteúdos do cabeçalho de uma tabela
<title>	meta dados	Define o título do documento
<tr>	tabela	Define uma linha em uma tabela
<tt>	formatação	Define texto de teletipo
<ul>	listas	Define uma lista não ordenada
<var>	formatação	Define uma variável

## APÊNDICE C – Lista de tags da linguagem HTML5 ordenadas por função

Fonte: < [http://www.w3schools.com/tags/ref\\_byfunc.asp](http://www.w3schools.com/tags/ref_byfunc.asp) >. Acesso em: 23 dez. 2013.

**New** : New tags in HTML5.

Tag	Description
<b>Basic</b>	
<u>&lt;!DOCTYPE&gt;</u>	Defines the document type
<u>&lt;html&gt;</u>	Defines an HTML document
<u>&lt;title&gt;</u>	Defines a title for the document
<u>&lt;body&gt;</u>	Defines the document's body
<u>&lt;h1&gt; to &lt;h6&gt;</u>	Defines HTML headings
<u>&lt;p&gt;</u>	Defines a paragraph
<u>&lt;br&gt;</u>	Inserts a single line break
<u>&lt;hr&gt;</u>	Defines a thematic change in the content
<u>&lt;!--...--&gt;</u>	Defines a comment
<b>Formatting</b>	
<u>&lt;acronym&gt;</u>	Not supported in HTML5. Use <abbr> instead. Defines an acronym
<u>&lt;abbr&gt;</u>	Defines an abbreviation
<u>&lt;address&gt;</u>	Defines contact information for the author/owner of a document/article
<u>&lt;b&gt;</u>	Defines bold text
<u>&lt;bdi&gt;</u> <b>New</b>	Isolates a part of text that might be formatted in a different direction from other text outside it
<u>&lt;bdo&gt;</u>	Overrides the current text direction
<u>&lt;big&gt;</u>	Not supported in HTML5. Use CSS instead. Defines big text
<u>&lt;blockquote&gt;</u>	Defines a section that is quoted from another source
<u>&lt;center&gt;</u>	Not supported in HTML5. Use CSS instead. Defines centered text
<u>&lt;cite&gt;</u>	Defines the title of a work
<u>&lt;code&gt;</u>	Defines a piece of computer code
<u>&lt;del&gt;</u>	Defines text that has been deleted from a document
<u>&lt;dfn&gt;</u>	Defines a definition term
<u>&lt;em&gt;</u>	Defines emphasized text
<u>&lt;font&gt;</u>	Not supported in HTML5. Use CSS instead. Defines font, color, and size for text
<u>&lt;i&gt;</u>	Defines a part of text in an alternate voice or mood



<code>&lt;ins&gt;</code>	Defines a text that has been inserted into a document
<code>&lt;kbd&gt;</code>	Defines keyboard input
<code>&lt;mark&gt;</code> <b>New</b>	Defines marked/highlighted text
<code>&lt;meter&gt;</code> <b>New</b>	Defines a scalar measurement within a known range (a gauge)
<code>&lt;pre&gt;</code>	Defines preformatted text
<code>&lt;progress&gt;</code> <b>New</b>	Represents the progress of a task
<code>&lt;q&gt;</code>	Defines a short quotation
<code>&lt;rp&gt;</code> <b>New</b>	Defines what to show in browsers that do not support ruby annotations
<code>&lt;rt&gt;</code> <b>New</b>	Defines an explanation/pronunciation of characters (for East Asian typography)
<code>&lt;ruby&gt;</code> <b>New</b>	Defines a ruby annotation (for East Asian typography)
<code>&lt;s&gt;</code>	Defines text that is no longer correct
<code>&lt;samp&gt;</code>	Defines sample output from a computer program
<code>&lt;small&gt;</code>	Defines smaller text
<code>&lt;strike&gt;</code>	<b>Not supported in HTML5. Use &lt;del&gt; instead.</b> Defines strikethrough text
<code>&lt;strong&gt;</code>	Defines important text
<code>&lt;sub&gt;</code>	Defines subscripted text
<code>&lt;sup&gt;</code>	Defines superscripted text
<code>&lt;time&gt;</code> <b>New</b>	Defines a date/time
<code>&lt;tt&gt;</code>	<b>Not supported in HTML5. Use CSS instead.</b> Defines teletype text
<code>&lt;u&gt;</code>	Defines text that should be stylistically different from normal text
<code>&lt;var&gt;</code>	Defines a variable
<code>&lt;wbr&gt;</code> <b>New</b>	Defines a possible line-break
<b>Forms</b>	
<code>&lt;form&gt;</code>	Defines an HTML form for user input
<code>&lt;input&gt;</code>	Defines an input control
<code>&lt;textarea&gt;</code>	Defines a multiline input control (text area)
<code>&lt;button&gt;</code>	Defines a clickable button
<code>&lt;select&gt;</code>	Defines a drop-down list
<code>&lt;optgroup&gt;</code>	Defines a group of related options in a drop-down list
<code>&lt;option&gt;</code>	Defines an option in a drop-down list
<code>&lt;label&gt;</code>	Defines a label for an <code>&lt;input&gt;</code> element
<code>&lt;fieldset&gt;</code>	Groups related elements in a form
<code>&lt;legend&gt;</code>	Defines a caption for a <code>&lt;fieldset&gt;</code> element

<code>&lt;datalist&gt;</code> <b>New</b>	Specifies a list of pre-defined options for input controls
<code>&lt;keygen&gt;</code> <b>New</b>	Defines a key-pair generator field (for forms)
<code>&lt;output&gt;</code> <b>New</b>	Defines the result of a calculation
<b>Frames</b>	
<code>&lt;frame&gt;</code>	<b>Not supported in HTML5.</b> Defines a window (a frame) in a frameset
<code>&lt;frameset&gt;</code>	<b>Not supported in HTML5.</b> Defines a set of frames
<code>&lt;noframes&gt;</code>	<b>Not supported in HTML5.</b> Defines an alternate content for users that do not support frames
<code>&lt;iframe&gt;</code>	Defines an inline frame
<b>Images</b>	
<code>&lt;img&gt;</code>	Defines an image
<code>&lt;map&gt;</code>	Defines a client-side image-map
<code>&lt;area&gt;</code>	Defines an area inside an image-map
<code>&lt;canvas&gt;</code> <b>New</b>	Used to draw graphics, on the fly, via scripting (usually JavaScript)
<code>&lt;figcaption&gt;</code> <b>New</b>	Defines a caption for a <code>&lt;figure&gt;</code> element
<code>&lt;figure&gt;</code> <b>New</b>	Specifies self-contained content
<b>Audio/Video</b>	
<code>&lt;audio&gt;</code> <b>New</b>	Defines sound content
<code>&lt;source&gt;</code> <b>New</b>	Defines multiple media resources for media elements ( <code>&lt;video&gt;</code> and <code>&lt;audio&gt;</code> )
<code>&lt;track&gt;</code> <b>New</b>	Defines text tracks for media elements ( <code>&lt;video&gt;</code> and <code>&lt;audio&gt;</code> )
<code>&lt;video&gt;</code> <b>New</b>	Defines a video or movie
<b>Links</b>	
<code>&lt;a&gt;</code>	Defines a hyperlink
<code>&lt;link&gt;</code>	Defines the relationship between a document and an external resource (most used to link to style sheets)
<code>&lt;nav&gt;</code> <b>New</b>	Defines navigation links
<b>Lists</b>	
<code>&lt;ul&gt;</code>	Defines an unordered list
<code>&lt;ol&gt;</code>	Defines an ordered list
<code>&lt;li&gt;</code>	Defines a list item
<code>&lt;dir&gt;</code>	<b>Not supported in HTML5. Use <code>&lt;ul&gt;</code> instead.</b> Defines a directory list
<code>&lt;dl&gt;</code>	Defines a description list
<code>&lt;dt&gt;</code>	Defines a term/name in a description list
<code>&lt;dd&gt;</code>	Defines a description of a term/name in a description list

<u>&lt;menu&gt;</u>	Defines a list/menu of commands
<u>&lt;command&gt;</u> <b>New</b>	Defines a command button that a user can invoke
<b>Tables</b>	
<u>&lt;table&gt;</u>	Defines a table
<u>&lt;caption&gt;</u>	Defines a table caption
<u>&lt;th&gt;</u>	Defines a header cell in a table
<u>&lt;tr&gt;</u>	Defines a row in a table
<u>&lt;td&gt;</u>	Defines a cell in a table
<u>&lt;thead&gt;</u>	Groups the header content in a table
<u>&lt;tbody&gt;</u>	Groups the body content in a table
<u>&lt;tfoot&gt;</u>	Groups the footer content in a table
<u>&lt;col&gt;</u>	Specifies column properties for each column within a <colgroup> element
<u>&lt;colgroup&gt;</u>	Specifies a group of one or more columns in a table for formatting
<b>Style/Sections</b>	
<u>&lt;style&gt;</u>	Defines style information for a document
<u>&lt;div&gt;</u>	Defines a section in a document
<u>&lt;span&gt;</u>	Defines a section in a document
<u>&lt;header&gt;</u> <b>New</b>	Defines a header for a document or section
<u>&lt;footer&gt;</u> <b>New</b>	Defines a footer for a document or section
<u>&lt;section&gt;</u> <b>New</b>	Defines a section in a document
<u>&lt;article&gt;</u> <b>New</b>	Defines an article
<u>&lt;aside&gt;</u> <b>New</b>	Defines content aside from the page content
<u>&lt;details&gt;</u> <b>New</b>	Defines additional details that the user can view or hide
<u>&lt;dialog&gt;</u> <b>New</b>	Defines a dialog box or window
<u>&lt;summary&gt;</u> <b>New</b>	Defines a visible heading for a <details> element
<b>Meta Info</b>	
<u>&lt;head&gt;</u>	Defines information about the document
<u>&lt;meta&gt;</u>	Defines metadata about an HTML document
<u>&lt;base&gt;</u>	Specifies the base URL/target for all relative URLs in a document
<u>&lt;basefont&gt;</u>	<b>Not supported in HTML5. Use CSS instead.</b> Specifies a default color, size, and font for all text in a document
<b>Programming</b>	
<u>&lt;script&gt;</u>	Defines a client-side script
<u>&lt;noscript&gt;</u>	Defines an alternate content for users that do not support client-side scripts

<u>&lt;applet&gt;</u>	Not supported in HTML5. Use <object> instead. Defines an embedded applet
<u>&lt;embed&gt;</u> <b>New</b>	Defines a container for an external (non-HTML) application
<u>&lt;object&gt;</u>	Defines an embedded object
<u>&lt;param&gt;</u>	Defines a parameter for an object

## APÊNDICE D – Lista de acentos e caracteres especiais (entities) em HTML

Fonte: adaptado de <<http://erikasarti.net/html/acentuacao-caracteres-especiais/>>. Acesso em: 18 dez. 2013.

### Caracteres Acentuados no Português

á	&aacute;	Á	&Aacute;	ã	&atilde;
à	&agrave;	À	&Agrave;	é	&eacute;
í	&iacute;	Í	&Iacute;	ó	&oacute;
ô	&ocirc;	Ô	&Ocirc;	ú	&uacute;
Â	&Atilde;	â	&acirc;	Ã	&Acirc;
Ê	&Eacute;	ê	&ecirc;	Ê	&Ecirc;
Õ	&Oacute;	õ	&otilde;	Õ	&Otilde;
Ú	&Uacute;	ç	&ccedil;	Ç	&Ccedil;

### Caracteres Especiais

	espaço	&nbsp;	&	e comercial	&amp;
¨	acento trema	&uml;	´	acento agudo	&acute;
“	aspas duplas esquerda	&ldquo;	”	aspas duplas direita	&rdquo;
,	aspas baixas simples	&sbquo;	„	aspas baixas duplas	&bdquo;
‹	aspas angulares simples esquerda	&lsaquo;	›	aspas angulares simples direita	&rsaquo;
–	travessão ‘en’	&ndash;	—	travessão ‘em’	&mdash;
...	reticências	&hellip;		barra vertical	&brvbar;
¶	parágrafo	&para;	§	parágrafo legal	&sect;
^	acento circunflexo	&circ;	~	acento til	&tilde;
¸	cedilha	&cedil;	”	aspas duplas	&quot;
‘	aspas simples esquerda	&lsquo;	’	aspas simples direita	&rsquo;
º	ordenal masculino	&ordm;	ª	ordinal feminino	&ordf;
«	aspas angulares duplas esquerda	&laquo;	»	aspas angulares duplas direita	&raquo;
	hífen oculto	&shy;	ˉ	macron	&macr;
•	marcador	&bull;	▸	marcador triangular	&#8227;

### Caracteres Comerciais

©	copyright	&copy;	®	marca registrada	&reg
¢	centavo	&cent;	₡	Mill (US\$.001)	&#8357;
₢	Cruzeiro (Brasil)	&#8354;	₣	Franco (França)	&#8355;
₣	Naira (Nigéria)	&#8358;	₧	Peseta (Espanha)	&#8359;
₪	New sheqel (Israel)	&#8362;	₫	Dong (Vietnã)	&#8363;
™	trade mark	&trade;	£	libra esterlina	&pound;
€	Euro	&euro;	¥	lene (yen)	&yen;
₯	Lira (Itália)	&#8356;	₯	Drachma (Grécia)	&#8367;
₹	Rupee (Índia)	&#8360;	₩	Won (Coréia)	&#8361;
₭	Kip (Laos)	&#8365;	₮	Tugrik (Mongólia)	&#8366;
\$	símbolo monetário	&curren;			

### Caracteres Matemáticos e Lógicos

$A^1$	elevado a um	$\&\sup1;$	$A_1$	um subscrito	$\&\#8321;$
$A^2$	ao quadrado	$\&\sup2;$	$A_2$	dois subscrito	$\&\#8322;$
$A^3$	ao cubo	$\&\sup3;$	$A_3$	três subscrito	$\&\#8323;$
$A^4$	elevado a quatro	$\&\#8308;$	$A_4$	quatro subscrito	$\&\#8324;$
$A^5$	elevado a cinco	$\&\#8309;$	$A_5$	cinco subscrito	$\&\#8325;$
$A^6$	elevado a seis	$\&\#8310;$	$A_6$	seis subscrito	$\&\#8326;$
$A^7$	elevado a sete	$\&\#8311;$	$A_7$	sete subscrito	$\&\#8327;$
$A^8$	elevado a oito	$\&\#8312;$	$A_8$	oito subscrito	$\&\#8328;$
$A^9$	elevado a nove	$\&\#8313;$	$A_9$	nove subscrito	$\&\#8329;$
$A^0$	zero sobrescrito	$\&\#8304;$	$_0$	zero subscrito	$\&\#8320;$
$A^=$	igual sobrescrito	$\&\#8316;$	$A_=$	igual subscrito	$\&\#8332;$
$A^+$	mais sobrescrito	$\&\#8314;$	$A_+$	mais subscrito	$\&\#8330;$
$A^-$	menos sobrescrito	$\&\#8315;$	$A_-$	menos subscrito	$\&\#8331;$
$A^()$	parênteses sobrescrito	$\&\#8317;$	$A^)$	parênteses sobrescrito	$\&\#8318;$
$A_()$	parênteses subscrito	$\&\#8333;$	$A_)$	parênteses subscrito	$\&\#8334;$
$A^n$	elevado a n	$\&\#8319;$	$A^i$	elevado a i	$\&\#8305;$
$\frac{1}{2}$	fração um meio	$\&\frac12;$	$\frac{3}{4}$	fração três quartos	$\&\frac34;$
$\frac{1}{3}$	fração um terço	$\&\#8531;$	$\frac{3}{5}$	fração três quintos	$\&\#8535;$
$\frac{1}{4}$	fração um quarto	$\&\frac14;$	$\frac{3}{8}$	fração três oitavos	$\&\#8540;$
$\frac{1}{5}$	fração um quinto	$\&\#8533;$	$\frac{4}{5}$	fração quatro quintos	$\&\#8536;$
$\frac{1}{6}$	fração um sexto	$\&\#8537;$	$\frac{5}{6}$	fração cinco sextos	$\&\#8538;$
$\frac{1}{8}$	fração um oitavo	$\&\#8539;$	$\frac{5}{8}$	fração cinco oitavos	$\&\#8541;$
$\frac{2}{3}$	fração dois terços	$\&\#8532;$	$\frac{7}{8}$	fração sete oitavos	$\&\#8542;$
$\frac{2}{5}$	fração dois quintos	$\&\#8534;$	$\neq$	diferente	$\&\neq;$
$\approx$	quase igual	$\&\asymp;$	$\prod$	PI	$\&\prod;$
$\cong$	aproximadamente igual	$\&\cong;$	$\sqrt{\phantom{x}}$	raiz quadrada	$\&\radic;$
$\propto$	proporcional	$\&\prop;$	$\infty$	infinito	$\&\infin;$
$\equiv$	Idêntico	$\&\equiv;$	$\angle$	ângulo	$\&\ang;$
$>$	maior que	$\&\gt;$	$\perp$	perpendicular	$\&\perp;$
$<$	menor que	$\&\lt;$	$'$	minuto	$\&\prime;$
$\leq$	menor ou igual	$\&\leq;$	$"$	segundo	$\&\Prime;$
$\geq$	maior ou igual	$\&\geq;$	$^\circ$	grau	$\&\deg;$
$\pm$	mais ou menos	$\&\plusmn;$	$\therefore$	consequentemente	$\&\there4;$
$-$	sinal de subtração	$\&\minus;$	$\cdot$	ponto	$\&\sdot;$
$\times$	sinal de multiplicação	$\&\times;$	$\cdot$	ponto do meio	$\&\middot;$
$\div$	sinal de divisão	$\&\divide;$	$\partial$	diferença parcial	$\&\part;$
$*$	asterisco	$\&\lowast;$	$\Im$	parte imaginária do número	$\&\image;$
$/$	barra de fração	$\&\frac1;$	$\nabla$	nabla	$\&\nabla;$
$\aleph$	alef	$\&\alefsym;$	$\Re$	parte real do número	$\&\real;$
$\text{‰}$	por-mil	$\&\permil;$	$\oplus$	soma direta	$\&\oplus;$
$\int$	sinal de integral	$\&\int;$	$\otimes$	produto de vetor	$\&\otimes;$
$\sum$	somatório	$\&\sum;$	$\emptyset$	produto vazio	$\&\oslash;$
$\in$	elemento de/pertence a	$\&\isin;$	$\emptyset$	produto vazio	$\&\Oslash;$
$\notin$	nã é elemento de	$\&\notin;$			
$\cap$	interseção	$\&\cap;$	$\cup$	união	$\&\cup;$
$\subset$	subconjunto de	$\&\sub;$	$\supset$	superconjunto de	$\&\sup;$
$\subseteq$	subconjunto de ou igual a	$\&\sube;$	$\supseteq$	superconjunto de ou igual a	$\&\supe;$
$\exists$	existe	$\&\exist;$	$\forall$	qualquer	$\&\forall;$
$\emptyset$	vazio	$\&\empty;$	$\neg$	não lógico	$\&\not;$
$\wedge$	e lógico	$\&\and;$	$\vee$	ou lógico	$\&\or;$

◇	losango	&loz;	↵	retorno de carro	&crarr;
┌	teto esquerdo	&lceil;	┐	teto direito	&rceil;
└	piso esquerdo	&lfloor;	┘	piso direito	&rfloor;

### Números Decorados

❶	&#10102;	❷	&#10103;	❸	&#10104;	❹	&#10105;	❺	&#10106;
❻	&#10107;	❼	&#10108;	❽	&#10109;	❾	&#10110;	❿	&#10111;
①	&#10112;	②	&#10113;	③	&#10114;	④	&#10115;	⑤	&#10116;
⑥	&#10117;	⑦	&#10118;	⑧	&#10119;	⑨	&#10120;	⑩	&#10121;
⑪	&#9312;	⑫	&#9313;	⑬	&#9314;	⑭	&#9315;	⑮	&#9316;
⑯	&#9317;	⑰	&#9318;	⑱	&#9319;	⑲	&#9320;	⑳	&#9321;
㉑	&#9322;	㉒	&#9323;	㉓	&#9324;	㉔	&#9325;	㉕	&#9326;
㉖	&#9327;	㉗	&#9328;	㉘	&#9329;	㉙	&#9330;	㉚	&#9331;
㉛	&#12881;	㉜	&#12882;	㉝	&#12883;	㉞	&#12884;	㉟	&#12885;
㊱	&#12886;	㊲	&#12887;	㊳	&#12888;	㊴	&#12889;	㊵	&#12890;
㊶	&#12891;	㊷	&#12892;	㊸	&#12893;	㊹	&#12894;	㊺	&#12895;
㊻	&#12977;	㊼	&#12978;	㊽	&#12979;	㊾	&#12980;	㊿	&#12981;
㊿	&#12982;	㊿	&#12983;	㊿	&#12984;	㊿	&#12985;	㊿	&#12986;
①	&#9450;								
❶	&#10122;	❷	&#10123;	❸	&#10124;	❹	&#10125;	❺	&#10126;
❻	&#10127;	❼	&#10128;	❽	&#10129;	❾	&#10130;	❿	&#10131;
⑪	&#9451;	⑫	&#9452;	⑬	&#9453;	⑭	&#9454;	⑮	&#9455;
⑯	&#9456;	⑰	&#9457;	⑱	&#9458;	⑲	&#9459;	㉑	&#9460;
①	&#9461;	②	&#9462;	③	&#9463;	④	&#9464;	⑤	&#9465;
⑥	&#9466;	⑦	&#9467;	⑧	&#9468;	⑨	&#9469;	⑩	&#9470;

### Letras decoradas

Ⓐ	&#9398;	Ⓑ	&#9399;	Ⓒ	&#9400;	Ⓓ	&#9401;	Ⓔ	&#9402;	Ⓕ	&#9403;
Ⓖ	&#9404;	Ⓖ	&#9405;	Ⓖ	&#9406;	Ⓖ	&#9407;	Ⓖ	&#9408;	Ⓖ	&#9409;
Ⓜ	&#9410;	Ⓝ	&#9411;	Ⓞ	&#9412;	Ⓟ	&#9413;	Ⓠ	&#9414;	Ⓡ	&#9415;
Ⓢ	&#9416;	Ⓣ	&#9417;	Ⓤ	&#9418;	Ⓤ	&#9419;	Ⓡ	&#9420;	Ⓡ	&#9421;
Ⓨ	&#9422;	Ⓩ	&#9423;	ⓐ	&#9424;	ⓑ	&#9425;	ⓒ	&#9426;	ⓓ	&#9427;
ⓔ	&#9428;	ⓕ	&#9429;	ⓖ	&#9430;	ⓗ	&#9431;	ⓓ	&#9432;	ⓓ	&#9433;
ⓓ	&#9433;	ⓓ	&#9434;	ⓓ	&#9435;	ⓓ	&#9436;	ⓓ	&#9437;	ⓓ	&#9438;
ⓓ	&#9439;	ⓓ	&#9440;	ⓓ	&#9441;	ⓓ	&#9442;	ⓓ	&#9443;	ⓓ	&#9444;
ⓓ	&#9445;	ⓓ	&#9446;	ⓓ	&#9447;	ⓓ	&#9448;	ⓓ	&#9449;		

### Outros acentos e caracteres especiais

ñ	&ntilde;	Ñ	&Ntilde;	ı	&iexcl;	¿	&iquest;	f	&fnof;	ß	&szlig;	μ	&micro;
ä	&auml;	Ä	&Auml;	å	&aring;	Å	&Aring;	ë	&euml;	Ë	&Euml;	è	&grave;
È	&Egrave;	ï	&iuml;	İ	&Iuml;	ì	&igrave;	Ì	&Igrave;	î	&icirc;	Î	&Icirc;
ö	&ouml;	Ö	&Ouml;	ò	&ograve;	Ò	&Ograve;	ù	&ugrave;	Û	&Ugrave;	û	&ucirc;
Û	&Ucirc;	ü	&uuml;	Ü	&Uuml;	ý	&yacute;	Ý	&Yacute;	ÿ	&yuml;	Ÿ	&Yuml;
æ	&aelig;	Æ	&AElig;	œ	&oelig;	Œ	&OElig;	†	&dagger;	‡	&Dagger;	š	&scaron;
Š	&Scaron;	þ	&thorn;	Þ	&THORN;	ð	&eth;	Ð	&ETH;				

### Caracteres gregos

α	&alpha;	A	&Alpha;	β	&beta;	B	&Beta;	γ	&gamma;	Γ	&Gamma;	δ	&delta;
Δ	&Delta;	ε	&epsilon;	E	&Epsilon;	ζ	&zeta;	Z	&Zeta;	η	&eta;	H	&Eta;
θ	&theta;	Θ	&Theta;	ι	&iota;	Ι	&lota;	κ	&kappa;	K	&Kappa;	λ	&lambda;
Λ	&Lambda;	μ	&mu;	M	&Mu;	ν	&nu;	N	&Nu;	Ξ	&xi;	Ξ	&Xi;
ο	&omicron;	O	&Omicron;	π	&pi;	Π	&Pi;	ρ	&rho;	P	&Rho;	σ	&sigma;
Σ	&Sigma;	ς	&sigmaf;	τ	&tau;	T	&Tau;	υ	&upsilon;	Υ	&Upsilon;	φ	&phi;
Φ	&Phi;	χ	&chi;	X	&Chi;	ψ	&psi;	Ψ	&Psi;	ω	&omega;	Ω	&Omega;
ϑ	&thetasym;	Υ	&upsih;	ϖ	&piv;								



## APÊNDICE E – XML

Na sequência, é apresentado o primeiro exemplo de código padrão XML em três versões que contém as mesmas informações, no entanto, cada codificação é realizada de uma determinada forma. Observe:

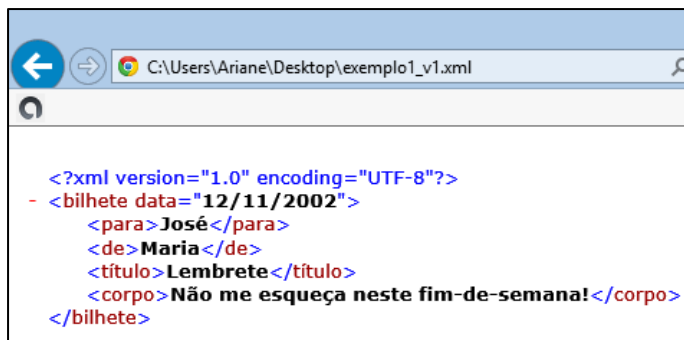
Exemplo 1<sup>5</sup>:

Versão 1 - Um elemento raiz (principal) bilhete com um atributo data e quatro elementos filho são usados na primeira versão:

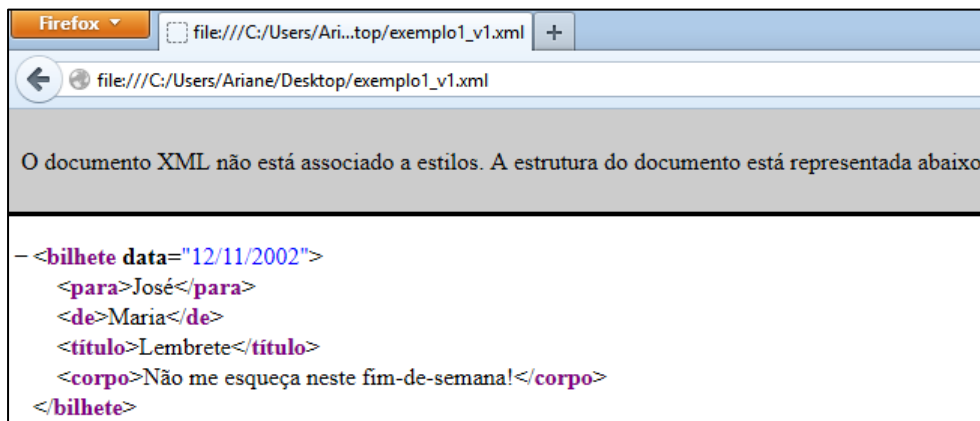
```
1 <?xml version="1.0" encoding="utf-8" ?>
2 <bilhete data="12/11/2002">
3     <para>José</para>
4     <de>Maria</de>
5     <título>Lembrete</título>
6     <corpo>Não me esqueça neste fim-de-semana!</corpo>
7 </bilhete>
```

Veja abaixo o exemplo da versão 1 em três navegadores.

Internet Explorer:

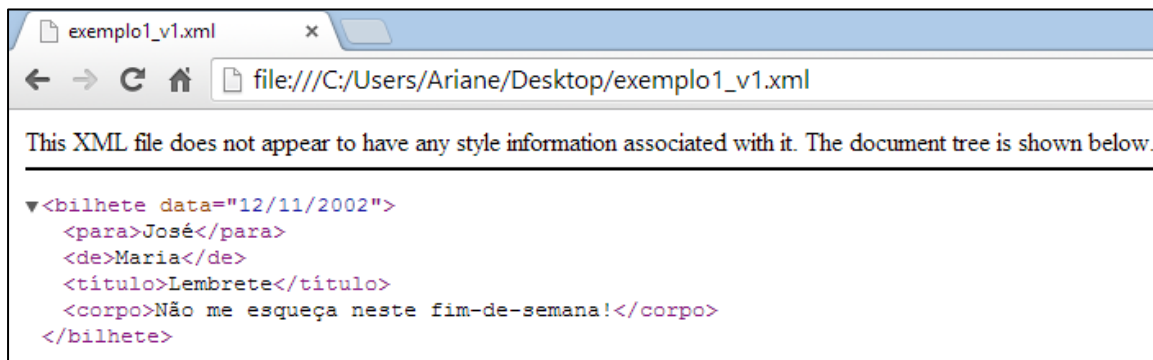


Firefox:



<sup>5</sup> Fonte: <<http://www.clem.ufba.br/tuts/xml/c05.htm>>. Acesso em: 21 dez. 2013.

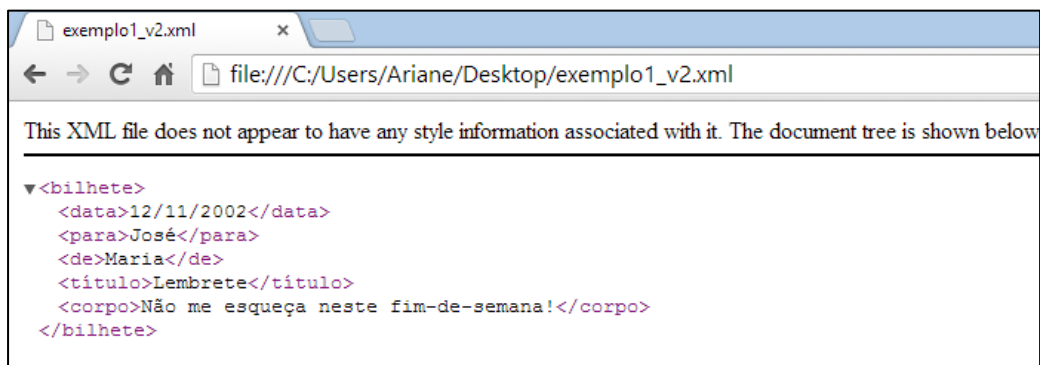
Chrome:



Versão 2 - Um elemento raiz (principal) bilhete e cinco elementos filho são usados na segunda versão:

```
1 <?xml version="1.0" encoding="utf-8" ?>
2 <bilhete>
3     <data>12/11/2002</data>
4     <para>José</para>
5     <de>Maria</de>
6     <título>Lembrete</título>
7     <corpo>Não me esqueça neste fim-de-semana!</corpo>
8 </bilhete>
```

Veja o exemplo da versão 2, agora, somente no navegador Chrome:

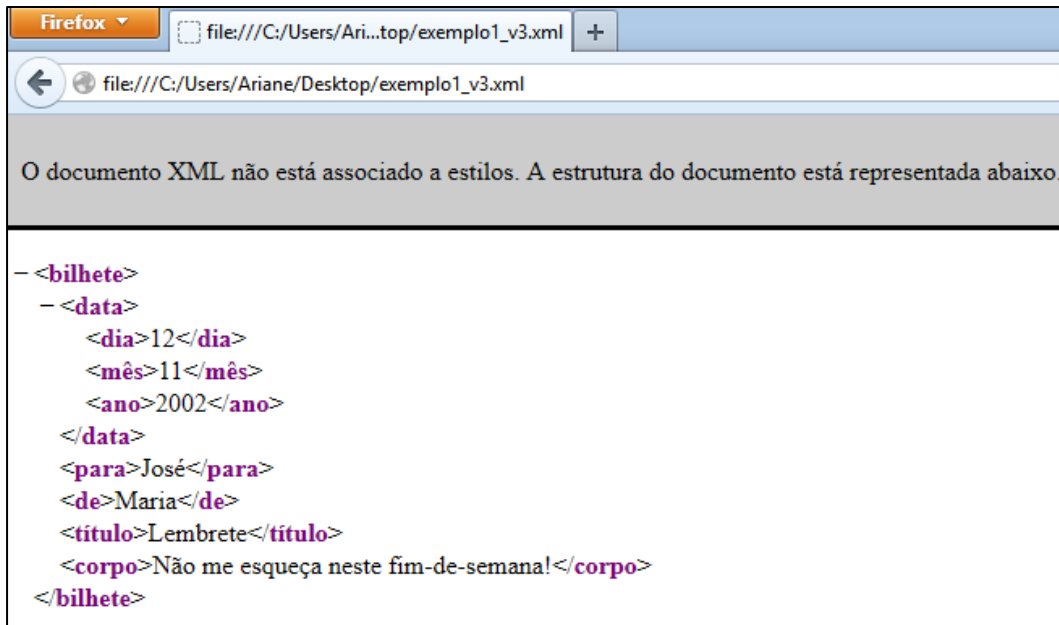


Versão 3 - Um elemento raiz (principal) bilhete e cinco elementos filho, sendo o elemento filho data usado de forma expandida na terceira versão:

```
1 <?xml version="1.0" encoding="utf-8" ?>
2 <bilhete>
3     <data>
4         <dia>12</dia>
5         <mês>11</mês>
6         <ano>2002</ano>
7     </data>
8     <para>José</para>
9     <de>Maria</de>
```

```
10      <título>Lembrete</título>
11      <corpo>Não me esqueça neste fim-de-semana!</corpo>
12 </bilhete>
```

Veja o exemplo da versão 3, agora, somente no navegador Firefox:



Devemos sempre dar preferência ao uso de elementos filho em XML invés de atributos. Os elementos filho parecem-se com dados e facilitam o armazenamento. Algumas vantagens do uso:

- Podem conter múltiplos valores;
- São facilmente expansíveis para futuras mudanças;
- Podem descrever estruturas;
- São mais fáceis de manipular por códigos de programas e mais fáceis de testar.

Observe agora o segundo exemplo de código XML.

Exemplo 2<sup>6</sup>:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!-- Edited by XMLSpy -->
3 <?xml-stylesheet type="text/css" href="cd_catalog.css"?>
4 <CATALOG>
5     <CD>
6         <TITLE>Empire Burlesque</TITLE>
7         <ARTIST>Bob Dylan</ARTIST>
8         <COUNTRY>USA</COUNTRY>
9         <COMPANY>Columbia</COMPANY>
10        <PRICE>10.90</PRICE>
11        <YEAR>1985</YEAR>
```

<sup>6</sup> Fonte: <[http://www.w3schools.com/xml/xml\\_display.asp](http://www.w3schools.com/xml/xml_display.asp)>, acesso em 23/12/2013.

```

12     </CD>
13     <CD>
14         <TITLE>Hide your heart</TITLE>
15         <ARTIST>Bonnie Tyler</ARTIST>
16         <COUNTRY>UK</COUNTRY>
17         <COMPANY>CBS Records</COMPANY>
18         <PRICE>9.90</PRICE>
19         <YEAR>1988</YEAR>
20     </CD>
21     <CD>
22         <TITLE>Eros</TITLE>
23         <ARTIST>Eros Ramazzotti</ARTIST>
24         <COUNTRY>EU</COUNTRY>
25         <COMPANY>BMG</COMPANY>
26         <PRICE>9.90</PRICE>
27         <YEAR>1997</YEAR>
28     </CD>
29     <CD>
30         <TITLE>1999 Grammy Nominees</TITLE>
31         <ARTIST>Many</ARTIST>
32         <COUNTRY>USA</COUNTRY>
33         <COMPANY>Grammy</COMPANY>
34         <PRICE>10.20</PRICE>
35         <YEAR>1999</YEAR>
36     </CD>
37 </CATALOG>

```

No exemplo 2, linha 3, aparece a referência à chamada do estilo CSS (ver item 2.3.4) que será empregado no documento XML:

```
<?xml-stylesheet type="text/css" href="cd_catalog.css"?>
```

O conteúdo do arquivo cd\_catalog.css é mostrado a seguir:

```

1 CATALOG
2 {
3     background-color: #ffffff;
4     width: 100%;
5 }
6 CD
7 {
8     display: block;
9     margin-bottom: 30pt;
10    margin-left: 0;
11 }
12 TITLE
13 {
14     color: #FF0000;
15     font-size: 20pt;
16 }
17 ARTIST
18 {
19     color: #0000FF;

```

```
20     font-size: 20pt;
21 }
22 COUNTRY,PRICE,YEAR,COMPANY
23 {
24     display: block;
25     color: #000000;
26     margin-left: 20pt;
27 }
```

Agora o documento XML está associado a um estilo. Veja a imagem parcial do exemplo no navegador Chrome:



## APÊNDICE F – PATTERN

Extraído de: <[http://www.w3schools.com/jsref/jsref\\_obj\\_regexp.asp](http://www.w3schools.com/jsref/jsref_obj_regexp.asp)>. Acesso em 10/jun/2016.

### Example

```
var patt = /w3schools/i
```

Example explained:

- **/w3schools/i** is a regular expression.
- **w3schools** is a pattern (to be used in a search).
- **i** is a modifier (modifies the search to be case-insensitive).

For a tutorial about Regular Expressions, read our [JavaScript RegExp Tutorial](#).

## Modifiers

Modifiers are used to perform case-insensitive and global searches:

### Modifier Description

<a href="#"><u>i</u></a>	Perform case-insensitive matching
<a href="#"><u>g</u></a>	Perform a global match (find all matches rather than stopping after the first match)
<a href="#"><u>m</u></a>	Perform multiline matching

## Brackets

Brackets are used to find a range of characters:

### Expression Description

<a href="#"><u>[abc]</u></a>	Find any character between the brackets
<a href="#"><u>[^abc]</u></a>	Find any character NOT between the brackets
<a href="#"><u>[0-9]</u></a>	Find any digit between the brackets
<a href="#"><u>[^0-9]</u></a>	Find any digit NOT between the brackets
<a href="#"><u>(x y)</u></a>	Find any of the alternatives specified

## Metacharacters

Metacharacters are characters with a special meaning:

### Metacharacter Description

<a href="#"><u>.</u></a>	Find a single character, except newline or line terminator
<a href="#"><u>\w</u></a>	Find a word character
<a href="#"><u>\W</u></a>	Find a non-word character
<a href="#"><u>\d</u></a>	Find a digit
<a href="#"><u>\D</u></a>	Find a non-digit character
<a href="#"><u>\s</u></a>	Find a whitespace character

<a href="#"><u>\s</u></a>	Find a non-whitespace character
<a href="#"><u>\b</u></a>	Find a match at the beginning/end of a word
<a href="#"><u>\B</u></a>	Find a match not at the beginning/end of a word
<a href="#"><u>\0</u></a>	Find a NUL character
<a href="#"><u>\n</u></a>	Find a new line character
<a href="#"><u>\f</u></a>	Find a form feed character
<a href="#"><u>\r</u></a>	Find a carriage return character
<a href="#"><u>\t</u></a>	Find a tab character
<a href="#"><u>\v</u></a>	Find a vertical tab character
<a href="#"><u>\xxx</u></a>	Find the character specified by an octal number xxx
<a href="#"><u>\xdd</u></a>	Find the character specified by a hexadecimal number dd
<a href="#"><u>\uxxxx</u></a>	Find the Unicode character specified by a hexadecimal number xxxx

## Quantifiers

### Quantifier Description

<a href="#"><u>n+</u></a>	Matches any string that contains at least one <i>n</i>
<a href="#"><u>n*</u></a>	Matches any string that contains zero or more occurrences of <i>n</i>
<a href="#"><u>n?</u></a>	Matches any string that contains zero or one occurrences of <i>n</i>
<a href="#"><u>n{X}</u></a>	Matches any string that contains a sequence of <i>X</i> <i>n</i> 's
<a href="#"><u>n{X,Y}</u></a>	Matches any string that contains a sequence of <i>X</i> to <i>Y</i> <i>n</i> 's
<a href="#"><u>n{X,}</u></a>	Matches any string that contains a sequence of at least <i>X</i> <i>n</i> 's
<a href="#"><u>n\$</u></a>	Matches any string with <i>n</i> at the end of it
<a href="#"><u>^n</u></a>	Matches any string with <i>n</i> at the beginning of it
<a href="#"><u>?=n</u></a>	Matches any string that is followed by a specific string <i>n</i>
<a href="#"><u>?!n</u></a>	Matches any string that is not followed by a specific string <i>n</i>

## RegExp Object Properties

### Property Description

<a href="#"><u>constructor</u></a>	Returns the function that created the RegExp object's prototype
<a href="#"><u>global</u></a>	Checks whether the "g" modifier is set
<a href="#"><u>ignoreCase</u></a>	Checks whether the "i" modifier is set
<a href="#"><u>lastIndex</u></a>	Specifies the index at which to start the next match
<a href="#"><u>multiline</u></a>	Checks whether the "m" modifier is set
<a href="#"><u>source</u></a>	Returns the text of the RegExp pattern

## RegExp Object Methods

### Method Description

<a href="#"><u>compile()</u></a>	Deprecated in version 1.5. Compiles a regular expression
<a href="#"><u>exec()</u></a>	Tests for a match in a string. Returns the first match
<a href="#"><u>test()</u></a>	Tests for a match in a string. Returns true or false
<a href="#"><u>toString()</u></a>	Returns the string value of the regular expression