

## Referências módulo 1 (até a aula 9):

Severo, C. *Internet: como criar home pages*. Rio de Janeiro: Editora Campus, 1996.  
Lemay, L. *Aprenda em uma semana HTML 4*. Rio de Janeiro: Editora Campus, 1998.  
Alcântara, A. et al. *Home pages: recursos e técnicas para criação de páginas WWW*. Rio de Janeiro: Editora Campus, 2000.  
Campbell B. & Darnell R. *Aprenda em uma semana Dynamic HTML*. Rio de Janeiro: Editora Campus, 1998.  
ICMC-USP. Instituto de Ciências Matemáticas e de Computação. USP. Disponível em: <http://www.icmsc.sc.usp.br/manuals/HTML>

## Home Pages

Documento projetado para ser a página principal de um Site. Funciona como a porta de entrada e deve conter um ou mais links para as demais páginas do Site ou para outros Sites relacionados.

Chama-se rede de computadores (*Net*) a utilização de diversos computadores que, de maneira interligada, trocam serviços ou informações. Se essa conexão se restringir a uma pequena região (um escritório por exemplo) é chamada **intranet**. Internet representa esta conexão em nível global. Mas esse é o assunto de outra disciplina, que tratará de redes de computadores.

## Aula 1: Introdução - a WEB e os Documentos HTML

Antes da WWW atingir um grande público era privilégio de poucos, mas agora com uma *Home Page* de conteúdo interessante no ar pode-se ser achado e visitado por pessoas de todos os lugares do mundo, o tempo todo. Empresas e profissionais já notaram as grandes portas que se abrem com essa enorme possibilidade de divulgação. Mas para isso ser possível alguém tem que fazer a elaboração do *site*; é isso que você vai aprender nesse nosso curso. Não é o máximo? Você deixará de ser apenas mais um visitante para ser um criador de *sites*!

### Objetivos:

Nesta primeira aula você:

- aprenderá os conceitos básicos da WEB,
- verá como é a estrutura de um documento HTML, desenvolvendo nosso primeiro exemplo,
- conhecerá os comandos básicos de formatação.

Você já terminará sua primeira aula aprendendo inclusive como acentuar palavras em HTML!

### Pré-requisitos:

Não há pré-requisitos para essa nossa primeira aula, embora seja sempre desejável que você conheça a Internet ou tenha "navegado" por ela alguma vez!

## 1. A World Wide Web - WWW

A idéia inicial da Internet surgiu, em 1969, como uma rede descentralizada para interligar instalações militares (ARPANET). Durante a década de 70 várias outras redes foram sendo interligadas à ARPANET e, em 1980, ela passou a se chamar Internet. No início da década de 80 são criadas redes de interconexão de instituições científicas (BITNET, CSNET, NSFNET). O ano de 1983 é um marco para a Internet, pois nele os militares abandonam o controle da ARPANET.

A World Wide Web (WWW), ou simplesmente Web, foi desenvolvida no final da década de 80 como uma forma de facilitar aos pesquisados o acesso a documentos científicos. Através da Web é possível, partindo de um determinado ponto, pesquisar outros documentos a ele relacionados,

### **A história aqui no Brasil:**

A Internet surgiu no Brasil em 1991, num primeiro momento interligando instituições acadêmicas por meio da Rede Nacional de Pesquisa. A partir de 1995, começa a utilização comercial e a popularização da rede.

Com o surgimento dos provedores de acesso, a Internet chega ao cidadão comum. A exposição e divulgação da rede através dos meios de comunicação de massa dão um grande impulso ao seu crescimento.

Os principais serviços da Internet são descritos pelos diversos protocolos de comunicação que ela disponibiliza. Alguns deles são:

- Correio Eletrônico - e-mail
- Conexões Remotas - telnet
- Transferência de Arquivos - ftp
- Transferência de hipertextos - HTTP

independente de sua localização física (que o usuário não tem a mínima necessidade de conhecer).

Com a Web, ganhou força a utilização, em larga escala, da técnica de **hipertextos**. Um **hipertexto** é um documento onde é possível incluir referências (em inglês, *Links*) a outros documentos. A seleção de uma destas referências, leva o usuário ao documento referenciado.

Inicialmente, a informação disponível na Web era encontrada principalmente sob a forma de textos e hipertextos. Aos poucos foram sendo incorporados elementos gráficos e animações aos documentos. Tais recursos ajudaram muito a popularizar a Web, aproximando-a mais do cidadão comum.

A popularização da Web criou uma série de novas aplicações, fazendo com que ela seja utilizada hoje para: divulgação científica, trabalho cooperativo, divulgação de informações culturais e livros eletrônicos, promoção de produtos e serviços, realização de comércio eletrônico e suporte técnico e vários outros usos que vão sendo inventados a cada instante.

## **2. Conceitos Básicos da Web**

Uma característica importante da Web (ou Internet) é que a informação é disponibilizada de forma independente do tipo de computador que será utilizado para a sua visualização. Para isso é necessário que os documentos sejam escritos utilizando um formato padronizado. Esta padronização é obtida através de uma linguagem chamada HTML: HyperText Markup Language. O padrão HTML é definido pelo **World Wide Web Consortium-W3C**, que é uma das entidades que controla a Internet. O padrão atual é representado pelo HTML 4.

O código HTML contém instruções de visualização de texto e informação para localização de outros documentos (links). Para poder visualizar corretamente o documento, o usuário necessita de um programa (navegador) capaz de interpretar esta linguagem, reconstituindo a informação segundo as instruções contidas no código.

Para obter algum documento na Web, o usuário deve fornecer ao navegador um conjunto de informações sobre este documento:

- a) seu protocolo de comunicação,
- b) o endereço, na Internet, da máquina na qual se encontra o documento,
- c) o diretório onde o documento está arquivado na máquina e

### **Editores de HTML**

Atualmente, existe uma grande quantidade de ferramentas, para auxiliar o desenvolvimento de páginas de Web.

Há vários níveis de ferramentas desde aquelas destinadas a converter documentos de um determinado formato para HTML, passando por editores visuais simples, até ferramentas complexas de gerenciamento de sites.

Exemplos destas ferramentas são:

- Filtros conversores: rtfhtml, psthtml.
- Editores de tags: HTML Assistant, HotDog, W3e.
- Editores Wysiwyg: Netscape Composer, MS Internet Assistant, MS FrontPage Express.
- Gerenciadores de site: MS FrontPage, AOLPress, Macromedia Dreamweaver.

Estas ferramentas porém não eliminam a necessidade de conhecer a linguagem HTML. Mesmo as ferramentas mais complexas mantêm a facilidade de visualizar e editar diretamente o código HTML, o qual é muitas vezes a forma mais fácil de se obter o efeito desejado na página.

d) o nome do documento.

Estes dados devem ser informados ao navegador segundo um formato padronizado que chamamos de URL.

A máquina que contém o documento a ser exibido deve estar executando um programa que se encarrega de receber o pedido do documento, localizá-lo no disco e enviá-lo para a máquina que o pediu. Este programa é chamado de **Servidor** de Web e o navegador que pede o documento é chamado de **Cliente**.

Agora que você já viu como o documento chega até o usuário, após sua solicitação, veja alguns conceitos básicos envolvidos neste processo.

## **2.1. Navegadores (Browsers)**

Para navegar pela Web é necessário um programa que chamamos de navegador (em inglês, Browser). Ele é responsável por: solicitar documentos na Internet e interpretá-los, exibindo-os para o usuário. Exemplos de navegadores: Netscape, Internet Explorer, Mosaic, Lynx.

## **2.2. Servidores (Web Servers)**

Para navegar pela Web também é necessário outro tipo de programa, chamado servidor. Os programas deste tipo foram especialmente elaborados para administrar o acesso às páginas HTML. Eles são executados nas máquinas onde estão guardados os documentos HTML. Estes programas são os responsáveis pelo envio dos documentos para as máquinas que os solicitam. Exemplos servidores: NCSA, CERN, Apache (em ambiente UNIX), PWS, IIS, Netscape, WebSite (em ambiente windows).

## **2.3. HTML (Hypertext Markup Language)**

É a linguagem padrão usada para a escrita de páginas da Web.

O HTML é uma linguagem de marcação, ou seja, seus comandos (chamados Tags) servem para informar aos programas de navegação os elementos que serão exibidos na página: cabeçalhos, textos em itálico, links, imagens etc. O navegador Web interpreta estes comandos e exibe a página para o usuário.

Um texto HTML não define a forma exata como o documento vai ser exibido. Isto depende do programa de navegação usado e das definições feitas pelo usuário.

## 2.4. URL (Uniform Resource Locator)

A URL é a convenção utilizada para indicar ao navegador a forma de localizar um endereço na rede. Por exemplo:  
`http://www.receita.fazenda.gov.br/IR2002/devolucao.htm`

Uma URL obedece ao seguinte formato:  
`protocolo://servidor/caminho/arquivo`

Onde:

`protocolo` – Indica a forma como vai ser realizada a comunicação entre o servidor e o cliente e também o tipo de serviço que será prestado. No caso de HTML o protocolo é o `http` (HyperText Transfer Protocol).

`servidor` – Endereço do servidor (ou máquina) na Internet. Pode ser dado na forma **nome\_da\_máquina.domínio** (como no exemplo acima) ou através do endereço IP da máquina (como em `146.164.2.68`).

`caminho` – Localização do arquivo no disco do servidor através de um diretório ou de uma lista de diretórios. (por exemplo: `http://www.ic.uff.br/~aconci/curso/formatos.html`, onde `~aconci` e `curso` são diretórios ou "pastas").

`arquivo` – Nome do arquivo desejado. Esta informação pode ser omitida. Neste caso, o servidor assume um nome padrão, que pode variar de instalação para instalação, mas normalmente é **home.html** ou **index.html**.

Você vai agora conhecer como se constitui um documento HTML, sua estrutura e seus comandos básicos de formatação.

**ASCII** é abreviatura de American Standard Code for Information Interchange, e identifica uma convenção amplamente usada em computação para codificar caracteres (letras, números e símbolos gráficos).

## 3. Características Gerais de HTML

Um documento escrito em HTML é um arquivo ASCII comum, contendo apenas os caracteres ASCII visíveis.

O navegador ignora qualquer caracter especial, inclusive aqueles que sugerem algum tipo de formatação ao texto (como `TAB`, `CR`, `LF`). Qualquer tipo de formatação deve ser informada através dos comandos conhecidos como **tags**.

Nos exemplos ao lado, A e HR são os nomes de duas **tags**. HREF, SIZE e WIDTH são os seus **atributos**. Os **valores** podem variar muito dependendo do significado do **atributo**!

O que são **tags derivadas**?  
Uma **tag derivada** é igual à **tag original** exceto por conter o caracter barra: / .  
Veja o exemplo da **tag** que define o **negrito**!

Ao construir uma página WWW, você precisa escolher uma estratégia de denominador comum para oferecer suporte à maioria dos navegadores, a menos que você objetive atender apenas a um grupo específico de usuários como nas situações de Intranet (ou redes locais).

As **tags**, ou **marcas**, se diferenciam do texto comum por estarem contidas entre o caracter "<" e o caracter ">".

Algumas **tags** contêm **atributos** que permitem configurar algumas características. E alguns **atributos** podem ter **valores** específicos. Estes **atributos** são colocados entre os delimitadores (< e >), após o nome da **tag**. Os **valores** vêm depois de um sinal de "=" colocado junto aos **atributos**. A sintaxe genérica de uma **tag** é :

```
<nome atributo1=valor atributo2= valor
....>
```

Como por exemplo:

```
<A HREF="http://faperj.br">
<HR SIZE=8 WIDTH=80%>
```

Há dois tipos de **tags**: **container tags** (ou emparelhada) e **empty tags**.

Vejamos em que são diferentes.

### 3.1. Container Tags

Servem para definir um efeito sobre um trecho do documento. Estas tags vêm sempre aos pares: uma tag indica o **início (tag de abertura)** do trecho e uma outra **tag derivada** indica o **fim (tag de fechamento)**. Todo o texto escrito entre as duas tags sofre o efeito indicado por elas.

Por exemplo, para indicar que uma parte do texto deve ser exibida em negrito utilizamos o par de **tags** <B> e </B>.

O seguinte trecho HTML:

```
Uma palavra em <B>negrito</B>
fica realçada
```

Seria exibido da seguinte forma:

Uma palavra em **negrito** fica realçada.

As **container tags** podem ser colocadas umas dentro das outras. O texto contido na **tag** mais interior sofre o efeito cumulativo de todas as outras **tags** "mais externas".

Por exemplo o seguinte trecho HTML:

```
Palavras em <i>italico e <B>negrito</B>
ficam realçadas </i> diferentemente
```

Seria exibido da seguinte forma:

Palavra em *italico* e **negrito** ficam realçadas diferentemente.

A linguagem HTML não faz a diferenciação entre letras **maiúsculas** e letras **minúsculas**. Assim escrever **<BR>** ou **<br>** tem o mesmo significado.

### 3.2. Empty Tags

São **tags** que produzem efeitos locais, normalmente introduzindo algum elemento no texto, e, portanto, não precisam de uma **tag finalizadora**. Um exemplo é a **tag <BR>** que insere no texto uma mudança de linha.

Por exemplo, o seguinte trecho HTML:

Um espaço em branco  
ou muitos na separacao de palavras tem o mesmo efeito.<br> Assim como uma linha em branco ou muitas.<br> Deve-se usar a tag adequada para mudar de linha.

Seria exibido da seguinte forma:

Um espaço em branco ou muitos na separacao de palavras tem o mesmo efeito.  
Assim como uma linha em branco ou muitas.  
Deve-se usar a tag adequada para mudar de linha.

No exemplo ao lado usamos a forma de escrever **"mais internamente"** as **tags** que são interiores a outras **tags**, para facilitar a identificação visual de onde as diversas partes do documento se iniciam e finalizam.

Esta forma de escrever, muito usada em linguagem de programação, é chamada de **indentação**.

Usar escrita **indentada** também pode ser útil para visualizar o efeito **cumulativo** que pode ocorrer com as **tags containers**, comentadas na página anterior.

### 3.3. Estrutura de um Documento HTML

Todo documento HTML tem a seguinte estrutura:

```
<HTML>

  < HEAD>
    <TITLE>
      Titulo da Pagina
    </TITLE>
  </HEAD>

  <BODY>
    No corpo do documento descreve-se tudo o que aparece dentro da pagina do navegador.
  </BODY>

</HTML>
```

### Como projetar bons documentos ?

Os passos a seguir dão algumas dicas dos aspectos que devem ser avaliados quando você for criar uma página na Web:

- avaliação do público alvo;
- definição do conteúdo;
- organização da estrutura da página;
- redação do conteúdo;
- programação visual e implementação da página.

O principal diferencial é que as tags de formatação lógicas se preocupam em definir uma idéia e não em precisar exatamente como essa idéia aparecerá.

Por exemplo, se você usar para ressaltar um texto, a tag **<EM>** fará com que ele seja exibido em *itálico* no Explorer e no Netscape, mas outro navegador poderá apresentá-lo em **negrito**, ou em um parágrafo à parte entre "aspas", etc.

A tag **HTML** indica a **área** onde deve estar contido o documento HTML. Isto não quer dizer que o Browser não exiba um texto colocado fora desta **tag**, mas essa não é uma boa prática. Porque dependendo do navegador que o usuário estiver usando algumas consequências não previsíveis podem ocorrer. Se você colocar o texto fora desta área, no mínimo você perde o controle sobre como o texto será visto pelo usuário.

A tag **HEAD** é o **cabeçalho** do documento. Nesta área são colocadas tags com informações relativas ao documento. A mais importante destas informações está contida na tag **TITLE** que deve sempre ser incluída em todas as páginas. O texto do TITLE é utilizado pelo navegador para nomear os links adicionados ao arquivo de "favoritos" (bookmarks) do usuário. Normalmente, o texto que aparece no interior da tag **TITLE** é visualizado na barra de títulos da janela do browser.

A tag **BODY** contém o documento propriamente dito. Nesta área, deve ser colocado tudo que representa a página a ser visualizada. Constitui a maior parte do documento HTML e inclui geralmente muitas outras tags no seu interior.

No final desta aula, o "exemplo atividade" descreve em detalhes estas partes; que tal ir lá dar uma olhada?

### 3.4. Comentários

Como em outras linguagens de programação, é possível inserir parte de texto que o usuário não tem acesso. Essas partes são chamadas **comentários**.

Num texto HTML, todo texto incluído entre **<!--** e **-->** é ignorado pelo Browser, ou seja é interpretado como um comentário do programador.

### 3.5. Comandos Básicos de Formatação

A linguagem HTML possui duas classes de elementos utilizados para modificar o estilo de apresentação de partes do texto: tags **físicas** e tags **lógicas**.

As **tags físicas** indicam, explicitamente, a forma como o autor deseja ver exibido o seu texto. Elas são mostradas na tabela 1.1.

**Tabela 1.1 -Tags físicas**

Elemento	Descrição	Efeito
<B>...</B>	Negrito	<b>texto</b> normal
<i>...</i>	Itálico	<i>texto</i> normal
<U>...</U>	Sublinhado	<u>texto</u> normal
<TT>...</TT>	Letras igualmente espaçadas	texto normal
<SUB> ... </SUB>	Subescrito	texto normal
<SUP> ... </SUP>	Sobrescrito	texto normal
<STRIKE> ... </STRIKE>	Riscado	<del>texto</del> normal
<BIG> ... </BIG>	Fonte grande	texto normal
<SMALL>...</SMALL>	Fonte pequena	texto normal

**NOTA:**

O pouco rigor de alguns navegadores permite que algumas **container tags** HTML sejam representadas como **empty tags**.

Uma destas é <P> . Pode acontecer do seu texto ser bem interpretado sem ser necessário colocar a **tag** de fim de parágrafo (</P>). No entanto, o **World Wide Web Consortium-W3C** recomenda que ela seja uma **container tag**.

As fontes (ou tipos de letras usadas para os textos), podem ser de dois tipos: **proporcionais** ou **mono-espaçadas**. As primeiras determinam um espaço para cada letra **proporcional à sua largura**. Nelas por exemplo, um ponto, '.', ocupa menos espaço que a letra 'i', e esta ocupa menos espaço que a letra 'm'.

Já as fontes **mono-espaçadas** determinam o mesmo espaço para qualquer que seja o símbolo gráfico a ser escrito, como nas máquinas de escrever antigas. Fontes **mono-espaçadas** possibilitam que um texto seja exibido respeitando a **endentação**. Para ter um texto escrito desta forma em HTML você deve escrevê-lo entre as tags <TT> e </TT> .

As **tags lógicas** expressam uma idéia que deve ser passada ao usuário e a forma como o texto será exibido depende do navegador. Algumas delas são descritas na tabela 1.2.

**Tabela 1.2 - Tags lógicas**

Elemento	Descrição	Explorer e Netscape
<STRONG>... </STRONG>	Texto forte	Negrito
<EM> ... </EM>	Texto enfatizado	Itálico
<CITE>...</CITE>	Citação	Itálico
<CODE> ... </CODE>	Código de programa	Mono-espaçado
<ADDRESS>... </ADDRESS>	Endereço	Itálico

Como você pode ver, os dois navegadores principais (atualmente no mercado), as exibem da mesma maneira. Mas



isso é apenas por acaso. Nada garante que esta forma de exibição continue nas próximas versões destes navegadores ou que os outros façam o mesmo.

## 4. Formataadores

Como foi dito anteriormente, o navegador ignora qualquer caracter especial de formatação, como os caracteres de margem e mudança de linha existentes no arquivo HTML. Quando é necessária uma formatação num documento, deve-se incluir uma das **tags** da tabela abaixo.

Os **atributos** devem ser colocados dentro das **tags** de abertura e podem ter **valores** ou não. Os **valores** possíveis de cada atributo também variam com o significado destes atributos, como pode ser visto na tabela ao lado.

**Tabela 1.3 - Tags de formatação**

Elemento	Descrição	Atributos
<BLOCKQUOTE>	Aumentar a margem	Nenhum
 	Quebra de linha	Nenhum
<PRE>	Parágrafos pré-formatados	Nenhum
<P>	Início de parágrafo	ALIGN
<HR>	Linha horizontal	SIZE, WIDTH, ALIGN e NOSHADE

O controle sobre o alinhamento da página pode ser conseguido através da margem. O par de tags <BLOCKQUOTE>...</BLOCKQUOTE> serve para aumentar a margem. Essa tag pode ser acumulada para conseguir margens maiores, como por exemplo:

```
<BLOCKQUOTE>
  texto com mais margem
  <BLOCKQUOTE>
    texto com mais margem ainda...
  </BLOCKQUOTE>
</BLOCKQUOTE>
```

### 4.1. Tag <PRE> : texto pré-formatado

Muitas vezes, é interessante fazer com que o navegador reproduza exatamente a formatação do texto escrito no arquivo HTML, sem ignorar espaços, mudanças de linha, tabulações e utilizando uma fonte mono-espçada para exibir o texto. Um exemplo típico disso é quando se deseja incluir uma listagem de um programa de computador em uma página HTML. Um trecho do código de um programa é melhor descrito se endentado como já fizemos em alguns exemplos desta aula.

Para incluir um texto pré-formatado com fonte mono-espaçada utiliza-se a tag `<PRE> ... </PRE>`. Esta tag, porém, não impede o navegador de interpretar outras tags que estejam em seu interior, permitindo realizar mudanças de fonte, estilo e cor do texto.

## 4.2. Alinhamento de Texto

Os textos contidos nos arquivos HTML são exibidos, salvo indicação em contrário, alinhados à esquerda da janela. Para ter o texto alinhado de forma diferente é necessário modificar o atributo `ALIGN` existente em algumas tags (como cabeçalhos `<P>` e `<HR>`).

O atributo `ALIGN` pode assumir os seguintes valores: `RIGHT` (direita), `CENTER` (centro) ou `LEFT` (esquerda).

Exemplo: `<P align=right>`

O alinhamento centralizado também pode ser obtido através da tag `<CENTER> ... </CENTER>`.

Exemplo: `<center>texto centralizado</center>`

A tag `<DIV> ... </DIV>` permite definir o alinhamento default para tudo que ela contiver.

Exemplo: `<DIV align=left>  
texto alinhado à esquerda  
</DIV>`

## 4.3. Atributos de Linha Horizontal `<HR>`

O atributo `SIZE` é utilizado para definir a espessura da linha em número de pontos (pixels).

O atributo `WIDTH` serve para definir a largura da linha. Esta largura pode ser especificada pelo número de pontos ou pelo percentual da janela que será ocupado pela linha.

O atributo `NOSHADE` (*no shade*, isto é, sem sombra) não tem valor. A sua simples inclusão faz com que a linha não seja desenhada com efeito sombreado.

O atributo `ALIGN` já foi descrito anteriormente na seção Alinhamento de Texto. Esse atributo aparece em diversas tags de HTML, com o mesmo conjunto de valores possíveis.

O exemplo abaixo, inclui uma linha de 2 pontos de espessura, ocupando 50% da janela e sem sombreado:

```
<HR SIZE=2 WIDTH=50% NOSHADE>
```

## 5. Cabeçalhos

Ao se redigir um documento é conveniente organizá-lo de forma clara atribuindo títulos e subtítulos às suas diversas partes.

A linguagem HTML oferece um conjunto de 6 cabeçalhos pré-definidos que podem ser incluídos no documento através da tag <Hn> . . . </Hn>, onde n pode ser um número de 1 a 6.

Os cabeçalhos aceitam o atributo ALIGN já descrito (o valor padrão, se não houver qualquer definição de alinhamento, é centralizado). Os textos nos cabeçalhos são escritos em negrito e seus tamanhos variam do maior H1 até o menor H6.

```
<H1>Titulo Principal</H1>
<H2 align=left>Titulo</H2>
<H3 align=right>Subtitulo</H3>
<H6>Todo este texto vai ser escrito em
negrito e centralizado como se fosse um
titulo</H6>
```

**Portabilidade** é a qualidade de um componente de hardware ou software que o torna capaz de ser utilizado em diferentes tipos de computadores. Algumas vezes a portabilidade ocorre mediante pequenas alterações, mas quanto menores essas alterações mais portátil ou maior será o grau de portabilidade do componente. Houaiss, A. *Dicionário da língua portuguesa*, Rio de Janeiro, Editora Objetiva, 2001.

## 6. Acentuação em HTML

Não existe uma padronização universal para definir os códigos associados aos caracteres acentuados. Quando produzimos uma página no ambiente Windows, utilizando diretamente os caracteres do Windows, esta página será visualizada sem problemas em grande parte das máquinas que utilizam este mesmo sistema (se o sistema estiver configurado para utilizar a norma ISO Latin 1), mas isso não será verdade para qualquer ambiente.

Como consequência é possível que um texto, cheio de caracteres acentuados e visualizados de forma perfeita na tela de quem o produziu, apareça cheio de caracteres estranhos na tela de alguém que esteja trabalhando em um ambiente diferente. Para garantir a **portabilidade** dos documentos nas mais diferentes plataformas, o HTML prevê uma série de códigos que devem ser utilizados no lugar de caracteres acentuados e outros caracteres especiais. A tabela 1.4 resume estes caracteres.

**Tabela 1.4 - Caracteres especiais e para acentuação**

Codificação em HTML			
á &aacute;	é &eacute;	Ç &ccedil;	& &amp;
ã &atilde;	ê &ecirc;	Ç &Ccedil;	“ &quot;
â &acirc;	ó &oacute;	< &lt;	® &reg;
à &agrave;	ü &uuml;	> &gt;	© &copy;

**Nota:**

O exemplo atividade está propositadamente e alguns acentos. Você entenderá o porquê quando finalizar os exercícios desta aula!!!

Mas ainda há muito mais coisas nele! Entenda ainda mais na próxima aula!!!

A melhor maneira de aprender é fazendo! Por isso estamos propondo um exemplo atividade que além de resumir o que vimos nesta aula, servirá para você exercitar o conteúdo.

**Exemplo Atividade:**

```
<HTML>
  <HEAD>
    <TITLE>Curso de
    Constru&ccedil;&atilde;o de
    P&aacute;ginas WEB</TITLE>
  </HEAD>
  <BODY>
    <H1 align=center >
    Primeiro Exemplo</H1>
    Primeiro leia este exemplo procurando
    identificar cada um dos elementos
    comentados na aula.
    Segundo, identifique quais as "Tags"
    que são "containers" e seus inícios e
    fins.
    Terceiro, vamos usar este exemplo
    para testar cada das formas de
    formatacao aprendidas. Assim copie
    este exemplo e salve-o em um editor
    de texto. Logo que ele seja salvo com
    a terminacao .htm, voce pode
    visualiza-lo em um navegador. Abra o
    arquivo escolhendo Arquivo/Abrir (ou
    File/Open)no menu Arquivo (ou File)
    do navegador. Depois de visualiza-lo
    como esta escreva cada frase de forma
    diferente no navegador, usando para
    isso cada uma das 9 formas diferentes
    de Tags Fisicas descritas: negrito,
    italico, sublinhado,etc.
    Quarto, vamos fazer todos os demais
    exercicios que seguem!!!
  </BODY>
</HTML>
```

Depois de responder cada exercício, salve-o em um editor de texto. Logo que ele seja salvo com a terminação .htm, é importante que você visualize-o em um navegador.

## Exercícios:

**1.** Utilize, no exemplo atividade, os separadores de quebra de linha `<BR>` e parágrafo `<P>`, separando adequadamente o texto. Além disso, desenhe linhas horizontais ocupando 25% da janela, separando os parágrafos que iniciam com as palavras: Segundo, Terceiro e Quarto dos parágrafos anteriores. Visualize o novo aspecto que o exemplo terá agora!

**2.** Atribua subtítulos aos parágrafos que você separou no exercício anterior, utilizando `<H3>` `</H3>`. Compare como fica a página se você agora usar `<H5>` `</H5>`.

**3.** Ao final do arquivo inclua (usando as opções *Copiar* e *Colar*, que geralmente estão nos menus *Editar* dos programas de edição de texto) como "texto formatado", um arquivo de texto qualquer que você já tenha armazenado antes (você deve usar que tag para isso?). Veja o resultado e responda: as tags `<PRE>` `</PRE>`, realmente funcionaram?

**4.** Mude o alinhamento de cada parágrafo e cabeçalho do texto (o subtítulo que você escreveu no exercício 2). Faça o primeiro estar à direita, o segundo centrado, o terceiro à esquerda e no quarto use a tag `<DIV>``</DIV>`. Depois responda: o que ocorreu no último caso?

**5.** Você já deve ter notado que o texto está sem acentos! Você sabe como melhorar isso, não? Então, mãos à obra!

## Resumo:

Nesta aula, você aprendeu: a estrutura básica da linguagem HTML, a formatar um documento, a incluir títulos, subtítulos e, a acentuar o texto corretamente, independentemente do navegador que esteja usando. Fez sua primeira "obra" em HTML e testou seus novos conhecimentos nos exercícios.

## Auto-avaliação:

Se você concluiu com sucesso os exercícios, podemos dizer que você fixou todos os detalhes desta aula e está pronto para tornar qualquer texto um hipertexto. Este é o assunto da próxima aula: **Listas e Links**.