

HTML

HTML (Hyper Text Markup Language) é uma linguagem utilizada para a edição e marcação de documentos da Web que possui etiquetas para formatar os textos e figuras que serão apresentados pelo navegador e criar ligações entre páginas, assim criando o conceito de hipertexto.

Essa linguagem é composta por comandos colocados entre parênteses angulares (<tag>) que informam como os textos e imagens serão exibidos na página.

Toda página Web é um arquivo de texto gravado com extensão **.htm** (Windows) ou **.html** (Unix) e os arquivos são gravados em um servidor. Quando o computador cliente solicita uma página, o computador servidor envia o html que o navegador se encarregará de interpretar e exibir a página escolhida.

A edição dos documentos em HTML pode ser criada em um simples editor de textos como o Bloco de Notas do Windows, mas também podem ser construídos sem a necessidade da digitação utilizando o Dreamweaver ou FrontPage.

O documento *HTML* pode ser escrito em qualquer *editor de textos*, desde que este tenha a capacidade de gravá-lo como código *ASCII* (*American Standard Code for Information Interchange* - código utilizado para representar textos quando há computadores envolvidos), isto é, como *texto puro*, sem formatação ou caracteres de controle. Pode-se usar, portanto, o Notepad do Windows, ou o Edit do MS-DOS. Pode-se também usar editores mais modernos, como o Word, sendo necessário neste caso que o documento seja gravado no formato texto e não como um ".doc" normal do MS-Word.

Conjunto de caracteres ASCII

O conjunto de caracteres de 7 bits do código padrão americano para intercâmbio de informações (ASCII, American Standard Code for Information Interchange) e que é amplamente usado para representar letras e símbolos encontrados em um teclado americano padrão. O conjunto de caracteres ASCII é igual aos 128 primeiros caracteres (0127) do conjunto de caracteres ANSI.

Conjunto de caracteres ANSI

O conjunto de caracteres de 8 bits do Instituto Nacional de Padronização Americano (ANSI, American National Standards Institute) que é usado pelo Microsoft Windows e que possibilita a representação de até 256 caracteres (0-255) através do teclado. Os 128 primeiros caracteres (0-127) correspondem a letras e símbolos de um teclado americano padrão. Os outros 128 caracteres (128-255) representam caracteres especiais, como letras de alfabetos internacionais, acentos, símbolos monetários e frações.

COMANDOS DE HTML

Um documento HTML possui tags que informam como a página será exibida. A maioria dessas tags possuem também sua correspondente de fechamento.

EX: <tag> texto</tag>

Este fechamento é necessário porque estas tags servem para definir a formatação de uma porção de textos, por isso devemos marcar onde começa e termina o texto com a formatação especificada por ela.

TAGs

TAGs são os "*comandos*" do *HTML*. Um documento *HTML* é composto de *TAGs* que farão com que o browser monte a página de acordo com a formatação definida pelos comando *HTML*.

Um *TAG* é sempre precedido de um caracter "<" (menor que) e seguido por um caracter ">" (maior que).

Exemplo de TAG: <HTML>

Na maioria dos casos, um *TAG* deve ter um correspondente, chamado de *TAG de fechamento*. Ou seja, um *TAG* indica onde começa sua área de influência, enquanto o seu correspondente *TAG de fechamento* indica onde termina a área de influência.

Um *TAG de fechamento* tem o mesmo nome do *TAG de abertura*, precedido do caracter "/".

Exemplo de TAG de fechamento: `</HTML>`

TAGs podem ser escritas em minúsculas ou maiúscula, mas nunca podem haver espaços em branco dentro de TAGs, exceto entre *atributos* e entre *aspas*.

Exemplo : `<HTML>`, `<html>`, `<HtMl>`, `<HtmL>`, ...

ATRIBUTOS

Um atributo define várias propriedades para uma tag. Por exemplo, a tag `` texto `` possui o atributo `COLOR` que determina qual a cor será da font utilizada no texto.

O atributo é sempre incluído na tag inicial e nunca na tag final, e o seu valor é delimitado por aspas simples ou duplas.

ESTRUTURA BÁSICA

A Estrutura Básica de um documento HTML é a seguinte:

`<html>` → Início do Documento

`<head>` → Início do Cabeçalho do Documento

`<title>` → Início do Título do Documento

Aqui entra o Título do documento

`</title>` → Fim do Título do Documento

`</head>` → Fim do Cabeçalho do Documento

`<body>` → Início do Conteúdo do Documento

Aqui entra todo o conteúdo que será exposto pelo Browser...

`</body>` → Fim do Conteúdo do Documento

`</html>` → Fim do Documento

Início e Fim do Documento

Os TAGs `<HTML>` . . . `</HTML>` indicam respectivamente o início e fim do documento.

Definições Lógicas sobre o Documento

Entre os TAGs `<HEAD>` . . . `</HEAD>` estão as *definições lógicas* sobre o documento. Delimitam a seção de cabeçalho do documento. Trata-se da primeira seção do documento.

Título da Página

Os TAGs `<TITLE>` . . . `</TITLE>` indicam o título do documento, que será apresentado na barra superior do browser. Estas marcas devem constar da seção de cabeçalho.

Conteúdo da Página

Entre os TAGs **<BODY> . . . </BODY>** estão o *conteúdo* da página que será exposto pelo Browser, incluindo cabeçalho e rodapé.

Corpo do documento. Entre estas marcas estará contido maior parte do conteúdo a ser apresentado, textos, imagens, etc.

TAG com Atributos

Exemplo:

`<html>` > Início do Documento

`<head>` > Início do Cabeçalho do Documento

`<title>` > Início do Título do Documento

Aqui entra o Título do documento

`</title>` > Fim do Título do Documento

`</head>` > Fim do Cabeçalho do Documento

`<body>` > Início do Conteúdo do Documento

`<p align="center">`Esta é a Home Page da XYZ Informática e este texto está centralizado.`</p>` > O TAG `<P>` (parágrafo) está acompanhado do Atributo ALIGN especificando o Valor "CENTER", determinando que o texto do parágrafo deve ser alinhado ao centro

`</body>` > Fim do Conteúdo do Documento

`</html>` > Fim do Documento

CSS

A CSS (Cascating Style Sheet), conhecido como Folhas de Estilo em Cascata, assim como a HTML, possui uma sintaxe própria, onde são definidas as regras exatas para o seu uso. Evidentemente, erros de sintaxe podem ocasionar o não funcionamento e por isso, a máxima atenção deve ser dada ao digitar os códigos.

A sintaxe possui três partes: seletor, propriedade e valor.

- **Seletor** – indica qual a TAG HTML será aplicada a propriedade, como IDs e Classes;
- **Propriedade** – é um elemento ligado à TAG, que em HTML chamamos de atributo, como por exemplo, *font*, *font face*, *color*, *background*, etc.
- **Valor** – tal como ocorre em HTML, o valor é ligado a propriedade. Por exemplo: no atributo "color" são as cores que podem ser inseridas como números hexadecimais ou o nome em inglês (red, blue, yellow, maroon, etc.)

Como escrever os códigos em CSS?

Seletor {propriedade: valor}

Em primeiro lugar vem o seletor e entre chaves ficam a propriedade e o valor, separados por dois pontos. Você poderá também organizar a forma de escrever os códigos, conforme mostrado abaixo:

Seletor {

```
Propriedade: valor
}
```

ou

```
Seletor
{
Propriedade: valor
}
```

Colocar uma cor de fundo em um documento HTML.

```
<style type="text/css">
```

```
Body {
Background: #09873b;
}
```

```
</style>
```

OBS: o uso de `<style type="text/css">` e o fechamento com `</style>` é para CSS interna, informando ao browser que *"isto aqui é um código CSS"*

O documento deverá ser salvo com a extensão *.html* ou *.htm*

Como deve ser escrito:

```
<html>
<head>
<title>Exemplo 1</title>
</head>
<style type="text/css">
body{
background: #09873b;
}
</style>
</html>
```

Salve como *exemplo1.html* e execute

Onde e como inserir os códigos CSS?

Os códigos CSS podem ser aplicados a um documento de três formas:

- In-line
- Interno
- Externo (importado)

A forma mais usada é a **externa**, pois podemos deixar todos os códigos CSS em um arquivo.

Se for necessário mudar a aparência do site, basta mudar as configurações desse único arquivo. Normalmente esse arquivo é elaborado no *bloco de notas*, e salvo com a extensão *.css*

Inserção de código CSS - forma "In-line"

```
<html>
<head>
<title>Exemplo 1</title>
</head>
<body style="background-color: #ffff80;">
```

```
<P style="color: #09873b; font-family: arial; font-size: 16px;">
Este texto aparece na cor hexadecimal #09873b, fonte arial
com tamanho 18<br> e cor de fundo amarela.
</p>
```

Este método é muito demorado de ser usado. Sua aplicação só faz sentido para trechos muito específicos.

Inserção de código CSS - forma "Interno"

Este método usa a TAG <style>. Observe que no método anterior (In-line) <style> era somente um atributo inserido a uma TAG HTML. Nesta forma de inserção <style> é uma TAG.

Com <style> pode-se escrever todos os códigos CSS dentro de uma página, porém, essas regras são aplicadas apenas a uma página.

Importante: As regras CSS devem ser inseridas entre <head> e </head>.

```
<html>
<head>
<title>Exemplo 2</title>
<style type="text/css">
Body {
background: #ffffcc;
}
h1 {
color: ff0000;
}
h2 {
color: #008000;
}
h3 {
color: #008080;
}
</style>
</head>
<h1>Texto vermelho</h>
<h2>Texto verde</h>
<h3>Texto azul-petróleo</h>
```

Salve como *exemplo2.html* e execute.

Esse método é mais vantajoso do que o anterior, pois para mudar a aparência da página, basta mudar os códigos escritos.

Apesar disso, o melhor de todos é o que veremos a seguir – o *importado*.

Quando o método importado é usado, as regras CSS ficam em um arquivo externo e não no documento HTML, como visto nos dois métodos anteriores.

Com este método é possível associar quantas páginas HTML forem necessárias a um único arquivo com as regras CSS. Ao mudar as regras desse arquivo CSS, o site inteiro altera-se automaticamente.

O arquivo externo com as regras CSS, nada mais é do que um arquivo com a extensão *.txt* renomeada para *.css*, podendo ser escrito no *bloco de notas*.

Inserção de código CSS - forma "Importado ou Externo"

Este método é o mais usado e recomendado. O curso de CSS proposto nesta apostila, utilizará este método.

A sintaxe básica para importar o arquivo *.css* é vista abaixo:

```
<html>
<head>
<title>   </title>
```

```
<link rel="stylesheet" type="text/css" href="nome do arquivo.css">
```

```
</head>
```

Como escrever as regras para CSS?

Veja abaixo a sintaxe para a criação do arquivo. Abra o bloco de notas e digite:

```
/*Regras CSS*/
body {background: #FFFFCC;}
h1 {color: #FF0000;}
h2 {color: #008000;}
h3 {color: #008080;}
```

Salve o arquivo como *fontes.css*. Lembre-se de salvá-lo na mesma pasta na qual está sendo desenvolvido o site, ou então, se você criar uma pasta específica para as regras CSS, não se esqueça de indicar o caminho "path" onde está o arquivo CSS.

Observe no arquivo acima, que as regras CSS são precedidas de um comentário: */*Regras CSS*/*

Os comentários não aparecem no browser do internauta, embora continue junto com o código.

Para contribuir com a organização e identificação dos trechos de código CSS, você pode contar com a possibilidade de inserir linhas de comentário durante o processo de implementação de uma página.

A finalidade do comentário é apenas de orientação para o webdesigner. Em HTML também existe uma forma de inserir comentários:

Sintaxe para CSS ↑ */* exemplo de comentário */*

Sintaxe para HTML ↑ *<!--exemplo de comentário -->*

Vamos à prática:

```
<html>
<head>
<title>Exemplo 3</title>
<link rel="stylesheet" type="text/css" href="fontes.css">
</head>

<h1>Texto escrito em vermelho</h>
```

```
<h2>Texto escrito em verde</h>  
<h3>Texto escrito em azul-petróleo</h>
```

Salve o arquivo como *exemplo3.html* e execute.

JAVASCRIPT

Javascript é uma linguagem que permite inserir conteúdos dinâmicos em páginas escritas em HTML. Com Javascript você tem muitas possibilidades para “incrementar” sua página com recursos interessantes, como, por exemplo, emitir caixas de diálogos com mensagem para o usuário.

O código Javascript pode ser inserido aleatoriamente ou vinculado a ocorrência de eventos. Os códigos soltos são executados na sequência em que aparecem na página e os atrelados a eventos são executados apenas quando o evento ocorre.

Sintaxe

```
<script language="javascript">  
  
</script>
```

Esse bloco de código pode ser inserido entre as tags <head></head> ou <body></body>. Para melhor visualização e facilidade de manutenção, recomenda-se que toda a lógica seja escrita no início do documento por meio da criação de funções a serem invocadas quando for necessário, normalmente atreladas a eventos.