



# Programação Web

**Professor: Euclides Paim**  
*[euclidespaim@gmail.com](mailto:euclidespaim@gmail.com)*



# ***Fundamentos de HTML***

**Professor: Euclides Paim**  
*euclidespaim@gmail.com*



# Programação Web

## *Sumário*

- **Fundamentos de HTML**
  - Links;
  - Multimídia;
  - Tabelas;
  - Listas;
  - Elementos:
    - *inline*
    - *block*
  - Atributo *charset*
  - Revisão.



<html>





# Programação Web

## *Hyperlinks HTML*

- **Hyperlinks HTML**

- Os *links* são encontrados em quase todas as páginas da web. Os *links* permitem que os usuários cliquem para ir de uma página para outra.
- Quando movemos o mouse sobre um *link*, a seta do mouse se transforma em uma pequena  **mão**.
- Um *link* não precisa ser um texto, ele pode ser uma  **imagem** ou qualquer outro  **elemento** HTML!
  - A tag HTML **<a>** define um *hiperlink* e possui a seguinte sintaxe:

```
<a href="url">link text</a>
```

- O atributo mais importante do elemento **<a>** é o atributo **href**, que indica o destino do *link*.
- O texto do *link* é a parte que ficará visível ao leitor.
- Por padrão, os *links* aparecerão da seguinte forma em todos os navegadores:
  - Um *link* **não** visitado está sublinhado e em  **azul**
  - Um *link* **visitado** está sublinhado e  **roxo**
  - Um *link* **ativo** está sublinhado e  **vermelho**



# Programação Web

## *Hyperlinks HTML*

- ***Hyperlinks HTML***
  - O Atributo ***target***
    - Por padrão, a página vinculada será exibida na janela do navegador atual. Para alterar isso, você deve especificar outro destino para o link.
    - O atributo *target* especifica onde abrir o documento vinculado.
    - O atributo *target* pode ter um dos seguintes valores:
      - *\_self* - Padrão. Abre o documento na mesma janela/guia em que foi clicado;
      - *\_blank* - Abre o documento em uma nova janela ou guia;
      - *\_parent* - Abre o documento no quadro pai;
      - *\_top* - Abre o documento em todo o corpo da janela.



# Programação Web

## *Hyperlinks HTML*

- **Imagem como um link**

- Para usar uma **imagem** como *link*, basta colocar a tag `<img>` dentro da tag `<a>`:

```
<a href="https://www.google.com.br/">
```

```

```

```
</a>
```



## Programação Web

### *Hyperlinks HTML*

- **Link para um endereço de e-mail**

- Usamos **mailto:** dentro do atributo **href** para criar um *link* que abre o programa de e-mail do usuário (para permitir que ele envie um novo e-mail):

```
<a href="mailto:someone@example.com">Send email</a>
```

- **Botão como um link**

- Para usarmos um botão HTML como um link, vamos precisar adicionar algum código *JavaScript*. *JavaScript* permite que você especifique o que acontece em certos eventos, como o **clique** de um botão:

```
<button onclick="document.location='default.js'">HTML Tutorial</button>
```





# Programação Web

## HTML Multimídia

- **HTML Multimídia**

- Multimídia na web é todo som, música, vídeo, filmes e animações que podem ser disponibilizadas para os usuários através de uma página HTML.
- Os primeiros navegadores da web tinham suporte apenas para texto, limitado a uma única fonte em uma única cor. Mais tarde, vieram os navegadores com suporte para cores, fontes, imagens e [multimídia](#)!
- O elemento HTML **<video>** é usado para mostrar um vídeo em uma página da web. Exemplo:

```
<video width="320" height="240" controls>  
  <source src="movie.mp4" type="video/mp4">  
  <source src="movie.ogv" type="video/ogg">  
  Seu navegador não suporta a tag vídeo.  
</video>
```



# Programação Web

## HTML Multimídia

- **HTML Multimídia**
  - Como funciona
    - O atributo de **controls** adiciona controles de vídeo, como reprodução, pausa e volume.
    - É uma boa ideia sempre incluir atributos de **largura** e **altura** (*width* e *height*). Se a altura e a largura não forem definidas, a página pode piscar enquanto o vídeo carrega.
    - O elemento **<source>** permite que você especifique arquivos de vídeo alternativos que o navegador pode escolher. O navegador usará o **primeiro** formato reconhecido.
    - O texto entre as tags **<video>** e **</video>** só será exibido em navegadores que **não** suportam o elemento **<video>**.



# Programação Web

## *HTML Multimídia*

- **HTML Multimídia**

- O elemento HTML **<audio>** é usado para reproduzir um arquivo de áudio em uma página da web.  
Exemplo:

```
<audio controls>  
  <source src="horse.ogg" type="audio/ogg">  
  <source src="horse.mp3" type="audio/mpeg">  
  Seu navegador não suporta o elemento áudio  
</audio>
```



# Programação Web

## HTML Multimídia

- HTML Multimídia
  - Áudio HTML - Como funciona
    - O atributo **controls** adiciona controles de áudio, como *play*, *pause* e *volume*.
    - O elemento **<source>** permite que você especifique arquivos de áudio alternativos que o navegador pode escolher. O navegador usará o **primeiro** formato reconhecido.
    - O texto entre as tags **<audio>** e **</audio>** só será exibido em navegadores que **não** suportam o elemento **<audio>**.



# Programação Web

## HTML Multimídia

- **Vídeos HTML do YouTube**

- Converter vídeos em formatos diferentes pode ser difícil e demorado. Uma solução mais fácil é permitir que o YouTube reproduza os vídeos em sua página da web.
  - **Id de vídeo do YouTube**
    - O YouTube exibirá um **id** (como *tgbNymZ7vqY*), quando você salva (ou reproduz) um vídeo. Você pode usar este id e referir-se ao seu vídeo no código HTML..

```
<iframe width="420" height="315"  
        src="https://www.youtube.com/embed/tgbNymZ7vqY">  
</iframe>
```



# Programação Web

## Estilos HTML - CSS

- **Estilos HTML - CSS**

- CSS significa *Cascading Style Sheets*.
- É usado para formatar o *layout* de uma página da web.

- **Usando CSS**

- Pode ser adicionado a documentos HTML de 3 maneiras:
  - **Inline** - usando o atributo `style` dentro dos elementos
  - **CSS Interno** - usando um elemento `<style>` na seção `<head>`
  - **Externo** - usando um elemento `<link>` para vincular a um arquivo CSS externo
- A maneira mais comum de adicionar CSS é manter os estilos em arquivos CSS **externos**. No entanto, no momento, usaremos estilos embutidos e internos, para facilitar o estudo.



# Programação Web

## Estilos HTML - CSS

- Estilos HTML - CSS

- *CSS inline*

- Um CSS embutido é usado para aplicar um estilo a um **único elemento** HTML.
    - Um CSS embutido usa o atributo **style** de um elemento HTML.
    - O exemplo a seguir define a cor do texto do elemento `<h1>` como azul e a cor do texto do elemento `<p>` como vermelho:

```
<h1 style="color:blue;">A Blue Heading</h1>
```

```
<p style="color:red;">A red paragraph.</p>
```



# Programação Web

## Estilos HTML - CSS

- Estilos HTML - CSS
  - *CSS interno*
    - Um CSS interno é usado para definir um estilo para uma **única página** HTML.
    - Um CSS interno é definido na seção **<head>** de uma página HTML, dentro de um elemento **<style>**.
    - O exemplo a seguir define a cor do texto de **TODOS** os elementos **<h1>** (nessa página) como azul e a cor do texto de **TODOS** os elementos **<p>** como vermelho.
    - Além disso, a página será exibida com uma cor de fundo "pó azul":

Exemplo no slide seguinte...





# Programação Web

## *Estilos HTML - CSS*

```
<!DOCTYPE html>
<html>
<head>
<style>
    body {background-color: powderblue;}
    h1    {color: blue;}
    p     {color: red;}
</style>
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```



# Programação Web

## *Tabelas HTML*

- Definir uma tabela HTML
- A tag **<table>** define uma tabela HTML.
  - Cada linha da tabela é definida com uma tag **<tr>**.
  - Cada cabeçalho da tabela é definido com uma tag **<th>**.
  - Cada dado/célula da tabela é definido com uma tag **<td>**.
  - Por padrão, o texto nos elementos **<th>** está em negrito e centralizado.
  - Por padrão, o texto nos elementos **<td>** são regulares e alinhados à esquerda.

Exemplo no slide seguinte...



# Programação Web

## *Tabelas HTML*

```
<table style="width:100%">
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```



# Programação Web

## Tabelas HTML

- Definir uma tabela HTML

- Adicionar uma borda:

- Para adicionar uma borda a uma tabela, use a propriedade CSS ***border***:

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

- Adicionar uma borda única:

- Para adicionar uma borda única (colapsada) a uma tabela, use a propriedade CSS ***border-collapse***:

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



# Programação Web

## *Tabelas HTML*

- Definir uma tabela HTML

- Adicionar um preenchimento à célula:

- Para adicionar um preenchimento à célula usamos a propriedade CSS ***padding***:

```
th, td {  
    padding: 15px;  
}
```

- Adicionar um espaço entre bordas

- Para definir o espaçamento da borda de uma tabela, use a propriedade CSS ***border-spacing***:

```
table {  
    border-spacing: 5px;  
}
```



# Programação Web

## *Listas HTML*

- **Definir uma lista HTML**
  - As listas HTML permitem que os desenvolvedores da web agrupem um conjunto de itens relacionados em listas.
- **Lista HTML não ordenada**
  - Uma lista não ordenada começa com a tag `<ul>`.
  - Cada item da lista começa com a tag `<li>`.
  - Os itens da lista serão marcados com marcadores (pequenos círculos pretos) por padrão:

```
<ul>  
  <li>Coffee</li>  
  <li>Tea</li>  
  <li>Milk</li>  
</ul>
```



# Programação Web

## *Listas HTML*

- Definir uma lista HTML
  - Lista HTML ordenada
    - Uma lista ordenada começa com a tag `<ol>`.
    - Cada item da lista começa com a tag `<li>`.
    - Os itens da lista serão marcados com um número por padrão:

```
<ol>  
  <li>Coffee</li>  
  <li>Tea</li>  
  <li>Milk</li>  
</ol>
```



# Programação Web

## *Listas HTML*

- **Blocos HTML e elementos embutidos**
  - Cada elemento HTML tem um valor de exibição padrão, dependendo de que tipo de elemento ele é. Existem dois valores de exibição: bloco e embutido.
- **Blocos HTML**
  - Um elemento de nível de bloco sempre começa em uma nova linha.
  - Um elemento de nível de bloco sempre ocupa toda a largura disponível (se estende para a esquerda e para a direita, tanto quanto pode).
  - Um elemento de nível de bloco tem uma margem superior e uma inferior, enquanto um elemento embutido não.

```
<div>Hello World</div>
```





# Programação Web

## *Listas HTML*

- **Blocos HTML e elementos embutidos**
  - **Elemento embutidos ou *inline***
    - Um elemento *inline* não começa em uma nova linha.
    - Um elemento embutido ocupa apenas a largura necessária.
    - Este é um elemento **<span>** dentro de um parágrafo.

`<span>Hello World</span>`



# Programação Web

## Listas HTML

- O elemento `<div>`
  - O elemento `<div>` geralmente é usado como um **contêiner** para outros elementos HTML.
  - O elemento `<div>` não tem atributos obrigatórios, mas estilo, classe e id são comuns.
  - Quando usado junto com CSS, o elemento `<div>` pode ser usado para estilizar blocos de conteúdo:

```
<div style="background-color:black;color:white;padding:20px;">
```

```
<h2>Londres</h2>
```

```
<p>Londres é a capital da Inglaterra. É a cidade mais populosa do Reino Unido, com uma área metropolitana de mais de 13 milhões de habitantes.</p>
```

```
</div>
```



# Programação Web

## *Listas HTML*

- O elemento `<span>`
  - O elemento `<span>` geralmente é usado como um **contêiner inline** para marcar uma parte de um texto ou uma parte de um documento.
  - O elemento `<span>` não tem atributos obrigatórios, mas estilo, classe e id são comuns.
  - Quando usado junto com CSS, o elemento `<span>` pode ser usado para estilizar partes do texto:

```
<p>Minha mãe tem olhos <span style="color:blue;font-weight:bold">azuis</span> e meu pai tem  
olhos <span style="color:darkolivegreen;font-weight:bold">verde escuros</span> como os  
meus.</p>
```



# Programação Web

## *Listas HTML*

- **O atributo HTML charset**

- Para exibir uma página HTML corretamente, um navegador da web deve conhecer o conjunto de caracteres usados na página. Isso é especificado na tag **<meta>**:
- Se não for especificado, UTF-8 será o conjunto de caracteres padrão em HTML.

`<meta charset="UTF-8">`

- **Entidades UTF-8**

- Muitos caracteres UTF-8 não podem ser digitados em um teclado, mas sempre podem ser exibidos usando números (chamados de números de entidade):
  - A é 65
  - B é 66
  - C é 67

# Programação Web

## *Listas HTML*

- Exemplo:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
</head>
<body>

<p>Vou exibir A B C</p>
<p>Vou exibir &#65; &#66; &#67;</p>

</body>
</html>
```



# Programação Web

## *Listas HTML*

- **Exemplo explicado:**
  - O elemento `<meta charset="UTF-8">` define o conjunto de caracteres.
  - Os caracteres A, B e C são exibidos pelos números 65, 66 e 67.
  - Para permitir que o navegador entenda que você está exibindo um caractere, você deve iniciar o número da entidade com `&#` e terminar com `;` (ponto e vírgula).
- **Entidades Emoji**
  - Emojis também são caracteres do alfabeto UTF-8:
    - 😊 é 128516;
    - 😍 é 128525;
    - 💕 é 128151.



# Programação Web

## Resumo

- **Fundamentos de HTML**

- Links;
- Multimídia;
- HTML CSS;
- Tabelas;
- Listas;
- Elementos:
  - *inline*
  - *block*
- Atributo *charset*
- Revisão.





# Referências



## Referências Básicas

SILVA, Maurício Samy. CSS3: desenvolva aplicações web profissionais com uso dos poderosos recursos de estilização das CSS3. São Paulo: Novatec, 2012.

SILVA, Maurício Samy. HTML 5: a linguagem de marcação que revolucionou a web. São Paulo: Novatec, 2011.

NIEDERAUER, Juliano. Desenvolvendo websites com PHP: aprenda a criar websites dinâmicos e interativos com PHP e bancos de dados. 2. ed. rev. e atual. São Paulo: Novatec, 2011.

## Referências Complementares

FLANAGAN, David. **o guia definitivo**. . O Really. 2012

SILVA, Maurício Samy. **Criando sites com HTML: sites de alta qualidade com HTML e CSS**. . Novatec. 2010

SOARES, Wallace. **PHP 5: conceitos, programação e integração com banco de dados**. . Érica. 2010

DALL'OGGIO, Pablo. **PHP: programando com orientação a objetos**. . Novatec. 2009

DEITEL, Paul J. Ajax,. **Rich Internet applications e desenvolvimento Web para programadores**. . Pearson Prentice Hall. 2009

IEPSEN, Edécio Fernandes. **Lógica de Programação e Algoritmos com JavaScript**. Novatec. 2018.

## Referências na Internet

<https://www.w3schools.com>

<https://developer.mozilla.org/pt-BR/docs/Web>

<https://illustrated.dev/advancedjs>