

# Notas da aula 1 - HTML e principais tags

---

## Protocolo HTTP

- **Protocolo** é o conjunto de regras sobre o modo como se dará a comunicação entre as partes envolvidas.
- O Hypertext Transfer Protocol, sigla **HTTP** (em português **Protocolo** de Transferência de Hipertexto) é um **protocolo** de comunicação utilizado para sistemas de informação de hipermídia, distribuídos e colaborativos. Ele é a base para a comunicação de dados da World Wide Web.

## Introdução ao HTML

---

- Entre todos os recursos envolvidos e necessários para criar uma aplicação web, o HTML é sem dúvida o mais importante.
- A essência de qualquer página web começa no HTML
- O HTML irá informar ao navegador o que cada elemento dentro da página representa, e sem ele não é possível saber o que cada coisa é em uma página web.

## HTML

- Desenvolvida em 1989 por Tim Berners-Lee e seus colaboradores no CERN (Conselho Europeu de Pesquisas Nucleares)
- Manter os documentos Organizados
- Em 1990 foi definida em especificações formais;
- Em 2000 a linguagem tornou-se uma norma internacional
- Versão Atual HTML5
- É uma linguagem de marcação utilizada na construção de páginas na Web.
- Documentos HTML podem ser interpretados por browsers
- Todo documento HTML possui marcadores (do inglês: *tags*), palavras entre parênteses angulares (chevron)(`<` e `>`)
- Esses marcadores são os comandos de formatação da linguagem.
- Um elemento é formado por um nome de marcador (*tag*), atributos, valores e filhos (que podem ser outros elementos ou texto)

## O que pode conter?

- Texto, imagens, Figuras, Folhas de Estilos, Cores...
- Quem padroniza tudo isso?

- W3C (World Wide Web Consortium)
- Estabelece compatibilidade das informações na Internet

É uma linguagem de marcação de hipertexto **Não é** uma linguagem de programação

- É uma linguagem de **marcação**!

## Evolução da HTML

- Desde a invenção da web, a HTML evoluiu por oito versões que são:
- HTML
- HTML +
- HTML 2.0
- HTML 3.0
- HTML 3.2
- HTML 4.0
- HTML 4.01
- HTML 5

## Tecnologias Web

### HTML

- Estruturar o conteúdo do documento

### CSS

- Uma *linguagem de folhas de estilos*. Isso quer dizer que o CSS permite que você aplique estilos seletivamente a elementos em documentos HTML.

### Javascript

- Fornece interação e lógica à página web
- Serve para manipular o comportamento da página

## Estrutura

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

```
</body>  
</html>
```

## Cabeçalhos

Headings (Cabeçalhos)

```
<h1>Do mais importante</h1>  
  
<h2></h2>  
  
<h3></h3>  
  
<h4></h4>  
  
<h5></h5>  
  
<h6>para o menos importante</h6>
```

## Parágrafo

```
<p> </p>
```

## Encode (Charset)

- URL encoding converte caracteres em um formato que pode ser transmitido pela internet.

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

## Criando o 1ª documento HTML

- Digite o exemplo mostrado pelo professor em um editor de texto simples, como o notepad, gedit, Notepad++, Sublime;
- Salve o arquivo com a extensão `.html`

## Negrito

```
<b>
```

- Colocando as palavras entre as tags `<b></b>` podemos exibir os caracteres em **negrito**.

## Itálico

- `<i>`
- Colocando as palavras entre as tags `<i></i>` podemos exibir os caracteres em itálico.

## Quebras de Linhas

- `<br>`
- Como já foi visto, o navegador exibirá automaticamente cada novo parágrafo ou título em uma nova linha. Mas se quiser adicionar uma nova quebra de linha no meio de um parágrafo?

## Linhas Horizontais

- `<hr>`
- Serve para criar uma quebra de linha entre os temas -- como uma alteração no tema em um livro ou uma nova cena em uma peça.

## Listas

Há muitas ocasiões em que precisamos usar listas. A HTML nos fornece três tipos diferentes.

- **Listas Ordenadas:** são listas onde cada item da lista é numerado. Ex: Conjuntos de passos para uma receita.
- **Listas não Ordenadas:** são listas que começam com um marcador (em vez de caracteres que indiquem ordem).

### Listas Ordenadas

- `<ol>` - A lista ordenada é criada com o elemento `<ol>`
- `<li>` - Cada item da lista é colocado entre uma tag de abertura `<li>` e uma tag de fechamento `</li>` (li significa item da lista).
- Utilizando o atributo `type=""` na tag `<ol></ol>` é possível indicar o tipo de ordenação. Assim:
- `type="1"` -- Lista com ordenação numérica
- `type="A"` -- Lista com ordenação alfabética Maiúscula
- `type="a"` - Lista com ordenação alfabética Minúscula
- `type="I"` -- Lista com ordenação Romana Maiúscula
- `type="i"` - Lista com ordenação Romana Minúscula

### Listas não Ordenadas

- `<ul>`- A lista não ordenada é criada com o elemento `</ul>`

- `<li>`- Cada item da lista é colocado entre uma tag de abertura `<li>` e uma tag de fechamento `</li>` (li significa item da lista).

## Links

### Criando Links

- Links são a característica definidora da Web, porque permitem pular de uma página web para outra -- possibilitando a própria ideia de navegar ou surfar na internet.
- Os tipos mais comuns de links:
  - Links de um site para outro
  - Links de uma página para outra no mesmo site
  - Links de uma parte da página web para outra parte da mesma página.
  - Links que abrem uma nova página do navegador
  - Links que iniciam o programa de e-mail e endereços um novo e-mail para alguém
- Links são Criados usando o elemento `<a>`
- Os usuários podem clicar qualquer coisa entre a tag de abertura `<a>` e a tag de fechamento `</a>`.
- Você especifica a página de destino do link utilizando o atributo `href`.

## Formulários

- Locais onde o usuário possa interagir com a página e enviar informações para serem processadas.
- Exemplo:
  - Formulário de Cadastro
  - Formulário de Login
  - Campo de Pesquisa
- Um formulário HTML é feito de um ou mais `widgets`.
- Esses widgets podem ser **campos de texto** (linha única ou de várias linhas), **caixas de seleção**, **botões**, **checkboxes** ou **radio buttons**.
- Na maior parte do tempo, estes elementos possuem uma legenda que descreve o seu objetivo.

## Formulários

### Tag `<form>`

- A tag `<form>` não apresenta uma referência visual quando criada, mas representa semanticamente um formulário.

## Atributos

- Atributos da tag `<form>`:

- `autocomplete="on|off"` - A medida em que usamos os campos de um formulário o navegador irá começar a sugerir valores para o que estamos digitando, com base em valores que foram digitados anteriormente.
- `action="string"` - Este atributo especifica para onde os dados do formulário serão enviados. Normalmente será o link de uma página no servidor:
- `method="get|post"` - Especifica o método do HTTP que será utilizado para a submissão (get ou post):
- `novalidate` - Impede a validação dos campos antes da submissão
- `target="_blank|self"` - Assim como na tag `<a>`, o target indica onde a página de destino será aberta.
- `target="_blank"` serve para abrir um link em uma nova aba do navegador.

## Elementos

- Elementos (Widgets) que podem ser utilizados no formulário:
- Os formulários possuem campos onde inserimos dados.
- Os campos mais comuns dos formulários são criados pela tag `<input>`, que tem o atributo type onde indicamos o tipo de campo que queremos.

## Elementos

- **Inputs de Texto**
- Para escrever um pequeno texto é utilizado a tag `<input>` (auto close)

### Atributos basicos tag `<input>`

#### **type = "text" (valor padrão é o text)**

- `placeholder` - texto que aparece quando o campo está vazio;
- `maxlength` - quantidade máxima de caracteres
- `required` - indica que o campo é necessário, não permitindo submeter o formulário se o campo estiver em branco;
- `disabled` - deixa o campo desabilitado;
- `readonly` - impede edições no campo, permitindo apenas que o usuário leia o valor.

## Elementos

- **Email, Search e URL**
  - Esses tipos são parecidos com o tipo "text"

- Teclados de tablets e smartphones se adaptam a cada tipo de input
- Tipos como **email** e **url** possuem validação
- **Exemplos**
- Validação Email e URL (ver na aula)

## Number

- Indica um campo do tipo **number**.
- Em computadores o campo fica com botões para aumentar e diminuir o número.
- Em dispositivos com teclado virtual é exibido um teclado numérico.

## Elemento **Radio Button**

- Cria um Radio Button, onde podemos selecionar apenas uma opção. Para obter a seleção exclusiva é necessário colocar o mesmo atributo "name".
- Isso faz com que o navegador entenda que ambos pertencem ao mesmo campo.
- Se passarmos nomes diferentes poderíamos selecionar os dois valores, o que não é o objetivo dos Radio Buttons.

## Atributos básicos tag `<input type="radio"/>`

- `checked="checked"` - Seleciona um elemento no checkbox

## Elemento **Checkbox**

- Diferente do Radio Button, onde só é possível selecionar uma opção, o CheckBox serve para selecionarmos mais de uma opção.

## Elemento **Date, Time e Datetime-local**

- **Datetime-local** - Serve para selecionar o tempo (horas, minutos).

## Elemento **Button, reset e submit**

- Esses tipos criam um botão.
- O tipo `"submit"` é usado para que o usuário clique quando quiser enviar o formulário.
- O tipo `"reset"` serve para quando o usuário quiser reiniciar o formulário. Cuidado, pois ao clicar o formulário inteiro será apagado!

## Elemento **Select**

- A tag `<select>` nos permite criar uma lista onde podemos selecionar um item. Cada uma das opções é criada pela tag `<option>`.

- O atributo "value" da tag option é o valor a ser enviado ao servidor, e o valor escrito dentro da tag é o que será exibido ao usuário.

### Atributos básicos tag `<select>`

- `selected` - Seleciona um elemento no select

### Elemento **Fieldset e Legend**

- Serve para criarmos um campo dentro do formulário para poder melhor organizar os campos.
- A tag `<legend>` serve para darmos um título à seção.

### Elemento **Label**

- Esta tag serve para darmos nomes aos campos. Se passarmos o id do input no atributo "for", estaremos indicando ao navegador que o Label é referente àquele campo.
- Quando o usuário clicar no texto do Label, o input receberá foco, permitindo que o usuário já entre com os dados.

### Dúvidas

**E-mail:** edkallennl@unipe.edu.br