

Tema 5: Formulário

O que são Formulários?

Os formulários em html são usados como uma das principais ferramentas de interatividade entre os usuários e um site ou aplicativo. Através deles que os usuários enviam dados para o site, sendo que na maior parte do tempo esses dados são enviados para um servidor web, porém a página web pode requisitar para usá-los em qualquer momento. Em geral quando necessitamos que o usuário forneça algum tipo de informação, é pelo formulário que fazemos esse pedido, que por sua vez envia os dados para um banco de dados, para o armazenamento ou utilização em outra página. Existem inúmeras finalidades na qual podemos usar os formulários, vamos conhecer as principais tags e aprender a utilizar o formulário em html.

A principal diferença entre um formulário html e um documento simples html é que na maioria das vezes, o dado coletado é enviado ao servidor. Nesse caso, é necessário ter um servidor web configurado para receber e processar os dados, vamos aprender a configurar um servidor web mais adiante, nesse capítulo vamos nos atentar para o formulário e seus atributos. Os formulários necessitam de validações em seus campos, é até possível fazer essas validações em html, mas no html 5, é indicado fazer as validações no código JavaScript incorporado no formulário.

Antes de codificar um formulário, é de grande ajuda ter ele **desenhado em algum arquivo**, ou até mesmo numa folha de papel, para facilitar na hora de codificar e não esquecer nenhum campo importante.

Criando um Formulário

Todo formulário em html deve começar **pela tag <form> e terminar fechando-a </form>**, esse elemento não é visual (quando colocamos somente a tag <form> na página, ao executar no navegador não nos mostra nada, é apenas para indicar que naquele local determinado tem um formulário), ela precisa de outros elementos para que o formulário ganhe vida. Vamos conhecer as tags do formulário e seus atributos.

Dentro da tag <form> deve ser configurado alguns atributos para indicar qual a finalidade daquele formulário, podemos identificar um formulário através do **atributo id="nome_do_formulário"**, em casos em que temos mais que um formulário na página, também temos o atributo **method (método) que define o método HTTP para o envio de dados, ele pode ser method="get" ou method="post"**. Vamos ver as diferenças de cada um:

GET = Os dados serão enviados junto com a URL, e desta forma, **serão visíveis para os usuários**, que podem fazer alterações, onde pode gerar alguns problemas e incidentes quanto a segurança dos dados.

POST = Os dados serão **enviados no corpo da mensagem HTTP, sendo assim serão invisíveis para os usuários, o método mais indicado para envio de dados**.

Outro atributo da tag <form> é o autocomplete, este atributo permite desabilitar(off) o comportamento padrão da maioria dos navegadores modernos, os quais sugerem valores para os campos baseados em preenchimentos anteriores, ou seja, esse atributo desabilita o autocompletar dos campos de dados pelos navegadores, mas sempre que se achar necessário ser habilitado pode ser colocado o valor (on).

Os formulários em html possui alguns elementos essenciais para seu bom funcionamento, vamos conhecer alguns deles para compreender melhor o uso dos formulários;

INPUT – Com a nova versão do HTML (5.0), esse elemento ganhou novos types (tipos), onde possibilita atribuir mais controles, porém ainda temos alguns navegadores que não reconhecem esses novos types, adotando assim ao valor padrão **<input type="text">**. O elemento input é um elemento que permite a entrada de vários tipos de dados, seu comportamento é definido pelo atributo type, dentre os tipos de entrada de dados usados pela tag <input>, podemos destacar alguns:

TEXT – Esse atributo permite a entrada de textos curtos, com limitação de uma linha, não apresenta restrições quanto ao conteúdo utilizado, e não possui validação. Além desse atributo, **o ideal** é que usamos o atributo id, que é usado para estilizar o elemento utilizando o CSS, e o atributo name, que serve para determinar o nome do campo, assim podendo ser identificado quando for enviado para o servidor na forma de uma variável, que por sua vez terá o valor igual ao conteúdo do atributo value.

LABEL – Faz a definição de um rótulo para melhorar o controle de campos.

LEGEND – Define um título para um conjunto de campos.

Exemplo de código:

```
<form>
  Nome <input type="text" name="nome" id="txtNome">
  Sobrenome <input type="text" name="sobrenome" id="txtSobrenome">
</form>
```

CHECKBOX – Usado juntamente com a tag input, esse atributo cria uma variada cadeia de opções selecionáveis, que podem ser escolhidas pelo usuário, permitindo também a seleção de mais de uma opção.

```
<html>

<head>
  <title> Formulários </title>
</head>

<body>
  <h1> Formulário </h1>
  <form>
    Selecione um ou vários gêneros:
    <input type="checkbox" name="generoFilme" value="1"> Ação
    <input type="checkbox" name="generoFilme" value="2"> Aventura
    <input type="checkbox" name="generoFilme" value="3"> Comédia
    <input type="checkbox" name="generoFilme" value="4"> Fantasia
    <input type="checkbox" name="generoFilme" value="5"> Ficção
    <input type="checkbox" name="generoFilme" value="6"> Terror
  </form>
</body>
```

```
</html>
```

Podemos observar que todos os elementos possuem o mesmo valor no atributo “name”, neste caso “generoFilme”, mas cada um está associado um valor diferente no atributo value.

RADIO – Esse atributo é semelhante ao anterior criando uma lista de opções, porém permitindo somente uma opção para ser selecionada.

```
<html>
```

```
<head>
```

```
<title> Formulários </title>
```

```
</head>
```

```
<body>
```

```
<h1> Input Radio </h1>
```

```
<form>
```

```
  Selecione uma Região:
```

```
  <input type="radio" name="regiao" value="sul"> Sul
```

```
  <input type="radio" name="regiao" value="sudeste"> Sudeste
```

```
  <input type="radio" name="regiao" value="norte"> Norte
```

```
  <input type="radio" name="regiao" value="nordeste"> Nordeste
```

```
  <input type="radio" name="regiao" value="centroeste"> Centro Oeste
```

```
</form>
```

```
</body>
```

```
</html>
```

Neste caso é importante lembrar que todos os elementos radio, que estão relacionados devem possuir o mesmo name.

PASSWORD – Esse atributo é muito utilizado para criar campos de entrada de senhas, protegendo os elementos visualmente, substituindo por um carácter **padrão o “ * ”**. Dificultando os curiosos. Exemplo de código:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Formulários</title>
```

```
</head>
```

```
<body>
```

```
<h1>Input password</h1>
```

```
<form>
```

```
  Senha: <input type="password" name="senha"> <br /><br />
```

```
  Confirmação: <input type="password" name="confirmacao">
```

```
        </form>
    </body>

</html>
```

DATE – Esse de tipo de atributo aceita apenas datas. Em alguns navegadores quando este campo é utilizado ele apresenta um calendário no qual podemos escolher uma data específica, em outros casos ele apresenta apenas uma máscara mostrando um formato que a informação será mostrada.

```
<!DOCTYPE html>
<html>

<head>
    <title>Formulários</title>
</head>

<body>
    <h1>Input Date</h1>
    <form>
        Data de Nascimento: <input type="date" name="nascimento" min="1950-01-01"
max="2020-12-31">
    </form>
</body>

</html>
```

Podemos determinar um intervalo de datas válidas, com os atributos “min” e “max”, sempre usadas no formato de datas norte americano ano-mês-dia, onde o ano deve conter 4 dígitos.

COLOR – Este atributo fornece ao usuário um elemento que permite selecionar uma cor, dependendo do navegador este elemento é apresentado de formas diferentes. Em geral o valor retornado está no formato hexadecimal (#AABBCC).

```
<!DOCTYPE html>
<html>

<head>
    <title>Formulários</title>
</head>

<body>
    <h1>Input Color</h1>
    <form>
        Selecione uma cor: <input type="color" name="cor" value="#000000">
    </form>
</body>
```

</html>

EMAIL – Este atributo permite uma verificação do conteúdo do campo para garantir que o formato digitado seja um email válido, pode ser enviado em branco, salvo com o uso do atributo required. Em casos que o atributo multiple esteja presente, é possível submeter mais de um endereço de email válido, para isso devemos separar cada email com uma vírgula (,). Exemplo do código:

```
<!DOCTYPE html>
<html>

<head>
  <title>Formulários</title>
</head>

<body>
  <h1>Input E-mail</h1>
  <form>
    E-mail: <input type="email" name="email" required multiple>
    <input type="submit" value="Enviar">
  </form>
</body>

</html>
```

NUMBER – Permite a entrada de dados numéricos permitindo definir através dos atributos “min” para o valor mínimo e “max” para o valor máximo. Outro atributo que é usado é o “step” que define o valor do incremento e decremento do campo.

Exemplo de código:

```
<!DOCTYPE html>
<html>

<head>
  <title>Formulários</title>
</head>

<body>
  <h1>Input Number</h1>
  <form>
    Ano de Nascimento: <input type="number" name="anoNascimento" min="1910"
max="2018" step="1" required>
  </form>
</body>

</html>
```

RANGE – Este atributo em geral apresenta uma barra horizontal, cujo valores são limitados pelos

atributos “min” e “max”, outro atributo usado é o “step”. Exemplo do código:

```
<!DOCTYPE html>
<html>

<head>
  <title>Formulários</title>
</head>

<body>
  <h1>Input Range</h1>
  <form>
    Barra de Range: <input type="range" name="barra" min="0" max="5000" step="100"
required>
  </form>
</body>

</html>
```

TIME – Este atributo permite ao usuário colocar um horário no formato de 12 ou 24 horas, com o formato hh:mm, em casos de 12 horas com a sigla AM/PM, e em casos 24 horas hh:mm:ss, isso dependendo do navegador. Exemplo do código:

```
<!DOCTYPE html>
<html>

<head>
  <title>Formulários</title>
</head>

<body>
  <h1>Input Time</h1>
  <form>
    Selecione o horário da Consulta: <input type="time" name="horaConsulta" required>
  </form>
</body>

</html>
```

URL – Este atributo é usado quando precisamos de uma validação do campo para verificar uma url válida digitada pelo usuário. Exemplo do código:

```
<!DOCTYPE html>
<html>

<head>
  <title>Formulários</title>
</head>
```

```

<body>
  <h1>Input URL</h1>
  <form>
    Informe o endereço do site: <input type="url" name="enderecoSite" required>
    <input type="submit" value="Enviar">
  </form>
</body>

</html>

```

HTML5 – Novas TAGS

Esses atributos do <input> apresentados até aqui são atributos que fazem parte do html já algum tempo, com a chegada da versão 5.0 do html, veio também alguns novos atributos que ajudam a interatividade com o usuário e a praticidade na implementação, vamos conhecer os principais.

PLACEHOLDER – Este atributo mostra uma “dica” de como o campo dever ser preenchido, na versão anterior só era possível com auxílio de scripts (JavaScript ou jQuery). Essa “dica” aparece no campo até que o usuário começa a digitar algo. As informações do placeholder ficam visíveis somente enquanto o campo está vazio, e não influencia no valor do campo em si. Exemplo de código:

```

<!DOCTYPE html>
<html>

<head>
  <title>Formulários</title>
</head>

<body>
  <h1>Inputs HTML5</h1>
  <form>
    <b>Nome Completo:</b>
    <input type="text" name="nome" placeholder="Digite seu nome">
    <br><br>

    <b>Data de nascimento:</b>
    <input type="date" name="nasc" placeholder="dd/mm/aa">
    <br><br>

    <b>Número de telefone:</b>
    <input type="tel" name="tel" placeholder="(DDD)#####-####">
  </form>
</body>

</html>

```

REQUIRED – Este atributo quando presente auxilia na validação do campo, o formulário não será “submetido/enviado” caso o campo esteja vazio. Veja o exemplo do código e navegador:

```
<!DOCTYPE html>
<html>

<head>
  <title>Formulários</title>
</head>

<body>
  <h1>Inputs HTML5</h1>
  <form>
    <b>CPF : </b>
    <input type="number" name="campo" required>
    <input type="submit" value="Enviar">
  </form>
</body>

</html>
```

AUTOFOCUS – Esse atributo é possível implementá-lo somente em um campo do formulário, pois sempre que o navegador carregar a página onde está o formulário o campo terá o foco, sendo assim o cursor aparecerá nesse campo predefinidamente. Este atributo garante que ao carregar a página do formulário, sempre seremos direcionados para o mesmo campo. Exemplo do código:

```
<!DOCTYPE html>
<html>

<head>
  <title>Formulários</title>
</head>

<body>
  <h1>Inputs HTML5</h1>
  <form action="http://www.google.com.br">
    <input type="number" name="numero" required autofocus>
    <input type="submit" value="Enviar"><br><br>
    <input type="text" name="nome" placeholder="Digite seu Nome">
  </form>
</body>

</html>
```

Devemos colocar o atributo no campo que terá o foco toda vez que a página será carregada, pode-se colocar em qualquer campo em específico, só devemos lembrar que esse atributo pode ser usado somente em um campo do formulário.

MAXLENGTH – Esse atributo permite a limitação da quantidade máxima de caracteres que podem ser inseridos na tag <input>, ideal para limitar campos com o type="text". Vamos ver o código como exemplo.

```
<!DOCTYPE html>
<html>

<head>
  <title>Formulários</title>
</head>

<body>
  <h1>Inputs HTML5</h1>
  <form action="#" method="post">
    <h2>Digite seu Primeiro Nome (limite de 20 caracteres)</h2>
    <input type="text" name="primeiroNome" maxlength="20" required>
    <input type="submit" value="Enviar">
  </form>
</body>

</html>
```

AUTOCOMPLETE - Em geral por padrão, os navegadores oferecem a função de autocomplete em formulários, porém em alguns momentos seja necessário desabilitar esta função, por exemplo, quando precisamos confirmar o endereço de um e-mail, login ou senha. Exemplo do código:

```
<!DOCTYPE html>
<html>

<head>
  <title>Formulários</title>
</head>

<body>
  <h1>Inputs HTML5</h1>
  <form action="#" method="post">
    <h3>Digite seu email:</h3>
    <input type="email" name="email" required>
    <h3>Confirme seu email:</h3>
    <input type="email" name="confirmacaoEmail" required autocomplete="off">
    <input type="submit" value="Enviar">
  </form>
</body>

</html>
```

SELECT – Essa tag utilizamos quando desejamos criar uma estrutura no modelo "drop-down", que mostrar para o usuário uma série de opções selecionáveis. Dependendo dos parâmetros

implantados no código, os usuários poderão selecionar uma ou mais opções. Exemplo do código:

```
<!DOCTYPE html>
<html>

<head>
  <title>Formulários</title>
</head>

<body>
  <h1>Tag Select</h1>
  <form>
    Selecione o Estado:
    <select name="estado">
      <option value="DF">Brasília</option>
      <option value="RJ">Rio de Janeiro</option>
      <option value="SP">São Paulo</option>
    </select>
  </form>
</body>

</html>
```

Podemos notar que o modo no qual escrevemos essa tag é diferente, sendo aberto a tag com <select> e após seu término fechando – a com </select>, onde todas as opções são usadas com ao atributo <option> ... </option>.

TEXTAREA – **Diferente da tag <input>**, que permite o usuário fornece apenas uma linha de dados, esta tag permite que seja fornecido uma quantidade de informações muito superior. Permitindo a inserção de texto de múltiplas linhas, sendo muito útil para armazenar mensagens mais longas ou trechos de textos.

Pode ser inserido 2 atributos para determinar o número máximo de linhas e colunas dentro da área de texto, o atributo rows é usado para limitar o número de linhas, e o atributo cols a quantidade de colunas. Exemplo do código:

```
<!DOCTYPE html>
<html>

<head>
  <title>Formulários</title>
</head>

<body>
  <h1>Tag TextArea</h1>
  <form>
    <h3>Deixe sua mensagem aqui:</h3>
    <textarea rows="5" name="mensagem"></textarea>
  </form>
```

</body>

</html>

BUTTON – É possível adicionar botões aos formulários de duas maneiras, primeiro é usando a tag <input> e segundo é usando a tag <button>. Os botões no formulário são de extrema importância, pois são através deles que a maioria das interações são acionadas. Uma das principais funções é invocar o método que realiza o envio/submissão dos dados do formulário para um servidor ou página em PHP.

Essa tag tem o modo de utilização sendo aberta com <button> e fechada com </button>. Vamos ver um exemplo do código:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <title>Formulários</title>
```

```
</head>
```

```
<body>
```

```
    <h1>Tag Button</h1>
```

```
    <form action="seu_script_php.php" method="post">
```

```
        <button type="submit">Enviar</button>
```

```
    </form>
```

```
</body>
```

```
</html>
```

Assim como a tag <input>, a tag <button> possui uma série de atributos importantes para diversas utilizações. É importante conhecermos os mais usados para deixar nossos formulários mais funcionais e práticos. Vamos conhecer os principais atributos da tag <button>:

TYPE – Determina o tipo do botão, ele aceita alguns valores como; (button), para botão no formato padrão, (reset), para um botão com a finalidade de resetar os campos preenchidos para um valor padrão, e (submit), para criar um botão com a função de enviar os dados do formulário.

NAME – É usado para nomear o botão.

VALUE – Usamos o value para definir o nome/valor que aparece no botão.

FORMACTION - Define a URL onde os dados submetidos serão processados.

AUTOFOCUS – Determina que o botão deve receber o foco sempre quando a página for carregada.

DISABLED – Faz com que o botão seja desabilitado e não responda aos eventos, como por exemplo ao click ou passar do mouse.

FORMMETHOD – Especifica como será realizada a submissão do formulário, seja no método GET

ou POST.

FORMNOVALIDATE – Determina que o formulário seja submetido/enviado sem ser validado, esse atributo só deve ser usado nos botões tipo submit.

OBJETOS E IFRAMES

Novas tags adicionadas como a <object> e a <iframe> vieram para ajudar na implementação e uso do html, e deixar mais leves a prática de adicionar objetos multimídia na página, vamos conhecer essas tags com mais detalhes:

<object> ... </object> - Esta tag é utilizada para incorporar tipos de mídias adicionais à página. Podendo ser áudio, vídeo, documento em pdf, entre outros. Com alguns atributos importantes para a utilização como width (largura) e height (altura) para ajustar as dimensões onde o objeto será adicionado e o type que determina o tipo do conteúdo exibido.

<iframe> ... </iframe> - Essa tag é uma espécie de bloco de conteúdo em linha, que é utilizado como um contêiner flexível para arquivos de mídia. Ele tem a posição flutuante dentro da página, significando que é colocado relativo à outros itens da página. Conheça alguns atributos da tag iframe.

NAME – Determina o nome do frame.

SRC – A fonte URL/caminho do objeto multimídia para ser inserido dentro do iframe.

SRCDOC – Insere um conteúdo HTML para ser mostrado dentro do iframe.

WIDTH – Determina a largura do iframe.

HEIGHT – Determina a altura do iframe.

<PARAM/> - Serve para customizar o iframe. Incluindo parâmetros adicionais para ir junto com o conteúdo.

<embed> ... </embed> - É usado para incorporar na página objetos externos, como plugins (flash player entre outros).

RESPONDA:

- 1) O que é um formulário HTML?
- 2) Qual a principal diferença entre um formulário em HTML e um documento simples HTML?
- 3) Para a tag <form> podem ser atribuídos o atributo _____ (atributo onde define o local através da url) e o _____ (atributo que define o método HTTP).
- 4) Qual a diferença entre os métodos de um formulário HTML?
- 5) Qual a tag usada para usada para adicionar uma área de texto no formulário?
- 6) Para inserir um botão no formulário eu uso a tag _____ ou a tag _____.
- 7) A tag <input type='date'> me retorna qual valor?
- 8) Qual a função da tag <map> no código em HTML.
- 9) Qual tag utilizo para inserir um campo de texto?
- 10) Qual o uso da tag <label>?

1-Um formulário HTML é uma ferramenta de interatividade entre usuários e um site ou aplicativo. Ele permite que os usuários forneçam dados, que são enviados para um servidor web para processamento ou armazenamento.

2-A principal diferença é que, na maioria das vezes, os dados coletados por um formulário HTML são enviados ao servidor. Isso requer um servidor web configurado para receber e processar os dados.

3-Os atributos para a tag <form> incluem o atributo action (atributo onde define o local através da URL) e o atributo method (atributo que define o método HTTP).

4-A diferença entre os métodos de um formulário HTML está no modo como os dados são enviados:

5-GET: Os dados são enviados junto com a URL, visíveis para os usuários.

POST: Os dados são enviados no corpo da mensagem HTTP, invisíveis para os usuários.

A tag usada para adicionar uma área de texto no formulário é <textarea>.

6-Para inserir um botão no formulário, você pode usar a tag <input> com o atributo type="submit" para um botão de envio. Ou, alternativamente, pode usar a tag <button>.

7-A tag <input type='date'> retorna uma data que pode ser selecionada pelo usuário.

8-A tag <map> em HTML é usada junto com as tags <area> para criar mapas de imagem interativos.

9-A tag utilizada para inserir um campo de texto é <input> com o atributo type="text".

10-A tag <label> é usada para definir um rótulo associado a um campo de formulário, melhorando a acessibilidade e usabilidade.