



Criação de formulários com HTML

Introdução

Os formulários HTML constituem um dos pontos de interação mais significativos entre um usuário e um website ou aplicativo. Um desenvolvedor web tem a capacidade de criar formulários HTML baseados em modelos específicos, utilizando os elementos HTML pertinentes, visto que esses formulários são construídos com base em elementos de formulário HTML. Estes últimos são utilizados para a criação de variados tipos de campos de formulário, como caixas de texto, botões, caixas de seleção, entre outros.

Os formulários HTML são processados pelos navegadores, e os dados são enviados para o servidor web. A maioria dos navegadores executa o processamento de formulários HTML em conformidade com as especificações do W3C para HTML5.

Objetivos da aula

- Definir o conceito de formulários HTML?
- Descrever a estrutura básica de um formulário em HTML
- Discutir os principais elementos aplicados a um formulário HTML.

Resumo

O que são formulários? Estrutura básica de um formulário

Os formulários são estruturas utilizadas para coletar a entrada de dados do usuário e enviar para o servidor.




A construção de um formulário inicia-se sempre com a *tag* `<form>` e termina com a *tag* `</form>`. Esses elementos são um contêiner para diferentes tipos de campos de entrada de dados, tais como: campos de texto, caixas de seleção, botões de opção, botões de envio etc. O elemento `<form>` possui alguns atributos referente à forma como o formulário será submetido:

- O atributo *action* define recebe como valor o destino dos dados que serão submetidos no formulário, que, normalmente é uma página em uma linguagem de backend como PHP, Java, ASP etc. Ex.: `<form action="/action_page.php">`
- O atributo *target* especifica onde exibir a resposta recebida após o envio do formulário, como por exemplo *_blank* (em uma nova janela), *_self* (na janela atual), *_parent* (no quadro pai), *_top* (em todo o corpo da janela) ou *framename* (em um iframe nomeado). Ex.: `<form action="/action_page.php" target="_blank">`
- O atributo *method* especifica o método HTTP a ser usado ao enviar os dados do formulário, que podem ser enviados como variáveis de URL (method="get") ou como transação HTTP post (method="post"). Exemplo: `<form action="/action_page.php" method="post">`

Tipos de campos para envio de dados


O elemento HTML `<input>` é o mais comumente usado nos formulários. Um elemento `<input>` pode ser exibido de várias maneiras, dependendo do valor do atributo *type*, por exemplo:

- `<input type="text">` Exibe um campo de entrada de texto de uma linha. 
- `<input type="radio">` Exibe um botão para cada item de uma série de opções, onde somente um item pode ser selecionado.
- `<input type="checkbox">` Exibe uma caixa de seleção para cada item de uma série de opções, onde pode-se selecionar mais de um item.
- `<input type="submit">` Exibe um botão enviar (para submeter o formulário)
- `<input type="button">` Exibe um botão clicável
- `<input type="date">` Exibe (ao clicar no campo) um calendário para selecionar uma data.
- `<input type="email">` Exibe um campo para entrada de um endereço de e-mail. O valor de entrada é validado automaticamente para garantir que seja um endereço de e-mail formatado corretamente.

Outro campo muito utilizado é a lista suspensa, que é definida pelo elemento `<select>` seguido das opções da lista, definidas pelos elementos `<option>`

```
<select id="carros" name="carros">
  <option value="gol">Gol</option>
  <option value="palio">Pálio</option>
  <option value="hb20">HB20</option>
  <option value="clio">Clio</option>
</select>
```

Todo campo do formulário deve ter um atributo *name* que deverá receber o nome do campo (sem espaços) para que o servidor possa pegar o valor enviado.

Para atribuir um rótulo ao campo, utiliza-se o elemento `<label>`. É importante a utilização deste elemento para que leitores de tela identifiquem o campo  pessoas com deficiência visual. Este elemento ajuda os usuários que têm dificuldade em clicar em regiões muito pequenas pois ao clicar no texto do `<label>`, ele seleciona o campo, mas, para isso, é necessário utilizar o atributo *for*, que deve receber o mesmo valor do atributo *id* do campo a que se refere, como no exemplo a seguir:

```
<label for="linguagem">Digite aqui a sua linguagem favorita</label>
```

```
<input type="text" id="linguagem" name="linguagemFavorita">
```

Abaixo um exemplo de formulário com campos de entrada de texto para nome e sobrenome. Veja que comumente, os valores dos atributos *for*, *id* e *name* são iguais.

```
<form>
```

```
    <label for="nome">Nome</label><br>
```

```
    <input type="text" id="nome" name="nome"><br>
```

```
    <label for="sobrenome">Sobrenome:</label><br>
```

```
    <input type="text" id="sobrenome" name="sobrenome">
```

```
</form>
```

Podemos também usar o elemento `<fieldset>` para agrupar campos do formulário, como no exemplo a seguir:

```
<form action="/action_page.php">
```

```
    <fieldset>
```

```
        <legend>Pessoa:</legend>
```

```
        <label for="nome">Nome:</label><br>
```

```
        <input type="text" id="nome" name="nome" value="John"><br>
```

```
        <label for="sobrenome">Sobrenome:</label><br>
```

```
        <input type="text" id="sobrenome" name="sobrenome" value="Doe"><br>
```

```
        <input type="submit" value="Submit">
```

</fieldset>

</form>



Podemos utilizar o atributo `readonly` para especificar que um campo de entrada é somente leitura. Um campo de entrada somente leitura não pode ser modificado.

Podemos usar também atributo `maxlength` para limitar número máximo de caracteres que podem ser inseridos em um campo.

A seguir mais um exemplo:

<form>

<label for="nome">Nome:</label>

<input type="text" id="nome" name="nome" value="John" readonly>

<label for="sobrenome">Sobrenome:</label>

<input type="text" id="sobrenome" name="sobrenome" value="Doe" maxlength>

</form>


Conteúdo Bônus

Criando um Formulário na sua Página Web

Aprenda a adicionar um formulário em sua página web com este passo a passo:

1. Comece abrindo uma nova seção em sua página web com a tag <section>.
2. Dentro desta seção, abra o formulário com a tag <form action="" method="get">.
3. Vamos adicionar um campo para o usuário inserir o seu nome. Para isso, utilize a tag de rótulo <label for="nome">Nome:</label>
. Em seguida, adicione o campo para o nome com a tag <input type="text" id="nome" name="nome" value="John" readonly>

.

4. Agora, vamos adicionar um campo para o sobrenome do usuário. Para isso, use o rótulo `<label for="sobrenome">Sobrenome:</label>
`. Em seguida, inclua  campo com a tag `<input type="text" id="sobrenome" name="sobrenome" value="Doe" maxlength="

`.

5. Vamos criar um campo em forma de lista suspensa para o usuário escolher sua linguagem preferida. Adicione um rótulo para a lista com `<label for="linguagem">Escolha sua linguagem preferida:</label>
`. Abra a lista com a tag `<select id="linguagem" name="linguagem">`.

6. Dentro desta lista, adicione os itens com as tags:

```
<option value="html">HTML</option>
```

```
<option value="css">CSS</option>
```

```
<option value="javascript">Javascript</option>
```

Em seguida, feche a lista suspensa com `</select>

`.

7. Agora, inclua um botão para que o usuário possa enviar os dados do formulário com a tag `<input type="submit" value="Enviar">`

8. Feche o formulário com a tag `</form>` e, em seguida, feche a seção com `</section>`.

9. Finalmente, volte ao navegador e atualize a página para visualizar as alterações.

Este guia rápido lhe ajudará a entender melhor como criar e configurar um formulário em sua página web. Pratique e aperfeiçoe suas habilidades!



Dica quente para você não esquecer

O elemento `<label>`:

- Define rótulos para os campos do formulário.
- É útil para usuários que necessitam de leitores de tela, porque o leitor lerá em voz alta o rótulo quando o usuário focar no campo dele.
- Ajuda os usuários que têm dificuldade em clicar em regiões muito pequenas (como botões de opção ou caixas de seleção) - pois quando o usuário clica no `<label>`, ele seleciona o campo deste rótulo
- O atributo *for* do `<label>` deve ser igual ao atributo *id* do campo a que ele se refere.

O atributo `method` do formulário

Observações sobre o método GET:

- Anexa os dados do formulário à URL, em pares nome/valor
- NUNCA use GET para enviar dados confidenciais! (os dados do formulário enviado são visíveis na URL!)
- O comprimento de um URL é limitado (2048 caracteres)
- Este método é útil para envios de formulários em que um usuário deseja marcar o resultado

- GET é bom para dados não seguros, como strings de consulta no Google



Observações sobre o POST:

- Anexa os dados do formulário dentro do corpo da solicitação HTTP (os dados do formulário enviado não são mostrados na URL)
- O POST não tem limitações de tamanho e pode ser usado para enviar grandes quantidades de dados.

Referência Bibliográfica

MDN Web Docs. [S. l.], 20 set. 2022. Disponível em:

<https://developer.mozilla.org/pt-BR/docs/Learn/Forms>. Acesso em: 20 set. 2022.

Atividade Prática 7: Criação de formulários com HTML

Título da Prática: Criação de Formulário de Contato HTML

Objetivo: O objetivo desta atividade é permitir que os alunos apliquem os conceitos aprendidos na aula “Criação de formulários com HTML”, criando um formulário de contato HTML que incorpora diferentes tipos de campos de entrada.

Materiais, Métodos e Ferramentas:

1. Computador com acesso à internet
2. Editor de texto (como o Sublime Text, Atom, Visual Studio Code, etc.)
3. Navegador web para visualização do resultado (como Google Chrome, Firefox, etc.)



Descrição da atividade

1º passo: Crie uma nova página HTML chamada “contato.html”.

2º passo: Na página, você deve criar um formulário de contato com os seguintes campos: Nome, Sobrenome, Email, Data de Nascimento, Linguagem de Programação Favorita (uma lista suspensa com opções como HTML, CSS, JavaScript, etc.) e uma área de texto para Mensagem.

3º passo: Utilize as tags <form>, <input>, <label>, <select>, <option> e <textarea> para criar o formulário e seus campos.

4º passo: Adicione semântica ao formulário utilizando a tag <fieldset> para agrupar os campos do formulário.

5º passo: Adicione um botão de envio ao formulário utilizando a tag <input> com o tipo “submit”.

6º passo: Salve seu trabalho e visualize a página no navegador.

Gabarito Atividade Prática

Aqui está um exemplo de como a página HTML pode ser estruturada:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

<title>Formulário de Contato</title>



</head>

<body>

<h1>Formulário de Contato</h1>

<form>

<fieldset>

<legend>Informações Pessoais</legend>

<label for="nome">Nome:</label>

<input type="text" id="nome" name="nome">

<label for="sobrenome">Sobrenome:</label>

<input type="text" id="sobrenome" name="sobrenome">

<label for="email">Email:</label>

<input type="email" id="email" name="email">

<label for="dataNascimento">Data de Nascimento:</label>

<input type="date" id="dataNascimento" name="dataNascimento">



<label for="linguagem">Linguagem de Programação Favorita:</label>

<select id="linguagem" name="linguagem">

<option value="html">HTML</option>

<option value="css">CSS</option>

<option value="javascript">JavaScript</option>

<!-- Adicione mais opções conforme necessário -->

</select>

<label for="mensagem">Mensagem:</label>

<textarea id="mensagem" name="mensagem"></textarea>

<input type="submit" value="Enviar">

</fieldset>

</form>

</body>

</html>



Lembre-se de substituir “nome”, “sobrenome”, “email”, “dataNascimento”, “linguagem” e “mensagem” com os nomes apropriados para seus campos de formulário.

[Ir para exercício](#)