

HTML

Formulários

Prof^a Gláucya Boechat
(glaucya.boechat@ufba.br)

Formulários - O que são?



- Os formulários são as famosas caixas de texto e botões que podemos encontrar em muitas páginas web.
- São muito utilizados para realizar buscas ou também para introduzir dados pessoais, por exemplo, em sites de comércio eletrônico.
- Os dados que o usuário introduz nestes campos são enviados ao correio eletrônico do administrador do formulário ou também, para um programa que se encarrega de processá-lo automaticamente.
- Os formulários HTML são estruturas que permitem que usuários submetam dados a uma página. Esses dados podem ser tratados e/ou armazenados dependendo da aplicação.

Tag <FORM>



- Os formulários são definidos por meio das tags **<FORM>** e **</FORM>**.
- Entre estas duas etiquetas colocaremos todos os campos e botões que compõem o formulário.
- Exemplo

<FORM **action=** “<https://computacao.ufba.br/>” **method=** “post” **enctype =** “text/plain”>

FORM - Atributo ACTION



<FORM **action= “https://computacao.ufba.br/”** **method= “post”** **enctype = “text/plain”**>

- **ACTION:** Define o tipo de ação a realizar com o formulário. Existem duas possibilidades:
 - O formulário é enviado a um programa ou script que processa seu conteúdo.
 - Temos de especificar o endereço do arquivo que contém o tal programa.

action= https://computacao.ufba.br/

- O formulário é enviado a um endereço de correio eletrônico. Exemplo:

action= mailto:glaucya.boechat@ufba.br

FORM - Atributo METHOD



5

- **METHOD:** Especifica a forma como o formulário será enviado.
- Os dados de formulários HTML podem ser enviados através de pelo menos dois deles: o método GET e o método POST.
- HTTP - Protocolo para comunicação entre cliente e servidor

<FORM action= “https://computacao.ufba.br/” method= “post” enctype = “text/plain”>

Enviando dados via GET



- A atributo method define como os dados são enviados
 - Os dados de formulários HTML podem ser enviados através de pelo menos dois deles: o método GET e o método POST.

```
<form action="http://dados.com" method="get">  
  <input name="nome" value="Glaucya">  
  <input name="sobrenome" value="Boechat">  
  <input type="submit">  
</form>
```

A solicitação HTTP tem esta aparência:

```
GET /?nome=Glaucya&sobrenome=Boechat HTTP/1.1  
Host: dados.com
```

Enviando dados via POST



- É o método que o navegador envia ao servidor para pedir uma resposta que leva em conta os dados fornecidos no corpo da solicitação HTTP: "Hey servidor, dê uma olhada nesses dados e envie-me de volta um resultado apropriado". Se um formulário é enviado usando esse método, os dados são anexados ao corpo do pedido HTTP.

```
<form action="http://dados.com" method="post">  
  <input name="nome" value="Glaucya">  
  <input name="sobrenome" value="Boechat">  
  <input type="submit">  
</form>
```

A solicitação HTTP tem esta aparência:

```
POST / HTTP/1.1  
Host: dados.com  
Content-Type:  
application/x-www-form-urlencoded  
Content-Length: 13
```

nome=Glaucya&sobrenome=Boechat

POST X GET

8

Característica	GET	POST
Visibilidade	Dados visíveis ao usuário	Dados “ocultos”
Segurança	Menos seguro	Mais seguro
Restrição de tamanho	Tamanho da URL (~2048)	Sem restrição
Restrição de tipo de dados	Apenas ASCII	Sem restrição
Botão voltar	Ok	Dados serão ressubmetidos
Ad. aos favoritos	Ok	Não é possível
Histórico do navegador	Parâmetros são salvos	Parâmetros não são salvos

FORM - Atributo ENCTYPE



<FORM **action= "glaucya.boechat@ufba.br"** **method= "post"** **enctype = "text/plain"**>

- **ENCTYPE:** Utiliza-se para indicar a forma na qual como a informação que é mandada pelo formulário.
 - No caso de enviar o formulário por correio eletrônico, o valor deste atributo deve ser **text/plain**. Assim, conseguimos que o conteúdo do formulário seja enviado como texto plano dentro do e-mail.
 - Se quisermos que o formulário se processe automaticamente por um programa, geralmente não utilizaremos este atributo, de forma que tome seu valor padrão, ou seja, não incluiremos *enctype* dentro da etiqueta <FORM>.

FORM - Tipos de campos



- `<input>` – Campo de entrada de dados;
- `<textarea>` – Área de texto, podendo conter diversas linhas de texto;
- `<button>` – Botão;
- `<select>` – Lista selecionável, também conhecida como drop-down;
- `<option>` – Lista de opções dentro de um drop-down;
- `<label>` – Rótulo ou legenda para um campo ou controle do formulário;

Campo de Texto - Texto Curto



- As caixas de texto são colocadas por meio da tag **<INPUT>**. Dentro desta tag temos de especificar o valor de dois atributos: **type** e **name**.
- Sintaxe:
<INPUT type=text name=nome>
 - O nome do elemento do formulário é de grande importância para poder identificá-lo em nosso programa de processamento ou no e-mail recebido.

Tag <INPUT> - Atributo SIZE



- Além destes dois atributos, existem outros atributos que podem ser de utilidade, mas que não são imprescindíveis:

`<INPUT type=text name=nome size=20>`

- **SIZE:** Define o tamanho da caixa em número de caracteres.
 - Se ao escrever o usuário chega ao final da caixa, o texto irá desfilando à medida que se escreve, fazendo desaparecer a parte de texto que fica à esquerda.

Nome:

Tag <INPUT> - Atributos MAXLENGTH



- **MAXLENGTH:** Indica o tamanho máximo do texto que pode ser tomado pelo formulário.
 - maxlength indica o tamanho máximo real do texto que pode ser escrito.
 - Podemos ter uma caixa de texto com um tamanho aparente (size) que é menor do que o tamanho máximo (maxlength).

<INPUT type=text name=nome size=20 maxlength=15>

Nome:

Tag <INPUT> - Atributo MINLENGTH



- MINLENGTH: Indica o tamanho mínimo do texto que pode ser tomado pelo formulário.
 - maxlength indica o tamanho mínimo real do texto que pode ser escrito.

<INPUT type=text name=nome size=20 minlength=5>

Nome:

Tag <INPUT> - Atributo VALUE



- **VALUE:** Atribui um valor inicial ao campo.
 - Em alguns casos pode ser interessante atribuir um valor definido ao campo em questão. Isto pode ajudar o usuário a preencher mais rapidamente o formulário ou a dar alguma idéia sobre a natureza de dados que se requerem.
- Exemplo:

<INPUT type=text name=nome size=20 value="Glaucya Boechat">

Nome:

Campo de texto - Texto Oculto



- Podemos esconder o texto escrito por meio de asteriscos de forma a fornecer uma certa confiabilidade.
- Estes tipos de campos são análogos aos de texto com somente uma diferença: o atributo `type=text` passa a ser **`type=password`**:

`<INPUT type=password name=senha size=20 >`

- Neste caso, ao escrever dentro do campo no lugar de texto serão vistos asteriscos. Estes campos são ideais para a introdução de dados confidenciais, principalmente códigos de acesso.

Senha:

Campo de texto - Texto Oculto



- **Exemplo:**

```
<form action="dados.html" method="get" enctype="text/plain">
```

```
  Login: <input name="login" type="text" size=15 >  <br><br>
```






```
  Senha:<input  type=password  name=senha size=15 ><br><br>
```

```
</form>
```

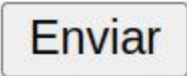



Login:

Senha:







Tipos de <input>

Modelo	Descrição	Exemplos básicos
text	Campo de texto de linha única. As quebras de linha são removidas automaticamente do valor de entrada.	 
password	Um campo de texto de linha única cujo valor está obscurecido. Alertará o usuário se o site não for seguro.	
<u>email</u>	Um campo para editar um endereço de e-mail. Possui parâmetros de validação e teclado relevantes no suporte a navegadores e dispositivos com teclados dinâmicos.	 


Tipos de <input>

Modelo	Descrição	Exemplos básicos
submit	Um botão que envia o formulário.	
reset	Um botão que redefine o conteúdo do formulário para os valores padrão.	
image	Um botão gráfico submit. Exibe uma imagem. O atributo alt é exibido se a imagem estiver ausente.	 image input 


Tipos de <input>


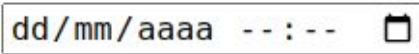
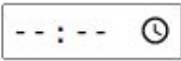
Modelo	Descrição	Exemplos básicos
range	Um controle para inserir um número cujo valor exato não é importante. Exibe como um widget de intervalo padronizado para o valor médio.	 
checkbox	Uma caixa de seleção que permite selecionar/desmarcar valores únicos.	 
radio	Um botão de opção, permitindo que um único valor seja selecionado entre várias opções com o mesmo nomevalor.	 

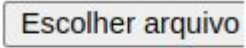





Tipos de <input>

Modelo	Descrição	Exemplos básicos
color	Um controle para especificar uma cor; abrindo um seletor de cores quando ativo em navegadores compatíveis.	
hidden	Um controle que não é exibido, mas cujo valor é enviado ao servidor.	



Tipos de <input>

dd/mm/aaaa 

Modelo	Descrição	Exemplos básicos
date	Um controle para inserir uma data (ano, mês e dia, sem hora). Abre um seletor de data ou rodas numéricas para ano, mês, dia quando ativo em navegadores compatíveis.	
<u>datetime-local</u>	Um controle para inserir uma data e hora, sem fuso horário. Abre um seletor de data ou rodas numéricas para componentes de data e hora quando ativo em navegadores compatíveis.	
time	Um controle para inserir um valor de tempo sem fuso horário.	

Modelo	Descrição	Exemplos básicos
file	Um controle que permite ao usuário selecionar um arquivo ou vários arquivos com o atributo multiple. É possível definir os tipos de arquivos que o controle pode selecionar.	 
url	Um campo para inserir um URL. Parece uma textentrada, mas possui parâmetros de validação e teclado relevante no suporte a navegadores e dispositivos com teclados dinâmicos.	 
search	Um campo de texto de linha única para inserir strings de pesquisa. Pode incluir um ícone de exclusão em navegadores de suporte que podem ser usados para limpar o campo. Exibe um ícone de pesquisa em vez da tecla Enter em alguns dispositivos com teclados dinâmicos.	 

Tipos de <input>

Modelo	Descrição	Exemplos básicos
number	Um controle para inserir um número. Exibe um spinner e adiciona validação padrão. Exibe um teclado numérico em alguns dispositivos com teclados dinâmicos.	
tel	Um controle para inserir um número de telefone. Exibe um teclado de telefone em alguns dispositivos com teclados dinâmicos.	

Campo de Texto - Área de texto



- Campo de texto que é possível escrever sobre um espaço composto de várias linhas, temos de utilizar a tag: **<TEXTAREA>**.

<TEXTAREA name=comentario rows=10 cols=40></TEXTAREA>

- Estes tipos de campos são práticos quando o conteúdo a enviar é um comentário, opinião, etc.
- Podemos definir as dimensões do campo a partir dos atributos **rows** (número de linhas do campo de texto) e **cols** (número de colunas do campo de texto).

Tag <TEXTAREA>

- É possível definir o conteúdo inicial do campo. Para isso, escreveremos dentro da etiqueta o conteúdo que desejamos atribuir.

Vejamos:

```
<TEXTAREA name = comentario rows = 10 cols = 40>
```

Escreva seu comentário

```
</TEXTAREA>
```

Escreva seu comentário

Lista de Opções



- As listas de opções são menus desdobráveis que nos permite escolher uma (ou várias) das múltiplas opções que nos propõem, temos de utilizar a tag: **<SELECT>**.

<SELECT nome=opcoes > ... </SELECT>

- Cada opção será incluída em uma linha precedida da etiqueta **<OPTION>**.

<SELECT name=turno>

<OPTION>Matutino</OPTION>

<OPTION>Vespertino</OPTION>

<OPTION>Noturno</OPTION>

</SELECT>



Lista de Opções

- Exemplo.

```
<SELECT name=turno>  
  <OPTION>Matutino</OPTION>  
  <OPTION>Vespertino</OPTION>  
  <OPTION>Noturno</OPTION>  
</SELECT>
```

Matutino ▼

Lista de Opções



- **SELECTED**: atributo que indica a opção escolhida por padrão.

`<OPTION selected>Outono</OPTION>`

- Exemplo

`<SELECT name=turno>`

`<OPTION>Matutino</OPTION>`

`<OPTION>Vespertino</OPTION>`

`<OPTION selected>Noturno</OPTION>`

`</SELECT>`

Lista de Opções



- **VALUE:** Define o valor da opção que será enviada se o usuário escolher essa opção.
 - Cada opção pode se associar a um número ou letra.
 - No exemplo abaixo, se o usuário escolher vespertino, a variável chamada turno que terá valor 3.

<SELECT name=turno>

<OPTION value=1>Matutino</OPTION>

<OPTION value=2>Vespertino</OPTION>

<OPTION value=3 selected>Noturno</OPTION>

</SELECT>

&turno=2

Textos com Múltiplas Escolhas (CHECKBOX)



```
<form method="POST" action="dados.html">
```

Escolha a(s) área(s) de seu interesse:


```
<input type="checkbox" name="educacao" value=1>educação<br>
```

```
<input type="checkbox" name="esporte" value=2>esporte
```

```
<input type="checkbox" name="saude" value=3>saúde
```

```
</form>
```

Escolha a(s) área(s) de seu interesse:

☐ educação

☐ esporte

☐ saúde

Textos com Múltiplas Escolhas



- **CHECKED**: indica que a opção está escolhida por padrão.
- Exemplo:

Escolha a(s) área(s) de seu interesse:

<input type="checkbox" name="educacao" value=1 checked>educação

<input type="checkbox" name="esporte" value=2>esporte

<input type="checkbox" name="saude" value=3>saúde

Escolha a(s) área(s) de seu interesse:

☒ educação

☐ esporte

☐ saúde

Textos com Escolha Única (RADIO)



Qual seu curso?

<input type="radio" name="curso" value="bcc">Ciência da Computação

<input type="radio" name="curso" value="lc">Licenciatura em Computação

<input type="radio" name="curso" value="bsi" checked>Sistemas de Informação

Qual seu curso?

- ☐ Ciência da Computação
- ☐ Licenciatura em Computação
- ☐ Sistemas de Informação

Textos com Escolha Única (RADIO)



- **CHECKED:** indica que a opção está escolhida por padrão.

Qual seu curso?

<input type="radio" name="curso" value="bcc">Ciência da Computação

<input type="radio" name="curso" value="lc">Licenciatura em Computação

<input type="radio" name="curso" value="bsi" checked>Sistemas de Informação

Qual seu curso?

☐ Ciência da Computação

☐ Licenciatura em Computação

☒ Sistemas de Informação

Botões de Ação Enviar (Submit) e Ação Redefinir (Reset)



- **SUBMIT:** envia os dados do formulário ao servidor
- **RESET:** redefine todas as entradas no formulário para seus valores iniciais..

```
<input type="submit" name="enviar">
```

```
<input type="reset" name="apagar">
```

Exemplo - O incrível monstro

<https://terrivel.herokuapp.com/monster>

```
<form action="https://terrivel.herokuapp.com/monster" method="GET">
```

```
Nome do monstro : <input name="nome" type="text"> <br>
```

```
Qtd de pedras : <input name="num_pedras" type="number" step="1" min="0"><br>
```

```
Cor : <input name="corCeu1" id="corCeu1" type="color"> <br>
```

```
Tipo de pedra: <select name="tipo_pedras">
```

```
    <option value="marroada">Marroada</option>
```

```
    <option value="ametista">Ametista</option>
```

```
</select><br>
```

```
Tipo de pedras sortidas
```

```
<input name="tipo_pedras_sortidas" type="radio" value="não" checked>não
```

```
<input name="tipo_pedras_sortidas" type="radio" value="sim">sim
```

```
<br><br>
```

```
<button type="submit">alimentar monstro</button>
```

```
<button type="submit" formmethod="POST">
```

```
    Alimentar monstro (Post) </button>
```

```
</form>
```

Nome do monstro :

Qtd de pedras :

Cor :

Tipo de pedra: ▼

Tipo de pedras sortidas ☐ não ☒ sim

Exemplo - O incrível monstro

<https://terrivel.herokuapp.com/monster>

Envio via POST

Nome do monstro :

Qtd de pedras :

Cor :

Tipo de pedra:

Tipo de pedras sortidas ☐ não ☒ sim

POST /monster HTTP/1.1
Host: terrivel.herokuapp.com

terrivel.herokuapp.com? nome=blabla&num_pedras=2&corCeul=%23a24b11&tipo_pedras=ametista&tipo_pedras_sortidas=sim



Exemplo - O incrível monstro

<https://terrivel.herokuapp.com/monster>

Envio via GET

https://terrivel.herokuapp.com/monster?nome=blabla&num_pedras=2&corCeu1=%23a24b11&tipo_pedras=ametista&tipo_pedras_sortidas=sim

Nome do monstro :

Qtd de pedras :

Cor :

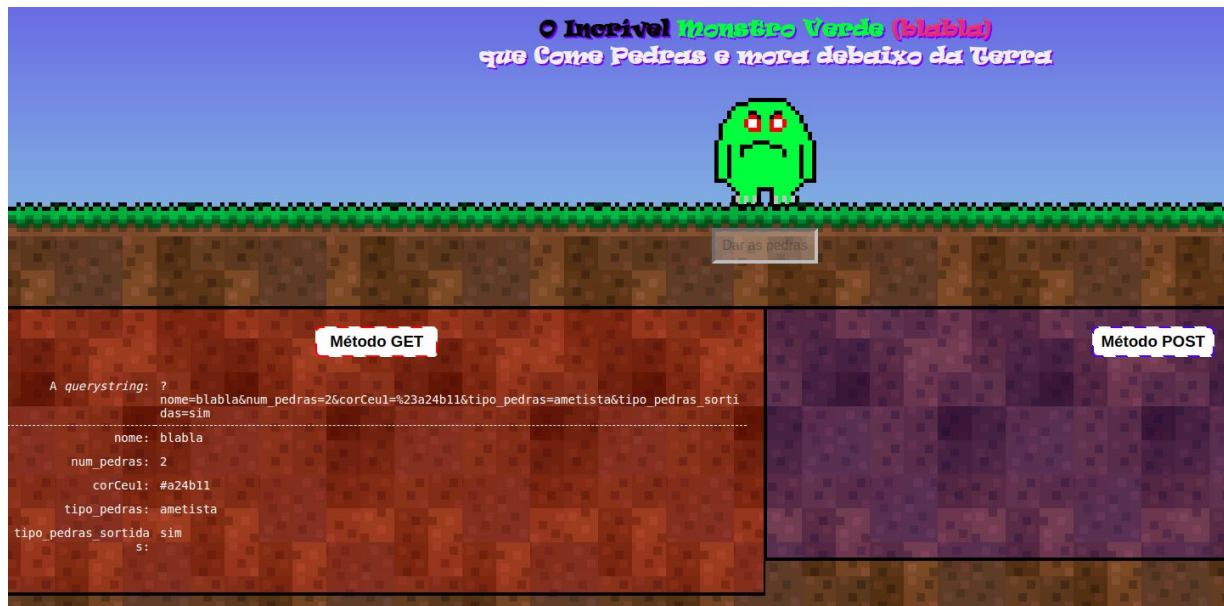
Tipo de pedra:

Tipo de pedras sortidas ☐ não ☒ sim

GET /monster HTTP/1.1

Host:

terrivel.herokuapp.com? **nome=blabla**
a&num_pedras=2&corCeu1=%23a24b11&
tipo_pedras=ametista&tipo_pedras_
sortidas=sim



Referências



1. W3Schools (<http://www.w3schools.com/>)
2. HTML.net <http://html.net/>
3. HTML: HyperTextMarkup Language. | MDN Mozilla
<https://developer.mozilla.org/en-US/docs/Web/HTML>
4. https://developer.mozilla.org/pt-BR/docs/Learn/Forms/Sending_and_retrieving_form_data
5. Flávio Coutinho. Envio de formulários. Recebendo info no servidor.
<https://fegemo.github.io/cefet-web/classes/ssn6/#envio-de-formularios>
6. Guia do desenvolvedor web (<https://developer.mozilla.org/pt-BR/docs/Web/Guide>)
7. Notas de aula da Prof^ª. Adriana Carniello. Desenvolvimento Web I. Instituto Federal de São Paulo - Campus Guarulhos.