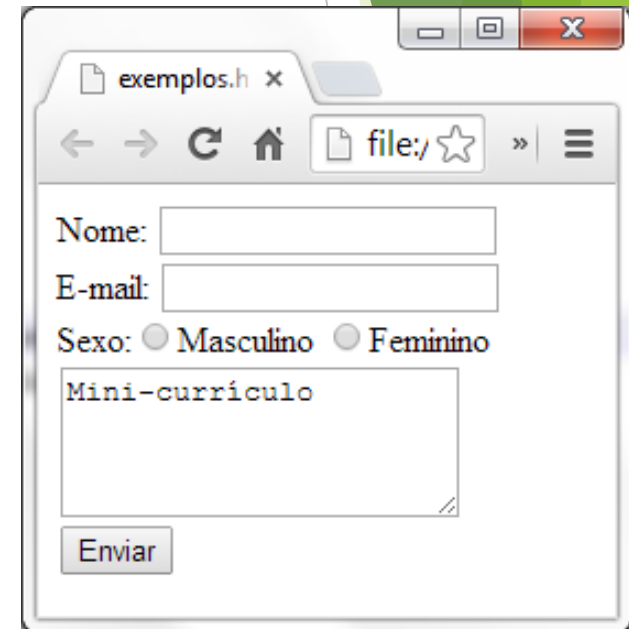


Formulários

Relembrando os conceitos

- ▶ Formulários são seções específicas de um documento projetados para
 - ▶ Coletar dados fornecidos pelo usuário
 - ▶ Enviá-los para serem processados por um *script*
- ▶ Os objetos que compõem um formulário são chamados de *controles*
- ▶ São exemplos de controles de um formulário
 - ▶ Botões
 - ▶ Entradas de texto
 - ▶ Checkboxes
 - ▶ Seleção de arquivos
 - ▶ Radio buttons
 - ▶ Controle oculto
 - ▶ Menus



A screenshot of a web browser window displaying a form. The browser's address bar shows 'file:/' and the tab is labeled 'exemplos.h'. The form contains the following elements: a text input field for 'Nome:', another for 'E-mail:', a 'Sexo:' label with two radio buttons labeled 'Masculino' and 'Feminino', a large text area labeled 'Mini-curriculo', and a button labeled 'Enviar'.

Formulários

Elementos de formulários - `form`

- ▶ O elemento `form` é o container para os controles do formulário
- ▶ Ele contém o *layout* composto pelos conteúdos do formulário
 - ▶ Pode conter textos e marcação HTML, além de seus controles
- ▶ Atributos principais:
 - ▶ `action` → define o endereço do programa (script) que irá processar o formulário após a sua submissão
 - ▶ `method` → define o método de envio dos dados (POST ou GET)

```
<form action="processa_contato.php" method="post">
..controles do formulário...
</form>
```

Formulários

Elementos de formulários - `input`

- O elemento `input` cria vários tipos de controle que são definidos pelo atributo `type`

Valor do atributo <code>type</code>	Controle criado
<code>hidden</code>	Envio oculto de dados
<code>text</code>	Para entrada de uma linha simples de texto
<code>password</code>	Semelhante ao anterior, mas os caracteres digitados não são legíveis (por exemplo, são substituídos por uma série de asteriscos)
<code>radio</code>	Radio button
<code>checkbox</code>	Checkbox
<code>file</code>	Seleção de arquivo para envio
<code>submit</code>	Botão de envio do formulário
<code>reset</code>	Botão para limpar os dados entrados pelo usuário
<code>button</code>	Botão associado a um tipo de processamento específico criado pelo autor
<code>image</code>	Botão gráfico de envio do formulário

Formulários

Elementos de formulários - input (continuação)

```
<form action="">
<input type="hidden" />

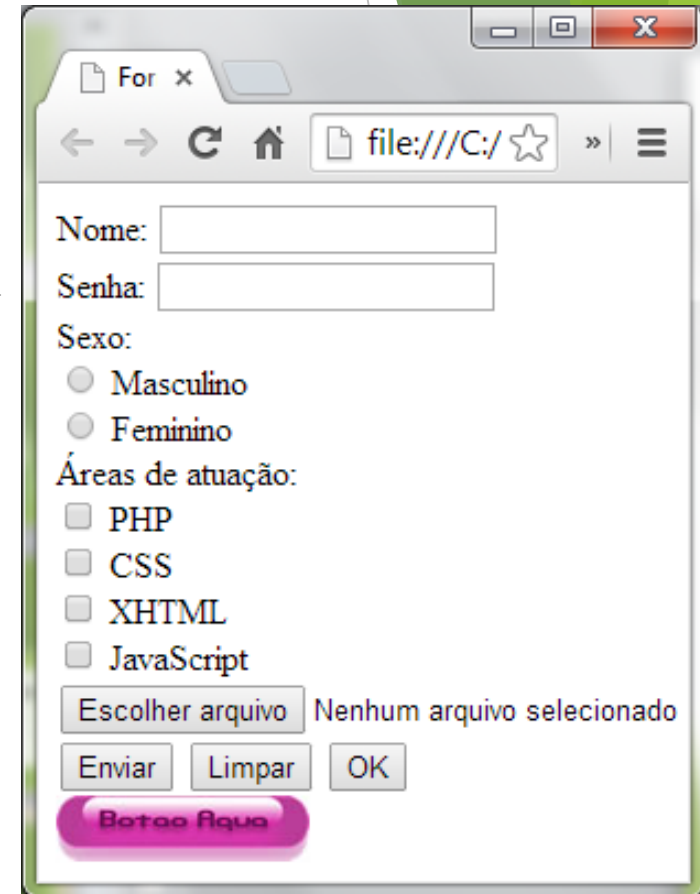
Nome: <input type="text" /><br />
Senha: <input type="password" /><br />

Sexo:<br />
<input type="radio" name="sexo" /> Masculino<br />
<input type="radio" name="sexo" /> Feminino<br />

Áreas de atuação:<br />
<input type="checkbox" /> PHP<br />
<input type="checkbox" /> CSS<br />
<input type="checkbox" /> XHTML<br />
<input type="checkbox" /> JavaScript<br />

<input type="file" /><br />

<input type="submit" value="Enviar" />
<input type="reset" value="Limpar" />
<input type="button" value="OK" /><br />
<input type="image" src="botao-aqua.jpg" />
</form>
```



Formulários

Elementos de formulários - `button`

- ▶ O elemento `button` cria controles do tipo botão de maneira semelhante ao elemento `input`
- ▶ A diferença é que o elemento `input` é vazio
 - ▶ Podemos inserir conteúdo usando o elemento `button`
- ▶ O tipo de botão criado é definido pelo atributo `type`:

Tipo	Finalidade
<code>submit</code>	Botão de envio do formulário
<code>reset</code>	Botão para limpar os dados entrados pelo usuário
<code>button</code>	Botão associado a scripts programados pelo autor

Formulários

Elementos de formulários - button (continuação)

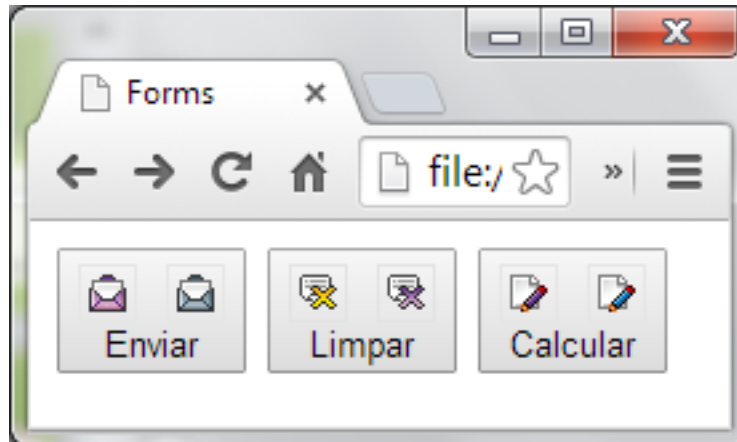
```
<form action="">
```

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

```
<button type="reset">  
    <br />Limpar  
</button>
```

```
<button type="button">  
    <br />Calcular  
</button>
```

```
</form>
```



Formulários

Elementos de formulários - `select`, `option` e `optgroup`

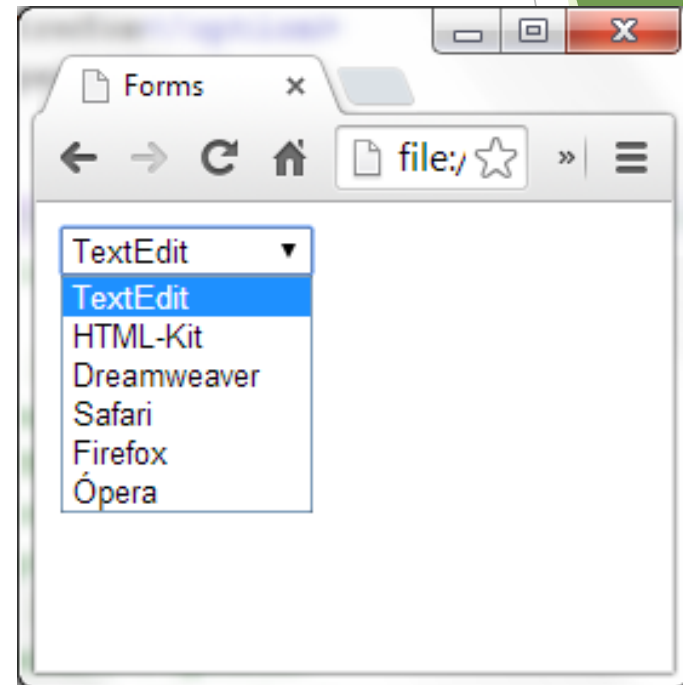
- ▶ O elemento `select` destina-se a criar um menu de escolhas em um formulário
- ▶ Cada item do menu é marcado com o elemento `option`
- ▶ Para agrupar itens do menu que tenham alguma relação entre si, usamos o elemento `optgroup`
 - ▶ Útil para fazer escolhas em uma lista muito extensa de opções
 - ▶ Agrupar opções de natureza idêntica
 - ▶ Facilita a localização das opções

Formulários

Elementos de formulários - select, option e optgroup (continuação)

► Sem usar optgroup

```
<form action="">
<select>
<option>TextEdit</option>
<option>HTML-Kit</option>
<option>Dreamweaver</option>
<option>Safari</option>
<option>Firefox</option>
<option>Ópera</option>
</select>
</form>
```

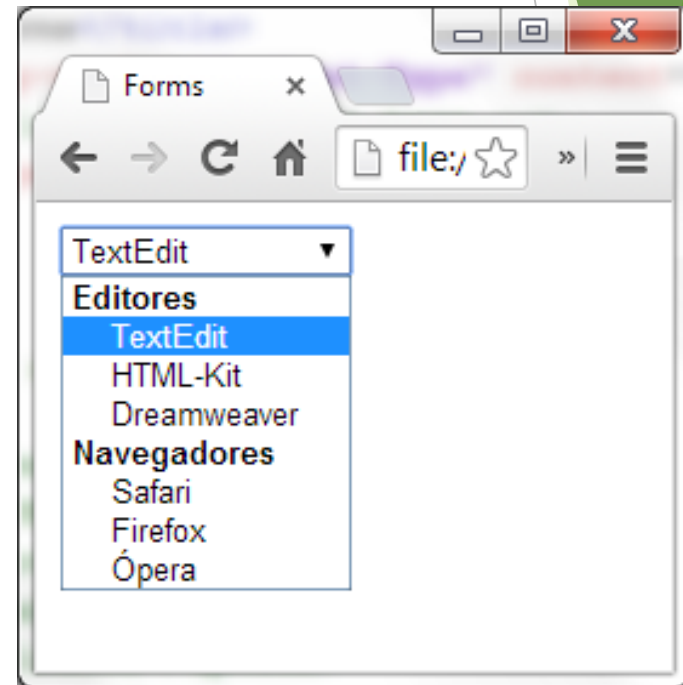


Formulários

Elementos de formulários - select, option e optgroup (continuação)

► Usando optgroup

```
<form action="">
<select>
<optgroup label="Editores">
<option>TextEdit</option>
<option>HTML-Kit</option>
<option>Dreamweaver</option>
</optgroup>
<optgroup label="Navegadores">
<option>Safari</option>
<option>Firefox</option>
<option>Ópera</option>
</optgroup>
</select>
```

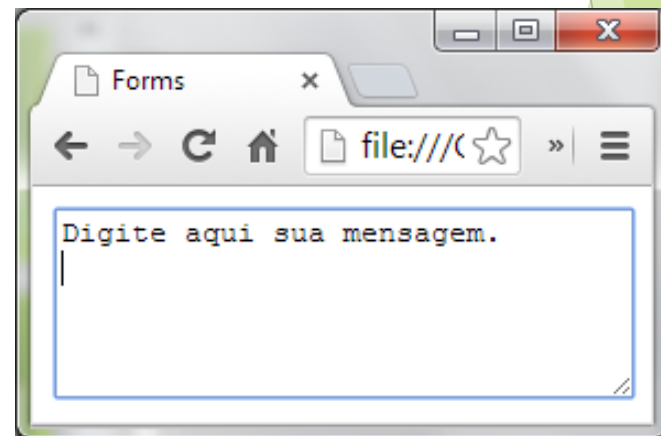


Formulários

Elementos de formulários - textarea

- ▶ O elemento `textarea` destina-se a criar um controle do tipo texto permitindo entradas do usuário em múltiplas linhas
- ▶ Atributos principais:
 - ▶ `rows` → define a quantidade de linhas do `textarea` (altura)
 - ▶ `cols` → define a quantidade de colunas do `textarea` (largura)

```
<form action="">  
<textarea rows="5" cols="30">  
Digite aqui sua mensagem.  
</textarea>  
</form>
```



Formulários

Elementos de formulários - `label`

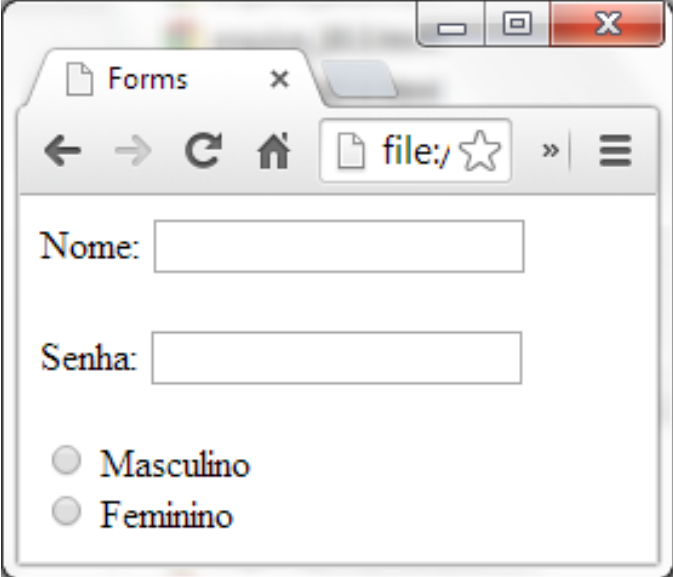
- ▶ A maioria dos controles de formulário não possuem rótulos, como campos de texto, *checkboxes*, *radio buttons* e menus.
- ▶ O elemento `label` é utilizado para especificar rótulos para controles
 - ▶ É um item importante para incrementar a acessibilidade (por exemplo, leitores de tela)
- ▶ Cada elemento `label` associa um rótulo a um (e somente um) controle no formulário, com a finalidade de agregar informação ao controle
- ▶ O atributo `for` é usado para associar um rótulo com o seu controle
 - ▶ O valor do atributo `for` deve ser o mesmo valor do atributo `id` do controle a ele associado

Formulários

Elementos de formulários - label (continuação)

► Associação explícita do elemento label

```
<form action="">
<label for="nome">Nome:</label>
<input type="text" id="nome" />
<br /><br />
<label for="senha">Senha:</label>
<input type="password" id="senha" />
<br /><br />
<input type="radio" id="masc" />
<label for="masc">Masculino</label>
<br />
<input type="radio" id="fem" />
<label for="fem">Feminino</label>
</form>
```



The screenshot shows a web browser window with a single tab titled 'Forms'. The address bar shows a local file path. The rendered form contains the following elements:

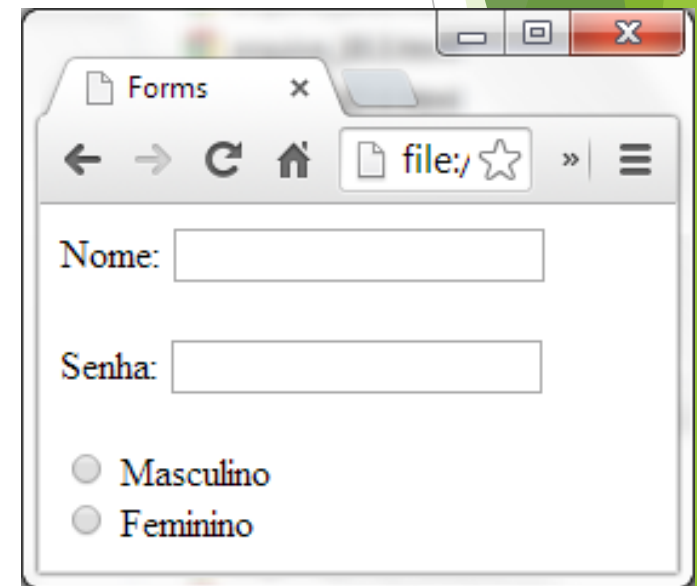
- A text input field preceded by the label 'Nome:'.
- A password input field preceded by the label 'Senha:'.
- Two radio buttons below the password field, labeled 'Masculino' and 'Feminino'.

Formulários

Elementos de formulários - label (continuação)

► Associação implícita do elemento label

```
<form action="">
<label>
    Nome:<input type="text" />
</label>
<br /><br />
<label>
    Senha:<input type="password" />
</label>
<br /><br />
<label>
    <input type="radio" />Masculino
</label>
<br />
<label>
    <input type="radio" />Feminino
</label>
</form>
```



Formulários

Elementos de formulários - `fieldset` e `legend`

- ▶ Têm a função de estruturar formulários
- ▶ O elemento `fieldset` destina-se a agrupar um conjunto de controles do formulário que tenham finalidades relacionadas
 - ▶ o conjunto de campos de texto destinados a coletar dados residenciais
 - ▶ o conjunto de *checkboxes* destinado a coletar preferências do usuário
- ▶ A cada `fieldset` pode ser associado um elemento `legend`, que se destina a identificar com um rótulo (ou título) o respectivo `fieldset`
 - ▶ O elemento `legend` é importante para incrementar a acessibilidade

Formulários

Elementos de formulários - fieldset e legend (continuação)

```
<form action="">
<fieldset>
<legend>Informações pessoais</legend><br />
<label>Nome: <input type="text" /></label><br />
<br />
<label>Endereço: <input type="text" />
</label><br /><br />
...outros controles coletando dados pessoais...
</fieldset>

<fieldset>
<legend>Histórico médico</legend><br />
<label><input type="checkbox" />
Alérgico</label><br />
<label><input type="checkbox" />
Tonteiras</label><br />
<label><input type="checkbox" />
Vacinação</label><br />
...outros controles para histórico médico...
</fieldset>
</form>
```

The screenshot shows a web browser window with a file:// URL. It displays a form with two distinct sections, each enclosed in a box with a title bar. The first section is titled 'Informações pessoais' and contains two text input fields labeled 'Nome:' and 'Endereço:'. Below these is a legend box labeled 'fieldset'. The second section is titled 'Histórico médico' and contains three checkboxes labeled 'Alérgico', 'Tonteiras', and 'Vacinação'. Below these is another legend box labeled 'fieldset'. Arrows from the code blocks point to these legend boxes.

Formulários

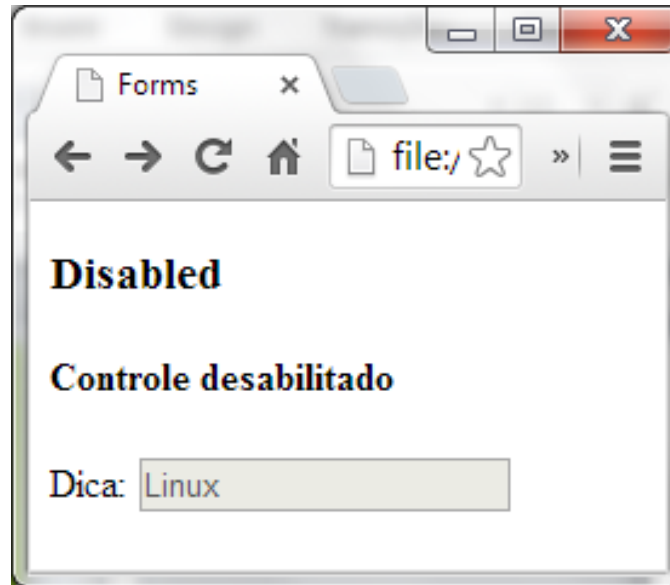
Atributos - `disabled` e `readonly`

- ▶ Pode ser necessário desabilitar a entrada de dados em um elemento ou colocá-lo como somente leitura
- ▶ O atributo `disabled` desabilita o controle para ações do usuário e produz os seguintes efeitos em um elemento:
 - ▶ Desabilita o controle para receber foco
 - ▶ Desabilita o controle na navegação por tabulação
 - ▶ Não envia o dado contido no controle para processamento quando o formulário for enviado
- ▶ Os seguintes elementos suportam o atributo `disabled`:
 - ▶ `button`
 - ▶ `input`
 - ▶ `optgroup`
 - ▶ `option`
 - ▶ `select`
 - ▶ `textarea`

Formulários

Atributos - disabled e readonly (cont.)

```
<form action="">
<label>
    Dica: <input disabled="disabled" value="Linux">
</label>
</form>
```



Formulários

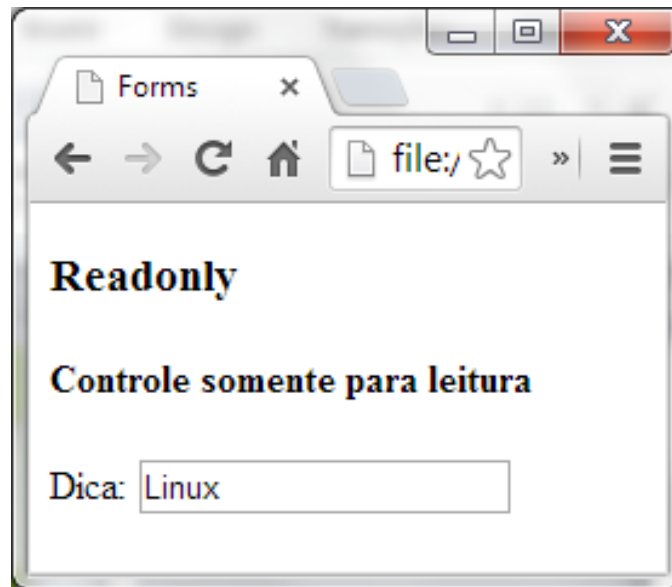
Atributos - `disabled` e `readonly`

- ▶ O atributo `readonly` põe o controle no modo somente para leitura e produz os seguintes efeitos em um elemento:
 - ▶ Recebe o foco, mas não pode ser modificado pelo usuário
 - ▶ São incluídos normalmente na navegação por tabulação
 - ▶ Pode enviar o dado contido no controle para processamento quando o formulário for enviado
- ▶ Os seguintes elementos suportam o atributo `readonly`:
 - ▶ `input`
 - ▶ `textarea`

Formulários

Atributos - disabled e readonly (cont.)

```
<form action="">
<label>
    Dica: <input readonly="readonly" value="Linux">
</label>
</form>
```



Formulários

Métodos de envio: POST e GET

- ▶ Os formulários podem ser submetidos para processamento por meio dos métodos `POST` ou `GET`

- ▶ Método `GET`

- ▶ os dados do formulário são anexados à URL (URI) do campo `action`, com um caractere '?' como separador
 - ▶ a URL resultante é enviada à página ou *script* que fará o processamento do formulário
 - ▶ os dados ficam armazenados no *array* associativo `$_GET`

- ▶ Método `POST`

- ▶ encapsula os dados do formulário dentro da mensagem que será enviada pelo protocolo HTTP à página ou *script* que fará o processamento do formulário
 - ▶ os dados ficam armazenados no *array* associativo `$_POST`

Os nomes das chaves do *array* equivalem aos nomes dos campos do formulário.

Formulários

Qual método usar?

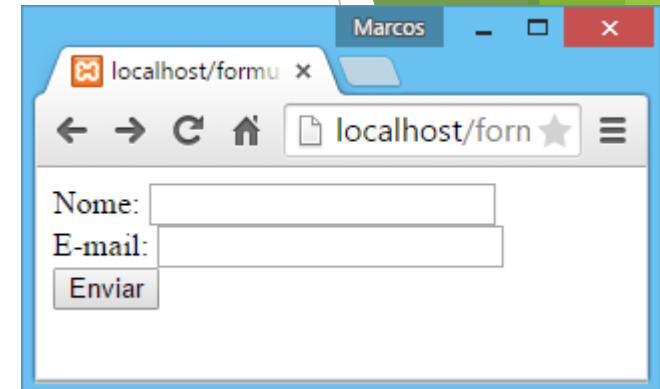
- ▶ O programador deve escolher o método mais adequado para cada caso, observando o seguinte:
 - ▶ O método `GET` pode ser utilizado sempre que o formulário possa ser enviado diversas vezes sem alterar o resultado obtido, ou seja, não causando efeitos colaterais
 - ▶ Exemplo: um campo de busca
 - ▶ O método `POST` deve ser utilizado quando enviamos uma quantidade maior de dados ou quando a submissão do formulário puder causar efeitos colaterais
 - ▶ Exemplo: gravação dos dados em um banco de dados
- ▶ Além disso, o método `POST` fornece uma certa segurança adicional, uma vez que os dados enviados não ficam visíveis na URL
 - ▶ Isso pode impedir o envio de dados inconsistentes por um usuário mal-intencionado

Formulários

Submetendo formulário via GET

formulario.php

```
1 <!DOCTYPE html>
2 <html>
3   <head></head>
4   <body>
5     <form action="processa.php" method="GET">
6       Nome: <input type="text" name="nome" />
7       <br />
8       E-mail: <input type="text" name="email" />
9       <br />
10      <input type="submit" value="Enviar" />
11    </form>
12  </body>
13 </html>
```

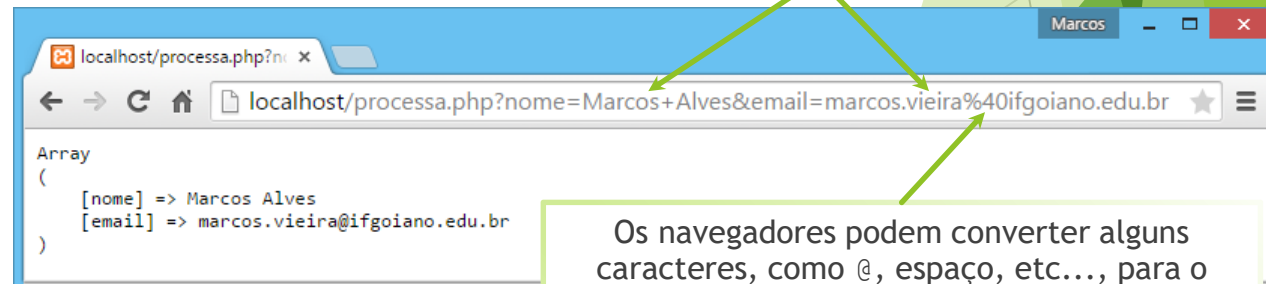


A screenshot of a web browser window titled 'Marcos'. The address bar shows 'localhost/formu'. The page displays a simple form with two text input fields labeled 'Nome:' and 'E-mail:', and a button labeled 'Enviar'.

Note que os dados do formulário são passados na URL:
? -> representa o início da cadeia de variáveis
& -> identifica o início de uma nova variável
= -> separa as variáveis dos seus respectivos valores

processa.php

```
1 <?php
2   echo "<pre>";
3   print_r($_GET);
4   echo "</pre>";
5   ?>
```



A screenshot of a web browser window titled 'Marcos'. The address bar shows 'localhost/processa.php?nome=Marcos+Alves&email=marcos.vieira%40ifgoiano.edu.br'. The page displays the output of the PHP script, showing the submitted data in a preformatted view:

```
Array
(
    [nome] => Marcos Alves
    [email] => marcos.vieira@ifgoiano.edu.br
)
```

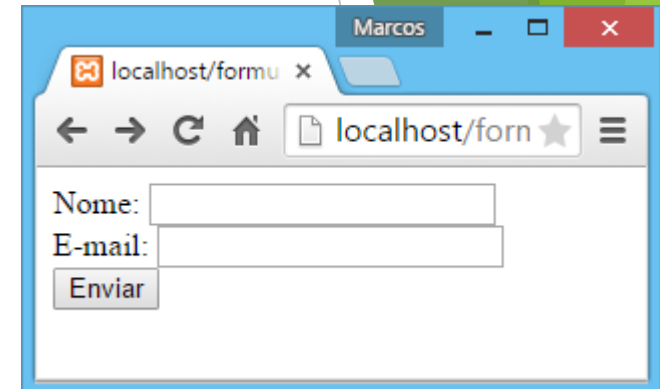
Os navegadores podem converter alguns caracteres, como @, espaço, etc..., para o valor hexadecimal de sua entidade HTML.
([mais informações](#))

Formulários

Submetendo formulário via POST

formulario.php

```
1 <!DOCTYPE html>
2 <html>
3   <head></head>
4   <body>
5     <form action="processa.php" method="POST">
6       Nome: <input type="text" name="nome" />
7       <br />
8       E-mail: <input type="text" name="email" />
9       <br />
10      <input type="submit" value="Enviar" />
11    </form>
12  </body>
13 </html>
```



Nome:

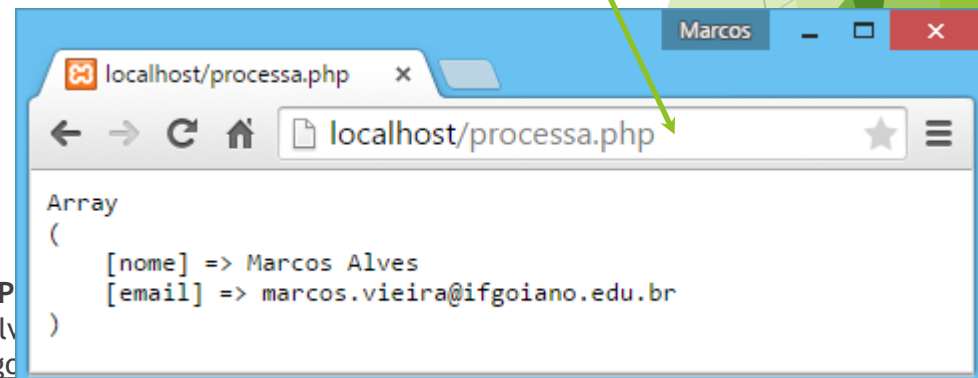
E-mail:

Enviar

Note que a URL não contém os dados do formulário.

processa.php

```
1 <?php
2   echo "<pre>";
3   print_r($_POST);
4   echo "</pre>";
5 <?>
```



localhost/processa.php

```
Array
(
    [nome] => Marcos Alves
    [email] => marcos.vieira@ifgoiano.edu.br
)
```

Exercício 31

31) Crie o formulário abaixo, salve-o como `meu_formulario.php` e aponte seu `action` para a página `exibe_dados.php`. Em seguida, crie a página `exibe_dados.php` e utilize-a para exibir os dados preenchidos no formulário.

Obs.: Crie uma versão de ambos arquivos com submissão de dados via `POST` e outra via `GET`.

`meu_formulario_post.php`

The screenshot shows a web browser window with the title 'Cadastro de Clientes'. The address bar shows 'localhost/meu_formulario_post.php'. The form is divided into three sections: 'Dados Pessoais', 'Endereço', and 'Observações'. The 'Dados Pessoais' section has fields for 'Nome', 'E-mail', 'Sexo' (with radio buttons for 'Masculino' and 'Feminino'), and 'Estado civil' (a dropdown menu). The 'Endereço' section has fields for 'Rua', 'Bairro', 'Cidade', 'Estado', and 'CEP'. The 'Observações' section is a large text area. At the bottom of the form are two buttons: 'Salvar' and 'Limpar'. A green arrow points from the 'Estado civil' dropdown menu to a list of options in a separate box.

Dados Pessoais

Nome:

E-mail:

Sexo: ☐ Masculino ☐ Feminino

Estado civil:

Endereço

Rua:

Bairro:

Cidade: Estado:

CEP:

Observações

Opções:

- Selecione...
- Casado
- Solteiro
- Divorciado
- Viúvo

`exibe_dados_post.php`

The screenshot shows a web browser window with the title 'localhost/exibe_dados_po'. The address bar shows 'localhost/exibe_dados_post.php'. The page displays the data submitted in the form, organized into three sections: 'Dados Pessoais', 'Endereço', and 'Observações'. The data is as follows: Name: Marcos Alves Vieira, E-mail: marcos.vieira@ifgoiano.edu.br, Sex: masculino, Estado civil: casado, Rua: Av. Pará, nº tal, Bairro: Centro, Cidade: Iporá, Estado: GO, CEP: 76200-000. The Observações section shows 'Sem observações no momento.'.

Dados Pessoais

Nome: Marcos Alves Vieira

E-mail: marcos.vieira@ifgoiano.edu.br

Sexo: masculino

Estado civil: casado

Endereço

Rua: Av. Pará, nº tal

Bairro: Centro

Cidade: Iporá Estado: GO

CEP: 76200-000

Observações

Sem observações no momento.

Exercício 32

32) Modifique o arquivo `exibe_dados.php` (GET ou POST) criado no exercício anterior para validar os dados enviados pelo formulário antes de exibi-los na tela, conforme as seguintes regras:

- a) Todos os campos são obrigatórios, com exceção do campo "Observações"
- b) O campo "E-mail" deve conter um endereço de e-mail válido
- c) O campo "Estado civil" deve conter uma opção diferente de "Selecione..."
- d) O campo "Estado" deve conter duas letras e estas devem representar um estado brasileiro. Ex: GO, MG...
- e) O campo "CEP" deve conter um CEP válido, no formato xxxxx-xxx, onde x é um número de 0 a 9

IMPORTANTE:

Uma mensagem deve ser mostrada para o usuário para cada campo que não passar na validação. Apresente também uma opção para voltar ao formulário e editar os dados (*link*, botão, etc...). Exemplos:

```
<a href="javascript:history.back()">Voltar</a>  
<button onclick="history.back()">Voltar</button>
```

