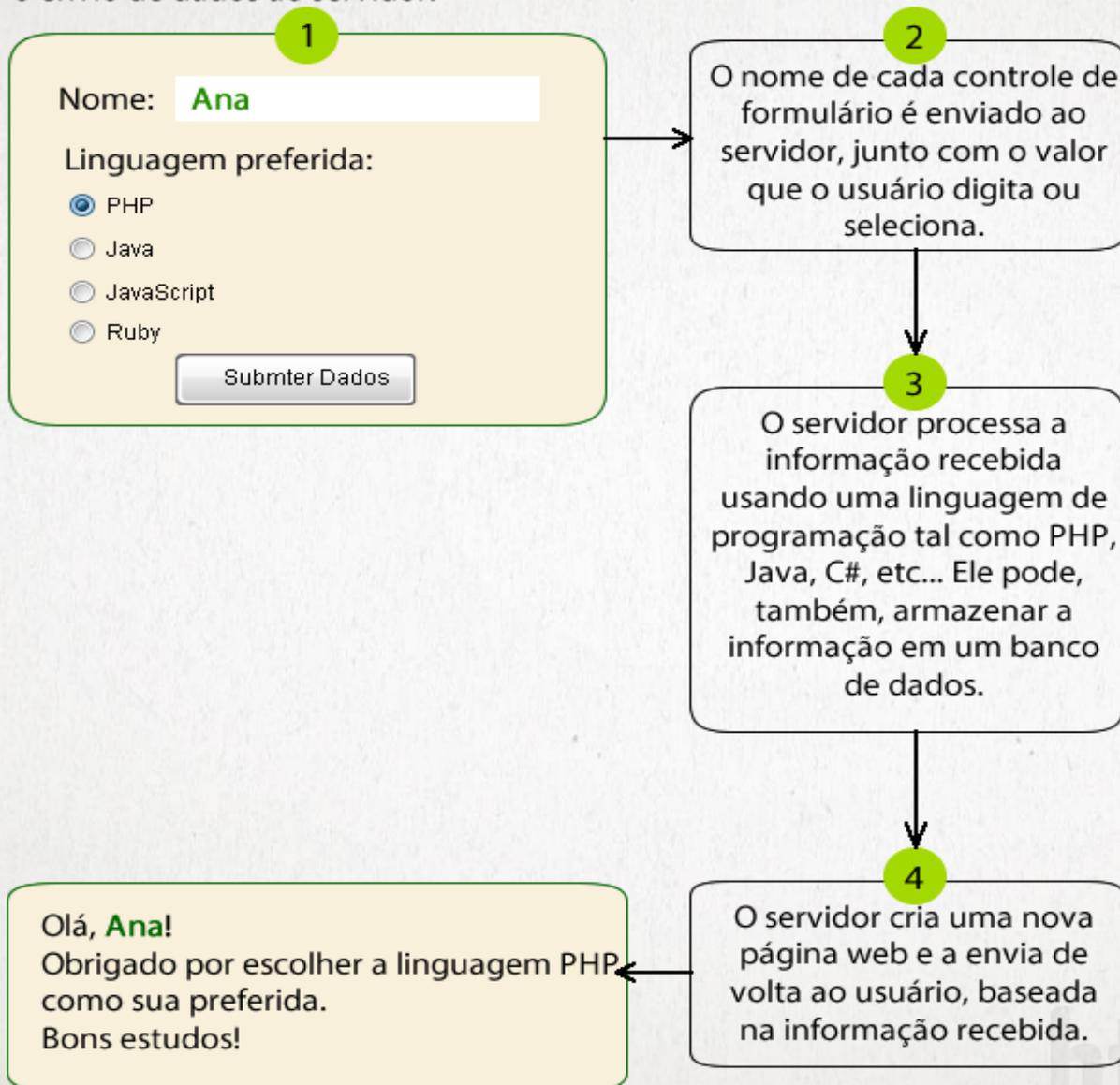


- Constitui-se numa das formas de a máquina cliente enviar informações para que um programa no servidor receba estes dados, processe-os e devolva uma resposta ao usuário, por meio do navegador, gerando uma nova página web. Neste curso, utilizaremos a linguagem PHP para tratamento dos dados no servidor;
- Todo e qualquer elemento de formulário, também chamado de controle de formulário, é inserido dentro da tag **<form>**.

COMO FUNCIONA UM FORMULÁRIO

Um usuário preenche um formulário e, então, pressiona um botão para o envio de dados ao servidor.



A TAG <form>

- É uma tag completa, exigindo abertura e fechamento;
- Possui diversos atributos. Os mais importantes são:
 - **action**
 - **method**
- O atributo **action** indica ao navegador o nome do programa ou script que, no servidor, irá receber os dados e processar a resposta;
- O atributo **method** define a forma com que o navegador irá transportar as informações até o servidor. Este formato pode ser get ou post. Existem diversas diferenças entre um método e outro (<http://marceloweb.info/principais-diferencas-entre-os-metodos-http-get-e-post/>).

```
<form action="recebeDados.php"  
method="post"> </form>
```

PRINCIPAIS CONTROLES DE FORMULÁRIO

- Rótulos;
- Caixas de texto (única linha);
- Caixas de senha;
- Caixas de texto (múltiplas linhas);
- Caixas de seleção;
- Botões de rádio;
- Botões de verificação;
- Botões de envio de dados.

ATRIBUTOS

- Os controles de formulário possuem diversos atributos para as mais variadas funções. Porém, um dos mais importantes, sob o aspecto do desenvolvimento para a web, é o atributo **name**;
- Este é compartilhado por vários tipos de elementos do formulário, que necessitam enviar dados ao servidor;
- Sempre que um elemento de formulário armazena uma informação importante, que necessita ser enviada ao servidor e tratada por alguma linguagem de programação, este elemento deve apresentar o atributo **name**;
- É por meio deste atributo que a aplicação web no servidor recebe a informação do formulário e associa esta informação ao elemento correspondente.

O ATRIBUTO name

- Regra geral, cada controle de formulário que produz uma informação relevante deve ter um atributo name diferente, de forma a permitir que a aplicação no servidor identifique adequadamente a que controle pertence a informação sendo remetida;
- Exceção feita a somente dois tipos de elementos: **radio** e **checkbox**. Os componentes destes dois grupos podem apresentar o **mesmo atributo name**;
- Ao dar nome a um atributo **name**, evite caracteres acentuados e espaços em branco. Use hífen ou sublinhado, se necessário.

RÓTULOS (<label>)

- Texto que define o tipo de informação que o controle enviará ao servidor. Este elemento está, geralmente, associado a outros controles do formulário. Exige fechamento;
- Atributo **for**, que associa o label a um determinado elemento do formulário. Na prática, quando o usuário clica sobre o label, o foco é automaticamente aplicado ao elemento correspondente, por meio do atributo **id**. Se o elemento da ligação for caixa de texto, o cursor é deslocado para este elemento.

```
<label for="senha"> Forneça sua senha: </label>
```

```
<input type="password" name="senha" id="senha">
```

Forneça sua senha:

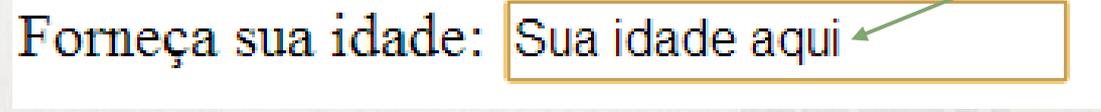
CAIXAS DE TEXTO SIMPLES (<input type="text">)

- Destinadas a receber uma única linha de informação. São um subgrupo de um grupo muito maior de elementos de formulário: a família **<input>**. Todo elemento da família **<input>** é uma **tag vazia e não necessita de fechamento**;
- Um atributo opcional é **value**, que define o que será mostrado dentro da caixa de texto;
- O que separa este elemento de outros elementos **<input>** da mesma família é o seu atributo **type**. No caso de caixa de texto, o atributo **type** tem o valor **text**. Exemplo:

<label> Forneça sua idade: </label>

<input type="text" name="idade" value="Sua idade aqui">

Forneça sua idade:



CAMPOS DE SENHA (<input type="password">)

- Elemento destinado a receber uma senha;
- O navegador substitui cada caractere digitado neste campo pelo símbolo * ou ●, para impedir que alguém visualize a informação enquanto está sendo digitada. Porém, a senha é transmitida ao servidor e por ele recebida no formato de texto, exatamente como foi digitada, sem nenhuma criptografia ou outro método de proteção de dados;
- Exemplo:

```
<label> Forneça sua senha: </label>
```

```
<input type="password" name="senha">
```

Forneça sua senha:

CAIXAS DE TEXTO COM MÚLTIPLAS LINHAS (<textarea>)

- Este elemento permite que o usuário digite várias linhas de texto;
- Exige abertura e fechamento;
- O conteúdo a ser exibido é digitado logo após a abertura da tag. Exemplo:

```
<label> Deixe aqui suas sugestões: </label>  
<br>
```

```
<textarea name="sugestoes">Texto aqui  
</textarea>
```

Deixe aqui suas sugestões:

Texto aqui

CAIXAS DE SELEÇÃO (<select>)

- Oferecem um menu retrátil, que permite a escolha de um ou mais valores, dependendo da presença ou não do atributo **multiple**;
- Para caixas de seleção de um só valor, mesmo que o usuário não faça nenhuma seleção, o navegador envia ao servidor sempre o **primeiro elemento**;
- Para caixas de seleção de múltiplos valores, a aplicação no servidor irá reportar um erro se o usuário não selecionar nenhum valor no controle de formulário;
- Os itens dentro de um menu de seleção são dados pela tag **<option>**. Opcionalmente, cada <option> pode apresentar o atributo **value**. O conteúdo deste atributo é enviado pelo navegador ao servidor;
- Podemos fazer com que o navegador apresente determinado item já selecionado usando o atributo **selected** dentro do elemento <option>.

EXEMPLO 1 COM SELECT SIMPLES

```
<label> Programa de TV preferido: </label>  
<select name="programa-tv">  
  <option value="Jornal"> Jornal </option>  
  <option value="Novela"> Novela </option>  
  <option value="Documentário" selected>  
    Documentário </option>  
</select>
```

Programa de TV preferido:

- Jornal
- Novela
- Documentário

EXEMPLO 2 COM SELECT MÚLTIPLO

```
<label> Programas de TV preferidos: </label>  
<br>
```

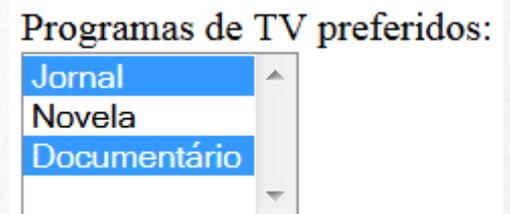
```
<select name="programas-tv[]" multiple>
```

```
  <option> Jornal </option>
```

```
  <option> Novela </option>
```

```
  <option> Documentário </option>
```

```
</select>
```



Programas de TV preferidos:

- Jornal
- Novela
- Documentário

• NOTAS:

- ✓ Uso obrigatório de `[]` logo após o atributo `name` do elemento. Sem estes colchetes, o servidor receberá apenas o último valor;
- ✓ Aqui a tag `<option>` também admite os atributos `value` e `selected`;
- ✓ Para o usuário marcar mais de um valor do menu, ele deve manter a tecla **control** ou **shift** pressionada e fazer a seleção com o mouse;
- ✓ Se o usuário não escolher um item do menu e, mesmo assim, enviar os dados ao servidor, o aplicativo em PHP retornará um erro.

BOTÕES DE RÁDIO (<input type="radio">)

- Fornece uma lista de opções, das quais apenas **UMA** única pode ser selecionada;
- O atributo **name** deste elemento deve ser o **mesmo** para todos os controles que se referem ao mesmo bloco de informações;
- O atributo obrigatório **value** representa a informação que o PHP irá receber nos servidor, enviada por este controle;
- O atributo **checked** faz o navegador mostrar aquele elemento já com a marca de seleção;
- Se o usuário não selecionar um botão de rádio, o aplicativo em PHP encarregado de receber e processar as informações no servidor acusará um erro.

EXEMPLO DE USO DE RADIO

```
<label> Sua faixa de renda: </label> <br>
```

```
<input type="radio" name="renda" value="Abaixo de R$2000,00"> Abaixo de R$2000,00 <br>
```

```
<input type="radio" name="renda" value="Igual ou acima de R$2000,00" checked> Igual ou acima de R$2000,00
```

Sua faixa de renda:

- Abaixo de R\$2000,00
- Igual ou acima de R\$2000,00

BOTÕES DE VERIFICAÇÃO (<input type="checkbox">)

- Fornece uma lista de opções, das quais **UMA OU MAIS** poderão ser selecionadas;
- O atributo **name** deste elemento deve ser o mesmo para todos os controles que se referem ao mesmo bloco de informações. Além disso, o atributo name deve usar um par de colchetes **[]**. Se os colchetes forem omitidos, o navegador enviará ao servidor somente o último elemento marcado;
- Se o usuário não selecionar, pelo menos uma opção, o aplicativo em PHP encarregado de receber e processar as informações no servidor acusará um erro;
- Com relação aos atributos **value** e **checked**, estes se comportam exatamente como quando utilizados com botões de rádio.

EXEMPLO DE USO DO CHECKBOX

```
<label> Suas especialidades: </label> <br>
```

```
<input type="checkbox" name="especial[]" value="PHP" checked> PHP <br>
```

```
<input type="checkbox" name="especial[]" value="Java"> Java <br>
```

```
<input type="checkbox" name="especial[]" value="JavaScript"> JavaScript
```

Suas especialidades:

- PHP
- Java
- JavaScript

BOTÕES (<input type="reset | submit | button">)

- Dois dos botões mais usados em um formulário são:
 - **O botão de limpeza dos campos:** quando acionado, o navegador recarrega o formulário com todos os campos em branco. Nada é enviado ao servidor;
 - **O botão de submissão dos dados:** quando pressionado, todas as informações do formulário são remetidas ao servidor e o navegador carrega novamente o formulário, com todos os campos nos seus valores-padrão;
 - **O botão de eventos:** quando pressionado, por si só, ele não submete nenhuma informação ao servidor, salvo se programação em JavaScript for fornecida. Sua função é oferecer a possibilidade de interação do formulário com uma linguagem de script na máquina cliente, como, por exemplo, associação a eventos do JavaScript;
- Do ponto de vista de uma aplicação para a web, o botão de limpeza e o de eventos não exigem a presença do atributo **name**, já que nada é remetido ao servidor. Porém, para o botão de submissão de dados, o atributo **name** é essencial;
- Existem muitos outros tipos de botões para formulários.

EXEMPLO DE USO DE BOTÕES

```
<label> Suas especialidades: </label> <br>  
<input type="checkbox" name="especial[]"> PHP <br>  
<input type="checkbox" name="especial[]"> Java <br>  
<input type="checkbox" name="especial[]"> JavaScript  
<br> <br>  
<input type="reset" value="Limpar campos">  
<input type="submit" name="enviar" value="Submeter  
dados">  
<input type="button" value="Executar JavaScript">
```

Suas especialidades:

- PHP
- Java
- JavaScript

Limpar campos

Submeter dados

Executar JavaScript

ENVIO DE ARQUIVOS AO SERVIDOR (<input type="file">)

- Controle de formulário que permite ao usuário subir arquivos para o servidor. A forma como este elemento é mostrado na página varia muito de navegador para navegador;
- Existem diversos atributos associados a este controle, mas um dos mais importantes é **multiple**. Ele é opcional, mas se estiver presente, o navegador permitirá que mais de um arquivo seja enviado ao mesmo tempo;
- Quando o atributo **multiple** for usado, repare a necessidade da presença de colchetes **[]** após o valor do atributo **name**;
- O script em PHP no servidor receberá o nome do arquivo e a informação propriamente dita contida no arquivo;
- Para que o navegador possa enviar arquivos, um novo atributo deve ser acrescentando à tag <form>: **enctype="multipart/form-data"** ;
- Outras questões importantes, como tipo de arquivo permitido e tamanho máximo de cada arquivo, são tratadas pela configuração do PHP no servidor.

EXEMPLOS DE USO DE ENVIO DE ARQUIVOS

```
<form action="receber-arquivos.php" method="post"
enctype="multipart/form-data">
  <label> Enviar fotos: </label>
  <input type="file" name="fotos[]" multiple>
</form>
```

```
<form action="receber-arquivos.php" method="post"
enctype="multipart/form-data">
  <label> Enviar UM pdf: </label>
  <input type="file" name="pdf">
</form>
```

Enviar fotos (Firefox): Nenhum arquivo selecionado.

Enviar fotos (Chrome): Nenhum arquivo selecionado

ENVIO DE DADOS OCULTOS (<input type="hidden">)

- Controle de formulário que permite envio de informações ao servidor sem que as mesmas estejam visíveis no formulário;
- Os atributos **name** e **value** são obrigatórios. Da mesma forma que com outros elementos, o name identifica o controle e o value é o dado recebido pelo PHP no servidor;
- Quando usado em conjunto com envio de arquivos no formulário, este controle é muito importante, permitindo fixar o tamanho máximo de cada arquivo enviado (em bytes). Se for usado com este propósito, este elemento deve aparecer sempre antes de <input type="file"> no HTML.

EXEMPLOS DE USO DE DADOS OCULTOS

- Formulário com campos ocultos, mas sem envio de arquivos:

```
<input type="hidden" name="CPF" value="18996745-84">
```

```
<input type="hidden" name="email" value="abc@meuservidor.com.br">
```

- Formulário com campos ocultos relativos a envio de arquivos. Neste exemplo, o tamanho máximo do arquivo pdf não pode ultrapassar 1024 bytes:

```
<form action="receber-arquivos.php" method="post" enctype="multipart/form-data">
```

```
<label> Enviar UM pdf: </label>
```

```
<input type="hidden" name="MAX_FILE_SIZE" value="1024">
```

```
<input type="file" name="pdf">
```

```
</form>
```

AGRUPADOR DE ELEMENTOS (<fieldset>)

- Existe um controle de formulário que existe unicamente com o objetivo de organizar e agrupar informações relacionadas entre si em um formulário. É o elemento **<fieldset>**. É puramente visual e não envia nenhuma informação ao servidor. Ele desenha uma linha retangular ao redor dos elementos que estão em seu interior. Seu elemento-filho é **<legend>**, responsável por mostrar texto que acompanha o agrupador;
- Ambas são tags completas e exigem fechamento.

EXEMPLOS DE USO AGRUPADOR DE ELEMENTOS

`<fieldset>`

`<legend>` Dados de acesso: `</legend>`

`<label>` Login: `</label>`

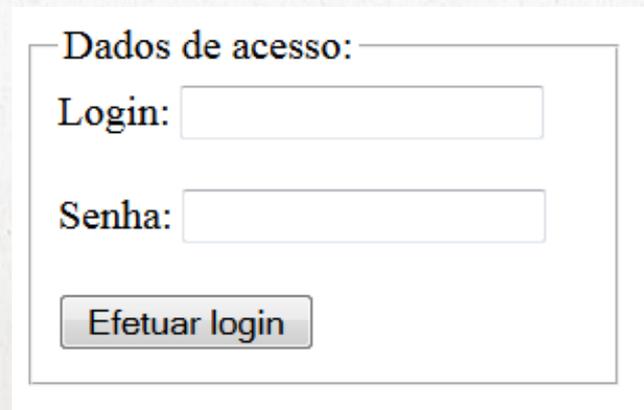
`<input type="text" name="login">` `
` `
`

`<label>` Senha: `</label>`

`<input type="password" name="senha">` `
` `
`

`<input type="submit" name="entrar"`
`value="Efetuar login">`

`</fieldset>`



The image shows a rendered HTML form. It features a rectangular border containing the text "Dados de acesso:" followed by a horizontal line. Below this, there are two input fields: "Login:" followed by a text input box, and "Senha:" followed by a password input box. At the bottom of the form is a button labeled "Efetuar login".

ATRIBUTOS READONLY E DISABLED

- São dois atributos que controlam a interação do usuário com o formulário. São aplicados à maioria dos elementos;
- Um controle com atributo **readonly** não pode ser editado pelo usuário, mas envia seu conteúdo ao servidor. Também pode receber o foco (via mouse ou tecla tab);
- Um controle com atributo **disabled** não pode ser editado de nenhuma forma, não envia seu conteúdo ao servidor e não pode receber o foco.



Login:

Senha:

```
<label> Login: </label>
```

```
<input type="text" name="login" readonly>
```

```
<label> Senha: </label>
```

```
<input type="password" name="senha" disabled>
```