

Desenvolvimento Web

Aula – Formulários HTML

Prof. Dr. Fernando Parra dos Anjos Lima
fernando.lima@tga.ifmt.edu.br

Aula Anterior

Exemplo de formulário

| | |
|---|---|
| CPF | <input type="text" value="999.999.999-99"/> |
| Endereço | <input type="text"/> |
| Cidade | <input type="text" value="Votuporanga"/> |
| Idade | <input type="text"/> |
| Quantidade | <input type="range" value="50"/> |
| E-mail | <input type="text" value="address@server"/> |
| Telefone | <input type="text" value="(xx) xxxxx-xxxx"/> |
| URL Site | <input type="text" value="http://yourdomain.aa"/> |
| <input type="button" value="Enviar"/> <input type="button" value="Limpar"/> | |

Caixa de Data e Horas

- Os primeiros campos são para datas e horas. Há diversos tipos de caixas de entrada de datas e horas na versão 5 do HTML. Todas essas caixas são adicionadas mudando o atributo type do elemento input.
- Os tipos possíveis são os seguintes:
 - date: Utilizado para coletar data (dia, mês e ano) sem fuso horário.
 - datetime-local: Utilizado para coletar data (dia, mês e ano) e hora (hora, minuto, segundo e fração de segundo) sem fuso horário.
 - month: Utilizado para coletar data composta por mês e ano sem fuso horário.
 - time: Utilizado para coletar hora (hora, minuto, segundo e fração de segundo) sem fuso horário.
 - week: Utilizado para coletar data composta por semana e ano sem fuso horário.

Caixa de Data e Horas

- Veja o Código:

```
<label for="data" >Data</label>  
<input id="data" name="data"  
       type="date"><br>
```

```
<label for="data_hora" >Data e Hora</label>  
<input id="data_hora" name="data_hora"  
       type="datetime-local"><br>
```

Mãos a Obra!!

- Crie um Documento básico com a estrutura HTML e um formulário.
- Crie os campos de data e data e hora:

```
<label for="data" >Data</label>  
<input id="data" name="data"  
       type="date"><br>
```

```
<label for="data_hora" >Data e Hora</label>  
<input id="data_hora" name="data_hora"  
       type="datetime-local"><br>
```

Mãos a Obra!!

- Resultado

Exemplo de formulário (nº 2)

Data

Data e Hora

Mãos a Obra!!

- Insira um botão Enviar e um botão Limpar.
- Resultado:

Exemplo de formulário (nº 2)

Data

Data e Hora

Caixa de Senha

- Outro tipo importante são as caixas de senhas, pois elas devem ser coletadas com caixas específicas para esse tipo de informação.

Username:

jdoe

Password:

.....|

Login

Caixa de Senha

- Normalmente, os navegadores utilizam símbolos como o asterisco ou o círculo para omitir o conteúdo das caixas de senha.
- Para adicionar uma caixa de senha em um formulário, devemos utilizar o elemento input com o valor password para o atributo type.

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

Caixa de Senha

- Veja o código:

```
<label for="senha" >Senha</label>  
<input id="senha" name="senha"  
       type="password" >
```

Mãos a Obra!!

- Crie o campo de senha e teste seu funcionamento.

```
<label for="senha" >Senha</label>  
<input id="senha" name="senha"  
       type="password" >
```

Mãos a Obra!!

- Resultado:

Exemplo de formulário (nº 2)

Data

Data e Hora

Senha



Caixas de Texto Longo

- Para coletar um texto com várias linhas, podemos utilizar o elemento textarea.
- Com esse elemento podemos ajustar a quantidade de linhas com o atributo rows e a quantidade de colunas com o atributo cols.
- Preste atenção, diferente dos demais campos este não é uma TAG INPUT, porém ele também deve ter um atributo name.

```
<textarea id="mensagem" name="mensagem" rows="4" cols="30"></textarea><br>
```

Mãos a Obra!!

- Crie o campo de texto longo e teste seu funcionamento.

```
<label for="mensagem">Mensagem</label>  
<textarea id="mensagem" name="mensagem"  
          rows="4" cols="30"></textarea><br>
```

Mãos a Obra!!

- Resultado:

Exemplo de formulário (nº 2)

| | |
|---|---|
| Data | <input type="text" value="dd/mm/aaaa"/> |
| Data e Hora | <input type="text" value="dd/mm/aaaa --:--"/> |
| Senha | <input type="password"/> |
| Mensagem | <div><div></div></div> |
| <div><div>Enviar</div><div>Limpar</div></div> | |

Checkboxes e Radios

- Para adicionar um checkbox em um formulário, devemos utilizar o elemento input com type igual a checkbox. Ao utilizar esse componente, é importante definir um valor para o atributo value.

☐ Java ☒ PHP ☒ Perl

- O campo do tipo checkbox também é chamado de campo de checagem múltipla devido a sua característica onde várias opções podem ser selecionadas.

Checkboxes e Radios

- No envio do formulário, esse valor é transmitido ao servidor se o checkbox correspondente estiver marcado.

```
<input type="checkbox" value="java" >
```

- Eventualmente é interessante agrupar um determinado conjunto de checkboxes. Por exemplo, considere um formulário que coleta as linguagens de programação que os usuários conhecem.

Checkboxes e Radios

- Para cada linguagem, podemos definir um checkbox. Para agrupá-los basta definir o atributo name com o mesmo valor para eles.

```
<label>Linguagem</label>  
<input type="checkbox" name="linguagens" value="java">Java  
<input type="checkbox" name="linguagens" value="php">PHP  
<input type="checkbox" name="linguagens" value="Perl">Perl<br>
```

- Observe que não foi colocado o atributo “for” no label, isso foi feito pois não há um único campo para receber o foco quando clicamos no rótulo. Outro ponto importante é que não se pode ter IDs duplicados.

Checkboxes e Radios

- Veja o código completo:

```
<label>Linguagem</label>
<input type="checkbox" name="linguagens"
      value="java">Java
<input type="checkbox" name="linguagens"
      value="php">PHP
<input type="checkbox" name="linguagens"
      value="Perl">Perl<br>
```

Mãos a Obra!!

- Veja o código completo:

```
<label>Linguagem</label>
<input type="checkbox" name="linguagens"
      value="java">Java
<input type="checkbox" name="linguagens"
      value="php">PHP
<input type="checkbox" name="linguagens"
      value="Perl">Perl<br>
```

Mãos a Obra!!

- Resultado:

Exemplo de formulário (nº 2)

Data

Data e Hora

Senha

Mensagem

Linguagem ☐ Java ☐ PHP ☐ Perl

Checkboxes e Radios

- Vamos agora a um segundo tipo de campo de checagem, o campo de checagem única. Ele é muito semelhante ao anterior, a diferença é que devemos usar o atributo type do input como radio.

☒ São Paulo ☐ Barcelona ☐ Milan

- Ao utilizar esse componente, é importante definir um valor para o atributo value. No envio do formulário, esse valor é transmitido ao servidor se o radio correspondente estiver marcado.

```
<input type="radio" value="M" >
```

Checkboxes e Radios

- Algo que devemos prestar muita atenção é quanto ao atributo name, pois para que o campo funcione corretamente devemos definir o mesmo name para todos os campos que fazem parte do grupo.

```
<label>Sexo</label>  
<input type="radio" name="sexo" value="M">Masc  
<input type="radio" name="sexo" value="F">Femi  
<input type="radio" name="sexo" value="I">Inde
```

- Caso os nomes não sejam definidos ou sejam diferentes será possível marcar todas as opções.

Checkboxes e Radios

- Veja o Código completo:

```
<label>Sexo</label>
<input type="radio" name="sexo"
      value="M">Masculino
<input type="radio" name="sexo"
      value="F">Feminino
<input type="radio" name="sexo"
      value="I">Indefinido<br>
```


Mãos a Obra!!

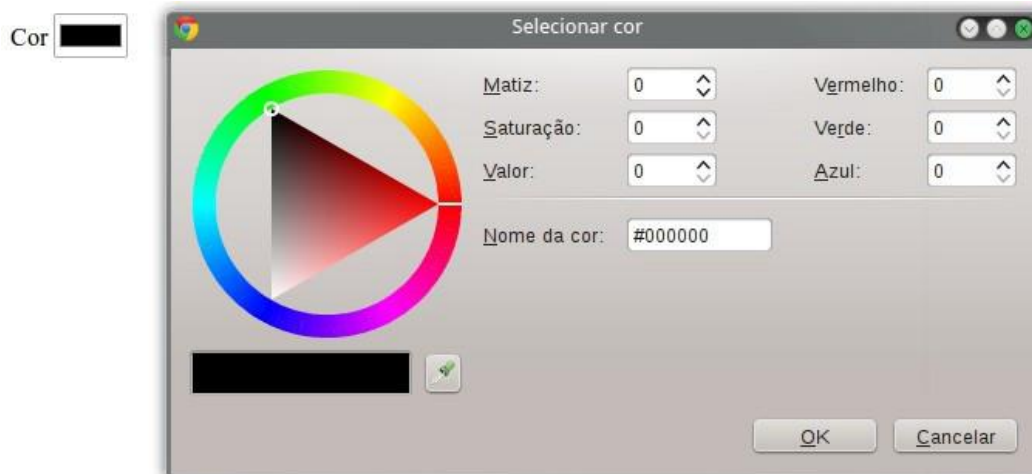
- Crie os campos radio:

```
<label>Sexo</label>
<input type="radio" name="sexo"
value="M">Masculino
<input type="radio" name="sexo"
value="F">Feminino
<input type="radio" name="sexo"
value="I">Indefinido<br>
```

Seleção de cores

- Em algumas situações pode ser necessário criar um campo para seleção de cores, o HTML5 possui um campo especial para essa finalidade.
- Para se criar um campo para seleção de cores basta criar um input com o valor do atributo type igual a color.

```
<input type="color" >
```



Seleção de cores

- Código completo:

```
<label for="cor" >Cor</label>  
<input id="cor" name="cor"  
      type="color" ><br>
```

Mãos a Obra!!

- Código completo:

```
<label for="cor" >Cor</label>  
<input id="cor" name="cor"  
      type="color" ><br>
```

Mãos a Obra!!

- Resultado: **Exemplo de formulário (nº 2)**

Data

Data e Hora

Senha

Mensagem

Linguagem ☐ Java ☐ PHP ☐ Perl

Sexo ☒ Masculino ☐ Feminino ☐ Indefinido

Cor

Campo de Upload

- Também chamado de botão de upload, esse campo serve para que possamos fazer upload de arquivos.
- Para adicioná-lo em um formulário, utilizamos o elemento `input` com o `type` igual a `file`.

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

Botão de *Upload* Nenhum arquivo selecionado

Campo de Upload

- **Importante:**
- O formulário que contém esse botão deve possuir o atributo enctype com o valor multipart/form-data.

```
<form action="pagina.html"  
      enctype="multipart/form-data" >
```

Mãos a Obra!!

- Crie um botão de upload no formulário:

```
<label for="botao_upload">  
    Botão de <i>Upload</i></label>  
<input type="file" name="botao_upload"  
        id="botao_upload"><br>
```


Mãos a Obra!!

- Resultado:

-

Linguagem ☐ Java ☐ PHP ☐ Perl

Sexo ☒ Masculino ☐ Feminino ☐ Indefinido

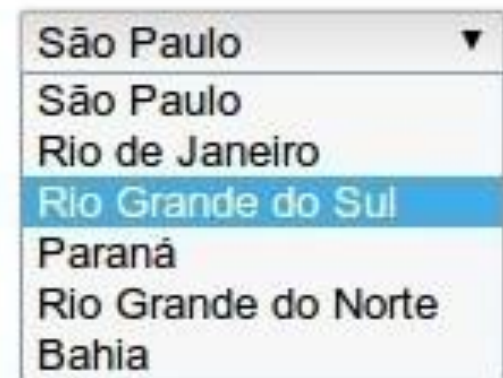
Cor

Botão de *Upload* Nenhum arquivo selecionado

Drop-Down List

- Este campo permite que os usuários selecionem um ou mais itens de uma lista de opções. Este campo é chamado de drop-down list ou apenas select.
- Para adicionar esse tipo de componente, devemos utilizar o elemento select.

```
<select id="estado" name="estado">  
</select>
```



Drop-Down List

- O código anterior serve apenas para declarar o campo, porém é necessário criar as opções.
- As opções devem ser definidas no conteúdo da tag select usando o elemento option.
- É o conteúdo do elemento option é exibido para os usuários.

```
<label for="estado">Estado</label>
<select name="estado" id="estado">
  <option value="SP">São Paulo</option>
```

O valor que será gravado.


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

O que aparece na tela.

Drop-Down List

- Esse elemento possui um atributo chamado value. Quando o formulário for enviado, o valor do atributo value é transmitido ao servidor se a opção correspondente foi selecionada.

```
<label for="estado">Estado</label>  
<select name="estado" id="estado">  
  <option value="SP">São Paulo</option>  
</select><br>
```



Drop-Down List

- Se esse atributo não estiver definido, o conteúdo do elemento option é transmitido ao servidor.

```
<select id="estado" name="estado">  
  <option>São Paulo</option>  
  <option>Rio de Janeiro</option>  
  <option>Rio Grande do Sul</option>
```

- Quando queremos que uma determinada opção esteja selecionada por padrão usamos o atributo selected do elemento option. Esse atributo não precisa de valor.

Drop-Down List

- Veja o exemplo a seguir:

```
<label for="estado" >Estado</label>
<select id="estado" name="estado">
  <option value="SP" selected>São Paulo</option>
  <option value="RJ">Rio de Janeiro</option>
  <option value="RS">Rio Grande do Sul</option>
  <option value="PR">Paraná</option>
  <option value="RN">Rio Grande do Norte</option>
  <option value="BA">Bahia</option>
</select><br>
```


Mãos a Obra!!

- Crie o drop down list mostrado a seguir:

```
<label for="estado" >Estado</label>
<select id="estado" name="estado">
  <option value="SP" selected>São Paulo</option>
  <option value="RJ">Rio de Janeiro</option>
  <option value="RS">Rio Grande do Sul</option>
  <option value="PR">Paraná</option>
  <option value="RN">Rio Grande do Norte</option>
  <option value="BA">Bahia</option>
</select><br>
```

Mãos a Obra!!

- Resultado:

Cor

Botão de *Upload* Escolher arquivo Nenhum arquivo selecionado

Estado

São Paulo ▼

E

Rio de Janeiro

Rio Grande do Sul

Paraná

Rio Grande do Norte

Bahia

Autocomplete

- Algumas vezes desejamos dar algumas opções aos usuários, mas não podemos “fechar” as opções, por isso não podemos usar um drop down list, assim podemos usar o recurso do autocomplete.
- Este recurso cria uma lista de sugestões para campos do tipo texto.



Autocomplete

- Para utilizar esse recurso, devemos criar uma lista de sugestões com o elemento `datalist`. É fundamental identificar as listas de sugestões com o atributo `id`.

```
<datalist id="comidas" >  
  
</datalist>
```

- As opções devem ser definidas no conteúdo do elemento `datalist` e elas são adicionadas com o elemento `option`. O atributo `value` de cada `option` define uma sugestão.

```
<option value="Lasanha">  
<option value="Pizza">  
<option value="Sopa">
```

Autocomplete

- Veja o código completo de um datalist:

```
<datalist id="comidas" >  
  <option value="Lasanha">  
  <option value="Pizza">  
  <option value="Sopa">  
  <option value="Salada">  
</datalist>
```

Autocomplete

- Com a lista de sugestões criada, devemos associá-la a uma caixa de entrada através do atributo list do elemento input.

```
<input id="comida_preferida" name="comida_preferida"  
type="text" list="comidas" >
```

- Assim temos nosso autocomplete pronto. Este recurso ajuda bastante na usabilidade dos sites, pois facilita o preenchimento dos campos.

Mãos a Obra!!

- Crie um datalist:

```
<datalist id="comidas">  
  <option value="Lasanha">  
  <option value="Pizza">  
  <option value="Suchi">  
  <option value="Sopa">  
  <option value="Salada">  
</datalist>
```

Mãos a Obra!!

- Crie o campo de texto associando o datalist recém criado a este campo:

```
<label for="comida_pref">  
    Comida pref.</label>  
<input type="text" name="comida_pref"  
        id="comida_pref"  
        list="comidas"><br>
```

Mãos a Obra!!

- Resultado:

Botão de *Upload* Escolher arquivo Nenhum arquivo selecionado

Estado São Paulo ▼

Comida pref. la ▼

- Lasanha
- Salada

Obrigado !!

Prof. Dr. Fernando Parra dos Anjos Lima
fernando.lima@tga.ifmt.edu.br