

# DESENVOLVIMENTO DE APLICAÇÕES WEB

## FORMULÁRIOS HTML

### 1 - Introdução

Como conversamos no início do curso, o objetivo das aulas de LAB é a realização de operações tendo formulários como entrada. Em geral, os dados fornecidos serão repassados pelo HTTP Server (Web Server) à aplicação que, com eles, realizará diversas operações como cálculos, tratamento das regras de negócio, acesso a banco de dados, etc.

No nosso curso, o tratamento dos dados será realizado por scripts PHP que obterão as informações recebidas dos formulários e fará a interação com um SGBD. Para isto, precisaremos conhecer alguns elementos básicos de formulários.

Tipos de entrada (não todos):

- a) text
- b) password
- c) submit
- d) image
- e) reset
- f) radio
- g) checkbox
- h) hidden
- i) select
- j) textarea
- k) file
- l) color
- m) search
- n) tel
- o) url
- p) email
- q) datetime
- r) datetime-local
- s) date
- t) time
- u) month
- v) week
- w) number
- x) range

Veremos, hoje, alguns destes tipos de entrada através de exemplos e, também, alguns atributos, lembrando que estão presentes apenas os relevantes para as atividades do

momento. Estilos CSS podem ser utilizados de forma semelhante como fizemos em aulas anteriores.

## 2. Estrutura geral de um formulário

```
<form method = "post" action = "xxxxxx.php">  
...  
<input type="text" name="nome" size=20 maxlength=30>  
...  
</form>
```

- a) A tag que identifica o formulário é

```
<form .....>  
.  
.  
.  
.  
</form>
```

- b) O atributo

```
method
```

indica qual a forma pela qual os dados serão enviados ao servidor. Você pode utilizar “get” ou “post”. Existem alguns aspectos sobre a escolha. No nosso, porém, em razão dos campos para usuário e senha, utilizaremos o método “post”. Além disso, o nosso formulário poderá ser utilizado para atualização dos dados.

- c) O atributo

```
action
```

indica o nome do script que tratará o formulário. De outro ângulo, podemos entender que se trata do script que será invocado pelo HTTP Server para o tratamento do formulário.

- d) Dentro do formulário ocorrerão um ou vários campos de entrada que serão indicados pela tag

```
<input ...>
```

- e) O atributo “type” da tag <input> especifica qual o controle do campo de entrada, ou seja, se o campo será um texto digitado, um botão de rádio, um checkbox, etc...

- f) O atributo

```
name
```

da tag de <input> indica o nome do campo de entrada, ou, mais ainda, o nome da variável com que o valor será passado ao script tratador.

### 3. Campo do tipo texto

Veja o exemplo

Digite o seu nome:

```
<input type="text" name="nome" size=20 maxlength=30>
```

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

Digite o seu apelido:

```
<input type="text" name="apelido" size=15 maxlength=20  
value="Não digite apelidos depreciativos">
```

O atributo

```
size
```

é o tamanho do campo visual de digitação e

```
maxlength
```

o tamanho máximo do campo de entrada.

O atributo

```
value
```

é o texto que aparecerá preenchido no campo e também o que será retornado na submissão. Para fim de apresentação, utilize o atributo “placeholder”.

### 4. Botão submit

Este “controle”, quando acionado, envia os dados do formulário ao webserver que os repassa ao script tratador. Ou seja, este botão aciona o script responsável pelo tratamento do formulário.

```
<input type="submit" name="submit" value="Envia">
```

O atributo “value” é o texto que aparece no botão de submissão. Se o atributo “name” estiver presente, faz com que o browser envie o par nome/valor correspondente ao botão pressionado. Isto permite que o formulário tenha vários botões do tipo “submit” e o script possa escolher qual ação realizar, de acordo com o botão utilizado para a submissão do formulário.

## 5. Entrada do tipo “Image”

Tem a mesma função da entrada do tipo “submit”, mas permite uma imagem em lugar de texto no botão.

```
<input type="image" src="/daw/image.gif" alt="Envia">
```

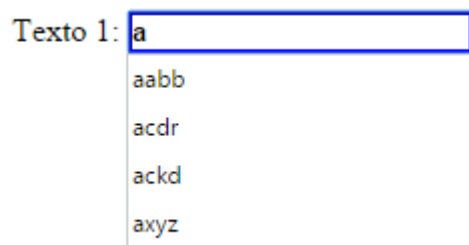
## 6. Atributo “required”

O campo de entrada com este atributo torna o seu preenchimento obrigatório, como mostra a figura abaixo. A mensagem aparece ao pressionar o botão de envio do formulário



## 7. Atributo “autocomplete”

O valor deste atributo assume “on” ou “off”, sendo o primeiro o valor “default”. Quando ativado, assim que começar a preencher o campo, apresenta as possibilidades conhecidas e eventos anteriores.



## 8. Atributo “placeholder”

Este atributo permite especificar no campo de entrada uma explicação breve do conteúdo. O valor não é considerado como valor digitado ao submeter o formulário. O campo

“value” em algumas situações pode ter o a mesma função, mas este campo é passado como conteúdo se não for alterado.

Texto para pesquisa:

## 9. Atributo “autofocus”

Este atributo permite especificar o campo de entrada que estará em foco quando o formulário for carregado. O cursor se posicionará no campo se ele for editável. Se mais que um campo tiver este atributo, o comportamento é dependente do produto utilizado.

## 10. Atributo “pattern”

A partir do HTML 5, é possível especificar o padrão de cadeias que pode ser aceito como entrada de dados. Vários exemplos com explicações estão disponíveis em muitas fontes. Um erro será apontado quando o elemento de submissão for acionado, mas a validação ocorre sem que o formulário seja enviado ao script tratador no lado servidor.

```
<label for="cpf"> Digite o seu CPF: </label>

<input type = "text" name = cpf required autofocus
placeholder = "999.999.999-99" id = "cpf"
title = "Formato: 999.999.999-99"
pattern="\d{3}\.\d{3}\.\d{3}\-\d{2}">
```

Digite o seu CPF:

! Faça corresponder o formato  
pedido.  
Formato: 999.999.999-99

## 11. O elemento “label”

Este elemento faz a ligação da mensagem com o campo de entrada através do seu atributo “id” com o atributo “for” da “tag” “label”. Com isto, o texto fica conectado ao elemento de entrada e quando o “mouse” faz o movimento sobre o texto, o “foco” passa para a entrada correspondente.

## 12. Campo do tipo password

Aspecto de um campo do tipo “password”:

```
<input type="password" name="senha" size=10 maxlength=16>
```

## 13. Botão “reset”

Este recurso faz com que todos os campos voltem à condição do início do formulário, sejam para textos, botões pré-escolhidos, checkbox, etc. O que foi escolhido ou digitado é desfeito. Os textos originais ou escolhas que estavam na apresentação inicial dos formulários ficarão válidos.

```
<input type= "reset" value="Apaga tudo">
```

## 14. Botões de rádio

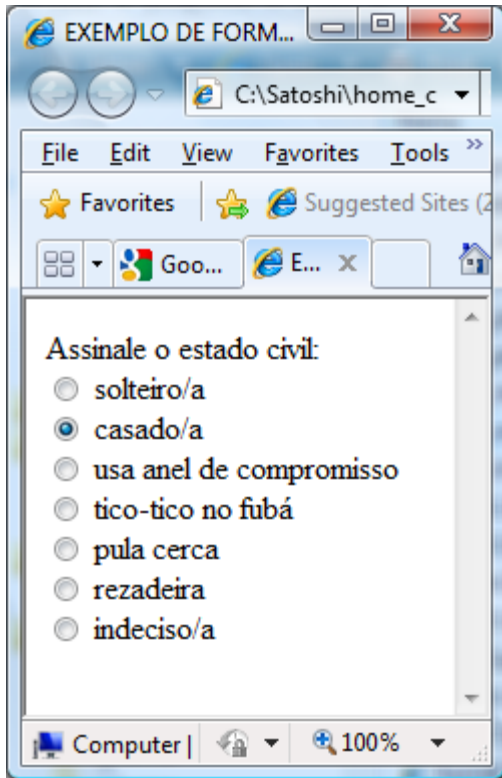
O próximo elemento de entrada que utilizaremos é composto por itens de “botões de rádio”. Veja o fragmento abaixo seguinte:

```
<label> Assinale o estado civil:<br><br>
  <input type="radio" name="estado_civil"
    value="solteiro"> solteiro/a <br>
  <input type="radio" name="estado_civil"
    value="casado"> casado/a <br>
  <input type="radio" name="estado_civil"
    value="compromissado"> usa anel de compromisso<br>
  <input type="radio" name="estado_civil"
    value="desconhecido"> tico-tico no fubá<br>
  <input type="radio" name="estado_civil"
    value="muitos" required> pula cerca<br>
  <input type="radio" name="estado_civil"
    value="esperancosa"> rezadeira<br>
  <input type="radio" name="estado_civil"
    value="indecisa"> indeciso/a<br><br>
</label>
```

O atributo “name” designa o que está sendo escolhido, neste caso, o estado civil. Este nome é importante porque é a variável pelo qual o valor escolhido é transmitido para o script tratador do formulário. O atributo “value” é o conteúdo. Ele difere do que é apresentado ao usuário.

Eventualmente, você deseja deixar marcado o elemento mais comum. Basta acrescentar a palavra chave “checked” para que o item apareça pré-escolhido. Testes demonstram que o atributo “required” pode estar em apenas uma das alternativas, mas é necessário validar nos navegadores que serão utilizados.

O aspecto visual do exemplo acima seria parecido com o seguinte:

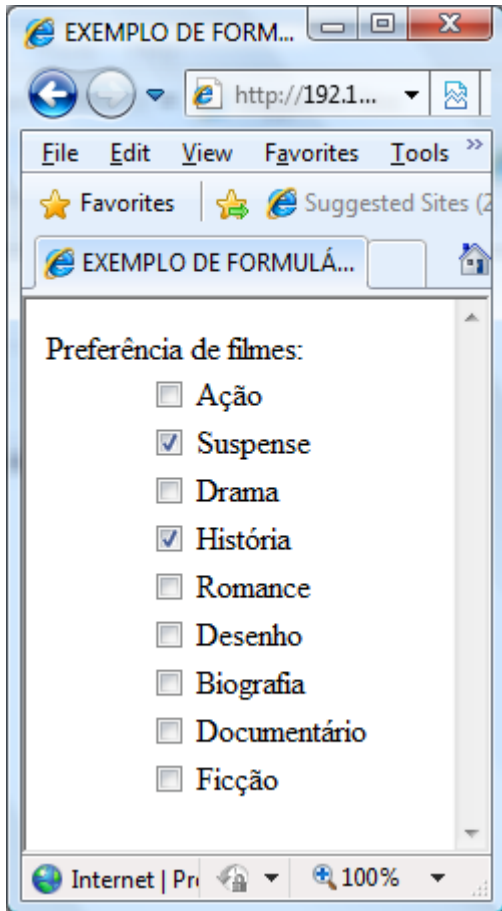


## 15. Check Box.

Outro tipo de entrada num form é o ‘Check box’. Diferentemente do “radio button” podem ser escolhidas várias alternativas. O atributo “type” da tag INPUT é o “checkbox”. Este tipo assume a seguinte forma geral:

```
<label> Preferência de filmes: <br><br>
  <input type="checkbox" name="preferencia_filme[]"
    value="1"> Ação<br>
  <input type="checkbox" name="preferencia_filme[]"
    value="2" checked> Suspense<br>
  <input type="checkbox" name="preferencia_filme[]"
    value="3"> Drama<br>
  <input type="checkbox" name="preferencia_filme[]"
    value="4" checked> História<br>
  <input type="checkbox" name="preferencia_filme[]"
    value="5"> Romance<br>
  <input type="checkbox" name="preferencia_filme[]"
    value="6"> Desenho<br>
  <input type="checkbox" name="preferencia_filme[]"
    value="7"> Biografia<br>
  <input type="checkbox" name="preferencia_filme[]"
    value="8"> Documentário<br>
  <input type="checkbox" name="preferencia_filme[]"
    value="9"> Ficção<br>
</label>
```

O aspecto visual das entradas acima seria o seguinte:



Observações:

- O mesmo nome utilizado no atributo "name" indica que todas as alternativas apresentadas referem-se ao mesmo item de entrada;
- Este tipo de entrada permite múltiplas escolhas, diferentemente do botão de rádio. Neste caso, desejando capturar duas ou mais alternativas assinaladas, você deverá indicar esta capacidade adicionando o símbolo de "[ ]" (de array) ao valor do atributo "name".

```
name="preferencia_filme[]"
```

Se for colocado somente o nome sem a indicação de que é um vetor, será permitido o múltiplo assinalamento, mas somente um valor será entregue ao script tratador.

- Para que uma alternativa apareça pré-selecionada, acrescentar o atributo checked à entrada.



```
<input type="checkbox" name="preferencia_filme[]"
      value="2" checked> Suspense<br>
```

- d) O valor do atributo “value” é o conteúdo que será repassado ao script tratador, se a respectiva alternativa for selecionada.

## 16. Entrada do tipo hidden.

Este tipo de entrada permite que seja passado ao script tratador do formulário dados pré-estabelecidos que não sejam apresentados ao usuário. Formato geral:

```
<input type="hidden" name="matricula" value="RA99999999">
```

Este tipo de entrada pode ser utilizado em formulários compostos de várias interações, de forma que valores digitados numa fase, ou parte deles, sejam reenviados na fase seguinte como campos ocultos.

## 17. Atividades de hoje

17.1. Crie um formulário que permita o cadastramento de uma pessoa num sistema hipotético com as características abaixo. Salve-o no arquivo form1.html, envie-o ao servidor e verifique. Em todas as entradas, utilize o elemento “label” e assinale-as como obrigatórias.

### Parte I – Dados gerais do formulário

- a) Título da janela do browser: Cadastramento de cliente
- b) Método “post”
- c) Script tratador, por enquanto, “xxx.php”

### Parte II – Identificação do funcionário cadastrante

- a) Nome do usuário (Deve aparecer o texto “Digite seu login” antes do campo). Este campo deve ter o foco na carga do formulário. Informe no próprio campo, o seu significado com as palavras “Seu Usuário” de modo que este texto não faça parte do conteúdo se não for alterado. Quando o formulário for carregado, este campo deve estar sob o “foco” ou cursor.
- a) Senha.

### Parte III – Dados do usuário a cadastrar:

- b) Nome do usuário. Coloque no atributo “value” a frase: “Nome do usuário”.
- c) CPF com tipo texto e atributo “pattern”.
- d) RG. Não devem aparecer sugestões com textos de digitações anteriores.
- e) Endereço. Apenas um campo texto.

- f) Estado. O campo “estado” do endereço poderia ser apresentado como uma “drop-down box”, mas por enquanto, crie três botões de rádio com os estados de São Paulo, Rio de Janeiro e Minas Gerais. O texto apresentado deve ser o nome, mas o valor retornado deve ser a sigla. Apresente o botão do estado de São Paulo pré-selecionado.
- g) Botões de rádio para escolher o tipo de pessoa. Assinale uma delas como pré-escolhida. O usuário deverá visualizar a classificação textual, mas o valor retornado deve ser o numérico.

Código	Classificação do usuário
-----	-----
1	Aluno
2	Funcionário
3	Administrador

- h) Botões de rádio para sexo. Valores de retorno “H” ou “M”. Apresentar na tela os textos “Homem” ou “Mulher”. O primeiro como “default”.
- i) Estado civil. Veja o quadro abaixo para montar os botões de rádio do seu formulário para escolha do estado civil. O valor de retorno deve ser o código e o valor apresentado, a descrição.

Código	Descrição
-----	-----
1	Solteiro
2	Casado
3	Desquitado
4	Divorciado
5	Viúvo
6	Separado
7	Amigado

- j) Botão “reset”
- k) Incluir um elemento “submit”.

17.2. Criar o arquivo form1a.html que permita escolher modelos de carros entre várias alternativas, apresentadas em seis entradas do tipo “checkbox” dentro de um formulário, com as seguintes condições:

- Deve enviar todas as alternativas assinaladas, mesmo que várias, ou seja, permite múltiplas escolhas;
- Pré-selecionar três modelos quaisquer;
- Envia códigos em letras, embora mostre os nomes dos modelos;
- Defina o atributo name com o valor “modelos”;
- A escolha é obrigatória. Utilize o elemento “label”.

17.3. Faça uma cópia do arquivo form1a.html com o nome form1b.html. Adicione um campo de entrada do tipo “hidden”, com os seguintes atributos:

Name: matricula

Value: seu registro acadêmico.

Carregue o formulário. Não aparecerá nada de diferente em relação ao item anterior. Utilizando o recurso de visualização do “código fonte” verifique o efeito do campo “hidden”.

17.4. Faça uma cópia do arquivo form1b.html com o nome form1c.html. Adicione entradas do tipo “submit”, de forma que, quando submetido, seja possível identificar qual delas foi pressionada. Defina o atributo name com o valor “destino” para todas elas.

- a) O primeiro botão deve ter o atributo value igual a "PUC";
- b) O segundo botão deve ter o atributo value igual a "google";
- c) O terceiro botão deve ter o atributo value igual a "UOL";

Neste exemplo, supõe-se que o redirecionamento de página ocorrerá sob o comando do servidor onde se localiza o script tratador. Na aula anterior, fizemos isto utilizando o elemento “<a>” que define um “link”. Existe diferença?

Até a próxima aula!

Prof. Satoshi Nagayama