

Hello World

Módulo HTML5,CSS3 E JS

Relembrando...

HTML

1. Tabelas
 - a. Table, tr, td, th, thead, tbody, tfoot
2. Iframes
3. Âncoras (marcadores)
4. Hipermídia
 - a. audio, video



Tabelas

Iframes

Âncoras

Hipermídia

Sobre a aula anterior...

Dúvidas?

HTML

Aula 03

Vamos dar continuidade
vendo mais elementos
HTML - Formulários



Formulário

O elemento formulário, representado pela tag `<form>`, é responsável por coletar dados de entrada do usuário.



Setup your account

Make sure to use a valid email address, you'll need to verify it before you can send any campaigns.

Name

Email Address

Username

Password

Company

Country

Timezone

Formulário - Elementos

Elementos de formulário são tags específicas para captar diferentes tipos de entradas do usuário, como datas, senhas, email, clique de botão, escolha de dados em uma lista, etc.

Elementos de um formulário são encapsulados pela tag `<form>`.

```
1▼ <!-- Inicia/Abre o formulário -->
2▼ <form>
3
4
5▼ <!-- Finaliza/Fecha o formulário -->
6  </form>
```

Formulário - Elemento: input

O elemento <input> é o mais importante de um formulário, pois com ele diversos tipos de dados podem ser coletados, a depender do que for colocado em seu atributo "type".

```
1▼ <!-- Inicia/Abre o formulário -->
2▼ <form>
3
4    <input type="" />
5
6▼ <!-- Finaliza/Fecha o formulário -->
7  </form>
```

Formulário - Elemento: input - type="text"

O elemento <input> é o mais importante de um formulário, pois com ele diversos tipos de dados podem ser coletados, a depender do que for colocado em seu atributo "type".

```
1 ▾ <!-- Inicia/Abre o formulário -->
2 ▾ <form>
3
4   <input type="text"/>
5
6 ▾ <!-- Finaliza/Fecha o formulário -->
7   </form>
```

Formulário - Elemento: label

O elemento `<label>` é usado para definir um rótulo/marcador para algum elemento do tipo `<input>` por exemplo.

Ele não renderiza nada de especial, ele serve para melhoria de usabilidade. Quando o usuário clica no nome referente a algum campo no formulário, ele foca direto no input que está "ligado" a ele.

Formulário - Elemento: label

```
1 <!-- Inicia/Abre o formulário -->
2 <form>
3
4   <label for="primeiroNome">Primeiro Nome</label>
5   <input type="text" name="primeiroNome" id="primeiroNome"/>
6
7 <!-- Finaliza/Fecha o formulário -->
8 </form>
```


Formulário - Atributo: placeholder

O atributo placeholder é utilizado para dar dica para o usuário, de como ele deve digitar num campo específico.

Esse atributo é valido quando o valor do atributo **type** é igual a:

- text
- search
- tel
- url
- email

Formulário - Atributo: placeholder

```
1 <!-- Inicia/Abre o formulário -->
2 <form>
3
4   <label for="email">Email</label>
5   <input type="text" name="email" id="email" placeholder="email@dominio.com"/>
6
7 <!-- Finaliza/Fecha o formulário -->
8 </form>
```

O uso de placeholder não exclui o uso de label. Cada um possui um propósito.

Label indica que tipo de informação é esperada no input, e o placeholder é dica de como inserir a informação.



Formulário - Elemento: input - type="password"

O atributo "type" de valor "password" indica que o input receberá dados a de senha, e por isso ele esconderá os caracteres digitados.

```
1 ▾ <!-- Inicia/Abre o formulário -->
2 ▾ <form>
3
4     <input type="password" name="senha"/>
5
6 ▾ <!-- Finaliza/Fecha o formulário -->
7  </form>
```

Formulário - Elemento: input - type="radio"

O atributo "type" de valor "radio" indica que o input receberá dados a partir da escolha de um valor de uma lista de itens pré-determinados.

```
<!-- Inicia/Abre o formulário -->  
<form>  
  
    <input type="radio"/>  
  
<!-- Finaliza/Fecha o formulário -->  
</form>
```

Formulário - Elemento: input - type="radio"

O atributo "type" de valor "radio" indica que o input receberá dados a partir da escolha de um valor de uma lista de itens pré-determinados.

```
<!-- Inicia/Abre o formulário -->  
<form> NOPE (not only this)  
  <input type="radio"/>  
<!-- Finaliza/Fecha o formulário -->  
</form>
```

Formulário - Elemento: input - type="radio"

```
1 <!-- Inicia/Abre o formulário -->
2 <form>
3
4 <label for="ser_humano">Ser Humano</label>
5 <input type="radio" value="ser_humano" id="ser_humano" name="tipo_de_ser"/>
6
7 <label for="alienigena">Alienígena</label>
8 <input type="radio" value="alienigena" id="alienigena" name="tipo_de_ser"/>
9
10 <!-- Finaliza/Fecha o formulário -->
11 </form>
```

Formulário - Elemento: input - type="checkbox"

O atributo "type" de valor "checkbox" indica que o input será uma caixa de seleção. Caso mais de uma seja criada, elas serão independentes, podendo ter zero ou mais de itens selecionados.

```
1 <!-- Inicia/Abre o formulário -->
2 <form>
3
4 <label for="ser_humano">Ser Humano</label>
5 <input type="checkbox" value="ser_humano" id="ser_humano" name="tipo_de_ser"/>
6
7 <label for="alienigena">Alienígena</label>
8 <input type="checkbox" value="alienigena" id="alienigena" name="tipo_de_ser"/>
9
10 <!-- Finaliza/Fecha o formulário -->
11 </form>
```


Formulário - Elemento: select

O elemento `<select>` define um seletor que criará uma lista drop-down.

Os itens desse elemento devem ser definidos com a tag `<option>`.

Por padrão, o primeiro elemento é o selecionado ao abrir a página. Caso queira definir um diferente do primeiro, tal opção deverá vir com o atributo "selected".

Formulário - Elemento: select

```
1 ▾ <!-- Inicia/Abre o formulário -->
2 ▾ <form>
3
4 ▾   <select name="filmes">
5 ▾     <option value="iron_man">Iron Man</option>
6 ▾     <option value="deadpool" selected>Deadpool</option>
7 ▾     <option value="guardioes_da_galaxia">Guardiões da Galáxia</option>
8 ▾     <option value="thor">Thor Ragnarok</option>
9   </select>
10
11 ▾ <!-- Finaliza/Fecha o formulário -->
12 </form>
```

Formulário - Elemento: datalist

O elemento `<datalist>` define uma lista pré-definida de opções para um input do tipo texto.

Para relacionar o input ao datalist, deverá ser adicionado ao input o atributo "list", que será de valor igual ao atributo "id" do datalist.

Formulário - Elemento: datalist

```
1 <!-- Inicia/Abre o formulário -->
2 <form>
3
4     <input list="browsers">
5     <datalist id="browsers">
6         <option value="Internet Explorer">
7         <option value="Firefox">
8         <option value="Chrome">
9         <option value="Opera">
10        <option value="Safari">
11    </datalist>
12
13 <!-- Finaliza/Fecha o formulário -->
14 </form>
```

Formulário - Elemento: textarea

O elemento `<textarea>` define uma área multi-linhas para a inserção de texto do usuário.

Ele pode receber os atributos "rows" (para indicar a quantidade de linhas padrão) e "cols" (para indicar a quantidade de colunas padrão).

```
1 <!-- Inicia/Abre o formulário -->
2 <form>
3
4   <textarea name="message" cols="30" rows="10"></textarea>
5
6 <!-- Finaliza/Fecha o formulário -->
7 </form>
```

Formulário - Elemento: input - type="button"

O atributo "type" de valor "button" fará com que o input se transforme em um botão que poderá receber eventos de click (onclick).

```
1 <!-- Inicia/Abre o formulário -->
2 <form>
3
4   <input type="button" onclick="alert('Clicou!!')" value="Clique aqui">
5
6 <!-- Finaliza/Fecha o formulário -->
7 </form>
```

Formulário - Elemento: button

O atributo "type" de valor "button" fará com que o input se transforme em um botão que poderá receber eventos de click (onclick).

```
1 <!-- Inicia/Abre o formulário -->
2 <form>
3
4   <button onclick="alert('Clicou! TOP!')">Clique Aqui</button>
5
6 <!-- Finaliza/Fecha o formulário -->
7 </form>
```

Formulário - Elemento: input - `type="submit"`

O atributo "type" de valor "submit" criará um botão, que será responsável pelo envio dos dados preenchidos no formulário para o manipulador do form.

O manipulador do form é definido pelo atributo "action", na tag de abertura do form. Ele receberá os dados e poderá os processar devidamente.

Formulário - Elemento: input - type="reset"

O atributo "type" de valor "reset" fará com que o input se transforme em um botão que, ao clicado, resetará todos os valores dos campos do formulário.

```
1 ▾ <!-- Inicia/Abre o formulário -->
2 ▾ <form>
3
4   <input type="reset" value="Limpar Campos"/>
5
6 ▾ <!-- Finaliza/Fecha o formulário -->
7   </form>
```

Formulário - Atributo: **method**

O formulário possui um atributo "method", que especifica o método HTTP (GET ou POST) para ser utilizado na apresentação de dados do formulário, ao ser enviado.

```
3 <form action="recebeDadosForm.html" method="get">  
4 ▼ <!-- ou -->  
5 ▼ <form action="recebeDadosForm.html" method="post">
```

Method: GET x POST

GET: quando não há sigilo nos dados, pois os mesmos serão enviados via URL (ficarão visíveis ao usuário).

POST: quando há dados sigilosos, pois serão enviados no corpo da requisição HTTP (não visível ao usuário).

Atributo "name"

O atributo "name" está presente em todos os elementos input.

Eles servem para nomear os campos em uma requisição HTTP (GET ou POST).

Todos os inputs deverão ter este atributo, pois apenas serão enviados na submissão se houver este atributo.

Atributo "value"

O atributo "value" indica o valor pré-determinado de um campo (como por exemplo, elementos "radio").

Caso seja usado em um elemento textual, esse atributo fará com que tal input seja carregado com o valor do "value" como padrão.

Time to code!

Formulário

1. Em <https://github.com/talitaoliveira/frontend-basics/> baixe o arquivo "handler.html" que se encontra em Aula 03/exercicio01 e o coloque na mesma pasta do arquivo deste exercício.
2. Crie um novo documento HTML que conterá um formulário com:
 - a. Um campo para nome (name = "firstName")
 - b. Um campo para sobrenome (name = "lastName")
 - c. Um radio para escolha de sexo (name = "gender")
 - d. Um botão para submissão
3. O formulário deverá ter uma action, que apontará para o arquivo baixado do github no passo (1).
4. Utilize o método GET

Resultado
Esperado

Formulário

Nome

Sobrenome

Sexo

- ☒ Masculino
☐ Feminino
☐ Outro

Enviar

Formulário - Container fieldset

Existem containers para organização de conteúdo de um formulário. Podemos utilizar o elemento `<fieldset>` para agrupar dados relacionados em um formulário, e o elemento `<legend>` para dar uma legenda ao que esse conjunto de dados representa.

```
<form action="" method="get">
  <fieldset>

    <legend>Dados Pessoais</legend>

    <label for="primeiroNome">Primeiro Nome: </label>
    <input type="text" id="primeiroNome" name="primeiroNome" />
    <br>
    <label for="ultimoNome">Ultimo Nome: </label>
    <input type="text" id="ultimoNome" name="ultimoNome" />
    <br>
    <label for="idade">Idade: </label>
    <input type="text" id="idade" name="idade" />

  </fieldset>

  <fieldset>

    <legend>Endereco</legend>

    <label for="endereco">Endereco: </label>
    <input type="text" id="endereco" name="endereco" />
    <br>
    <label for="bairro">Bairro: </label>
    <input type="text" id="bairro" name="bairro" />

  </fieldset>

  <input type="submit" value="Enviar">

</form>
```

Dados Pessoais

Primeiro Nome:

Ultimo Nome:

Idade:

Endereco

Endereco:

Bairro:

Enviar

Entradas HTML5

O HTML5 possui diversos outros novos tipos de entrada pré-definidas, dentre eles:

- color
- date
- datetime-local
- email
- month
- number
- range
- search
- time
- url
- week

Formulário - Elemento: input - `type="color"`

— — —

O atributo "type" de valor "color" serve para tipos de entrada que esperam uma cor. A depender do browser usado, poderá aparecer uma paleta de seleção de cores.

Formulário - Elemento: input - `type="date"`

— — —

O atributo "type" de valor "date" serve para tipos de entrada que esperam uma data. A depender do browser usado, poderá aparecer um campo de seleção de datas.

Podem ser adicionados ao elemento os atributos "max" e "min", que permitirá que uma data máxima possível de escolha seja adicionada (exemplo: "2000-01-02")

Formulário - Elemento: input - `type="datetime-local"`

— — —

O atributo "type" de valor "datetime-local" serve para tipos de entrada que esperam uma data, mas sem fuso horário. A depender do browser usado, poderá aparecer um campo de seleção de datas.

Formulário - Elemento: input - `type="email"`

O atributo "type" de valor "email" serve para tipos de entrada que esperam uma email.

Alguns browsers salvam o e-mail a partir deste tipo para dar como sugestão em futuras inserções. Além disso, alguns dispositivos móveis alteram seu teclado para facilitar o acesso ao "@" e ao ".com"

Formulário - Elemento: input - `type="month"`

— — —

O atributo "type" de valor "month" serve para tipos de entrada que esperam uma data, mas especificamente selecionando apenas o mês do ano. Dependendo do browser utilizado, um campo de seleção de mês poderá aparecer.

Formulário - Elemento: input - `type="number"`

— — —

O atributo "type" de valor "number" serve para tipos de entrada que esperam um valor numérico.

Também podem ser definidos atributos "max" e "min" para este elemento com este tipo, dando um range de números aceitos.

Formulário - Elemento: input - `type="range"`

— — —

O atributo "type" de valor "range" serve para tipos de entrada que esperam um valor numérico, que será selecionado a partir de um controle deslizante.

Também podem ser definidos atributos "max" e "min" para este elemento com este tipo, dando um range de números aceitos.

Formulário - Elemento: input - `type="search"`

— — —

O atributo "type" de valor "search" funciona como um campo textual comum, mas é definido como um campo de pesquisa.

Formulário - Elemento: input - type="time"

— — —

O atributo "type" de valor "time" funciona como um campo que permite ao usuário a escolha de tempo (sem fuso horário).

Dependendo no browser escolhido para uso, poderá aparecer um campo de seleção de tempo.

Formulário - Elemento: input - `type="url"`

— — —

O atributo "type" de valor "url" funciona como um campo que espera receber uma URL.

Dependendo no browser escolhido para uso, a URL inserida pode ser automaticamente validada quando submetida.

Formulário - Elemento: input - type="week"

— — —

O atributo "type" de valor "week" funciona como um campo que espera receber uma semana e um ano

Dependendo no browser escolhido para uso, aparecerá um campo para escolha da semana e do ano.

Desafio

1. Utilizando os conhecimentos adquiridos em HTML, iremos construir um site mais complexo
2. O tema é livre, mas deve conter mais de uma página, mesmo que ainda sem estilo, que represente a elaboração de alguma ideia sua
3. Se baseiem em ideias de construção de outros sites para montar sua própria ideia
4. Utilizaremos o site criado agora futuramente, incrementando com CSS e JS
5. Enviem, por e-mail, dúvidas e o resultado do site, para que haja um acompanhamento da atividade
6. Envios até ...

Sugestão: <https://github.com/estevanmaito/apenas-crie-websites>

Dúvidas?

Contato

— — —



github.com/talitaoliveira



litaa.olivera@gmail.com



linkedin.com/in/litaaoliveira



[@liitacherry](https://twitter.com/liitacherry)

