



## Módulo 2

### Forms

< [thefutureisblue.me](https://thefutureisblue.me) />



# Objetivos da aula

Forms



# O que é um form?

O elemento HTML `<form>` representa uma seção de um documento que contém controles interativos que permitem ao usuário submeter informação a um determinado servidor web.

# O que é um **form**?

Os elementos de formulário são tipos diferentes de entrada, como campos de texto, caixas de seleção, botões de opção, botões de envio, entre outros.

# Criando um form

Para criar um novo form, basta utilizarmos a tag `<form>` no seu inicio, e a tag `</form>` ao seu fim.

Entre essas tags, devem haver a componentização do que deve ser preenchido no devido formulário.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Página de exemplo - Blue</title>
  </head>
  <body>
    <p>Abaixo terá um formulário:</p>
    <form>
      <p>Aqui devem ser definidos os componentes do formulário</p>
    </form>
  </body>
</html>
```





# Componente: **input**

Um dos principais componentes de um formulário é o componente de entrada `<input>`.

Ele pode ter vários tipos diferentes dependendo de seu uso, como: text, radio, checkbox, password, date, entre outros.

Para definir um tipo, basta utilizar o atributo **type**.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Página de exemplo - Blue</title>
  </head>
  <body>
    <p>Abaixo terá um formulário:</p>
    <form>
      <input type="text" />
      <input type="radio" />
      <input type="checkbox" />
      <input type="password" />
      <input type="date" />
    </form>
  </body>
</html>
```



# Componente: **input**

Uma das informações mais importantes de um componente do tipo `<input>` é o atributo `name`.

O atributo `name` é utilizado para nomear o dado que será referenciado o `input`. É por meio desse nome que o servidor conseguira identificar qual informação ele representa.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Página de exemplo - Blue</title>
  </head>
  <body>
    <p>Abaixo terá um formulário:</p>
    <form>
      <input type="text" name="input-text" />
      <input type="radio" name="input-radio" />
      <input type="checkbox" name="input-checkbox" />
      <input type="password" name="input-password" />
      <input type="date" name="input-date" />
    </form>
  </body>
</html>
```





# Componente: **input**

Alguns tipos de <input> requerem que seja adicionado um valor a eles, já que seus valores não são editáveis pelo usuário, como é o caso dos tipos: **radio** e **checkbox**.

Esses valores devem ser adicionados no atributo **valor**.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Página de exemplo - Blue</title>
  </head>
  <body>
    <p>Abaixo terá um formulário:</p>
    <form>
      <input type="text" name="input-text" />
      <input type="radio" name="input-radio" />
      <input type="checkbox" name="input-checkbox" />
      <input type="password" name="input-password" />
      <input type="date" name="input-date" />
    </form>
  </body>
</html>
```



# radio ou checkbox ?

valor 1 ☒ valor 2 ☐ valor 3 ☐

valor 1 ☐ valor 2 ☐ valor 3 ☐

Inputs do tipo radio são utilizados quando é necessário escolher uma entre várias opções.

Nesse caso, todos os inputs para uma determinada opção devem ter os mesmos nomes, porém seus valores devem ser diferentes.

Inputs do tipo checkbox são utilizados quando várias opções são selecionáveis.

# Componente: label

O componente `<label>` é utilizado para definir um texto que referencie um input.

É utilizado o atributo `for` que deve indicar o `id` do componente input que está sendo referenciado.

Com a configuração correta, ao usuário clicar no texto do label, o foco será transferido para o input referenciado.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Página de exemplo - Blue</title>
  </head>
  <body>
    <p>Formulário de cadastro:</p>
    <form>
      <div>
        <label for="nome">Nome: </label>
        <input type="text" id="nome" name="nome" />
      </div>
      <div>
        <label for="h">Homem</label>
        <input type="radio" id="h" value="h" name="sexo" />
        <label for="m">Mulher</label>
        <input type="radio" id="m" value="m" name="sexo" />
      </div>
      <div>
        <label for="nascimento">Nascimento</label>
        <input type="date" id="nascimento" name="nascimento" />
      </div>
      <div>
        <label for="opt1">Desejo receber sms</label>
        <input type="checkbox" id="opt1" name="options" />
      </div>
      <div>
        <label for="opt2">Desejo receber email</label>
        <input type="checkbox" id="opt2" name="options" />
      </div>
      <div>
        <label for="senha">Senha</label>
        <input type="password" id="senha" name="senha" />
      </div>
    </form>
  </body>
</html>
```

# Componente: **select...option**

O componente `<select>` é utilizado para trazer uma lista de opções para que o usuário possa efetuar uma seleção.

Cada opção deve ser inserida em uma `<option>` e o valor selecionado é o que será enviado pelo formulário.

A primeira opção é definida automaticamente para o formulário, então é muito comum ver ela vazia, forçando uma seleção.

```
<div>
  <label for="cidade">Cidade</label>
  <select id="cidade" name="cidade">
    <option></option>
    <option>Curitiba</option>
    <option>São Paulo</option>
    <option>Porto Alegre</option>
    <option>Recife</option>
    <option>Alagoas</option>
  </select>
</div>
```





# Componente: **select...option**

Também é possível criar agrupamentos de seleções, categorizando-as para o usuário.

Para isso, basta utilizar o elemento `<optgroup>` com a label referente ao agrupamento.

```
<label for="cidade">Cidade</label>
<select id="cidade" name="cidade">
  <option></option>
  <optgroup label="Sul">
    <option>Curitiba</option>
    <option>Porto Alegre</option>
  </optgroup>
  <optgroup label="Sudeste">
    <option>São Paulo</option>
    <option>Rio de Janeiro</option>
  </optgroup>
  <optgroup label="Nordeste">
    <option>Recife</option>
    <option>Alagoas</option>
  </optgroup>
</select>
```



# Input type submit

O input type Submit gera um botão em tela que efetuará a ação de envio do formulário. O valor definido é o valor que será apresentado pelo botão.

```
<input type="submit" value="Enviar formulário!"/>
```

O tipo de envio e para onde será enviada as informações deve ser estipulada pelo <form> por meio dos atributos action e type.

```
<form action="" method="GET">
```

A action representa para qual endereço a informação será enviada e o método representa qual tipo de protocolo HTTP será utilizado para enviar a informação.



# Funcionamento form



Quando um formulário é submetido pelo usuário, ele envia as informações ao endereço especificado utilizando o **protocolo HTTP**, podendo utilizar os **métodos GET ou POST**.

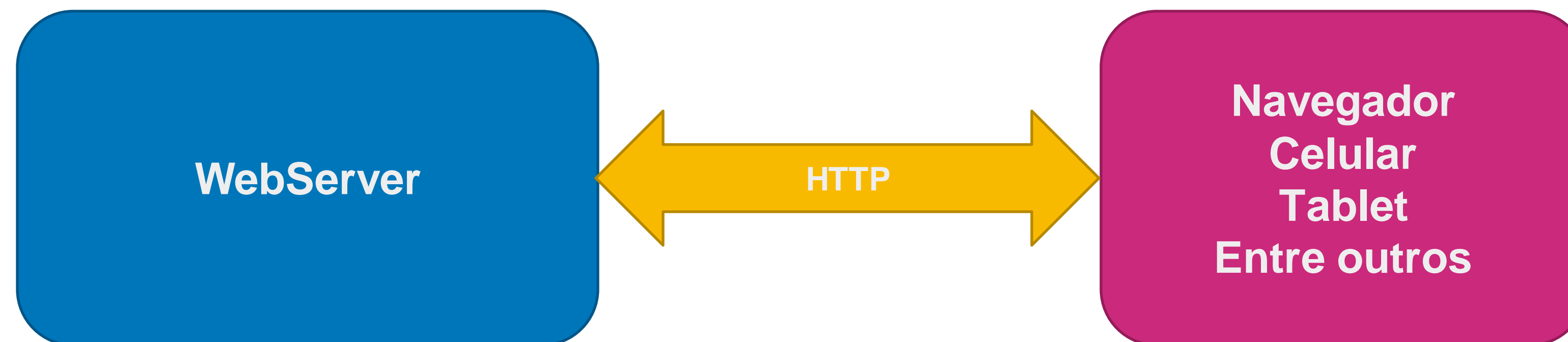


# O que é HTTP?

*Hypertext Transfer Protocol* (HTTP) é um protocolo utilizado para transmissão de documentos hipermídia.

Foi desenvolvido para comunicação síncrona entre navegadores web e servidores web porém, pode ser utilizado para outros propósitos.

É também um protocolo sem estado ou *stateless protocol*, que significa que o servidor não mantém nenhum dado entre suas requisições.



# Métodos HTTP?

Existem inúmeros métodos diferentes HTTP que podem ser utilizados para uma comunicação.


Porém, quando falamos em formulários, os únicos métodos utilizados são o POST e o GET.

# Exemplo GET

Formulário de cadastro:

Nome:


Homem ☒ Mulher ☐

Nascimento  

Desejo receber sms ☒

Desejo receber email ☒

Senha

Cidade  

*As informações transmitidas pelo método GET são passadas diretamente pela URL referente ao método action do formulário.*

*Atenção: Se o método action estiver vazio, o envio será para o endereço atual.*

```
| C:/dev/frontend/pagina.html?nome=João&sexo=h&nascimento=2021-04-07&options=on&options=on&senha=123123&ci...
```






# Exemplo POST

Formulário de cadastro:

Nome:


Homem ☒ Mulher ☐

Nascimento  

Desejo receber sms ☒

Desejo receber email ☒

Senha

Cidade  

*As informações transmitidas pelo método POST são passadas pelo corpo da requisição HTTP.*

*Por não poder ser visto pela URL, ele é um pouco mais seguro, mas ainda assim é possível visualizar os dados da requisição pelo navegador.*



# Exemplo POST

Name

pagina.html

1 solicitações 2.2 KB transferidos 2

× Cabeçalhos Visualizar Resposta Iniciador Temporização

Política de referenciador: strict-origin-when-cross-origin

▼ Cabeçalhos de Solicitação

⚠ Cabeçalhos provisórios são mostrados

Content-Type: application/x-www-form-urlencoded

Origin: null

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4398.96 Safari/537.36

▼ Dados de formulário Exibir código-fonte URL de exibição codificada

nome: João

sexo: h

nascimento: 2021-04-07

options: on

options: on

senha: 123123

cidade: Curitiba



# Configurações extra

Definir um elemento utilizando a opção **disabled**, faz com que ele não seja editável.

```
<label for="nome">Nome: </label>  
<input type="text" id="nome" name="nome" disabled/>
```

Nome:



# Configurações extra

Definir um elemento utilizando a opção **placeholder**, faz com que um texto apareça quando um **input text** estiver vazio.

```
<label for="nome">Nome: </label>  
<input type="text" id="nome" name="nome" placeholder="Informe um nome..."/>
```

Nome:




# Atenção!


Navegadores atuais utilizam vários tipos diferentes de input, mas lembre-se que sua aplicação deve funcionar na maioria dos navegadores, **incluindo os antigos!**

Fora que alguns podem ser bem *bugados*...

# Atenção!

```
<form>  
  <input type="week">  
</form>
```

Semana --, ---- 

abril de 2021  

Semana	D	S	T	Q	Q	S	S
13	28	29	30	31	1	2	3
14	4	5	6	7	8	9	10
15	11	12	13	14	15	16	17
16	18	19	20	21	22	23	24
17	25	26	27	28	29	30	1
18	2	3	4	5	6	7	8

Esta semana

# Atenção!

Sempre teste seu layout em vários navegadores diferentes, para não ter surpresas inesperadas.





# Vamos praticar alguns exemplos!



**Botando para  
rodar!**

Por hoje é só!  
Obrigado! =)