[https://devdocs.io/](https://devdocs.io/" \t "_blank)

<https://vargasleticia.notion.site/Frontend-Est-tico-Atividades-Aula-3-96a4679dfdd84138a3468d41c9b55a3e?pvs=4>

<https://www.w3schools.com/tags/att_input_type.asp>

<https://www.w3schools.com/tags/tag_select.asp>

**checkbox**

<fieldset>

<legend>Choose your monster's features:</legend>

<div>

<input type="checkbox" id="scales" name="scales" checked />

<label for="scales">Scales</label>

</div>

<div>

<input type="checkbox" id="horns" name="horns" />

<label for="horns">Horns</label>

</div>

</fieldset>

o label pode envelopar o input e assim você não precisa colocar o id e o for. Você usa o name pra fazer o link.

Action é o destino pra onde irá o formulário.

**Separando os dados (**Method get)

<form action=”reponjs.html”>

Split separa

const urlParams = window.location.search

const separa = urlParams.split(‘&’)

for(item of separa){ item.split(‘=’)}

um solucao:

const params = Object.fromEntries(new URLSearchParams(window.location.search.slice(1)));

**Uso do POST** – algo mais seguro e deve ser o utilizado.

Colocamos aqui o end point

<form action=[http://](http://servidor)localhost:3000/form method=”post”>

Servidor NODE pra fazer o envio das informações.

Json server - “fake api”

Precisa criar um .arquivo json ex: db.json ; esse

servidor

Comando no node:

json-server db.json

Depois do servidor colocamos o nome que

“form”: [

]

**Instalação da Fake API - Json Server**

<https://www.npmjs.com/package/json-server>

* Pré-requisitos
  + NodeJS 12+ >> <https://nodejs.org/pt-br>
  + NPM >> <https://rockcontent.com/br/blog/npm/>

Vamos instalar de forma global no computador, utilize o seguinte comando no cmd

npm install -g json-server

Após a instalação podemos criar estruturas JSON que irão simular o backend.

Para rodar o servidor utilizamos o seguinte comando (na pasta onde se encontra o a estrutura JSON criada)

**### Input text + label + for + propriedade "id”**

A relação entre os elementos HTML `input` (tipo `text`), `label`, os atributos `for` e `id` é uma parte importante da acessibilidade e usabilidade de formulários web. Vamos analisar cada elemento e atributo e explicar como eles se relacionam:

1. \*\*`<input type="text">`:\*\* O elemento `<input>` com `type="text"` é usado para criar um campo de entrada de texto em um formulário web. Isso permite que os usuários insiram texto, como nomes, endereços de e-mail, mensagens, etc.

2. \*\*`<label>`:\*\* O elemento `<label>` é usado para rotular um campo de entrada em um formulário. O rótulo fornece uma descrição ou identificação do campo, tornando-o mais acessível e compreensível para os usuários.

3. \*\*`for` e `id`:\*\* O atributo `for` em um elemento `<label>` e o atributo `id` em um elemento `<input>` estão relacionados. Através da propriedade `for`, um rótulo é associado a um campo de entrada específico. O valor do atributo `for` deve corresponder ao valor do atributo `id` do campo de entrada que o rótulo está rotulando.

\*\*Exemplo de Uso:\*\*

```html

<label for="nome">Nome:</label>

<input type="text" id="nome" name="nome" required>

```

Neste exemplo, o rótulo "Nome:" está associado ao campo de entrada de texto. O atributo `for` do `<label>` tem o valor "nome", que corresponde ao atributo `id` do campo de entrada `<input>`. Essa associação entre o rótulo e o campo de entrada permite que os usuários cliquem no rótulo para ativar o campo de entrada correspondente, tornando a interação mais fácil e acessível.

Em resumo, a relação entre `input` (tipo `text`), `label`, `for` e `id` é usada para criar rótulos descritivos para campos de entrada em formulários e melhorar a usabilidade e acessibilidade, garantindo que os rótulos estejam associados aos campos corretos. Isso também é útil para dispositivos de auxílio, como leitores de tela, que podem ler os rótulos para os usuários, facilitando a compreensão do formulário.

**Exemplo de inputs**

<form>

<div>

<label for="text">Texto:</label>

<input type="text" id="text" name="text" />

</div>

<div>

<label for="password">Senha:</label>

<input type="password" id="password" name="password" />

</div>

<div>

<label for="email">Email:</label>

<input type="email" id="email" name="email" />

</div>

<div>

<label for="number">Número:</label>

<input type="number" id="number" name="number" />

</div>

<div>

<label for="tel">Telefone:</label>

<input type="tel" id="tel" name="tel" placeholder="(12)12345-6789" pattern="[0-9]{2}[0-9]{5}-[0-9]{4}" />

</div>

<div>

<label for="date">Data:</label>

<input type="date" id="date" name="date" />

</div>

<div>

<label for="checkbox">Checkbox:</label>

<input type="checkbox" id="checkbox" name="checkbox" />

</div>

<div>

<label for="radio">Radio:</label>

<input type="radio" id="radio" name="radio" />

</div>

<div>

<label for="file">Arquivo:</label>

<input type="file" id="file" name="file" />

</div>

<div>

<label for="range">Intervalo:</label>

<input type="range" id="range" name="range" />

</div>

<div>

<label for="color">Cor:</label>

<input type="color" id="color" name="color" />

</div>

<div>

<button type="submit" >Enviar</button>

</div>

</form>