

01

Filtrando os pacientes da tabela

Transcrição

Continuando com a construção do site da Aparecida Nutricionista, já conseguimos implementar a adição e remoção dos pacientes. Seria interessante se disponibilizássemos um filtro, com o qual a busca fosse realizada usando-se algum critério. Por exemplo, se quiséssemos acessar os dados de um determinado paciente, bastaria digitarmos seu nome em um campo de pesquisa e, então, apenas os dados do mesmo seriam listados. A tendência de uma tabela de dados é que a lista aumente cada vez mais, e caso se queira utilizar algum tipo de filtro, queremos que ela possa fazer uma busca mais específica.

Começaremos implementando um campo de busca na tabela, adicionando uma tag `<input>` no `index.html`. Criaremos uma `label` com o texto `Filtre:`. Dentro da tag, teremos também `placeholder`, com o texto `Digite o nome do paciente`.

```
<section class="container">
  <h2>Meus pacientes</h2>
  <label for="filtrar-tabela">Filtre:</label>
  <input type="text" name="filtro" id="filtrar-tabela" placeholder="Digite o nome do paciente">
  <table>
    <thead>
      <tr>
        <th>Nome</th>
        <th>Peso(kg)</th>
        <th>Altura(m)</th>
        <th>Gordura Corporal(%)</th>
        <th>IMC</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>Paulo</td>
        <td>100</td>
        <td>-2.00</td>
        <td>10</td>
        <td>Altura inválida!</td>
      </tr>
      <tr>
        <td>João</td>
        <td>80</td>
        <td>1.72</td>
        <td>40</td>
        <td>27.04</td>
      </tr>
      <tr>
        <td>Erica</td>
        <td>54</td>
        <td>1.64</td>
        <td>14</td>
        <td>20.08</td>
      </tr>
      <tr>
        <td>Douglas</td>
        <td>85</td>
        <td>1.73</td>
        <td>24</td>
        <td>28.40</td>
      </tr>
      <tr>
        <td>Tatiana</td>
        <td>48</td>
        <td>1.55</td>
        <td>19</td>
        <td>19.98</td>
      </tr>
    </tbody>
  </table>
</section>
```

Após a alteração, será exibido um campo com o filtro na tela:



The screenshot shows a web browser window with the following content:

- Address Bar:** file:///Users/Caelum/Documents/introducao-javascript/index.html
- Page Title:** Aparecida Nutricionista
- Section Title:** Meus pacientes
- Search Input:** A text input field labeled "Filtre:" with the placeholder "Digite o nome do paciente". A red arrow points to this input field.
- Table:** A table with the following data:

Nome	Peso(kg)	Altura(m)	Gordura Corporal(%)	IMC
Paulo	100	-2.00	10	Altura inválida!
João	80	1.72	40	27.04
Erica	54	1.64	14	20.08
Douglas	85	1.73	24	28.40
Tatiana	48	1.55	19	19.98

Faremos melhorias também na parte de estilo, e adicionaremos o seguinte código no `index.css`:

```
#filtrar-tabela {
  width: 200px;
  height: 35px;
  margin-bottom: 10px;
}
```

Nome	Peso(kg)	Altura(m)	Gordura Corporal(%)	IMC
Paulo	100	-2.00	10	Altura inválida!
João	80	1.72	40	27.04
Erica	54	1.64	14	20.08
Douglas	85	1.73	24	28.40
Tatiana	48	1.55	19	19.98

Acessando o conteúdo digitado no campo de filtragem

Resta agora implementarmos o código JavaScript que fará essa filtragem. Vamos criar o arquivo `filtra.js`, na pasta `js`. Como é um arquivo novo, vamos importá-lo em `index.html`:

```
<script src="js/calcula-imc.js" ></script>
<script src="js/form.js" ></script>
<script src="js/remover-paciente.js" ></script>
<script src="js/filtra.js" ></script>
```

Nosso objetivo é que conforme o nome for digitado no campo, os pacientes relacionados à busca sejam listados. Por exemplo, ao digitarmos **Doug**, o paciente **Douglas** deverá ser exibido. O conteúdo de texto que está sendo digitado no campo de busca deverá ser pego.

Só que quando queremos filtrar? Conforme o usuário vai digitando, a cada letra digitada. Ou seja, queremos detectar o evento de **digitar**, conhecido como **input de dados**. Toda vez que alguém inserir dados no campo, esse evento será disparado.

Inicialmente, selecionaremos o campo de texto no `filtra.js`:

```
var campoFiltro = document.querySelector("#filtrar-tabela");
console.log(campoFiltro);
```

Testando no browser, veremos que estamos conseguindo selecioná-lo. E ao clicarmos no campo, veremos o seguinte código no console:

```
<input type="text" name="filtro" id="filtrar-tabela" placeholder="Digite o nome do paciente">
```

Queremos que ao começarmos a digitar no filtro, seja iniciado o processo de filtragem, e para isto, adicionaremos um escutador de evento. Com o `addEventListener()` escutaremos um evento de `input`. Passaremos como segundo parâmetro uma função com a ação a ser executada quando alguém clicar no campo.

```
var campoFiltro = document.querySelector("#filtrar-tabela");

campoFiltro.addEventListener("input", function(){
  console.log("Digitaram no campo");
});
```

The screenshot shows a web page titled "Meus pacientes" with a table of patient data. A search input field above the table contains the text "Douglas". The browser's developer tools are open, showing the console with the message "Digitaram no campo" and the element "Altura inválida!" highlighted with a red box.

Nome	Peso(kg)	Altura(m)	Gordura Corporal(%)	IMC
Paulo	100	-2.00	10	Altura inválida!
João	80	1.72	40	27.04
Erica	54	1.64	14	20.08
Douglas	85	1.73	24	28.40
Tatiana	48	1.55	19	19.98

A cada letra que inserimos, a função é chamada, o `value` do campo é filtrado na tabela. Poderemos utilizar `campoFiltro.value` dentro da função, no entanto, `campoFiltro` é o **dono** do evento. Utilizaremos a palavra de contexto `this`, referente ao próprio dono do evento.

```
var campoFiltro = document.querySelector("#filtrar-tabela");

campoFiltro.addEventListener("input", function(){
  console.log(this.value);
});
```

Conforme o nome é digitado, podemos ir comparando com os de todos os pacientes da tabela com o `querySelectAll()`. Exibiremos aqueles que forem iguais, e os que forem diferentes serão escondidos. Vamos, então, selecionar todos os pacientes:

```
var campoFiltro = document.querySelector("#filtrar-tabela");

campoFiltro.addEventListener("input", function() {
  console.log(this.value);
  var pacientes = document.querySelectorAll(".paciente");
});
```

No entanto, queremos fazer a comparação com o nome dos pacientes, não com a `tr`. Precisaremos iterar sobre os pacientes, para então acessarmos o nome de cada um:

```
var campoFiltro = document.querySelector("#filtrar-tabela");

campoFiltro.addEventListener("input", function() {
  console.log(this.value);
  var pacientes = document.querySelectorAll(".paciente");
  for (var i = 0; i < pacientes.length; i++) {
    var paciente = pacientes[i];
  }
});
```

Agora, selecionaremos o paciente, e a partir dele a `td` com a classe `info-nome`. Daí, extrairemos o nome do paciente:

```
var campoFiltro = document.querySelector("#filtrar-tabela");

campoFiltro.addEventListener("input", function() {
  var pacientes = document.querySelectorAll(".paciente");

  for (var i = 0; i < pacientes.length; i++) {
    var paciente = pacientes[i];
    var tdNome = paciente.querySelector(".info-nome");
    var nome = tdNome.textContent;
  }
});
```

A partir de então temos acesso ao nome de todos os pacientes da tabela, e também ao conteúdo de texto do campo de filtragem, só precisaremos esconder todos os pacientes que são diferentes do conteúdo de texto, e mostrar os que são iguais.

Veremos isso mais adiante.