

06

Mão na massa: filtrando a tabela

Começando deste ponto ?

Começando deste ponto? Você pode fazer o [download \(<https://s3.amazonaws.com/caelum-online-public/introducao-javascript/stages/introducao-javascript-capitulo-8.zip>\)](https://s3.amazonaws.com/caelum-online-public/introducao-javascript/stages/introducao-javascript-capitulo-8.zip) completo do projeto do capítulo anterior e continuar seus estudos a partir deste capítulo.

Neste capítulo nós vamos implementar um campo de texto para filtrar os pacientes de nossa tabela pelo nome. Vamos lá:

1- Comece criando o `input` e `label` do campo aonde usuário irá digitar. No seu `index.html`, entre o título "Meus Pacientes" e a tabela, adicione:

```
//index.html
...
<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>
...

```



2- Vamos aproveitar e estilizar um pouco este campo, para que ele fique com uma letra um pouco maior. No seu `estilos.css` adicione o estilo abaixo:

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

3- Agora vamos para o Javascript, como se trata de uma nova funcionalidade, vamos criar um novo arquivo na pasta `/js` chamado `filtra.js`:

```
introducao-javascript
└── js
    ├── calcula-imc.js
    ├── form.js
    ├── remover-paciente.js
    └── filtra.js
```

4- E vamos importá-lo em nosso HTML ao fim da tab `<body>`:

```
</section>
```

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

5- Vamos começar selecionando o campo de input, pois temos o interesse de pegar o seu valor para realizar a busca:

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

6- Vamos adicionar um novo evento nele, o evento de `input` que é responsável por detectar quando o usuário começar a digitar no campo:

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

campoFiltro.addEventListener("input", function() {

});
```

7- Para realizar a busca, vamos utilizar a seguinte técnica: Toda vez que o usuário começar a digitar alguma coisa, vamos esconder todos os itens da tabela, e só exibir aquele que batem com a busca. E quando o usuário deixar o campo em branco, vamos voltar a exibir os itens da tabela. Para esconder os itens, vamos criar uma nova classe chamada de `invisivel` no `estilos.css`:

```
//estilos.css
....
.invisivel {
    display: none;
}
```

Um simples `display:none` irá ocultar as linhas da tabela. Vamos aplicá-la quando usuário digitar algo.

8- Sabemos que o que foi digitado pelo usuário pode ser acessado através do `this.value`, afinal o `this` neste caso é o `<input>` e podemos ter acesso ao seu conteúdo com a propriedade `.value`. Se detectarmos algo escrito no campo, vamos esconder todos os pacientes da tabela colocando a classe `.invisivel`, caso contrário, vamos remover a classe `invisivel` e exibi-los novamente:

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

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

    if (this.value.length > 0) {
        for (var i = 0; i < pacientes.length; i++) {
            var paciente = pacientes[i];
            paciente.classList.add("invisivel");
        }
    } else {
```

```
for (var i = 0; i < pacientes.length; i++) {  
    var paciente = pacientes[i];  
    paciente.classList.remove("invisivel");  
}  
}  
});
```

Experimente digitar algo na tabela, verá que todos os pacientes somem, mas ao deixar o campo em branco, ou seja, com o `.length` de tamanho zero, os pacientes voltam a aparecer.