

## Removendo pacientes

### Transcrição

Agora que já estamos validando totalmente os nossos pacientes, podemos nos concentrar em adicionar novas funcionalidades ao site da Aparecida.

Ela pode adicionar um paciente com algum dado errado, sem querer, faltando alguma letra no nome, ou algum número no peso, por exemplo. O paciente passará pela validação, mas com os dados incorretos. E ela não conseguirá editá-lo nem removê-lo, para adicioná-lo novamente.

Então vamos implementar a opção de remoção de pacientes para a Aparecida. Ela pode ser feita através de um ícone, ou para ser mais simples, um duplo clique! Ao clicarmos duas vezes em um paciente, o mesmo é removido.

### Escutando o evento de duplo clique nas linhas

Se queremos colocar um evento de duplo clique nas linhas, devemos escutar esse evento para cada linha da tabela, então vamos fazê-lo no JavaScript.

Como estamos criando uma funcionalidade nova, em nada relacionado com o formulário ou com o cálculo de IMC, vamos criar um novo arquivo na pasta **js**, o **remover-paciente.js**. É nesse arquivo que ficará toda a lógica de remoção.

Antes de mais nada, vamos importar esse arquivo no **index.html**:

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

Primeiramente, devemos selecionar as linhas. Todas elas possuem a classe **paciente**, portanto iremos selecioná-las:

```
var pacientes = document.querySelectorAll(".paciente");
```

Agora, para cada linha e paciente, faremos algo:

```
var pacientes = document.querySelectorAll(".paciente");

pacientes.forEach(function(paciente) {

});
```

E para cada paciente, adicionaremos um escutador de eventos para escutar o evento de duplo clique, o **dblclick**:

```
var pacientes = document.querySelectorAll(".paciente");

pacientes.forEach(function(paciente) {
  paciente.addEventListener("dblclick", function() {
```

```
});  
});
```

Agora falta sabermos como removeremos um elemento do DOM com JavaScript.

## Removendo um elemento do HTML com JavaScript

A função do JavaScript responsável por remover um elemento do HTML é a `remove()`. Então temos que chamá-la no paciente que foi clicado.

Quem será clicado? O dono do evento, certo? Este é que sofrerá a ação de duplo clique e executará a função. Para acessar o dono do evento, em que o evento está atrelado, utilizaremos uma palavra reservada do JavaScript chamada `this`:

```
var pacientes = document.querySelectorAll(".paciente");  
  
pacientes.forEach(function(paciente) {  
  paciente.addEventListener("dblclick", function() {  
    this.remove();  
  });  
});
```

O `this` é uma palavra de contexto ligada com quem acionou o evento, a quem o evento está atrelado. Como o evento está atrelado ao paciente, o `this` fará referência a ele.

Ao testarmos na página, e darmos um duplo clique em um paciente nativo, o mesmo é removido! Então conseguimos implementar a remoção de um paciente de um jeito bem fácil.

Mas nem tudo é perfeito, e essa implementação tem um pequeno defeito: ao adicionarmos um paciente pelo formulário, não conseguimos removê-lo. Isso porque o novo paciente não estava na página no momento em que o **remover-paciente.js** foi carregado (lembrando que o navegador abre a página e vai lendo o seu HTML, carregando os nossos scripts logo depois).

Então, o script seleciona somente os pacientes que já estão na página. Ao adicionar um ou mais pacientes, eles não estão escutando o evento! No momento em que um paciente for adicionado, deveríamos colocá-lo para ouvir o evento e executar o mesmo código, mas aí estaríamos duplicando código, que já sabemos não ser uma boa prática.

Veremos a seguir um outro jeito de removermos os pacientes nativos e aqueles que forem sendo adicionados depois da página ser carregada.