

## Evitando o comportamento padrão de um evento

### Transcrição

O comportamento padrão do botão que se encontra no formulário é limpar os dados preenchidos nos campos, recarregar a página e enviar os dados. Nós queremos evitar esse comportamento padrão.

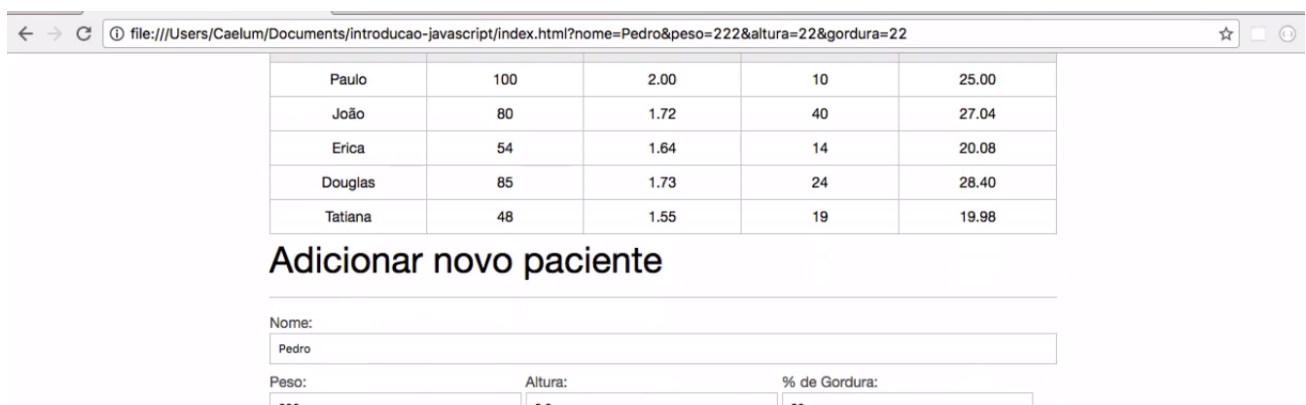
Para isso, devemos chamar uma função característica dos eventos, que faz com que o evento não faça seu comportamento padrão. Desta forma, ele não executará este comportamento e reagirá conforme o que pedimos no evento, no caso, a exibição de uma mensagem. Implementaremos a função `event.preventDefault()` :

```
var botaoAdicionar = document.querySelector("#adicionar-paciente");
botaoAdicionar.addEventListener("click", function(){
    event.preventDefault();
});
```

Essa função faz exatamente o que seu nome diz, prevenindo o comportamento padrão do evento. Mas para utilizá-la, receberemos como segundo parâmetro a função anônima, o `event` :

```
var botaoAdicionar = document.querySelector("#adicionar-paciente");
botaoAdicionar.addEventListener("click", function(event) {
    event.preventDefault();
    console.log("Oi eu sou o botao e fui clicado");
});
```

Em seguida, testaremos novamente a página, preenchendo o formulário e clicando no botão "Adicionar". Podemos reparar em duas coisas: a página não foi recarregada e o formulário não foi limpo. Além disso, conseguimos ver a mensagem no console.



Nome	Peso	Altura	% de Gordura	Índice de Massa Corporal
Paulo	100	2.00	10	25.00
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

### Adicionar novo paciente

Nome:

Peso:  Altura:  % de Gordura:

A função `event.preventDefault()` conseguiu evitar o comportamento padrão do formulário.

Conseguimos detectar o evento de clique no botão (e no título), em seguida, implementaremos o que queremos de fato: adicionar os dados do paciente na tabela. Veremos como fazer isso com JavaScript mais adiante!