

Adicionando pacientes na tabela

Transcrição

Para adicionarmos o paciente na tabela, devemos capturar seus dados do formulário. Para isto, sabemos que devemos trazê-los do mundo HTML para o do JavaScript, selecionando-os.

Vamos adicionar um *id* ao formulário:

```
<form id="form-adiciona">
  <!-- restante do formulário omitido -->
</form>
```

Agora conseguimos selecioná-lo no mundo JavaScript:

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

  var form = document.querySelector("#form-adiciona");
});
```

Através do formulário, temos acesso aos seus *inputs*, acessando através da sua propriedade `name`. É possível vermos no HTML que todos os *inputs* possuem uma propriedade `name`, bastando utilizá-lo para termos acesso ao *input*:

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

  var form = document.querySelector("#form-adiciona");

  var nome = form.nome;

});
```

Mas não estamos interessados no campo em si, no `input`, e sim no seu valor. Para acessar o valor de um campo em JavaScript, utilizaremos a propriedade `value`:

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

  var form = document.querySelector("#form-adiciona");

  var nome = form.nome.value;

});
```

Sabendo disso, poderemos capturar o valor do restante dos campos:

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

    var form = document.querySelector("#form-adiciona");

    var nome = form.nome.value;
    var peso = form.peso.value;
    var altura = form.altura.value;
    var gordura = form.gordura.value;

});
```

Com isso, conseguimos trazer os dados do formulário do HTML para o JavaScript, e adicionar o paciente na tabela.

Criando um elemento HTML com JavaScript

Com os dados em mãos, como vamos fazer para "criar" um paciente? Até agora vimos como trazer algo do mundo HTML para o mundo JavaScript, mas agora é o contrário: temos os dados no mundo JavaScript e queremos criar uma nova linha na tabela com eles, que é no mundo HTML. Ou seja, queremos **criar um elemento HTML** utilizando JavaScript.

Para isso, vamos utilizar a função `createElement`, que como o próprio nome indica, cria um elemento HTML. Poderemos olhar o HTML da tabela e ver que se queremos criar um novo paciente, devemos criar uma nova `tr`, que por sua vez possui várias `td`s, cada uma com o seu dado.

Logo, se queremos criar um paciente, vamos criar uma nova `tr`. Para criarmos um elemento, precisaremos informar onde queremos criar, isto é, em `document`:

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

    var form = document.querySelector("#form-adiciona");

    var nome = form.nome.value;
    var peso = form.peso.value;
    var altura = form.altura.value;
    var gordura = form.gordura.value;

    var pacienteTr = document.createElement("tr");

});
```

Agora que sabemos criar um elemento, podemos criar as `td`s:

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

```
var form = document.querySelector("#form-adiciona");

var nome = form.nome.value;
var peso = form.peso.value;
var altura = form.altura.value;
var gordura = form.gordura.value;

var pacienteTr = document.createElement("tr");

var nomeTd = document.createElement("td");
var pesoTd = document.createElement("td");
var alturaTd = document.createElement("td");
var gorduraTd = document.createElement("td");
var imcTd = document.createElement("td");

});
```

Só que criamos seis elementos distintos, seria como se tivéssemos criado um HTML assim:

```
<tr></tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
```

Sabemos que temos que colocar os dados dentro das `td`s, e depois pegá-las e colocá-las na `tr`. Vamos por passos, primeiro colocando os dados dentro das `td`s, que já sabemos como é feito.

Lembra quando alteramos o conteúdo de texto do título? Utilizamos a propriedade `textContent`, passando para ela o novo dado. Pois aqui é a mesma coisa, vamos alterar o conteúdo de texto das `td`s, passando os seus respectivos valores, capturados do formulário:

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

    var form = document.querySelector("#form-adiciona");

    var nome = form.nome.value;
    var peso = form.peso.value;
    var altura = form.altura.value;
    var gordura = form.gordura.value;

    var pacienteTr = document.createElement("tr");

    var nomeTd = document.createElement("td");
    var pesoTd = document.createElement("td");
    var alturaTd = document.createElement("td");
    var gorduraTd = document.createElement("td");
    var imcTd = document.createElement("td");

    nomeTd.textContent = nome;
```

```
    pesoTd.textContent = peso;
    alturaTd.textContent = altura;
    gorduraTd.textContent = gordura;

  });
```

Utilizando o JavaScript para adicionar um elemento dentro de outro

Agora devemos adicionar cada `td` dentro da `tr`. No JavaScript existe uma função chamada `appendChild`. Ela adiciona como filho o elemento passado a ela como parâmetro. Logo, vamos chamar essa função na `tr`, passando a `td` como parâmetro, uma por uma:

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

  var form = document.querySelector("#form-adiciona");

  var nome = form.nome.value;
  var peso = form.peso.value;
  var altura = form.altura.value;
  var gordura = form.gordura.value;

  var pacienteTr = document.createElement("tr");

  var nomeTd = document.createElement("td");
  var pesoTd = document.createElement("td");
  var alturaTd = document.createElement("td");
  var gorduraTd = document.createElement("td");
  var imcTd = document.createElement("td");

  nomeTd.textContent = nome;
  pesoTd.textContent = peso;
  alturaTd.textContent = altura;
  gorduraTd.textContent = gordura;

  pacienteTr.appendChild(nomeTd);
  pacienteTr.appendChild(pesoTd);
  pacienteTr.appendChild(alturaTd);
  pacienteTr.appendChild(gorduraTd);

});
```

Agora já temos algo mais próximo do que é um paciente, e falta colocarmos ele dentro da tabela. Para isto, vamos selecionar a tabela, o `tbody` do HTML através do seu `id`, e utilizar a função `appendChild` para adicionar a `tr` como sua filha:

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

  var form = document.querySelector("#form-adiciona");
```

```
var nome = form.nome.value;
var peso = form.peso.value;
var altura = form.altura.value;
var gordura = form.gordura.value;

var pacienteTr = document.createElement("tr");

var nomeTd = document.createElement("td");
var pesoTd = document.createElement("td");
var alturaTd = document.createElement("td");
var gorduraTd = document.createElement("td");
var imcTd = document.createElement("td");

nomeTd.textContent = nome;
pesoTd.textContent = peso;
alturaTd.textContent = altura;
gorduraTd.textContent = gordura;

pacienteTr.appendChild(nomeTd);
pacienteTr.appendChild(pesoTd);
pacienteTr.appendChild(alturaTd);
pacienteTr.appendChild(gorduraTd);

var tabela = document.querySelector("#tabela-pacientes");

tabela.appendChild(pacienteTr);

});
```

Agora, no momento em que preenchemos os dados do formulário e clicamos em **Adicionar**, o paciente é adicionado à tabela!

Ainda não estamos calculando o IMC, mas faremos isso no próximo capítulo, vendo como construir um código bem organizado.