

Mãos na Massa: Adicionando um paciente na tabela

Com o form criado e já escutando nossos eventos de click, vamos começar a implementar a adição de um novo paciente na tabela:

1- Nosso primeiro passo é selecionar o `<form>` para que tenhamos acesso aos campos de `input` com os dados do paciente. Vamos trazê-lo do HTML:

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

2- Depois de selecionado o form, precisamos ter acesso a cada um dos inputs. Para fazer isto podemos nos apoiar numa característica do form, que é o acesso que temos aos seus inputs como se fossem propriedades do form, bastando usar como nome da propriedade os atributos *name* dos inputs. Como temos `inputs` com os *name's* com valor de `nome`, `altura`, `peso` e `gordura` faremos assim:

```
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;  
});
```

Lembrando que para pegar o valor digitado dentro de um `input`, devemos acessar a sua propriedade `value`.

3- Agora com os valores extraídos do formulário HTML para o Javascript, podemos começar a criar um novo paciente na tabela. Sabemos que pacientes são representados por `<tr>`'s na tabela, então precisamos criar um elemento `<tr>` dentro do Javascript. Para isto, vamos utilizar a função `document.createElement()`:

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

4- Agora precisamos criar cada uma das `<td>`'s que ficarão dentro da `<tr>` . Como um paciente tem 5 atributos (nome, peso, altura, gordura e imc). Vamos utilizar a mesma função para criar estas `<td>`'s :

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

5- Temos que colocar os valores que extraímos do form dentro de cada uma das `<td>`'s , então vamos preenchê-las com nossa conhecida `.textContent` :

```
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;
});
```

6- Agora temos 5 `<td>`'s e 1 `<tr>` , mas elas não estão conectadas. O certo é posicionar as `<td>`'s dentro de cada `<tr>`'s . Vamos fazer isso através da função `appendChild()` :

```
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);  
});
```

7- Por último precisamos adicionar esta <tr> na tabela. Para isto vamos usar uma tática parecida, selecionar a tabela em seguida usar a função .appendChild() :

```
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 sim estamos adicionando o paciente a partir do formulário, mas veja que ele ainda está sem a coluna de IMC. Vamos resolver isto no próximo capítulo com boas práticas de reuso de código, até lá!