

Mão na massa: organizando o código em vários arquivos

Começando deste ponto?

Começando deste ponto? Você pode fazer o [download \(<https://s3.amazonaws.com/caelum-online-public/introducao-javascript/stages/introducao-javascript-capitulo-5.zip>\)](https://s3.amazonaws.com/caelum-online-public/introducao-javascript/stages/introducao-javascript-capitulo-5.zip) completo do projeto do capítulo anterior e continuar seus estudos a partir deste capítulo.

Neste capítulo vamos organizar nosso código para que ele fique mais fácil de manter e para que sigamos as boas práticas.

1- Nossa arquivo `principal.js` estava começando a ficar muito grande, logo vamos separá-lo em arquivo Javascript individuais, aonde cada arquivo ficará responsável por uma funcionalidade do sistema. Primeiro, crie um novo arquivo chamado `form.js`, e importe-o no fim no seu HTML:

```
// Restante do código HTML
    </section>
    </main>
    <!-- Importação dos Javascripts AQUI -->
    <script src="js/principal.js"></script>
    <script src="js/form.js"></script>
</body>
</html>
```

2- Com o arquivo criado e importado, mova tudo que estiver relacionado com nosso formulário de adicionar paciente para lá. Basicamente tudo dentro do event listener do click do botão:

```
//form.js

var botaoAdicionar = document.querySelector("#adicionar-paciente");
botaoAdicionar.addEventListener("click", function(event) {
    // Todo o código da função.
});
```

3- Aproveite e renomeie o seu arquivo `principal.js` para um nome mais semântico, como `calcula-imc.js`, que diz melhor o que aquele código faz.

4- Agora que já fizemos esta separação em arquivos, vamos começar a melhorar mais ainda o código. É a hora de exportarmos certas partes do código para funções, assim fica mais fácil reutilizá-las. Um bom exemplo é o código responsável por calcular um IMC, que deve ser usado tanto na função cálculo IMC quanto quando criamos um novo paciente através do formulário. Crie a função `calculaImc` em seu arquivo `calcula-imc.js` como abaixo:

```
//calcula-imc.js

// restante do código

function calculaImc(peso, altura){
    var imc = 0;
```

```

imc = peso / (altura * altura);
return imc.toFixed(2);
}

```

5- Agora substitua o antigo cálculo na mão que fazíamos dentro do `if` por uma chamada a nossa recém criada função `calculaImc`, passando os parâmetros de acordo:

```

//calcula-imc.js
// Restante do código
if (pesoEhValido && alturaEhValida) {
    var imc = calculaImc(peso,altura);
    imcTd.textContent = imc;
}
// Restante do código

```

6- Vamos também chamar a função `calculaImc` no nosso `form.js`, para que o IMC do paciente também seja calculado quando ele for inserido na tabela. Adicione junto dos outros `<td's>` para que o conteúdo do `tdImc` seja o retorno da função `calculaImc`

```

//form.js
nomeTd.textContent = nome;
pesoTd.textContent = peso;
alturaTd.textContent = altura;
gorduraTd.textContent = gordura;
//Adicionar aqui:
imcTd.textContent = calculaImc(peso,altura);

```

7- E não vamos esquecer de colocar o `tdImc` também dentro do `<tr>` paciente:

```

//form.js

pacienteTr.appendChild(nomeTd);
pacienteTr.appendChild(pesoTd);
pacienteTr.appendChild(alturaTd);
pacienteTr.appendChild(gorduraTd);
//Adicionar aqui:
pacienteTr.appendChild(imcTd);

```

Agora quando o paciente for adicionado na tabela, seu IMC também será calculado e inserido automaticamente!