

Mão na massa: validando o form

Começando deste ponto?

Começando deste ponto? Você pode fazer o [download \(<https://s3.amazonaws.com/caelum-online-public/introducao-javascript/stages/introducao-javascript-capitulo-6.zip>\)](https://s3.amazonaws.com/caelum-online-public/introducao-javascript/stages/introducao-javascript-capitulo-6.zip) completo do projeto do capítulo anterior e continuar seus estudos a partir deste capítulo.

Vamos aproveitar as regras de validação que tínhamos no `calcula-imc.js` para poder validar os pacientes que adicionamos pelo formulário e poder exibir as mensagens de erro.

1- Vamos começar exportando a lógica de validação de peso e altura para duas funções externas, a `validaPeso` e a `validaAltura`. Em seu arquivo `calcula-imc.js` crie as seguintes funções:

```
function validaPeso(peso) {
  if (peso > 0 && peso < 1000) {
    return true;
  } else {
    return false;
  }
}

function validaAltura(altura) {
  if (altura > 0 && altura <= 3.00) {
    return true;
  } else {
    return false;
  }
}
```

2- Agora faça as seguintes alterações no seu for para usarmos nossas funções validadoras:

```
for (var i = 0; i < pacientes.length; i++) {

  //Restante do código
  ...
  // Alteração
  var alturaEhValida = validaAltura(altura);
  var pesoEhValido = validaPeso(peso);
  // Alteração
  if (!pesoEhValido) {
    ...
  }
  // Alteração
  if (!alturaEhValida) {
    ...
  }
  // Alteração
  if (pesoEhValido && alturaEhValida) {
    var imc = calculaImc(peso, altura);
    tdImc.textContent = imc;
  }
}
```

```

    }
}

```

3- Vamos agora criar a função que irá validar o usuário quando submetermos o form. O nome desta será `validaPaciente`, que receberá o objeto paciente e retornará um array de `erros` :

```

// form.js
function validaPaciente(paciente) {

    var erros = [];

    if (paciente.nome.length == 0) {
        erros.push("O nome não pode ser em branco");
    }

    if (paciente.gordura.length == 0) {
        erros.push("A gordura não pode ser em branco");
    }

    if (paciente.peso.length == 0) {
        erros.push("O peso não pode ser em branco");
    }

    if (paciente.altura.length == 0) {
        erros.push("A altura não pode ser em branco");
    }

    if (!validaPeso(paciente.peso)) {
        erros.push("Peso é inválido");
    }

    if (!validaAltura(paciente.altura)) {
        erros.push("Altura é inválida");
    }

    return erros;
}

```

Esta função cria um array e caso encontre algum erro em alguma propriedade do paciente, adiciona ao array de erros a mensagem que será exibida depois.

4- Agora vamos chamar nossa função `validaPaciente` quando clicarmos no botão adiciona e tratar de exibir a mensagem de erro caso haja alguma:

```

botaoAdicionar.addEventListener("click", function(event) {
    event.preventDefault();
    var form = document.querySelector("#form-adiciona");
    var paciente = obtemPacienteDoFormulario(form);
    var pacienteTr = montaTr(paciente);

    //Adição aqui
    var erros = validaPaciente(paciente);

    if (erros.length > 0) {

```

```

        // Aqui vai entrar o código de exibir erros.

    }
    // Resto do código
    ...
});
```

5- Com toda a lógica preparada, basta apenas exibirmos as mensagens de erro de dentro do Array. Para isto, vamos criar uma `` acima do formulário de adicionar pacientes para exibir as mensagens:

```

//principal.html

...
<ul id="mensagens-erro"></ul>
<form>
...
```

6- Crie a função `exibeMensagensDeErro` , que recebe o array de erros e cria uma `` para cada mensagem de erro:

```

function exibeMensagensDeErro(erros) {
    var ul = document.querySelector("#mensagens-erro");
    ul.innerHTML = "";

    erros.forEach(function(erro) {
        var li = document.createElement("li");
        li.textContent = erro;
        ul.appendChild(li);
    });
}
```

7- Agora basta chamar a função `exibeMensagemErro()` dentro do if que verifica se existe algum erro no array:

```

//form.js
botaoAdicionar.addEventListener("click", function(event) {
    event.preventDefault();
    var form = document.querySelector("#form-adiciona");
    var paciente = obtemPacienteDoFormulario(form);

    var erros = validaPaciente(paciente);

    if (erros.length > 0) {
        //Adição aqui
        exibeMensagensDeErro(erros);
        return;
    }
    // Resto do código
    var pacienteTr = montaTr(paciente);
    ...
});
```

8- Para que as mensagens de erro fiquem vermelhas, adicione o seguinte estilo abaixo em seu `estilos.css` :

```
// estilos.css
// Restante do CSS
...
#mensagens-erro{
    color: red;
}
```

9- Por último, vamos adicionar nosso paciente na tabela, além disso, precisamos remover as mensagens de erro caso seja adicionado um paciente válido.

O código para isso é:

```
var tabela = document.querySelector("#tabela-pacientes");
tabela.appendChild(pacienteTr);
form.reset();

var mensagensErro = document.querySelector("#mensagens-erro");
mensagensErro.innerHTML = "";
```

Nosso código completo será:

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

    var erros = validaPaciente(paciente);

    if (erros.length > 0) {
        //Adição aqui

        exibeMensagensDeErro(erros);
        return;
    }
    var pacienteTr = montaTr(paciente);
    var tabela = document.querySelector("#tabela-pacientes");
    tabela.appendChild(pacienteTr);
    form.reset();

    var mensagensErro = document.querySelector("#mensagens-erro");
    mensagensErro.innerHTML = "";
});
```