

Operadores Lógicos

Transcrição

Conseguimos calcular o IMC do paciente Paulo, mas o que acontece se alguém, no momento de digitar o código HTML, colocar um número negativo no campo de peso? Ou se incluir um número absurdo, como 10 mil quilos? Ou, sem querer, adicionar uma altura de 200 ?

```
IMC = 10000 / 2.00 x 2.00  
IMC = 10000 / 4.00  
IMC = 2500
```

O resultado será um valor impossível. Às vezes, na pressa de digitar, a pessoa pode acabar colocando um número a mais ou um sinal a menos, e dentre tantas outras informações, acaba não notando esses erros no HTML.

Valores improváveis não deveriam ser aceitos pela tabela. Como já possuímos os dados do peso e da altura no código JavaScript, é possível **validar** as informações. Desta forma, o IMC do paciente só será calculado se o peso e a altura forem válidos, e sempre teremos um IMC que reflita um valor real. Se o peso for muito alto ou a altura muito baixa, por exemplo, será exibida uma mensagem de aviso informando que os dados do paciente são inválidos e devem ser verificados.

Atualmente, o `principal.js` está assim:

```
var titulo = document.querySelector(".titulo");  
titulo.textContent = "Aparecida Nutricionista";  
  
var paciente = document.querySelector("#primeiro-paciente");  
  
var tdPeso = paciente.querySelector(".info-peso");  
var peso = tdPeso.textContent;  
  
var tdAltura = paciente.querySelector(".info-altura");  
var altura = tdAltura.textContent;  
  
var tdImc = paciente.querySelector(".info-imc");  
  
var imc = peso / (altura * altura);  
tdImc.textContent = imc;
```

Antes de calcularmos o IMC, poderemos validar os dados, começando pelo peso. Com um `if`, verificaremos se o peso é menor ou igual a 0, ou seja, **negativo**. Se este for o caso, uma mensagem de alerta será exibida no console. Iremos adicionar o `if` logo abaixo da variável `tdImc`:

```
var tdAltura = paciente.querySelector(".info-altura");  
var altura = tdAltura.textContent;  
  
var tdImc = paciente.querySelector(".info-imc");  
  
if (peso <= 0) {
```

```
    console.log("Peso inválido");  
  }
```

Se colocarmos um valor inválido para o peso no index.html ...

```
<tbody id="tabela-pacientes">  
  <tr class="paciente" id="primeiro-paciente">  
    <td class="info-nome">Paulo</td>  
    <td class="info-peso">-100</td>  
    <td class="info-altura">2.00</td>  
    <td class="info-gordura">10</td>  
    <td class="info-imc">0</td>  
  </tr>
```

Além disso, 1000 quilos também é um valor muito alto para um ser humano. Então, vamos verificá-lo também, adicionando outro `if` :

```
var paciente = document.querySelector("#primeiro-paciente");  
  
var tdPeso = paciente.querySelector(".info-peso");  
var peso = tdPeso.textContent;  
  
var tdAltura = paciente.querySelector(".info-altura");  
var altura = tdAltura.textContent;  
  
var tdImc = paciente.querySelector(".info-imc");  
  
if (peso <= 0) {  
  console.log("Peso inválido");  
}  
  
if (peso > 1000) {  
  console.log("Peso inválido");  
}  
  
var imc = peso / (altura * altura);  
tdImc.textContent = imc;
```

No console, será exibida a mensagem `Peso inválido` .

The screenshot shows a web browser window with the address bar displaying `file:///Users/Caelum/Documents/introducao-javascript/index.html`. The page has a dark header with the text "Aparecida Nutricionista" and a main heading "Meus pacientes". Below the heading is a table with 5 columns: "Nome", "Peso(kg)", "Altura(m)", "Gordura Corporal(%)", and "IMC". The table contains 6 rows of patient data. Below the table, a browser console is open, showing a message "Peso inválido" with a reference to `principal.js:16`.

Nome	Peso(kg)	Altura(m)	Gordura Corporal(%)	IMC
Paulo	-100	2.00	10	-25
João	80	1.72	40	0
Erica	54	1.64	14	0
Douglas	85	1.73	24	0
Tatiana	46	1.55	19	0

Console log: `Peso inválido` (principal.js:16)

No entanto, imagine o que aconteceria se para cada validação de peso, fosse necessário adicionar mais um `if`? O código acabaria ficando imenso! Porém, é possível juntar os dois tipos de verificações em um único `if`, sendo que cada um deles retornará `true` ou `false` para as verificações, ou seja, um resultado **booleano**. Tentaremos concatenar as verificações:

```
var tdAltura = paciente.querySelector(".info-altura");
var altura = tdAltura.textContent;

var tdImc = paciente.querySelector(".info-imc");

if (peso < 0 peso > 1000) {
  console.log("Peso inválido");
}
```

Porém, como faremos para juntar as duas condições, sendo que o resultado deverá se enquadrar em uma ou outra condição? A palavra "ou" não existe no JavaScript, mas existe um operador lógico nas linguagens de programação que poderá ser utilizado: `||`, duas barras verticais que recebem o nome de *pipes*:

```
var tdAltura = paciente.querySelector(".info-altura");
var altura = tdAltura.textContent;

var tdImc = paciente.querySelector(".info-imc");

if (peso < 0 || peso > 1000) {
  console.log("Peso inválido");
}
```

Se a primeira condição, `peso < 0`, for verdadeira, ou a segunda condição, `peso > 1000`, for verdadeira, o código no `if` será executado. Ao testarmos, veremos que o funcionamento é o mesmo, sem a necessidade de criarmos um `if` para cada caso.

Seria interessante também verificarmos o valor de altura, considerando inválida aquela que for menor que 0 ou maior que 3:

```
if (altura < 0 || altura > 3.00) {  
    console.log("Altura inválida");  
}
```

Se testarmos rodar o código com um valor fora de uma das condições, a mensagem `Altura inválida` será exibida no console. Mas o cálculo do IMC ainda será realizado, mesmo nos casos em que trabalharmos com alturas e/ou pesos inválidos, o que não é o que queremos. Neste caso, gostaríamos que o IMC não fosse calculado.

##Cálculo do IMC somente com peso e altura válidos

Para que o IMC não seja calculado quando o peso e a altura forem inválidos, poderemos criar duas variáveis, uma validação para cada caso. Vamos inicializá-las com `true` :

```
var pesoEhValido = true;  
var alturaEhValida = true;  
  
if (peso < 0 || peso > 1000) {  
    console.log("Peso inválido");  
}  
  
if (altura < 0 || altura > 3.00) {  
    console.log("Altura inválida!");  
}
```

Poderemos testar inserindo um valor de altura inválido. No console, veremos a mensagem "Altura inválida!". Ou seja, estamos presumindo que ambos serão válidos. Mas caso não sejam, seus valores deverão mudar para `false` dentro dos `if` s:

```
if (peso < 0 || peso > 1000){  
    console.log("Peso inválido!");  
    pesoEhValido = false;  
}  
  
if (altura < 0 || altura > 3.00){  
    console.log("Altura inválida!");  
    alturaEhValida = false;  
}
```

Os valores das variáveis começarão como verdadeiras, mas se algo der errado, serão setadas como `false` . Após as alterações, o código ficará assim:

```
var paciente = document.querySelector("#primeiro-paciente");  
  
var tdPeso = paciente.querySelector(".info-peso");  
var peso = tdPeso.textContent;  
  
var tdAltura = paciente.querySelector(".info-altura");  
var altura = tdAltura.textContent;  
  
var tdImc = paciente.querySelector(".info-imc");
```

```
var pesoEhValido = true;
var alturaEhValida = true;

if (peso < 0 || peso > 1000) {
  console.log("Peso inválido!");
  pesoEhValido = false;
}

if (altura < 0 || altura > 3.00) {
  console.log("Altura inválida!");
  alturaEhValida = false;
}

var imc = peso / (altura * altura);
tdImc.textContent = imc;
```

Agora, antes de calcular o IMC, faremos uma pequena verificação das duas variáveis: se a primeira (alturaEhValida) for verdadeira **E** a segunda (pesoEhValido) também, o IMC será calculado.

Para verificarmos se uma condição **E** outra são verdadeiras, utilizaremos o operador lógico representado por dois & (&& **):**

```
var pesoEhValido = true;
var alturaEhValida = true;

if (peso < 0 || peso > 1000) {
  console.log("Peso inválido!");
  pesoEhValido = false;
}

if (altura < 0 || altura > 3.00) {
  console.log("Altura inválida!");
  alturaEhValida = false;
}

if (alturaEhValida && pesoEhValido) {
  var imc = peso / (altura * altura);
  tdImc.textContent = imc;
}
```

Ao testarmos, iremos verificar se o peso ou a altura são inválidos e, neste caso, o IMC não será calculado. Na tela, veremos impresso o valor 0 na coluna da tabela.

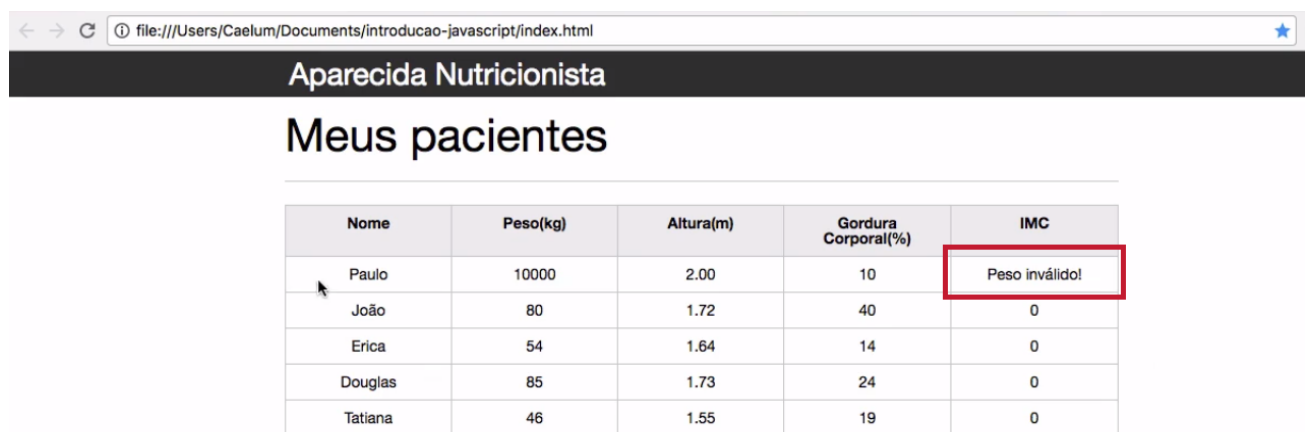


A screenshot of a web browser displaying a table titled "Meus pacientes". The table has five columns: "Nome", "Peso(kg)", "Altura(m)", "Gordura Corporal(%)", and "IMC". The "IMC" column is highlighted with a red border. The data in the table is as follows:

Nome	Peso(kg)	Altura(m)	Gordura Corporal(%)	IMC
Paulo	100	4.00	10	0
João	80	1.72	40	0
Erica	54	1.64	14	0
Douglas	85	1.73	24	0
Tatiana	46	1.55	19	0

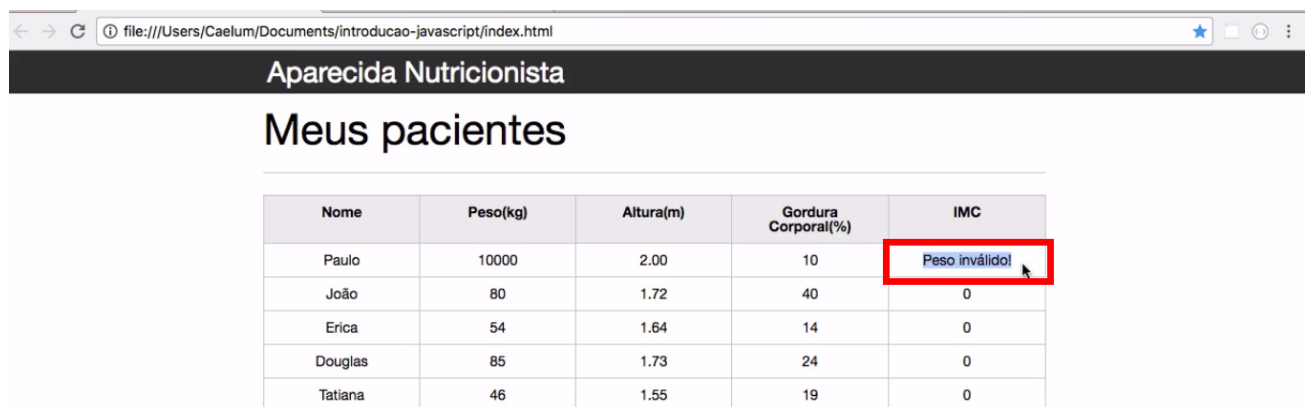
Os usuários geralmente não ficam com o console do navegador aberto, então criaremos uma mensagem mais visível para eles. Na hora de verificar o peso e a altura, se algum deles for inválido, deixaremos uma mensagem mais significativa alterando-se o `textContent` :

```
if (peso < 0 || peso > 1000) {  
    console.log("Peso inválido!");  
    pesoEhValido = false;  
    tdImc.textContent = "Peso inválido!";  
}  
  
if (altura < 0 || altura > 3.00){  
    console.log("Altura inválida!");  
    alturaEhValida = false;  
    tdImc.textContent = "Altura inválida!";  
}  
  
if (alturaEhValida && pesoEhValido){  
    var imc = peso / (altura * altura);  
    tdImc.textContent = imc;  
}
```



Nome	Peso(kg)	Altura(m)	Gordura Corporal(%)	IMC
Paulo	10000	2.00	10	Peso inválido!
João	80	1.72	40	0
Erica	54	1.64	14	0
Douglas	85	1.73	24	0
Tatiana	46	1.55	19	0

Observe que alteramos o `textContent` para "Peso inválido" e "Altura inválida". Se colocarmos um peso ou uma altura inválidos no HTML e testar novamente a nossa página, a mensagem aparecerá conforme o esperado.



Nome	Peso(kg)	Altura(m)	Gordura Corporal(%)	IMC
Paulo	10000	2.00	10	Peso inválido!
João	80	1.72	40	0
Erica	54	1.64	14	0
Douglas	85	1.73	24	0
Tatiana	46	1.55	19	0

É possível refazermos o teste alterando o valor da altura no `index.html` :

```
<tbody id="tabela-pacientes">  
    <tr class="paciente" id="primeiro-paciente">
```

```
<td class="info-nome">Paulo</td>
<td class="info-peso">-100</td>
<td class="info-altura">4.00</td>
<td class="info-gordura">10</td>
<td class="info-imc">0</td>
</tr>
```

Quando colocamos uma altura de 4.00, na tabela será exibida a mensagem de Altura Inválida!.



Nome	Peso(kg)	Altura(m)	Gordura Corporal(%)	IMC
Paulo	100	4.00	10	Altura inválida!
João	80	1.72	40	0
Erica	54	1.64	14	0
Douglas	85	1.73	24	0
Tatiana	46	1.55	19	0

Há uma pequena falha nas nossas condições se o peso for igual a 1000. O IMC continuará sendo calculado. Faremos um ajuste no `if` para que isso ocorra, agora a condição será de `>=`:

```
//...

if (peso <= 0 || peso >= 1000) {
  console.log("Peso inválido!");
  pesoEhValido = false;
  tdImc.textContent = "Peso inválido!";
}

if (altura <= 0 || altura >= 3.00){
  console.log("Altura inválida!");
  alturaEhValida = false;
  tdImc.textContent = "Altura inválida!";
}

if (alturaEhValida && pesoEhValido){
  var imc = peso / (altura * altura);
  tdImc.textContent = imc;
}
```

Tomamos o mesmo cuidado com os valores de `peso`. Observe que como não queremos que o valor 0 seja considerado válido para `altura` e `peso`, iremos alterar a condição para `<=`. Detectamos mais um erro que poderia ocorrer no código!

Com esta última validação, chegamos ao fim da aula. Vimos vários assuntos, como as lógicas com o `if` e sobre como calcular o IMC adicionando os parênteses. Mostramos o uso adequado dos operadores matemáticos e o operador lógico de **OU** (`||`), utilizado no `if`. Vimos também o `&&`, usado com o significado de **E** na condição.

