

Melhorando programa de IMC

Transcrição

Na aula anterior utilizamos como exemplo dois times de futebol. Nossa objetivo foi aprender a dotar o programa de certa inteligência para fornecer uma resposta aos usuários.

Nesta aula vamos retornar ao programa do IMC. Lembrando que a OMS - Organização Mundial da Saúde - determina que indivíduos com IMC menor que 18.5 estão abaixo da média, com IMC 35 estão acima da média e entre 18.5 e 35 estão na média aceitável.

Nosso programa é personalizado, ele questiona ao usuário seu nome, altura e peso. Ao em vez de aparecer apenas a mensagem indicando o IMC da pessoa, vamos incluir uma mensagem que avise qual é o status em que a pessoa se encontra, abaixo, acima ou na média dos valores do IMC.

Primeiro, vamos aplicar outro `if` no código do IMC para que ele ajude a fornecer o diagnóstico. Abra o código do arquivo `imc.html` e depois do cálculo do Índice de Massa Corporal do usuário, vamos introduzir um `if()` para analisar se o valor é menor que 18.5. Se ele estiver com IMC abaixo da média, isso será indicado ao usuário. Por hora, o código está da seguinte maneira:

```
var imc = calculaImc(alturaInformada, pesoInformado);
mostra("O seu imc é " + imc);

if(imc < 18.5) {
  mostra("Seu IMC indica que você está ABAIXO do peso.");
}
```

Rode seu programa e faça um novo teste com os valores 60 quilos e 1.82 de altura. Faça também um prova utilizando uma altura menor para que o indivíduo fique na média do IMC e, nesse caso, nenhuma mensagem deve ser mostrada!

Vamos verificar ainda a condição oposta, usando valores que indiquem obesidade:

```
if(imc > 35) {
  mostra("Seu IMC indica que você está ACIMA do peso.");
}
```

Faça o teste utilizando o peso de 99 quilos e 1.6 de altura.

E se o `imcDoUsuario` está acima de 18.5 **e, ao mesmo tempo**, abaixo de 35? Você pode verificar essas duas situações, uma de cada vez. Primeiro, fazemos o `if` para saber se a pessoa está acima de 18.5:

```
if(imc > 18.5) {
  // nao basta estar acima de 18.5
  // preciso verificar se está abaixo de 35
}
```

E dentro disso também verificamos se o indivíduo está abaixo de 35. Lembrando que para verificar a condição utilizamos o `if`. Então, teremos o seguinte:

```
if(imc > 18.5) {
  if(imc < 35) {
    mostra("OK! Seu IMC está entre os dois limites.");
  }
}
```

Observando o que fizemos acima pode parecer complicado, pois temos um `if` dentro de outro! Mas, repare como a leitura do código está fluída. Isso ocorre porque usamos o recurso de **indentação** para deixar claro que o segundo `if` está dentro do primeiro. Se não tivéssemos usado o `TAB` ficaria difícil perceber esse detalhe, quer ver? Observe o exemplo de uma péssima indentação, um código extremamente mal formatado:

```
if(imc > 18.5) {
if(imc < 35) {
  mostra("OK! Seu IMC está entre os dois limites.");
}
}
```

Apesar do código acima funcionar perfeitamente é difícil lê-lo e saber o que acontece nele.

É importante que o aluno se esforce desde agora para manter uma indentação correta. O código deve possuir sempre um `TAB` a mais para a direita quando ele pertencer a um trecho de código - ou **bloco** de código - de fora. Em um código bem indentado fica claro, por exemplo, que a função `mostra` pertence ao bloco do `if(imcDoUsuario < 35)`, que por sua vez pertence ao bloco do `if(imc > 18.5)`.

Em um mesmo código é frequente a verificação se mais de uma condição é verdadeira. Essa situação é tão comum que existe uma maneira mais curta e simples de fazer dois `ifs` um dentro do outro. Existe uma maneira de escrever, "quero que isso **E** isso seja verdadeiro", e essa forma do **E** é feita utilizando um operador que até pode parecer estranho, mas é comumente utilizado, o `&&`. Observe no código abaixo seu uso:

```
if(imc > 18.5 && imc < 35) {
  mostra("OK! Seu IMC está entre os dois limites.");
}
```

Essa fórmula é bem mais legível do que escrever dois `ifs`, um dentro do outro. A leitura dessa condição fica da seguinte maneira: "se o `imcDoUsuario` é maior que 18.5 **E** o `imcDoUsuario` é menor que 35, faça...".

Existe ainda um pequeno problema com nosso código! Se o IMC for exatamente igual a `18.5` ou a `35`, nenhuma das condições que construímos será verdadeira e, consequentemente, nada será impresso. Por isso, existem os comparadores `>=` e `<=`. Vamos considerar que os valores exatos fazem parte do IMC sadio, então, escrevemos a condição com novos operadores de comparação. O código ficará da seguinte maneira:

```
if(imc >= 18.5 && imc <= 35) {
```

```
mostra("OK! Seu IMC está entre os dois limites.");  
}
```