

04

Boolean condicionais

Transcrição

Vamos explorar um pouco mais o funcionamento do `if`, para o qual criaremos mais uma classe. É recomendado criá-las para termos um histórico do que está sendo montado, passo a passo. Em `TestaCondicional2`, teremos o código mais ou menos parecido com o que estávamos vendo até então:

```
public class TestaCondicional2 {  
    public static void main(String[] args) {  
        System.out.println("testando condicionais");  
        int idade = 16;  
        int quantidadePessoas = 3;  
  
        if (idade >= 18) {  
            System.out.println("você tem mais de 18 anos");  
            System.out.println("seja bem vindo");  
        } else {  
            if(quantidadePessoas >= 2) {  
                System.out.println("você não tem 18, mas " + "pode entrar, pois está acompanhado");  
            } else {  
                System.out.println("infelizmente você não pode entrar");  
            }  
        }  
    }  
}
```

Porém, não é muito legal quando o código tem muitos `if`s e `else`s encadeados, algo academicamente denominado **complexidade ciclomática** ou **complexidade condicional**. Neste nosso exemplo, poderíamos juntar os casos em que a pessoa tem mais de 18 anos e está acompanhada em uma condicional única.

Para isso, utilizaremos o operador `ou`, `||` - no Java, não existe `or` ou `and` como palavras chave.

```
public class TestaCondicional2 {  
    public static void main(String[] args) {  
        System.out.println("testando condicionais");  
        int idade = 16;  
        int quantidadePessoas = 3;  
  
        if (idade >= 18 || quantidadePessoas >= 2) {  
            System.out.println("seja bem vindo");  
        } else {  
            System.out.println("infelizmente você não pode entrar");  
        }  
    }  
}
```

Vamos salvar e rodar o código para ver o que acontece? Será impresso no Console:

```
testando condicionais
seja bem vindo
```

Para este operador, basta apenas uma das condições ser `true`. Há também o `e`, ou `&&`, para quando houver necessidade de se ter mais de 18 anos e estar acompanhado, por exemplo. Isto é, se mantivermos `idade` como 16 e `quantidadePessoas = 1;`, obteremos infelizmente você não pode entrar.

Aprendemos sobre tipos de variáveis como o `int` e o `double`, para inteiros e pontos flutuantes, respectivamente, o `char` para quando se usa apenas um caractere, entre outros. Além deles, existe o `boolean`, palavra chave do Java que é um tipo de variável que só aceita `true` (verdadeiro) ou `false` (falso), e fazem parte das palavras reservadas do Java.

```
public class TestaCondisional2 {
    public static void main(String[] args) {
        System.out.println("testando condicionais");
        int idade = 16;
        boolean acompanhado = true;

        if (idade >= 18 && acompanhado) {
            System.out.println("seja bem vindo");
        } else {
            System.out.println("infelizmente você não pode entrar");
        }
    }
}
```

No Java, `=` atribui, enquanto `==` compara. Em `boolean`, no caso de `acompanhado == true`, o próprio `acompanhado` já é um valor booleano, portanto, `== true` não é necessário.

O que também aparece com certa frequência é, à direita do `boolean`, colocarmos uma **expressão booleana** como `idade >= 18 && acompanhado`. Sendo assim, poderíamos usar simplesmente `boolean acompanhado = quantidadePessoas >= 2;`, o que fará com que se conclua se a pessoa está acompanhada ou não. Com a `idade` sendo 20, se rodarmos o código, obteremos `seja bem vindo`.

Também é possível imprimirmos "valor de acompanhado" e concatená-lo com `acompanhado`, deixando o código final assim:

```
public class TestaCondisional2 {
    public static void main(String[] args) {
        System.out.println("testando condicionais");
        int idade = 20;
        int quantidadePessoas = 3;
        boolean acompanhado = quantidadePessoas >= 2;

        System.out.println("valor de acompanhado = " + acompanhado);

        if (idade >= 18 && acompanhado) {
            System.out.println("seja bem vindo");
        } else {
            System.out.println("infelizmente você não pode entrar");
        }
    }
}
```

```
    }  
}
```

Salvaremos e rodaremos mais uma vez, e imprimiremos o seguinte:

```
testando condicionais  
valor de acompanhado = true  
seja bem vindo
```