

Comandos de seleção (if/else)

Mateus Raeder

Expressões booleanas

- ▶ Tipo boolean
 - ▶ Tipo de dado primitivo
 - ▶ Possui os valores: **true** e **false**

Por exemplo:

```
boolean op; // declara uma variável di tipo boolean
op = false; // atribui valor false para a variável booleana op
op = true;  // atribui valor true para a variável booleana op
```

Expressão relacional

- Expressões relacionais permitem comparar dois valores através de operadores relacionais, resultando em true ou false
 - Exemplos:
 - `x > 13` será **true** se o valor da variável `x` for superior a 13
 - `aux + valor == 10` será **true** se o valor da soma de `aux` com `valor` for = a 10
- Operadores relacionais

Operador	Significado
>	Maior que
<	Menor que
>=	Maior ou igual a
<=	Menor ou igual a
==	Igual a
!=	Diferente de

Exercícios

Sabendo que:

```
int a = 0;  
double b = -2.5;  
boolean c = true;
```

Responda o que segue abaixo:

- | | | |
|----|----------------|-------|
| a) | $a > 0$ | false |
| b) | $a == 0$ | true |
| c) | $b \leq -1$ | true |
| d) | $b \neq 2.5$ | true |
| e) | c | true |
| f) | $a + 2 \geq 3$ | false |
| g) | $b + 2.5 == 0$ | true |

Expressão lógica

- Expressões lógicas utilizam operadores lógicos, resultando também em true ou false
 - Exemplos:
 - `(x > 13) && (aux + 10 == 0)` será **true** se o valor da variável *x* for superior a 13 **e** se o valor da soma de *aux* com 10 for igual a 0
 - `(a < 2) || (c == false)` será **true** se o valor de *a* for inferior a 2 **ou** se o valor de *c* for false
- Operadores lógicos

Operador	Significado
!	Negação lógica (inverte o valor da expressão (exemplo: !op))
&&	e lógico: será true se ambos os lados forem true
	ou lógico: será true se pelo menos um dos lados for true

Exercícios

Sabendo que:

```
int a = 0;  
double b = -2.5;  
boolean c = true;
```

Responda o que segue abaixo:

- | | |
|--------------------------------|-------|
| a) !c | false |
| b) a == 0 && c | true |
| c) b <= -1 a < 0 | true |
| d) b != 2.5 a == 0 | true |
| e) c !c | true |
| f) a + 2 >= 3 && b == -2.5 | false |
| g) b + 2.5 == 0 && c && a <= 3 | true |

Instrução IF

Sintaxe 1 *(se dentro do if houver somente uma linha de comando):*

```
if (condição)
    comando;
```

Condição: qualquer valor ou expressão booleana
Comando: qualquer instrução válida da linguagem

Sintaxe 2 *(se dentro do if houver mais de uma linha de comando):*

```
if (condição)
{
    comando 1;
    comando 2;
    ...
    comando n;
}
```

Na instrução IF, somente se a condição for verdadeira (ou seja, com valor **true**), o(s) comando(s) serão executados.

Instrução IF

■ Exemplos

```
if(i > 7)
    System.out.println("Valor de i é maior que 7");
```

```
if(op == 2)
{
    System.out.println("Opção 2");
    op = 0;
    System.out.println("Opção setada para 0 novamente");
}
```

```
if(a != 0 && a % 2 == 0)
    System.out.println("O valor de a é par!");
```


Instrução IF - ELSE

Sintaxe 1 (se dentro do if ou do else houver somente uma linha de comando):

```
if (condição)
    comando;
else
    comando;
```

Sintaxe 2 (se dentro do if ou do else houver mais de uma linha de comando):

```
if (condição)
{
    comandos;
    ...
}
else
{
    comandos;
    ...
}
```

Na instrução IF- ELSE, se o valor da condição for **true**, os comandos do bloco **IF** serão executados. Se o valor da condição for **false**, os comandos do bloco **ELSE** serão executados.

Instrução IF - ELSE

■ Exemplos

```
if(i > 7)
    System.out.println("Valor de i é maior que 7");
else
    System.out.println("Valor de i é menor ou igual a 7");
```

```
if(a != 0 && a % 2 == 0)
{
    imprimeDados();
    System.out.println("O valor de a é par!");
}
else
{
    imprimeErro();
    System.out.println("O valor de a não é par!");
}
```

Usando ELSE-IF

```
if(a != 0 && a % 2 == 0){  
    imprimeDados();  
    System.out.println("O valor de a é par!");  
}else if(a == 0)  
    System.out.println("O valor de a é 0!");  
else if(a % 2 != 0)  
    System.out.println("O valor de a é ímpar!");  
else  
    imprimeErro();
```