



**SEJA PROTAGONISTA DA
SUA VIDA E NÃO PLATEIA
DA VIDA DOS OUTROS.**

@primeiromilhao





Desenvolvimento de Aplicações Desktop

Expressões

Java

Professor: Charles Leite

Expressões

- Uma EXPRESSÃO é uma combinação de um ou mais **operandos** e **operadores** que geralmente realizam um cálculo
- O valor calculado pode não ser necessariamente um número, embora, normalmente, seja
 - Mas, pode ser um booleano, um caractere, ou um objeto
- Os operandos podem ser:
 - Valores (literais), variáveis, constantes ou outras fontes de dados
- Os operadores podem ser:
 - Aritméticos
 - Relacionais
 - Lógicos

Veremos
depois!

Expressões Aritméticas

- Uma EXPRESSÃO ARITMÉTICA computa resultados numéricos, através de operadores aritméticos
- Operadores aritméticos:
 - Adição (+)
 - Pode ser usado na forma unária (+ *operando*) → Não produz efeito algum, apenas enfatiza a sua natureza positiva
 - Pode também ser usado para concatenar *strings*
 - Subtração (-)
 - Pode ser usado na forma unária (- *operando*) → Produz o efeito de negar um valor numérico
 - Multiplicação (*)
 - Divisão (/)
 - Resto da divisão (%)

Expressões Aritméticas

- Para os operadores de divisão (/) e resto da divisão (%), quando os operadores (ambos) forem inteiros, o resultado será um inteiro
 - A parte decimal é descartada

- Portanto, $14/3$

$$8/12$$

$$14\%3$$

$$8\%12$$

Expressões Aritméticas

- A avaliação ou cálculo de uma expressão faz-se obedecendo regras de associação e precedência estabelecidas na linguagem

```
int total = 3 + 4/2 ;
```

Divisão (/) tem precedência sobre soma (+)

Expressão é avaliada e o resultado é atribuído a *total* que armazena agora o valor 5

Java

Expressões Aritméticas

- Ordem de precedência dos operadores:

| Nível de precedência | Operador | Ordem de avaliação (associação) |
|----------------------|----------|---|
| 1 | () | Parênteses internos |
| 2 | * | Se houver vários operadores desse tipo, eles serão avaliados da esquerda para a direita |
| | / | |
| | % | |
| 3 | + | Se houver vários operadores desse tipo, eles serão avaliados da esquerda para a direita |
| | - | |

`resultado = 3 + 4 % 3 * 5`

Expressões Aritméticas

- O lado direito e esquerdo de um comando de atribuição podem, também, conter a mesma variável

```
int contador = 3;
```

Primeiro, 1 é adicionado ao valor original de contador

```
contador = contador + 1; ➔ 4
```



Depois o resultado é armazenado em contador (sobrescrevendo o seu valor original)

- Para simplificar ...

Expressões Aritméticas

- Os operadores de INCREMENTO e DECREMENTO são operadores unários
 - Usam somente um operando
- O operador de incremento (++) soma 1 ao seu operando
- O operador de decremento (--) subtrai 1 de seu operando
- Por exemplo:
 - A instrução
`++contador`
é funcionalmente equivalente a
`contador = contador + 1`

Expressões Aritméticas

- Esses operadores podem ser empregados de forma pós-fixada ou pré-fixada

`contador++;` **OU** `++ contador;`

Incrementa (decrementa) a variável **contador** mas retorna como resultado o valor antigo dela

Incrementa (decrementa) a variável **contador** e retorna como resultado o novo valor dela

- Quando usados de forma isolada, têm comportamentos equivalentes
- Porém, quando usados como parte de expressões maiores, eles podem ter comportamentos diferentes

Expressões Aritméticas

- Portanto, em expressões maiores, devem ser usados com cuidado

```
int contador = 3;
```

```
contador++;
```

contador agora armazena 4

```
++contador;
```

contador agora armazena 5

```
int valor = contador++;
```

valor agora armazena 5 e
depois contador é
incrementado para 6

```
valor = ++ contador;
```

contador é incrementado
para 7 e agora valor armazena
7



Expressões Aritméticas

- É comum fazermos algum tipo de operação com uma variável e depois armazenar o valor da operação na própria variável
- Java oferece uma forma mais compacta de fazer isso, usando o operador de atribuição

x += exp

x += 1;



x = x + 1;

x -= exp

x -= 2;



x = x - 2;

x *= exp

x *= k;



x = x * k;

x /= exp

x /= 3;



x = x / 3;

x %= exp

x %= 3;



x = x % 3;

DÚVIDAS ...



Referências

- DEITEL, Harvey. **Java Como Programar.** *Capítulo 2, páginas 41-43.* 8ª edição. São Paulo: Pearson Prentice Hall, 2010.