

Fundamentos de Java

4-3: Tipos de dados e operadores

Atividades Práticas

Objetivos da Lição:

- Usar tipos de dados primitivos em código Java
- Especificar literais dos tipos primitivos e das Strings
- Demonstrar como inicializar variáveis
- Descrever as regras do escopo de um método
- Reconhecer quando uma expressão exige uma conversão de tipo
- Aplicar casting no código Java
- Usar operadores aritméticos
- Usar o operador de atribuição
- Usar um método da classe Math
- Acessar um método da classe Math da API do Java

Vocabulário:

Identifique a palavra do vocabulário para cada definição a seguir.

	Mecanismos nomeados de armazenamento primitivo ou de objetos definidos em um programa. O valor designado pode ou não ser alterado (constantes).
	Os símbolos são usados para operações de adição, subtração, multiplicação, divisão e aritmética modular em expressões matemáticas e fórmulas.
	O grupo de tipos de dados Java que não usam a palavra-chave new quando declarados ou inicializados. Tipos de Dados Primitivos armazenam o valor no mesmo local da memória que o nome da variável.
	O menor tipo primitivo Java (1 byte) que pode conter um valor inteiro.
	Esse tipo de dado (8 bytes) é o maior tipo de inteiro.
	Os padrões de formatação e de nomenclatura que a maioria dos programadores seguem.
	Esse tipo de dado Java primitivo (4 bytes) pode conter valores inteiros.
	Esse tipo de dado Java primitivo (8 bytes) é o maior primitivo que pode conter um valor decimal.
	Quando um valor é designado a uma variável pela primeira vez.
	Esse tipo de dado Java primitivo (4 bytes) pode ser inicializado com um número decimal antecedendo uma letra f. Exemplo: float x = 3.5f;
	Pode ser qualquer número, texto ou outra informação que represente um valor ; usado para inicializar um tipo primitivo.

	Uma instrução Java quando uma variável é definida, mas não necessariamente recebe um valor. Exemplo: <code>int x;</code>
	Essa palavra descreve a precedência matemática que uma variável tem em um programa Java.
	Um tipo de dado Java primitivo (2 bytes) que pode conter valores de caractere únicos. Exemplo: "a", "#" ou "X"
	Usado para descrever o bloco de código, no qual existe uma variável em um programa. Um bloco de código é denotado por {}.
	O processo de modificar explicitamente um tipo de dado para se tornar um tipo de dado diferente.
	Um conceito no qual um número é sempre arredondado para o inteiro mais próximo.
	O sinal de igual "=" usado em uma instrução Java para designar um valor a uma variável.
	O processo de modificação de um tipo de dado para se tornar um tipo de dado diferente, isso pode ser implícito ou explícito.
	Um tipo de dado Java primitivo (2 bytes) que contém números inteiros em um intervalo mais curto que um int.
	Um tipo Java primitivo de um bit que pode conter o valor verdadeiro ou falso.

Tente/solucione:

- Escreva um programa que determinará a base e a altura de um triângulo e calculará e exibirá a área do triângulo usando a fórmula abaixo.

$$A = \frac{1}{2}bh$$

- Escreva as seguintes fórmulas matemáticas em Java. Você precisará usar métodos da classe Math, bem como aninhar métodos e parênteses para impor a ordem das operações para calcular corretamente a resposta. Suponha que todas as variáveis nas fórmulas já foram declaradas e inicializadas.

- $a = \frac{\sqrt{x^8-6}}{4}$

- $b = x^y - 6^x$

- $c = 4\cos\left(\frac{z}{5}\right) - \sin x^2$

- $d = x^4 - \sqrt{6x - y^3}$

- $e = \frac{1}{y - \frac{1}{x-2y}}$

- $f = 7(\cos(\sqrt{5 - \sin\sqrt{3x-4}}))$

- Um ônibus contém 45 pessoas. A escola usará um ônibus somente se puder preenchê-lo completamente. O restante das pessoas será transportado em vans. Escreva um programa que determinará o número de pessoas inscritas para ir a uma viagem ao campo. O programa deve imprimir o número de ônibus necessários e o número total de pessoas que deverão ser transportadas em vans.

4. Escreva verdadeiro ou falso nos espaços em branco no programa a seguir para mostrar o valor true_false da variável booleana enquanto o programa é executado.

```
int i=5;

int j=6;

boolean true_false;

true_false=(j<5); _____

true_false=(j>3); _____

true_false=(j<i); _____

true_false=(i<5); _____

true_false=(j<=5); _____

true_false=(6<6); _____

true_false=(i!=j); _____

true_false=(i==j || i<50); _____

true_false=(i==j && i<50); _____

true_false=(i>j || true_false && j>=4); _____

true_false=(!(i<2 && j==5)); _____

true_false=!true_false; _____
```

5. Explique por que cada uma das declarações na segunda lista está errada.

```
boolean gameOver = false;
int students=50,classes=3;
double sales_tax;
short number1;
```

```
int 2beOrNot2be;
float price index;
double lastYear'sPrice;
long class;
```

6. Explique por que cada uma das declarações na segunda lista não segue as convenções de nomes de variáveis.

```
int cadence=3, speed=55,
gear=4;
final double SALES_TAX=.06;
double gearRatio=.5;
int currentGear=5;
```

```
int c=3,s=55,g=4;
final double salesTax=.06;
double gearratio=.05,Gear=4;
int current_gear;
```