**Variáveis, Tipos de Dados e Operadores Matemáticos em Java**

**Conceituação e criação**

**O que é uma variável?**

Um espaço na memória do computador, onde se pode guardar valores.

Existem 4 tipos:

- Instância: objeto

- Classe: classe

- Local: dentro de métodos

- Parâmetro: Na assinatura do método

**Padrão de definição:**

<?visibilidade?><?modificador?>tipo nome<?=valorInicial?>;

Visibilidade: “public”, “protected” e “private”

Modificador: “static” e “final”

Tipo: tipo de dado (Por isso a linguagem é fortemente tipada, pois temos que prover.)

Nome: nome que é fornecido a variável;

Valor Inicial: um valor inicial, caso deseje.

**Convenções e regras para variáveis:**

* Não devem começar com números;
* Embora permitido, “$” e “\_” devem ser evitados;
* São case-sensitive (vai diferenciar maiúsculos e minúsculos);
* Sem espaços;
* Não pode ser as palavras reservadas do Java: abstract, continue, for, new, switch, assert, default, goto, package, synchronizedboolean, do, if, private, this, break, Double, case, Implements, throw, protected, byte, else, import, public, throws, entre outras já utilizadas como comandos...

**Boas práticas para criação de variáveis:**

* Sempre começar com letra minúscula;
* Nomes expressivos;
* Notação camelo;
* Quando constante (final) maiúscula e separada por “\_”.

**Tipos de dado**

**O que são tipos de dados?**

São os valores e consequentemente operações que as variáveis podem assumir e sofrer, respectivamente.

**Tipificação**:

* Estática (forte) VS dinâmica (fraco)

- Estática: Quando a linguagem obriga você a definir o tipo de sua variável já no meio de sua criação.

- Dinâmica você não é obrigado a definir o tipo da variável no meio da sua criação.

* Primitivo X Composto

- Primitivo: Atômicos e homogênicos, como numéricos e textuais.

- Composto: Está muito ligado a OO, é heterogêneo, composto por vários tipos de dados primitivos.

**Opções de tipos:**

* Textual

OBS: Quando utilizamos caracteres de forma isolada utilizamos aspas simples(‘ ‘), caso contrário, aspas duplas(“ ”).

Exemplo:

- char: caracteres de 16-bit unicode -> char c = ‘\u0084’; ou char c = ‘T’;

- String: um tipo “especial” -> String s = “T”;

* Numeral

Exemplo:

- byte: –128 até 127 -> byte b = 15; (pequeno)

- short: -32.768 até 32.767 -> short s = -15785; (médio)

- int: -2.147.483.648 até 2.147.483.647 -> int i = 8515785; (grande)

- long: -9.223.372.036.854.775.808 até 9.223.372.036.854.775.808 -> long l = 5985475262L;

- float: +/-3.40282347E+38F -> float f = 3.14... (f);

- double: +/- 1.72265125621126+35 -> double d = 3.14... (d);

* Lógico

Exemplo:

- boolean: true e false -> boolean s = false;

* Objeto

(Será explicado na aula de OO)

**Boas práticas sobre os tipos de dados:**

* Não é adequado utilizar tudo como String (CPF, data, enfim.)

**Operadores Aritméticos/Matemáticos**

**O que são operadores aritméticos?**

“São símbolos especiais quais são capazes de realizar ações específicas em um, dois ou mais operandos e, em seguida, retornar um resultado.”

Tipos:

* Pós-fixado: exp++ ou exp— (++ soma e – subtrai, usa a expressão para depois somar ou subtrair)
* Prefixado: ++exp ou --exp ((++ soma e – subtrai, soma ou subtrai a expressão)
* Aritméticos: +, -, \*, / e %
* Atribuição: =, +=, -=, \*=, /= e %=

**Conversões/Casting**

**O que é casting?**

É a transformação de uma determinada variável de tipo menos específico para um tipo mais específico ou vice-versa.

Tipos:

* Upcast (implícito) – promoção de um tipo de dado, melhorando ele.
* Downcast (explícito) – rebaixar o tipo de dado.

