

Conceituação e criação.

Conceituação :

“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?>;

V: “public”, “protected” e “private”

M: “static” e “final”

T: tipo de dado

N: nome que é fornecido a variável

VI: um valor inicial caso se deseje

Convenções e regras:

- Não devem começar com números;
- Embora permitido, “\$” e “_” devem ser evitados;
- São case-sensitive(vai diferenciar maiúsculas de minúsculas);
- Sem espaços;
- Não pode ser as palavras reservadas do Java;

Boas práticas:

- Sempre começar com letra minúscula;
- Nomes expressivos(um nome claro do que a variante faz);
- Notação camelo;
- Quando constante(final) maiúscula e separada por “_”;

Tipos de dados

Conceituação :

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

Tipificação:

- Estática(forte) e Dinâmica(fraco)
- Primitivo ou Composto

Opções de tipos:

- Textual
- Numeral
- Lógico
- Objeto

Utilização

Tipo de dado	Valor default
byte	0
short	0
int	0
long	0L
float	0.0f
double	0.0d
char	"\u0000"
String(e objetos)	null
boolean	false

Boas práticas:

Usar de forma adequada cada tipo de dado para cada informação

Operadores Aritméticos

Conceituação :

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

Tipificação:

- pós-fixado: `exp++` ou `exp--`
- prefixado: `++exp` ou `--exp`
- aritmético: `+`, `-`, `*`, `/` e `%`
- atribuição: `=`, `+=`, `-=`, `*=`, `/=` e `%=`

Utilização

Operador	Precedência
Pós-fixado	<code>exp++</code> , <code>exp--</code>
Prefixado	<code>++exp</code> , <code>--exp</code>
Multiplicativo	<code>*</code> , <code>/</code> , <code>%</code>
Aditivo	<code>+</code> , <code>-</code>
Atribuição	<code>=</code> , <code>+=</code> , <code>-=</code> , <code>*=</code> , <code>/=</code> , <code>%=</code>

Conversões (casting)

Conceituação :

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

Tipificação:

- Upcast(implícito)
- Downcast (explícito)

Utilização

	byte	short	char	int	long	float	double
byte		U - I	char	U - I	U - I	U - I	U - I
short	D - E		char	U - I	U - I	U - I	U - I
char	D - E	D - E		U - I	U - I	U - I	U - I
int	D - E	D - E	D - E		U - I	U - I	U - I
long	D - E	D - E	D - E	D - E		U - I	U - I
float	D - E	D - E	D - E	D - E	D - E		U - I
double	D - E	D - E	D - E	D - E	D - E	D - E	

U – I : Upcast – Implícito D – E: Downcast - Explícito