



Técnicas de Programação e Algoritmos - TPA

– Tipos de Dados
– JAVA –

Variáveis de Tipos Primitivos

Em Java, uma variável deve:

- ✓ ser declarada antes de ser usada
- ✓ ter um tipo definido (o tipo não muda)
- ✓ iniciar o valor da variável antes de usá-la
- ✓ ser usada dentro do escopo (método ou bloco)

Declaração: `<tipo da variável> <nome da variável>;`

Declaração e atribuição: `<tipo> <nome> = <valor>;`

<i>Tipo</i>	<i>Tamanho (bits)</i>	<i>Valor Mínimo</i>	<i>Valor Máximo</i>	<i>Sem Sinal</i>
boolean	1	false	true	X
char	16	0	$2^{16} - 1$	X
byte	8	-2^7	$2^7 - 1$	
short	16	-2^{15}	$2^{15} - 1$	
int	32	-2^{31}	$2^{31} - 1$	
long	64	-2^{63}	$2^{63} - 1$	
float	32			
double	64			

Tipos de dados em Java

- **int**: Formado por 32 bits, suporta valores entre -2.147.483.648 e 2.147.483.648;
- **long**: Formado por 64 bits, é usado quando você realmente precisa guardar valores muito altos;
- **float**: Formado por 32 bits, é usado para guardar valores em ponto flutuante que possuam até 7 casas decimais;
- **double**: Formado por 64 bits, é utilizado quando se deseja armazenar valores em ponto flutuante com até 15 casas decimais. Um valor em ponto flutuante no Java é, por default, um valor double;
- **boolean**: Possui apenas os valores lógicos true (verdadeiro) e false (falso);
- **char**: Formado por 16 bits, armazena caracteres Unicode (além da tabela ASCII, possibilita o uso de caracteres que possam ser utilizados em qualquer idioma do mundo).

Operadores

■ Aritméticos

Função	Sinal
Adição	+
Subtração	-
Multiplicação	*
Divisão	/
Resto da divisão	%
Incremento	++
Decremento	--

Operadores

■ Relacionais

Função	Sinal
Igual	==
Diferente	!=
Maior que	>
Maior ou igual a	>=
Menor que	<
Menor ou igual a	<=

Operadores

- Lógicos

Função	Sinal
E	&&
OU	
Não	!

Conversão de tipos

Supondo a variável x	Converter em	y recebe o valor convertido
✓ Entre tipos numéricos		
int x = 10	float	float y = (float) x
int x = 10	double	double y = (double) x
float x = 10.5	int	int y = (int) x
✓ De string para numéricos		
String x = "10"	int	int y = Integer.parseInt(x)
String x = "20.5"	float	float y = Float.parseFloat(x)
String x = "20.5"	double	double y = Double.parseDouble(x)
✓ De numéricos para string		
int x = 10	String	String y = Integer.toString(x) ou String y = String.valueOf(x)
float x = 10.5	String	String y = Float.toString(x) ou String y = String.valueOf(x)
double x = 10.5	String	String y = Double.toString(x) ou String y = String.valueOf(x)

Inserção de Comentários

```
// Comentários em uma única linha
```

```
/* Comentários em  
  * várias linhas  
  */
```

```
/** Comentários inseridos no formato reconhecido  
  * por um utilitário de documentação chamado javadoc  
  * fornecido pela Sun junto com o JDK  
  */
```


Identificadores

As regras para nomeação de identificadores (variáveis, nomes de função, classes ou label) seguem a seguinte regra:

- ✓ nomes devem começar com letra ou os caracteres _ ou \$
- ✓ os caracteres seguintes podem conter números, letras, _ ou \$

Veja exemplos de nomes de identificadores:

valor	// válido
\$preco	// válido
20itens	// inválido
_teste	// válido
INT	// válido

Observação: O Java considera diferença entre maiúsculas e minúscula.

Recursos básicos da linguagem

```
// Declaração de variáveis
int num1 = 0, op = 0;
double valor;
String usuario;

// Declaração de constantes
final double pi = 3.1416;
```

- ❑ Tipos primitivos são escritos sempre com letras minúsculas.
- ❑ As variáveis devem ser inicializadas na declaração
- ❑ O Java disponibiliza algumas classes que podem ser utilizadas como tipos (como a String no exemplo acima).