

# Orientação a objetos



Dominar os identificadores, palavras chave, tipos e operadores  
Saber utilizar as estruturas de programação  
Variáveis, Tipos, Operadores e Estruturas de repetição

- Variáveis
- Tipos Primitivos
- Estruturas de Seleção
- Estruturas de Repetição
- Conversões
- Operadores

Palavras-chave, também conhecidas como palavras reservadas da linguagem, são palavras que não podem ser usadas como identificadores, ou seja, não podem ser usadas para representar variáveis, classes ou nomes de métodos.

abstract	continue	for	new	switch
assert	default	goto	package	synchronized
boolean	do	if	private	this
break	double	implements	protected	throw
byte	else	import	public	throws
case	enum	instanceof	return	transient
catch	extends	int	short	try
char	final	interface	static	void
class	finally	long	strictfp	volatile
const	float	native	super	while

São armazenadas na memória RAM da máquina. As variáveis podem guardar dados de tipos numéricos, textos, booleanos e referências de objetos. O nome de uma variável não pode começar com um número e não pode ser uma palavra reservada.

**Declaração** - Tipo da variável mais o nome da variável.

```
int numero;  
double media;
```

A declaração de uma variável pode ser realizada em qualquer linha. Não é necessário declarar todas as variáveis no começo do bloco.

```
int numero = 30;  
System.out.println ( numero );  
  
double numero2= 87.3;  
System.out.println ( numero2);
```

### **Variáveis de instância ou atributos**

As variáveis de instâncias são definidas dentro de um classe, e só são inicializadas quando a classe é instanciada..

A partir do Java 10 foi introduzido o recurso de inferência de tipos para variáveis, após atribuição do valor seu tipo não pode ser modificado.

```
public class Variaveis {  
    public static void main(String[] args) {  
        var nome = "José";  
        var numero = 200;  
  
        System.out.println("Nome:" + nome);  
        System.out.println("Número:" + numero);  
    }  
}
```

Um variável do tipo primitivo armazena um valor do seu tipo que foi declarado. Abaixo uma lista do tipos principais primitivos. As variáveis devem ser declaradas respeitando-se a sintaxe básica “**tipo nomeVariavel**” esta convenção é chamada de **CamelCase**.

Tipo	Tamanho
byte	1 byte
short	2 bytes
int	4 bytes
long	8 bytes
float	4 bytes
double	8 bytes
boolean	1 bit
char	2 bytes

O tipo primitivo char armazena apenas um caractere. Quando é necessário armazenar um texto, devemos utilizar o tipo String.

1) Criar uma nova pasta com o nome **Aula2** e abrir no vscode

- Criar a classe **ExercicioVariaveis**
- Declarar as variáveis: nome, idade, peso e altura

Deverá ser impresso no console o seguinte resultado:

```
O funcionario João tem:  
idade:20  
altura:1.75  
peso:52.5
```

2) Criar uma nova classe com o nome **CalculadoraMedia**. Criar 4 variáveis com o nome nota1, nota2, nota3 e nota 4 com valores iniciais qualquer e exibir a média no console

```
A média é:9.0
```

```
public class ExercicioVariaveis {  
    public static void main(String[] args){  
        int idade = 20;  
        double altura = 1.75;  
        double peso = 52.5;  
        System.out.println("O funcionario João tem:" + "\nidade:" + idade + "\naltura:" + altura + "\npeso:" + peso);  
    }  
}
```

```
public class CalculaMedia {  
    public static void main(String[] args) {  
        double nota1=10, nota2=9, nota3=8, nota4=9;  
        System.out.println("A média é:" + (nota1 + nota2 + nota3 + nota4)/4);  
    }  
}
```



É possível atribuímos o valor de um tipo de variável a uma de outro tipo. Conversões de tipos primitivos boolean não podem ser feitas.

```
public class Conversao {  
    public static void main(String[] args) {  
        int a = 100;  
        float b = a;  
  
        double c = 4.19;  
        // int d = c; não executa  
        int d = (int) c;  
  
        float e = (float) c;  
        float f = 6.18f;  
  
        System.out.println(b + "\n" + c + "\n" + d + "\n" + e + "\n" + f);  
    }  
}
```

Conversão implícita. A variável **b** de um tipo maior receberá o valor da variável **a**.

casting de um double para um inteiro.

Uma variável **float** não pode receber um **double** sem conversão pois todos os literais com ponto flutuante são **double**. A letra **f** indica que a variável é um do tipo **float**.

### Casting Possíveis

Abaixo os tipos possíveis de casting em Java. A indicação impl. Quer dizer que o cast é implícito e automático, ou seja, você não precisa indicar o cast explicitamente. Além disso, o tipo boolean não pode ser convertido para outro tipo.

PARA:	byte	short	char	int	long	float	double
DE:							
byte	----	Impl.	(char)	Impl.	Impl.	Impl.	Impl.
short	(byte)	----	(char)	Impl.	Impl.	Impl.	Impl.
char	(byte)	(short)	----	Impl.	Impl.	Impl.	Impl.
int	(byte)	(short)	(char)	----	Impl.	Impl.	Impl.
long	(byte)	(short)	(char)	(int)	----	Impl.	Impl.
float	(byte)	(short)	(char)	(int)	(long)	----	Impl.
double	(byte)	(short)	(char)	(int)	(long)	(float)	----

Declare duas variáveis do tipo **double** e realize sua soma.

Em seguida, realize o casting da operação para **int** para realizar sua divisão.

```
public class Casting {  
  
    public static void main(String[] args) {  
        double a = 10.5;  
        double b = 2;  
        double soma = a + b;  
        System.out.println("Soma:" + soma);  
  
        int divisao = (int)(a + b)/2;  
        System.out.println(divisao);  
    }  
}
```

Os operadores aritméticos seguem as mesmas regras seguidas em álgebra. Quando existem vários operadores de mesma precedência, ela é avaliada da esquerda pra direita.

Operador	Símbolo	Precedência
Multiplicação	*	1º
Divisão	/	1º
Resto	%	1º
Soma	+	2º
Subtração	-	2º

**Importante:** a precedência também é válida para parênteses mais internos quando presente, assim como na álgebra.