

Laboratório de Desenvolvimento de Algoritmos



- ✓ Variáveis
- ✓ Entrada e saída

Dicas que te tornarão um programador melhor



Comente sempre seu código.
Verá que será útil no futuro.

Dicas que te tornarão um programador melhor

- ✓ Termine (quase...) todas as linhas com PONTO E VÍRGULA (;)
- ✓ Quando ocorrer um erro de compilação, dê um duplo clique sobre a mensagem de erro para destacar o comando errado no programa e verifique também a linha anterior, que pode ser a responsável pelo erro, especialmente se faltar o (;)
- ✓ Como convenção, todos os nomes de classe (identificador) em Java iniciam com uma letra maiúscula.
- ✓ Use comentários, iniciados por // ou /* ... */



Dicas que te tornarão um programador melhor



Exemplos e exercícios resolvidos no GitHub

<https://github.com/msanches/LDA>

O Que é GitHub e Para Que é Usado?

<https://www.hostinger.com.br/tutoriais/o-que-github/>

Tutorial do GIT Básico – Introdução ao GIT

<https://www.hostinger.com.br/tutoriais/tutorial-do-git-basics-introducao/>



Hello World – do GitHub

<https://guides.github.com/activities/hello-world/>



GitHub Training & Guides

<https://www.youtube.com/githubguides>

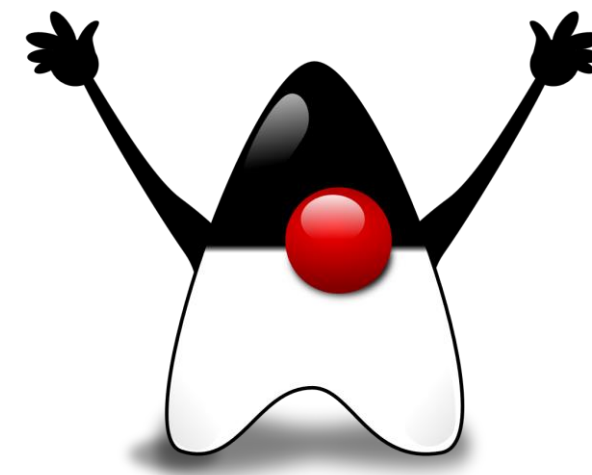


Git e Github para iniciantes

<https://www.udemy.com/course/git-e-github-para-iniciantes/>

Conceitos abordados nesta aula

A proposta desta aula é apresentar os conceitos de variáveis e entrada e saída de dados na linguagem Java.



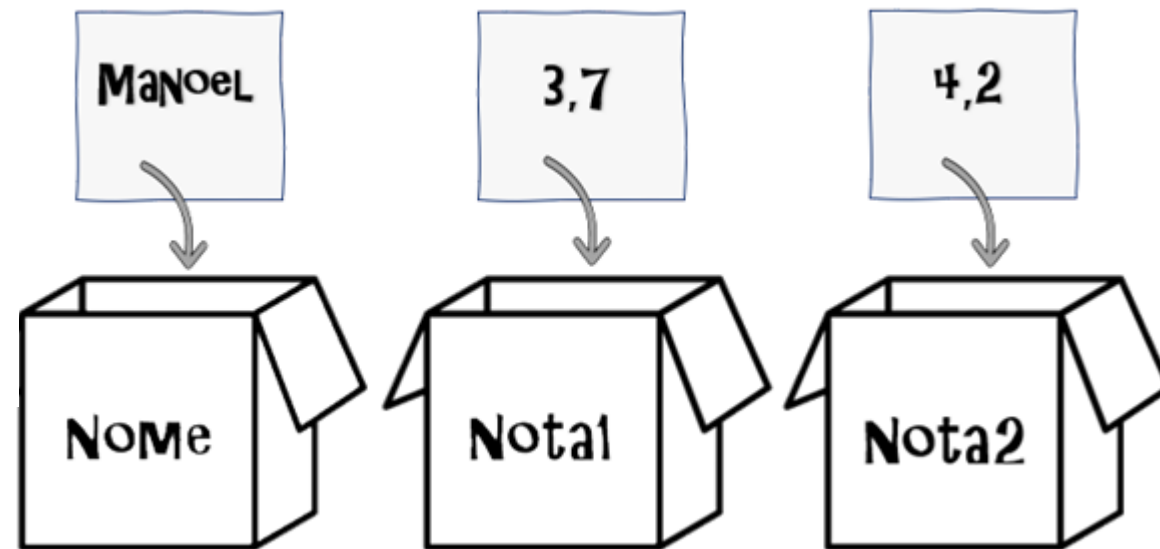
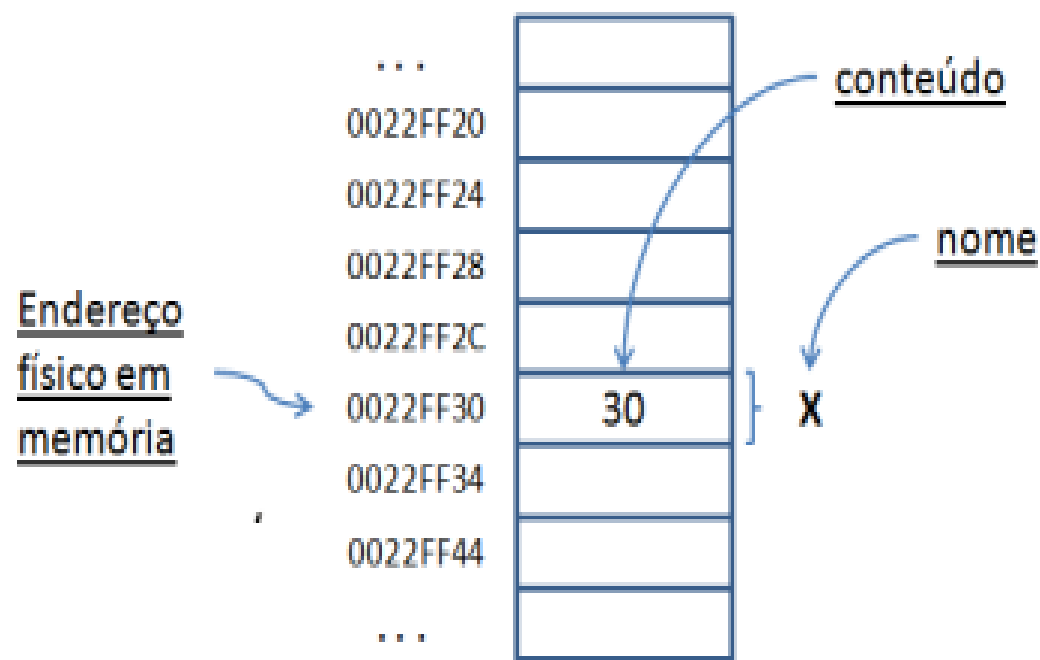
Estrutura de um algoritmo em pseudocódigo

```
algoritmo nomeDoAlgoritmo
    declaração de variáveis utilizadas
inicio
    comandos para resolver o problema
fim
```



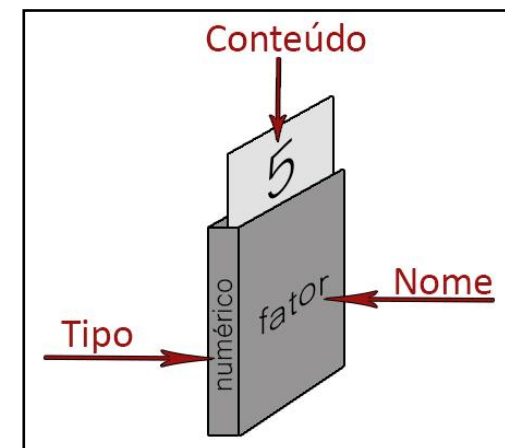
Variáveis

- ✔ Espaço de memória que pode receber um valor e sofrer alteração no decorrer do algoritmo/tempo.
- ✔ Toda variável tem um nome único que a identifica (**identificador**), um valor e o tipo correspondente à informação a ela atribuída.



Variáveis

- ✓ Nos algoritmos, cada variável corresponde a uma posição de memória.
- ✓ Embora uma variável possa assumir diferentes valores, ela só pode armazenar um valor a cada instante.
- ✓ Uma variável possui três atributos:
 - um nome (ou **identificador**),
 - um tipo de dado e
 - a informação por ela guardada.
- ✓ Cada linguagem de programação estabelece suas próprias regras de formação de nomes de variáveis.



Declaração de variáveis no pseudocódigo

- ✓ **inteiro: a, b, c**

para armazenar valores inteiros, independente do tamanho e sinal

- ✓ **caractere: d, e**

para armazenar strings ou caracteres individuais

- ✓ **real: h, g**

para armazenar qualquer valor com casas decimais, independente da precisão ou tamanho

- ✓ **lógico: i, j**

para armazenar valores lógicos (verdadeiro ou falso)



Declaração de variáveis no pseudocódigo

- ✓ **inteiro: a, b, c**

{..., -3, -2, -1, 0, 1, 2, 3, ...}

- ✓ **caractere: d, e**

{a, b, c, ..., %, #, \$}

- ✓ **real: h, g**

{..., $-\pi$, -0.8, 0, 1.1, $\frac{3}{4}$, $\sqrt{2}$ }

- ✓ **lógico: i, j**

{True, False}



Declaração de variáveis na linguagem Java

- ✓ **int a, b, c;**

para armazenar valores inteiros, independente do tamanho e sinal

- ✓ **char d; ou String e;**

para armazenar caracteres individuais ou strings

- ✓ **float h, g;**

para armazenar qualquer valor com casas decimais, independente da precisão ou tamanho

- ✓ **boolean i, j;**

para armazenar valores lógicos (verdadeiro ou falso)



Tipos de dados primitivos em Java

Classificação	Tipo	Descrição
Lógico	boolean	Pode possuir valores true (verdadeiro) ou false (falso)
Inteiro	byte	Abrange de -128 a 127 (8 bits)
	short	Abrange de -32768 a 32767 (16 bits)
	int	Abrange de -2.147.483.648 a 2.147.483.647 (32 bits)
	long	Abrange de -2^{63} a $2^{63} - 1$ (64 bits)
Ponto flutuante	float	Abrange de $-3.4028E+38$ a $3.4028E+38$ (32 bit) com precisão simples
	double	Abrange de $-1.7976E+308$ a $1.7976E+308$ (32 bit) com precisão dupla
Caractere	char	Pode armazenar um caractere Unicode (16 bits) ou um inteiro entre 0 e 65535

O **double** é a opção padrão para a representação de números em ponto flutuante, devido a baixa precisão do tipo **float**.

Comandos de entrada e saída

✓ Comandos de saída em Pseudocódigo:

```
algoritmo exemplo1
  inicio
    escreva("Olá mundo!!!")
  fim
```

→ Exibe mensagem na tela

✓ Comandos de entrada em Pseudocódigo:

```
algoritmo exemplo2
  inicio
    literal nome
    escreva("Digite o seu nome: ")
    leia(nome)
    escreva("Olá" + nome)
  fim
```

→ Lê um valor digitado pelo usuário

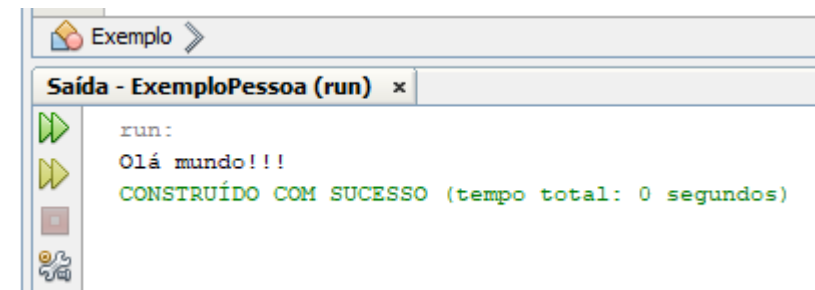
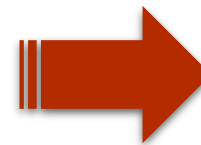


Comandos de entrada e saída

Comandos de saída em Java:

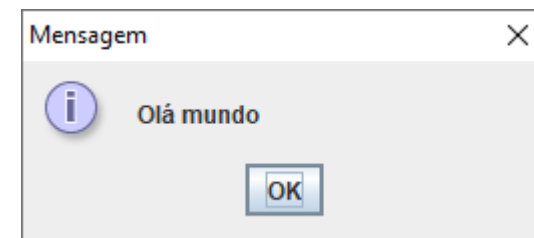
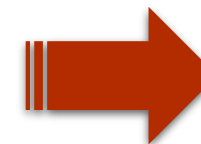
✓ Modo texto:

```
public class Exemplo {  
    public static void main(String[] args) {  
        System.out.println("Olá Mundo!!!");  
    }  
}
```



✓ Modo gráfico:

```
public class Exemplo {  
    public static void main(String[] args) {  
        JOptionPane.showMessageDialog(null, "Olá Mundo!!!");  
    }  
}
```

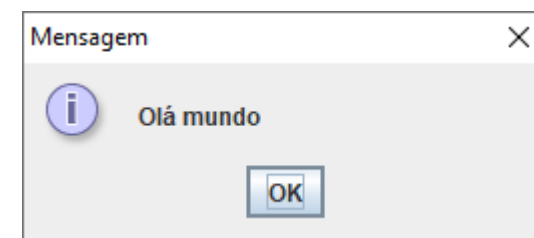
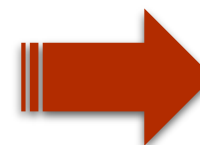


Comandos de entrada e saída

Comandos de saída em Java:

- ✓ Modo gráfico:

```
public static void main (String [] args){  
    JOptionPane.showMessageDialog(null, "Olá mundo");  
}
```



Chamada para o método
showMessageDialog da classe
JOptionPane

Exibe a caixa de diálogo no centro da tela.

Para utilizar o comando de entrada no modo texto, é necessário importar o pacote:

```
import javax.swing.JOptionPane;
```

Exemplo de aplicação

1- Elaborar um programa que exiba na tela a mensagem: “O Sol já nasceu em Sampa!”:

a) Em pseudocódigo

b) No modo texto

c) No modo gráfico

```
1 algoritmo exemplo1
2     inicio
3         escreva("O Sol já nasceu em Sampa!")
4     fim
5
```



Exemplo de aplicação

1- Elaborar um programa que exiba na tela a mensagem: “O Sol já nasceu em Sampa!”:

a) Em pseudocódigo

```
Exemplo.java x
Source History
1 package com.mycompany.exemplo;
2 /**
3  * Programa: Exibe mensagem na tela
4  * @author marco
5  */
6 public class Exemplo {
7     //O método main inicia a execução do aplicativo Java
8     public static void main(String[] args) {
9         System.out.println("O Sol já nasceu em Sampa");
10    } //fim do método main
11 } //fim da classe Exemplo
```



Exemplo de aplicação

1- Elaborar um programa que exiba na tela a mensagem: “O Sol já nasceu em Sampa!”:

a) Em pseudocódigo

```
Exemplo.java x
Source History
1 package com.mycompany.exemplo;
2 import javax.swing.JOptionPane;
3 /**
4  * Programa: Exibe mensagem na tela
5  * @author marco
6  */
7 public class Exemplo {
8     //O método main inicia a execução do aplicativo Java
9     public static void main(String[] args) {
10         JOptionPane.showMessageDialog(null, "O Sol já nasceu em Sampa!");
11     } //fim do método main
12 } //fim da classe Exemplo
```







Comandos de entrada e saída

O método showMessageDialog em Java:

```
public static void showMessageDialog  
    (Component parentComponent,  
     Object message, String title, int messageType)
```

As constantes que especificam o ícone da janela são :

- ✓ JOptionPane.ERROR_MESSAGE
- ✓ JOptionPane.INFORMATION_MESSAGE
- ✓ JOptionPane.WARNING_MESSAGE
- ✓ JOptionPane.QUESTION_MESSAGE
- ✓ JOptionPane.PLAIN_MESSAGE





Icon	Code	IDE Value
No icon	JOptionPane.PLAIN_MESSAGE	-1
	JOptionPane.ERROR_MESSAGE	0
	JOptionPane.INFORMATION_MESSAGE	1
	JOptionPane.WARNING_MESSAGE	2
	JOptionPane.QUESTION_MESSAGE	3

Comandos de entrada e saída

O método showMessageDialog em Java:

```
public static void showMessageDialog  
    (Component parentComponent,  
     Object message, String title, int messageType)
```

As constantes que especificam o ícone da janela são :

Icon	Code	IDE Value
No icon	JOptionPane.PLAIN_MESSAGE	-1
	JOptionPane.ERROR_MESSAGE	0
	JOptionPane.INFORMATION_MESSAGE	1
	JOptionPane.WARNING_MESSAGE	2
	JOptionPane.QUESTION_MESSAGE	3

Exemplo



```
1 package com.mycompany.exemplo;
2 import javax.swing.JOptionPane;
3 /**
4  * Programa: Exibe mensagem na tela
5  * @author marco
6  */
7 public class Exemplo {
8     //O método main inicia a execução do aplicativo Java
9     public static void main(String[] args) {
10         JOptionPane.showMessageDialog(null, "Olá Mundo", "Olá Mundo",
11             JOptionPane.INFORMATION_MESSAGE);
12     } //fim do método main
13 } //fim da classe Exemplo
```

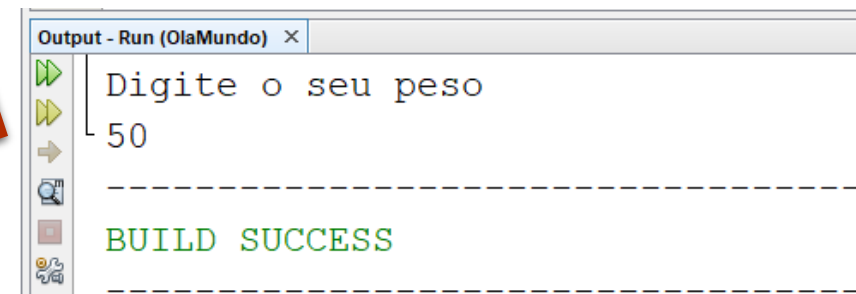
The image shows a Java IDE window titled 'Exemplo.java'. The code defines a package 'com.mycompany.exemplo', imports 'javax.swing.JOptionPane', and contains a class 'Exemplo' with a 'main' method. The 'main' method calls 'JOptionPane.showMessageDialog' with the message 'Olá Mundo', the title 'Olá Mundo', and 'JOptionPane.INFORMATION_MESSAGE'. To the right of the code editor, a small Java application window titled 'Olá Mundo' is displayed. It features an information icon (i) and an 'OK' button.

Comandos de entrada e saída

Comandos de entrada em Java:

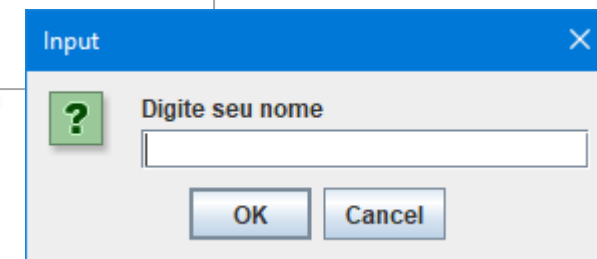
✓ Modo texto:

```
public static void main(String[] args) {  
    Scanner leia = new Scanner(System.in);  
    System.out.println("Digite o seu peso");  
    float peso = leia.nextFloat();  
}
```



✓ Modo gráfico:

```
public static void main(String[] args) {  
    String nome = JOptionPane.showInputDialog(null, "Digite seu nome");  
}
```

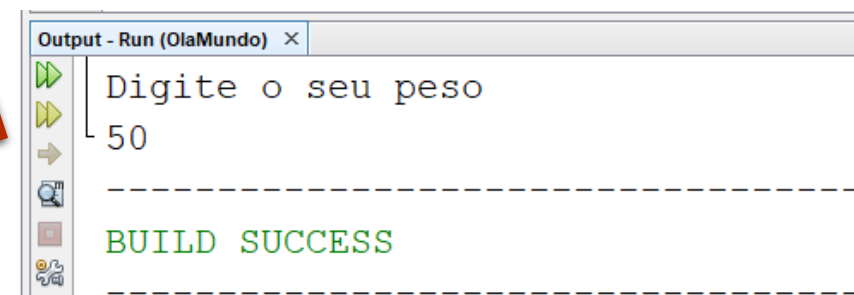


Comandos de entrada e saída

Comandos de entrada em Java:

- ✓ Modo texto:

```
public static void main(String[] args) {  
    Scanner leia = new Scanner(System.in);  
    System.out.println("Digite o seu peso");  
    float peso = leia.nextFloat();  
}
```



- ✓ Para utiliza o comando de entrada no modo texto, é necessário importar o pacote:

```
import java.util.Scanner;
```

- ✓ Cada tipo de dado primitivo exige uma chamada do método para retornar o valor especificado na entrada de dados, sempre seguindo o formato **nextTipoDado()**.

Comandos de entrada e saída

Comandos de entrada em Java:

- ✔ Cada tipo de dado primitivo exige uma chamada do método para retornar o valor especificado na entrada de dados, sempre seguindo o formato **nextTipoDado()**.

```
Scanner leia = new Scanner(System.in);  
float peso = leia.nextFloat();  
double salario = leia.nextDouble();  
int idade = leia.nextInt();  
byte valor1 = leia.nextByte();  
long valor2 = leia.nextLong();  
boolean b1 = leia.nextBoolean();  
String nome = leia.nextLine();
```

Observe o tipo de dado
para String!!!

Exemplo de aplicação

2- Elaborar um programa que solicite seu nome e exiba na tela a mensagem: “Olá, Fulano!”

```
1 algoritmo exemplo2
2     inicio
3         literal nome
4         escreva("Digite o seu nome: ")
5         leia(nome)
6         escreva("Olá" + nome)
7     fim
8
```



Exemplo de aplicação

2- Elaborar um programa que solicite seu nome e exiba na tela a mensagem: “Olá, Fulano!”

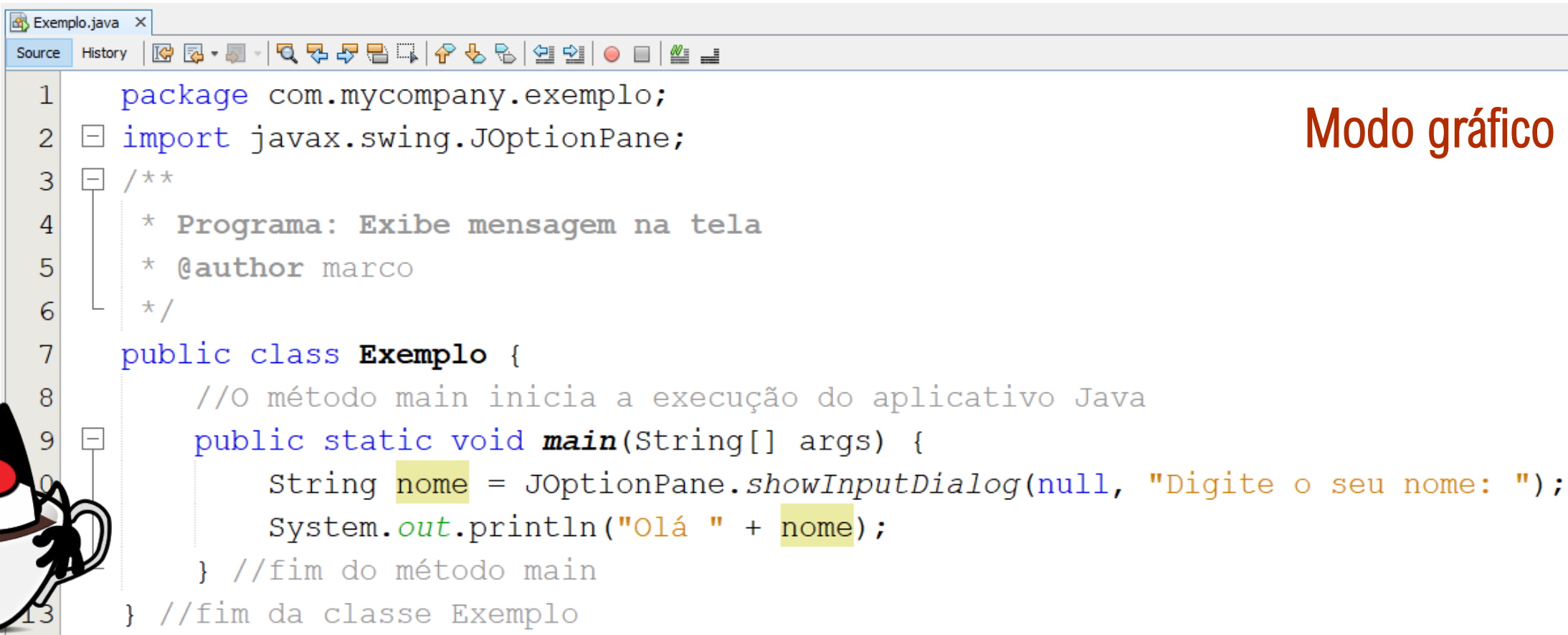



```
Exemplo.java x
Source History
1 package com.mycompany.exemplo;
2 import java.util.Scanner;
3 /**
4  * Programa: Exibe mensagem na tela
5  * @author marco
6  */
7 public class Exemplo {
8     //O método main inicia a execução do aplicativo Java
9     public static void main(String[] args) {
10         Scanner sc = new Scanner(System.in);
11         System.out.println("Digite o seu nome: ");
12         String nome = sc.nextLine();
13         System.out.println("Olá " + nome);
14     } //fim do método main
15 } //fim da classe Exemplo
```

Modo texto

Exemplo de aplicação

2- Elaborar um programa que solicite seu nome e exiba na tela a mensagem: “Olá, Fulano!”



```
1 package com.mycompany.exemplo;
2 import javax.swing.JOptionPane;
3 /**
4  * Programa: Exibe mensagem na tela
5  * @author marco
6  */
7 public class Exemplo {
8     //O método main inicia a execução do aplicativo Java
9     public static void main(String[] args) {
10         String nome = JOptionPane.showInputDialog(null, "Digite o seu nome: ");
11         System.out.println("Olá " + nome);
12     } //fim do método main
13 } //fim da classe Exemplo
```

Modo gráfico

Principais operadores em Java

Aritméticos

Operação	Operador	Expressão algébrica	Expressão Java
Adição	+	$x + y$	$x + y$
Subtração	-	$x - y$	$x - y$
Multiplicação	*	xy	$x * y$
Divisão	/	x/y ou $\frac{x}{y}$ ou $x : y$	x / y
Resto da divisão	%	$x \bmod y$	$x \% y$
Incremento	++	$x + 1$	$x++$
Decremento	--	$x - 1$	$x--$

Principais operadores em Java

IMPORTANTE!!



As operações aritméticas em Java obedecem as mesmas regras da matemática com relação à precedência dos operadores e parênteses.

Portanto:

- ✓ As operações são resolvidas a partir dos parênteses mais internos até os mais externos.
- ✓ Primeiro resolvemos as multiplicações, divisões e módulos.
- ✓ Por fim, resolvemos as adições e subtrações.



Principais operadores em Java

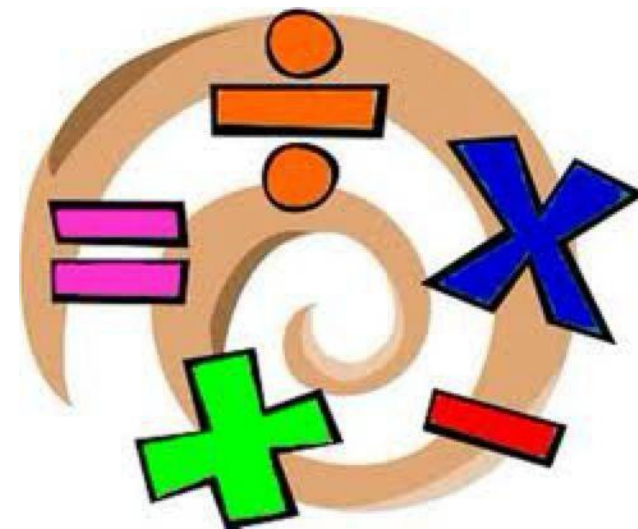
Exemplos:

1- $10 + 20 \times 30 = ??$

2- $(10 + 20) \times 30 = ??$

3- $42 \div 30 - 3 = ??$

4- $(10 + 2) \times 6 - 2 = ??$



Principais operadores em Java

Relacionais

Operador	Operador em Java	Exemplo	Significado
Igualdade	<code>==</code>	<code>x == y</code>	x é igual a y
Diferença	<code>!=</code>	<code>x != y</code>	x é diferente de y
Maior que	<code>></code>	<code>x > y</code>	x é maior que y
Maior ou igual a	<code>>=</code>	<code>x >= y</code>	x é maior ou igual a y
Menor que	<code><</code>	<code>x < y</code>	x é menor que y
Menor ou igual a	<code><=</code>	<code>x <= y</code>	x é menor ou igual a y

Principais operadores em Java

Lógicos

Operador	Operador em Java	Exemplo	Significado
E lógico ou AND	&&	x && y	x E y
OU lógico ou OR	 	x y	x OU y
Negação ou NOT	!	!x	NÃO x

Conversões de tipos em Java

Supondo a variável x	Conversão para	y recebe o valor convertido	Resultado
int x = 10	float	float y = (float) x	10.0
float x = 10.5f	int	int y = (int) x	10
String x = "10"	int	int y = Integer.parseInt(x)	10
String x = "10.5"	float	float y = Float.parseFloat(x)	10.5
int x = 10 float x = 10.5	String	String y = String.valueOf(x)	"10" "10.5"

Exemplo de aplicação

3- Elaborar um programa que solicite os dados de altura (em metros) e peso (em kg) de uma pessoa, calcule e visualize seu IMC (Índice de Massa Corporal).

É dada a fórmula:

$$\text{IMC} = \text{peso} / \text{altura}^2$$

```
imc.alg x
1 algoritmo imc
2     inteiro peso, altura, imc
3     inicio
4         escreva("Digite o seu peso em kg: ")
5         leia(peso)
6         escreva("Digite a sua altura em m: ")
7         leia(altura)
8         imc = peso / (altura*altura)
9         escreva("O seu IMC é: " + imc + "kg/m²")
10    fim
```



Exemplo de aplicação

```
Imc.java x
Source History
1 import javax.swing.JOptionPane;
2 /**
3  * @author marco
4  */
5 public class Imc {
6     public static void main(String args[]){
7         float peso, altura, imc;
8         peso = Float.parseFloat(JOptionPane.showInputDialog(
9             null, "Digite o seu peso em kg:"));
10        altura = Float.parseFloat(JOptionPane.showInputDialog(
11            null, "Digite a sua altura em m: "));
12        imc = peso / (altura*altura);
13        JOptionPane.showMessageDialog(null, "O seu IMC é " + imc + "kg/m²");
14        System.out.println("O seu IMC é " + imc + "kg/m²");
15    }
16 }
```



Exemplo de aplicação

```
Imc.java x
Source History
1 import javax.swing.JOptionPane;
2 /**
3  * @author marco
4  */
5 public class Imc {
6     public static void main(String args[]){
7         float peso, altura, imc;
8         peso = Float.parseFloat(JOptionPane.showInputDialog(
9             null, "Digite o seu peso em kg:"));
10        altura = Float.parseFloat(JOptionPane.showInputDialog(
11            null, "Digite a sua altura em m: "));
12        imc = peso / (altura*altura);
13        JOptionPane.showMessageDialog(null, String.format("O seu IMC é %.2f kg/m²", imc));
14        System.out.printf("O seu IMC é %.2f kg/m²", imc);
15    }
16 }
```

Saída formatada para duas casas decimais

printf("%.2f", imc) para duas casas decimais

String.format("%.2f", imc) para 2 casas decimais



Exemplo de aplicação

4- Crie um programa que solicita ao usuário o modelo de um carro, a quantidade de quilômetros rodados e o seu valor. Após, mostre os dados do carro.

```
carro.alg x
1 algoritmo carro
2     literal modelo
3     inteiro quilometragem
4     real valor
5     inicio
6         escreva("Digite o modelo do veículo: ")
7         leia(modelo)
8         escreva("Digite a quilometragem do veículo: ")
9         leia(quilometragem)
10        escreva("Digite o valor do veículo: ")
11        leia(valor)
12        escreva("Dados do veículo: \n Modelo: " +
13                modelo + "\n Quilometragem: " +
14                quilometragem + "km\n Valor: R$ " + valor))
15    fim
```



Exemplo de aplicação

```
Carro.java
Source History
1 import javax.swing.JOptionPane;
2 /**
3  * @author marco
4  */
5 public class Carro {
6     public static void main(String[] args) {
7         String modelo;
8         int quilometragem;
9         float valor;
10        modelo = JOptionPane.showInputDialog(null, "Digite o modelo do veículo:");
11        quilometragem = Integer.parseInt(JOptionPane.showInputDialog(null,
12            "Digite a quilometragem do veículo: "));
13        valor = Float.parseFloat(JOptionPane.showInputDialog(null, "Digite o valor do veículo R$"));
14        JOptionPane.showMessageDialog(null, "Dados do veículo: \n Modelo: " +
15            modelo + "\n Quilometragem: " +
16            quilometragem + "km\n Valor: R$ " + valor);
17    }
18 }
```



Exemplo de aplicação

5- Crie um algoritmo que calcule a média aritmética de 4 números reais digitados pelo usuário e exiba o resultado.

```
media.alg x
1 algoritmo media
2   real n1, n2, n3, n4, media
3   inicio
4       escreva("Digite o primeiro número: ")
5       leia(n1)
6       escreva("Digite o segundo número: ")
7       leia(n2)
8       escreva("Digite o terceiro número: ")
9       leia(n3)
10      escreva("Digite o quarto número: ")
11      leia(n4)
12      media = (n1+n2+n3+n4) / 4
13      escreva("A média dos números digitados é: " + media)
14  fim
```



Exemplo de aplicação

```
Media.java x
Source History
1 import javax.swing.JOptionPane;
2 /**
3  * @author marco
4  */
5 public class Media {
6     public static void main(String[] args) {
7         float n1, n2, n3, n4, media;
8         n1 = Float.parseFloat(JOptionPane.showInputDialog(null, "Digite o primeiro número: "));
9         n2 = Float.parseFloat(JOptionPane.showInputDialog(null, "Digite o segundo número: "));
10        n3 = Float.parseFloat(JOptionPane.showInputDialog(null, "Digite o terceiro número: "));
11        n4 = Float.parseFloat(JOptionPane.showInputDialog(null, "Digite o quarto número: "));
12        media = (n1 + n2 + n3 + n4)/4;
13        JOptionPane.showMessageDialog(null, "A média dos números digitados é: " + media);
14    }
15 }
```



Alguma dúvida????



Então, agora é pra valer... Exercícios de aplicação



Observações sobre exercícios

- ✔ Todos os exercícios devem ser resolvidos em algoritmo e Java.
- ✔ Os códigos podem ser feito no NetBeans, Eclipse, VSCode ou no Repl.it.
- ✔ Após finalizar todos os exercícios da aula, compacte os arquivos e pastas e envie no Blackboard.



Exercícios

- 1- Crie um programa que solicite ao usuário a sua idade expressa em anos, meses e dias (variáveis separadas). Calcule e mostre a idade expressa apenas em dias. Para isso considere 1 ano = 365 dias, 1 mês = 30 dias.
- 2- Faça um programa que receba três notas, calcule e mostre a média aritmética.
- 3- Escreva um programa para calcular o valor de uma prestação em atraso (prestacao). Para isso, obtenha o valor da prestação (valorPrestacao), a porcentagem de multa pelo atraso (multa) e a quantidade de dias de atraso (qtdeDias). Calcular e mostrar o valor da prestação atualizado, sabendo que:

$$\text{prestacao} = \text{valorPrestacao} + (\text{valorPrestacao} * (\text{multa} / 100) * \text{qtdeDias})$$

Exercícios

4- Faça um programa que receba o salário de um funcionário, calcule e mostre o novo salário, sabendo-se que este sofreu um aumento de 25%.

5- Faça um programa que receba o salário base de um funcionário, calcule e mostre o salário a receber, sabendo-se que o funcionário tem gratificação de 5% sobre o salário base e paga imposto de 7% também sobre o salário base.

6- Faça um programa que calcule e mostre a área de um triângulo a partir da leitura da base e altura (em cm). Sabe-se que:

$$\text{Área} = (\text{base} * \text{altura})/2.$$

Créditos

Esta aula foi elaborada com base no material produzido e cedido gentilmente pelos **Professores Alcides, Lédon, Ana e Cristiane.**





That's all Folks!