

Laboratório de Desenvolvimento de Algoritmos

Conteúdo:

- Tipos e Dados
- Variáveis
- Comandos de saídas modo gráfico e texto
- Comandos de entrada modo gráfico

Tipos de Dados

- Representa o tipo de informação que será manipulada
 - Exemplos:
 - 10 - (inteiro)
 - “casa” - (caracteres)
- Em Java, os tipos são oito: **byte, short, int, long, float, double, char** e **boolean**.

Classificação dos tipos

Caracteres alfanuméricos	char
	String
Números Inteiros	byte
	int
	short
	long
Números Reais	float
	double
Lógico	boolean

Tipos Inteiros

Tipo	Tamanho	Valor Mínimo	Valor Máximo
byte	1 byte	-128	127
short	2 bytes	-32.768	32.767
int	4 bytes	-2.147.483.648	2.147.483.647
long	8 bytes	-9.223.372.036. 854.775.808	9.223.372.036. 854.775.807

- ▶ A coluna **Tamanho** indica a **quantidade de memória** consumida por cada um dos tipos inteiros.

Tipos Reais (ponto flutuante)

Tipo	Tamanho	Valor Mínimo	Valor Máximo	Precisão
float	4 bytes	-3.4028E+38	3.4028E+38	6-7 dígitos
double	8 bytes	-1.7976E+308	1.7976E+308	15 dígitos

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

Tipo Caracter/Alfanumérico

- ▶ Armazenagem de dados alfanuméricos. O tipo char também pode ser usado como um dado inteiro com valores na faixa entre 0 e 65535.

Tipo	Tamanho	Valor Mínimo	Valor Máximo
char	2 bytes	0	65535
String	Representado como objeto		

- ▶ O tipo caracter (char) é sempre representado por aspas simples ('h'). Caracteres representados por aspas duplas ("h") na verdade são Strings.

Tipo Lógico

- ▶ Em Java temos o tipo lógico **boolean**, que é o tipo de dado mais simples. Pode assumir apenas um entre dois valores: **true** ou **false**.

Tipo	Tamanho	Valor
boolean	1 bit	true ou false

Variáveis

- São espaços ou alocações nas quais os dados são armazenados.
- A declaração de uma variável instrui o programa a reservar um espaço na memória para que seja possível armazenar um dado de determinado tipo.
- Quantidade de memória reservada para uma variável é definida pelo seu tipo.

Declaração de Variáveis

- Pode ser feita em qualquer parte do corpo de uma classe.
- Seria interessante definir as variáveis sempre no mesmo ponto do código, como no início do programa.
- É composta por:
 - Um tipo
 - Um identificador (nome)
 - Um valor
 - Ponto-e-vírgula

Exemplos de Declarações de Variáveis

//Tipos inteiros

byte idade;

short num=10;

int cod;

long qtde;

//Tipos reais

float preco;

double receber;

//Tipo caracter/String

char letra;

String nome;

//Tipo lógico

boolean escolha;

Declaração de Variáveis

```
public class Variavel {  
    public static void main (String args[ ]) {  
        //<tipo da variável> <nome da variável>  
        int idade;  
        float valor, media;  
    }  
}
```

Identificador (nome)

- É através dele que você irá se referir a variável no código para lhe atribuir algum dado ou para recuperar um dado que foi armazenado.
- Regras para os nomes das variáveis:
 - Podem começar com letra minúscula ou maiúscula, underscore (_), ou símbolo de dólar (\$);
 - Não podem conter pontuação nem espaços;
 - Não podem ser utilizadas palavras reservadas da linguagem.

Palavras reservadas

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

Case-sensitive

- Como o Java é case-sensitive, a variável
 $Ar \neq ar$

Exemplo de nomes válidos:	Exemplos de nomes inválidos:
codigo	1teste
quantidade	nome cliente
\$nome	cli.nome
_nro	x - y
indice1	código

Atribuição de valores a variáveis

- Existem várias maneiras de atribuir valores a variáveis:
 - Informando logo da declaração da variável, o valor que ela irá receber:

```
int cod = 123;  
float preco = 10.4f;  
char letra = 'A';  
String nome="Lukas"
```

Identifica um
dado do tipo
float

- Definir que uma variável assumo o valor de uma outra variável:

```
int cod1,cod2;  
cod1=54;  
cod2=cod1;
```

Atribuição de Valores às Variáveis

```
public class Variaveis {  
    public static void main (String args[ ]) {
```

```
        int idade;
```

```
        float valor, media;
```

```
        //inicialização das variáveis
```

```
        idade = 20;
```

```
        valor = 15.80f;
```

```
        media = 55.4f;
```

```
    }
```

```
}
```


Atribuição de valores a variáveis

- Atribuir o resultado de uma expressão a uma variável:

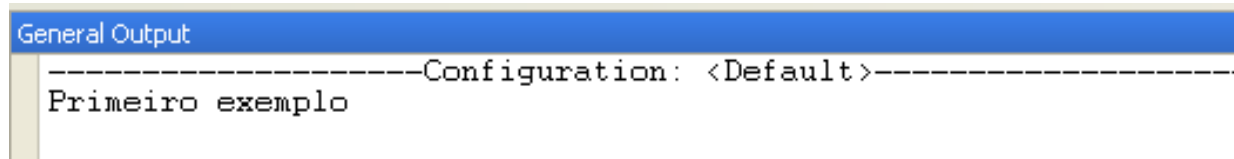
```
float preco,qtde_comprada;  
double receber;  
receber = preco*qtde_comprada;
```

- Entre outras possibilidades.

Comandos de saída de dados

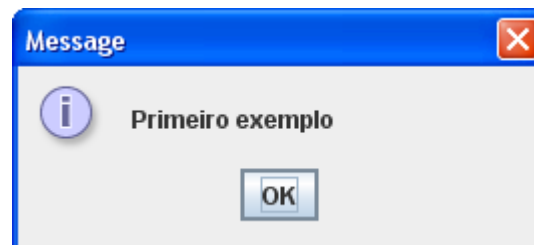
– Modo texto:

```
System.out.println("Primeiro exemplo");
```



– Modo Gráfico:

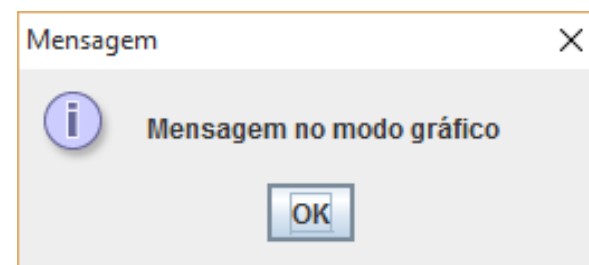
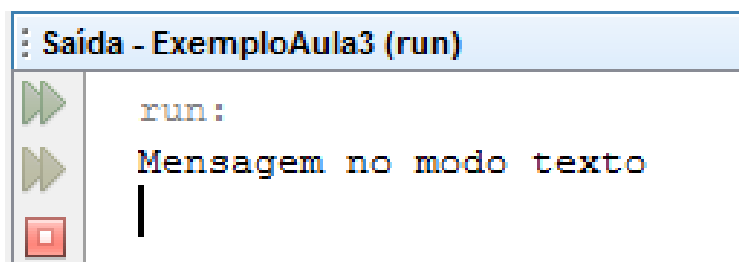
```
JOptionPane.showMessageDialog(null,"Primeiro exemplo");
```



Exemplo

```
import javax.swing.JOptionPane;

public class Exemplo1 {
    public static void main(String[] args) {
        System.out.println("Mensagem no modo texto");
        JOptionPane.showMessageDialog(null, "Mensagem no modo gráfico");
    }
}
```



Comandos de saída – Modo Gráfico

- Para utilizarmos o comando de saída gráfico, é necessário importar o pacote que fornece essa funcionalidade no Java. Para importar o pacote para nosso código, utilizamos:

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

- Carrega a classe **JOptionPane** do pacote **javax.swing**. Ajuda a definir interfaces gráficas com o usuário para seu aplicativo.

Comandos de saída – Modo Gráfico

- ▶ Um pacote (package) em Java é um diretório no qual ficam armazenadas uma ou mais classes predefinidas que estão agrupadas em categorias de classes relacionadas.
- ▶ A inclusão dessas classes nos programas é feita pela linha abaixo (chamada diretiva):

```
import  nome-do-pacote.nome-da-classe;
```

Exemplos:

```
import java.io.*; //todas as classes do pacote io
import java.util.Date; //A classe Date do pacote util
import java.awt.*; //todas as classes do pacote awt
```

Mais pacotes, pág. 271 no livro do Deitel

Detalhes do exemplo

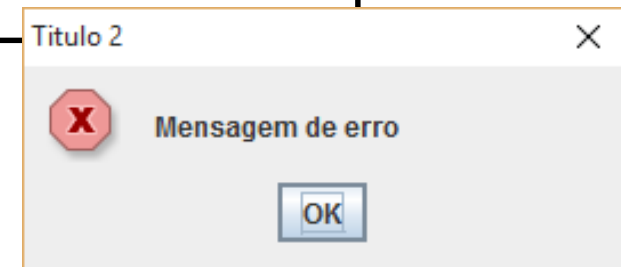
```
JOptionPane.showMessageDialog(null,  
    "Mensagem no modo gráfico");
```

- ▶ Indica uma chamada para o método ***showMessageDialog*** da classe **JOptionPane**, exige dois argumentos, separados por vírgula.
- ▶ Primeiro argumento: **null** → ajuda o Java a posicionar a caixa de diálogo (caixa de diálogo aparece no centro da tela).
- ▶ Segundo argumento: **A mensagem que será exibida**

Alterando as propriedades da janela

- A função *showMessageDialog* pode ser utilizada com quatro argumentos (na ordem), sendo eles:
 - a janela,
 - a mensagem,
 - o título da janela e
 - o tipo de mensagem.

```
//Exibe uma janela com ícone do erro, com mensagem e título  
JOptionPane.showMessageDialog (null,  
    "Mensagem de erro", "Titulo 2 ",  
    JOptionPane.ERROR_MESSAGE);
```



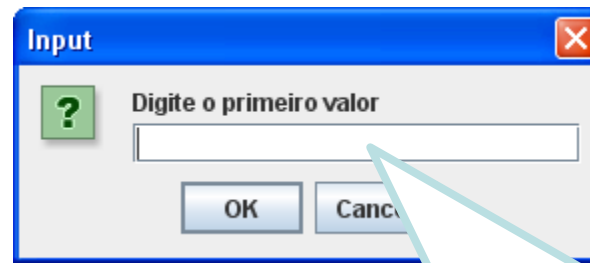
Caracteres de escape

- A linguagem java permite usar uma sequencia de caracteres de controle ao qual chamamos de caracteres especiais

Seq.de Escape	Descrição
\n	Nova linha. Posiciona o cursor de tela no início da próxima linha
\t	Tabulação horizontal. Move o cursor de tela para a próxima parada de tabulação
\\	Barra invertida. Utilizada para imprimir um caracter barra invertida
\"	Aspas duplas. Utilizada para imprimir um caracter aspas duplas <code>System.out.println("\nexemplo\" ");</code> Exibe => "exemplo"
\'	Aspas simples. Utilizada para imprimir o caracter aspas simples

Entrada de Dados no Modo Gráfico

```
valor = JOptionPane.showInputDialog("Digite o primeiro valor");
```

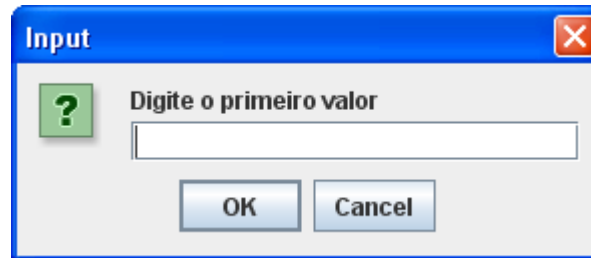


A variável
que recebe
o conteúdo
é uma
String

Mesmo que o usuário
digite um valor
numérico, o conteúdo
será sempre uma
String

Também precisamos *importar* o pacote...

Entrada de Dados no Modo Gráfico



Para fazer cálculos com os valores da caixa de entrada de dados, precisamos converter o valor para um tipo numérico do Java, por exemplo:

```
int n1;  
n1 = Integer.parseInt(valor);
```

Conversões de tipos

- ▶ Para o tipo int:

`Integer.parseInt (variável)`

- ▶ Para o tipo double:

`Double.parseDouble(variável)`

- ▶ Para o tipo float:

`Float.parseFloat(variável)`

- ▶ Para o tipo long:

`Integer.parseInt(variável)`

- ▶ Para o tipo char:

`entrada.charAt(0);`

Do algoritmo para codificação

1) Utiliza bibliotecas específicas?

- ✓ Import;

2) Descrição de classes (uma classe por arquivo .java)

- ✓ Atributos
 - Tipo de dados (variáveis)
- ✓ Construtor(es)
- ✓ Métodos
 - Tipo de retorno
 - Parâmetros

3) Programa principal

- ✓ Repetir etapa 1
- ✓ Declarar variáveis

Exercícios

1. Escreva um programa em Java que exiba exatamente a mensagem abaixo, para isso utilize apenas um comando de saída (modo gráfico ou texto)

1. Introdução
2. Comandos de saída
 1. “System.out.println”
 2. “JOptionPane.showMessageDialog”
 1. Diferentes configurações

Prestar atenção, eu quero que o programa exiba exatamente a mensagem acima (inclusive as aspas e as tabulações) em um único comando de saída

Exercícios

2. Faça um programa em Java que solicite ao usuário dois números, calcule e mostre na tela o resto da divisão (operador %) do primeiro pelo segundo.

Exercícios

3. Faça um programa em Java para calcular o valor de uma prestação atualizada. Para isso, obtenha o valor da prestação, a multa pelo atraso e a quantidade de dias de atraso. Calcular e mostrar o valor da prestação atualizado, sabendo que:

$$\text{prestacaoAtual} = \text{valorPrestacao} + (\text{valorPrestacao} * (\text{multa} * 0,01) * \text{qtdeDias})$$

Exercícios

4. Faça um programa em Java que solicite o valor do raio de uma esfera e calcule e mostre o seu comprimento, sabendo que:

$$\text{comprimento} = 2 \cdot 3,14 \cdot \text{raio}$$