

Lógica de Programação

Prof. Dr. Bruno Queiroz Pinto

Resumo da Aula

- Variáveis
- Comandos de saída
- Expressões aritméticas
- Comentários

Programas em java

No Replit o nome será **Main**

```
public class NomePrograma {  
    public static void main(String args[]) {  
        //Aqui escrevemos o código de nosso programa  
    }  
}
```

- Nome da classe : Utilizar o padrão "Camel Case"
 - No momento sempre utilizaremos o nome Main
 - Cada Palavra do nome tem a primeira letra maiúscula
- `public static void main(String args[])`
 - Função que permite a execução de códigos na linguagem java.
 - **'Todo'** o seu código executável estará aqui
 - O código deve estar dentro de um abre chaves **{** e um fecha chaves **}**.

Comandos de saída

- Permite a saída de dados
- Quando o programa exibe alguma informação para o usuário.
 - "O programa está apresentando dados."

Dispositivo de SAÍDA



- Em Java:

```
System.out.println("Bom dia");
```

```
System.out.println(2022);
```

```
System.out.println("Estamos no ano de " + 2022);
```

Permite concatenar
saídas independentes

Comandos de saída

- Imprimir na mesma linha

```
System.out.print( "Palavra1" );
```

```
System.out.print( "Palavra2" );
```

```
System.out.print( "Palavra3" );
```

Comandos de saída

- Imprimir uma quantidade definida de casas decimais

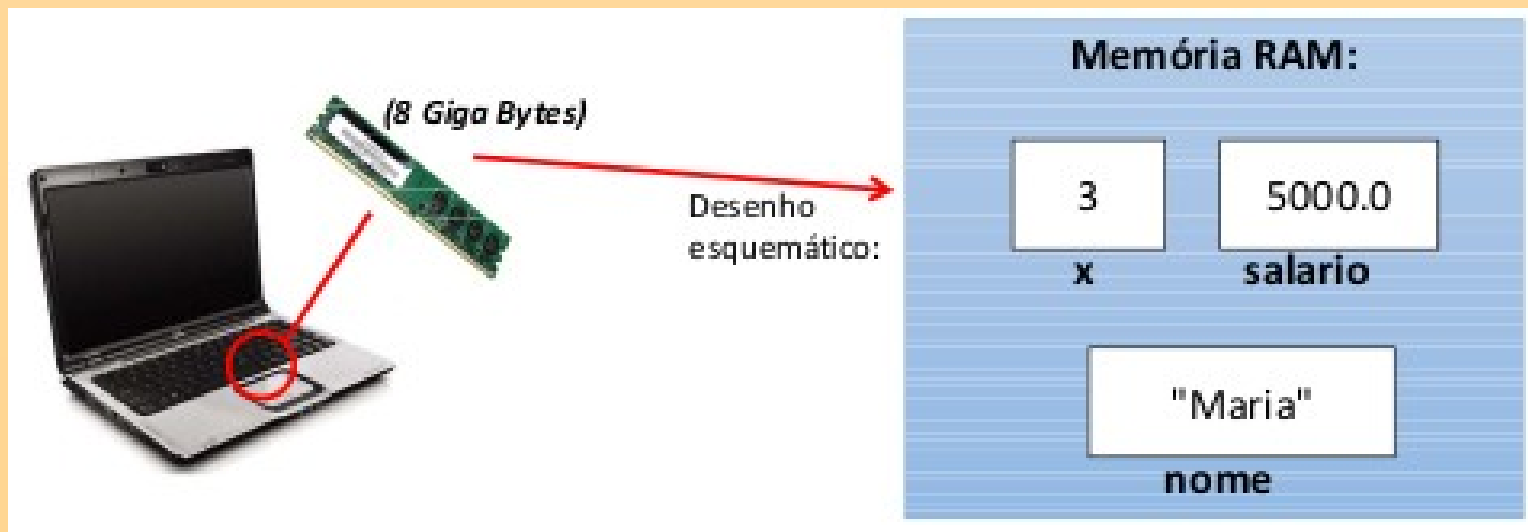
```
System.out.printf("%.2f\n", 10.67654);  
// imprime 10.68 (arredonda o valor)
```

Iremos utilizá-lo bastante com números reais, adiante reforçamos esse conteúdo

\n → comando para quebrar linha – similar a um <ENTER>

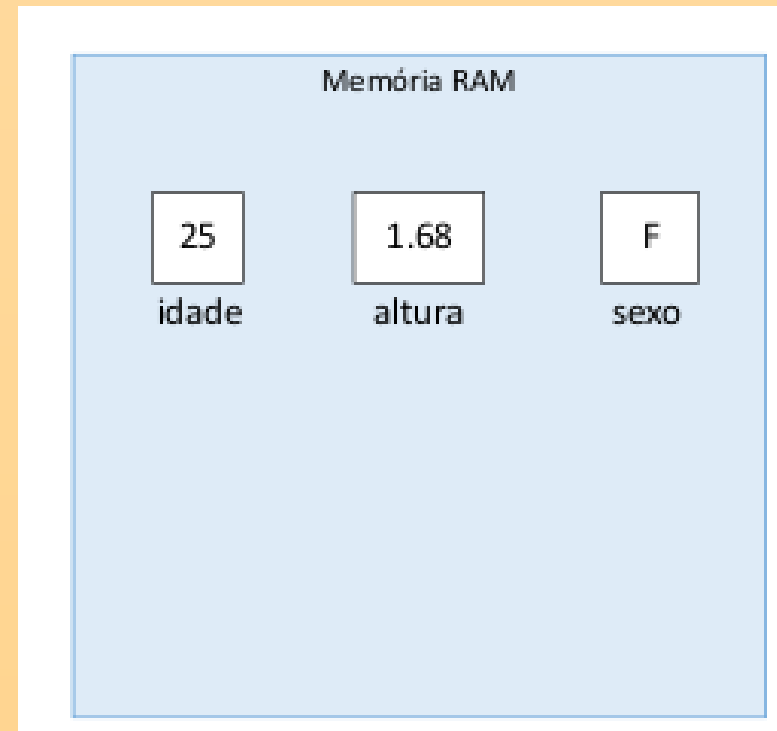
Variáveis : definição informal

- Em programação, uma variável é uma porção de memória (RAM) utilizada para armazenar dados durante a execução dos programas.



Como definir nomes de variáveis

- `<tipo> <nome> = <valor inicial>;`
 - Valor inicial é opcional
- Exemplos:
 - `int idade = 25;`
 - `double altura = 1.68;`
 - `char sexo = 'F';`
 - **`String nome = "Ifbm";`**
 - **`boolean eCorreto = true;`**



Como definir nomes de variáveis

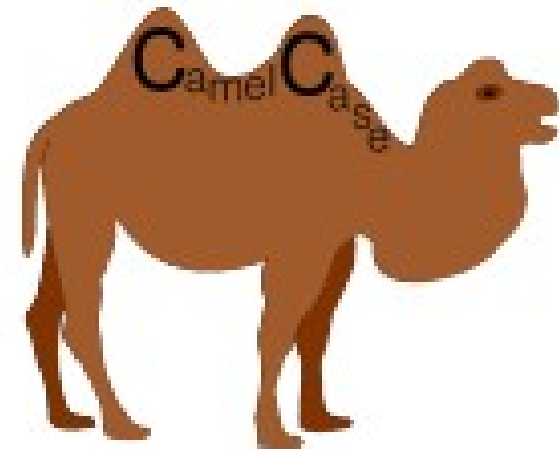
- Não pode começar com dígito: use uma letra ou _
- Não pode ter espaço em branco
- Não usar acentos ou til
- Sugestão: use o padrão "camel case"

Errado:

```
int 5minutos;  
int salário;  
int salário do funcionario;
```

Correto:

```
int _5minutos;  
int salario;  
int salarioDoFuncionario;
```



Principais tipos de dados

- Int → permite armazenar um número inteiro
- double ou float → permite armazenar um número real (número com ponto flutuante)
- **String** → permite armazenar um texto
- char → permite armazenar apenas um caracter

Números reais:

As linguagens de programação foram criadas no idioma inglês, então utilizamos o '.' no lugar da vírgula

Como imprimir uma variável

```
int ano = 2022;  
System.out.println(ano); // imprime apenas "2022"  
System.out.println("Estamos no ano de " + ano); // imprime "Estamos no ano de 2022"
```

Soma (apenas números)

```
int parte1 = 20;  
int parte2 = 22;  
System.out.println(parte1 + parte2); // imprime 42  
System.out.println("Ano " + parte1 + parte2); // imprime "Ano 2022"
```

Concatena (texto e números)

??????

```
System.out.println("Ano " + (parte1 + parte2));
```

Entrada de dados

- Precisamos utilizar uma ferramenta/biblioteca do java chamada Scanner.
 - Necessário importar a ferramenta para dentro de nosso projeto
 - Necessário criar uma variável que representa o leitor do teclado(Scanner).

Dispositivo de ENTRADA



```
import java.util.Scanner;

public class Programa {
    public static void main(String args[]) {
        Scanner s = new Scanner(System.in);
        System.out.println("Digite dois números : ");
        int numero1 = s.nextInt();
        int numero2 = s.nextInt();
        System.out.println("primeiro número = " + numero1);
        System.out.println("segundo número = " + numero2);
    }
}
```

Entrada de dados

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

Indicar que irá utilizar essa ferramenta/biblioteca

```
Scanner s = new Scanner(System.in);
```

Cria um leitor(Scanner)
para a entrada padrão do sistema operacional(System.in)
Salva numa variável **s** (**s** é o nosso leitor de teclado)

```
int numero1 = s.nextInt();  
int numero2 = s.nextInt();
```

Utiliza uma função para pegar o próximo número inteiro digitado
Salva em uma variável do tipo de dados inteiro

```
String texto1 = s.nextLine();  
String texto2 = s.next();
```

O Java têm duas funções principais para fazer a leitura de textos.
nextLine() → permite capturar um texto com diversas palavras.
next() → permite capturar apenas a próxima palavra digitada

```
double numero3 = s.nextDouble();  
float numero4 = s.nextFloat();
```

Temos duas funções para capturar números reais:
- nextDouble() → funções que permitem capturar números reais

double permite armazenar números maiores.

Entrada de dados

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

Indicar que irá utilizar essa ferramenta/biblioteca

```
Scanner s = new Scanner(System.in);
```

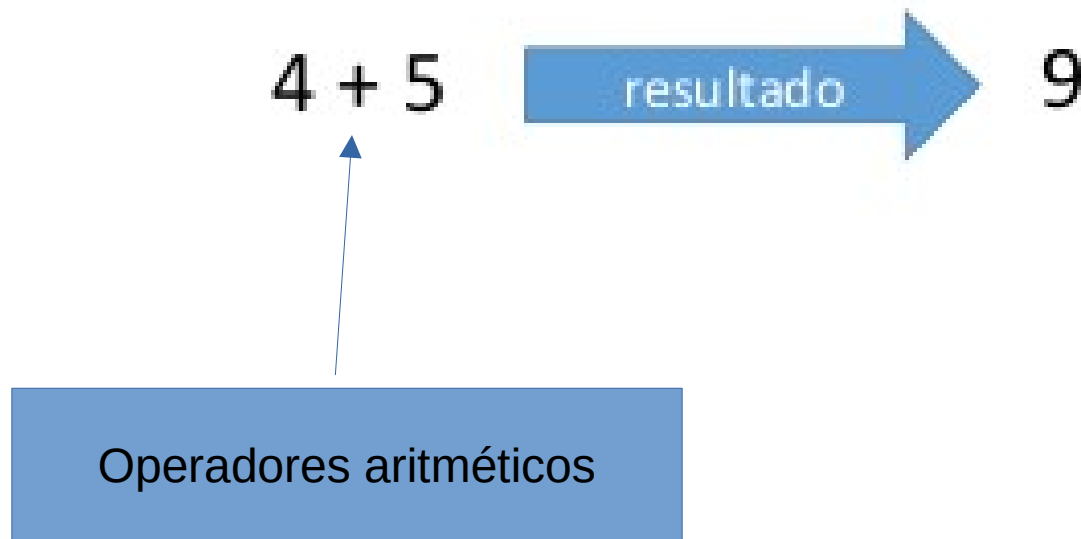
Cria um leitor(Scanner)
para a entrada padrão do sistema operacional(System.in)
Salva numa variável **s** (**s** é o nosso leitor de teclado)

```
char letra = s.next().charAt(0);
```

O Java não apresenta uma função própria para ler um caractere.
Entretanto podemos utilizar uma função que manipula texto e
Então pegar o primeiro caractere da palavra.
Ou seja, precisa pegar um caractere, utilize:
- s.next().charAt(0).

No momento iremos utilizar pouco

Expressões aritméticas



Expressões aritméticas

C, C++,
Java, C# →

Operador	Significado
+	adição
-	subtração
*	multiplicação
/	divisão
%	resto da divisão ("mod")

Precedência: 1º lugar: * / %
2º lugar: + -

Considera a ordem de ocorrência
Entre * / %
Entre + -

Exemplos de expressões aritméticas

- Exemplos de expressões aritméticas

- $2 * 6 / 3$ Resultado = 4
- $3 + 2 * 4$ Resultado = 11
- $(3 + 2) * 4$ Resultado = 20
- $60 / (3 + 2) * 4$ Resultado = 48
- $60 / ((3 + 2) * 4)$ Resultado = 3

Programar

Sem variáveis

```
System.out.println(2 * 6 / 3);
```

Com variáveis

```
int a=2;  
int b=6;  
int c=3;  
System.out.println(a * b / c);
```

Exemplos com o operador "mod"

- Exemplos do uso do mod(resto)

- $14 \% 3$ Resultado = 2

- $19 \% 5$ Resultado = 4



Programar

Sem variáveis

```
System.out.println(14 % 3 );
```

Com variáveis

```
int a=14;  
int b=3;  
System.out.println(a % b);
```

Exemplos em números reais

- A divisão de dois números inteiros gera um inteiro.

- Para produzir um número real, você precisa inserir a multiplicação por 1.0;

- `double x = 1.0*10/3;`

- Ou utilizar um recurso da linguagem Java, conhecido como Cast.

- `double x = (double) 10/3;`

Nesse último caso, o computador força a gerar um número real.

- Imprimindo números reais. Necessário definir a quantidade de casas decimais após o '.'

- Utilizando o `System.out.println(x);` // imprime 16 casas decimais após o .

- imprime : 3.3333333333333335

- Utilizando o `System.out.printf("%.2f \n", x);` //você decide a quantidade de casas decimais.

- imprime : 3.33

\n é o comando para quebrar linha, opcional.

.2f indica que nesse local será impresso um número real com duas casas decimais