



# Java II


Bem-vindo ao estudo sobre Introdução à Linguagem Java. Este estudo ajudará na compreensão de alguns conceitos e práticas que são importantes no contexto de programação. Vamos conhecer melhor esses conceitos?

## Introdução a Linguagens

Neste módulo serão apresentados alguns tipos de linguagens computacionais.

## Linguagem de máquina

O computador necessita de ações específicas para fazer o processamento dos dados. Essas ações devem ser entendidas pelo computador através de uma linguagem de máquina. Esta linguagem é composta somente por números, representados de forma binária, que equivalem às operações e operandos usados no processamento do programa. A linguagem de máquina é complexa para o entendimento humano, então para auxiliar o desenvolvimento de um programa foi proposto outro tipo de linguagem representada por comandos, que reproduz tarefas executadas pelo computador, a linguagem de montagem (*assembly*). Entretanto, a linguagem de montagem continuou sendo um desafio para o desenvolvimento de softwares pela sua complexidade menor que a linguagem de máquina, mas ainda assim complexa. Além

disso, programas desenvolvidos nesta linguagem são específicos. isto é, alguns programas funcionam para determinados processadores , porém não funcionam em outros processadores diferentes (RIBEIRO, 2019).

## Linguagem de montagem

Uma Linguagem de Montagem pura é a linguagem onde cada comando traduz exatamente uma instrução de máquina. *Assembly* ou linguagem de montagem é uma notação mais legível do que a de linguagem de máquina. O código de máquina torna-se legível pela substituição dos valores brutos por símbolos chamados mnemônicos, mas ainda assim mais complexo que outras linguagens (RIBEIRO, 2019). Seu objetivo é gerar códigos que se comunicam com os dispositivos computacionais, como microprocessadores e microcontroladores.

## Linguagem de Programação

A linguagem de programação é formal e permite que um programador escreva um conjunto de instruções para criação de programas que manipulam o comportamento físico e lógico de uma máquina. É a comunicação entre o hardware (máquina) e o programador. Seu funcionamento é feito por através de uma série de instruções, palavras-chaves, regras semânticas e sintáticas e símbolos (RIBEIRO, 2019).

Abaixo, exemplos de linguagens de programação:

- Java
- JavaScript



- Python
- Linguagem C
- Linguagem C++
- Linguagem C#
- PHP



## JAVA

*“O Java é uma tecnologia usada para desenvolver aplicações que tornam a Web mais divertida e útil. O Java não é a mesma coisa que o javascript, que é uma tecnologia simples usada para criar páginas Web e só é executado no seu browser”* (JAVA, 2021). Em 2020, a linguagem Java completou 25 anos de existência. É uma linguagem considerada moderna e orientada a objetos, que se adaptou à internet e aos dispositivos móveis. Além disso, tem tudo que uma linguagem de programação permite, incluindo toda a estrutura estudada em algoritmos nesta disciplina.

## Linguagem de marcação

A linguagem de marcação é uma linguagem que anota o texto para que o computador possa manipulá-lo. É de fácil entendimento porque as anotações são escritas de forma a distingui-las do próprio texto (marcações), com o objetivo de estruturar o texto e dar significado (semântica) para que um navegador possa exibi-lo (COMER, 2016).



## Algumas características de uma linguagem de marcação:



- Utilizadas para definir formatos, maneiras de exibição e padrões dentro de um documento. Exemplo: HTML
- Não possuem qualquer estrutura de controle como as linguagens de programação tradicionais (por exemplo, comandos condicionais ou de repetição).

## EXEMPLO

Para exemplificar uma introdução à linguagem Java, vamos desenvolver um programa Java que declara variáveis, recebe informações pela linha de comando, calcula a adição de dois números e mostra essas informações pela linha de comando.

```
1 //salvar como Programa01.java
2
3 class Programa01
4 {
5     public static void main (String entrada[])
6     {
7         //declaração de variáveis
8         int NumInt;
9         double NumReal, soma;
10        char Caracter;
11
12        //entrada de dados
13        NumInt = Integer.parseInt(entrada[0]);
14        NumReal = Double.parseDouble(entrada[1]);
15        Caracter = (entrada[2]).charAt(0);
16        //processamento
17        soma = (double)NumInt + NumReal;
18        //saida de resultados
19        System.out.println((double)NumInt + " + " +
20        NumReal + " = " + soma + " sinal " + Caracter);
21
22        System.exit(0);
23    }
24 }
```



Perceba que utilizamos como entrada de dados o vetor `entrada` nas posições 0, 1 e 2. Nas linhas 13, 14 e 15 do programa.



Convertemos a `entrada[0]` para inteiro com o método `Integer.parseInt`, a `entrada[1]` para real com o método `Double.parseDouble` e a `entrada[2]` para char com o método `CharAt(0)`.

Como a variável `soma` e o valor de `NumReal` são reais, antes de realizar a adição, convertemos o conteúdo inteiro da variável `NumInt` para double na linha 17 do programa.

Mostramos o resultado do cálculo, passando a mensagem por parâmetro no método `System.out.println` que mostra a mensagem para o usuário pelo Prompt de Comando.

```
//salvar como Programa01.java
```

```
class Programa01  
  
{  
  
    public static void main (String entrada[])  
  
    {
```



```
//declaração de variáveis
```



```
int NumInt;
```

```
double NumReal, soma;
```

```
char Caracter;
```

```
// entrada de dados
```

```
NumInt = Integer.parseInt(entrada[0]);
```

```
NumReal = Double.parseDouble(entrada[1]);
```

```
Caracter = (entrada[2]).charAt(0);
```

```
// processamento
```

```
soma = (double)NumInt + NumReal;
```

```
// saída de resultados
```

```
System.out.println((double)NumInt + " + " + NumReal + " = " + soma  
+ " sinal " + Caracter);
```

```
System.exit(0);
```

```
}
```

```
}
```



## EXEMPLO COM OPERAÇÕES MATEMÁTICA



Vamos desenvolver um programa Java que declara variáveis, recebe dois números pela linha de comando, calcula o resto da divisão de dois números, o quociente da divisão de dois números, a raiz quadrada do primeiro número e a potência do primeiro número pelo segundo número e mostra essas informações pelo Prompt de Comando.

```
1 //salvar como Programa02.java
2 class Programa02
3 {
4     public static void main (String entrada[])
5     {
6         int n1, n2;
7         int mod, div;
8         double raiz, pot;
9         String msg = "";
10
11         //entrada de dados
12         n1 = Integer.parseInt(entrada[0]);
13         n2 = Integer.parseInt(entrada[1]);
14         //processamento
15         mod = n1 % n2;
16         div = (int)n1 / (int)n2;
17         raiz = Math.sqrt(n1);
18         pot = Math.pow(n1, n2);
19         //saida de resultados
20         msg = "n1 = " + n1 + "    n2 = " + n2 + "\n";
21         msg = msg + "resto da divisao de n1 por n2 = " + mod + "\n";
22         msg = msg + "quociente da divisao de n1 por n2 = " + div + "\n";
23         msg = msg + "raiz quadrada de n1 = " + raiz + "\n";
24         msg = msg + "potencia de n1 e n2 = " + pot + "\n";
25         System.out.println(msg);
26         System.exit(0);
27     }
28 }
```

Neste programa, você pode perceber que, na linguagem Java, o resto da divisão de dois números é utilizado pelo operador %, o quociente da divisão de dois números inteiros é utilizado pelo operador / , a raiz quadrada, utilizamos o método Math.sqrt e a potência, utilizamos o método Math.pow. Veja as linhas 16 a 18 no programa Java.

Perceba que no Java, podemos utilizar o operador / para realizar o cálculo de divisão de dois números reais para ter o resultado da divisão, ou pa

realizar o cálculo de determinar o quociente (sem o resto) da divisão de dois números inteiros.



Veja que na linha 17, utilizamos (int) antes das variáveis, para garantir que elas são variáveis inteiras.

Neste programa, utilizou-se uma variável msg para concatenar as mensagens para serem exibidas para o usuário, observe as linhas 21 a 25 do programa. Esse é um recurso utilizada pela autora, apenas para que o código fique mais limpa e mais fácil de ser entendido.

//salvar como Programa02.java

```
class Programa02  
  
{  
  
    public static void main (String entrada[])  
  
    {  
  
        int n1, n2;  
  
        int mod, div;  
  
        double raiz, pot;
```





```
String msg = "";
```



```
// entrada de dados
```

```
n1 = Integer.parseInt(entrada[0]);
```

```
n2 = Integer.parseInt(entrada[1]);
```

```
// processamento
```

```
mod = n1 % n2;
```

```
div = (int)n1 / (int)n2;
```

```
raiz = Math.sqrt(n1);
```

```
pot = Math.pow(n1, n2);
```

```
// saída de resultados
```

```
msg = msg + "n1 = " + n1 + " e n2 = " + n2 + "\n";
```

```
msg = msg + "resto da divisao de n1 por n2 = " + mod + "\n";
```

```
msg = msg + "quociente da divisao de n1 por n2 = " + div + "\n";
```

```
msg = msg + "raiz quadrada de n1 = " + raiz + "\n";
```

```
msg = msg + "potencia de n1 por n2 = " + pot + "\n";
```

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



System.exit(0);



}

}


## EXEMPLO COM A BIBLIOTECA SWING

Vamos desenvolver um programa Java que declara variáveis, recebe dois números inteiros usando interface com usuário, calcula o resto da divisão dos dois números, a raiz quadrado do primeiro e do segundo números e mostra essas informações.

```
1 //salvar como Programa03.java
2 import javax.swing.*;
3
4 class Programa03
5 {
6     public static void main (String entrada[])
7     {
8         //declaração de variáveis
9         int n1, n2, mod;
10        double raiz1, raiz2;
11        String msg="";
12        //entrada de dados
13        n1 = Integer.parseInt(JOptionPane.showInputDialog("Digite um numero inteiro"));
14        n2 = Integer.parseInt(JOptionPane.showInputDialog("Digite outro numero inteiro"));
15        //processamento
16        mod = n1 % n2;
17        raiz1 = Math.sqrt(n1);
18        raiz2 = Math.sqrt(n2);
19
20        //saida de resultados
21        msg = msg + "resto da divisao de " + n1 + " por " + n2 + " = " + mod + "\n";
22        msg = msg + "raiz quadrada de " + n1 + " = " + raiz1 + "\n";
23        msg = msg + "raiz quadrada de " + n2 + " = " + raiz2 + "\n";
24        JOptionPane.showMessageDialog(null, msg);
25
26        System.exit(0);
27    }
28 }
```

Perceba que, neste exemplo, importamos a biblioteca `javax.swing.*` com o comando `import` na linha 2 do programa.



Perceba que realizamos a entrada de dados para este programam Java, usando o método `JOptionPane.showInputDialog` que está de , na biblioteca `javax.swing.*`. E a saída de dados foi realizada com o método `JOptionPane.showMessageDialog`.


É uma forma de se realizar a interação humano-computador com uma interface mais amigável.

### Atividade extra

Assista ao filme "Particle Fever" O longa mostra as expectativas dos cientistas em torno do lançamento do LHC (large Hadron Collider), o maior acelerador de partículas do mundo. Por que assistir? A história mostra os bastidores da busca pelo famoso Bóson de Higgs. Também conhecido como "a partícula de Deus", ele é a peça que faltava para uma complexa teoria que explica grande parte dos fenômenos naturais. A dica é de Fábio de Vicente, professor de Física da Unesp de Rio Claro.

### Referência Bibliográfica



- COMER, D. E. **Redes de computadores e internet**. 6. ed. Porto Alegre: Bookman, 2016 
- **JAVA**. Disponível em: [https://www.java.com/pt-BR/about/whatis\\_java.jsp](https://www.java.com/pt-BR/about/whatis_java.jsp) Último acesso em: Julho de 2021
- PUGA, S.; RISSETTI, G. **Lógica de programação e estruturas de dados, com aplicações em Java**. Pearson: 2016.
- RIBEIRO, J. A. **Introdução à programação e aos algoritmos**. 1. ed. Rio de Janeiro: LTC, 2019

**Ir para questão**

