



Programação Orientada a Objetos Utilizando Java

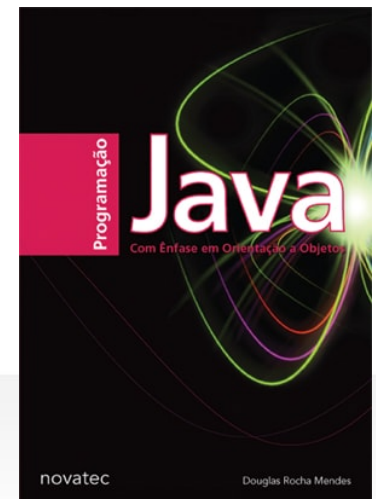
Professor MSc. Odair Jacinto da Silva
odair.silva@unimetrocamp.edu.br

Consulte sempre o calendário acadêmico

- <https://www.wyden.com.br/unimetrocamp/calendario-academico-20192>
- Datas das avaliações AP1, AP2 e AP3
- Datas para solicitação de substitutivas
- Data da substitutiva
- Consulte também o cronograma no Integrees

Livro recomendado

- Existem diversos livros sobre Java e Orientação a Objetos na biblioteca.
- Referência principal
 - Programação Java-com ênfase Em Orientação a Objetos
 - Douglas Rocha Mendes
 - Editora Novatec
 - 2009



Avaliações

Durante o semestre serão aplicadas três avaliações: AP1, AP2 e AP3.

Sendo seus pesos: 30%, 30% e 40%, respectivamente.

Se $AP1*0,3+AP2*0,3+AP3*0,4 \geq 5,0$ e frequência em aula $\geq 75\%$ então o aluno estará aprovado.

A prova substitutiva poderá ser solicitada quando o aluno perder uma das avaliações.

Deverá ser solicitada de acordo com o calendário acadêmico.

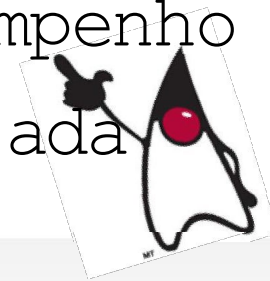
Java

- Linguagem de Programação
 - Criada pela Sun Microsystems
 - Especificada pelo JCP (Java Community Process)
- Plataforma
 - Ferramentas
 - APIs (Application Programming Interface)
 - Ambiente de execução – JRE (Java Runtime Environment)



Java

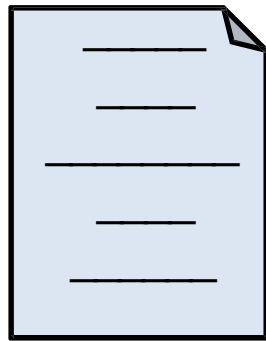
- Características:
 - Simples
 - Robusta
 - Orientada a Objetos
 - Segura
 - Portável
 - Multi-Plataforma
 - Segura
 - Multithreaded
 - Compatibilidade com Redes
 - Arquitetura neutra
 - Dinâmica
 - Alto desempenho
 - Interpretada



Java

- Ambiente de execução
 - JRE (Java Runtime Environment)
- Kit de desenvolvimento
 - JDK (Java Development Kit)
 - JRE
 - Compilador (javac)
 - Depurador (jdb)
 - Empacotador (jar)
 - Outros

Programa fonte



OlaMundo.java

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

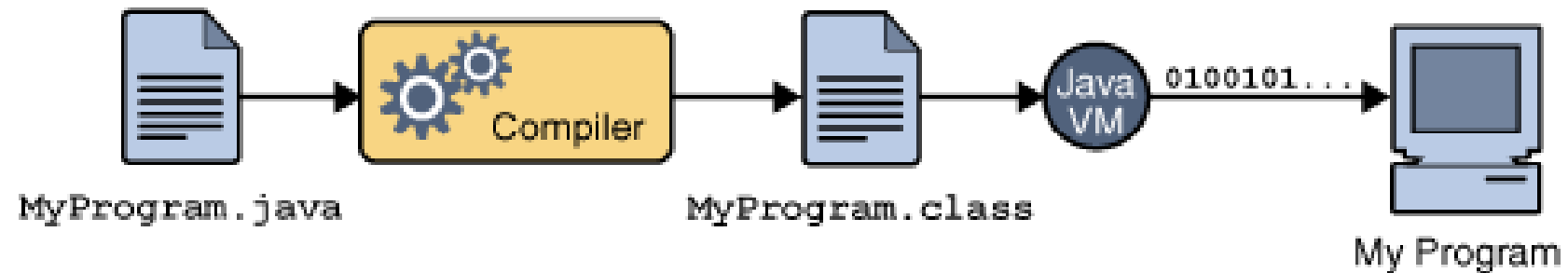
O nome da classe deve ser o mesmo nome do arquivo!

Compilação

- Converte código fonte em bytecodes.
 - Extensão do Arquivo: `.class`
- Compilador Java:
 - Javac

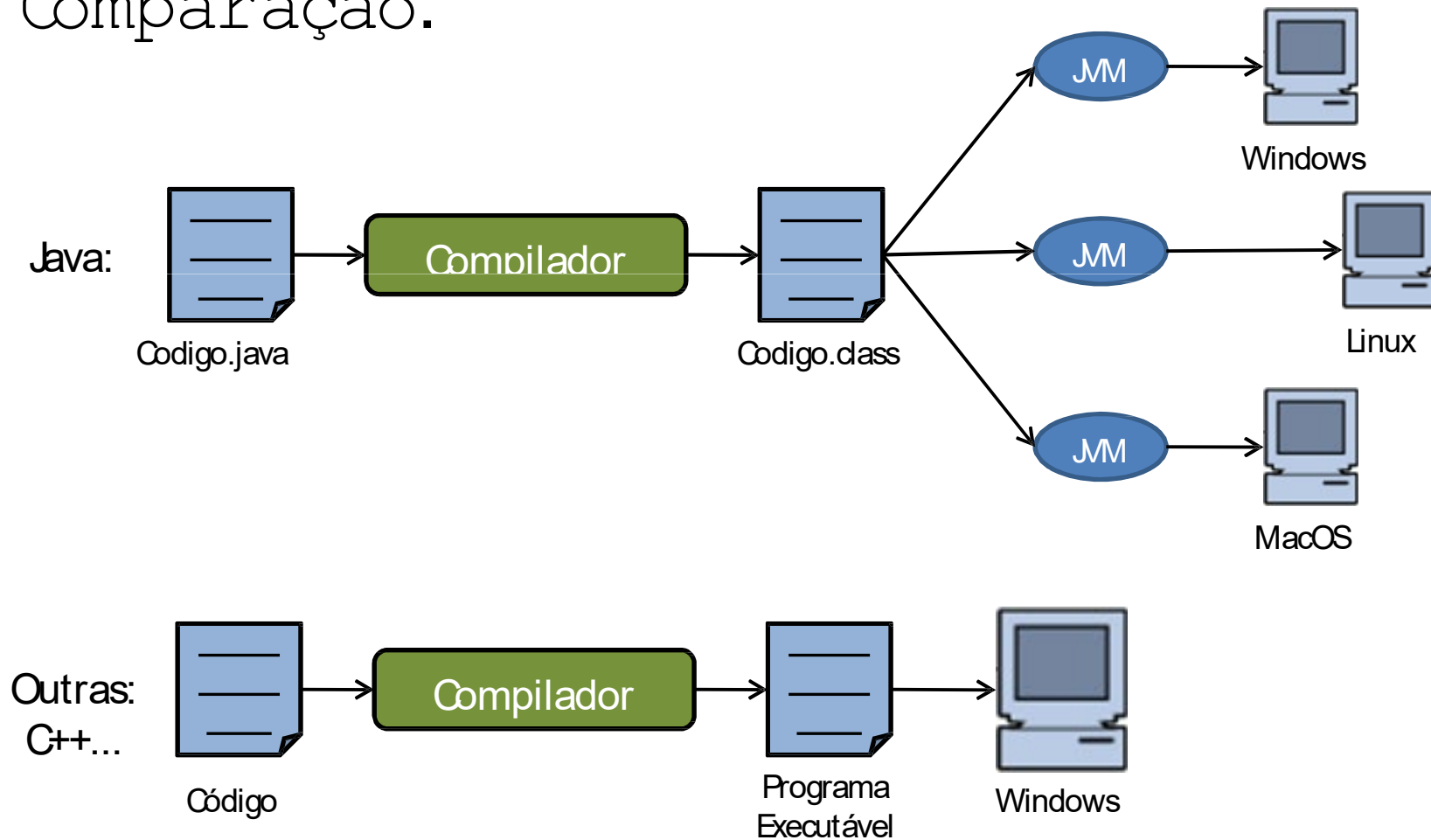
Compilação

- Processo de Compilação:



Compilação

- Comparação:



Execução

- Máquina Virtual Java (Java Virtual Machine – JVM)
- Máquina virtual que interpreta e executa código Java compilado
- Possibilita que a linguagem seja Multi-Plataforma
- Segurança

IDE

Integrated Development Environment

- Eclipse
- NetBeans
- BlueJ
- Notepad



PRIMEIRO PROGRAMA EM JAVA

Primeiros passos

- Baixar, instalar e configurar o JDK no SO:
 - <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
- Codificar:
 - Utilizar qualquer editor (texto puro)
 - bloco de notas, wordpad , jedit, vi, kedit, pico, gedit, ...
 - Salvar arquivo com a extensão .java
 - Ex.: Nome_Classe.java

Java SE

Java EE

Java ME

Java SE Advanced & Suite

Java Embedded

Java DB

Web Tier

Java Card

Java TV

New to Java

Community

Java Magazine

Overview

Downloads

Documentation

Community

Technologies

Training

Java SE Downloads



Java Platform (JDK) 9



NetBeans with JDK 8

Java Platform, Standard Edition

Java SE 9.0.4

Java SE 9.0.4 includes important bug fixes. Oracle strongly recommends that all Java SE 9 users upgrade to this release.

[Learn more](#)

- Installation Instructions
- Release Notes
- Oracle License
- Java SE Licensing Information User Manual
 - Includes Third Party Licenses
- Certified System Configurations
- Readme

JDK

DOWNLOAD

Server JRE

DOWNLOAD

JRE

DOWNLOAD

Java SDKs and Tools

- [Java SE](#)
- [Java EE and Glassfish](#)
- [Java ME](#)
- [Java Card](#)
- [NetBeans IDE](#)
- [Java Mission Control](#)

Java Resources

- [Java APIs](#)
- [Technical Articles](#)
- [Demos and Videos](#)
- [Forums](#)
- [Java Magazine](#)
- [Developer Training](#)



Tutorial para configurar variáveis de ambiente:

<https://www.devmedia.com.br/preparacao-do-ambiente-para-desenvolvimento-em-java/25188>

Compilação e execução

- Compilar :
 - **javac** Nome_Classe.java
- Executar :
 - **java** Nome_Classe

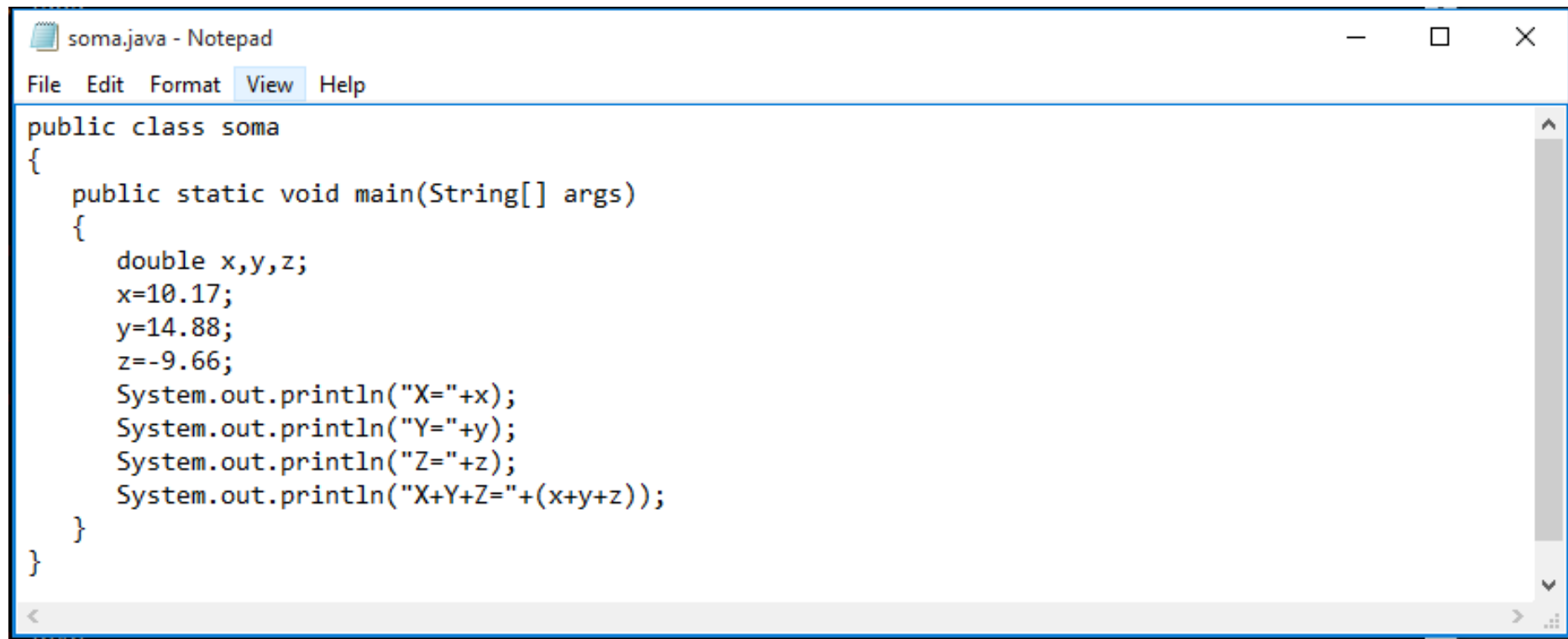
Exemplo 1

- `OlaMundo.java`

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

Exemplo 2

- soma.java

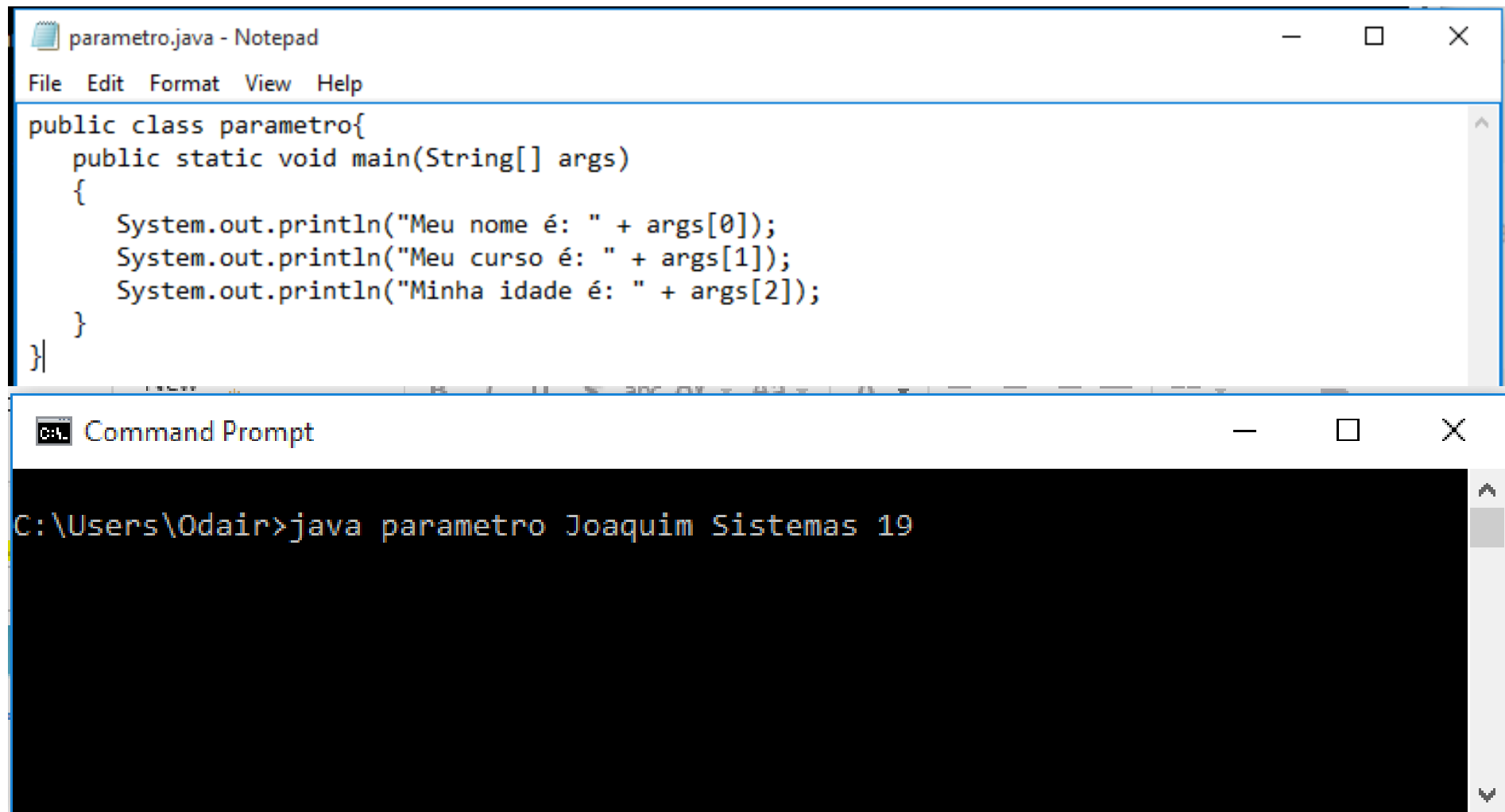
A screenshot of a Notepad window titled "soma.java - Notepad". The window has a menu bar with "File", "Edit", "Format", "View", and "Help". The text area contains the following Java code:

```
public class soma
{
    public static void main(String[] args)
    {
        double x,y,z;
        x=10.17;
        y=14.88;
        z=-9.66;
        System.out.println("X="+x);
        System.out.println("Y="+y);
        System.out.println("Z="+z);
        System.out.println("X+Y+Z="+(x+y+z));
    }
}
```

The code defines a public class named 'soma' with a main method. Inside the main method, three double variables x, y, and z are declared and assigned values 10.17, 14.88, and -9.66 respectively. Then, four lines of output are printed using System.out.println, showing the individual values and their sum.

Exemplo 3

- parametro.java



The image shows two overlapping windows. The top window is a Notepad editor titled 'parametro.java - Notepad'. It contains the following Java code:

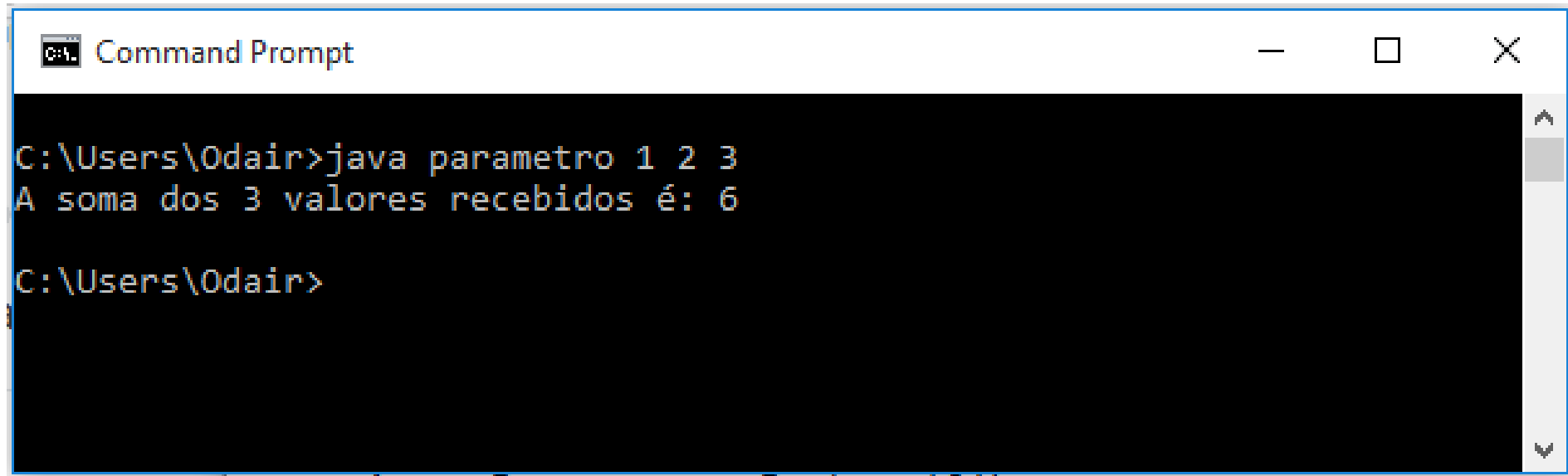
```
public class parametro{  
    public static void main(String[] args)  
    {  
        System.out.println("Meu nome é: " + args[0]);  
        System.out.println("Meu curso é: " + args[1]);  
        System.out.println("Minha idade é: " + args[2]);  
    }  
}
```

The bottom window is a Command Prompt titled 'Command Prompt'. It shows the command being executed:

```
C:\Users\Odair>java parametro Joaquim Sistemas 19
```

Exemplo 4

- Faça um programa que receba 3 parâmetros numéricos inteiros e apresente sua soma, conforme exemplo abaixo.



```
Command Prompt

C:\Users\Odair>java parametro 1 2 3
A soma dos 3 valores recebidos é: 6

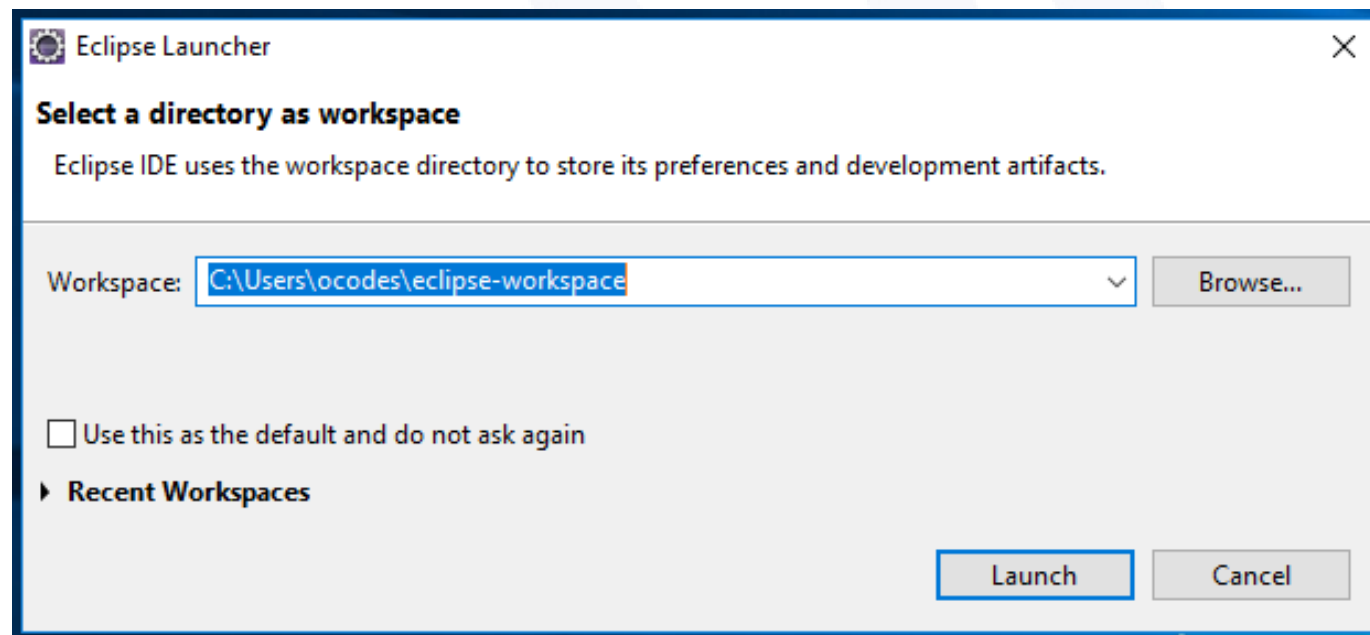
C:\Users\Odair>
```

Eclipse

- **Eclipse** é um IDE para desenvolvimento Java, porém suporta várias outras linguagens a partir de plugins como C/C++, PHP, ColdFusion, Python, Scala e plataforma Android.
- Ele foi desenvolvido em Java e segue o modelo open source de desenvolvimento de software.
- Fonte: [https://pt.wikipedia.org/wiki/Eclipse_\(software\)](https://pt.wikipedia.org/wiki/Eclipse_(software))

Como Criar um Novo Projeto

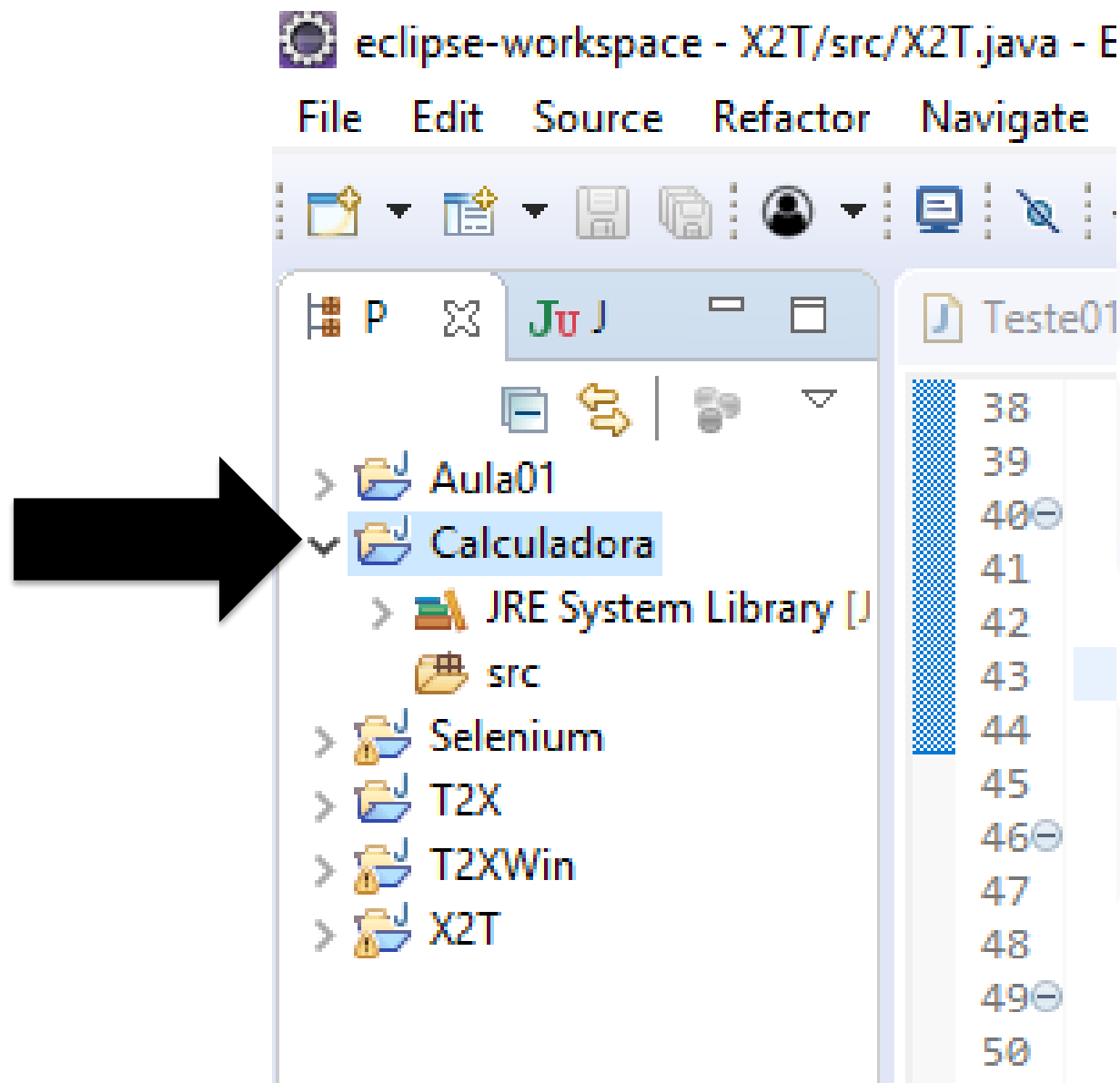
- 1-Execute o Eclipse
- 2-Indique a pasta os projetos serão criados.



Como Criar um Novo Projeto

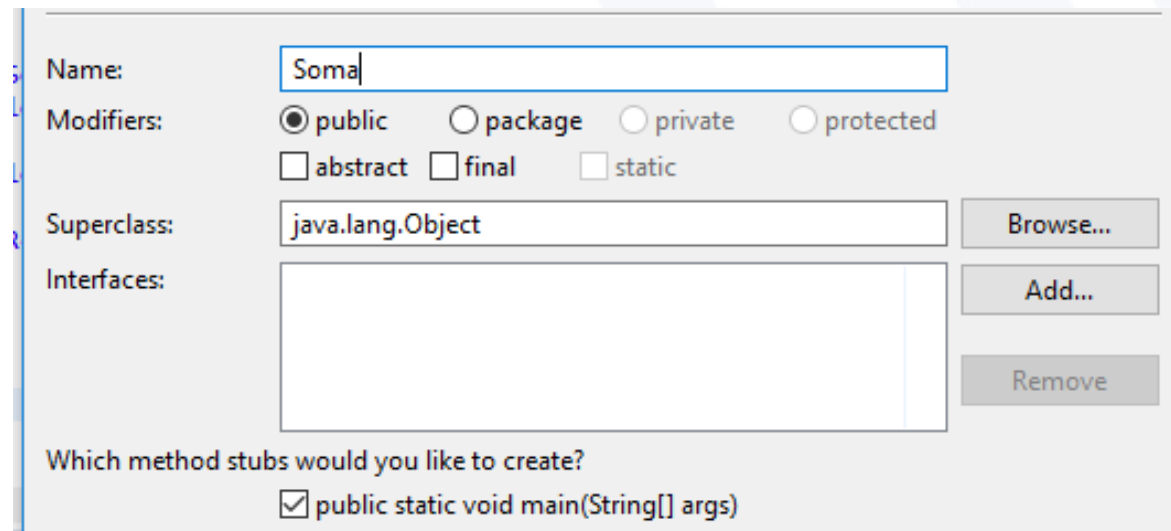
- 3-No Eclipse, selecione File -> New -> Java Project.
- 4-Escreva o nome projeto no campo Project name, por exemplo, Calculadora.
- 5-Clique no botão Finish.

Como Criar um Novo Projeto



Criar uma Nova Classe

- 1-Clique com o botão da direita sobre o nome do projeto (Calculadora).
- 2-Selezione New -> Class
- 3-Informe o nome da Classe e clique na checkbox “public static void main...”



Name:

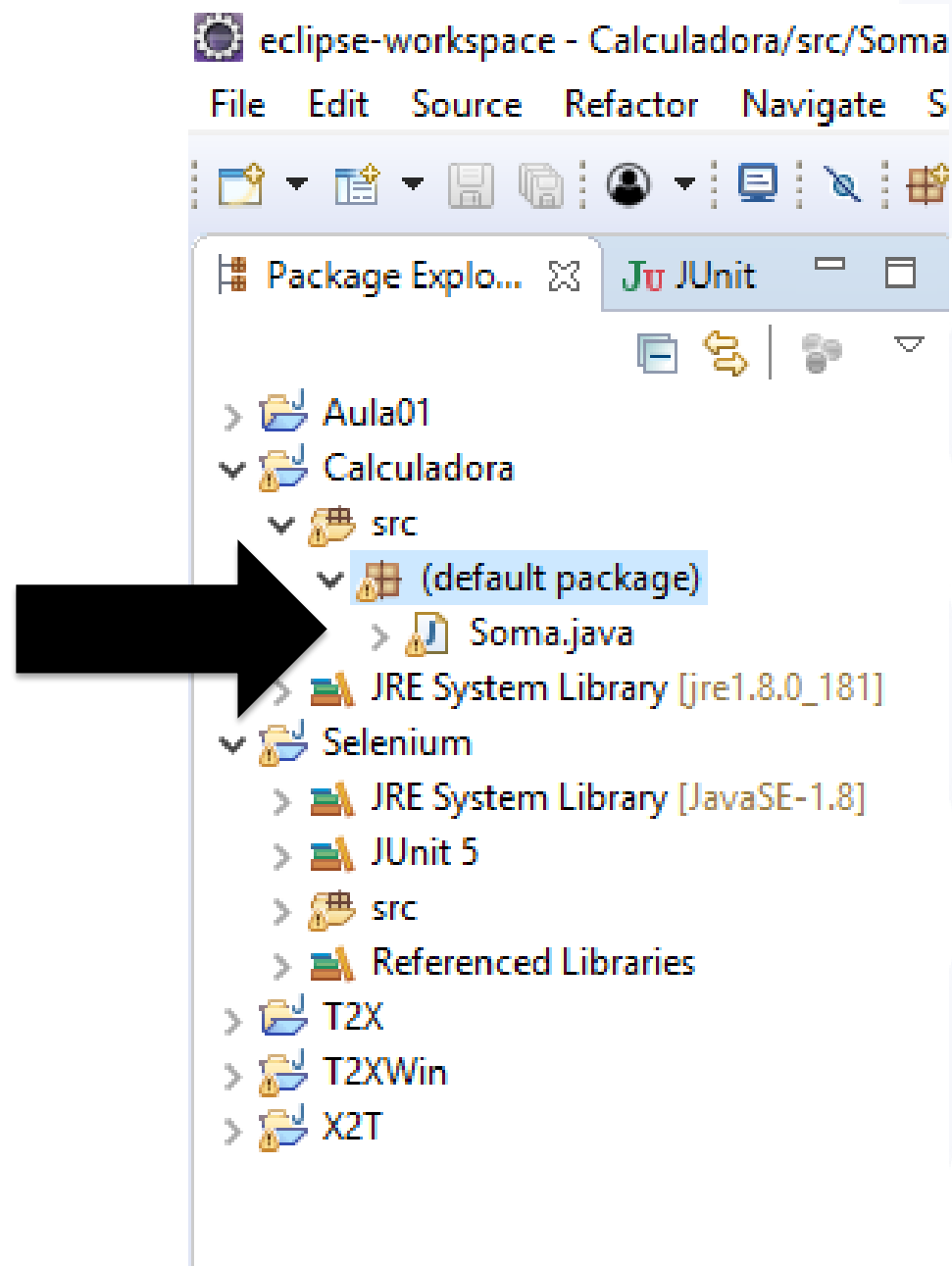
Modifiers: ☒ public ☐ package ☐ private ☐ protected
☐ abstract ☐ final ☐ static

Superclass:

Interfaces:

Which method stubs would you like to create?
☒ public static void main(String[] args)

Criar uma Nova Classe



Classe Soma

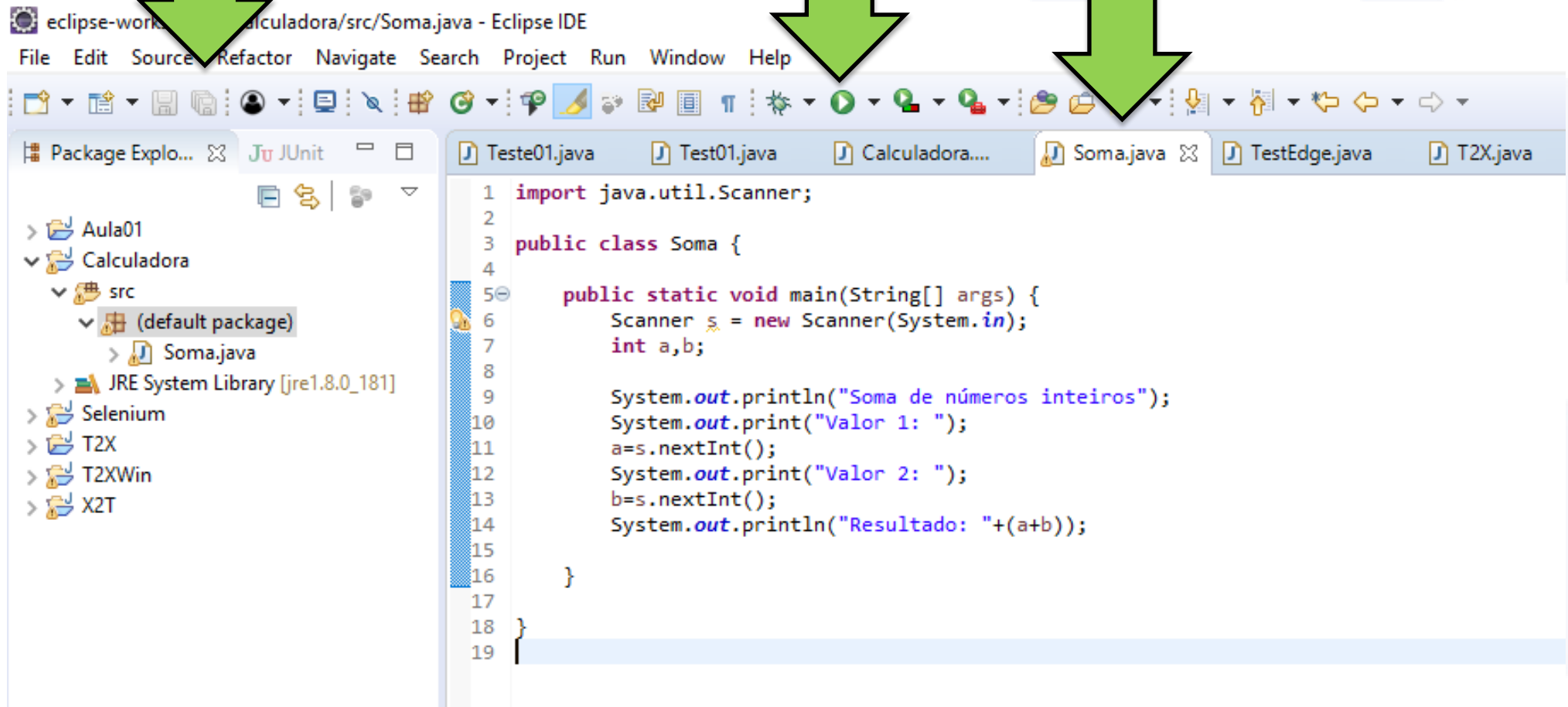
The screenshot displays an IDE interface with the following components:

- Package Explorer (Left):** Shows a project structure with folders 'Aula01', 'Calculadora', 'Selenium', 'T2X', 'T2XWin', and 'X2T'. The 'Calculadora' folder is expanded, showing a 'src' folder containing a '(default package)' folder, which in turn contains the 'Soma.java' file.
- Editor (Center):** Displays the code for 'Soma.java'. The code is as follows:

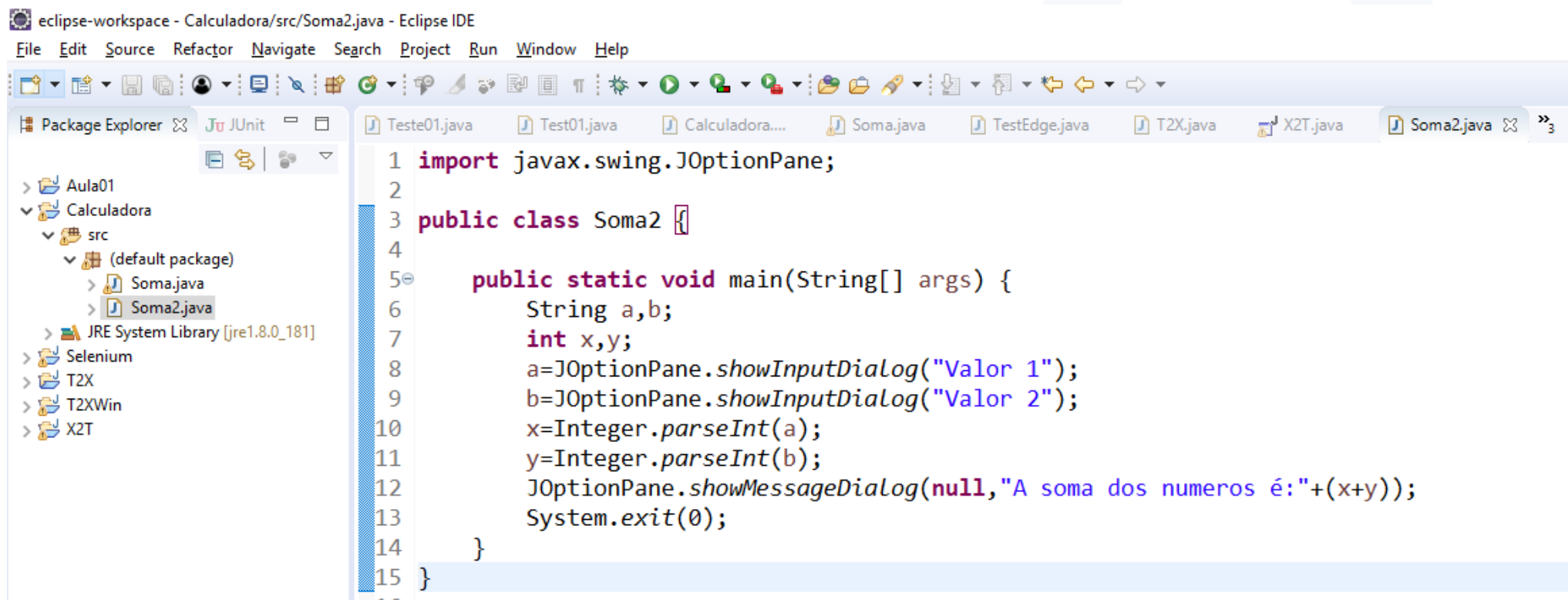
```
1 import java.util.Scanner;
2
3 public class Soma {
4
5     public static void main(String[] args) {
6         Scanner s = new Scanner(System.in);
7         int a,b;
8
9         System.out.println("Soma de números inteiros");
10        System.out.print("Valor 1: ");
11        a=s.nextInt();
12        System.out.print("Valor 2: ");
13        b=s.nextInt();
14        System.out.println("Resultado: "+(a+b));
15        System.exit(0);
16    }
17
18 }
19
```
- Console (Bottom):** Shows the execution output for 'Soma [Java Application]'. The output is:

```
<terminated> Soma [Java Application] C:\Program Files\Java\jre1.8.0_181\bin\javaw.exe (Aug 13, 2018, 1:31:26 PM)
Soma de números inteiros
Valor 1: 60
Valor 2: 60
Resultado: 120
```

Executar



Entrada com JOptionPane



The screenshot shows the Eclipse IDE interface. The Package Explorer on the left displays a project named 'Calculadora' with a source folder 'src' containing 'Soma2.java'. The main editor window shows the following Java code:

```
1 import javax.swing.JOptionPane;
2
3 public class Soma2 {
4
5     public static void main(String[] args) {
6         String a,b;
7         int x,y;
8         a=JOptionPane.showInputDialog("Valor 1");
9         b=JOptionPane.showInputDialog("Valor 2");
10        x=Integer.parseInt(a);
11        y=Integer.parseInt(b);
12        JOptionPane.showMessageDialog(null,"A soma dos numeros é:"+(x+y));
13        System.exit(0);
14    }
15 }
```

Exercícios

- 1-Fazer as classes Subtração, Produto e Divisão (tratar a divisão por zero).
- 2-Fazer um programa que calcule o IMC. $IMC = massa / (altura * altura)$. *Utilizar variáveis float e o método `Float.parseFloat()`.*