



# -Introdução ao Java e NetBeans

Disciplina: Linguagem de Programação  
Profa. Me Juliana Pasquini  
18/8/2025

# Roteiro:

---

- ✓ Conceitos básicos sobre Java
- ✓ Instalação do NetBeans
- ✓ Criação de um projeto no NetBeans
- ✓ Interface Gráfica

# Introdução a linguagem Java

- ✓ Java é uma linguagem de programação orientada a objeto lançada pela Sun em 1995



- ✓ A evolução da linguagem é controlada pelo Java Community Process ([www.jcp.org](http://www.jcp.org)).

# Linguagem Java

Java é uma linguagem de **programação interpretada** e seu funcionamento depende de uma plataforma conhecida como **Máquina Virtual Java – JVM** (do inglês Java Virtual Machine).

Java é considerada **multiplataforma** pois o desenvolvimento de uma aplicação não depende do sistema operacional usado. Esse conceito é conhecido como: **“Write once, run anywhere”**. (*“Escreva uma vez, execute em qualquer lugar”*). Foi a principal propaganda da Sun ao lançar a linguagem Java.

ByteCode



JVM



SO

# Compilação para bytecode

Programas Java não são traduzidos diretamente para a **linguagem de máquina** como outras linguagens estaticamente compiladas e sim para uma representação intermediária, chamada de **bytecodes**.

# Compilação para bytecode

## Exemplo

Nome do arquivo: HelloWorld.java

Um programa Java  
em formato texto

```
public class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Hello, world!");  
    }  
}
```



O processo de compilação transforma o  
código texto em um programa em bytecodes

F4 D9 00 03 0A B2 FE FF FF 09 02 01 01 2E 2F 30 62 84 3D 29 3A C1

O programa fonte possui a extensão  
.java e o de saída .class

Nome do arquivo:  
HelloWorld.class

# Linguagem Java

O que são JVM, JDK e JRE?

**JVM:** é a **java virtual machine**, não é possível fazer o download dela sozinha, a JVM sempre vem acompanhada.

**JRE:** Java Runtime Environment, **ambiente de execução Java**, formado pela **JVM e bibliotecas**, tudo que se precisa para executar uma aplicação Java. Mas não é suficiente para desenvolvimento de aplicações.

**JDK:** Java Development Kit: Kit necessário para o desenvolvimento de aplicações. Deve-se fazer o download do JDK do Java SE (Standard Edition). Ele é formado pela JRE somado a ferramentas, como o compilador de bytecode javac.

# Sumarizando

Arquivo texto  
(código-fonte .java)



**Compilação (análise lexicográfica e semântica) usa o JDK (Java Development Kit)**

Bytecode .class  
(para todas as arquiteturas)

Portabilidade: independência de plataforma – “write once run anywhere”



**Execução pela JVM, que na verdade é o JRE (Java Runtime Environment)**

Código de máquina  
(executado pelo hardware)



# Ambiente de desenvolvimento de software

- ✓ Os Ambientes Integrados de Desenvolvimento de software são conhecidos em inglês pela sigla IDE (Integrated Development Environment)
- ✓ Os IDEs são apenas ambientes que utilizamos para criar nossos programas, ou seja, são ambientes onde digitamos, debugamos, compilamos e executamos nossos programas

**Observação: não se deve confundir a linguagem de programação Java com o ambiente usado para o desenvolvimento dos nossos programas em Java.**

# Ambiente de desenvolvimento de software

- ✓ Na verdade, para desenvolvermos nossos programas em Java não precisamos utilizar um IDE, pois podemos digitá-los em uma ferramenta qualquer de texto, como o Bloco de Notas, e para compilarmos precisamos utilizar o JDK (Java Development Kit).

# Ambiente de desenvolvimento de software,



- ✓ Independente do ambiente, que utilizamos para escrevermos o programa, precisamos do JDK para a compilação e do JRE (Java Runtime Environment) para a execução
- ✓ O JRE é também conhecido por JVM

# Instalação do NetBeans

- ✓ Os IDEs gratuitos mais utilizados para o desenvolvimento de software são o NetBeans, Eclipse e VSCode. Ambos possuem muitos recursos que auxiliam o programador
- ✓ Aqui utilizaremos o NetBeans

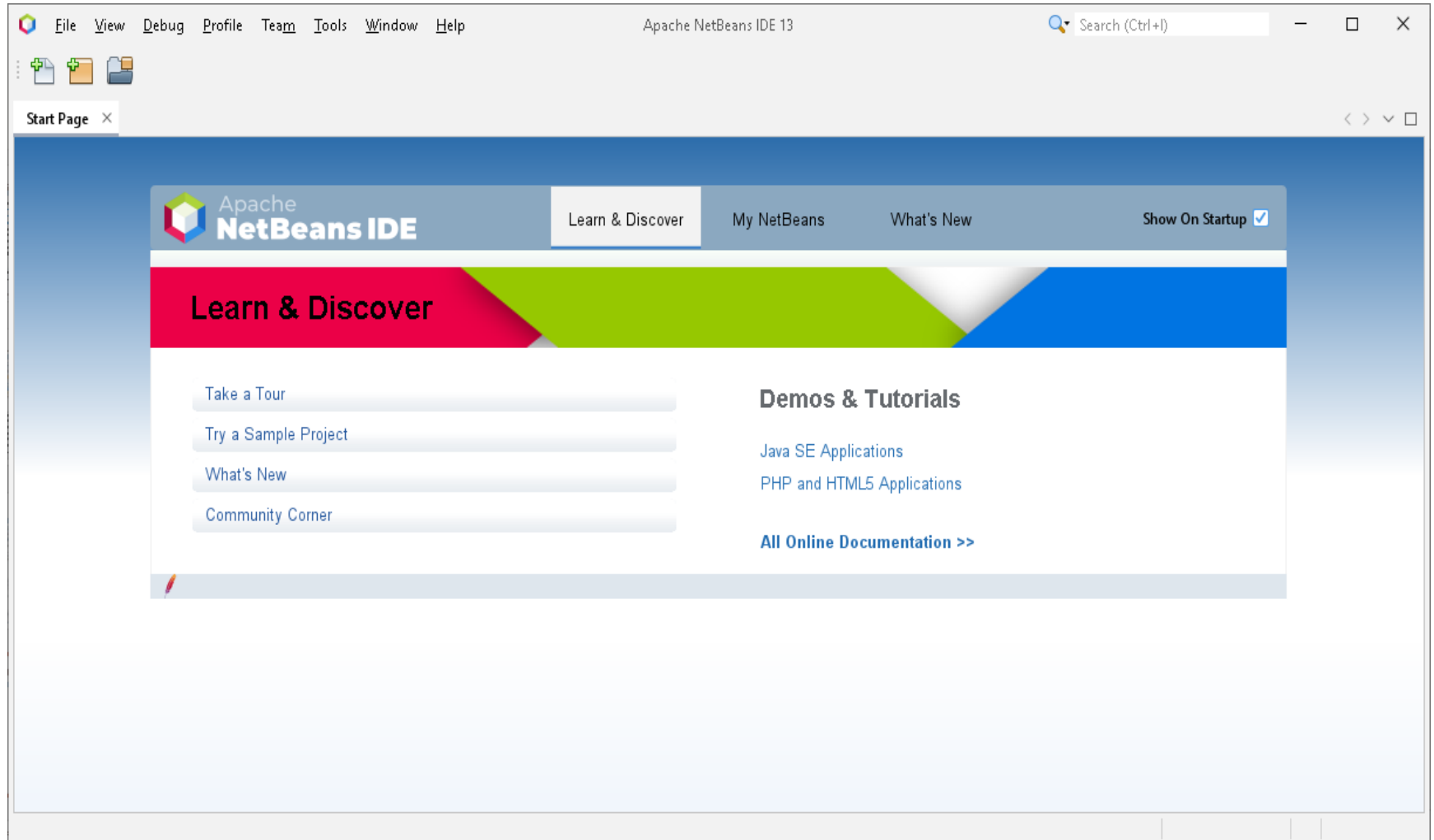
# Instalação do NetBeans

(continuação)

- ✓ Para instalar este IDE precisamos baixá-lo de [www.netbeans.org](http://www.netbeans.org) e também precisamos baixar o JDK e JRE.
- ✓ Instale primeiro o JDK e JRE, para depois instalar o NetBeans  
<http://www.oracle.com/technetwork/java/javase/downloads/>
- ✓ Em 2008 a Oracle Corporation adquire a empresa responsável pela linguagem Java, a Sun Microsystems.
- ✓ OBS: Tutorial de instalação está no Teams.

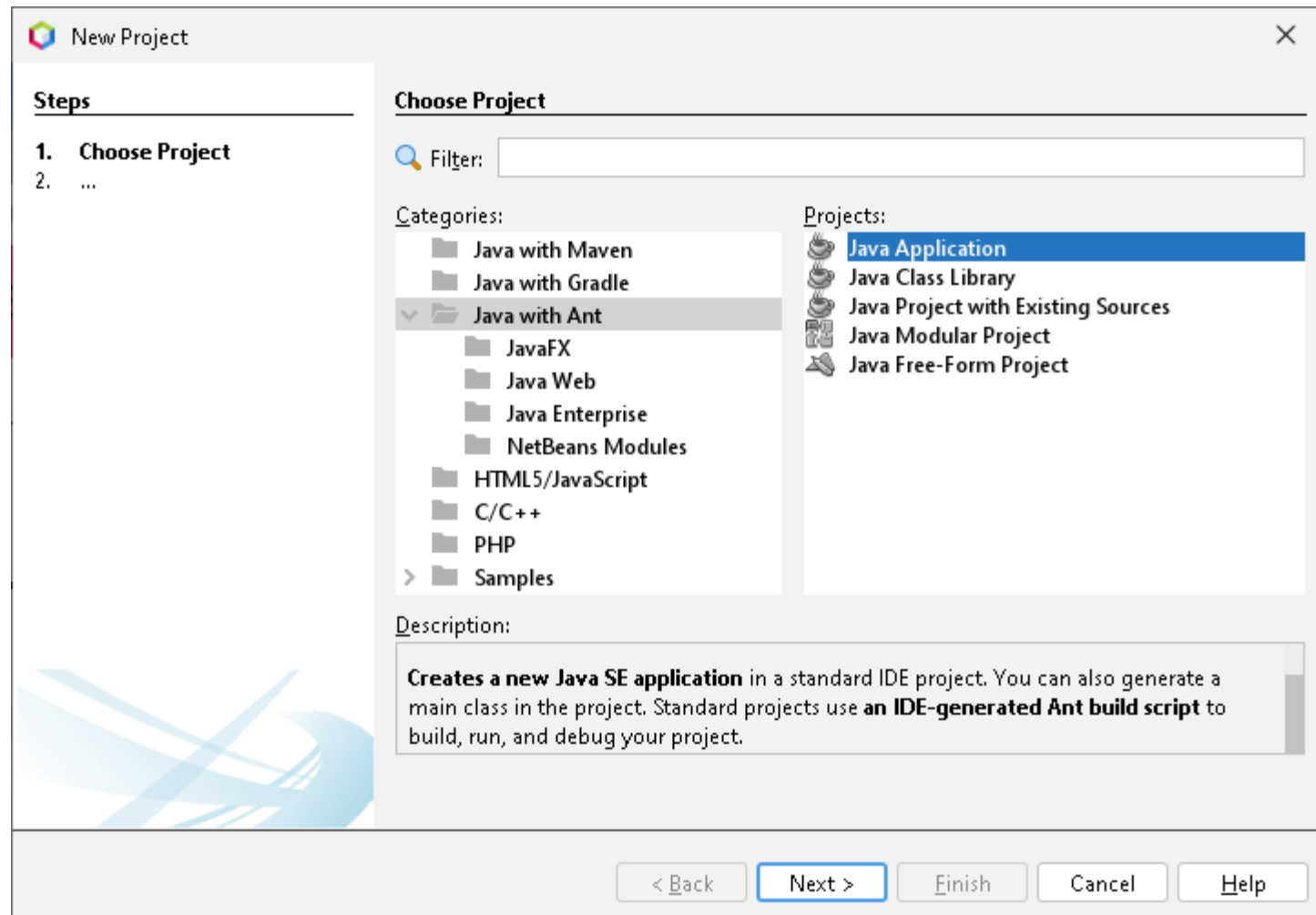
# Tela de abertura (Apache NetBeans – Versão 13)

20



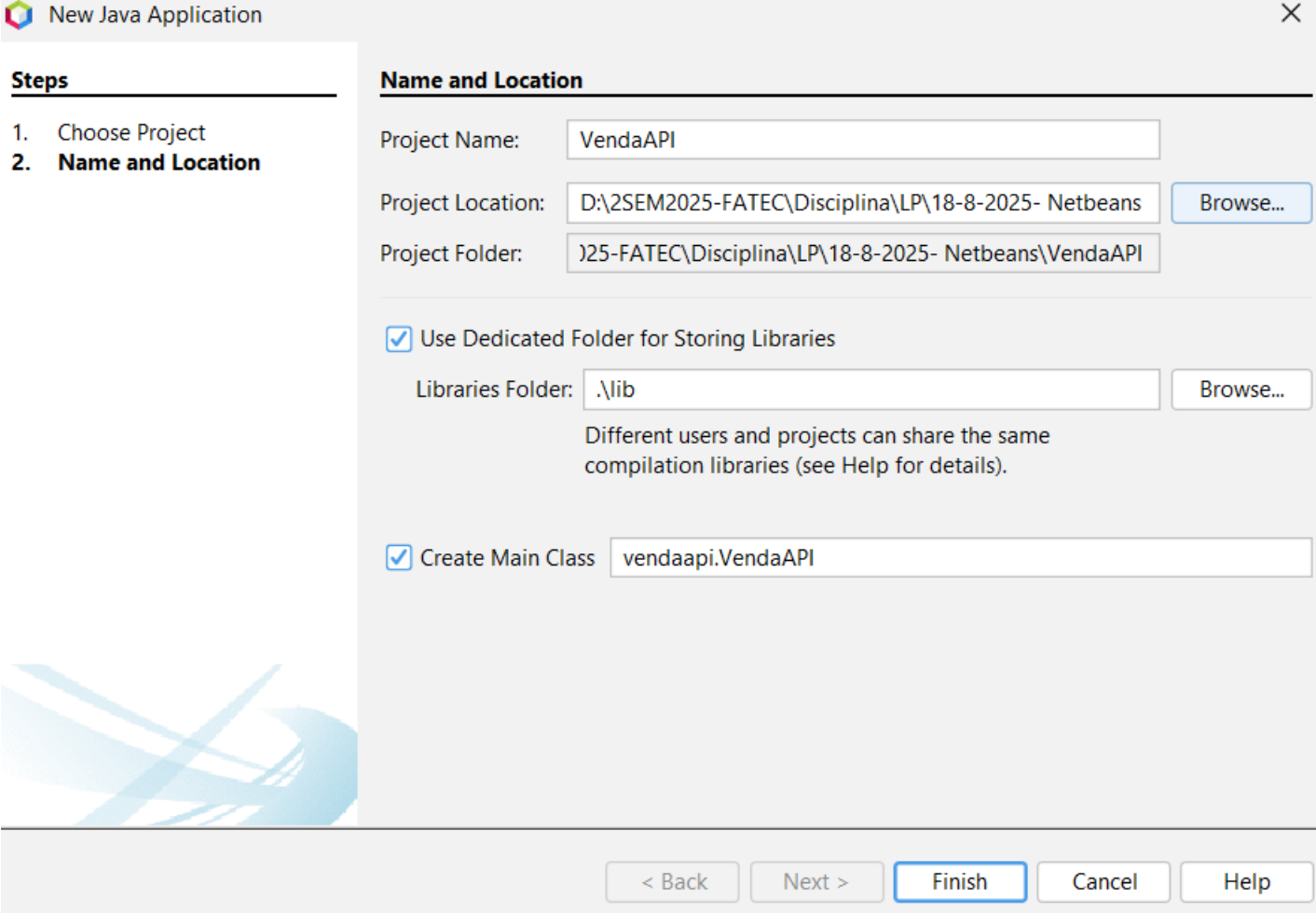
# Criando um Projeto (Apache NetBeans – Versão 13)

21



# Criando um Projeto (Apache NetBeans – Versão 13)

22



New Java Application

**Steps**

1. Choose Project
2. **Name and Location**

**Name and Location**

Project Name: VendaAPI

Project Location: D:\2SEM2025-FATEC\Disciplina\LP\18-8-2025- Netbeans [Browse...](#)

Project Folder: 25-FATEC\Disciplina\LP\18-8-2025- Netbeans\VendaAPI

☒ Use Dedicated Folder for Storing Libraries

Libraries Folder: .\lib [Browse...](#)

Different users and projects can share the same compilation libraries (see Help for details).

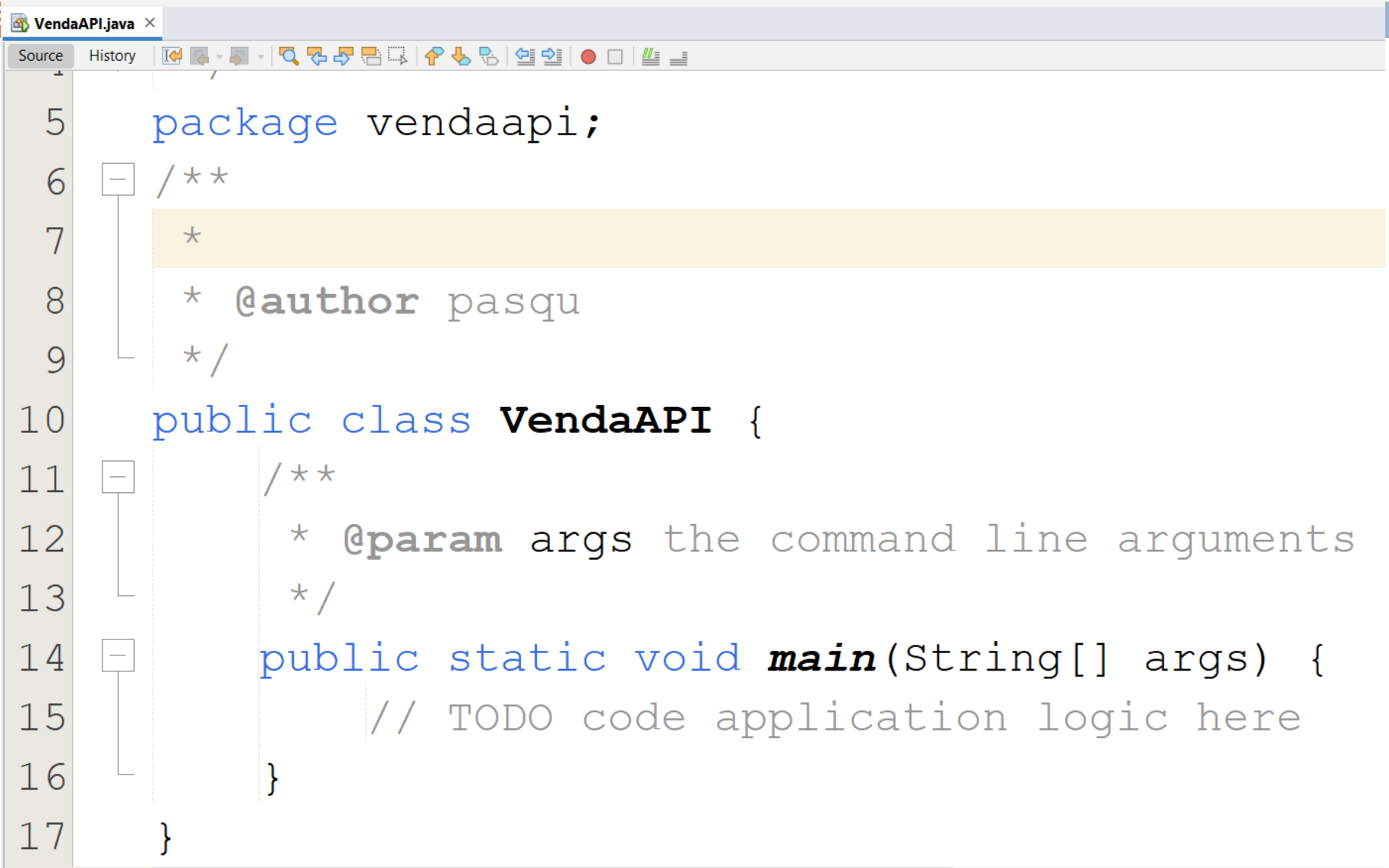
☒ Create Main Class vendaapi.VendaAPI

< Back Next > **Finish** Cancel Help



# Estrutura de uma Classe

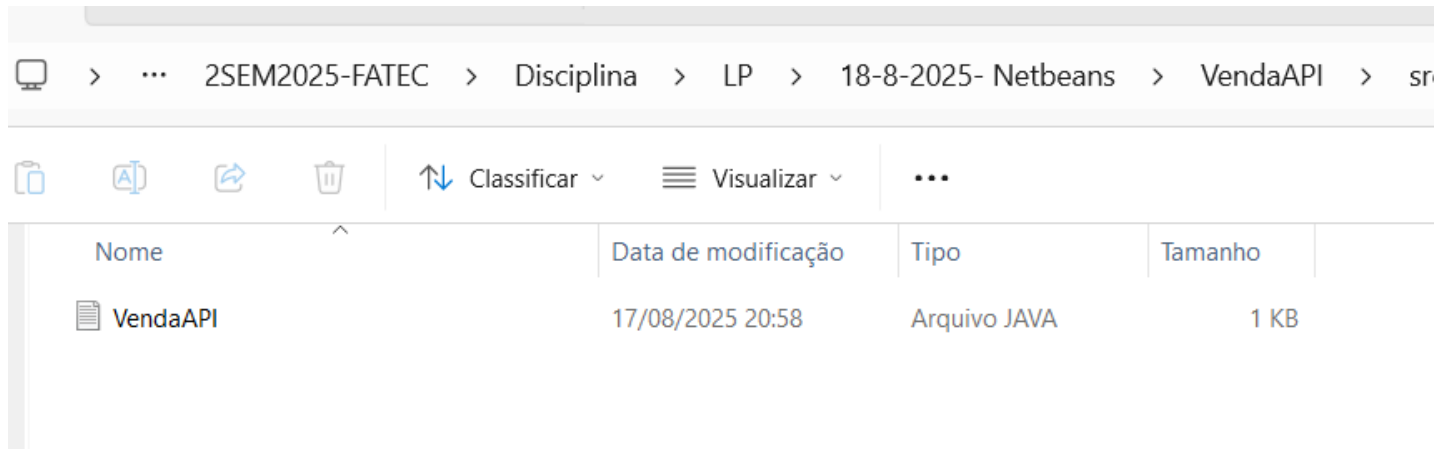
(Apache NetBeans – Versão 13)



```
5 package vendaapi;
6 /**
7  *
8  * @author pasqu
9  */
10 public class VendaAPI {
11     /**
12      * @param args the command line arguments
13      */
14     public static void main(String[] args) {
15         // TODO code application logic here
16     }
17 }
```

# Fonte do Programa – VendaAPI.java

24

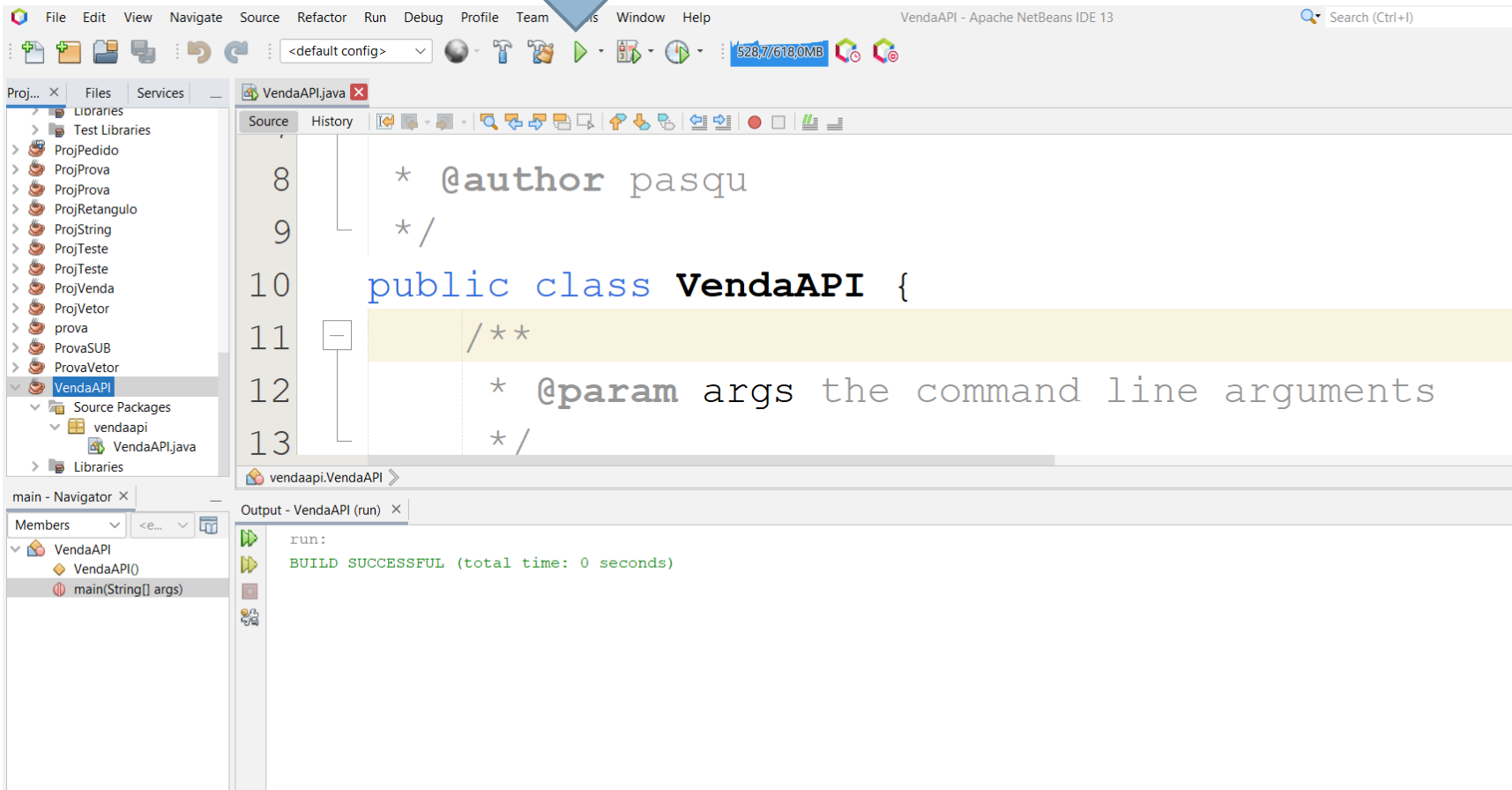


The image shows a file explorer window with a breadcrumb path: > ... 2SEM2025-FATEC > Disciplina > LP > 18-8-2025- Netbeans > VendaAPI > sr. Below the path is a toolbar with icons for file operations and a menu with 'Classificar' and 'Visualizar'. The main area is a table listing files.

Nome	Data de modificação	Tipo	Tamanho
VendaAPI	17/08/2025 20:58	Arquivo JAVA	1 KB

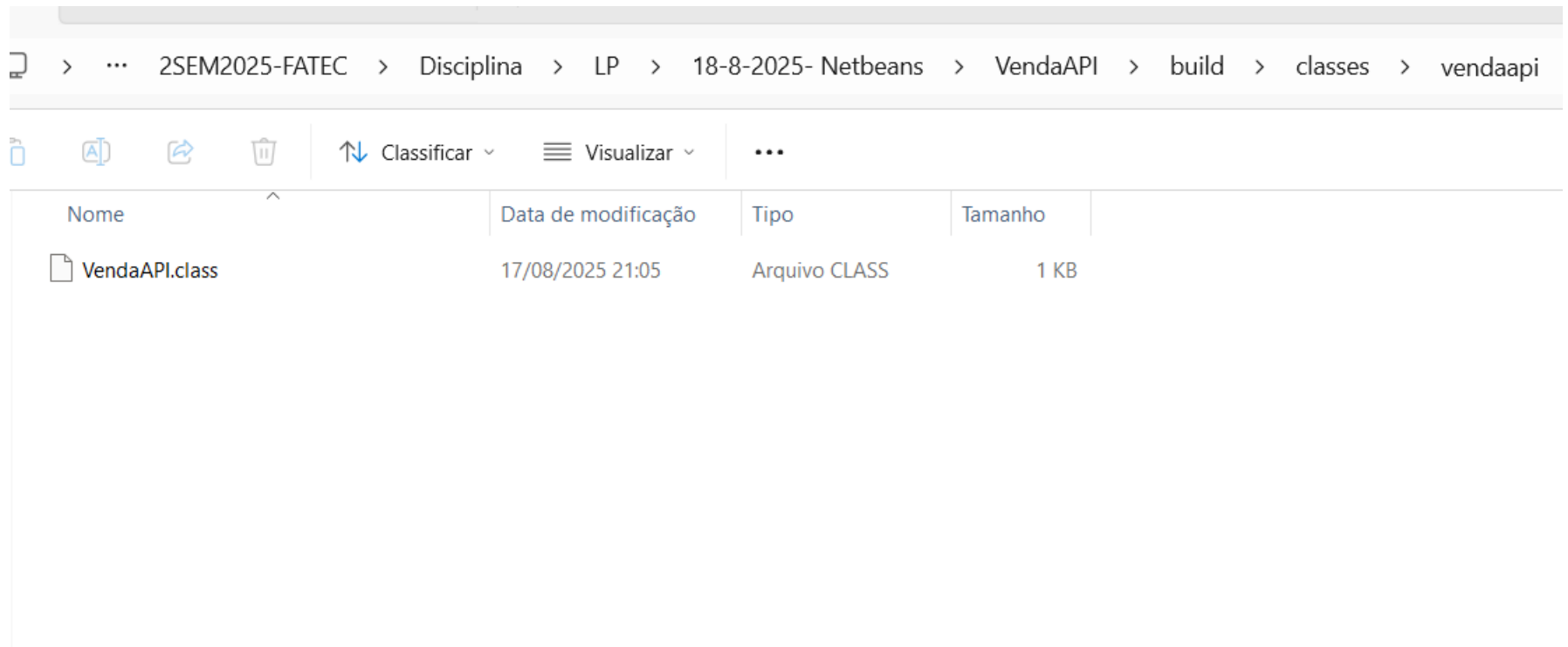
# Executando o Projeto


25



# Saída VendaAPI.class

26



2SEM2025-FATEC > Disciplina > LP > 18-8-2025- Netbeans > VendaAPI > build > classes > vendaapi				
Classificar ↑ ↓ Visualizar ☰ ...				
Nome	Data de modificação	Tipo	Tamanho	
 VendaAPI.class	17/08/2025 21:05	Arquivo CLASS	1 KB	

# Boas práticas de programação Java

- ✓ Usa-se nomear as classes Java capitalizando as palavras, por exemplo, `Principal` e `MinhaPrimeiraClasse`
- ✓ Usa-se nomear os métodos com letras minúsculas, por exemplo, `main`

# Boas práticas de programação Java

- ✓ Usa-se nomear as classes e métodos com nomes pertinentes e quando estes contêm espaços, por exemplo, `minha primeira classe` e `meu primeiro método`, deve-se nomear retirando os espaços e capitalizando as palavras, desta forma ficaria:
  - ✓ `MinhaPrimeiraClasse` para o nome da classe
  - ✓ `meuPrimeiroMetodo` no caso do método usá-se a primeira letra em minúsculo

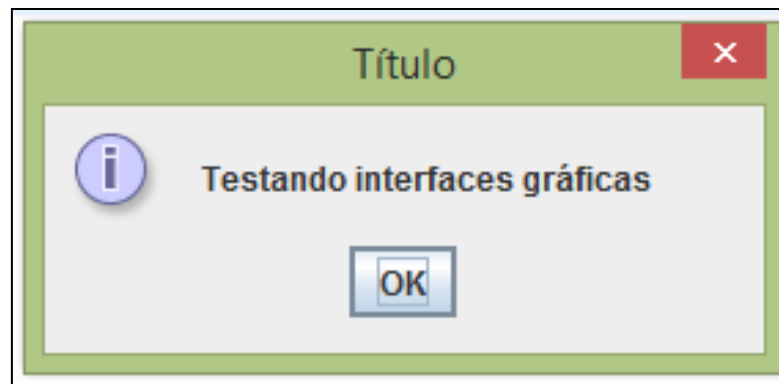
# Exercícios - 01

29

- **Exercício 1** - Fazer um programa que imprime os números de 1 a 10 na tela.

# Interface Gráfica

Uma interface gráfica com o usuário (Graphical User Interface - GUI) apresenta um mecanismo amigável ao usuário para interagir com um aplicativo.



GUIs são muito úteis pois dão uma aparência e um comportamento peculiar aos aplicativos. Fornecer aos aplicativos diferentes componentes de interface com o usuário acelera a aprendizagem de uso do aplicativo.



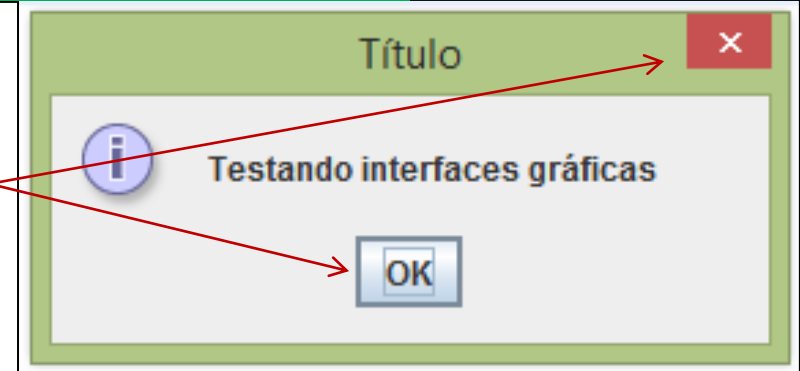
# Interface Gráfica

A classe JOptionPane do Java (javax.swing) fornece caixas de diálogo pré-empacotadas tanto para entrada como para saída.

```
public static void main(String[] args){  
    JOptionPane.showMessageDialog(null, "Testando interfaces gráficas",  
                                "Título", 1);  
}
```

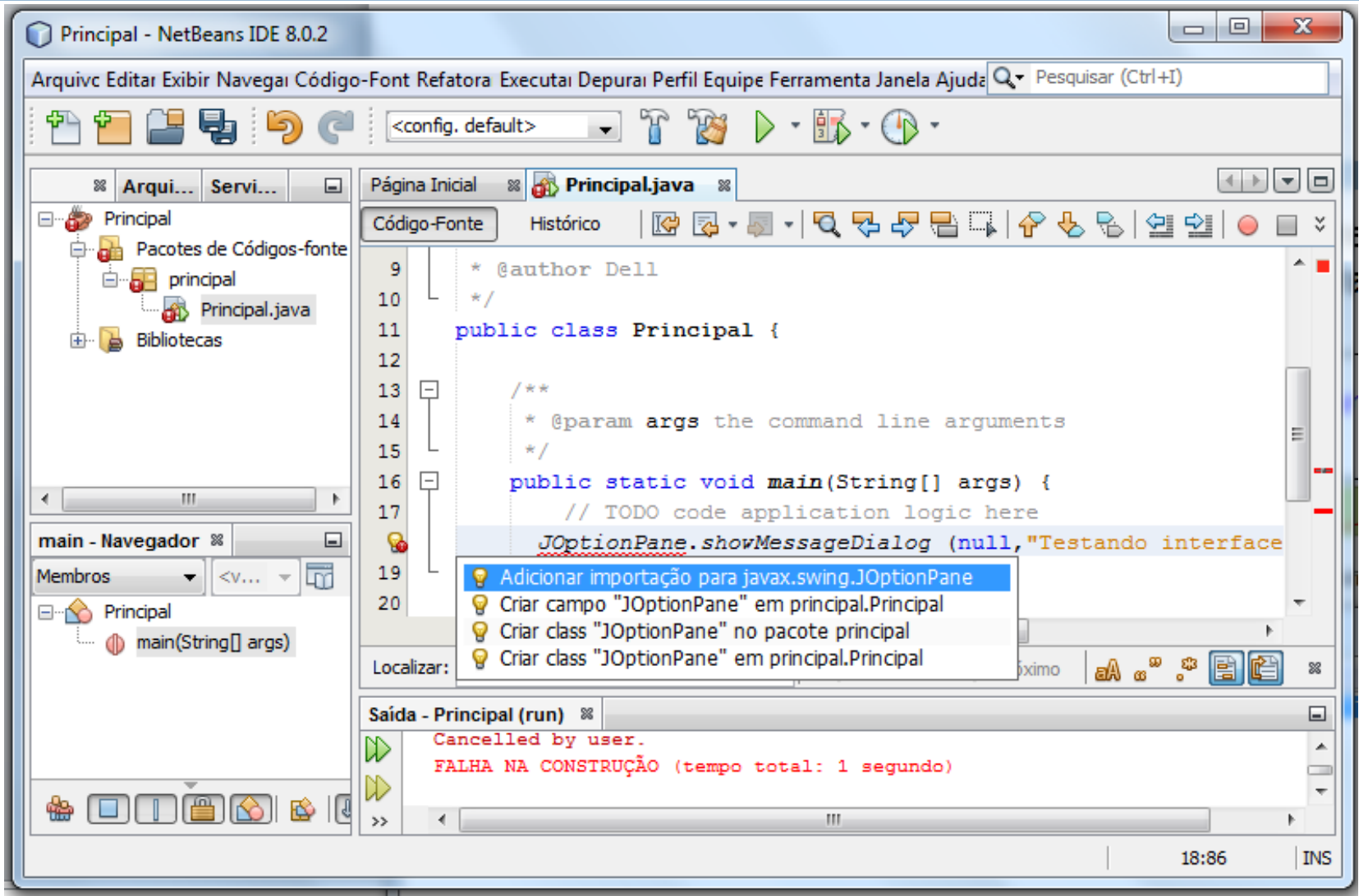
Trocar para 2 e depois 3

Os botões OK e fechar são fornecidos.



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

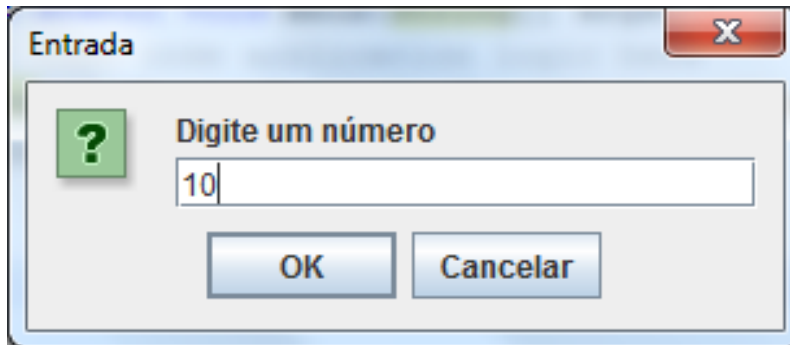
# Interface Gráfica



# Interface Gráfica

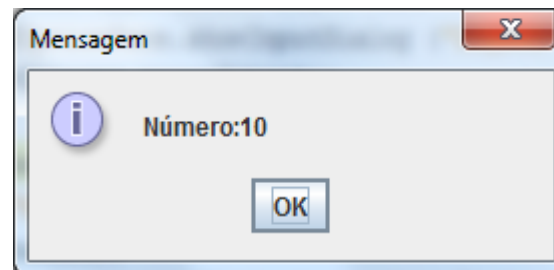
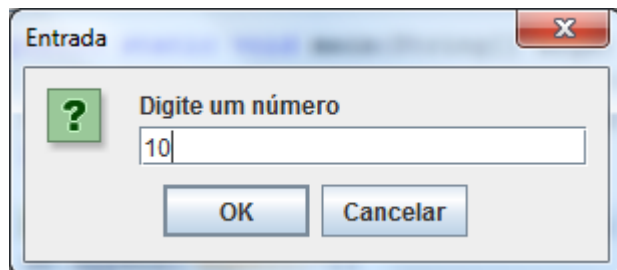
É possível receber valores por meio dos métodos da classe JOptionPane.

```
public static void main(String[] args) {  
    // TODO code application logic here  
    String n1=JOptionPane.showInputDialog ("Digite um número");  
    int numero=Integer.parseInt(n1);  
}
```



# Interface Gráfica

```
public static void main(String[] args) {  
    // TODO code application logic here  
    String n1=JOptionPane.showInputDialog ("Digite um número");  
    int numero=Integer.parseInt(n1);  
  
    StringBuilder sb=new StringBuilder();  
    sb.append("Número:");  
    sb.append(numero);  
    JOptionPane.showMessageDialog(null,sb.toString());  
}  
}
```



# Exercícios - 02

36

- Faça um programa que leia dois números fornecidos pelo usuário e exibe a soma desses números.

# Sites Importantes

- ✓ GUJ (<http://www.guj.com.br>)
  - ✓ Sun (<https://www.oracle.com/sun/index.html>)
  - ✓ comitê (<http://www.jcp.org>)
  - ✓ <http://www.oracle.com/technetwork/java/>
  - ✓ história da linguagem Java em: <http://www.java.com/en/javahistory/>
  - ✓ [Tutorial para Início Rápido do Java do NetBeans IDE \(apache.org\)](http://www.apache.org/netbeans/tutorial/JavaNetBeansIDE.html)
- [https://netbeans.apache.org/kb/docs/java/quickstart\\_pt\\_BR.html](https://netbeans.apache.org/kb/docs/java/quickstart_pt_BR.html)