

Introdução a Java E/S

P. O. O.

Prof. Grace



Em Programação Orientada a Objetos

Classe



Abstração

Molde

“Codifico apenas 1 vez”



Objeto



Concreto

Ocorrência real

“Instancio quantos precisar a partir de uma mesma classe”



Membros de uma Classe

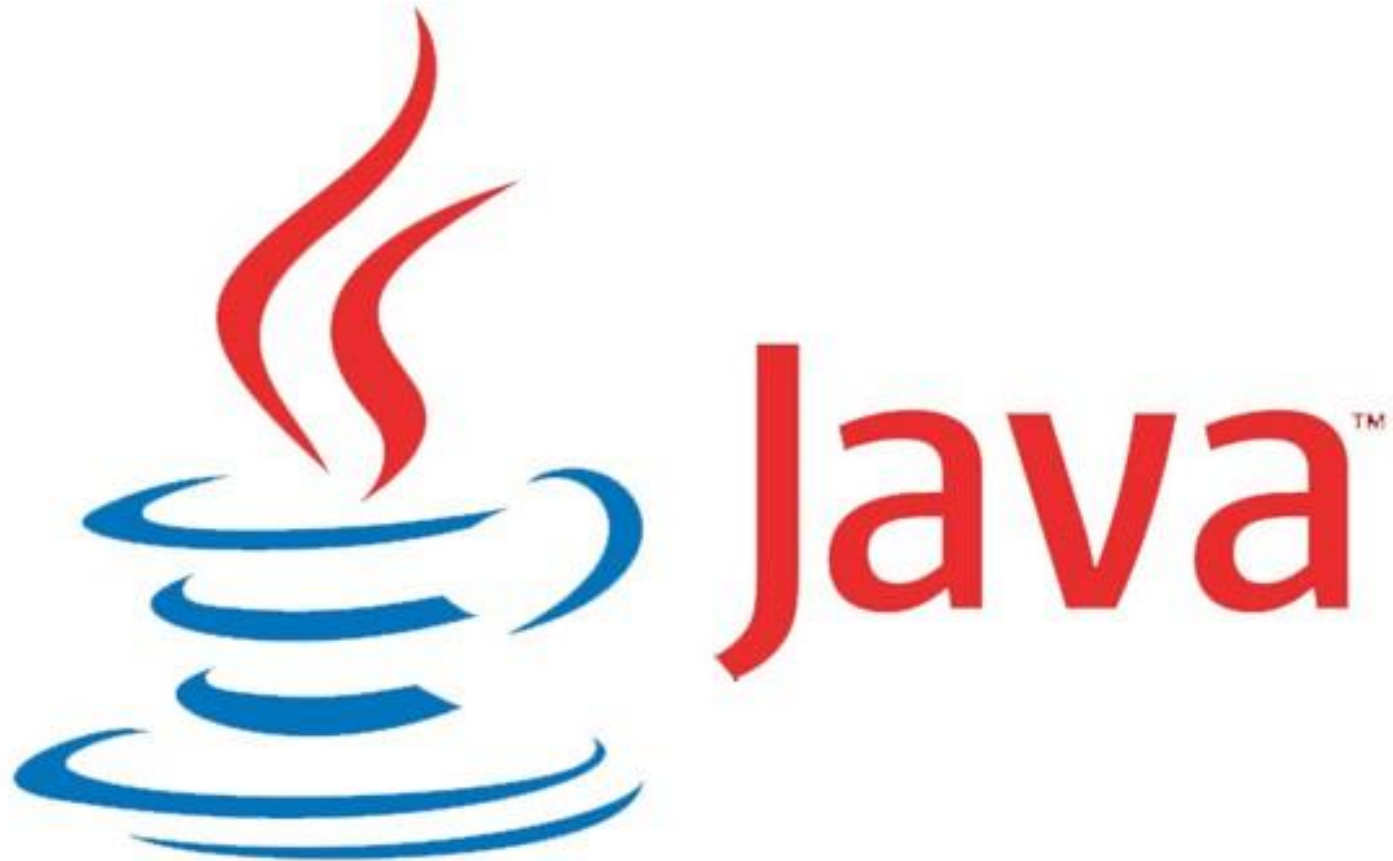
Uma classe é composta por:

- Comportamentos ou operações
- Características ou atributos
(modelo)

OU

- Métodos ou funções
- Dados ou variáveis
(implementação)

Conhecendo o Java





Linguagem Java

- Programas são **classes** que utilizam outras **classes**:
 - Classes programadas por nós, programadores;
 - Bibliotecas de classes já existentes: Java API's;
- Vantagem de reuso: não precisamos “reinventar a roda”;
- Entretanto, nem todas as classes existentes são adequadas para o que precisamos => precisamos programar nossas próprias classes



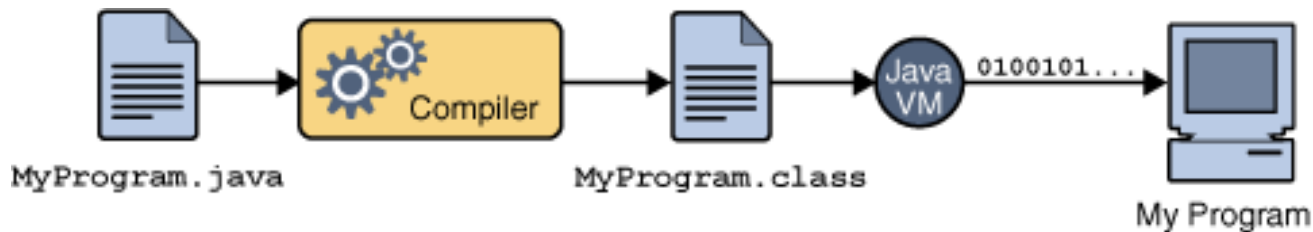
Plataforma JAVA

- A linguagem java exige ambiente apropriado;
- Plataforma Java: ambiente desenvolvido pela Sun, (adquirida pela Oracle), para execução e desenvolvimento de sw independentes de S.O.
- Ambiente de execução de aplicações JRE (Java Runtime Environment)
 - JVM (Java Virtual Machine) + API's



Plataforma JAVA

- Mecanismos de tradução
 - Código compilado (para linguagem intermediária: bytecode)
 - Código interpretado (de bytecode para linguagem de máquina)



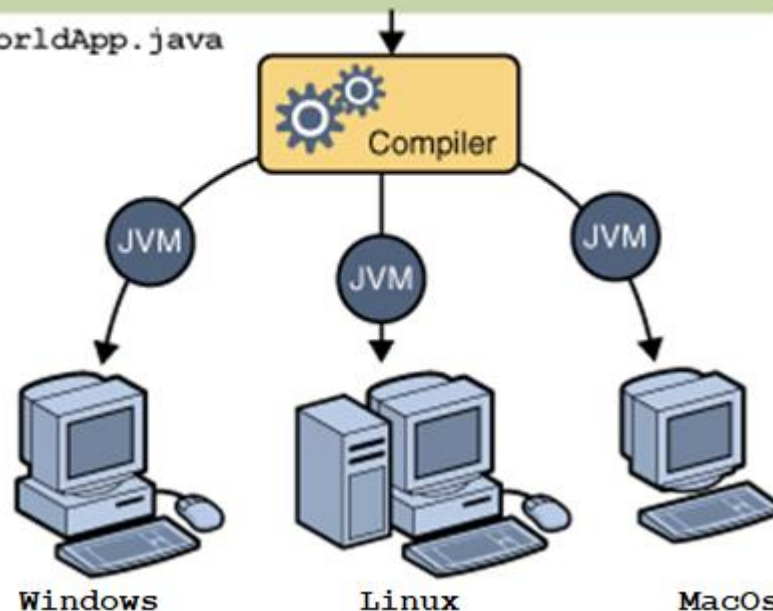
Múltiplas plataformas



Java Program

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

HelloWorldApp.java





Plataforma JAVA

- Ambiente de desenvolvimento – JDK (Java Development Kit)
 - Conjunto de utilitários para desenvolvimento de software para plataforma Java
 - Inclui: compilador (código fonte Java para bytecode) e o ambiente de execução JRE
 - Não inclui: ferramentas integradas de desenvolvimento (IDE)

Ferramentas Integradas



- IDE – Integrated Development Environment
 - Editor
 - Compilador
 - Execução
 - Debug

Mais simples e leves:



Mais complexos e pesados:





Tecnologias JAVA

- Java SE – Java Standard Edition: básico
- Java EE – Java Enterprise Edition: aplicações corporativas e Internet (inclui JSE)
- Java ME – Java Micro Edition: dispositivos eletrônicos compactos e móveis;
- JavaFX: para aplicações interativas;
- Dentre outras...

Instalação do JAVA



- Download do site da Oracle - Java SE 21 e 17 (LTS):

<https://www.oracle.com/java/technologies/javase-downloads.html>

Java 21 and Java 17 available now

JDK 21 is the latest long-term support release of Java SE Platform.

[Learn about Java SE Subscription](#)

[JDK 21](#) [JDK 17](#) [GraalVM for JDK 21](#) [GraalVM for JDK 17](#)

JDK Development Kit 21.0.2 downloads

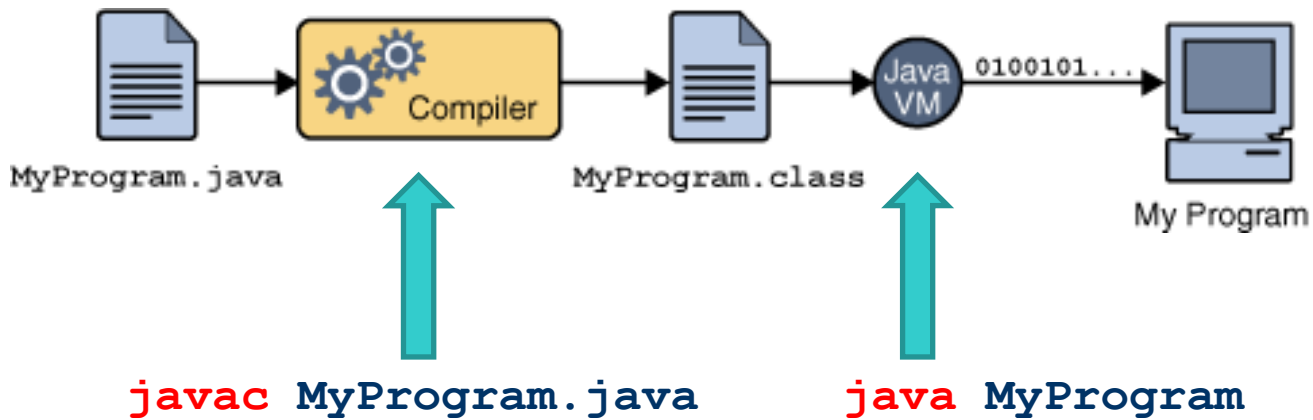
JDK 21 binaries are free to use in production and free to redistribute, at no cost, under the [Oracle No-Fee Terms and Conditions](#) (NFTC).

JDK 21 will receive updates under the NFTC, until September 2026, a year after the release of the next LTS. Subsequent JDK 21 updates will be licensed under the [Oracle License](#) (OTN) and production use beyond the [limited free grants](#) of the OTN license will [require a fee](#).

[Linux](#) [macOS](#) [Windows](#)

Product/file description	File size	Download
ARM64 Compressed Archive	186.57 MB	https://download.oracle.com/java/21/latest/jdk-21_linux-aarch64_bin.t

Desenvolvimento em JAVA



Programa em Java

- Todo programa Java é uma classe:
 - Possui obrigatoriamente um método principal;
 - Método principal contém a sequência lógica do programa;
 - Pode ter outros métodos (funções);
 - Utiliza outras classes e objetos em suas instruções;
- Nem toda classe é um programa;

Estrutura de um Programa Java

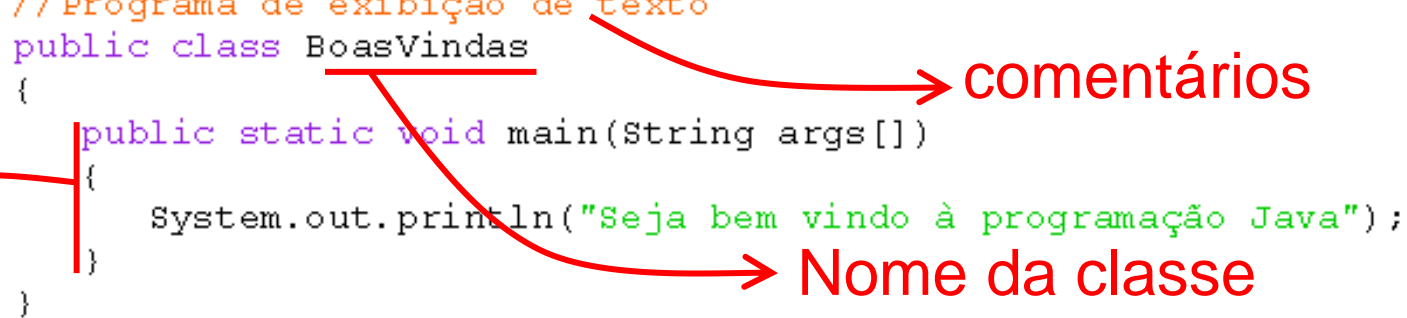
```
public class Nome_da_Classe
{
    public static void main (String args[ ])
    {
        <bloco de instruções do programa>;
    }
}
```

Salvar como **Nome_da_Classe.java**

Exemplo de edição de programas

Programa BoasVindas.java

```
//Programa de exibição de texto
public class BoasVindas
{
    public static void main(String args[])
    {
        System.out.println("Seja bem vindo à programação Java");
    }
}
```



→ Método principal da classe

Primeiro programa JAVA

Para compilar (NA LINHA DE COMANDO) :

<Drive e Diretório>\javac BoasVindas.java

Para Interpretar (NA LINHA DE COMANDO) :

<Drive e Diretório>\java BoasVindas

Exemplo com Bloco de Notas

```
BoasVindas.java - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
public class BoasVindas
{
    public static void main (String args[ ]){
        System.out.println ("seja bem-vindo a programação Java");
    }
}
```

```
C:\Windows\system32\CMD.exe

C:\Users\Grace>javac BoasVindas.java → Compilando código fonte
C:\Users\Grace>java BoasVindas → Interpretando ByteCode
Seja bem-vindo a programação Java

C:\Users\Grace>dir BoasVindas.*
O volume na unidade C não tem nome.
O Número de Série do Volume é D6C6-E263

Pasta de C:\Users\Grace
23/02/2015  10:55          449 BoasVindas.class
23/02/2015  10:52          140 BoasVindas.java
                2 arquivo(s)          589 bytes
                0 pasta(s) 229.414.137.856 bytes disponíveis

C:\Users\Grace>
```

Usando jGrasp

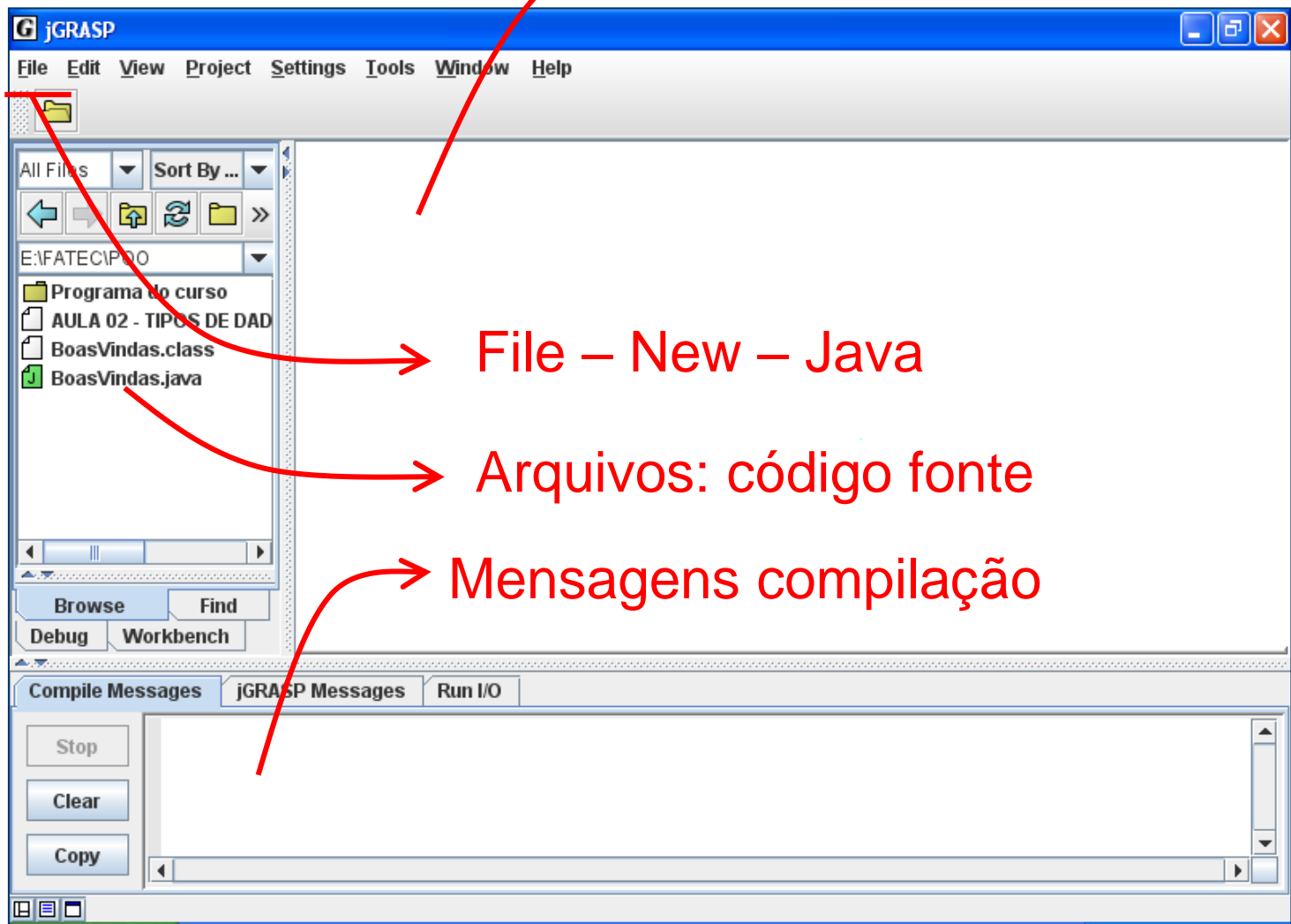
- Ambiente integrado de desenvolvimento
- Diversas linguagens: Java, C++, C, etc.
- www.jgrasp.org

The logo for jGRASP, featuring the text 'jGRASP' in a large, bold, black serif font. The 'j' is lowercase and has a dot, while 'GRASP' is in all caps. The text is set against a white background with a subtle drop shadow effect.

Copyright 1999-2022 Auburn University
Samuel Ginn College of Engineering
Computer Science and Software Engineering
Created by James H Cross II & Larry Barowski

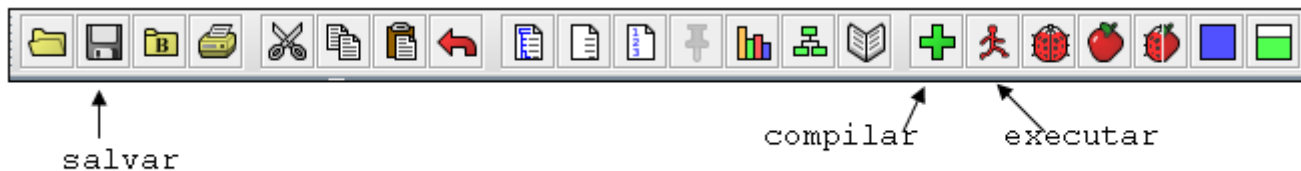
v 2.0.6_09

Tela principal



Edição de programas

Botões mais usados



Compile sempre antes de executar!

Programa Java

```
//Programa de exibição de texto
public class BoasVindas
{
    public static void main(String args[])
    {
        System.out.println("Seja bem vindo à programação Java");
    }
}
```

Linhas de comentário

- Usadas para documentar o programa
- Ignoradas pelo compilador

// comenta uma linha

/* múltiplas linhas */

```
/* Programa que imprime  
uma mensagem de  
boas vindas          */  
public class BoasVindas  
{  
    //Método principal que inicia a execução do aplicativo Java  
    public static void main(String args[])  
    {  
        System.out.println("Seja bem vindo à programação Java");  
    } //fim do método  
} //fim da classe
```

Identificadores (classes e variáveis)

- Nome de classes
 - Por padrão, iniciar com maiúscula
 - Quando nome composto, iniciais maiúsculas
- Variáveis:
 - Iniciar com minúscula
- Regras gerais:
 - Caracteres permitidos: letras; dígitos; sublinhado; cifrão;
 - Não iniciar com dígitos;
 - Não conter espaço em branco;
- Ex.: BoasVindas1; \$value, _valor, cod_ent, botao7

SAÍDA PADRÃO: `System.out`

- Exibe caracteres/ dados na tela
- Formatos:
 - `System.out.print(args);` // cursor na mesma linha
 - `System.out.println(args);`// cursor para próx. linha



string entre aspas ou variáveis

Sequência de escape: caracteres especiais

Sequência de escape	Descrição
<code>\n</code>	Nova linha.
<code>\t</code>	Tabulação.
<code>\r</code>	Posiciona o curso no início da linha atual.
<code>\\</code>	Imprime uma barra invertida.
<code>\"</code>	Imprime aspas.

```
System.out.println("Seja bem vindo\nà programação Java");
```


**Seja bem vindo
à programação Java**

Experimente alterar no seu programa

Método System.out.printf

- Exibe dados formatados na tela

```
// método principal da classe
public static void main (String args[])
{
    System.out.printf ("Bem vindo\n a programacao %s!\n", "Java");
} // fim do método principal
```



Experimente agora com o método *printf*

Entrada de dados – Classe Scanner

- Pertence ao pacote **java.util**
- Cláusula **import**: aparecer antes da definição da classe
- Se esquecermos, causa erro!

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

Lembrando

Classe



Abstração

Molde



Objeto



Concreto

Ocorrência real

Instância

Classe ***Scanner***

X objeto ***input***
(variável do tipo Scanner)

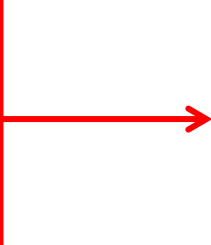
Exemplo

```
// declaração da variável input  
Scanner input ;
```

```
// instanciando objeto: alocado espaço na memória do computador  
input = new Scanner (System.in);
```

```
// chamando métodos a partir do objeto input
```

```
num1 = input.nextInt( );  
num2 = input.nextFloat( );  
num3 = input.nextDouble( );  
nome = input.nextLine( );
```



Métodos públicos
da classe Scanner
chamados a partir
do objeto *input*

Entrada de dados – Classe Scanner

```
// Adicao.java
// Programa de adiç o que exibe a soma de dois n meros.
import java.util.Scanner;
```

```
public class Adicao
{
```

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

```
        Scanner input = new Scanner( System.in );
        int num1;
        int num2;
        int soma;
```

```
        System.out.print( "Digite o primeiro n mero: " );
        num1 = input.nextInt(); //L  o primeiro n mero
```

```
        System.out.print( "Digite o segundo n mero: " );
        num2 = input.nextInt(); //L  o segundo n mero
```

```
        soma = num1 + num2;
```

```
        System.out.printf( "A soma   %d\n", soma );
```

```
    }
```

```
}
```

vari vel do tipo Scanner => objeto

vari veis de tipo primitivo

Codifique, salve, **compile** e execute seu 2  programa!

Tipos primitivos do Java

Tipos	Primitivo	Valores possíveis		Valor Padrão	Tamanho
		Menor	Maior		
Inteiro	byte	-128	127	0	8 bits
	short	-32768	32767	0	16 bits
	int	-2.147.483.648	2.147.483.647	0	32 bits
	long	-9.223.372.036.854.770.000	9.223.372.036.854.770.000	0	64 bits
Ponto Flutuante	float	-1,4024E-37	3.40282347E + 38	0	32 bits
	double	-4,94E-307	1.79769313486231570E + 308	0	64 bits
Caractere	char	0	65535	\0	16 bits
Booleano	boolean	false	true	false	1 bit

Operadores e precedência

OPERAÇÃO JAVA	OPERADOR
adição	+
subtração	-
multiplicação	*
divisão	/
resto	%

OPERADORES
*, /, %
+, -

Entrada de valores reais - double

```
// Adicao.java
// Programa de adiçaõ que exibe a soma de dois números.
import java.util.Scanner;
public class Adicao2
{
    public static void main( String args[] )
    {
        Scanner input = new Scanner( System.in );
        double num1;
        double num2;
        double soma;

        System.out.print( "Digite o primeiro número: " );
        num1 = input.nextDouble(); //Lê o primeiro número

        System.out.print( "Digite o segundo número: " );
        num2 = input.nextDouble(); //Lê o segundo número

        soma = num1 + num2;

        System.out.printf( "A soma é %f\n", soma );
    }
}
```

Codifique seu 3º programa!

Formatação de reais

```
▶▶ Digite o primeiro número: 4,5
▶▶ Digite o segundo número: 5,5
A soma é 10,000000
```

```
System.out.printf( "A soma é %5.2f\n", soma );
```

total de casas
(parte inteira + vírgula + casas decimais)

número de casas decimais

```
▶▶ Digite o primeiro número: 5,5
▶▶ Digite o segundo número: 6
A soma é 11,50
```

Entrada de cadeia de caracteres - String

```
// LeituraString.java
// Programa que lê e exibe cadeia de caracteres.
import java.util.Scanner;
```

```
public class LeituraString
```

```
{
```

```
    public static void main( String args[] )
```

```
    {
```

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

```
        String frase;
```

```
        System.out.print( "Digite uma frase: " );
```

```
        frase = input.nextLine();
```

```
        System.out.printf( "Frase lida: %s\n", frase );
```

```
    }
```

```
}
```

→ Não é um tipo primitivo

Entrada de dados de tipos distintos

```
1 import java.util.Scanner;
2 public class TiposDistintos
3 {
4     public static void main( String args[] )
5     {
6         Scanner ent_Str = new Scanner( System.in );
7         Scanner ent_Int = new Scanner( System.in );
8         Scanner ent_Doub = new Scanner( System.in );
9
10        String nome;
11        int idade;
12        double salario;
13
14        System.out.print( "Digite o nome :" );
15        nome = ent_Str.nextLine();
16
17        System.out.print( "Digite a idade:" );
18        idade = ent_Int.nextInt();
19
20        System.out.print( "Digite o salário:" );
21        salario = ent_Doub.nextDouble();
22
23        System.out.printf( "Dados lidos:\n" );
24        System.out.printf( "Nome : %s\n", nome );
25        System.out.printf( "Idade : %d\n", idade );
26        System.out.printf( "Salário: R$ %7.2f\n", salario );
27    }
28 }
```

Objetos distintos
Para tipos diferentes

Atividade 1 – Temperatura

- Faça um programa em Java que leia uma temperatura em Fahrenheit, calcule e exiba seu valor em Celsius.

Fórmula: $C = (F - 32) * 5 / 9$.

Atividade 2 – Ambiente Java

- Escolha apenas uma das opções abaixo.
- Para desenvolver aplicações Java, o que preciso instalar na minha máquina **obrigatoriamente**:
 - () IDE
 - () JVM
 - () JDK
 - () JRE
 - () API