

The slide features a light gray background with abstract green geometric shapes. On the left, a solid green triangle points downwards. On the right, a complex arrangement of overlapping translucent green triangles in various shades of green creates a dynamic, layered effect.

# Entrada e saída básicas em Java

Prof.: Nelson Bellincanta Filho

# Entrada não formatada

- ▶ A entrada direta de dados se faz por meio do teclado associado ao console:

```
int c = System.in.read(); //lê um caractere como inteiro
```

- ▶ O caractere é lido por uma chamada a uma função do Java composta de três partes:
  - ▶ System - Classe que disponibiliza um objeto in para qualquer aplicação;
  - ▶ in - Objeto que representa o dispositivo de entrada de dados do console (teclado);
  - ▶ read - Método que lê caracteres individuais da entrada.

```
1  /*
2  Classe para entrada de dados não formatada: EntradaNaoFormatada.java
3  IFPR – Campus Cascavel
4  Disciplina: Programação Orientada à Objetos
5  Professor: Nelson Bellincanta
6  Data da criação: 21/03/2023
7  */
8
9  package P00.Aula04;
10
11  import java.io.IOException; // Importação da classe IOException para tratar exceções
12
13  public class EntradaNaoFormatada {
14      //Cláusula throws permite que um erro seja descartado, mesmo que ele ocorra.
15      public static void main (String args[]) throws IOException {
16          System.out.println("Entrada Não Formatada!"); // Exibe mensagem inicial
17          System.out.print("Pressione um caractere: "); // // Exibe mensagem de orientação
18          int i = System.in.read(); //Lê um caractere como inteiro
19          System.out.println("ASCII = " + i); // Exibe valor lido
20      }
21  }
```

# Saída não formatada

- ▶ A saída de dados para o console exibe (ou imprime) informações em modo texto, como segue:

```
System.out.println("Oi!"); // exibe a mensagem Oi!
```

- ▶ Assim como na entrada não formatada, essa chamada é composta de três partes:
  - ▶ System - Classe que disponibiliza um objeto out para qualquer aplicação;
  - ▶ out - Objeto que representa o dispositivo de saída do console (sua janela);
  - ▶ println - Método que imprime valores, mensagens e objetos.

# Saída não formatada

- ▶ Permite exibir
  - ▶ Mensagens de texto;
  - ▶ Valores literais;
  - ▶ Conteúdo de variáveis;
  - ▶ Expressões;
  - ▶ Objetos;
  - ▶ Ou qualquer combinação desses elementos que resulte numa cadeia de caracteres, finalizando a linha em que ocorre a exibição.

```
1  /*
2  Classe para saída de dados não formatada: SaidaNaoFormatada.java
3  IFPR – Campus Cascavel
4  Disciplina: Programação Orientada à Objetos
5  Professor: Nelson Bellincanta
6  Data da criação: 21/03/2023
7  */
8
9
10 public class SaidaNaoFormatada{
11     public static void main(String args[]){
12         int numero = 1234; //Declaração e inicialização da variável do tipo inteiro
13
14         System.out.println("Saída Não Formatada!"); // Exibe mensagem inicial
15         System.out.println(1234); // Exibe o valor inteiro
16         System.out.println(2.36); // Exibe o valor real
17         System.out.println("Palavra"); // Exibe a String
18         System.out.println("Uma frase com algumas palavras."); // Exibe a String
19         System.out.println(numero); // Exibe o valor da variável
20         System.out.println(numero*10 + 333); // Exibe o resultado de uma expressão
21         //Exibe a concatenação de texto e o valor da variável
22         System.out.println("O valor: " + numero
23             + " está armazenado na variável numero.");
24     }
25 }
```

# Entrada formatada

- ▶ A partir da versão 5 do Java, foi incluída a classe `Scanner` como uma alternativa para entrada formatada de dados.

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

```
1  /*
2  Classe exemplo do uso da classe Scanner: ClassScanner.java
3  IFPR – Campus Cascavel
4  Disciplina: Programação Orientada à Objetos
5  Professor: Nelson Bellincanta
6  Data da criação: 21/03/2023
7  */
8
9
10 import java.util.Scanner; // Importação da classe Scanner do pacote java.util
11
12 public class ClassScanner {
13     public static void main (String args[]) {
14
15         Scanner s = new Scanner(System.in); // Prepara entrada de dados
16
17         System.out.println("Entrada formatada – Classe Scanner"); // Exibe mensagem inicial
18         System.out.print("Digite um valor inteiro: "); // Mensagem de orientação
19         int i = s.nextInt(); // Declaração e inicialização da variável
20         System.out.println("O valor do inteiro digitado é = " + i); // Exibe valor lido
21
22         s.close(); // Fecha objeto leitor
23     }
24 }
```



```
1  /*
2  Classe exemplo do uso da classe Scanner: ClassScanner2.java
3  IFPR – Campus Cascavel
4  Disciplina: Programação Orientada à Objetos
5  Professor: Nelson Bellincanta
6  Data da criação: 21/03/2023
7  */
8
9  import java.util.Scanner; // Importação da classe Scanner do pacote java.util
10
11  public class ClassScanner2 {
12      public static void main (String args[]) {
13
14          Scanner s = new Scanner(System.in); // Prepara entrada de dados
15
16          System.out.println("Entrada formatada – Classe Scanner"); // Mensagem inicial
17
18          System.out.print("Digite um valor inteiro: "); // Mensagem de orientação
19          int i = s.nextInt(); // Declaração e inicialização da variável
20          System.out.println("O valor do inteiro digitado é = " + i); // Exibe o valor lido
21
22          System.out.print("Digite a uma palavra: "); // Mensagem de orientação
23          String palavra = s.next(); // Declaração e inicialização da variável
24          System.out.println("A palavra digitada é = " + palavra); // Exibe valor lido
25
26          System.out.print("Digite uma frase: "); // Mensagem de orientação
27          String frase = s.nextLine(); // Declaração e inicialização da variável
28          System.out.println("A frase digitada é = " + frase); // Exibe valor lido
29
30
31          System.out.print("Digite um valor real: "); // Mensagem de orientação
32          double d = s.nextDouble(); // Declaração e inicialização da variável
33          System.out.println("O Valor do tipo real digitado é = " + d); // Exibe valor lido
34
35          s.close(); // Fecha objeto leitor
36      }
37  }
```

```
1  import java.util.Scanner; // importação da classe Scanner do pacote java.util
2
3  public class ClassScanner3 {
4      public static void main (String args[]) {
5
6          Scanner s = new Scanner(System.in); //Criação do objeto para entrada de dados
7
8          System.out.println("Entrada formatada - Classe Scanner"); // Mensagem inicial
9
10         System.out.print("Digite o 1º valor inteiro: "); // Mensagem de orientação
11         int i = s.nextInt(); // Declaração e inicialização da variável
12         s.nextLine(); // Limpeza da entrada após leitura de números
13         System.out.println("O valor do 1º inteiro digitado é = " + i); // Exibe o valor lido
14
15         System.out.print("Digite uma palavra: "); // Mensagem de orientação
16         String palavra = s.nextLine(); // Declaração e inicialização da variável
17         System.out.println("A palavra digitada é = " + palavra); // Exibe o valor lido
18
19         System.out.print("Digite um inteiro: "); // Mensagem de orientação
20         int i2 = s.nextInt(); // Declaração e inicialização da variável
21         s.nextLine(); // Limpeza da entrada após leitura de números
22         System.out.println("Valor = " + i2); // Exibe o valor lido
23
24         System.out.print("Digite uma frase: "); // Mensagem de orientação
25         String frase = s.nextLine(); // Declaração e inicialização da variável
26         System.out.println("A frase digitada é = " + frase); // Exibe o valor lido
27
28         System.out.print("Digite um real: "); // Mensagem de orientação
29         double d = s.nextDouble(); // Declaração e inicialização da variável
30         s.nextLine(); // Limpeza da entrada após leitura de números
31         System.out.println("Valor digitado = " + d); // Exibe o valor lido
32
33         System.out.print("Digite uma 2ª palavra: "); // Mensagem de orientação
34         String palavra2 = s.nextLine(); // Declaração e inicialização da variável
35         System.out.println("A 2ª palavra digitada é = " + palavra2); // Exibe o valor lido
36
37         s.close(); // Fecha objeto da classe Scanner
38     }
39 }
```

# Saída formatada

- ▶ A partir da versão 5 do Java é oferecida uma forma conveniente para realização da saída formatada de dados com o método `printf()` que pode ser utilizado por meio do objeto `out` disponível na classe `System`;
- ▶ Sua sintaxe é:

```
System.out.printf(<formatação> [, expr1 [, expr2 [ ...]]] );
```

# Saída formatada

- ▶ Marcadores comuns são :
  - ▶ %d para inteiros de qualquer tipo;
  - ▶ %f para reais de qualquer tipo;
  - ▶ %c para caractere; e
  - ▶ %s para o tipo String.

```
1  /*
2  Classe exemplo saída formatada com printf(): SaidaFormatada.java
3  IFPR – Campus Cascavel
4  Disciplina: Programação Orientada à Objetos
5  Professor: Nelson Bellincanta
6  Data da criação: 21/03/2023
7  */
8
9  import java.util.Scanner; // Importação da classe Scanner do pacote java.util
10
11  public class SaidaFormatada {
12      public static void main(String[] args) {
13          Scanner sc = new Scanner(System.in); // Criação do objeto para entrada de dados
14
15          System.out.println("Saída Formatada – printf()"); // Mensagem inicial
16
17          System.out.print("Informe o nome da pessoa: "); // Mensagem de orientação
18          String nome = sc.nextLine(); // Declaração e inicialização da variável
19          System.out.print("Informe a idade da pessoa: "); // Mensagem de orientação
20          int idade = sc.nextInt(); // Declaração e inicialização da variável
21          System.out.print("Informe a altura da pessoa: "); // Mensagem de orientação
22          double altura = sc.nextDouble(); // Declaração e inicialização da variável
23
24          System.out.printf("%s tem %d anos e %fm de altura", nome, idade, altura); // Saída formatada com printf()
25          sc.close(); // Fecha objeto da classe Scanner
26      }
27  }
```

```
1  /*
2  Classe exemplo para saída formatada com printf(): SaidaFormatada2.java
3  IFPR – Campus Cascavel
4  Disciplina: Programação Orientada à Objetos
5  Professor: Nelson Bellincanta
6  Data da criação: 21/03/2023
7  */
8
9  import java.util.Scanner; // Importação da classe Scanner do pacote java.util
10
11 public class SaidaFormatada2 {
12     public static void main(String[] args) {
13         Scanner sc = new Scanner(System.in); // Criação do objeto para entrada de dados
14
15         System.out.println("Saída Formatada – printf()"); // Exibe mensagem inicial
16
17         System.out.print("Informe o nome da pessoa: "); // Exibe mensagem de orientação
18         String nome = sc.nextLine(); // Declaração e inicialização da variável
19         System.out.print("Informe a idade da pessoa: "); // Exibe mensagem de orientação
20         int idade = sc.nextInt(); // Declaração e inicialização da variável
21         System.out.print("Informe a altura da pessoa: "); // Exibe mensagem de orientação
22         double altura = sc.nextDouble(); // Declaração e inicialização da variável
23
24         System.out.printf("Nome: %s\nIdade: %d\nAltura: %.2fm\n", nome, idade, altura); // Saída formatada com printf()
25         sc.close(); // Fecha objeto da classe Scanner
26     }
27 }
```

# Entrada com JOptionPane

É uma classe que possibilita a criação de uma caixa de dialogo padrão que ou solicita um valor para o usuário ou retorna uma informação.



```
1  /*
2  Classe exemplo entrada com JOptionPane: EntradaComJOptionPane.java
3  IFPR – Campus Cascavel
4  Disciplina: Programação Orientada à Objetos
5  Professor: Nelson Bellincanta
6  Data da criação: 21/03/2023
7  */
8
9  import javax.swing.JOptionPane;
10
11  public class EntradaComJOptionPane {
12      public static void main(String args[]) {
13          String aux;
14          float largura, comprimento, area, perimetro;
15          try {
16              aux = JOptionPane.showInputDialog("Entre com o comprimento");
17              comprimento = Float.parseFloat(aux);
18
19              aux = JOptionPane.showInputDialog("Entre com a largura");
20              largura = Float.parseFloat(aux);
21
22              area = comprimento * largura;
23              perimetro = comprimento * 2 + largura * 2;
24              JOptionPane.showMessageDialog(null, "Área : " + area + ", Perímetro : " + perimetro);
25          } catch (NumberFormatException erro) {
26              JOptionPane.showMessageDialog(null, "Houve erro na conversão, digite apenas caracteres numéricos"
27                  + erro.toString());
28          }
29          System.exit(0);
30      }
31  }
```



# Referências

- ▶ DEITEL, Harvey M.; DEITEL, Paul J. **Java: como programar**. 10ª ed. Editora Pearson Education do Brasil, 2017.
- ▶ FURGERI, Sérgio. **Java 8: Ensino Didático**. 1ª ed., São Paulo: Érica, 2015.
- ▶ JUNIOR, Peter Jandl. **Java Guia do Programador**. 4ª Edição: Atualizado para Java 16. Novatec Editora, 2021.