

	<p>FACULDADE DE TECNOLOGIA- FATEC</p> <p>Curso: Desenvolvimento de Software Multiplataforma</p> <p>Disciplina: Técnicas de Programação II</p>
---	---

Aluno: Kevin de Almeida Brandão	RA: 1091392323045	Data: 09/03/2024
--	--------------------------	-------------------------

EXERCÍCIO 02: revisão arrays e métodos

1. Construa um programa que armazene o código, a quantidade, o valor de compra e o valor de venda de 30 produtos. A listagem pode ser de todos os produtos ou somente de um ao se digitar o código.

```

10 public class Main1 {
    Run | Debug
11 public static void main(String... strings) {
12     Scanner scanner = new Scanner(System.in);
13
14     // Criando um array para armazenar os produtos
15     Produto[] produtos = new Produto[30];
16
17     // Inicializando os produtos
18     for (int i = 0; i < produtos.length; i++) {
19         produtos[i] = new Produto();
20         produtos[i].codigo = i + 1; // Considerando que os códigos começam em 1
21         System.out.println("Digite a quantidade do produto " + produtos[i].codigo + ":");
22         produtos[i].quantidade = scanner.nextInt();
23         System.out.println("Digite o valor de compra do produto " + produtos[i].codigo + ":");
24         produtos[i].valorCompra = scanner.nextDouble();
25         System.out.println("Digite o valor de venda do produto " + produtos[i].codigo + ":");
26         produtos[i].valorVenda = scanner.nextDouble();
27     }
28
29     // Menu para listar produtos
30     System.out.println(x:"Escolha uma opção:");
31     System.out.println(x:"1 - Listar todos os produtos");
32     System.out.println(x:"2 - Listar um produto específico");
33     int opcao = scanner.nextInt();
34
35     switch (opcao) {
36         case 1:
37             // Listar todos os produtos
38             System.out.println(x:"Listagem de todos os produtos:");
39             for (Produto produto : produtos) {
40                 System.out.println("Código: " + produto.codigo);
41                 System.out.println("Quantidade: " + produto.quantidade);
42                 System.out.println("Valor de Compra: " + produto.valorCompra);
43                 System.out.println("Valor de Venda: " + produto.valorVenda);
44                 System.out.println(x:"-----");
45             }
46             break;
47         case 2:
48             // Listar um produto específico
49             System.out.println(x:"Digite o código do produto que deseja listar:");
50             int codigoDesejado = scanner.nextInt();
51             for (Produto produto : produtos) {
52                 if (produto.codigo == codigoDesejado) {
53                     System.out.println("Código: " + produto.codigo);
54                     System.out.println("Quantidade: " + produto.quantidade);
55                     System.out.println("Valor de Compra: " + produto.valorCompra);
56                     System.out.println("Valor de Venda: " + produto.valorVenda);
57                     break;
58                 }
59             }
60             break;
61         default:
62             System.out.println(x:"Opção inválida.");
63             break;

```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

Microsoft Windows [versão 10.0.22621.3155]
(c) Microsoft Corporation. Todos os direitos reservados.

```

C:\Users\Kevin Brandao\Desktop\Aulas_java> cmd /C ""C:\Users\Kevin Brandao\AppData\Roaming\Code\User
\globalStorage\pleiades.java-extension-pack-jdk\java\21\bin\java.exe" --enable-preview -XX:+ShowCode
DetailsInExceptionMessages -cp "C:\Users\Kevin Brandao\AppData\Roaming\Code\User\workspaceStorage\e8
e5515bcbaa0d37db79ccbcc981138b\redhat.java\jdt_ws\Aulas_java_ea406e02\bin" Main1 "
Digite a quantidade do produto 1:
23
Digite o valor de compra do produto 1:
12
Digite o valor de venda do produto 1:

```

+ v ...

JavaS
Run:

2. Faça um programa que leia dois conjuntos de números inteiros, tendo cada um 10 elementos. Ao final o programa deve listar os elementos comuns aos conjuntos.

```
public class Main2 {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
  
        // Definindo os conjuntos de números  
        int[] conjunto1 = new int[10];  
        int[] conjunto2 = new int[10];  
  
        // Lendo os elementos do primeiro conjunto  
        System.out.println("Digite os elementos do primeiro conjunto:");  
        for (int i = 0; i < conjunto1.length; i++) {  
            conjunto1[i] = scanner.nextInt();  
        }  
  
        // Lendo os elementos do segundo conjunto  
        System.out.println("Digite os elementos do segundo conjunto:");  
        for (int i = 0; i < conjunto2.length; i++) {  
            conjunto2[i] = scanner.nextInt();  
        }  
  
        // Encontrando elementos comuns usando um HashSet  
        Set<Integer> elementosComuns = new HashSet<>();  
        for (int num1 : conjunto1) {  
            for (int num2 : conjunto2) {  
                if (num1 == num2) {  
                    elementosComuns.add(num1);  
                    break; // Não é necessário continuar a busca se o elemento foi encontrado  
                }  
            }  
        }  
  
        // Exibindo os elementos comuns  
        if (elementosComuns.isEmpty()) {  
            System.out.println("Não há elementos comuns nos conjuntos.");  
        } else {  
            System.out.println("Elementos comuns aos conjuntos:");  
            for (int elemento : elementosComuns) {  
                System.out.println(elemento);  
            }  
        }  
  
        scanner.close();  
    }  
}
```

3. Construa um programa que leia dados para um vetor de 100 elementos inteiros. Imprimir o maior e o menor, sem ordenar, o percentual de números pares e a média dos elementos do vetor.

```

public class Main3 {
    Run | Debug
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        // Criando um vetor de 100 elementos inteiros
        int[] vetor = new int[100];

        // Lendo os elementos do vetor
        System.out.println("Digite os 100 elementos do vetor:");
        for (int i = 0; i < vetor.length; i++) {
            vetor[i] = scanner.nextInt();
        }

        // Encontrando o maior e o menor elemento
        int maior = vetor[0];
        int menor = vetor[0];
        for (int i = 1; i < vetor.length; i++) {
            if (vetor[i] > maior) {
                maior = vetor[i];
            }
            if (vetor[i] < menor) {
                menor = vetor[i];
            }
        }

        // Calculando o percentual de números pares
        int numerosPares = 0;
        for (int num : vetor) {
            if (num % 2 == 0) {
                numerosPares++;
            }
        }
        double percentualPares = (double) numerosPares / vetor.length * 100;

        // Calculando a média dos elementos
        int soma = 0;
        for (int num : vetor) {
            soma += num;
        }
        double media = (double) soma / vetor.length;

        // Imprimindo os resultados
        System.out.println("Maior elemento: " + maior);
        System.out.println("Menor elemento: " + menor);
        System.out.println("Percentual de números pares: " + percentualPares + "%");
        System.out.println("Média dos elementos: " + media);

        scanner.close();
    }
}

```

4. Faça um programa que leia 10 números inteiros, armazene-os em um vetor, solicite um valor de referência inteiro e:

- a. imprima os números do vetor que são maiores que o valor referência**
- b. retorne quantos números armazenados no vetor são menores que o valor de referência**
- c. retorne quantas vezes o valor de referência aparece no vetor**

```

3  public class Main4 {
4      public static void main(String... strings) {
5          Scanner scanner = new Scanner(System.in);
6
7          // Criando um vetor de 10 elementos inteiros
8          int[] vetor = new int[10];
9
10         // Lendo os elementos do vetor
11         System.out.println(x: "Digite 10 números inteiros:");
12         for (int i = 0; i < vetor.length; i++) {
13             vetor[i] = scanner.nextInt();
14         }
15
16         // Solicitando o valor de referência
17         System.out.println(x: "Digite um valor de referência inteiro:");
18         int valorReferencia = scanner.nextInt();
19
20         // a. Imprimir os números do vetor maiores que o valor de referência
21         System.out.println(x: "Números maiores que o valor de referência:");
22         for (int num : vetor) {
23             if (num > valorReferencia) {
24                 System.out.println(num);
25             }
26         }
27
28         // b. Contar quantos números do vetor são menores que o valor de referência
29         int menoresQueReferencia = 0;
30         for (int num : vetor) {
31             if (num < valorReferencia) {
32                 menoresQueReferencia++;
33             }
34         }
35         System.out.println("Quantidade de números menores que o valor de referência: " +
36
37         // c. Contar quantas vezes o valor de referência aparece no vetor
38         int vezesReferencia = 0;
39         for (int num : vetor) {
40             if (num == valorReferencia) {
41                 vezesReferencia++;
42             }
43         }
44         System.out.println("Quantidade de vezes que o valor de referência aparece no veto
45
46         scanner.close();
47     }
48 }
49

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Microsoft Windows [versão 10.0.22621.3155]
(c) Microsoft Corporation. Todos os direitos reservados.

C:\Users\Kevin Brandao\Desktop\Aulas_java> cmd /C ""C:\Users\Kevin Brandao\AppData\Roaming\Code\User\globalStorage\pleiades.java-extension-pack-jdk\java\21\bin\java.exe" --enable-preview -XX:+ShowCodeDetailsInExceptionMessages -cp "C:\Users\Kevin Brandao\AppData\Roaming\Code\User\workspaceStorage\e8e5515bcb0d37db79ccbc981138b\redhat.java\jdt_ws\Aulas_java_ea406e02\bin" Main4 "

Digite 10 números inteiros:

1
2
3
45
5
6
4
3
4
56

Digite um valor de referência inteiro:

6
45
56

Números maiores que o valor de referência:

5. Ler dois conjuntos de números reais, armazenando-os em vetores e calcular o produto escalar entre eles. Os conjuntos têm 5 elementos cada. Imprimir os dois conjuntos e o produto escalar, sendo que o produto escalar é dado por: $x_1 * y_1 + x_2 * y_2 + \dots + x_n * y_n$.

```

3 public class Main5 {
    Run | Debug
4 public static void main(String... strings) {
5     Scanner scanner = new Scanner(System.in);
6
7     // Definindo os vetores de números reais com 5 elementos cada
8     double[] vetorX = new double[5];
9     double[] vetorY = new double[5];
10
11     // Lendo os elementos do primeiro vetor
12     System.out.println("Digite os 5 elementos do primeiro vetor:");
13     for (int i = 0; i < vetorX.length; i++) {
14         vetorX[i] = scanner.nextDouble();
15     }
16
17     // Lendo os elementos do segundo vetor
18     System.out.println("Digite os 5 elementos do segundo vetor:");
19     for (int i = 0; i < vetorY.length; i++) {
20         vetorY[i] = scanner.nextDouble();
21     }
22
23     // Calculando o produto escalar
24     double produtoEscalar = calcularProdutoEscalar(vetorX, vetorY);
25
26     // Imprimindo os vetores e o produto escalar
27     System.out.println("Primeiro vetor:");
28     imprimirVetor(vetorX);
29     System.out.println("Segundo vetor:");
30     imprimirVetor(vetorY);
31     System.out.println("Produto escalar: " + produtoEscalar);
32
33     scanner.close();
34 }
35
36 // Método para calcular o produto escalar entre dois vetores
37 public static double calcularProdutoEscalar(double[] vetor1, double[] vetor2) {
38     double produtoEscalar = 0.0;
39     for (int i = 0; i < vetor1.length; i++) {
40         produtoEscalar += vetor1[i] * vetor2[i];
41     }
42     return produtoEscalar;
43 }
44
45 // Método para imprimir um vetor
46 public static void imprimirVetor(double[] vetor) {
47     for (double num : vetor) {
48         System.out.print(num + " ");
49     }
50     System.out.println(); // Quebra de linha após imprimir o vetor
51 }
52 }
53

```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

Microsoft Windows [versão 10.0.22621.3155]

(c) Microsoft Corporation. Todos os direitos reservados.

```

C:\Users\Kevin Brandao\Desktop\Aulas_java> cmd /C ""C:\Users\Kevin Brandao\AppData\Roaming\Code\workspaceStorage\pleiades.java-extension-pack-jdk\java\21\bin\java.exe" --enable-preview -XX:+S
DetailsInExceptionMessages -cp "C:\Users\Kevin Brandao\AppData\Roaming\Code\User\workspaceStorage\515bcb0d37db79ccbcc981138b\redhat.java\jdt_ws\Aulas_java_ea406e02\bin" Main5 "

```

Digite os 5 elementos do primeiro vetor:

1
2
3
4
5

Digite os 5 elementos do segundo vetor:

2
3
4
5
6

Primeiro vetor:

1.0 2.0 3.0 4.0 5.0

Segundo vetor:

2.0 3.0 4.0 5.0 6.0

Produto escalar: 70.0

C:\Users\Kevin Brandao\Desktop\Aulas_java>

6. Faça um programa que calcule o desvio padrão de um vetor v contendo $n = 10$ números, onde m é a média do vetor.

```
public class Main6 {  
    Run | Debug  
    public static void main(String... strings) {  
        Scanner scanner = new Scanner(System.in);  
  
        // Definindo o vetor com 10 elementos  
        double[] vetor = new double[10];  
  
        // Lendo os elementos do vetor  
        System.out.println(x: "Digite os 10 números:");  
        for (int i = 0; i < vetor.length; i++) {  
            vetor[i] = scanner.nextDouble();  
        }  
  
        // Calculando a média dos elementos do vetor  
        double media = calcularMedia(vetor);  
  
        // Calculando o desvio padrão  
        double desvioPadrao = calcularDesvioPadrao(vetor, media);  
  
        // Imprimindo o desvio padrão  
        System.out.println("Desvio padrão: " + desvioPadrao);  
  
        scanner.close();  
    }  
  
    // Método para calcular a média de um vetor de números  
    public static double calcularMedia(double[] vetor) {  
        double soma = 0.0;  
        for (double num : vetor) {  
            soma += num;  
        }  
        return soma / vetor.length;  
    }  
  
    // Método para calcular o desvio padrão de um vetor de números  
    public static double calcularDesvioPadrao(double[] vetor, double media) {  
        double somaDiferencasQuadrado = 0.0;  
        for (double num : vetor) {  
            somaDiferencasQuadrado += Math.pow(num - media, 2);  
        }  
        double variancia = somaDiferencasQuadrado / vetor.length;  
        return Math.sqrt(variancia);  
    }  
}
```

7. Leia dois vetores de inteiros x e y, cada um com 5 elementos (assuma que o usuário informa elementos repetidos). Calcule e mostre os vetores resultantes em cada caso abaixo:

a. Soma entre x e y: soma de cada elemento de x com o elemento da mesma posição em y.

b. Produto entre x e y: multiplicação de cada elemento de x com o elemento da mesma posição em y.

c. Diferença entre x e y: todos os elementos de x que não existam em y.

d. Intersecção entre x e y: apenas os elementos que aparecem nos dois vetores. e. União entre x e y: todos os elementos de x, e todos os elementos de y que não estão em x.

```

public class Main7 {
    Run | Debug
    public static void main(String... strings) {
        int[] x = { 1, 2, 3, 4, 5 };
        int[] y = { 3, 4, 5, 6, 7 };

        // a. Soma entre x e y
        int[] soma = somaEntreVetores(x, y);
        System.out.println("Soma entre x e y: " + Arrays.toString(soma));

        // b. Produto entre x e y
        int[] produto = produtoEntreVetores(x, y);
        System.out.println("Produto entre x e y: " + Arrays.toString(produto));

        // c. Diferença entre x e y
        int[] diferenca = diferencaEntreVetores(x, y);
        System.out.println("Diferença entre x e y: " + Arrays.toString(diferenca));

        // d. Intersecção entre x e y
        int[] interseccao = interseccaoEntreVetores(x, y);
        System.out.println("Intersecção entre x e y: " + Arrays.toString(interseccao));

        // e. União entre x e y
        int[] uniao = uniaoEntreVetores(x, y);
        System.out.println("União entre x e y: " + Arrays.toString(uniao));
    }

    // Soma entre x e y
    public static int[] somaEntreVetores(int[] x, int[] y) {
        int[] resultado = new int[x.length];
        for (int i = 0; i < x.length; i++) {
            resultado[i] = x[i] + y[i];
        }
        return resultado;
    }

    // Produto entre x e y
    public static int[] produtoEntreVetores(int[] x, int[] y) {
        int[] resultado = new int[x.length];
        for (int i = 0; i < x.length; i++) {
            resultado[i] = x[i] * y[i];
        }
        return resultado;
    }

    // Diferença entre x e y
    public static int[] diferencaEntreVetores(int[] x, int[] y) {
        Set<Integer> setY = new HashSet<>();
        for (int num : y) {
            setY.add(num);
        }

        int[] diferenca = Arrays.stream(x)
            .filter(num -> !setY.contains(num))
            .toArray();
        return diferenca;
    }
}

```

8. Gere uma matriz de 10 x 10 elementos. Calcule a soma dos elementos que estão na diagonal principal.

```

1 public class Main1 {
    Run | Debug
2     public static void main(String[] args) {
3         // Definindo a matriz de 10x10
4         int[][] matriz = new int[10][10];
5
6         // Preenchendo a matriz com valores aleatórios (para este exemplo)
7         for (int i = 0; i < 10; i++) {
8             for (int j = 0; j < 10; j++) {
9                 matriz[i][j] = (int) (Math.random() * 10); // Valor aleatório de 0 a 9
10            }
11        }
12
13        // Exibindo a matriz (apenas para visualização)
14        System.out.println("Matriz:");
15        for (int i = 0; i < 10; i++) {
16            for (int j = 0; j < 10; j++) {
17                System.out.print(matriz[i][j] + " ");
18            }
19            System.out.println();
20        }
21
22        // Calculando a soma dos elementos da diagonal principal
23        int somaDiagonalPrincipal = 0;
24        for (int i = 0; i < 10; i++) {
25            somaDiagonalPrincipal += matriz[i][i];
26        }
27
28        // Exibindo a soma dos elementos da diagonal principal
29        System.out.println("Soma dos elementos da diagonal principal: " + somaDiagonalPrincipal);
30    }
31 }
32

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

C:\Users\Kevin Brandao\Desktop\Aulas_java> cmd /C ""C:\Users\Kevin Brandao\AppData\Roaming\Code\User
\globalStorage\pleiades.java-extension-pack-jdk\java\21\bin\java.exe" --enable-preview -XX:+ShowCode
DetailsInExceptionMessages -cp "C:\Users\Kevin Brandao\AppData\Roaming\Code\User\workspaceStorage\e8
e5515bcb0d37db79ccb0c981138b\redhat.java\jdt_ws\Aulas_java_ea406e02\bin" Main1 "

```

Matriz:

```

5 1 9 0 9 4 0 5 3 8
9 9 6 7 0 5 1 8 1 4
3 1 3 6 6 6 5 0 2 5
7 2 0 8 2 7 0 8 7 5
7 4 5 8 7 8 1 9 6 2
3 9 8 4 6 1 4 5 8 1
2 1 9 3 6 2 4 8 2 7
8 2 2 0 2 0 0 2 0 0
0 9 8 6 3 2 1 4 8 2
0 5 4 6 2 3 6 8 7 5

```

Soma dos elementos da diagonal principal: 52

```

C:\Users\Kevin Brandao\Desktop\Aulas_java>

```

JavaSE-2

Run: Ma

9. Faça um programa para gerar automaticamente números entre 0 e 99 de uma cartela de bingo. Sabendo que cada cartela deverá conter 5 linhas de 5 números, gere estes dados de modo a não ter números repetidos dentro das cartelas. O programa deve exibir na tela a cartela gerada. 1 de 2

```

1  -
2  1+ import java.util.ArrayList;
3  2+ import java.util.Collections;
4  3+ import java.util.List;
5  4+
6  5+ public class Main9 {
7  6+     public static void main(String[] args) {
8  7+         int[][] cartela = gerarCartela();
9  8+
10 9+         // Exibir a cartela gerada
1110+         System.out.println("Cartela de Bingo:");
1211+         for (int i = 0; i < 5; i++) {
1312+             for (int j = 0; j < 5; j++) {
1413+                 System.out.printf(format: "%2d ", cartela[i][j]);
1514+             }
1615+             System.out.println();
1716+         }
1817+     }
1918+
2019+     public static int[][] gerarCartela() {
2120+         int[][] cartela = new int[5][5];
2221+         List<Integer> numerosDisponiveis = new ArrayList<>();
2322+
2423+         // Preencher a lista com números de 0 a 99
2524+         for (int i = 0; i < 100; i++) {
2625+             numerosDisponiveis.add(i);
2726+         }
2827+
2928+         // Embaralhar os números para garantir aleatoriedade
3029+         Collections.shuffle(numerosDisponiveis);
3130+
3231+         // Preencher a cartela
3332+         for (int i = 0; i < 5; i++) {
3433+             for (int j = 0; j < 5; j++) {
3534+                 cartela[i][j] = numerosDisponiveis.remove(index:0);
3635+             }
3736+         }
3837+
3938+         return cartela;
4039+     }
4140+ }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

c:\Users\Kevin Brandao\Desktop\Aulas_java>clear
'clear' não é reconhecido como um comando interno
ou externo, um programa operável ou um arquivo em lotes.

```

```

c:\Users\Kevin Brandao\Desktop\Aulas_java> c: && cd "c:\Users\Kevin Brandao\Desktop\Aulas_java" && c
md /C ""C:\Users\Kevin Brandao\AppData\Roaming\Code\User\globalStorage\pleiades.java-extension-pack-
jdk\java\21\bin\java.exe" --enable-preview -XX:+ShowCodeDetailsInExceptionMessages -cp "C:\Users\Kev
in Brandao\AppData\Roaming\Code\User\workspaceStorage\e8e5515bcbaa0d37db79ccbcc981138b\redhat.java\j
dt_ws\Aulas_java_ea406e02\bin" Main9 "
Cartela de Bingo:
48 31 65 74 56
27 61 89 36 85
28 43 77 20 6
9 82 80 66 72
57 22 19 68 87

```

10. Faça uma função e um programa de teste para o cálculo do volume de uma esfera. Sendo que o raio é passado por parâmetro.

$$V = \frac{4}{3} * \pi * R^3$$


```

1  public class Main10 {
    Run | Debug
2      public static void main(String... strings) {
3          // Raio da esfera
4          double raio = 5.0;
5
6          // Calcular o volume da esfera
7          double volume = calcularVolumeEsfera(raio);
8
9          // Exibir o resultado
10         System.out.println("Volume da esfera com raio " + raio + ": " + volume);
11     }
12
13     public static double calcularVolumeEsfera(double raio) {
14         double pi = Math.PI;
15
16         double volume = (4.0 / 3.0) * pi * Math.pow(raio, 3);
17
18         return volume;
19     }
20 }
21

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Microsoft Windows [versão 10.0.22621.3155]
(c) Microsoft Corporation. Todos os direitos reservados.

C:\Users\Kevin Brandao\Desktop\Aulas_java> cmd /C ""C:\Users\Kevin Brandao\AppData\Roaming\Code\U
\globalStorage\pleiades.java-extension-pack-jdk\java\21\bin\java.exe" --enable-preview -XX:+ShowC
DetailsInExceptionMessages -cp "C:\Users\Kevin Brandao\AppData\Roaming\Code\User\workspaceStorage
e5515bcbaa0d37db79ccbcc981138b\redhat.java\jdt_ws\Aulas_java_ea406e02\bin" Main10 "
Volume da esfera com raio 5.0: 523.5987755982989

11. Faça uma função que receba 3 números inteiros como parâmetro, representando horas, minutos e segundos, e os converta em segundos.

```

public class Main11 {
    Run | Debug
    public static void main(String[] args) {
        // Exemplo de horas, minutos e segundos
        int horas = 42;
        int minutos = 30;
        int segundos = 45;

        // Chamada da função para converter para segundos
        int totalSegundos = converterParaSegundos(horas, minutos, segundos);

        // Exibindo o resultado
        System.out.println("Total de segundos: " + totalSegundos);
    }

    public static int converterParaSegundos(int horas, int minutos, int segundos) {
        // Calculando o total de segundos
        int totalSegundos = horas * 3600 + minutos * 60 + segundos;

        return totalSegundos;
    }
}

```

TERMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Microsoft Windows [versão 10.0.22621.3155]
 Microsoft Corporation. Todos os direitos reservados.

C:\Users\Kevin Brandao\Desktop\Aulas_java> cmd /C ""C:\Users\Kevin Brandao\AppData\Roaming\Code\User\workspaceStorage\pleiades.java-extension-pack-jdk\java\21\bin\java.exe" --enable-preview -XX:+ShowCodeDetailsInExceptionMessages -cp "C:\Users\Kevin Brandao\AppData\Roaming\Code\User\workspaceStorage\e85bcbaa0d37db79ccbcc981138b\redhat.java\jdt_ws\Aulas_java_ea406e02\bin" Main11 "
 Total de segundos: 9045

12. Faça uma função que receba por parâmetro dois valores X e Z. Calcule e retorne o resultado de X^Z para o programa principal. Atenção não utilize nenhuma função pronta de exponenciação.

```
lista_exercicios_2 > J Main12.java > Main12 > main(String...)
1 public class Main12 {
2     public static void main(String... strings) {
3         // Exemplo de valores para X e Z
4         int x = 2;
5         int z = 5;
6
7         // Chamada da função para calcular a potência
8         Long resultado = calcularPotencia(x, z);
9
10        // Exibindo o resultado
11        System.out.println("Resultado de " + x + "^" + z + " = " + resultado);
12    }
13
14    public static Long calcularPotencia(int x, int z) {
15        // Inicializando o resultado como 1
16        Long resultado = 1;
17
18        // Loop para calcular a potência
19        for (int i = 0; i < z; i++) {
20            resultado *= x;
21        }
22
23        return resultado;
24    }
25 }
26
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
\globalStorage\pleiades.java-extension-pack-jdk\java\21\bin\java.exe" --enable-preview -XX:+ShowCode
DetailsInExceptionMessages -cp "C:\Users\Kevin Brandao\AppData\Roaming\Code\User\workspaceStorage\e8
e5515bcbad37db79c9cbcc981138b\redhat.java\jdt_ws\Aulas_java_ea486e02\bin" Main12 "
Resultado de 2^5 = 32
```

13. Crie uma função que receba como parâmetro um valor inteiro e gere como saída n linhas com pontos de exclamação o, conforme o exemplo abaixo (para n = 5):

!
!!
!!!

!!!!

!!!!!!

```
lista_exercicios_2 > J Main13.java > Main13 > gerarExclamacoes(int)
1  public class Main13 {
    Run | Debug
2      public static void main(String[] args) {
3          int n = 5; // Defina o valor de n
4          gerarExclamacoes(n);
5      }
6
7      public static void gerarExclamacoes(int n) {
8          for (int i = 1; i <= n; i++) {
9              for (int j = 1; j <= i; j++) {
10                 System.out.print(s:"KEVIN ! ");
11             }
12             System.out.println(); // Pula para a próxima linha após imprimir os pontos
13         }
14     }
15 }
16
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
dt_ws\Aulas_java_ea406e02\bin" Main13 "
K
KK
KKK
KKKK
KKKKK
```

14. Escreva uma função que gera um triângulo de altura e lados n e base $2*n-1$. Por exemplo, a saída para $n = 6$ seria:

```

1  public class Main14 {
    Run | Debug
2      public static void main(String[] args) {
3          int n = 6; // Defina o valor de n
4          gerarTriangulo(n);
5      }
6
7      public static void gerarTriangulo(int n) {
8          // Loop para iterar pelas linhas
9          for (int i = 1; i <= n; i++) {
10             // Loop para imprimir os espaços em branco à esquerda
11             for (int j = 1; j <= n - i; j++) {
12                 System.out.print(s: " ");
13             }
14             // Loop para imprimir os asteriscos
15             for (int k = 1; k <= 2 * i - 1; k++) {
16                 System.out.print(s: "K");
17             }
18             // Pula para a próxima linha após imprimir uma linha do triângulo
19             System.out.println();
20         }
21     }
22 }
23

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

dt_ws\Aulas_java_ea406e02\bin" Main13 "

K
KK
KKK
KKKK
KKKKK

c:\Users\Kevin Brandao\Desktop\Aulas_java> c: && cd "c:\Users\Kevin Brandao\Desktop\Aulas_java" && c
md /C ""C:\Users\Kevin Brandao\AppData\Roaming\Code\User\globalStorage\pleiades.java-extension-pack-
jdk\java\21\bin\java.exe" --enable-preview -XX:+ShowCodeDetailsInExceptionMessages -cp "C:\Users\Kev
in Brandao\AppData\Roaming\Code\User\workspaceStorage\e8e5515bcbaa0d37db79ccbcc981138b\redhat.java\j
dt_ws\Aulas_java_ea406e02\bin" Main13 "

KEVIN
KEVINKEVIN
KEVINKEVINKEVIN
KEVINKEVINKEVINKEVIN
KEVINKEVINKEVINKEVINKEVIN

c:\Users\Kevin Brandao\Desktop\Aulas_java> c: && cd "c:\Users\Kevin Brandao\Desktop\Aulas_java" && c
md /C ""C:\Users\Kevin Brandao\AppData\Roaming\Code\User\globalStorage\pleiades.java-extension-pack-
jdk\java\21\bin\java.exe" --enable-preview -XX:+ShowCodeDetailsInExceptionMessages -cp "C:\Users\Kev
in Brandao\AppData\Roaming\Code\User\workspaceStorage\e8e5515bcbaa0d37db79ccbcc981138b\redhat.java\j
dt_ws\Aulas_java_ea406e02\bin" Main13 "

KEVIN !
KEVIN ! KEVIN !
KEVIN ! KEVIN ! KEVIN !
KEVIN ! KEVIN ! KEVIN ! KEVIN !
KEVIN ! KEVIN ! KEVIN ! KEVIN ! KEVIN !