



FACULDADE DE TECNOLOGIA- FATEC  
Curso: Desenvolvimento de Software Multiplataforma  
Disciplina: Técnicas de Programação II

**Aluno:** Kevin de Almeida Brandão **RA:**1091392323045 **Data:** 02/03/2024

## EXERCÍCIO 01

### **1. O que é uma variável? Descreva como é feita a declaração de variável e explique em termos de memória sua relação.**

R: Uma variável é um espaço de armazenamento na memória do computador que contém um valor específico, como um número, uma string de texto ou outro tipo de dado. Em termos simples, é como uma caixa onde podemos guardar informações para serem usadas posteriormente.

A declaração de uma variável é feita atribuindo a ela um nome e um tipo de dado. Por exemplo, em muitas linguagens de programação, podemos declarar uma variável chamada "idade" do tipo inteiro (int) e atribuir a ela o valor 25 da seguinte forma:

```
int idade = 25;
```

Quando essa declaração é feita, o computador reserva um espaço na memória para armazenar o valor da variável "idade".

Essa reserva de memória depende do tipo de dado da variável. Por exemplo, um inteiro geralmente ocupa 4 bytes na memória, enquanto uma string pode ocupar mais espaço, dependendo do seu tamanho. Quando a variável é utilizada no programa, o computador acessa esse espaço de memória para obter ou modificar o valor armazenado.

### **2. Linguagens de programação podem ser compiladas e interpretadas, explique o conceito de cada contexto ressaltando suas diferenças. Em relação à Linguagem Java, em qual desses contextos se encaixa?**

R:

Compiladas:

Nas linguagens compiladas, o código-fonte escrito pelo programador é traduzido para linguagem de máquina (código binário) por um programa chamado compilador.

O código fonte é transformado em um arquivo executável que pode ser diretamente executado pelo computador. Exemplos de linguagens compiladas incluem C, C++, e Rust.

Interpretadas:

Nas linguagens interpretadas, o código-fonte é executado linha por linha por um programa chamado interpretador.

O interpretador traduz e executa cada linha do código conforme necessário, sem a necessidade de gerar um arquivo executável separado. Exemplos de linguagens interpretadas incluem Python, JavaScript e Ruby.

Agora, em relação à linguagem Java, ela se encaixa em ambos os contextos. Java é inicialmente compilada para um bytecode intermediário pela máquina virtual Java (JVM) e depois interpretada pela JVM em tempo de execução. Isso significa que o código Java é compilado uma vez para bytecode e pode ser executado em qualquer plataforma que tenha uma JVM, tornando-o uma linguagem "compilada" e "interpretada" ao mesmo tempo.

### **3. Explique o que é: classe e objeto. Qual relação entre eles?**

R:

Classe:

Uma classe é como um modelo ou um plano para criar objetos. Ela define os atributos (propriedades) e métodos (comportamentos) que os objetos de sua categoria terão. Por exemplo, se você estiver criando um programa para representar carros, uma classe chamada "Carro" poderia ter atributos como cor, marca, modelo, e métodos como ligar(), desligar(), acelerar(), etc. Uma classe serve como um tipo de molde para criar objetos que compartilham as mesmas características e comportamentos.

Objeto:

Um objeto é uma instância de uma classe. Em termos simples, é uma entidade que é criada usando a estrutura definida pela classe. Por exemplo, se "Carro" é a classe, então um objeto seria um carro específico com características específicas, como um carro vermelho da marca Toyota modelo Corolla. Cada objeto tem seus próprios valores para os atributos definidos na classe e pode chamar os métodos definidos na classe.

Relação entre classe e objeto:

A relação entre classe e objeto é como a relação entre um plano e uma casa construída usando esse plano. A classe é o plano que define as características e comportamentos que os objetos terão. Os objetos são as instâncias reais, criadas com base nesse plano, com valores específicos para os atributos e a capacidade de realizar as ações definidas pelos métodos. Em resumo, a classe é a estrutura abstrata e os objetos são as instâncias concretas dessa estrutura.

## Questões para prática da sintaxe e aplicação de lógica

4 - Crie um programa que receba um número inteiro n e leia n valores do teclado e compute quantos destes valores são negativos.

```
J 4.java x
J 4.java
1 public class ContNegativos {
2     public static void main(String... args) {
3         //Scanner é uma classe em Java que fornece métodos para
4         //analisar tipos de dados primitivos e strings a partir de
5         //fontes de entrada, como o teclado,
6         //arquivos e strings. Ele é encontrado no pacote java.util.
7         Scanner scanner = new Scanner(System.in);
8
9         // Pedindo ao usuário para inserir o número de valores
10        System.out.print("Digite o número de valores a serem inseridos: ");
11        // nextInt Este método espera que o próximo token de entrada seja
12        // um número inteiro e o converte para o tipo
13        int n = scanner.nextInt();
14
15        int contadorNegativos = 0;
16
17        // Pedindo ao usuário para inserir os valores e contando os negativos
18        System.out.println("Digite os valores:");
19        for (int i = 0; i < n; i++) {
20            int valor = scanner.nextInt();
21            if (valor < 0) {
22                contadorNegativos++;
23            }
24        }
25
26        // Exibindo o resultado
27        System.out.println("Total de valores negativos: " + contadorNegativos);
28
29        // Fechando o scanner
30        scanner.close();
31    }
32 }
```

5 - Crie um programa que receba um valor inteiro informado pelo usuário e determine se esse valor é um número primo. Observe que os números primos são aqueles que são divisíveis apenas por eles próprios e por 1.

J 5.java

```
1 public class VerificadorNumeroPrimo {
2     public static void main(String... args) {
3         Scanner scanner = new Scanner(System.in);
4
5         // Pedindo ao usuário para inserir o número
6         System.out.print("Digite um número inteiro positivo: ");
7         int numero = scanner.nextInt();
8
9         boolean primo = true;
10
11        // Verificando se o número é primo
12        if (numero <= 1) {
13            primo = false;
14        } else {
15            for (int i = 2; i < numero; i++) {
16                if (numero % i == 0) {
17                    primo = false;
18                    break;
19                }
20            }
21        }
22
23        // Exibindo o resultado
24        if (primo) {
25            System.out.println(numero + " é um número primo.");
26        } else {
27            System.out.println(numero + " não é um número primo.");
28        }
29
30        // Fechando o scanner
31        scanner.close();
32    }
33 }
```

6. Crie um programa que permita calcular o valor do salário líquido de um funcionário considerando o desconto do INSS. O desconto da contribuição do INSS é dado segundo a tabela:

```
6.java
public class CalculadoraINSS {
    public static void main(String... args) {
        Scanner scanner = new Scanner(System.in);

        // Tabela de alíquotas e deduções do INSS
        double[][] tabelaINSS = {
            {1100.0, 7.5, 0.0},
            {2203.48, 9.0, 16.5},
            {3305.22, 12.0, 82.6},
            {6433.57, 14.0, 148.72}
        };

        // Pedindo ao usuário para inserir o salário bruto
        System.out.println("#####");
        System.out.println("CALCULADORA DE INSS");
        System.out.println("#####");
        System.out.print("Salário Bruto ....: ");
        double salarioBruto = scanner.nextDouble();

        // Calculando o desconto do INSS
        double descontoINSS = 0.0;
        for (int i = 0; i < tabelaINSS.length; i++) {
            if (salarioBruto <= tabelaINSS[i][0]) {
                descontoINSS = (salarioBruto * tabelaINSS[i][1] / 100) - tabelaINSS[i][2];
                break;
            }
        }

        // Calculando o salário líquido
        double salarioLiquido = salarioBruto - descontoINSS;

        // Exibindo os resultados
        System.out.println("Alíquota INSS (%): " + tabelaINSS[0][1]);
        System.out.println("Desconto INSS .....: " + String.format("%.2f", descontoINSS));
        System.out.println("Salário Líquido ...: " + String.format("%.2f", salarioLiquido));
        System.out.println("#####");

        // Fechando o scanner
        scanner.close();
    }
}
```

7 - Crie uma classe que represente um aluno considerando os atributos: nome, p1, p2. Em outra classe, no mesmo pacote crie uma classe principal (Main.java) e:

- crie uma variável a1 do tipo aluno que receba a instância de Aluno;
- atribua valores arbitrários;
- mostre os valores;
- crie uma variável a2 que receba a1;
- altere os valores da p1 e p2 na variável a2 e mostre os valores dos atributos de a1 e a2;

f. os valores são diferentes? explique.

```
7aluno.java X
J 7aluno.java
1  public class Aluno {
2      private String nome;
3      private double p1;
4      private double p2;
5
6      // Construtor
7      public Aluno(String nome, double p1, double p2) {
8          this.nome = nome;
9          this.p1 = p1;
10         this.p2 = p2;
11     }
12
13     // Getters e Setters
14     public String getNome() {
15         return nome;
16     }
17
18     public void setNome(String nome) {
19         this.nome = nome;
20     }
21
22     public double getP1() {
23         return p1;
24     }
25
26     public void setP1(double p1) {
27         this.p1 = p1;
28     }
29
30     public double getP2() {
31         return p2;
32     }
33
34     public void setP2(double p2) {
35         this.p2 = p2;
36     }
37 }
```

```

38
39 public class Main {
40     public static void main(String[] args) {
41         // Criando uma instância da classe Aluno e atribuindo valores arbitrários
42         Aluno a1 = new Aluno("João", 7.5, 8.0);
43
44         // Mostrando os valores
45         System.out.println("Valores de a1: ");
46         System.out.println("Nome: " + a1.getNome());
47         System.out.println("P1: " + a1.getP1());
48         System.out.println("P2: " + a1.getP2());
49
50         // Criando uma nova variável a2 que recebe a1
51         Aluno a2 = a1;
52
53         // Alterando os valores de p1 e p2 na variável a2
54         a2.setP1(6.0);
55         a2.setP2(7.0);
56
57         // Mostrando os valores dos atributos de a1 e a2
58         System.out.println("\nValores de a1 após alteração em a2: ");
59         System.out.println("Nome: " + a1.getNome());
60         System.out.println("P1: " + a1.getP1());
61         System.out.println("P2: " + a1.getP2());
62
63         System.out.println("\nValores de a2: ");
64         System.out.println("Nome: " + a2.getNome());
65         System.out.println("P1: " + a2.getP1());
66         System.out.println("P2: " + a2.getP2());
67     }
68 }
69
70 /*explicação:
71 Quando atribuímos a1 para a2, estamos na verdade fazendo com que a2 aponte para o mesmo objeto na memória que a1.
72 Portanto, quando alteramos os valores de p1 e p2 em a2, essas alterações são refletidas no objeto original ao
73 qual a1 também está apontando. Portanto, ao mostrar os valores de a1 após as alterações em a2.
74 */

```

**9 - As aplicações financeiras apresentam diversas modalidades cujos rendimentos têm variações; os rendimentos do tipo renda fixa, são indexados à taxa Selic. Dado a isso crie um programa que calcule o rendimento em 12 meses, mês a mês, dada as possibilidades de investimento abaixo e indique qual o mais vantajoso, considerando os valores a serem informados taxa selic e o capital para aplicação:**

**a. CDB Banco Pague Mais**

**i. 85% da taxa selic**

**ii. taxa administrativa 10% do rendimento**

**b. CBD Fura Bolso**

**i. 93% da taxa selic**

**ii. taxa administrativa fixa de 1% sobre o capital;**

**iii. 6% imposto sobre o rendimento;**

**O programa deverá mostrar para cada investimento saída similar a:**

**Taxa selic: x**

**Capital investido:**

**Mes Capital Rendimento Total**

**1 1000,00 100,00 1100,00**

2 1100,00 110,00 1210,00

Rendimento Total: xxxxx

Descontos: yyyyyy

Rendimento Líquido: zzzzz

**OBS: feito com ajuda do meu amigo IA**

```
J 9java x
J 9java
1 import java.util.ArrayList;
2 import java.util.List;
3 import java.util.Scanner;
4
5 class Investimento {
6     private String nome;
7     private double taxaSelic;
8     private double taxaAdministrativa;
9     private double imposto;
10    private double capital;
11
12    public Investimento(String nome, double taxaSelic, double taxaAdministrativa, double imposto, double capital) {
13        this.nome = nome;
14        this.taxaSelic = taxaSelic;
15        this.taxaAdministrativa = taxaAdministrativa;
16        this.imposto = imposto;
17        this.capital = capital;
18    }
19
20    public String getNome() {
21        return nome;
22    }
23
24    public List<Double> calcularRendimento() {
25        List<Double> rendimentos = new ArrayList<>();
26        double total = capital;
27
28        for (int i = 1; i <= 12; i++) {
29            double rendimentoMes = capital * (taxaSelic * (nome.equals("CDB Banco Pague Mais") ? 0.85 : 0.93));
30            rendimentoMes -= rendimentoMes * taxaAdministrativa;
31            if (nome.equals("CDB Fura Bolso")) {
32                rendimentoMes -= rendimentoMes * imposto;
33            }
34            total += rendimentoMes;
35            rendimentos.add(rendimentoMes);
36        }
37        rendimentos.add(total);
38        return rendimentos;
39    }
40 }
41
42 public class Main {
43     public static void main(String[] args) {
44         Scanner scanner = new Scanner(System.in);
45
46         System.out.print("Informe a taxa Selic: ");
47         double taxaSelic = scanner.nextDouble();
48
49         System.out.print("Informe o capital para aplicação: ");
50         double capital = scanner.nextDouble();
51
52         Investimento cdbPagueMais = new Investimento("CDB Banco Pague Mais", taxaSelic, 0.10, 0, capital);
53         Investimento cdbFuraBolso = new Investimento("CDB Fura Bolso", taxaSelic, 0.01, 0.06, capital);
54
55         System.out.println("\nInvestimento CDB Banco Pague Mais:");
56         exibirResultados(cdbPagueMais.calcularRendimento());
57
58         System.out.println("\nInvestimento CDB Fura Bolso:");
59         exibirResultados(cdbFuraBolso.calcularRendimento());
60     }
61
62     public static void exibirResultados(List<Double> rendimentos) {
63         System.out.println("Taxa Selic: " + rendimentos.get(0));
64         System.out.println("Capital investido: " + rendimentos.get(0));
65         System.out.println("Mes Capital Rendimento Total");
66         for (int i = 0; i < 12; i++) {
67             System.out.printf("%d %2f %2f %2f\n", i + 1, rendimentos.get(0), rendimentos.get(i), rendimentos.get(i) + rendimentos.get(0));
68         }
69         System.out.printf("Rendimento Total: %2f\n", total);
70         System.out.printf("Descontos: %2f\n", descontos);
71         System.out.printf("Rendimento Líquido: %2f\n", rendimentoLiquido);
72     }
73 }
74
```



**10 - Uma venda de alimentos precisa contabilizar suas vendas. Para tanto, crie um programa que solicite o usuário o informe de 10 vendas. A cada venda deverá ser informado: código produto, valor unitário e quantidade. Use uma classe para representar a Venda. O programa deverá:**

- a. A cada venda informada mostrar o valor total (quantidade \* valor unitário);**
- b. Valor médio vendido;**
- c. Mostrar o código do produto, quantidade e valor total do produto com maior venda;**
- d. Mostrar o código do produto, quantidade e valor total do produto com menor venda;**
- e. Mostrar o valor Total Vendido;**
- f. Mostrar a quantidade total de itens vendidos.**

```

J 10.java x
J 10.java
3  class Venda {
8      public Venda(int codigoProduto, double valorUnitario, int quantidade) {
12  }
13
14      public double calcularValorTotal() {
15          return valorUnitario * quantidade;
16      }
17
18      public int getCodigoProduto() {
19          return codigoProduto;
20      }
21
22      public int getQuantidade() {
23          return quantidade;
24      }
25  }
26
27  public class Main {
28      public static void main(String[] args) {
29          Scanner scanner = new Scanner(System.in);
30          Venda[] vendas = new Venda[10];
31
32          double totalVendido = 0;
33          int quantidadeTotal = 0;
34
35          // Solicitar informações das vendas
36          for (int i = 0; i < 10; i++) {
37              System.out.println("Informe os dados da venda " + (i + 1) + ":");
38              System.out.print("Código do produto: ");
39              int codigoProduto = scanner.nextInt();
40              System.out.print("Valor unitário: ");
41              double valorUnitario = scanner.nextDouble();
42              System.out.print("Quantidade: ");
43              int quantidade = scanner.nextInt();
44
45              vendas[i] = new Venda(codigoProduto, valorUnitario, quantidade);
46
47              double valorTotalVenda = vendas[i].calcularValorTotal();
48              totalVendido += valorTotalVenda;
49              quantidadeTotal += quantidade;
50
51              System.out.println("Valor total da venda: " + valorTotalVenda);
52          }
53
54          // Calcular valor médio vendido
55          double valorMedioVendido = totalVendido / 10;
56
57          // Encontrar a maior e menor venda
58          Venda maiorVenda = vendas[0];
59          Venda menorVenda = vendas[0];
60          for (int i = 1; i < 10; i++) {
61              if (vendas[i].calcularValorTotal() > maiorVenda.calcularValorTotal()) {
62                  maiorVenda = vendas[i];
63              }
64              if (vendas[i].calcularValorTotal() < menorVenda.calcularValorTotal()) {
65                  menorVenda = vendas[i];
66              }
67          }
68
69          // Exibir resultados
70          System.out.println("\nValor médio vendido: " + valorMedioVendido);
71          System.out.println("Produto com maior venda:");
72          System.out.println("Código do produto: " + maiorVenda.getCodigoProduto());
73          System.out.println("Quantidade: " + maiorVenda.getQuantidade());
74          System.out.println("Valor total da venda: " + maiorVenda.calcularValorTotal());
75          System.out.println("Produto com menor venda:");
76          System.out.println("Código do produto: " + menorVenda.getCodigoProduto());
77          System.out.println("Quantidade: " + menorVenda.getQuantidade());
78          System.out.println("Valor total da venda: " + menorVenda.calcularValorTotal());
79          System.out.println("Valor total vendido: " + totalVendido);
80          System.out.println("Quantidade total de itens vendidos: " + quantidadeTotal);

```