

Exercício 1: Classe de Pessoas Crie uma classe chamada Pessoa com atributos como nome, idade e sexo. Escreva um método para exibir os detalhes da pessoa.

```
*Pessoa.java X
1 package principal;
2
3 public class Pessoa {
4     // Atributos da classe
5     private String nome;
6     private int idade;
7     private char sexo;
8
9     //
10 public Pessoa(String nome, int idade, char sexo) {
11     this.nome = nome;
12     this.idade = idade;
13     this.sexo = sexo;
14 }
15
16 // Mostra os detalhes dos Seres
17 public void exibirDetalhes() {
18     System.out.println("Nome: " + nome);
19     System.out.println("Idade: " + idade);
20     System.out.println("Sexo: " + sexo);
21 }
22
23 public static void main(String[] args) {
24     // Exemplo de uso da classe Pessoa
25     Pessoa pessoa1 = new Pessoa("João", 25, 'M');
26     Pessoa pessoa2 = new Pessoa("Maria", 30, 'F');
27
28     System.out.println("Detalhes da Pessoa 1:");
29     pessoa1.exibirDetalhes();
30
31     System.out.println("\nDetalhes da Pessoa 2:");
32     pessoa2.exibirDetalhes();
33 }
34 }
```

Exercício 2: Calculadora Simples Crie uma classe Calculadora com métodos para adição, subtração, multiplicação e divisão. Peça ao usuário para inserir dois números e a operação desejada.

```

1 package principal;
2
3 import java.util.Scanner;
4
5 public class Calculadora {
6     // Método para adição
7     public static double adicao(double num1, double num2) {
8         return num1 + num2;
9     }
10
11     // Método para subtração
12     public static double subtracao(double num1, double num2) {
13         return num1 - num2;
14     }
15
16     // Método para multiplicação
17     public static double multiplicacao(double num1, double num2) {
18         return num1 * num2;
19     }
20
21     // Método para divisão
22     public static double divisao(double num1, double num2) {
23         if (num2 == 0) {
24             System.out.println("Erro! Não é possível dividir por zero.");
25             return Double.NaN; // Retorna um valor de NaN (Not-a-Number)
26         }
27         return num1 / num2;
28     }
29
30     public static void main(String[] args) {
31         Scanner scanner = new Scanner(System.in);
32
33         System.out.print("Digite o primeiro número: ");
34         double numero1 = scanner.nextDouble();
35
36         System.out.print("Digite o segundo número: ");
37         double numero2 = scanner.nextDouble();
38
39         System.out.print("Digite a operação desejada (+, -, *, /): ");
40         char operacao = scanner.next().charAt(0);
41
42         double resultado = 0.0;
43     }
44 }
```

```

44         switch (operacao) {
45             case '+':
46                 resultado = adicao(numero1, numero2);
47                 break;
48             case '-':
49                 resultado = subtracao(numero1, numero2);
50                 break;
51             case '*':
52                 resultado = multiplicacao(numero1, numero2);
53                 break;
54             case '/':
55                 resultado = divisao(numero1, numero2);
56                 break;
57             default:
58                 System.out.println("Operação inválida!");
59         }
60
61         System.out.println("Resultado: " + resultado);
62
63         scanner.close();
64     }
65 }
66

```

Exercício 3: Conversor de Moeda Crie uma classe `ConversorMoeda` que converte uma quantia de dólares para euros. O valor da taxa de câmbio deve ser armazenado em uma constante.

```

1 package principal;
2 public class ConversorMoeda {
3     // Taxa de câmbio apenas.
4     private static final double TAXA_CAMBIO = 0.85;
5
6     // Converter dólares para euros
7     public static double converterDolaresParaEuros(double dolares) {
8         return dolares * TAXA_CAMBIO;
9     }
10
11     public static void main(String[] args) {
12         double valorEmDolares = 100.0;
13
14         double valorEmEuros = converterDolaresParaEuros(valorEmDolares);
15
16         System.out.println(valorEmDolares + " dólares são equivalentes a " + valorEmEuros + " euros.");
17     }
18 }
19

```

Exercício 4: Números Primos Escreva um programa para verificar se um número fornecido pelo usuário é primo ou não. Crie uma classe NumeroPrimo com um método para realizar essa verificação.

```
NumeroPrimo.java X
1 package principal;
2 import java.util.Scanner;
3
4 public class NumeroPrimo {
5     // Verificar se um número é primo
6     public static boolean ehPrimo(int numero) {
7         if (numero <= 1) {
8             return false;
9         }
10        if (numero <= 3) {
11            return true;
12        }
13        if (numero % 2 == 0 || numero % 3 == 0) {
14            return false;
15        }
16
17        for (int i = 5; i * i <= numero; i += 6) {
18            if (numero % i == 0 || numero % (i + 2) == 0) {
19                return false;
20            }
21        }
22
23        return true;
24    }
25
26    public static void main(String[] args) {
27        Scanner scanner = new Scanner(System.in);
28
29        System.out.print("Digite um número para verificar se é primo: ");
30        int numero = scanner.nextInt();
31
32        if (ehPrimo(numero)) {
33            System.out.println(numero + " é um número primo.");
34        } else {
35            System.out.println(numero + " não é um número primo.");
36        }
37
38        scanner.close();
39    }
40 }
```

Exercício 5: Cadastro de Alunos Crie uma classe Aluno com atributos como nome, matrícula e notas. Escreva métodos para calcular a média das notas e verificar se o aluno foi aprovado.

Aluno.java X

```
1 package principal;
2 import java.util.Arrays;
3
4 public class Aluno {
5     private String nome;
6     private String matricula;
7     private double[] notas;
8
9     public Aluno(String nome, String matricula, double[] notas) {
10         this.nome = nome;
11         this.matricula = matricula;
12         this.notas = notas;
13     }
14
15     public double calcularMedia() {
16         if (notas.length == 0) {
17             return 0.0;
18         }
19
20         double soma = 0.0;
21         for (double nota : notas) {
22             soma += nota;
23         }
24
25         return soma / notas.length;
26     }
27
28     public boolean estaAprovado(double mediaMinima) {
29         double media = calcularMedia();
30         return media >= mediaMinima;
31     }
32 }
```

```

public boolean estaAprovado(double mediaMinima) {
    double media = calcularMedia();
    return media >= mediaMinima;
}

public static void main(String[] args) {
    String nome = "João";
    String matricula = "2023001";
    double[] notas = {8.5, 7.0, 9.2, 6.8}; // Substitua com as notas do aluno

    Aluno aluno = new Aluno(nome, matricula, notas);

    double mediaMinima = 7.0; // Define basicamente a média

    double media = aluno.calcularMedia();
    boolean aprovado = aluno.estaAprovado(mediaMinima);

    System.out.println("Nome: " + aluno.nome);
    System.out.println("Matrícula: " + aluno.matricula);
    System.out.println("Notas: " + Arrays.toString(aluno.notas));
    System.out.println("Média: " + media);

    if (aprovado) {
        System.out.println("O aluno está aprovado!");
    } else {
        System.out.println("O aluno não está aprovado.");
    }
}
}

```

Exercício 6: Conta Bancária Crie uma classe `ContaBancaria` com atributos como número da conta, saldo e métodos para depositar e sacar dinheiro. Exiba o saldo após cada transação.

```

ContaBancaria.java X
1 package principal;
2 public class ContaBancaria {
3     private String numeroConta;
4     private double saldo;
5
6
7     public ContaBancaria(String numeroConta, double saldoInicial) {
8         this.numeroConta = numeroConta;
9         this.saldo = saldoInicial;
10    }
11
12    // Vai exibir o meu saldo
13    public void exibirSaldo() {
14        System.out.println("Saldo da conta " + numeroConta + ": R$" + saldo);
15    }
16
17    // Aqui será a forma para realizar um depósito
18    public void depositar(double valor) {
19        if (valor > 0) {
20            saldo += valor;
21            System.out.println("Depósito de R$" + valor + " realizado com sucesso.");
22            exibirSaldo();
23        } else {
24            System.out.println("Valor de depósito inválido. O valor deve ser maior que zero.");
25        }
26    }
27
28    // Aqui será a forma para realizar um saque suaves..
29    public void sacar(double valor) {
30        if (valor > 0 && valor <= saldo) {
31            saldo -= valor;
32            System.out.println("Saque de R$" + valor + " realizado com sucesso.");
33            exibirSaldo();
34        } else if (valor <= 0) {
35            System.out.println("Valor de saque inválido. O valor deve ser maior que zero.");
36        } else {
37            System.out.println("Saldo insuficiente para o saque de R$" + valor);
38        }
39    }
40
41    public static void main(String[] args) {
42        // Criando uma conta
43        ContaBancaria minhaConta = new ContaBancaria("12345", 1000.0);
44
45        // Abaixo e o saldo inicial
46        minhaConta.exibirSaldo();
47
48        // Aqui realizo os meus depósitos e saques.
49        minhaConta.depositar(500.0);
50        minhaConta.sacar(200.0);
51        minhaConta.sacar(1500.0);
52
53
54        minhaConta.exibirSaldo();
55    }
56 }
57

```

Exercício 7: Lista de Compras Crie uma classe ListaCompras que permite ao usuário adicionar e remover itens de uma lista de compras. Implemente métodos para listar os itens e calcular o custo total.

ListaCompras.java X

```
1 package principal;
2 import java.util.ArrayList;
3
4
5 public class ListaCompras {
6     private List<ItemCompra> listaDeCompras;
7
8
9     public ListaCompras() {
10         listaDeCompras = new ArrayList<>();
11     }
12
13     //Aqui eu adiciono um item à lista de compras
14     public void adicionarItem(String nome, double preco, int quantidade) {
15         ItemCompra item = new ItemCompra(nome, preco, quantidade);
16         listaDeCompras.add(item);
17         System.out.println("Item adicionado à lista: " + item);
18     }
19
20     //Você remover um item da lista de compras
21     public void removerItem(String nome) {
22         for (ItemCompra item : listaDeCompras) {
23             if (item.getNome().equalsIgnoreCase(nome)) {
24                 listaDeCompras.remove(item);
25                 System.out.println("Item removido da lista: " + item);
26                 return;
27             }
28         }
29         System.out.println("Item não encontrado na lista: " + nome);
30     }
31
32     //Você listar os itens na lista de compras
33     public void listarItens() {
34         System.out.println("Lista de Compras:");
35         for (ItemCompra item : listaDeCompras) {
36             System.out.println(item);
37         }
38     }
39 }
```


ListaCompras.java

```
39
40 //De certa forma irei calcular o custo total da lista de compras
41 public double calcularCustoTotal() {
42     double custoTotal = 0.0;
43     for (ItemCompra item : listaDeCompras) {
44         custoTotal += item.getPrecoTotal();
45     }
46     return custoTotal;
47 }
48
49 public static void main(String[] args) {
50     ListaCompras lista = new ListaCompras();
51
52     lista.adicionarItem("Maçã", 2.5, 3);
53     lista.adicionarItem("Leite", 1.0, 2);
54     lista.adicionarItem("Pão", 3.0, 1);
55
56     lista.listarItens();
57
58     lista.removerItem("Leite");
59
60     lista.listarItens();
61
62     double custoTotal = lista.calcularCustoTotal();
63     System.out.println("Custo Total da Compra: R$" + custoTotal);
64 }
65 }
66
67 class ItemCompra {
68     private String nome;
69     private double preco;
70     private int quantidade;
71
72     public ItemCompra(String nome, double preco, int quantidade) {
73         this.nome = nome;
74         this.preco = preco;
75         this.quantidade = quantidade;
76     }
77
78     public String getNome() {
79         return nome;
80     }
81
82     public double getPrecoTotal() {
83         return preco * quantidade;
84     }
85 }
```

Exercício 8: Relógio Digital Crie uma classe Relogio que simule um relógio digital. Implemente métodos para definir a hora e os minutos e para exibir a hora atual.

```
| "Relogio.java" X
1 package principal;
2 public class Relogio {
3     private int hora;
4     private int minutos;
5
6
7     public Relogio() {
8         hora = 0;
9         minutos = 0;
10    }
11
12    // Serão definidos as horas e minutos
13    public void definirHora(int hora, int minutos) {
14        if (hora >= 0 && hora <= 23 && minutos >= 0 && minutos <= 59) {
15            this.hora = hora;
16            this.minutos = minutos;
17        } else {
18            System.out.println("Hora ou minutos inválidos. A hora deve estar entre 0 e 23, e os minutos entre 0 e 59.")
19        }
20    }
21
22    // Vai ser exibido a hora total
23    public void exibirHoraAtual() {
24        System.out.println("Hora atual: " + String.format("%02d", hora) + ":" + String.format("%02d", minutos));
25    }
26
27    public static void main(String[] args) {
28        Relogio relógio = new Relogio();
29
30        relógio.definirHora(10, 30);
31        relógio.exibirHoraAtual();
32
33        relógio.definirHora(25, 70); // Tentativa de definir uma hora inválida
34        relógio.exibirHoraAtual();
35    }
36 }
37
```

Exercício 9: Conversor de Temperatura Crie uma classe `ConversorTemperatura` que converte uma temperatura em graus Celsius para Fahrenheit. Implemente métodos para realizar essa conversão.

```
1 package principal;
2 public class ConversorTemperatura {
3     // Celsius para Fahrenheit
4     public static double celsiusParaFahrenheit(double celsius) {
5         double fahrenheit = (celsius * 9/5) + 32;
6         return fahrenheit;
7     }
8
9     // Fahrenheit para Celsius
10    public static double fahrenheitParaCelsius(double fahrenheit) {
11        double celsius = (fahrenheit - 32) * 5/9;
12        return celsius;
13    }
14
15    public static void main(String[] args) {
16        double celsius = 25.0;
17        double fahrenheit = celsiusParaFahrenheit(celsius);
18
19        System.out.println(celsius + " graus Celsius equivalem a " + fahrenheit + " graus Fahrenheit.");
20
21        double fahrenheit2 = 77.0;
22        double celsius2 = fahrenheitParaCelsius(fahrenheit2);
23
24        System.out.println(fahrenheit2 + " graus Fahrenheit equivalem a " + celsius2 + " graus Celsius.");
25    }
26 }
27
```

Exercício 10: Calculadora de IMC Escreva um programa que calcule o Índice de Massa Corporal (IMC) de uma pessoa. Crie uma classe `CalculadoraIMC` com métodos para calcular o IMC com base no peso e altura inseridos pelo usuário.



*CalculadoraIMC.java X

```
1 package principal;
2 import java.util.Scanner;
3
4 public class CalculadoraIMC {
5
6     public static double calcularIMC(double peso, double altura) {
7         if (peso <= 0 || altura <= 0) {
8             System.out.println("Peso e altura devem ser números positivos.");
9             return -1.0; // Valor inválido
10        }
11
12        double imc = peso / (altura * altura);
13        return imc;
14    }
15
16    public static void main(String[] args) {
17        Scanner scanner = new Scanner(System.in);
18
19        System.out.print("Digite seu peso (em kg): ");
20        double peso = scanner.nextDouble();
21
22        System.out.print("Digite sua altura (em metros): ");
23        double altura = scanner.nextDouble();
24
25        double imc = calcularIMC(peso, altura);
26
27        if (imc != -1.0) {
28            System.out.println("Seu IMC é: " + imc);
29
30            if (imc < 18.5) {
31                System.out.println("Abaixo do peso.");
32            } else if (imc < 24.9) {
33                System.out.println("Peso normal.");
34            } else if (imc < 29.9) {
35                System.out.println("Sobrepeso.");
36            } else if (imc < 34.9) {
37                System.out.println("Obesidade Grau I.");
38            } else if (imc < 39.9) {
39                System.out.println("Obesidade Grau II.");
40            } else {
41                System.out.println("Obesidade Grau III (Obesidade Mórbida).");
42            }
43        }
44
45        scanner.close();
46    }
47 }
```