

Prática 01 – POO2

Nome: Wallace Geraldo Marcos Pinheiro

Criação da classe geral:

```
public class geral {  
    String titulo;  
    int ano;  
    public geral(String titulo, int ano)  
    {  
        this.titulo =titulo;  
        this.ano = ano;  
    }  
}
```

Criação da classe livro:

```
public class livro extends geral {  
  
    private String autor;  
  
    public livro(String titulo, String autor, int ano) {  
        super(titulo, ano);  
        this.autor = autor;  
    }  
  
    @Override  
    public String toString() {  
        return "Livro: " + getTitulo() + ", Autor: " + getAutor() + ",  
Ano: " + getAno();  
    }  
  
    public String getAutor() {  
        return autor;  
    }  
    public String getTitulo()  
    {  
        return titulo;  
    }  
    public int getAno()  
    {  
        return ano;  
    }  
}
```

Criação da classe Revistas:

```
public class revistas extends geral{  
    String Org;
```

```

        int vol;
        int nro;

        public revistas(String titulo, String Org, int vol, int nro, int
ano)
        {
            super(titulo,ano);
            this.Org = Org;
            this.vol = vol;
            this.nro = nro;

        }
        public String toString() {
            return "Revista: " + getTitulo() + ", Org: " + getOrg() + ",
Volume: " + getVol() +
                ", Numero: " + getNro() + ", Ano: " + getAno();
        }
        public String getOrg() {
            return Org;
        }

        public int getVol() {
            return vol;
        }

        public int getNro() {
            return nro;
        }
        public String getTitulo()
        {
            return titulo;
        }
        public int getAno()
        {
            return ano;
        }
    }
}

```

Criação da classe Main:

```

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.IOException;
import java.util.ArrayList;
import java.util.Scanner;

public class main {
    private ArrayList<geral> listaGeral = new ArrayList<>();
    private JFrame frame;
    private JTextField titulo;
    private JTextField autor;
    private JTextField ano;
    private JTextField org;
    private JTextField vol;
    private JTextField nro;

    public static void main(String[] args) {
        main instancia = new main();
    }
}

```

```

        int opcao;
        Scanner sc = new Scanner(System.in);
        System.out.println("Digite se você quer um livro[0] ou
revistas[1]");
        opcao = sc.nextInt();
        if (opcao == 0) {
            instancia.iniciarLivro();
        } else {
            instancia.iniciarRevistas();
        }
    }

    public void incluirRevista(String a, String b, int c, int d,
int e) {
        geral revista = new revistas(a, b, c, d, e);
        listaGeral.add(revista);
        gravarNoArquivo(revista.toString());
    }

    private void gravarNoArquivo(String conteudo) {
        try (BufferedWriter writer = new BufferedWriter(new
FileWriter("conteudo.txt", true))) {
            writer.write(conteudo);
            writer.newLine();
        } catch (IOException ex) {
            ex.printStackTrace();
        }
    }

    private void iniciarRevistas() {
        frame = new JFrame();
        JPanel inputPanel = new JPanel(new GridLayout(0, 2));
        // Layout para organizar os componentes verticalmente

        JLabel titulo2Label = new JLabel("REVISTAS");
        inputPanel.add(Box.createVerticalStrut(10));
        inputPanel.add(titulo2Label);

        JLabel tituloLabel = new JLabel("Título:");
        inputPanel.add(tituloLabel);
        titulo = new JTextField();
        inputPanel.add(titulo);
        titulo.setColumns(10);

        JLabel OrgLabel = new JLabel("Org:");
        inputPanel.add(OrgLabel);
        org = new JTextField();
        inputPanel.add(org);
        org.setColumns(10);

        JLabel VolLabel = new JLabel("Vol:");
        inputPanel.add(VolLabel);
        vol = new JTextField();
        inputPanel.add(vol);
        vol.setColumns(10);

        JLabel NroLabel = new JLabel("Nro:");
        inputPanel.add(NroLabel);
        nro = new JTextField();
        inputPanel.add(nro);
        nro.setColumns(10);
    }

```

```

        JLabel anoLabel = new JLabel("Ano:");
        inputPanel.add(anoLabel);
        ano = new JTextField();
        inputPanel.add(ano);
        ano.setColumns(10);

        JButton incluirButton = new JButton("Incluir");
        incluirButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                incluirRevista(titulo.getText(),
org.getText(), Integer.parseInt(vol.getText()),
Integer.parseInt(nro.getText()), Integer.parseInt(ano.getText()));
            }
        });

        JButton listagemButton = new JButton("Listagem");
        listagemButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                JFrame listagemFrame = new
JFrame("Listagem de Revistas");

                JTextArea listagemTextArea = new
JTextArea();

                JScrollPane scrollPane = new
JScrollPane(listagemTextArea);

                listagemFrame.getContentPane().add(scrollPane, BorderLayout.CENTER);

                for (geral item : listaGeral) {
                    if (item instanceof revistas)
{
                        listagemTextArea.append(item.toString() + "\n");
                    }
                }

                listagemFrame.setSize(400, 300);

                listagemFrame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
                listagemFrame.setVisible(true);
            }
        });

        JButton livroButton = new JButton("Livros");
        livroButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                iniciarLivro();
            }
        });

        inputPanel.add(incluirButton);
        inputPanel.add(listagemButton);
        inputPanel.add(livroButton);

        frame.getContentPane().add(inputPanel,
BorderLayout.NORTH);

        frame.setSize(400, 300);

```

```

        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setVisible(true);
    }

    public void imprimirLista() {
        System.out.println("Conteúdo da Lista:");
        for (geral item : listaGeral) {
            System.out.println(item);
        }
    }

    public void incluirLivro(String a, String b, int c) {
        geral livro = new livro(a, b, c);
        listaGeral.add(livro);
        System.out.println("Livro incluído: " + livro);
    }

    private void iniciarLivro() {
        frame = new JFrame();
        JPanel inputPanel = new JPanel(new GridLayout(0, 2));
        // Layout para organizar os componentes verticalmente

        JLabel tituloLabel = new JLabel("LIVROS");
        inputPanel.add(Box.createVerticalStrut(10));
        inputPanel.add(tituloLabel);
        JLabel tituloLabel2 = new JLabel("Título:");
        inputPanel.add(tituloLabel2);
        titulo = new JTextField();
        inputPanel.add(titulo);
        titulo.setColumns(10);

        JLabel autorLabel = new JLabel("Autor:");
        inputPanel.add(autorLabel);
        autor = new JTextField();
        inputPanel.add(autor);
        autor.setColumns(10);

        JLabel anoLabel = new JLabel("Ano:");
        inputPanel.add(anoLabel);
        ano = new JTextField();
        inputPanel.add(ano);
        ano.setColumns(10);

        JButton incluirButton = new JButton("Incluir");
        incluirButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                incluirLivro(titulo.getText(),
                    autor.getText(), Integer.parseInt(ano.getText()));
            }
        });

        JButton listagemButton = new JButton("Listagem");
        listagemButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                JFrame listagemFrame = new
JFrame("Listagem de Livros");

                JTextArea listagemTextArea = new
JTextArea();

                JScrollPane scrollPane = new

```

```

JScrollPane(listagemTextArea);

listagemFrame.getContentPane().add(scrollPane, BorderLayout.CENTER);

        for (geral item : listaGeral) {
            if (item instanceof livro) {

listagemTextArea.append(item.toString() + "\n");
                }
            }

        listagemFrame.setSize(400, 300);

listagemFrame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
        listagemFrame.setVisible(true);
    }

});

JButton revistaButton = new JButton("Revistas");
revistaButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        iniciarRevistas();
    }
});

inputPanel.add(incluirButton);
inputPanel.add(listagemButton);
inputPanel.add(revistaButton);

        frame.getContentPane().add(inputPanel,
BorderLayout.NORTH);

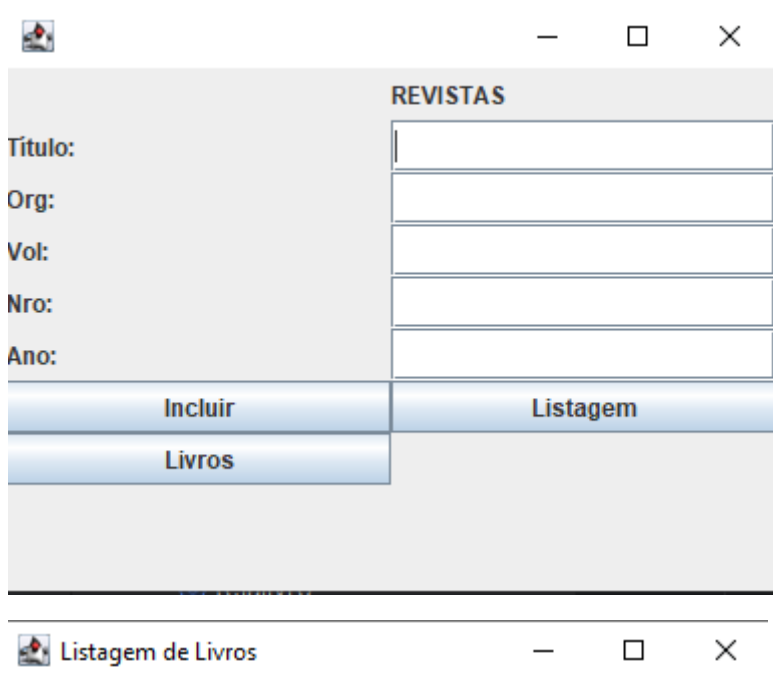
        frame.setSize(400, 300);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setVisible(true);
    }
}

```

Telas:

The screenshot shows a Java Swing window titled "LIVROS". The window contains a form with the following elements:

- Labels:** "Titulo:", "Autor:", and "Ano:" are positioned on the left side of the form.
- Input Fields:** Three empty text input fields are positioned on the right side, corresponding to the labels above them.
- Buttons:**
 - "Incluir" and "Listagem" are side-by-side buttons below the input fields.
 - "Revistas" is a button centered below the "Incluir" and "Listagem" buttons.



Explicação: Primeiramente verifiquei se existia atributos em comum entre livros e revistas, assim, com esses atributos eu criei uma classe para eles e também criei um construtor somente para eles. Após, eu criei as classes livros e revistas, que foram classes herdadas da classe geral declarando o restante dos atributos faltantes e chamado o super da classe geral. Após, na classe main, criei um pequeno menu, onde o usuário escolhe se ele quer revistas ou livro, logo após ele escolher uma opção, a tela abre para que ele possa digitar os dados. Caso, ele insira os dados corretamente e clique em incluir, os dados serão adicionados em um arquivo, e caso ele clique em listagem, os dados serão

mostrados. Usei o instance of dentro de livro e de revistas, para que na hora que peça para listar os livros ou revistas, ele só liste só o que foi pedido. Caso clique em listagem na página de livros, vai mostrar somente os livros adicionados. E caso clique no outro botão, ele redireciona para a página solicitada.