**Curso Profissional de Gestão e Programação de Sistemas Informáticos**

**Disciplina de Programação e Sistemas de Informação**

Módulo 11 – Programação Orientada a Objetos Avançada

Aluno: Carlos Tojal

Professora: Matilde Vieira

Caldas da Rainha, Janeiro de 2020

**Índice**

[Agradecimentos 4](#_Toc20780012)

[Índice de Figuras 5](#_Toc20780013)

[Siglas/Abreviaturas 6](#_Toc20780014)

[Introdução 1](#_Toc20780015)

[1. Conceitos do módulo 8 2](#_Toc20780016)

[1.1. Conceito de Programação Orientada a Eventos 2](#_Toc20780017)

[1.1.1. Eventos 2](#_Toc20780018)

[1.1.2. Exemplo de Programação Orientada a Eventos 2](#_Toc20780019)

[1.2. Conceito de Programação Orientada a Objetos 3](#_Toc20780020)

[1.2.1. Objetos 3](#_Toc20780021)

[1.2.2. Exemplo de Programação Orientada a Objetos 3](#_Toc20780022)

[Um exemplo da programação orientada a objetos é a criação de uma janela, que resulta da instanciação de uma classe que permite tal. 3](#_Toc20780023)

[Neste caso específico, essa classe deverá também conter métodos que permitem por exemplo a definição do tamanho da janela, estilos ou adição de componentes. 3](#_Toc20780024)

[Isto significa ainda que provavelmente esta classe deverá também conter atributos com alguns destes valores. 3](#_Toc20780025)

[1.3. Conceito de Interface Gráfica 4](#_Toc20780026)

[1.4. Conceito de Programa 4](#_Toc20780027)

[1.5. Conceito de Atributo 4](#_Toc20780028)

[1.6. Conceito de propriedades 4](#_Toc20780029)

[2. Momentos de trabalho na Programação Orientada a Objetos 5](#_Toc20780030)

[3. Tema 6](#_Toc20780031)

[3.1.1 O que é uma API? 6](#_Toc20780032)

[3.1.2 Tipos de API 7](#_Toc20780033)

[3.1.3 O que é uma API do sistema 7](#_Toc20780034)

[3.1.4 Em que circunstâncias são usadas APIs do sistema 7](#_Toc20780035)

[3.1.5 Como funcionam as APIs do sistema em JAVA 8](#_Toc20780036)

[3.1.6 Exemplos práticos da utilização de APIs do sistema na programação em JAVA 8](#_Toc20780037)

[3.1.7 APIs na prática 10](#_Toc20780038)

[3.1.8 O nosso projeto em prática 13](#_Toc20780039)

[Conclusão 18](#_Toc20780040)

[Apêndice A 19](#_Toc20780041)

[Referências Bibliográficas 20](#_Toc20780042)

Agradecimentos

Não podia deixar de agradecer a algumas pessoas pela conclusão deste módulo. Estas são:

* A professora Matilde Vieira, pelos conhecimentos transmitidos.
* A minha família, pelo tempo que dispensei com a realização de trabalhos.
* A mim próprio, pela vontade de aprender, motivação e ambição.

Introdução

O presente portfólio foi realizado no âmbito do módulo 11 da disciplina de programação e sistemas de informação, intitulado “Programação Orientada a Objetos Avançada”.

Portfólio

**Agenda**

**Contato.java**

//

// Copyright (c) Carlos Tojal 2020

// Agenda

// Contato.java

//

public class Contato {

// Atributos

private String nome;

private String telefone;

private String email;

// Construtores

public Contato() {

}

public Contato(String nome, String telefone, String email) {

this.nome = nome;

this.telefone = telefone;

this.email = email;

}

// Getters e Setters

public String getNome() {

return nome;

}

public String getTelefone() {

return telefone;

}

public String getEmail() {

return email;

}

public void setNome(String nome) {

this.nome = nome;

}

public void setTelefone(String telefone) {

this.telefone = telefone;

}

public void setEmail(String email) {

this.email = email;

}

}

GestorAgenda.java

import java.util.ArrayList;

import java.util.Iterator;

//

// Copyright (c) Carlos Tojal 2020

// Agenda

// GestorAgenda.java

//

public class GestorAgenda {

// Construtor

public GestorAgenda() {

}

// Métodos

public void adicionarContato(Contato contato, ArrayList<Contato> agenda) {

if(!agenda.contains(contato))

agenda.add(contato);

else

System.out.println("O contato já existe na agenda.");

}

public void listarContato(Contato contato) {

System.out.println("\n\n\*\* Contato \*\*");

System.out.println("Nome: " + contato.getNome());

System.out.println("Telefone: " + contato.getTelefone());

System.out.println("Email: " + contato.getEmail());

}

public void listarAgenda(ArrayList<Contato> agenda) {

for(int i = 0; i < agenda.size(); i++) {

listarContato(agenda.get(i));

}

}

}

**Menus.java**

//

// Copyright (c) Carlos Tojal 2020

// Agenda

// Menus.java

//

import java.util.ArrayList;

import java.util.Scanner;

public class Menus {

// Construtor

public Menus() {

}

public int menu() {

Scanner scanner = new Scanner(System.in);

int opt = 0;

do {

System.out.println("\n\*\* Agenda \*\*\n");

System.out.println("1. Adicionar Contato");

System.out.println("2. Listar Contatos");

System.out.println("3. Remover Contato");

System.out.println("0. Sair\n");

System.out.print("Opção: ");

opt = scanner.nextInt();

} while(opt < 0 || opt > 3);

return opt;

}

public void menuAdicionarContato(ArrayList<Contato> agenda) {

Scanner scanner = new Scanner(System.in);

GestorAgenda gestorAgenda = new GestorAgenda();

Contato contato = new Contato();

System.out.println("\n\*\* Adicionar Contato \*\*\n");

System.out.print("Nome: ");

contato.setNome(scanner.nextLine());

System.out.print("Telefone: ");

contato.setTelefone(scanner.nextLine());

System.out.print("Email: ");

contato.setEmail(scanner.nextLine());

gestorAgenda.adicionarContato(contato, agenda);

}

public void menuRemoverContato(ArrayList<Contato> agenda) {

GestorAgenda gestorAgenda = new GestorAgenda();

String query;

boolean found = false;

gestorAgenda.listarAgenda(agenda);

System.out.println("\nTelefone ou email a remover: ");

query = new Scanner(System.in).nextLine();

for(int i = 0; i < agenda.size(); i++) {

if(String.valueOf(agenda.get(i).getTelefone()).equals(query) || agenda.get(i).getEmail().equals(query)) {

agenda.remove(agenda.get(i));

found = true;

}

}

if(found)

System.out.println("Contato removido com sucesso.");

else

System.out.println("Contato não encontrado.");

}

}

**Principal.java**

//

// Copyright (c) Carlos Tojal 2020

// Agenda

// Principal.java

//

import java.util.ArrayList;

public class Principal {

static ArrayList<Contato> agenda = new ArrayList<Contato>();

static Menus menus = new Menus();

static GestorAgenda gestorAgenda = new GestorAgenda();

public static void main(String[] args) {

int opt;

do {

opt = menus.menu();

switch(opt) {

case 1:

menus.menuAdicionarContato(agenda);

break;

case 2:

gestorAgenda.listarAgenda(agenda);

break;

case 3:

menus.menuRemoverContato(agenda);

break;

}

}while(opt > 0);

}

}

Container\_component

**ScrollPaneDemo.java**

import javax.swing.ImageIcon;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JScrollPane;

public class ScrollPaneDemo extends JFrame {

public ScrollPaneDemo() {

super("ScrollPane Demo");

ImageIcon img = new ImageIcon("Imagem.jpg");

JScrollPane png = new JScrollPane(new JLabel(img));

getContentPane().add(png);

setSize(300,250);

setVisible(true);

}

public static void main(String[] args) {

new ScrollPaneDemo();

}

}

**UsaJButton.java**

import javax.swing.JFrame;

import javax.swing.JButton;

public class UsaJButton {

public static void main(String[] args) {

JFrame janela = new JFrame();

janela.setTitle("JButton");

janela.setSize(350,150);

janela.setLocation(50,50);

janela.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

JButton botao = new JButton("OK");

janela.add(botao);

janela.setVisible(true);

}

}

**UsaJCheckBox.java**

import javax.swing.JFrame;

import javax.swing.JCheckBox;

import java.awt.FlowLayout;

public class UsaJCheckBox {

public static void main(String[] args) {

JFrame janela = new JFrame();

janela.setTitle("JCheckBox");

janela.setSize(350,150);

janela.setLocation(50,50);

janela.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

janela.setLayout(new FlowLayout());

JCheckBox caixaVerificacao1 = new JCheckBox("Branco");

JCheckBox caixaVerificacao2 = new JCheckBox("Preto");

JCheckBox caixaVerificacao3 = new JCheckBox("Amarelo");

janela.add(caixaVerificacao1);

janela.add(caixaVerificacao2);

janela.add(caixaVerificacao3);

janela.setVisible(true);

}

}

**UsaJComboBox.java**

//

// Copyright (c) Carlos Tojal 2020

// Container\_Component

// UsaJComboBox.java

//

import javax.swing.JFrame;

import javax.swing.JComboBox;

import javax.swing.JLabel;

import java.awt.FlowLayout;

public class UsaJComboBox {

public static void main(String[] args) {

JFrame janela = new JFrame();

janela.setTitle("JComboBox");

janela.setSize(350, 150);

janela.setLocationRelativeTo(null);

// janela.setLocation(50, 50);

janela.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

janela.setLayout(new FlowLayout());

JLabel rotulo = new JLabel("Escreva o seu nome: ");

JComboBox<String> caixaCombinacao = new JComboBox<String>();

janela.add(rotulo);

caixaCombinacao.addItem("Branco");

caixaCombinacao.addItem("Preto");

caixaCombinacao.addItem("Amarelo");

janela.add(caixaCombinacao);

janela.setVisible(true);

}

}

**UsaJFrame.java**

//

// Copyright (c) Carlos Tojal 2020

// Container\_Component

// UsaJFrame.java

//

import javax.swing.JFrame;

public class UsaJFrame {

public static void main(String[] args) {

JFrame janela = new JFrame();

janela.setTitle("JFrame");

janela.setSize(350, 150);

janela.setLocation(50, 50);

janela.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

janela.setVisible(true);

}

}

**UsaJLabel.java**

//

// Copyright (c) Carlos Tojal 2020

// Container\_Component

// UsaJLabel.java

//

import javax.swing.JFrame;

import javax.swing.JLabel;

public class UsaJLabel {

public static void main(String[] args) {

JFrame janela = new JFrame();

janela.setTitle("JPanel");

janela.setSize(350, 150);

janela.setLocationRelativeTo(null);

// janela.setLocation(50, 50);

janela.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

JLabel rotulo = new JLabel("Escreva o seu nome: ");

janela.add(rotulo);

janela.setVisible(true);

}

}

**UsaJOptionPane.java**

import javax.swing.JOptionPane;

public class UsaJOptionPane {

public static void main (String[] args) {

JOptionPane.showMessageDialog(null,"Obrigado por ter utilizado este programa");

JOptionPane.showInputDialog("Escreva o seu nome");

JOptionPane.showConfirmDialog(null,"Deseja guardar as alterações?");

}

}

**UsaJOptionPane.java**

//

// Copyright (c) Carlos Tojal 2020

// Container\_Component

// UsaJPanel.java

//

import javax.swing.JFrame;

import javax.swing.JPanel;

public class UsaJPanel {

public static void main(String[] args) {

JFrame janela = new JFrame();

janela.setTitle("JPanel");

janela.setSize(350, 150);

janela.setLocationRelativeTo(null);

// janela.setLocation(50, 50);

janela.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

JPanel painel = new JPanel();

janela.add(painel);

janela.setVisible(true);

}

}

**UsaJRadioButton.java**

import javax.swing.JFrame;

import javax.swing.JRadioButton;

import java.awt.FlowLayout;

public class UsaJRadioButton {

public static void main(String[] args) {

JFrame janela = new JFrame();

janela.setTitle("JRadioButton");

janela.setSize(350,150);

janela.setLocation(50,50);

janela.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

janela.setLayout(new FlowLayout());

JRadioButton botaoRadio1 = new JRadioButton("Branco");

JRadioButton botaoRadio2 = new JRadioButton("Preto");

JRadioButton botaoRadio3 = new JRadioButton("Amarelo");

janela.add(botaoRadio1);

janela.add(botaoRadio2);

janela.add(botaoRadio3);

janela.setVisible(true);

}

}

**UsaJTextField.java**

//

// Copyright (c) Carlos Tojal 2020

// Container\_Component

// UsaJTextField.java

//

import javax.swing.JFrame;

import javax.swing.JTextField;

import javax.swing.JLabel;

import java.awt.FlowLayout;

public class UsaJTextField {

public static void main(String[] args) {

JFrame janela = new JFrame();

janela.setTitle("JTextField");

janela.setSize(350, 150);

janela.setLocationRelativeTo(null);

// janela.setLocation(50, 50);

janela.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

janela.setLayout(new FlowLayout());

JLabel rotulo = new JLabel("Escreva o seu nome: ");

JTextField caixaTexto = new JTextField(10);

janela.add(rotulo);

janela.add(caixaTexto);

janela.setVisible(true);

}

}

**UtilizaBorderLayout.java**

import java.awt.BorderLayout;

import javax.swing.JFrame;

import javax.swing.JButton;

public class UtilizaBorderLayout {

public static void main(String[] args) {

JFrame janela = new JFrame();

janela.setTitle("BorderLayout");

janela.setSize(350,150);

janela.setLocation(50,50);

janela.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

janela.setLayout(new BorderLayout());

JButton botaoNorte = new JButton("Norte");

JButton botaoSul = new JButton("Sul");

JButton botaoOeste = new JButton("Oeste");

JButton botaoEste = new JButton("Este");

JButton botaoCentro = new JButton("Centro");

janela.add("North", botaoNorte);

janela.add("South", botaoSul);

janela.add("West", botaoOeste);

janela.add("East", botaoEste);

janela.add("Center", botaoCentro);

janela.setVisible(true);

}

}

**UtilizaCardLayout.java**

import java.awt.BorderLayout;

import java.awt.GridLayout;

import java.awt.CardLayout;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.JLabel;

import javax.swing.JButton;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class UtilizaCardLayout implements ActionListener {

JFrame janela = new JFrame();

JPanel painelBotoes = new JPanel();

JButton botao1 = new JButton("Painel 1");

JButton botao2 = new JButton("Painel 2");

JPanel painelRotulo1 = new JPanel();

JPanel painelRotulo2 = new JPanel();

JLabel rotulo1 = new JLabel("Painel 1");

JLabel rotulo2 = new JLabel("Painel 2");

JPanel painelRotulos = new JPanel();

public static void main(String[] args) {

new UtilizaCardLayout();

}

private UtilizaCardLayout() {

janela.setTitle("Primeira aplicação gráfica - CardLayout");

janela.setSize(350, 150);

janela.setLocation(50, 50);

janela.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

janela.setLayout(new BorderLayout());

painelBotoes.setLayout(new GridLayout(1, 2));

painelBotoes.add(botao1);

painelBotoes.add(botao2);

painelRotulo1.add(rotulo1);

painelRotulo2.add(rotulo2);

painelRotulos.setLayout(new CardLayout());

painelRotulos.add(painelRotulo1, "p1");

painelRotulos.add(painelRotulo2, "p2");

janela.add("North", painelBotoes);

janela.add("South", painelRotulos);

botao1.addActionListener(this);

botao2.addActionListener(this);

janela.setVisible(true);

}

public void actionPerformed(ActionEvent e) {

CardLayout cl = (CardLayout) painelRotulos.getLayout();

if (e.getSource() == botao1)

cl.show(painelRotulos, "p1");

if (e.getSource() == botao2)

cl.show(painelRotulos, "p2");

}

}

**UtilizaFlowLayout.java**

import java.awt.FlowLayout;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JTextField;

import javax.swing.JButton;

public class UtilizaFlowLayout {

public static void main(String[] args) {

JFrame janela = new JFrame();

janela.setTitle("FlowLayout");

janela.setSize(350,150);

janela.setLocation(50,50);

janela.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

janela.setLayout(new FlowLayout());

JLabel rotulo = new JLabel("Escreva o seu nome: ");

JTextField caixaTexto = new JTextField(10);

JButton botao = new JButton("OK");

janela.add(rotulo);

janela.add(caixaTexto);

janela.add(botao);

janela.setVisible(true);

}

}

**UtilizaGridLayout.java**

import java.awt.GridLayout;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JTextField;

import javax.swing.JComboBox;

import javax.swing.JButton;

public class UtilizaGridLayout {

public static void main(String[] args) {

JFrame janela = new JFrame();

janela.setTitle("GridLayout");

janela.setSize(350,150);

janela.setLocation(50,50);

janela.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

janela.setLayout(new GridLayout(3,2));

JLabel rotulo1 = new JLabel("Seleccione um fruto: ");

JTextField caixaTexto = new JTextField(10);

JLabel rotulo2 = new JLabel ("Seleccione uma bebida: ");

JComboBox<String> caixaCombinacao = new JComboBox<String>();

caixaCombinacao.addItem("Água");

caixaCombinacao.addItem("Leite");

caixaCombinacao.addItem("Vinho");

JButton botao = new JButton("OK");

janela.add(rotulo1);

janela.add(caixaTexto);

janela.add(rotulo2);

janela.add(caixaCombinacao);

janela.add(botao);

janela.setVisible(true);

}

}

Exemplos

DivZero.java

//

// Copyright (c) Carlos Tojal 2019

// Exemplo1

// DivZero.java

//

public class DivZero {

public static void main(String[] args) {

int a = 1, b = 0;

System.out.println(a/b);

System.out.println("Divisão por zero");

System.out.println("Fim do programa");

}

}

/\*

public class DivZero {

public static void main(String[] args) {

int a = 1, b = 0;

try {

System.out.println(a/b);

} catch(Exception e) {

System.out.println("Divisão por zero");

}

System.out.println("Fim do programa");

}

}\*/

**ExcepcaoEscalada.java**

//

// Copyright (c) Carlos Tojal 2019

// Exemplo1

// ExcepcaoEscalada.java

//

public class ExcepcaoEscalada {

public static void metodoComErro() {

System.out.println(3 / 0);

}

public static void main(String[] args) {

try {

metodoComErro();

} catch(RuntimeException rte) {

rte.printStackTrace();

}

System.out.println("Fim do Programa");

}

}

**ExcepcaoEscalada2.java**

//

// Copyright (c) Carlos Tojal 2019

// Exemplo1

// ExcepcaoEscalada2.java

//

public class ExcepcaoEscalada2 {

public static void main(String[] args) throws java.io.IOException {

int n = 0;

n = System.in.read();

System.out.println("n=='a'" + (n=='a'));

System.out.println("n=" + n);

}

}

**ExcepcaoEscalada2a.java**

//

// Copyright (c) Carlos Tojal 2019

// Exemplo1

// ExcepcaoEscalada2a.java

//

public class ExcepcaoEscalada2a {

public static void main(String[] args) {

int n = 0;

try {

n = System.in.read();

} catch(Exception e) {

e.printStackTrace();

} finally {

System.out.println("n=='a'" + (n=='a'));

System.out.println("n=" + n);

}

}

}

**Exemplo1.java**

//

// Copyright (c) Carlos Tojal 2020

// Exemplo1

//

import java.util.Iterator;

import java.util.LinkedList;

import java.util.List;

// Iterator - Interface do pacote java.util.

// Permite percorrer as coleções do framework Collections que implementam a interface "Collection". Fornece os métodos next(), hasNext() e ermove().

public class Exemplo1 {

public static void main(String[] args) {

List<String> list = new LinkedList<>();

list.add("Welcome");

list.add("to");

list.add("our");

list.add("city");

System.out.println("The list is given as: " + list);

Iterator<String> itr = list.iterator();

while(itr.hasNext()) {

System.out.println(itr.next());

}

itr.remove();

System.out.println("After the remove() method is called: " + list);

}

}

**Exemplo2.java**

//

// Copyright (c) Carlos Tojal 2020

// Exemplo2

//

import java.util.ArrayList;

import java.util.Iterator;

import java.util.List;

// For-Each consiste num ciclo for adaptado a collections.

// Percorre todos os elementos de qualquer collection do framework "Collection".

public class Exemplo2 {

public static void main(String[] args) {

List<String> listaDeNomes = new ArrayList<String>();

listaDeNomes.add("Gustavo");

listaDeNomes.add("Maria");

listaDeNomes.add("José");

listaDeNomes.add("João");

listaDeNomes.add("Ana");

Iterator<String> iteratorDeNomes = listaDeNomes.iterator();

List<String> listaConvertidaDoIterator = new ArrayList<>();

while(iteratorDeNomes.hasNext()) {

listaConvertidaDoIterator.add(iteratorDeNomes.next());

}

listaConvertidaDoIterator.forEach(System.out::println);

}

}

**Exemplo2a.java**

//

// Copyright (c) Carlos Tojal 2020

// Exemplo3

//

import java.util.ArrayList;

import java.util.Iterator;

import java.util.List;

public class Exemplo2a {

public static void main(String[] args) {

List<String> listaDeNomes = new ArrayList<String>();

listaDeNomes.add("Gustavo");

listaDeNomes.add("Maria");

listaDeNomes.add("José");

listaDeNomes.add("João");

listaDeNomes.add("Ana");

Iterator<String> iteratorDeNomes = listaDeNomes.iterator();

List<String> listaConvertidaDoIterator = new ArrayList<>();

iteratorDeNomes.forEachRemaining(n -> listaConvertidaDoIterator.add(n));

iteratorDeNomes.forEachRemaining(listaConvertidaDoIterator::add);

listaConvertidaDoIterator.forEach(System.out::println);

}

}

**Exemplo3.java**

//

// Copyright (c) Carlos Tojal 2020

// Exemplo3

//

import java.util.ArrayList;

public class Exemplo3 {

public static void main(String[] args) {

ArrayList<String> books = new ArrayList<String>();

books.add("C");

books.add("Java");

books.add("PHP");

for(String obj: books) {

System.out.println(obj);

books.add("C++");

}

}

}

**TesteCatch.java**

//

// Copyright (c) Carlos Tojal 2019

// Exemplo1

// TesteCatch.java

//

public class TesteCatch {

public static void main(String[] args) {

int a;

try {

a = Integer.parseInt("123a");

} catch(NumberFormatException nfe) {

System.out.println("NumberFormatException");

} catch(ArithmeticException ae) {

System.out.println("ArithmeticExcetion");

} catch(ArrayIndexOutOfBoundsException aioobe) {

System.out.println("ArrayIndexOutOfBoundsException");

} catch(Exception e) {

System.out.println("Bloco Exception");

} finally {

System.out.println("Bloco Finally");

}

System.out.println("Fim do programa");

}

}

**TesteCatch1.java**

//

// Copyright (c) Carlos Tojal 2019

// Exemplo1

// TesteCatch1.java

//

public class TesteCatch1 {

public static void main(String[] args) {

int[] a = new int[6];

try {

a[8] = 12;

} catch(NumberFormatException nfe) {

System.out.println("NumberFormatException");

} catch(ArithmeticException ae) {

System.out.println("ArithmeticExcetion");

} catch(ArrayIndexOutOfBoundsException aioobe) {

System.out.println("ArrayIndexOutOfBoundsException");

} catch(Exception e) {

System.out.println("Bloco Exception");

} finally {

System.out.println("Bloco Finally");

}

System.out.println("Fim do programa");

}

}

ExemplosGUI

DisplayImage.java

//

// Copyright (c) Carlos Tojal 2020

// ExemplosGUI

// DisplayImage.java

//

import java.awt.image.BufferedImage;

import java.io.File;

import java.io.IOException;

import javax.imageio.ImageIO;

import javax.swing.ImageIcon;

import javax.swing.JFrame;

import javax.swing.JLabel;

public class DisplayImage {

static String IMG\_PATH = "Imagem.jpg";

public static void main(String[] args) throws IOException {

JFrame frame = new JFrame();

BufferedImage img = ImageIO.read(new File(IMG\_PATH));

ImageIcon icon = new ImageIcon(img);

JLabel label = new JLabel(icon);

frame.add(label);

frame.setExtendedState(frame.getExtendedState() | JFrame.MAXIMIZED\_BOTH);

frame.setVisible(true);

}

}

**ExtendedFrame.java**

//

// Copyright (c) Carlos Tojal 2020

// ExemplosGUI

// ExtendedFrame.java

//

import java.awt.EventQueue;

import javax.swing.JFrame;

import javax.swing.UIManager;

import javax.swing.UnsupportedLookAndFeelException;;

public class ExtendedFrame {

public static void main(String[] args) {

new ExtendedFrame();

}

public ExtendedFrame() {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());

} catch(ClassNotFoundException | InstantiationException | IllegalAccessException | UnsupportedLookAndFeelException e) {

}

JFrame frame = new JFrame();

frame.setExtendedState(JFrame.MAXIMIZED\_BOTH);

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setVisible(true);

}

});

}

}

FT3

ConversorTemperatura.java

//

// Copyright (c) Carlos Tojal 2020

// FT3

// ConversorTemperatura.java

//

import java.awt.GridLayout;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JRadioButton;

import javax.swing.ButtonGroup;

import javax.swing.JPanel;

import javax.swing.JTextField;

import javax.swing.JButton;

import javax.swing.JOptionPane;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

public class ConversorTemperatura implements ActionListener {

JFrame janela = new JFrame();

JLabel rotulo1 = new JLabel("Tipo de conversão: ");

JRadioButton botaoRadio1 = new JRadioButton("ºC->ºF");

JRadioButton botaoRadio2 = new JRadioButton("ºF->ºC");

ButtonGroup grupo = new ButtonGroup();

JPanel painel = new JPanel();

JLabel rotulo2 = new JLabel("Valor a converter: ");

JTextField caixaTexto1 = new JTextField(5);

JButton botao1 = new JButton("Converter");

private ConversorTemperatura() {

janela.setTitle("Conversor Temperatura");

janela.setSize(300, 125);

janela.setLocation(50, 50);

janela.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

janela.setLayout(new GridLayout(3, 2));

janela.add(rotulo1);

painel.setLayout(new GridLayout(1, 2));

painel.add(botaoRadio1);

painel.add(botaoRadio2);

janela.add(painel);

janela.add(rotulo2);

janela.add(caixaTexto1);

janela.add(botao1);

grupo.add(botaoRadio1);

grupo.add(botaoRadio2);

botao1.addActionListener(this);

janela.setVisible(true);

}

public static void main(String[] args) {

new ConversorTemperatura();

}

public void actionPerformed(ActionEvent e) {

double resultado;

String mensagem = "";

if(e.getSource() == botao1) {

if(botaoRadio1.isSelected()) {

resultado = Double.parseDouble(caixaTexto1.getText()) \* 9 / 5 + 32;

mensagem = Double.toString(resultado) + "ºF";

} else if(botaoRadio2.isSelected()) {

resultado = Double.parseDouble(caixaTexto1.getText()) - 32 \* 5 / 9;

mensagem = Double.toString(resultado) + "ºC";

} else {

mensagem = "Não selecionou nenhuma opção!";

}

}

JOptionPane.showMessageDialog(null, mensagem);

}

}

**ConversorTemperaturaVersaoAvancada.java**

//

// Copyright (c) Carlos Tojal 2020

// FT3

// ConversorTemperaturaVersaoAvancada.java

//

import java.awt.GridLayout;

import java.awt.FlowLayout;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JRadioButton;

import javax.swing.ButtonGroup;

import javax.swing.JPanel;

import javax.swing.JTextField;

import javax.swing.JButton;

import javax.swing.JOptionPane;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

import java.awt.event.ItemListener;

import java.awt.event.ItemEvent;

public class ConversorTemperaturaVersaoAvancada implements ActionListener, ItemListener {

JFrame janela = new JFrame();

JLabel rotulo1 = new JLabel("Tipo de conversão: ");

JRadioButton botaoRadio1 = new JRadioButton("ºC->ºF");

JRadioButton botaoRadio2 = new JRadioButton("ºF->ºC");

ButtonGroup grupo = new ButtonGroup();

JPanel painel = new JPanel();

JLabel rotulo2 = new JLabel("Valor a converter: ");

JTextField caixaTexto1 = new JTextField(5);

JButton botao1 = new JButton("Converter");

private ConversorTemperaturaVersaoAvancada() {

janela.setTitle("Conversor Temperatura");

janela.setSize(300, 125);

janela.setLocation(50, 50);

janela.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

janela.setLayout(new GridLayout(3, 2));

janela.add(rotulo1);

painel.setLayout(new GridLayout(1, 2));

painel.add(botaoRadio1);

painel.add(botaoRadio2);

janela.add(painel);

janela.add(rotulo2);

janela.add(caixaTexto1);

janela.add(botao1);

grupo.add(botaoRadio1);

grupo.add(botaoRadio2);

botaoRadio1.addItemListener(this);

botaoRadio2.addItemListener(this);

botao1.addActionListener(this);

janela.setVisible(true);

}

public static void main(String[] args) {

new ConversorTemperaturaVersaoAvancada();

}

public void actionPerformed(ActionEvent e) {

double resultado;

String mensagem = "";

if(e.getSource() == botao1) {

if(botaoRadio1.isSelected()) {

resultado = Double.parseDouble(caixaTexto1.getText()) \* 9 / 5 + 32;

mensagem = Double.toString(resultado) + "ºF";

} else if(botaoRadio2.isSelected()) {

resultado = Double.parseDouble(caixaTexto1.getText()) - 32 \* 5 / 9;

mensagem = Double.toString(resultado) + "ºC";

} else {

mensagem = "Não selecionou nenhuma opção!";

}

}

JOptionPane.showMessageDialog(null, mensagem);

}

public void itemStateChanged(ItemEvent e) {

if(e.getSource() == botaoRadio1) {

if(e.getStateChange() == ItemEvent.SELECTED)

JOptionPane.showMessageDialog(null, "ºC->ºF");

}

if(e.getSource() == botaoRadio2) {

if(e.getStateChange() == ItemEvent.SELECTED)

JOptionPane.showMessageDialog(null, "ºF->ºC");

}

}

}

**UtilizaBufferedInputStream.java**

//

// Copyright (c) Carlos Tojal 2020

// FT3

// UtilizaBufferedInputOutputStream1.java

//

import java.io.FileInputStream;

import java.io.FileOutputStream;

import java.io.BufferedInputStream;

import java.io.BufferedOutputStream;

import java.io.FileNotFoundException;

import java.io.IOException;

public class UtilizaBufferedInputOutputStream1 {

static FileInputStream fis;

static FileOutputStream fos;

static BufferedInputStream bis;

static BufferedOutputStream bos;

static int conteudo;

public static void main(String[] args) {

try {

fis = new FileInputStream("Imagem1.jpg");

bis = new BufferedInputStream(fis);

fos = new FileOutputStream("Imagem3.jpg");

bos = new BufferedOutputStream(fos);

while((conteudo=bis.read()) != -1)

bos.write(conteudo);

fis.close();

bis.close();

fos.close();

bos.close();

} catch(FileNotFoundException fnfe) {

System.out.println("Ficheiro não encontrado.");

} catch(IOException ioe) {

System.out.println("Não foi possível ler o ficheiro.");

}

}

}

**UtilizaFileInputOutputStream1.java**

//

// Copyright (c) Carlos Tojal 2020

// FT3

// UtilizaFileOutputStream1.java

//

import java.io.FileInputStream;

import java.io.FileOutputStream;

import java.io.FileNotFoundException;

import java.io.IOException;

public class UtilizaFileInputOutputStream1 {

static FileInputStream fis;

static FileOutputStream fos;

static int conteudo;

public static void main(String[] args) {

try {

fis = new FileInputStream("Inteiros1.txt");

fos = new FileOutputStream("Inteiros2.txt");

while((conteudo=fis.read()) != -1)

fos.write(conteudo);

fis.close();

fos.close();

} catch(FileNotFoundException fnfe) {

System.out.println("Ficheiro não encontrado.");

} catch(IOException ioe) {

System.out.println("Não foi possível ler o ficheiro.");

}

}

}

**UtilizaBufferedInputOutputStream2.java**

//

// Copyright (c) Carlos Tojal 2020

// FT3

// UtilizaFileOutputStream2.java

//

import java.io.FileInputStream;

import java.io.FileOutputStream;

import java.io.FileNotFoundException;

import java.io.IOException;

public class UtilizaFileInputOutputStream2 {

static FileInputStream fis;

static FileOutputStream fos;

static int conteudo;

public static void main(String[] args) {

try {

fis = new FileInputStream("Caracteres1.txt");

fos = new FileOutputStream("Caracteres2.txt");

while((conteudo=fis.read()) != -1)

fos.write(conteudo);

fis.close();

fos.close();

} catch(FileNotFoundException fnfe) {

System.out.println("Ficheiro não encontrado.");

} catch(IOException ioe) {

System.out.println("Não foi possível ler o ficheiro.");

}

}

}

**UtilizaBufferedInputOutputStream3.java**

//

// Copyright (c) Carlos Tojal 2020

// FT3

// UtilizaFileOutputStream3.java

//

import java.io.FileInputStream;

import java.io.FileOutputStream;

import java.io.FileNotFoundException;

import java.io.IOException;

public class UtilizaFileInputOutputStream3 {

static FileInputStream fis;

static FileOutputStream fos;

static int conteudo;

public static void main(String[] args) {

try {

fis = new FileInputStream("Imagem1.jpg");

fos = new FileOutputStream("Imagem2.jpg");

while((conteudo=fis.read()) != -1)

fos.write(conteudo);

fis.close();

fos.close();

} catch(FileNotFoundException fnfe) {

System.out.println("Ficheiro não encontrado.");

} catch(IOException ioe) {

System.out.println("Não foi possível ler o ficheiro.");

}

}

}

FT4

**UtilizaDataInputOutputStream.java**

//

// Copyright (c) Carlos Tojal 2020

// FT4

// UtilizaDataInputOutputStream1.java

//

import java.io.FileInputStream;

import java.io.FileOutputStream;

import java.io.DataInputStream;

import java.io.DataOutputStream;

import java.io.FileNotFoundException;

import java.io.IOException;

public class UtilizaDataInputOutputStream1 {

static FileOutputStream fos;

static DataOutputStream dos;

public static void main(String[] args) {

float j = 0.5f;

try {

fos = new FileOutputStream("Floats.txt");

dos = new DataOutputStream(fos);

for(int i = 0; i < 3; i++) {

System.out.println(i + j);

dos.writeFloat(j + i);

}

} catch(FileNotFoundException fnfe) {

System.out.println("Ficheiro não encontrado.");

} catch(IOException ioe) {

System.out.println("Não foi possível ler o ficheiro.");

}

}

}

FT5

UtilizaArray.java

//

// Copyright (c) Carlos Tojal 2020

// FT5

// UtilizaArray.java

//

import java.util.ArrayList;

import java.lang.IndexOutOfBoundsException;

public class UtilizaArray {

public static void main(String[] args) {

ArrayList<Integer> arr = new ArrayList<Integer>();

arr.add(Integer.valueOf(2));

arr.add(4);

arr.add(6);

arr.add(8);

arr.add(10);

// System.out.println(arr.get(2));

try {

System.out.println(arr.get(5));

} catch(IndexOutOfBoundsException ioobe) {

System.out.println("O indice passado como parametro num dos metodos que esta a utilizar esta fora dos limites");

}

}

}

FT6

**UtilizaArray.java**

//

// Copyright (c) Carlos Tojal 2020

// FT5

// UtilizaArray.java

//

import java.util.ArrayList;

import java.lang.IndexOutOfBoundsException;

public class UtilizaArray {

public static void main(String[] args) {

ArrayList<Integer> arr = new ArrayList<Integer>();

arr.add(Integer.valueOf(2));

arr.add(4);

arr.add(6);

arr.add(8);

arr.add(10);

// System.out.println(arr.get(2));

try {

System.out.println(arr.get(5));

} catch(IndexOutOfBoundsException ioobe) {

System.out.println("O indice passado como parametro num dos metodos que esta a utilizar esta fora dos limites");

}

}

}

FT6

**UtilizaSequenceInputOutputStream.java**

//

// Copyright (c) Carlos Tojal 2020

// FT6

// UtilizaSequenceInputStream.java

//

import java.io.File;

import java.io.InputStream;

import java.io.FileInputStream;

import java.io.SequenceInputStream;

import java.io.FileWriter;

import java.io.BufferedWriter;

import java.io.IOException;

import java.io.FileNotFoundException;

import java.lang.Exception;

public class UtilizaSequenceInputStream {

public static void main(String[] args) {

File f;

InputStream is1;

InputStream is2;

SequenceInputStream sis;

FileWriter fw;

BufferedWriter bw;

double num = 0;

try {

is1 = new FileInputStream("Inteiros1.txt");

is2 = new FileInputStream("Inteiros2.txt");

sis = new SequenceInputStream(is1, is2);

num = sis.read();

sis.close();

} catch (IOException ioe) {

System.out.println("Erro na leitura.");

} catch (Exception e) {

System.out.println("Erro.");

}

try {

f = new File("Inteiros1e2.txt");

fw = new FileWriter(f);

bw = new BufferedWriter(fw);

bw.write(String.valueOf(num));

bw.close();

} catch (IOException ioe) {

System.out.println("Erro na escrita.");

} catch (Exception e) {

System.out.println("Erro.");

}

}

}

FT7

UtilizaFileReaderWriter.java

//

// Copyright (c) Carlos Tojal 2020

// FT7

// UtilizaFileReaderWriter.java

//

import java.io.Reader;

import java.io.FileReader;

import java.io.BufferedReader;

import java.io.Writer;

import java.io.FileWriter;

import java.io.BufferedWriter;

import java.io.IOException;

import java.lang.Exception;

public class UtilizaFileReaderWriter {

public static void main(String[] args) {

Writer w;

BufferedWriter bw;

Reader r;

BufferedReader br;

try {

w = new FileWriter("Strings.txt");

bw = new BufferedWriter(w);

for(int i = 0; i < 5; i++)

bw.write(String.valueOf(i+1) + " linha\n");

bw.close();

} catch(IOException ioe) {

System.out.println("Erro na escrita do ficheiro.");

} catch(Exception e) {

System.out.println("Erro.");

}

try {

String line;

r = new FileReader("Strings.txt");

br = new BufferedReader(r);

while((line = br.readLine()) != null)

System.out.println(line);

} catch(IOException ioe) {

System.out.println("Erro na leitura do ficheiro.");

} catch(Exception e) {

System.out.println("Erro.");

}

}

}

FT8

UtilizaRandomAccessFile.java

//

// Copyright (c) Carlos Tojal 2020

// FT8

// UtilizaRandomAccessFile.java

//

import java.io.File;

import java.io.RandomAccessFile;

import java.io.IOException;

import java.lang.Exception;

public class UtilizaRandomAccessFile {

public static void main(String[] args) {

int [] arr = new int[] {1, 2, 3, 4, 5, 6, 7, 8};

try {

File f = new File("numeros.txt");

RandomAccessFile raf = new RandomAccessFile(f, "rw");

for (int i = 0; i < 8; i++)

raf.write(arr[i]);

System.out.println("Posicao no ficheiro: " + raf.getFilePointer() + " bytes");

raf.seek(16);

raf.seek(20);

System.out.println(raf.read());

raf.close();

} catch(IOException ioe) {

System.out.println("Erro na escrita do ficheiro.");

} catch(Exception e) {

System.out.println("Erro.");

}

}

}

FT9

UtilizaFile1.java

//

// Copyright (c) Carlos Tojal 2020

// FT9

// UtilizaFile1.java

//

import java.io.File;

import java.io.IOException;

import java.lang.Exception;

public class UtilizaFile1 {

public static void main(String[] args) {

File file = new File(".." + File.separatorChar + "FT9" + File.separatorChar + "UtilizaFile1.java");

if(file.exists()) {

System.out.println("Nome: " + file.getName());

System.out.println("Diretorio: " + file.getParent());

System.out.println("Diretorio completo: " + file.getAbsoluteFile().getParent());

} else {

System.out.println("O ficheiro especificado nao existe");

}

}

}

**UtilizaFile2.java**

//

// Copyright (c) Carlos Tojal 2020

// FT9

// UtilizaFile2.java

//

import java.io.File;

import java.io.IOException;

import java.lang.Exception;

public class UtilizaFile2 {

public static void main(String[] args) {

try {

File f1 = new File("Inteiros.txt");

File f2, f3, f4, f5;

if (f1.exists()) {

f2 = new File("NumsInteiros.txt");

f1.renameTo(f2);

f1.delete();

}

f3 = new File("LingProg");

f3.mkdir();

if (f3.exists()) {

f4 = new File("LingProg" + File.separatorChar + "Modulo11.txt");

f5 = new File("LingProg" + File.separatorChar + "Modulo12.txt");

f4.createNewFile();

f5.createNewFile();

}

String[] list = f3.list();

for(String filename: list)

System.out.println(filename);

} catch(IOException ioe) {

System.out.println("Erro no acesso aos diretorios e/ou ficheiros.");

} catch(Exception e) {

System.out.println("Erro.");

}

}

}

SistemaBancario

**Banco.java**

//

// Copyright (c) Carlos Tojal 2020

// SistemaBancario

// Banco.java

//

public class Banco

{

private int senha;

private double saldo,levantamento,cpmf,limite;

private String nome,sobrenome, genero;

public Banco(String nom, String sobre)

{

this.nome = nome;

this.sobrenome= sobre;

}

public Banco(int sen)

{

this.senha = sen;

}

public Banco(double sald, double lev, double cp, double lm)

{

if (sald > 0.00 && lev > 0.00)

this.saldo = sald;

this.levantamento = lev;

this.cpmf = cp;

this.limite = lm;

}

public void setNome(String n)

{

this.nome = n;

}

public void setSobre(String s)

{

this.sobrenome = s;

}

public void setGenero(String g) {

this.genero = g;

}

public void setSaldo(double sa)

{

this.saldo = sa;

}

public void setLevantamento(double lev)

{

this.levantamento = lev;

}

public void setCpmf(double cm)

{

this.cpmf = cm;

}

public void setLimite(double lt)

{

this.limite = lt;

}

public void setSenha(int sh)

{

this.senha = sh;

}

//exibe o valor armazenado no set

public String getNome()

{

return nome;

}

public String getSobre()

{

return sobrenome;

}

public String getGenero() {

return genero;

}

public double getSald()

{

return saldo;

}

public double getLevantamento()

{

return levantamento;

}

public double getCpmf()

{

return cpmf;

}

public double getLimite()

{

return limite;

}

public int getSenha()

{

return senha;

}

}

**SistemaBancario.java**

import javax.swing.JOptionPane;

import java.lang.Exception;

import java.lang.NumberFormatException;

//

// Copyright (c) Carlos Tojal 2020

// SistemaBancario

// SistemaBancario.java

//

public class SistemaBancario

{

public static void main(String args[])

{

Banco objtlimite = new Banco(0.00, 1.00, 5.00, 0.50);

Banco objtnome = new Banco("a", "b");

Banco objtsobre = new Banco("c", "d");

Banco objtsaldo = new Banco(0.00, 1.00, 5.00, 0.50);

Banco objtlevantamento = new Banco(0.00, 1.00, 5.00, 0.50);

Banco objtcpmf = new Banco(0.00, 1.00, 5.00, 0.50);

Banco obtjsenha = new Banco("456", "8521");

String name = JOptionPane.showInputDialog("Digite o seu nome");

objtnome.setNome(name);

String sobre = JOptionPane.showInputDialog("Digite o seu sobrenome");

objtsobre.setSobre(sobre);

String genero;

do {

genero = JOptionPane.showInputDialog("Digite o seu genero (masculino/feminino)");

objtsobre.setGenero(genero);

if(!genero.equals("masculino") && !genero.equals("feminino"))

JOptionPane.showMessageDialog(null, "Genero indisponível.", "Genero indisponível", JOptionPane.ERROR\_MESSAGE);

} while(!genero.equals("masculino") && !genero.equals("feminino"));

String senha = JOptionPane.showInputDialog("Digite sua senha para acesso");

int se = Integer.parseInt(senha);

obtjsenha.setSenha(se);

String textoAviso = "Dados registados com sucesso! ";

if(genero.equals("masculino"))

textoAviso += "Sr. ";

else

textoAviso += "Sra. ";

textoAviso += name;

String aviso = String.format(textoAviso);

String aviso2 = String.format("ID: %s\nSenha: %s ", name, senha);

JOptionPane.showMessageDialog(null, aviso);

JOptionPane.showMessageDialog(null, aviso2);

//valida dados

String id = JOptionPane.showInputDialog("Digite sua id");

String ss = JOptionPane.showInputDialog("Digite sua senha");

int sh = Integer.parseInt(ss);

if (id != name && se != sh){

JOptionPane.showMessageDialog(null, "Dados Inválidos");

}

else

{

String deposito;

boolean numeroInvalido = false;

double dep = 0.00;

do {

deposito = JOptionPane.showInputDialog("Faca um deposito na conta");

try {

dep = Double.parseDouble(deposito);

} catch (NumberFormatException nfe) {

JOptionPane.showMessageDialog(null, "Valor invalido.", "Erro", JOptionPane.ERROR\_MESSAGE);

numeroInvalido = true;

}

}while(numeroInvalido);

if (dep < 20.00)

{

JOptionPane.showMessageDialog(null, "De acordo com o contrato valor invalido para deposito");

}

else {

JOptionPane.showMessageDialog(null, "Obrigado por realizar um deposito");

dep = dep;

objtsaldo.setSaldo(dep);

double limit;

limit = dep \* (2);

objtlimite.setLimite(limit);

String avisolimite = String.format("Seu limite para emprestimos e: E%.2f", objtlimite.getLimite() );

JOptionPane.showMessageDialog(null, avisolimite);

String levantamento;

numeroInvalido = false;

double lev = 0.00;

do {

levantamento = JOptionPane.showInputDialog("Realize um levantamento ");

try {

lev = Double.parseDouble(levantamento);

} catch (NumberFormatException nfe) {

JOptionPane.showMessageDialog(null, "Valor invalido.", "Erro", JOptionPane.ERROR\_MESSAGE);

numeroInvalido = true;

}

}while(numeroInvalido);

if (lev > dep) {

JOptionPane.showMessageDialog(null, "Saldo Insuficiente");

}

else {

lev = lev;

objtlevantamento.setLevantamento(lev);

double r;

r = dep - lev;

double cpm = (lev \* 2) / 100;

objtcpmf.setCpmf(cpm);

double set = r - cpm;

objtsaldo.setSaldo(set);

limit = set \* (2);

objtlimite.setLimite(limit);

int cont = 785236;

String exibi = String.format("Numero da Conta %s\nSaldo E%.2f\nLevantamento realizado E%.2f\nLimite para emprestimo E%.2f\nValor de CPMF E%.2f\n\nSistema Desenvolvido por Carlos Tojal Version 1.1", cont, objtsaldo.getSald(), objtlevantamento.getLevantamento(), objtlimite.getLimite(), objtcpmf.getCpmf());

JOptionPane.showMessageDialog(null, exibi);

}

}

}

}

}

Streams

LeituraCaracteres.java

//

// Copyright (c) Carlos Tojal 2020

// Streams

// LeituraCaracteres.java

//

import java.io.FileReader;

import java.io.FileWriter;

import java.io.BufferedReader;

import java.io.BufferedWriter;

import java.io.IOException;

public class LeituraCaracteres {

public static void main(String[] args) {

try {

FileWriter fw = new FileWriter("Strings.txt");

BufferedWriter bw = new BufferedWriter(fw);

for(int i = 1; i < 10; i++) {

bw.write(i + "a linha");

bw.newLine();

}

bw.close();

/\* Neste caso, o FileReader não pode ser utilizado sozinho porque

nenhum dos seus métodos read devolve Strings \*/

FileReader fr = new FileReader("Strings.txt");

BufferedReader br = new BufferedReader(fr);

while(br.ready())

System.out.print((char)br.read());

br.close();

} catch(IOException e) {

System.out.println("Erro");

}

}

}

Projeto pessoal adicional

bank

AccountManagement.java

//

// Copyright (c) Carlos Tojal 2020

// Bank

// AccountManagement.java

//

package management;

import java.util.ArrayList;

import java.io.File;

import java.io.FileReader;

import java.io.FileWriter;

import java.io.BufferedReader;

import java.io.BufferedWriter;

import java.io.FileNotFoundException;

import java.io.IOException;

import java.lang.Exception;

import structures.Client;

import structures.Account;

import structures.AccountMovement;

public class AccountManagement {

// Constructor

public AccountManagement() {

}

// Methods

public void loadAccountMovements(Account account) {

ArrayList<AccountMovement> movements = new ArrayList<AccountMovement>();

String raw;

AccountMovement movement = new AccountMovement();

File f;

FileReader fr;

BufferedReader br;

// Loads client accounts

try {

f = new File("account\_movements.csv");

fr = new FileReader(f);

br = new BufferedReader(fr);

while ((raw = br.readLine()) != null) {

if (account.getId().equals(raw.split(";")[0])) {

String account\_id = raw.split(";")[0];

double value = Double.parseDouble(raw.split(";")[1]);

byte type = (byte) Integer.parseInt(raw.split(";")[2]);

account.getAccountMovements().add(new AccountMovement(account\_id, value, type));

account.setAccountMovements(account.getAccountMovements());

if(type == 1)

account.deposit(value);

else

account.withdraw(value);

}

}

fr.close();

br.close();

} catch(FileNotFoundException e) {

System.out.println("[ERROR] File \"accounts.csv\" was not found.");

// e.printStackTrace();

} catch(IOException e) {

System.out.println("[ERROR] Couldn't load file \"accounts.csv\".");

// e.printStackTrace();

} catch(Exception e) {

System.out.println("[ERROR] An error has occurred.");

// e.printStackTrace();

}

}

public void registerAccountMovement(AccountMovement accountMovement) {

File f;

FileWriter fw;

BufferedWriter bw;

try {

fw = new FileWriter("account\_movements.csv", true);

bw = new BufferedWriter(fw);

bw.newLine();

bw.write(accountMovement.getAccount\_id() + ";" + accountMovement.getValue() + ";" + accountMovement.getType());

bw.close();

} catch(FileNotFoundException e) {

System.out.println("[ERROR] File \"account\_movements.csv\" was not found.");

// e.printStackTrace();

} catch(IOException e) {

System.out.println("[ERROR] Couldn't load file \"account\_movements.csv\".");

// e.printStackTrace();

} catch(Exception e) {

System.out.println("[ERROR] An error has occurred.");

// e.printStackTrace();

}

}

public void loadAccounts(ArrayList<Client> clients) {

ArrayList<Account> accounts = new ArrayList<Account>();

String raw;

Account account = new Account();

File f;

FileReader fr;

BufferedReader br;

// Loads client accounts

try {

f = new File("accounts.csv");

fr = new FileReader(f);

br = new BufferedReader(fr);

while ((raw = br.readLine()) != null) {

for (int i = 0; i < clients.size(); i++) {

if (clients.get(i).getId().equals(raw.split(";")[0])) {

String client\_id = raw.split(";")[0];

String id = raw.split(";")[1];

clients.get(i).getAccounts().add(new Account(client\_id, id, 0.0));

// load last added account movements

loadAccountMovements(clients.get(i).getAccounts().get(clients.get(i).getAccounts().size() - 1));

clients.get(i).setAccounts(clients.get(i).getAccounts());

}

}

}

fr.close();

br.close();

} catch(FileNotFoundException e) {

System.out.println("[ERROR] File \"accounts.csv\" was not found.");

// e.printStackTrace();

} catch(IOException e) {

System.out.println("[ERROR] Couldn't load file \"accounts.csv\".");

// e.printStackTrace();

} catch(Exception e) {

System.out.println("[ERROR] An error has occurred.");

// e.printStackTrace();

}

}

public void registerAccount(Account account) {

File f;

FileWriter fw;

BufferedWriter bw;

try {

fw = new FileWriter("accounts.csv", true);

bw = new BufferedWriter(fw);

bw.newLine();

bw.write(account.getClient\_id() + ";" + account.getId());

bw.close();

} catch(FileNotFoundException e) {

System.out.println("[ERROR] File \"accounts.csv\" was not found.");

// e.printStackTrace();

} catch(IOException e) {

System.out.println("[ERROR] Couldn't load file \"accounts.csv\".");

// e.printStackTrace();

} catch(Exception e) {

System.out.println("[ERROR] An error has occurred.");

// e.printStackTrace();

}

}

}

**ClientManagement.java**

//

// Copyright (c) Carlos Tojal 2020

// Bank

// ClientManagement.java

//

package management;

import java.util.ArrayList;

import java.util.Random;

import java.io.File;

import java.io.FileReader;

import java.io.FileWriter;

import java.io.BufferedReader;

import java.io.BufferedWriter;

import java.io.PrintWriter;

import java.io.FileNotFoundException;

import java.io.IOException;

import java.lang.Exception;

import structures.Client;

import structures.Account;

import management.AccountManagement;

public class ClientManagement {

// Constructor

public ClientManagement() {

}

// Methods

public String generateId() {

int leftLimit = 48; // number 0

int rightLimit = 57; // letter 9

int targetStringLength = 8;

Random random = new Random();

StringBuilder buffer = new StringBuilder(targetStringLength);

for (int i = 0; i < targetStringLength; i++) {

int randomLimitedInt = leftLimit + (int)

(random.nextFloat() \* (rightLimit - leftLimit + 1));

buffer.append((char) randomLimitedInt);

}

String generatedString = buffer.toString();

return generatedString;

}

public void updateClients(ArrayList<Client> clients) {

AccountManagement accountManagement = new AccountManagement();

try {

File file = new File("clients.csv");

PrintWriter writer = new PrintWriter(file);

writer.print("");

writer.close();

for (int i = 0; i < clients.size(); i++)

registerClient(clients.get(i));

} catch(IOException e) {

System.out.println("[ERROR] Couldn't load file \"clients.csv\".");

// e.printStackTrace();

} catch(Exception e) {

System.out.println("[ERROR] An error has occurred.");

// e.printStackTrace();

}

}

public ArrayList<Client> loadClients() {

AccountManagement accountManagement = new AccountManagement();

ArrayList<Client> clients = new ArrayList<Client>();

ArrayList<Account> accounts = new ArrayList<Account>();

String raw;

Client client = new Client();

File f;

FileReader fr;

BufferedReader br;

// Loads clients

try {

f = new File("clients.csv");

fr = new FileReader(f);

br = new BufferedReader(fr);

while ((raw = br.readLine()) != null) {

client = new Client();

client.setId(raw.split(";")[0]);

client.setName(raw.split(";")[1]);

client.setUsername(raw.split(";")[2]);

client.setPin(Integer.parseInt(raw.split(";")[3]));

client.setAccounts(new ArrayList<Account>());

client.setAccess\_level(Byte.parseByte(raw.split(";")[4]));

clients.add(client);

}

fr.close();

br.close();

// Loads client accounts

accountManagement.loadAccounts(clients);

} catch(FileNotFoundException e) {

System.out.println("[ERROR] File \"clients.csv\" was not found.");

// e.printStackTrace();

} catch(IOException e) {

System.out.println("[ERROR] Couldn't load file \"clients.csv\".");

// e.printStackTrace();

} catch(Exception e) {

System.out.println("[ERROR] An error has occurred.");

// e.printStackTrace();

}

return clients;

}

public void registerClient(Client client) {

File f;

FileWriter fw;

BufferedWriter bw;

try {

fw = new FileWriter("clients.csv", true);

bw = new BufferedWriter(fw);

bw.newLine();

bw.write(client.getId() + ";" + client.getName() + ";" + client.getUsername() + ";" + client.getPin() + ";" + client.getAccess\_level());

bw.close();

} catch(FileNotFoundException e) {

System.out.println("[ERROR] File \"clients.csv\" was not found.");

// e.printStackTrace();

} catch(IOException e) {

System.out.println("[ERROR] Couldn't load file \"clients.csv\".");

// e.printStackTrace();

} catch(Exception e) {

System.out.println("[ERROR] An error has occurred.");

// e.printStackTrace();

}

}

}

**Account.java**

//

// Copyright (c) Carlos Tojal 2020

// Bank

// Account.java

//

package structures;

import java.util.ArrayList;

import structures.AccountMovement;

public class Account {

// Attributes

private String client\_id;

private String id;

private double balance;

private ArrayList<AccountMovement> accountMovements;

// Constructors

public Account() {

this.client\_id = "ClientID";

this.id = "AccountID";

this.balance = 0.00;

this.accountMovements = new ArrayList<AccountMovement>();

}

public Account(String client\_id, String id, double balance) {

this.client\_id = client\_id;

this.id = id;

this.balance = balance;

this.accountMovements = new ArrayList<AccountMovement>();

}

public Account(String client\_id, String id, double balance, ArrayList<AccountMovement> accountMovements) {

this.client\_id = client\_id;

this.id = id;

this.balance = balance;

this.accountMovements = accountMovements;

}

// Getters and setters

public String getClient\_id() {

return client\_id;

}

public void setClient\_id(String client\_id) {

this.client\_id = client\_id;

}

public String getId() {

return id;

}

public void setId(String id) {

this.id = id;

}

public double getBalance() {

return balance;

}

public ArrayList<AccountMovement> getAccountMovements() {

return accountMovements;

}

public void setAccountMovements(ArrayList<AccountMovement> accountMovements) {

this.accountMovements = accountMovements;

}

// Methods

public void deposit(double value) {

if(value > 0)

this.balance += value;

}

public void withdraw(double value) {

if(value > 0) {

if((this.balance - value) >= 0)

this.balance -= value;

}

}

}

**AccountMovement.java**

//

// Copyright (c) Carlos Tojal 2020

// Bank

// AccountMovement.java

//

package structures;

public class AccountMovement {

// Attributes

private String account\_id;

private double value;

private byte type; // 1 - deposit; 2 - withdraw

// Constructors

public AccountMovement() {

this.account\_id = "AccountID";

this.value = 0.00;

this.type = 1;

}

public AccountMovement(String account\_id, double value, byte type) {

this.account\_id = account\_id;

this.value = value;

this.type = type;

}

// Getters and setters

public String getAccount\_id() {

return account\_id;

}

public void setAccount\_id(String account\_id) {

this.account\_id = account\_id;

}

public double getValue() {

return value;

}

public void setValue(double value) {

this.value = value;

}

public byte getType() {

return type;

}

public void setType(byte type) {

this.type = type;

}

}

**Client.java**

//

// Copyright (c) Carlos Tojal 2020

// Bank

// Client.java

//

package structures;

import java.util.ArrayList;

public class Client {

// Attributes

private String id;

private String name;

private String username;

private int pin;

private ArrayList<Account> accounts;

private byte access\_level; // 1 - client; 2 - bank employee

// Constructors

public Client() {

this.id = "1234";

this.name = "ClientName";

this.accounts = new ArrayList<Account>();

}

public Client(String id, String name, ArrayList<Account> accounts) {

this.id = id;

this.name = name;

this.accounts = accounts;

}

// Getters and setters

public String getId() {

return id;

}

public void setId(String id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getUsername() {

return username;

}

public void setUsername(String username) {

this.username = username;

}

public int getPin() {

return pin;

}

public void setPin(int pin) {

this.pin = pin;

}

public ArrayList<Account> getAccounts() {

return accounts;

}

public void setAccounts(ArrayList<Account> accounts) {

this.accounts = accounts;

}

public byte getAccess\_level() {

return access\_level;

}

public void setAccess\_level(byte access\_level) {

this.access\_level = access\_level;

}

}

**ClientOptionsWindow.java**

//

// Copyright (c) Carlos Tojal 2020

// Bank

// ClientOptionsWindow.java

//

package ui;

import java.util.ArrayList;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.GridLayout;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.JOptionPane;

import javax.swing.JButton;

import java.lang.Exception;

import structures.Client;

import ui.ListClientAccountsWindow;

import ui.CreateClientAccountWindow;

public class ClientOptionsWindow {

// Attributes

private Client client;

private JFrame frame;

private JPanel panel;

private JButton list\_accounts;

private JButton create\_account;

// Constructor

public ClientOptionsWindow(Client client) {

frame = new JFrame("Client Options - Bank");

panel = new JPanel(new GridLayout(1, 2));

list\_accounts = new JButton("List Accounts");

create\_account = new JButton("Create Account");

this.client = client;

panel.add(list\_accounts);

panel.add(create\_account);

frame.add(panel);

frame.setSize(400, 200);

frame.setResizable(false);

frame.setLocationRelativeTo(null);

frame.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

frame.setVisible(true);

list\_accounts.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

new ListClientAccountsWindow(client, (byte) 1);

frame.dispose();

}

});

create\_account.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

new CreateClientAccountWindow(client);

frame.setVisible(false);

}

});

}

}

**ControlPanel.java**

//

// Copyright (c) Carlos Tojal 2020

// Bank

// ControlPanel.java

//

package ui;

import java.util.ArrayList;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.WindowAdapter;

import java.awt.event.WindowEvent;

import java.awt.GridLayout;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.JLabel;

import javax.swing.JButton;

import javax.swing.JOptionPane;

import structures.Client;

import management.ClientManagement;

public class ControlPanel {

// Attributes

private Client client;

private ArrayList<Client> clients;

private ClientManagement clientManagement;

private JFrame frame;

private JPanel panel;

private JLabel title;

private JButton register\_client;

private JButton list\_clients;

private JButton view\_data;

private JButton deposit;

private JButton withdraw;

// Constructor

public ControlPanel(Client client) {

this.client = client;

clientManagement = new ClientManagement();

clients = clientManagement.loadClients();

frame = new JFrame("Bank");

panel = new JPanel(new GridLayout(5, 2));

title = new JLabel("Welcome, " + client.getName());

register\_client = new JButton("Register client");

list\_clients = new JButton("List clients");

view\_data = new JButton("View data");

deposit = new JButton("Deposit");

withdraw = new JButton("Withdraw");

panel.add(title);

if(client.getAccess\_level() == 2) {

panel.add(register\_client);

panel.add(list\_clients);

}

panel.add(view\_data);

panel.add(deposit);

panel.add(withdraw);

frame.add(panel);

frame.setSize(600, 450);

frame.setResizable(false);

frame.setLocationRelativeTo(null);

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setVisible(true);

register\_client.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

new RegisterClientWindow();

}

});

list\_clients.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

new ListClientsWindow();

}

});

view\_data.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e) {

String data = "";

data += "Client ID: " + client.getId() + "\n";

data += "Name: " + client.getName() + "\n";

data += "Username: " + client.getUsername() + "\n";

data += "PIN: " + client.getPin() + "\n";

data += "Access level: " + client.getAccess\_level() + "\n";

data += "Number of accounts: " + client.getAccounts().size();

JOptionPane.showMessageDialog(null, data, "Client data", JOptionPane.PLAIN\_MESSAGE);

}

});

deposit.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

new ListClientAccountsWindow(client, (byte) 2);

}

});

withdraw.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

new ListClientAccountsWindow(client, (byte) 3);

}

});

}

}

**CreateClientAccountWindow.java**

//

// Copyright (c) Carlos Tojal 2020

// Bank

// CreateClientAccountWindow.java

//

package ui;

import java.util.ArrayList;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.WindowEvent;

import java.awt.GridLayout;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.JLabel;

import javax.swing.JTextField;

import javax.swing.JButton;

import javax.swing.JOptionPane;

import java.lang.NumberFormatException;

import java.lang.Exception;

import structures.Client;

import structures.Account;

import management.ClientManagement;

import management.AccountManagement;

public class CreateClientAccountWindow {

// Attributes

private Client client;

private Account account;

private ClientManagement clientManagement;

private AccountManagement accountManagement;

private JFrame frame;

private JPanel panel;

private JLabel id\_label;

private JTextField id;

private JLabel name\_label;

private JTextField name;

private JButton confirm;

// Constructor

public CreateClientAccountWindow(Client client) {

frame = new JFrame("Create Client Account - Bank");

panel = new JPanel(new GridLayout(5, 1));

id\_label = new JLabel("Account ID: ");

id = new JTextField();

name\_label = new JLabel("Client Name: ");

name = new JTextField();

confirm = new JButton("Confirm");

this.client = client;

clientManagement = new ClientManagement();

accountManagement = new AccountManagement();

account = new Account(client.getId(), clientManagement.generateId(), 0.00);

id.setText(account.getId());

id.setEditable(false);

name.setText(client.getName());

name.setEditable(false);

panel.add(id\_label);

panel.add(id);

panel.add(name\_label);

panel.add(name);

panel.add(confirm);;

frame.add(panel);

frame.setSize(400, 500);

frame.setResizable(false);

frame.setLocationRelativeTo(null);

frame.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

frame.setVisible(true);

confirm.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

accountManagement.registerAccount(account);

JOptionPane.showMessageDialog(null, "Account created successfully.", "Success", JOptionPane.INFORMATION\_MESSAGE);

frame.dispose();

}

});

}

}

**DepositWindow.java**

//

// Copyright (c) Carlos Tojal 2020

// Bank

// DepositWindow.java

//

package ui;

import javax.swing.JOptionPane;

import java.lang.NumberFormatException;

import java.lang.Exception;

import structures.Account;

import structures.AccountMovement;

import management.AccountManagement;

public class DepositWindow {

// Attributes

boolean success = true;

AccountMovement accountMovement;

AccountManagement accountManagement;

// Constructor

public DepositWindow(Account account) {

accountMovement = new AccountMovement();

accountMovement.setType((byte) 1);

accountManagement = new AccountManagement();

accountMovement.setAccount\_id(account.getId());

do {

try {

accountMovement.setValue(Double.parseDouble(JOptionPane.showInputDialog("Value to deposit: ")));

} catch (NumberFormatException e) {

success = false;

System.out.println("[ERROR] Invalid value input.");

JOptionPane.showMessageDialog(null, "Invalid value input.", "Error", JOptionPane.ERROR\_MESSAGE);

}

if(accountMovement.getValue() > 0) {

account.deposit(accountMovement.getValue());

} else {

success = false;

System.out.println("[ERROR] Only positive values are allowed.");

JOptionPane.showMessageDialog(null, "Only positive values are allowed.", "Error", JOptionPane.ERROR\_MESSAGE);

}

}while(!success);

account.getAccountMovements().add(accountMovement);

accountManagement.registerAccountMovement(accountMovement);

account.setAccountMovements(account.getAccountMovements());

}

}

**ListAccountMovementsWindow.java**

//

// Copyright (c) Carlos Tojal 2020

// Bank

// ListAccountMovementsWindow.java

//

package ui;

import java.util.ArrayList;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.WindowEvent;

import java.awt.event.MouseAdapter;

import java.awt.event.MouseEvent;

import java.awt.GridLayout;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.JLabel;

import javax.swing.JList;

import javax.swing.JScrollPane;

import javax.swing.ListSelectionModel;

import javax.swing.JOptionPane;

import java.lang.Exception;

import structures.Account;

import structures.Client;

import management.AccountManagement;

import management.ClientManagement;

public class ListAccountMovementsWindow {

// Attributes

private Client client;

private ArrayList<Client> clients;

private ArrayList<String> options;

private ClientManagement clientManagement;

private JFrame frame;

private JPanel panel;

private JLabel label;

private JList<String> list;

// Constructor

public ListAccountMovementsWindow(Account account) {

frame = new JFrame("List Account Movements - Bank");

panel = new JPanel(new GridLayout(1, 1));

label = new JLabel("No movements in this account.");

if(account.getAccountMovements().size() > 0) {

options = new ArrayList<String>();

for (int i = 0; i < account.getAccountMovements().size(); i++) {

String output = "";

output += account.getAccountMovements().get(i).getType() == 1 ? "DEPOSIT - " : "WITHDRAW - ";

output += account.getAccountMovements().get(i).getValue();

options.add(output);

}

list = new JList<String>((String[]) options.toArray(new String[0]));

JScrollPane scrollPane = new JScrollPane(list);

panel.add(scrollPane);

} else {

panel.add(label);

}

frame.add(panel);

frame.setSize(400, 500);

frame.setResizable(false);

frame.setLocationRelativeTo(null);

frame.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

frame.setVisible(true);

}

}

**ListClientAccountsWindow.java**

//

// Copyright (c) Carlos Tojal 2020

// Bank

// ListClientAccountsWindow.java

//

package ui;

import java.util.ArrayList;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.WindowEvent;

import java.awt.event.MouseAdapter;

import java.awt.event.MouseEvent;

import java.awt.GridLayout;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.JList;

import javax.swing.JScrollPane;

import javax.swing.ListSelectionModel;

import javax.swing.JOptionPane;

import java.lang.Exception;

import structures.Client;

import management.ClientManagement;

public class ListClientAccountsWindow {

// Attributes

private Client client;

private ArrayList<Client> clients;

private ArrayList<String> options;

private ClientManagement clientManagement;

private JFrame frame;

private JPanel panel;

private JList<String> list;

// Constructor

public ListClientAccountsWindow(Client client, byte action) {

frame = new JFrame("List Client Accounts - Bank");

panel = new JPanel(new GridLayout(1, 1));

this.client = client;

clientManagement = new ClientManagement();

clients = clientManagement.loadClients();

options = new ArrayList<String>();

for(int i = 0; i < client.getAccounts().size(); i++)

options.add(client.getAccounts().get(i).getId() + " - " + client.getAccounts().get(i).getBalance());

list = new JList<String>((String[])options.toArray(new String[0]));

JScrollPane scrollPane = new JScrollPane(list);

// 1 - list movements; 2 - deposit; 3 - withdraw

list.addMouseListener(new MouseAdapter() {

public void mouseClicked(MouseEvent evt) {

JList list = (JList) evt.getSource();

if(evt.getClickCount() == 2) {

int index = list.locationToIndex(evt.getPoint());

if(action == 1)

new ListAccountMovementsWindow(client.getAccounts().get(index));

else if(action == 2)

new DepositWindow(client.getAccounts().get(index));

else

new WithdrawWindow(client.getAccounts().get(index));

frame.dispose();

}

}

});

panel.add(scrollPane);

frame.add(panel);

frame.setSize(400, 500);

frame.setResizable(false);

frame.setLocationRelativeTo(null);

frame.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

frame.setVisible(true);

}

}

**ListClientsWindow.java**

//

// Copyright (c) Carlos Tojal 2020

// Bank

// ListClientsWindow.java

//

package ui;

import java.util.ArrayList;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.WindowEvent;

import java.awt.event.MouseAdapter;

import java.awt.event.MouseEvent;

import java.awt.GridLayout;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.JList;

import javax.swing.JScrollPane;

import javax.swing.ListSelectionModel;

import javax.swing.JOptionPane;

import java.lang.Exception;

import structures.Client;

import management.ClientManagement;

public class ListClientsWindow {

// Attributes

private ArrayList<Client> clients;

private ArrayList<String> options;

private ClientManagement clientManagement;

private JFrame frame;

private JPanel panel;

private JList<String> list;

// Constructor

public ListClientsWindow() {

frame = new JFrame("List Clients - Bank");

panel = new JPanel(new GridLayout(1, 1));

clientManagement = new ClientManagement();

clients = clientManagement.loadClients();

options = new ArrayList<String>();

for(int i = 0; i < clients.size(); i++)

options.add(clients.get(i).getId() + " (" + clients.get(i).getUsername() + ")");

list = new JList<String>((String[])options.toArray(new String[0]));

JScrollPane scrollPane = new JScrollPane(list);

list.addMouseListener(new MouseAdapter() {

public void mouseClicked(MouseEvent evt) {

JList list = (JList) evt.getSource();

if(evt.getClickCount() == 2) {

int index = list.locationToIndex(evt.getPoint());

//JOptionPane.showMessageDialog(null, options.get(index), "Message", JOptionPane.ERROR\_MESSAGE);

new ClientOptionsWindow(clients.get(index));

frame.dispose();

}

}

});

panel.add(scrollPane);

frame.add(panel);

frame.setSize(400, 500);

frame.setResizable(false);

frame.setLocationRelativeTo(null);

frame.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

frame.setVisible(true);

}

}

**LoginWindow.java**

//

// Copyright (c) Carlos Tojal 2020

// Bank

// LoginWindow.java

//

package ui;

import java.util.ArrayList;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.WindowEvent;

import java.awt.GridLayout;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.JLabel;

import javax.swing.JTextField;

import javax.swing.JPasswordField;

import javax.swing.JButton;

import javax.swing.JOptionPane;

import javax.swing.UIManager;

import javax.swing.UnsupportedLookAndFeelException;

import java.lang.NumberFormatException;

import java.lang.Exception;

import structures.Client;

import management.ClientManagement;

public class LoginWindow {

// Attributes

private Client client;

private ArrayList<Client> clients;

private ClientManagement clientManagement;

private JFrame frame;

private JPanel panel;

private JLabel username\_label;

private JTextField username;

private JLabel pin\_label;

private JTextField pin;

private JButton login;

// Constructor

public LoginWindow() {

try {

UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());

System.out.println("[MESSAGE] Set system look and feel.");

} catch(UnsupportedLookAndFeelException e) {

System.out.println("[ERROR] System look and feel not supported.");

} catch(ClassNotFoundException e) {

System.out.println("[ERROR] System look and feel not found.");

} catch(InstantiationException e) {

System.out.println("[ERROR] Error setting system look and feel.");

} catch(IllegalAccessException e) {

System.out.println("[ERROR] Couldn't access system look and feel.");

}

clientManagement = new ClientManagement();

clients = clientManagement.loadClients();

frame = new JFrame("Login - Bank");

panel = new JPanel(new GridLayout(5, 1));

username\_label = new JLabel("Username: ");

username = new JTextField();

pin\_label = new JLabel("PIN: ");

pin = new JPasswordField();

login = new JButton("Login");

panel.add(username\_label);

panel.add(username);

panel.add(pin\_label);

panel.add(pin);

panel.add(login);

frame.add(panel);

frame.setSize(400, 300);

frame.setResizable(false);

frame.setLocationRelativeTo(null);

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setVisible(true);

login.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

boolean inputSuccess = true;

client = new Client();

client.setUsername(username.getText());

try {

client.setPin(Integer.parseInt(pin.getText()));

} catch(NumberFormatException nfe) {

inputSuccess = false;

System.out.println("[ERROR] Invalid PIN input.");

JOptionPane.showMessageDialog(null, "Invalid PIN input.", "Error", JOptionPane.ERROR\_MESSAGE);

}

if(inputSuccess) {

if(login(client, clients)) {

System.out.println("[MESSAGE] Login successful.");

frame.setVisible(false);

new ControlPanel(client);

} else {

System.out.println("[MESSAGE] Invalid username or password.");

JOptionPane.showMessageDialog(null, "Invalid username or password.", "Error", JOptionPane.ERROR\_MESSAGE);

}

}

}

});

}

// Getters

public Client getClient() {

return client;

}

// Methods

public boolean login(Client client, ArrayList<Client> clients) {

boolean success = false;

for(int i = 0; i < clients.size(); i++) {

if(clients.get(i).getUsername().equals(client.getUsername()) && clients.get(i).getPin() == client.getPin()) {

this.client = clients.get(i);

success = true;

return success;

}

}

return success;

}

}

**RegisterClientWindow.java**

//

// Copyright (c) Carlos Tojal 2020

// Bank

// RegisterClientWindow.java

//

package ui;

import java.util.ArrayList;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.WindowEvent;

import java.awt.GridLayout;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.JLabel;

import javax.swing.JTextField;

import javax.swing.JPasswordField;

import javax.swing.JButton;

import javax.swing.JOptionPane;

import java.lang.NumberFormatException;

import java.lang.Exception;

import structures.Client;

import management.ClientManagement;

public class RegisterClientWindow {

// Attributes

private Client client;

private ArrayList<Client> clients;

private ClientManagement clientManagement;

private JFrame frame;

private JPanel panel;

private JLabel name\_label;

private JTextField name;

private JLabel username\_label;

private JTextField username;

private JLabel pin\_label;

private JTextField pin;

private JButton register;

// Constructor

public RegisterClientWindow() {

frame = new JFrame("Register Client - Bank");

panel = new JPanel(new GridLayout(7, 1));

name\_label = new JLabel("Name: ");

name = new JTextField();

username\_label = new JLabel("Username: ");

username = new JTextField();

pin\_label = new JLabel("PIN: ");

pin = new JPasswordField();

register = new JButton("Register");

panel.add(name\_label);

panel.add(name);

panel.add(username\_label);

panel.add(username);

panel.add(pin\_label);

panel.add(pin);

panel.add(register);

frame.add(panel);

frame.setSize(400, 500);

frame.setResizable(false);

frame.setLocationRelativeTo(null);

frame.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

frame.setVisible(true);

register.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

boolean inputSuccess = true;

client = new Client();

clientManagement = new ClientManagement();

clients = clientManagement.loadClients();

client.setId(clientManagement.generateId());

client.setName(name.getText());

client.setUsername(username.getText());

client.setAccess\_level((byte) 1);

try {

client.setPin(Integer.parseInt(pin.getText()));

} catch(NumberFormatException nfe) {

inputSuccess = false;

System.out.println("[ERROR] Invalid PIN input.");

JOptionPane.showMessageDialog(null, "Invalid PIN input.", "Error", JOptionPane.ERROR\_MESSAGE);

}

if(inputSuccess) {

clientManagement.registerClient(client);

JOptionPane.showMessageDialog(null, "Client registered successfully.", "Success", JOptionPane.INFORMATION\_MESSAGE);

frame.dispose();

}

}

});

}

// Getters

public Client getClient() {

return client;

}

}

**WithdrawWindow.java**

//

// Copyright (c) Carlos Tojal 2020

// Bank

// WithdrawWindow.java

//

package ui;

import javax.swing.JOptionPane;

import java.lang.NumberFormatException;

import java.lang.Exception;

import structures.Account;

import structures.AccountMovement;

import management.AccountManagement;

public class WithdrawWindow {

// Attributes

boolean success = true;

AccountMovement accountMovement;

AccountManagement accountManagement;

// Constructor

public WithdrawWindow(Account account) {

accountMovement = new AccountMovement();

accountMovement.setType((byte )2);

accountManagement = new AccountManagement();

accountMovement.setAccount\_id(account.getId());

do {

try {

accountMovement.setValue(Double.parseDouble(JOptionPane.showInputDialog("Value to withdraw: ")));

} catch (NumberFormatException e) {

success = false;

System.out.println("[ERROR] Invalid value input.");

JOptionPane.showMessageDialog(null, "Invalid value input.", "Error", JOptionPane.ERROR\_MESSAGE);

}

if(accountMovement.getValue() > 0) {

account.withdraw(accountMovement.getValue());

} else {

success = false;

System.out.println("[ERROR] Only positive values are allowed.");

JOptionPane.showMessageDialog(null, "Only positive values are allowed.", "Error", JOptionPane.ERROR\_MESSAGE);

}

}while(!success);

account.getAccountMovements().add(accountMovement);

accountManagement.registerAccountMovement(accountMovement);

account.setAccountMovements(account.getAccountMovements());

}

}

**Main.java**

//

// Copyright (c) Carlos Tojal 2020

// Bank

// Main.java

//

import ui.LoginWindow;

public class Main {

public static void main(String[] args) {

new LoginWindow();

}

}

Conclusão

Concluindo, com este módulo tive a oportunidade de adquirir conhecimentos avançados acerca da programação orientada a objetos, tais como:

* Streams
* Exceções
* Listas dinâmicas
* Interface Gráfica (GUI)

Considero os objetivos alcançados, através da realização dos projetos/fichas cujo código está presente no portfólio acima.