**Week 2: Programming Practice**



**CODE:**

package presentation;

import business.Invoice;

import java.awt.Dimension;

import java.awt.GridBagConstraints;

import java.awt.GridBagLayout;

import java.awt.GridLayout;

import java.awt.Insets;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JTextArea;

import javax.swing.JTextField;

public class Invoice\_GUI extends JFrame {

 public static final int WINDOW\_WIDTH = 450;

 public static final int WINDOW\_HEIGHT = 400;

 private Invoice aInvoice;

 private JButton btnCalculate;

 private JButton btnClear;

 private JButton btnExit;

 private JTextField txtProductName;

 private JTextField txtQuantity;

 private JTextField txtCostPerItem;

 private JTextArea txtTotalCost;

 public Invoice\_GUI() {

 super();

 createPanel();

 setFrame();

 }

 private void createPanel() {

 super.setLayout(new GridLayout(0, 2));

 //TODO: create the GUI components and add them to the form for:

 /\*

 1. Product name label

 2. Product name textfield (name: txtProductName)

 3. Quantity label

 4. Quanitity textfield (name: txtQuantity)

 5. Item cost label

 6. Item cost textfield (name: txtCost)

 7. JButton to calculate the cost (name: btnCalculate

 8. Add a CalculateHandler handler object to the btnCalculate addActionListener

 \*/

 JLabel lblProduct = new JLabel("Product Name");

 lblProduct.setHorizontalAlignment( JLabel.CENTER );

 this.add(lblProduct);

 txtProductName = new JTextField(10);

 this.add(txtProductName);

 JLabel lblQuantity = new JLabel("Product Quantity (1 to 1000):");

 lblQuantity.setHorizontalAlignment( JLabel.CENTER );

 this.add(lblQuantity);

 txtQuantity = new JTextField(10);

 this.add(txtQuantity);

 JLabel lblCost = new JLabel("Item Cost (10 to 10000):");

 lblCost.setHorizontalAlignment( JLabel.CENTER );

 this.add(lblCost);

 txtCostPerItem = new JTextField(10);

 this.add(txtCostPerItem);

 JLabel lblTotalCost = new JLabel("Total Cost");

 lblTotalCost.setHorizontalAlignment( JLabel.CENTER );

 this.add(lblTotalCost);

 txtTotalCost = new JTextArea();

 this.add(txtTotalCost);

 btnCalculate = new JButton("Calculate");

 btnCalculate.addActionListener(new CalculateHandler() );

 this.add(btnCalculate);

 }

 private void setFrame() {

 Dimension windowSize = new Dimension(WINDOW\_WIDTH, WINDOW\_HEIGHT);

 super.setTitle("Week 2 Practice Program--Invoice");

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

 super.setSize(windowSize);

 super.setMinimumSize(windowSize);

 super.setMaximumSize(windowSize);

 super.setLocationRelativeTo(null);

 super.setVisible(true);

 }

 private class CalculateHandler implements ActionListener {

 @Override

 public void actionPerformed(ActionEvent ae) {

 boolean valid;

 if (aInvoice == null) {

 //TODO: create a new invoice object

 aInvoice = new Invoice();

 }

 aInvoice.setProductName(txtProductName.getText());

 try {

 int quanity = Integer.parseInt(txtQuantity.getText());

 aInvoice.setQuantityPurchased(quanity);

 valid = true;

 }

 catch (NumberFormatException ex) {

 JOptionPane.showMessageDialog(null, "Quantity needs to be a number between 1 and 1000",

 "Illegal Quanity value",

 JOptionPane.ERROR\_MESSAGE);

 valid = false;

 txtQuantity.setText("");

 txtQuantity.requestFocus();

 }

 if (valid) {

 try {

 double cost = 0;

 //TODO: write the state to extract the double value from the txtCost field

 cost = Double.parseDouble( txtCostPerItem.getText() );

 aInvoice.setPricePerItem(cost);

 }

 catch (NumberFormatException ex) {

 valid = false;

 JOptionPane.showMessageDialog(null, "Costs needs to be a number between 10 and 10000",

 "Illegal Cost value",

 JOptionPane.ERROR\_MESSAGE);

 valid = false;

 txtCostPerItem.setText("");

 txtCostPerItem.requestFocus();

 }

 }

 if (valid) {

 //TODO: using the toString method of the invoice object to set the output text

 txtTotalCost.setText(aInvoice.toString() );

 }

 }

 }

}