**“Año de la recuperación y consolidación de la economía peruana”**

**UNIVERSIDAD PERUANA LOS ANDES**



**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN**

**Asignatura:**

**TALLER VII: Desarrollo De Aplicaciones I**  
  
**Apache Maven: Semana 3**

**DOCENTE:** Mg. Raúl Fernández Bejarano

**ESTUDIANTE:**

Huaman Cárdenas, Jhon Frank

**CICLO – SECCIÓN:** 4to – A1

**Huancayo – 2025**

## **1. Enunciado**

Una tienda ha puesto en oferta la venta de un producto ofreciendo un porcentaje de descuento sobre el importe de la compra de acuerdo con la siguiente tabla:

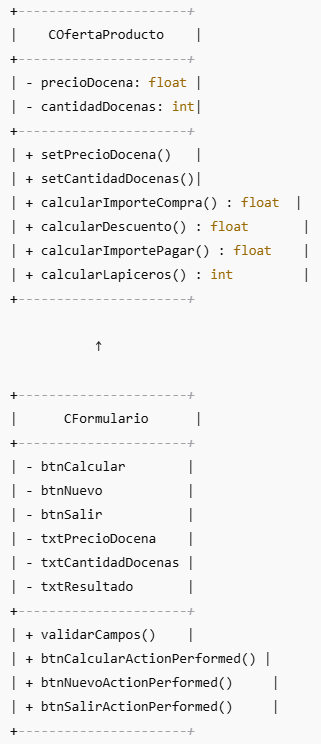
|  |  |
| --- | --- |
| **Docenas adquiridas** | **Descuento** |
| ≥ 10 | 20% |
| < 10 | 10% |

Adicionalmente, la tienda obsequia lapiceros de acuerdo con a la siguiente tabla:

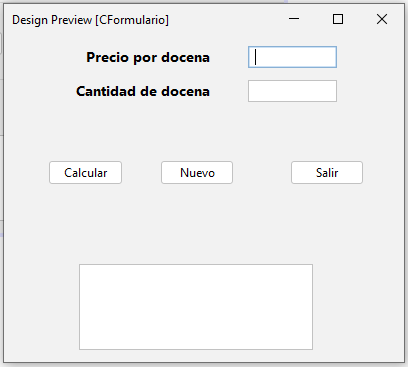
|  |  |
| --- | --- |
| **Importe a pagar (S/.)** | **Cantidad de Lapiceros** |
| ≥ 200 | 2 por cada docena adquirida |
| < 200 | 0 |

Dado el precio de la docena y la cantidad de docenas adquiridas, diseñe un programa que determine el importe de la compra, el importe del descuento, el importe a pagar y la cantidad de lapiceros de obsequio.

**1.1. Diagrama de Clase**



**1.2. Diseño de la Aplicación**



**1.3. Código de la Aplicación**

**JFrame Form (Source)**

**Importamos:**   
 **package Vista;**

**import Clases.COfertaProducto;**

**import java.awt.Color;**

**import java.awt.Dimension;**

**import javax.swing.JOptionPane;**  
   
   
 **public class CFormulario extends javax.swing.JFrame {**

**COfertaProducto oferta = new COfertaProducto();**

**public CFormulario() {**

**initComponents();**

**formulario();**

**}**

**private void formulario() {**

**setTitle("Oferta de Producto");**

**setLocationRelativeTo(null);**

**setResizable(false);**

**setSize(new Dimension(400, 350));**

**getContentPane().setBackground(Color.white);**

**panelResultado.setVisible(false);**

**btnCalcular.setEnabled(false);**

**panelBotones.setVisible(true);**

**}**

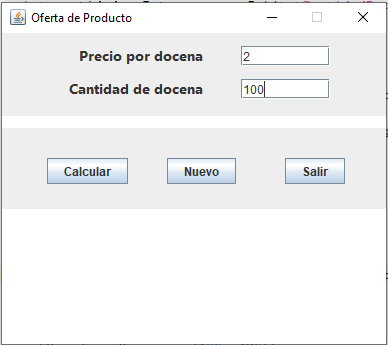
**Los cálculos se realizan dentro de los métodos (btn, txt, etc):**

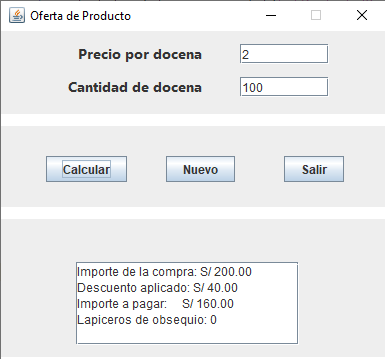
private void btnCalcularActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here: try { float precio = Float.parseFloat(txtPrecioDocena.getText().trim()); int cantidad = Integer.parseInt(txtCantidadDocenas.getText().trim());

oferta.setPrecioDocena(precio);  
 oferta.setCantidadDocenas(cantidad);  
  
 float importeCompra = oferta.calcularImporteCompra();  
 float descuento = oferta.calcularDescuento();  
 float importePagar = oferta.calcularImportePagar();  
 int lapiceros = oferta.calcularLapiceros();  
  
 String resultado = """  
 Importe de la compra: S/ %.2f  
 Descuento aplicado: S/ %.2f  
 Importe a pagar: S/ %.2f  
 Lapiceros de obsequio: %d  
 """.formatted(importeCompra, descuento, importePagar, lapiceros);  
  
 txtResultado.setText(resultado);  
 panelResultado.setVisible(true);  
 setSize(new Dimension(400, 420));  
 } catch (NumberFormatException e) {  
 JOptionPane.showMessageDialog(this, "Ingrese valores numéricos válidos.");  
 }  
}   
  
private void btnNuevoActionPerformed(java.awt.event.ActionEvent evt) {   
 // TODO add your handling code here:  
 txtPrecioDocena.setText("");  
 txtCantidadDocenas.setText("");  
 txtResultado.setText("");  
 panelResultado.setVisible(false);  
 setSize(new Dimension(400, 350));  
 btnCalcular.setEnabled(false);  
}   
  
private void btnSaliActionPerformed(java.awt.event.ActionEvent evt) {   
 // TODO add your handling code here:  
 int r = JOptionPane.showConfirmDialog(this, "¿Deseas salir?", "Salir", JOptionPane.YES\_NO\_OPTION);  
 if (r == JOptionPane.YES\_OPTION) {  
 System.exit(0);  
 }  
}   
  
private void validarCampos() {  
 String precio = txtPrecioDocena.getText().trim();  
 String cantidad = txtCantidadDocenas.getText().trim();  
  
 boolean habilitar = !precio.isEmpty() && !cantidad.isEmpty();  
  
 // Verificamos que ambos campos sean numéricos válidos  
 try {  
 Float.parseFloat(precio);  
 Integer.parseInt(cantidad);  
 } catch (NumberFormatException e) {  
 habilitar = false;  
 }  
  
 btnCalcular.setEnabled(habilitar);  
}

**1.4. Resultado**





## **2. Enunciado**

Una empresa de transportes cubre la ruta Lima – Huancayo en dos turnos: mañana y noche. Los precios de los pasajes se dan en la siguiente tabla:

|  |  |
| --- | --- |
| **Turno** | **Tarifa Horaria (S/.)** |
| Mañana | S/. 37.5 |
| Noche | S/. 37.5 |

Como oferta especial, la empresa aplica un porcentaje de descuento sobre el importe de la compra de acuerdo con a la siguiente tabla:

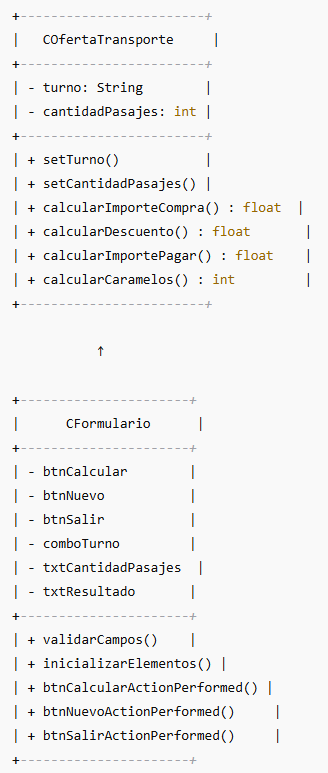
|  |  |
| --- | --- |
| **Cantidad de pasajes** | **Descuento** |
| ≥ 15 | 8% |
| < 15 | 5% |

Adicionalmente, la empresa obsequia caramelos de acuerdo con la siguiente tabla.

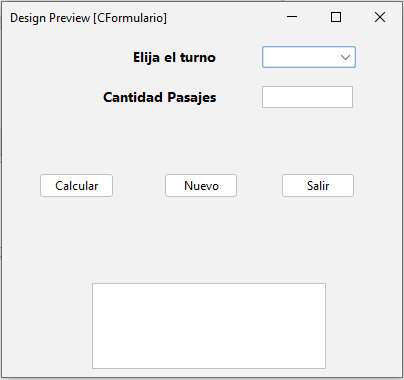
|  |  |
| --- | --- |
| **Importe por pagar** | **Descuento** |
| > 200 | 2 por cada boleto |
| ≤ 200 | 0 |

Dado el turno y la cantidad de pasajes adquiridos por un cliente, diseñe un programa que determine el importe de la compra, el importe del descuento, el importe a pagar y la cantidad de caramelos de obsequio.

**2.1. Diagrama de Clase**



**2.2. Diseño de la Aplicación**



**2.3. Código de la Aplicación**

**JFrame Form (Source)**

**Importamos:**

**package Vista;**

**import Clase.COfertaTransporte;**

**import java.awt.Color;**

**import java.awt.Dimension;**

**import javax.swing.DefaultListCellRenderer;**

**import javax.swing.JOptionPane;**

**EN EL PUBLIC CLASS PONER ESTO:**  
 **public class CFormulario extends javax.swing.JFrame {**

**private final COfertaTransporte oferta = new COfertaTransporte();**

**private DefaultListCellRenderer listRenderer;**

**public CFormulario() {**

**initComponents();**

**formulario();**

**inicializarElementos();**

**}**

**private void formulario() {**

**setTitle("Oferta de Transportes");**

**setLocationRelativeTo(null);**

**setResizable(false);**

**setSize(new Dimension(400, 350));**

**getContentPane().setBackground(Color.white);**

**panelResultado.setVisible(false);**

**btnCalcular.setEnabled(false);**

**panelBotones.setVisible(true);**

**}**

**private void validarCampos() {**

**String turno = (String) comboTurno.getSelectedItem();**

**String cantidadStr = txtCantidadPasajes.getText().trim();**

**boolean habilitar = turno != null && !turno.isEmpty() && !cantidadStr.isEmpty();**

**try {**

**int cantidad = Integer.parseInt(cantidadStr);**

**if (cantidad <= 0) {**

**habilitar = false;**

**}**

**} catch (NumberFormatException e) {**

**habilitar = false;**

**}**

**btnCalcular.setEnabled(habilitar);**

**}**

**private void inicializarElementos(){**

**// Centrado del texto de cboCategoria**

**listRenderer = new DefaultListCellRenderer();**

**listRenderer.setHorizontalAlignment(DefaultListCellRenderer.CENTER);**

**comboTurno.setRenderer(listRenderer);**

**this.comboTurno.addItem("Seleccione");**

**this.comboTurno.addItem("Mañana");**

**this.comboTurno.addItem("Tarde");**

**this.comboTurno.addItem("Noche");**

**comboTurno.addItemListener(e -> validarCampos());**

**}**  
  
private void btnCalcularActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here: try { String turnoSeleccionado = (String) comboTurno.getSelectedItem(); if (turnoSeleccionado == null || turnoSeleccionado.isEmpty()) { JOptionPane.showMessageDialog(this, "Seleccione un turno."); return; }

int cantidad = Integer.parseInt(txtCantidadPasajes.getText().trim());  
 if (cantidad <= 0) {  
 JOptionPane.showMessageDialog(this, "Ingrese una cantidad válida mayor que cero.");  
 return;  
 }  
  
 oferta.setTurno(turnoSeleccionado);  
 oferta.setCantidadPasajes(cantidad);  
  
 float importeCompra = oferta.calcularImporteCompra();  
 float descuento = oferta.calcularDescuento();  
 float importePagar = oferta.calcularImportePagar();  
 int caramelos = oferta.calcularCaramelos();  
  
 String resultado = String.format(  
 "Importe de la compra: S/. %.2f\n" +  
 "Descuento aplicado: S/. %.2f\n" +  
 "Importe a pagar: S/. %.2f\n" +  
 "Caramelos de obsequio: %d\n",  
 importeCompra, descuento, importePagar, caramelos);  
  
 txtResultado.setText(resultado);  
 panelResultado.setVisible(true);  
 setSize(new Dimension(400, 420));  
 } catch (NumberFormatException e) {  
 JOptionPane.showMessageDialog(this, "Ingrese valores numéricos válidos.");  
 }  
}   
  
private void btnNuevoActionPerformed(java.awt.event.ActionEvent evt) {   
 // TODO add your handling code here:  
 comboTurno.setSelectedIndex(0);  
 txtCantidadPasajes.setText("");  
 txtResultado.setText("");  
 panelResultado.setVisible(false);  
 setSize(new Dimension(400, 350));  
 btnCalcular.setEnabled(false);  
}   
  
private void btnSalirActionPerformed(java.awt.event.ActionEvent evt) {   
 // TODO add your handling code here:  
 int r = JOptionPane.showConfirmDialog(this, "¿Deseas salir?", "Salir", JOptionPane.YES\_NO\_OPTION);  
 if (r == JOptionPane.YES\_OPTION) {  
 System.exit(0);  
 }  
}  
  
  
private void txtCantidadPasajesKeyReleased(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

validarCampos();

}

private void comboTurnoItemStateChanged(java.awt.event.ItemEvent evt) {

// TODO add your handling code here:

validarCampos();

}

**2.4. Resultado**

