Materia: Programación 2

Nombre: Torres Santiago Irving

Maestra: María del Carmen Santiago Díaz

“Documentación”

Matricula: 201740632

Benemérita Universidad Autónoma De Puebla



El Proyecto se conforma por 13 archivos de java , donde cada uno es una clase y un archivo txt, que nos permite alamacenar la información del almacen.

import java.io.FileNotFoundException;

import java.io.IOException;

import javax.swing.JOptionPane;

import java.io.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.\*;

import javax.swing.JOptionPane;

public class EjecutaMain {

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

Empresa Walmart=new Empresa();

try{

String archivo="Lista\_De\_Productos.txt";

Walmart.inventarioInicial(archivo);

}

catch(FileNotFoundException e){

System.out.println( "no es posiblee ");

JOptionPane.showMessageDialog(null, "No se ha encontrado el archivo");

System.exit(0);

}

VentanaPrincipal Ventana=new VentanaPrincipal(Walmart);

}

}

Este es el Código De la Clase EjecutaMain.java nos permite iniciar el código y en el vamos a ver si el archivo donde esta el inventario existe o no, al construir el objeto de la tienda llamamos su método inventario inicial, que hará la lectura de los datos y los meterá en la clase Empresa.

Ya construida la empresa damos inicio al código creando una clase Ventana principal que da inicio a la parte visual del código.

public class VentanaPrincipal extends JFrame{

int hancho,alto;

Empresa Tienda;

JPanel MenuPanel;

Letrero[] Encabezados= new Letrero[100];

Letrero[] ProdName= new Letrero[100];

Letrero[] ProdPzas= new Letrero[100];

Letrero[] ProdPrice= new Letrero[100];

CajaTexto[] Cajita=new CajaTexto[100];

public VentanaPrincipal(Empresa walmart) {

hancho=1000;

alto=600;

Tienda=walmart;

setTitle("Lista De Productos");

setBounds(50,50,hancho,alto);

setLocationRelativeTo(null);

setDefaultCloseOperation(this.EXIT\_ON\_CLOSE);

ConfigurarPanel();

ConfigurarProductos();

ConfigurarBotonComprar();

ConfigurarEncabezado();

setVisible(true);

MenuPanel.setVisible(true);

}

public void ConfigurarPanel() {

MenuPanel=new JPanel();

// configuramos el panel

MenuPanel.setLayout(null);

MenuPanel.setBackground( Color.blue );

MenuPanel.setBounds(50,50,hancho,alto);

this.getContentPane().add(MenuPanel);

}

public void ConfigurarBotonComprar() {

BotonComprar Comprar=new BotonComprar();

ActionListener oyenteCompra=new ActionListener() {

public void actionPerformed( ActionEvent ae ) {

boolean Validar=true;

boolean Inventario=true;

int carrito=0;

for(int i=1;i<=Tienda.TotProd;i++) {

Cajita[i].CheckNumero();

if( Cajita[i].numero==false )

Validar=false;

if( Cajita[i].Valor>Tienda.ListaP[i].Piezas )

Inventario=false;

carrito+=Cajita[i].Valor;

Tienda.ListaP[i].Ventas=Cajita[i].Valor;

}

Tienda.ProductosCliente=carrito;

int tipoerror=0;

if( Validar==false )

tipoerror=1;

else

if( carrito>5 )

tipoerror=2;

else

if( Inventario==false )

tipoerror=3;

VentanaPago pagar;

VentanaError error;

if( tipoerror==0 ) {

pagar=new VentanaPago( Tienda );

setVisible(false);

}

else

error=new VentanaError(tipoerror);

}

};

Comprar.addActionListener( oyenteCompra );

MenuPanel.add( Comprar );

}

public void ConfigurarEncabezado() {

Encabezados[1]=new Letrero("producto",10,20,100,50);

Encabezados[2]=new Letrero("piezas",300,20,100,50);

Encabezados[3]=new Letrero("precio",450,20,100,50);

Encabezados[4]=new Letrero("compra",600,20,100,50);

for(int i=1;i<=4;i++)

MenuPanel.add( Encabezados[i] );

}

public void ConfigurarProductos() {

for(int i=1;i<=Tienda.TotProd;i++) {

String NombreP=Tienda.ListaP[i].Nombre;

ProdName[i]=new Letrero( NombreP,10,50+i\*40,250,30 );

MenuPanel.add( ProdName[i] );

String piezas=String.valueOf( Tienda.ListaP[i].Piezas );

ProdPzas[i]=new Letrero(piezas,300,50+i\*40,30,30 );

MenuPanel.add( ProdPzas[i] );

String price=String.valueOf( Tienda.ListaP[i].Precio );

ProdPrice[i]=new Letrero(price,450,50+i\*40,30,30 );

MenuPanel.add( ProdPrice[i] );

Cajita[i]=new CajaTexto( 600,50+i\*40 );

MenuPanel.add( Cajita[i] );

}

}

}

Esta es la clase VentanaPrincipal.java en ella mostramos la primer parte donde se muestran los productos y la información del inventario además por cada producto hay una caja de texto donde el usuario inserta la cantidad que desea comprar por cada uno, explicaremos método por método dentro de la clase .

public VentanaPrincipal(Empresa walmart)

este es el constructor aquí configuramos la ventana y el panel para mostrar la infromacion necesaria, además recibimos la información de la empresa. La cual almacenamos en la variable Tienda.

Dentro del mismo método se llaman a otras funciones, las cuales pertenecen a la misma clase , el detalle es que por fines de orden y buena administraciónes se construyen.

public void ConfigurarPanel() {

aquí configuramos el panel asignamos el color, las dimensiones y lo agregamos a la ventana.

public void ConfigurarBotonComprar() {

aquí configuramos el botón que aparece para realizar la compra , construimos el botón le damos dimensiones, y además se programa una subfunción que contiene las siguientes instrucciones al presionar el botón. Esas instrucciones primeramente recopilan la información de las cajas de texto y verifican que no exista un error, en caso de que si hay un error se construye una ventana para mostrar el error correspondiente.

Si no existen errores abre la siguiente ventana donde se va a realizar el pago.

public void ConfigurarEncabezado()

aquí configuramos todas las etiquetas que van a arriba para enlistar los productos , configuramos los textos y metemos los textos al panel.

public void ConfigurarProductos()

aquí configuramos todas las etiquetas que muestran la información de cada producto, lo cual es el nombre del producto , el total de piezas disponibles , el precio de cada pieza y la caja de texto para indicar las unidades deseadas. Todo esto va en forma de lista

import java.awt.Color;

import javax.swing.JFrame;

import javax.swing.JPanel;

public class VentanaError extends JFrame{

int hancho,alto;

JPanel ErrorPanel;

int NumeroError;

public VentanaError (int numerror) {

NumeroError=numerror;

hancho=600;

alto=110;

setVisible(true);

//System.out.println(ArrMapas[1].n+" "+ArrMapas[1].m);

setTitle("Error de compra");

setBounds(200,200,hancho,alto);

setLocationRelativeTo(null);

setDefaultCloseOperation(this.EXIT\_ON\_CLOSE);

ConfigurarPanel();

ConfigurarTexto();

ErrorPanel.setVisible(true);

}

public void ConfigurarPanel() {

ErrorPanel=new JPanel();

// configuramos el panel

ErrorPanel.setLayout(null);

ErrorPanel.setBackground( Color.gray );

ErrorPanel.setBounds(50,50,hancho,alto);

this.getContentPane().add(ErrorPanel);

}

public void ConfigurarTexto() {

String txt="";

if( NumeroError==1 )

txt="Las cajas de texto deben tener unicamente caracteres de digitos";

else

if( NumeroError==2 )

txt="No puedes Comprar mas de 5 productos.";

else

if( NumeroError==3 )

txt="No puedes comprar mas productos de los que existen";

Letrero punto1=new Letrero( txt ,10,10,500,50);

ErrorPanel.add(punto1);

}

}

Esta es la clase VentanaError.java , la cual es llamada por el botón ‘Comprar’ , de la ventana principal en caso de que exista un error, nuevamente explicamos método por método.

public VentanaError (int numerror) {

Este es el contructor, recibimos un entero que nos dice que tipo de error para elegir el texto adecuado dentro la ventana, construimos la ventana definiciendo las dimensiones y el texto para ser mostrado.

public void ConfigurarPanel()

Aquí configuramos el panel donde vamos a meter la etiqueta del error que esta sucediendo, configuramos las dimensiones color para agregar el panel a la ventana

public void ConfigurarTexto()

Aquí con respecto al valor que recibimos vemos que texto se va a mostrar en una etiqueta, para que el usuario pueda ver el error.

import java.awt.Color;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JFrame;

import javax.swing.JPanel;

public class VentanaPago extends JFrame{

int hancho,alto;

JPanel PagoPanel;

Empresa Tienda;

CajaTexto Caja;

public VentanaPago ( Empresa tiendita ) {

Tienda=tiendita;

hancho=550;

alto=200;

setVisible(true);

//System.out.println(ArrMapas[1].n+" "+ArrMapas[1].m);

setTitle("paga por tus productos");

setBounds(200,200,hancho,alto);

setLocationRelativeTo(null);

setDefaultCloseOperation(this.EXIT\_ON\_CLOSE);

ConfigurarPanel();

ConfigurarTexto();

ConfigurarCajaDeTexto();

ConfigurarBotonPago();

PagoPanel.setVisible(true);

}

public void ConfigurarPanel() {

PagoPanel=new JPanel();

// configuramos el panel

PagoPanel.setLayout(null);

PagoPanel.setBackground( Color.gray );

PagoPanel.setBounds(50,50,hancho,alto);

this.getContentPane().add( PagoPanel );

}

public void ConfigurarTexto() {

String txt="";

txt="inserta un monto mayor o igual con el que vas a pagar, monto=";

int suma=0;

for(int i=1;i<=Tienda.TotProd;i++)

suma+=Tienda.ListaP[i].Ventas\*Tienda.ListaP[i].Precio;

txt+=suma;

Tienda.VentaCliente=suma;

Letrero oracion=new Letrero( txt ,10,10,500,50);

PagoPanel.add( oracion );

}

public void ConfigurarCajaDeTexto() {

Caja=new CajaTexto( 10,110 );

PagoPanel.add( Caja );

}

public void ConfigurarBotonPago() {

BotonPago Pagar=new BotonPago(350,100);

ActionListener oyentePago=new ActionListener() {

public void actionPerformed(ActionEvent e) {

Caja.CheckNumero();

Tienda.PagoCliente=Caja.Valor;

if( Caja.numero==true && Caja.Valor>=Tienda.VentaCliente ) {

VentanaTicket ticket= new VentanaTicket(Tienda);

setVisible(false);

}

}

};

Pagar.addActionListener( oyentePago );

PagoPanel.add( Pagar );

}

}

Esta es la clase VentanaPago.java en esta ventana mostramos el costo de todos los productos que se van a comprar e ingresamos el total con el que el cliente va a a pagar para poder darle el cambio en caso de que sea necesario.

public VentanaPago ( Empresa tiendita ) {

Este es el constructor, recibimos la información de la tienda, donde incluye los productos que se van a comprar. Aquí diseñamos el tamaño de la ventana el texto y el botón para poder pagar los productos.

public void ConfigurarPanel() {

configuramos el panel para poder meter el texto y el botón, agregamos el panel a la ventana

public void ConfigurarTexto() {

aquí calculamos el total de los productos a mostrar para insertarlo en el texto, también dentro del texto se aclara que la cantidad a ingresar debe ser mayor o igual. Para realizar el pago adecuadamente.

public void ConfigurarCajaDeTexto()

Configuramos la caja de texto donde el usuario insertar el valor con el que va a pagar, esta caja se agrega al panel

public void ConfigurarBotonPago() {

aquí se configura el botón “Pago”, el cual nos permite abrir la ultima ventana con el ticket y finalizar la compra. Se construye el botón y además se programa las acciones a realizar al ser presionado, el botón no funciona si el formato del numero es erroneo o no es suficiente el dinero para realizar el pago.

import java.awt.Color;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JFrame;

import javax.swing.JPanel;

public class VentanaTicket extends JFrame{

Empresa Tienda;

JPanel TicketPanel;

int hancho,alto;

Letrero[] Encabezados= new Letrero[10];

Letrero[] ProdName= new Letrero[100];

Letrero[] ProdPzas= new Letrero[100];

Letrero[] ProdPrice= new Letrero[100];

Letrero Total,DineroTotal;

Letrero Pago,DineroPago;

Letrero Cambio,DineroCambio;

Letrero Piezas,TotalPiezas;

public VentanaTicket(Empresa tienda) {

Tienda=tienda;

hancho=600;

alto=100+40\*( 5+Tienda.ProductosCliente );

setTitle("Ticket De Compra");

setBounds(50,50,hancho,alto);

setLocationRelativeTo(null);

setDefaultCloseOperation(this.EXIT\_ON\_CLOSE);

ConfigurarPanel();

ConfigurarEncabezado();

ConfigurarProductos();

ConfigurarTextoFinal();

ConfigurarBotonSalir();

ConfigurarBotonVenta();

setVisible( true );

TicketPanel.setVisible(true);

}

public void ConfigurarPanel() {

TicketPanel=new JPanel();

TicketPanel.setLayout(null);

TicketPanel.setBackground( Color.white );

TicketPanel.setBounds(50,50,hancho,alto);

this.getContentPane().add(TicketPanel);

}

public void ConfigurarEncabezado() {

Encabezados[1]=new Letrero("producto",10,20,100,50);

Encabezados[2]=new Letrero("piezas",300,20,100,50);

Encabezados[3]=new Letrero("Costo",450,20,100,50);

for(int i=1;i<=3;i++)

TicketPanel.add( Encabezados[i] );

}

public void ConfigurarProductos() {

for(int i=1,con=1;i<=Tienda.TotProd;i++)

if( Tienda.ListaP[i].Ventas!=0 ) {

String NombreP=Tienda.ListaP[i].Nombre;

ProdName[i]=new Letrero( NombreP,10,50+con\*40,250,30 );

TicketPanel.add( ProdName[i] );

String piezas=String.valueOf( Tienda.ListaP[i].Ventas );

ProdPzas[i]=new Letrero(piezas,300,50+con\*40,50,30 );

TicketPanel.add( ProdPzas[i] );

int res=Tienda.ListaP[i].Ventas\*Tienda.ListaP[i].Precio;

String price=String.valueOf( res );

ProdPrice[i]=new Letrero(price,450,50+con\*40,50,30 );

TicketPanel.add( ProdPrice[i] );

con++;

}

}

public void ConfigurarTextoFinal() {

String Tem;

int pos=40\*Tienda.ProductosCliente+50;

pos+=40;

Tem="Total";

Total=new Letrero(Tem,300,pos,100,30);

TicketPanel.add( Total );

Tem=String.valueOf( Tienda.VentaCliente );

DineroTotal=new Letrero(Tem,450,pos,50,30);

TicketPanel.add( DineroTotal );

pos+=40;

Tem="Pago";

Pago=new Letrero(Tem,300,pos,100,30);

TicketPanel.add( Pago );

Tem=String.valueOf( Tienda.PagoCliente );

DineroPago=new Letrero(Tem,450,pos,50,30);

TicketPanel.add( DineroPago );

pos+=40;

Tem="Cambio";

Cambio=new Letrero(Tem,300,pos,100,30);

TicketPanel.add( Cambio );

int Resta=Tienda.VentaCliente-Tienda.PagoCliente;

Tem=String.valueOf( -Resta );

DineroCambio=new Letrero(Tem,450,pos,50,30);

TicketPanel.add( DineroCambio );

pos+=40;

Tem="Piezas";

Piezas=new Letrero(Tem,300,pos,100,30);

TicketPanel.add( Piezas );

Tem=String.valueOf( Tienda.ProductosCliente );

TotalPiezas=new Letrero(Tem,450,pos,50,30);

TicketPanel.add( TotalPiezas );

}

public void ConfigurarBotonSalir() {

int y=Tienda.ProductosCliente\*40+90;

BotonSalir Salir=new BotonSalir(10,y);

ActionListener oyenteSalir=new ActionListener() {

public void actionPerformed(ActionEvent ae) {

setVisible(false);

System.exit(0);

///exit(0);

}

};

Salir.addActionListener( oyenteSalir );

TicketPanel.add( Salir );

}

public void ConfigurarBotonVenta() {

int y=Tienda.ProductosCliente\*40+90+80;

BotonVenta VentaB=new BotonVenta(10,y);

ActionListener oyeVenta=new ActionListener() {

public void actionPerformed(ActionEvent ae) {

Tienda.Actualizar();

setVisible(false);

VentanaPrincipal Ventana=new VentanaPrincipal(Tienda);

}

};

VentaB.addActionListener( oyeVenta );

TicketPanel.add( VentaB );

}

}

Esta es la clase VentanaTicket.java es la ultima ventana donde muestra los productos que fueron comprados, además de costo y el cambio que será entregado.

public VentanaTicket(Empresa tienda) {

Este es el constructor recibe la información de los productos y los cuales también se vendieron. Aquí se construye la ventana y se llaman a las siguientes funciones para programar lo faltante

public void ConfigurarPanel()

Se configura el panel y se agrega a la ventana, asignamos las dimensiones y lo hacemos visible.

public void ConfigurarEncabezado() {

Se construyen las etiquetas superiores que muestran el texto , para crear 3 columnas “producto”, “cantidad” y precio, se definen las dimensiones por cada texto y se insertan al panel.

public void ConfigurarProductos()

se crean las etiquetas con los productos que se compraron para enlistarlos y mostrar la información adecuada sobre cada producto, el nombre, el total de piezas adquiridas, el costo por esas piezas adquiridas. Los productos llevan orden y se agregan al panel

public void ConfigurarTextoFinal()

se crean los textos finales que están debajo de la lista de productos, donde se mostrara el total de todo lo comprado, el valor con el que pago el cliente y el cambio a dar, cada etiquita se agrega al panel.

public void ConfigurarBotonSalir() {

se crea el botón para cerrar el sistema, y la función que se hará al ser presionado el boton. Simplemente va a detener el código.

public void ConfigurarBotonVenta() {

se crea el botón para recibir a un nuevo cliente, y se programa la función que se ejecuta al ser presionado el botón, si es presionado el botón se llama el metodo actualizar de la tienda para poder restar los elementos vendidos al invetario, y se crea una nueva ventana principal pero con los datos actualizados.

public class Producto {

String Nombre;

int Precio,Piezas,Ventas=0;

public Producto(String nombre,int precio,int piezas){

Nombre=nombre;

Precio=precio;

Piezas=piezas;

Ventas=0;

}

public String toString() {

String Imprimir="";

Imprimir+=Nombre+" ";

Imprimir+=Precio+" ";

Imprimir+=Piezas+" ";

return Imprimir;

}

}

Esta es la clase Producto.java contiene toda la información de cada producto, guarda el nombre, el precio, el total de piezas , y temporalmente las ventas que se hacen de ese producto sobre un cliente.

public Producto(String nombre,int precio,int piezas){

este es el constructor se llama una sola vez, al leer la información del archivo, y se manda esta información leida, que es el nombre, el precio y el total de piezas, las demás variables se les asigna el valor de 0 .

public String toString()

solo se usa para imprimir la información, pero no se utiliza durante la ejecución del sistema , se programo para fines de diseño.

import java.awt.Color;

import java.awt.Font;

import javax.swing.JLabel;

import javax.swing.SwingConstants;

public class Letrero extends JLabel{

public Letrero( String Cadena,int x1,int y1,int largo,int ancho) {

setText(Cadena);

setOpaque(true);

setHorizontalAlignment( SwingConstants.CENTER );

setVerticalAlignment( SwingConstants.CENTER );

setLayout( null );

//setLocation(x1,y1);

setForeground( Color.black );

setFont( new Font("Serif",Font.BOLD ,15) );

setBounds(x1,y1,largo,ancho);

}

}

Esta es la clase Letrero.java se utiliza en muchas ocasiones para el diseño de texto dentro de la ventana.

public Letrero( String Cadena,int x1,int y1,int largo,int ancho) {

únicamente lo conforma el constructor, que recibe el texto, las coordenadas en X y Y, las dimensiones de la etiqueta. Utilizando toda esa información se construye y se inserta a su panel correspondiente.

import java.util.Scanner;

import java.io.BufferedReader;

import java.io.File;

import java.io.FileReader;

import java.io.IOException;

public class Empresa {

int TotProd;

Producto[] ListaP=new Producto[100];

int VentaCliente;

int PagoCliente;

int ProductosCliente;

public Empresa() {

TotProd=0;

VentaCliente=0;

PagoCliente=0;

ProductosCliente=0;

}

public void inventarioInicial(String Archivo) throws IOException {

File f=new File( Archivo );

FileReader rf=new FileReader(f);

BufferedReader br=new BufferedReader( rf );

TotProd=Integer.parseInt( br.readLine() );

String NombreTemp;

int PrecioTemp,PiezasTemp;

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

NombreTemp=br.readLine();

PrecioTemp=Integer.parseInt( br.readLine() );

PiezasTemp=Integer.parseInt( br.readLine() );

ListaP[i]=new Producto(NombreTemp,PrecioTemp,PiezasTemp);

}

}

public void Actualizar() {

for(int i=1;i<=this.TotProd;i++) {

this.ListaP[i].Piezas-=ListaP[i].Ventas;

this.ListaP[i].Ventas=0;

}

VentaCliente=0;

PagoCliente=0;

ProductosCliente=0;

}

}

Esta es la clase Empresa.java guardamos un arreglo con la información de todos los productos y guarda temporalmente información del cliente durante su compra, cosas como el cuanto va a comprar, y con cuanto va a pagar .

public Empresa()

es simplemente el constructor donde se a todas las variables se les asigna el 0

public void inventarioInicial(String Archivo) throws IOException

aquí se va a leer toda la información del inventario, recibe en un string el nombre del archivo , y hará lectura del mismo, para poder almacenar la información de cada producto.

public void Actualizar()

este método se llama cuando se termina la compra y va a haber otra, simplemente por cada producto se le resta al total de piezas el total de ventas , y se limpian algunas variable para darles un nuevo uso .

import javax.swing.JTextField;

public class CajaTexto extends JTextField{

String Palabra;

boolean numero;

int Valor;

public CajaTexto(int x1,int y1) {

setText("0");

setBounds(x1,y1,60,30);

}

public void CheckNumero(){

numero=true;

Palabra=this.getText();

for(int i=0;i<Palabra.length();i++)

if( Palabra.charAt(i)<'0' || Palabra.charAt(i)>'9' )

numero=false;

if( numero==true )

Valor=Integer.parseInt( Palabra );

else

Valor=0;

}

}

Esta es la clase CajaDeTexto.java es para construir todos esos espacios donde se deberá insertar un numero

public CajaTexto(int x1,int y1) {

es el constructor donde recibe las coordenadas en X y Y , dentro del constructor se declara que todas las cajas inicialmente tienen el texto “0”.

public void CheckNumero()

aquí se verifica que el texto ingresado este correcto y sea un numero valido. Toda esa información se almacena en las variables boolean numero; int Valor;

import java.awt.Font;

import javax.swing.JButton;

public class BotonComprar extends JButton{

public BotonComprar() {

setVisible( true );

setText("Realizar Compra");

setEnabled(true);// interaccion boton

setMnemonic('b');//alt+letra

setFont( new Font("Serif",Font.BOLD ,15) );

setBounds(840, 510, 150, 50);

}

}

Esta es la clase BotonComprar.java simplemente lo conforma un constructor donde se configura todo el aspecto visual del botón, que es la fuente, el texto, las dimensiones y su ubicación.

import java.awt.Font;

import javax.swing.JButton;

public class BotonPago extends JButton{

public BotonPago (int x1,int y1) {

setVisible( true );

setText("Pagar");

setEnabled(true);// interaccion boton

setFont( new Font("Serif",Font.BOLD ,15) );

setBounds(x1, y1, 150, 50);

}

}

Esta es la clase BotonVenta.java simplemente lo conforma un constructor recibe las coordenadas X y Y se configura todo el aspecto visual del botón, que es la fuente, el texto, las dimensiones y s ubicación.

import java.awt.Font;

import javax.swing.JButton;

public class BotonSalir extends JButton{

public BotonSalir (int x1,int y1) {

setVisible( true );

setText("Salir");

setEnabled(true);// interaccion boton

setFont( new Font("Serif",Font.BOLD ,15) );

setBounds(x1, y1, 150, 50);

}

}

Esta es la clase BotonSalir.java simplemente lo conforma un constructor recibe las coordenadas X y Y se configura todo el aspecto visual del botón, que es la fuente, el texto, las dimensiones y s ubicación.

import java.awt.Font;

import javax.swing.JButton;

public class BotonVenta extends JButton{

public BotonVenta(int x1,int y1) {

setVisible( true );

setText("Nuevo \n Cliente");

setEnabled(true);// interaccion boton

setFont( new Font("Serif",Font.BOLD ,15) );

setBounds(x1, y1, 150, 50);

}

}

Esta es la clase BotonVenta.java simplemente lo conforma un constructor recibe las coordenadas X y Y se configura todo el aspecto visual del botón, que es la fuente, el texto, las dimensiones y s ubicación.