MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE

NATIONAL TECHNICAL UNIVERSITY

“KHARKOV POLYTECHNICAL INSTITUTE”

LABORATORY WORK № 1

# “Individual Project”

Created by student of 1.КН.201.8e1

Chukwu Irele Omike

Checked by

KHARKIV 2019

# Java software project :

# GUI with jFrame :

# Execution of the work involves the use of JDK since version 8.40.

## 2 Coursework

### 2.1 Study of the problem and study the software technologies needed to solve it

The main task of this stage of -obtaining and analysis of the task for the course work, the study of mathematical methods for solving the problem. In addition, the Unified Modeling Language (UML) tools must be mastered for further design.

### 2.2 Defining software requirements. Analysis and design

According to the UML notation, software requirements are reflected in the use case diagram. It is necessary to distinguish the main options, the options included in them and those that extend them.

At this stage, it is necessary to study the mathematical methods of solving the problem.

To identify the core subjects of the subject area, it is necessary to develop a class diagram from a conceptual point of view. In order to find out the interaction of objects, it is advisable to develop sequence diagrams for different use cases.

Software classes are designed, object interaction and algorithms for individual operations are determined. The structure and composition of packages (modules) that provide the program implementation are being developed. The structure of classes and packages (modules) must meet the requirements of the Model-view-controller (MVC) pattern design: create independent parts such as model (model, classes responsible for processing domain data regardless of user interaction), external view (user interaction tools that include windows and visual components of the graphical user interface) and controller (usually a separate class responsible for the interaction of graphical user interface tools with model classes).

Prior to the implementation of the software, you must determine the format of the output data. Because one of the requirements for the software is to present the output in XML format, a document schema is created. Data sets should also be designed to validate the program and be prepared as XML output files. The simplest options should be checked manually and graphically interpreted manually.

### 2.3 Implementation and testing of the calculated part of the software

According to the previously designed diagram, classes are created to provide the necessary mathematical functions and mathematical methods for solving the problem. It is necessary to provide mechanisms for independent testing of individual classes.

During the implementation of classes that are responsible for the design part of the project, it may be necessary to make changes to the diagrams that were previously designed, including class diagrams.

The implementation of the calculated part of the software ends with the creation of a console application for checking the functionality of all functions that can be performed without graphical tools. The work of the program should be demonstrated on different sets of source data that involve both general behavior and specific cases.

Note: It is desirable for software engineering students to use unit test automation (for example, JUnit).

Console application work, in addition to implementing direct data processing, involves reading data from an XML document and writing it to an XML document, as well as generating a report in HTML or PDF format.

### 2.4 Implementing a graphical user interface application

At this stage, we design the graphical user interface. It is advisable to use visual design software to design the GUI. The main window of the program should contain the main menu, tables and separate rows for entering and editing data, buttons for performing the most important functions, as well as elements for displaying results.

In a separate part of the main window (or in a separate window), a graphical interpretation of the solution of the problem is implemented.

Customization of modules that implement the user interface can be performed autonomously. In places of access to data processing algorithms, so-called "stubs" can be temporarily placed.

This phase is completed by compiling all modules into a single project for further testing.

### 2.5 Testing

Testing the program as a whole is to select several (at least six) variants of the datasets, reflecting the general and partial cases of the solution of the problem, different dimensions, different ranges of numbers (or other data). Testing options are prepared in separate files before implementation.

Negative testing results in one or more development stages being repeated.

### 2.6 Development of the assistance system

The purpose of the assistance system is to provide the user with information about the features and capabilities of the software. The information may be presented as interlinked HTML pages. You can also use other standard help pages.

The homepage should include a brief description of the program and links to other pages. A brief description of mathematical methods (or other data processing methods), hardware and software requirements and user instructions, instructions for using the program, including graphical user interface tools, should be provided on separate pages.

# Project details :

# It involves development of a GUI application for order creation and invoice generation , below is the work flow intended;

# 

# Programing language : java

# Gui : Jframe with spring

# IDE : eclipse 2019

# Jre : jdk.1.8

# Functions Used :

# SalesID()

# Items()

# validatetable()

# calculatePrice()

# getClientsDetails()

# getItemsDetails()

# validateClientDetails()

# clear()

# cleartabel()

# search()

# Forms Used :

# Mainform

# Invoice

# ViewOrder

# login

# Code excuted :

# Mainfform.java:

import java.awt.Color;

import java.awt.EventQueue;

import java.awt.Font;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.FocusAdapter;

import java.awt.event.FocusEvent;

import java.awt.event.KeyAdapter;

import java.awt.event.KeyEvent;

import java.awt.event.MouseAdapter;

import java.awt.event.MouseEvent;

import java.io.BufferedReader;

import java.io.BufferedWriter;

import java.io.FileNotFoundException;

import java.io.FileReader;

import java.io.FileWriter;

import java.io.IOException;

import java.util.ArrayList;

import java.util.List;

import java.util.StringTokenizer;

import java.util.Vector;

import javax.swing.DefaultCellEditor;

import javax.swing.JButton;

import javax.swing.JComboBox;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JPanel;

import javax.swing.JScrollPane;

import javax.swing.JSeparator;

import javax.swing.JTabbedPane;

import javax.swing.JTable;

import javax.swing.JTextArea;

import javax.swing.JTextField;

import javax.swing.SwingConstants;

import javax.swing.border.EmptyBorder;

import javax.swing.border.LineBorder;

import javax.swing.table.DefaultTableModel;

import javax.swing.table.TableColumn;

public class MainForm extends JFrame {

private JPanel contentPane;

public JTextField txtName;

private JTextField textField\_1;

public JTextField txtPhone;

public JTable table;

public static JTextField txtSalesID;

public JTable table\_1;

public JLabel lblSID;

public JLabel lblerror;

public JButton btnAddItem;

public JButton btnSaveOrder;

int sid;

int SalesID;

static String TempID ;

public JLabel lblTotal;

public JTextArea txtAddress;

public JLabel lblresultID;

public JLabel result;

public JLabel SalesNumber;

public JLabel ClientName;

public JLabel ClientPhone;

public JTextArea ClientAddress;

public JLabel Total1;

private JTextField textField;

private JTextField textField\_2;

private JTable table\_2;

public JPanel panel\_8;

public JLabel label\_33;

public JLabel label\_26;

public JLabel label\_28;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

MainForm frame = new MainForm();

frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the frame.

\*/

public MainForm() {

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setBounds(100, 100, 780, 738);

contentPane = new JPanel();

contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

setContentPane(contentPane);

contentPane.setLayout(null);

JTabbedPane tabbedPane = new JTabbedPane(JTabbedPane.TOP);

tabbedPane.setForeground(Color.DARK\_GRAY);

tabbedPane.setBackground(Color.RED);

tabbedPane.setBounds(0, 79, 764, 620);

contentPane.add(tabbedPane);

JPanel panel\_1 = new JPanel();

tabbedPane.addTab("Veiw Sales Order", null, panel\_1, null);

panel\_1.setLayout(null);

JPanel panel\_4 = new JPanel();

panel\_4.setBounds(0, 0, 759, 178);

panel\_1.add(panel\_4);

panel\_4.setLayout(null);

JLabel lblSearchForSales = new JLabel("<html>\r\n<u>Search for Sales Order</u>\r\n<html>");

lblSearchForSales.setFont(new Font("Tahoma", Font.BOLD, 12));

lblSearchForSales.setBounds(307, 11, 164, 14);

panel\_4.add(lblSearchForSales);

JLabel lblNewLabel\_4 = new JLabel("Sales ID:");

lblNewLabel\_4.setFont(new Font("Tahoma", Font.BOLD, 12));

lblNewLabel\_4.setBounds(237, 38, 64, 14);

panel\_4.add(lblNewLabel\_4);

JLabel lblResul = new JLabel("Result for Sales ID:");

lblResul.setFont(new Font("Tahoma", Font.BOLD, 12));

lblResul.setBounds(217, 111, 131, 14);

panel\_4.add(lblResul);

txtSalesID = new JTextField();

txtSalesID.setBounds(317, 36, 115, 20);

panel\_4.add(txtSalesID);

txtSalesID.setColumns(10);

JButton btnSearch = new JButton("Search");

btnSearch.setBackground(Color.BLUE);

btnSearch.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

search();

}

});

btnSearch.setFont(new Font("Tahoma", Font.BOLD, 12));

btnSearch.setBounds(282, 67, 89, 23);

panel\_4.add(btnSearch);

result = new JLabel("<Sales ID>");

result.setFont(new Font("Tahoma", Font.BOLD, 12));

result.setBounds(371, 112, 78, 14);

panel\_4.add(result);

JPanel panel\_5 = new JPanel();

panel\_5.setBorder(new LineBorder(new Color(0, 0, 0)));

panel\_5.setBounds(33, 136, 685, 25);

panel\_4.add(panel\_5);

panel\_5.setLayout(null);

lblresultID = new JLabel("<Sales ID>");

lblresultID.setFont(new Font("Tahoma", Font.BOLD, 12));

lblresultID.setBounds(32, 0, 85, 24);

panel\_5.add(lblresultID);

JSeparator separator\_2 = new JSeparator();

separator\_2.setBackground(new Color(0, 0, 0));

separator\_2.setForeground(new Color(0, 0, 0));

separator\_2.setOrientation(SwingConstants.VERTICAL);

separator\_2.setBounds(168, 0, 14, 25);

panel\_5.add(separator\_2);

JButton btnNewButton\_1 = new JButton("View");

btnNewButton\_1.setFont(new Font("Tahoma", Font.BOLD, 12));

btnNewButton\_1.setBounds(287, 1, 85, 23);

panel\_5.add(btnNewButton\_1);

JSeparator separator\_1 = new JSeparator();

separator\_1.setBackground(new Color(0, 0, 0));

separator\_1.setForeground(new Color(0, 0, 0));

separator\_1.setOrientation(SwingConstants.VERTICAL);

separator\_1.setBounds(452, 0, 14, 25);

panel\_5.add(separator\_1);

JButton btnGenerateInvoice = new JButton("Generate Invoice");

btnGenerateInvoice.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

panel\_8.setEnabled(true);

}

});

btnGenerateInvoice.setFont(new Font("Tahoma", Font.BOLD, 12));

btnGenerateInvoice.setBounds(506, 1, 135, 23);

panel\_5.add(btnGenerateInvoice);

JPanel panel\_6 = new JPanel();

panel\_6.setBounds(0, 177, 759, 415);

panel\_1.add(panel\_6);

panel\_6.setLayout(null);

JLabel lblViewSalesOrder = new JLabel("<html>\r\n<u>View Sales Order</u>\r\n<html>");

lblViewSalesOrder.setFont(new Font("Tahoma", Font.BOLD, 12));

lblViewSalesOrder.setBounds(319, 11, 112, 14);

panel\_6.add(lblViewSalesOrder);

JLabel lblSalesId\_1 = new JLabel("Sales ID:");

lblSalesId\_1.setFont(new Font("Tahoma", Font.BOLD, 12));

lblSalesId\_1.setBounds(30, 39, 73, 14);

panel\_6.add(lblSalesId\_1);

SalesNumber = new JLabel("");

SalesNumber.setBounds(113, 40, 46, 14);

panel\_6.add(SalesNumber);

JLabel lblOrderDate = new JLabel("Order Date:");

lblOrderDate.setFont(new Font("Tahoma", Font.BOLD, 12));

lblOrderDate.setBounds(445, 39, 78, 14);

panel\_6.add(lblOrderDate);

JLabel OrderDate = new JLabel("");

OrderDate.setFont(new Font("Tahoma", Font.BOLD, 12));

OrderDate.setBounds(548, 39, 73, 14);

panel\_6.add(OrderDate);

JLabel lblSalesPerson = new JLabel("Sales Person:");

lblSalesPerson.setFont(new Font("Tahoma", Font.BOLD, 12));

lblSalesPerson.setBounds(30, 73, 84, 14);

panel\_6.add(lblSalesPerson);

JLabel lblClien = new JLabel("Client Name:");

lblClien.setFont(new Font("Tahoma", Font.BOLD, 12));

lblClien.setBounds(30, 98, 84, 14);

panel\_6.add(lblClien);

JLabel lblClientPhone\_1 = new JLabel("Client Phone:");

lblClientPhone\_1.setFont(new Font("Tahoma", Font.BOLD, 12));

lblClientPhone\_1.setBounds(30, 123, 84, 14);

panel\_6.add(lblClientPhone\_1);

JLabel lblClientAddress\_1 = new JLabel("Client Address: ");

lblClientAddress\_1.setFont(new Font("Tahoma", Font.BOLD, 12));

lblClientAddress\_1.setBounds(444, 99, 96, 14);

panel\_6.add(lblClientAddress\_1);

ClientPhone = new JLabel("");

ClientPhone.setBounds(133, 124, 99, 14);

panel\_6.add(ClientPhone);

JLabel SalesPerson = new JLabel("");

SalesPerson.setFont(new Font("Tahoma", Font.BOLD, 12));

SalesPerson.setBounds(133, 74, 99, 14);

panel\_6.add(SalesPerson);

ClientName = new JLabel("");

ClientName.setBounds(133, 99, 99, 14);

panel\_6.add(ClientName);

JPanel panel\_7 = new JPanel();

panel\_7.setLayout(null);

panel\_7.setBounds(10, 191, 739, 155);

panel\_6.add(panel\_7);

JScrollPane scrollPane\_1 = new JScrollPane();

scrollPane\_1.setBounds(0, 0, 739, 136);

panel\_7.add(scrollPane\_1);

table\_1 = new JTable();

table\_1.setModel(new DefaultTableModel(

new Object[][] {

},

new String[] {

"Item ID", "Item Name", "Item Description", "Quantity", "Rate", "Price"

}

));

scrollPane\_1.setViewportView(table\_1);

JLabel label\_8 = new JLabel("Total:");

label\_8.setFont(new Font("Tahoma", Font.BOLD, 13));

label\_8.setBounds(0, 134, 56, 25);

panel\_7.add(label\_8);

Total1 = new JLabel("");

Total1.setFont(new Font("Tahoma", Font.BOLD, 12));

Total1.setBounds(51, 140, 233, 14);

panel\_7.add(Total1);

ClientAddress = new JTextArea();

ClientAddress.setBounds(550, 94, 112, 89);

panel\_6.add(ClientAddress);

JPanel panel = new JPanel();

tabbedPane.addTab("Sales Order Creation", null, panel, null);

JLabel lblNewLabel = new JLabel("<html>\r\n<u>Sales Order Creation<u>\r\n</html>");

lblNewLabel.setBounds(286, 11, 150, 15);

lblNewLabel.setFont(new Font("Tahoma", Font.BOLD, 12));

JLabel lblSalesId = new JLabel("Sales ID:");

lblSalesId.setFont(new Font("Tahoma", Font.BOLD, 12));

lblSalesId.setBounds(24, 42, 59, 14);

lblSID = new JLabel("");

lblSID.setFont(new Font("Tahoma", Font.BOLD, 12));

lblSID.setBounds(129, 42, 130, 14);

panel.setLayout(null);

panel.add(lblNewLabel);

panel.add(lblSalesId);

panel.add(lblSID);

JLabel lblNewLabel\_2 = new JLabel("Client Name:");

lblNewLabel\_2.setFont(new Font("Tahoma", Font.BOLD, 12));

lblNewLabel\_2.setBounds(24, 67, 83, 14);

panel.add(lblNewLabel\_2);

txtName = new JTextField();

txtName.setBounds(129, 67, 130, 20);

panel.add(txtName);

txtName.setColumns(10);

JLabel lblClientAddress = new JLabel("Client Address:");

lblClientAddress.setFont(new Font("Tahoma", Font.BOLD, 12));

lblClientAddress.setBounds(22, 92, 97, 14);

panel.add(lblClientAddress);

txtAddress = new JTextArea();

txtAddress.setBounds(129, 100, 130, 85);

panel.add(txtAddress);

JLabel lblClientPhone = new JLabel("Client Phone:");

lblClientPhone.setFont(new Font("Tahoma", Font.BOLD, 12));

lblClientPhone.setBounds(361, 68, 89, 14);

panel.add(lblClientPhone);

JSeparator separator1 = new JSeparator();

separator1.setForeground(Color.BLACK);

separator1.setBackground(Color.BLACK);

separator1.setSize(739, 15);

separator1.setLocation(10, 226);

panel.add(separator1);

txtPhone = new JTextField();

txtPhone.addKeyListener(new KeyAdapter() {

@Override

public void keyPressed(KeyEvent arg0) {

}

@Override

public void keyTyped(KeyEvent e) {

char caracter = e.getKeyChar();

if(((caracter<'0')||(caracter > '9'))&&(caracter !='\b'))

{

e.consume();

}

}

});

txtPhone.addFocusListener(new FocusAdapter() {

@Override

public void focusGained(FocusEvent arg0) {

}

});

txtPhone.setBounds(476, 67, 130, 20);

panel.add(txtPhone);

txtPhone.setColumns(10);

JLabel lblOrderDetails = new JLabel("<html>\r\n<u>Order Details<u>\r\n</html>");

lblOrderDetails.setFont(new Font("Tahoma", Font.BOLD, 12));

lblOrderDetails.setBounds(315, 241, 89, 14);

panel.add(lblOrderDetails);

JPanel panel\_3 = new JPanel();

panel\_3.setBounds(10, 282, 739, 155);

panel.add(panel\_3);

panel\_3.setLayout(null);

JScrollPane scrollPane = new JScrollPane();

scrollPane.setBounds(0, 0, 739, 136);

panel\_3.add(scrollPane);

table = new JTable();

/\*table.addFocusListener(new FocusAdapter() {

@Override

public void focusLost(FocusEvent arg0) {

DefaultTableModel model= new DefaultTableModel();

model =(DefaultTableModel)table.getModel();

int n = model.getRowCount();

if(model.getValueAt(n, 4)== null)

{

btnAddItem.setEnabled(false);

btnSaveOrder.setEnabled(false);

lblerror.setText("please enter the Quantity");

}

}

});\*/

table.addMouseListener(new MouseAdapter() {

@Override

public void mouseClicked(MouseEvent arg0) {

validatetable();

DefaultTableModel model = (DefaultTableModel) table.getModel();

model = (DefaultTableModel)table.getModel();

int n = model.getRowCount() - 1;

String m = (String) model.getValueAt(n, 4);

if((m != null)&&(m != ""))

{

calculatePrice();

}

}

});

table.setModel(new DefaultTableModel(

new Object[][] {

{null, null, null, null, null, null},

},

new String[] {

"Item ID", "Item Name", "Item Description", "Rate", "Quantity", "price"

}

) {

Class[] columnTypes = new Class[] {

String.class, String.class, String.class, String.class, Object.class, String.class

};

public Class getColumnClass(int columnIndex) {

return columnTypes[columnIndex];

}

});

scrollPane.setViewportView(table);

JLabel lblNewLabel\_3 = new JLabel("Total:");

lblNewLabel\_3.setFont(new Font("Tahoma", Font.BOLD, 13));

lblNewLabel\_3.setBounds(0, 134, 56, 25);

panel\_3.add(lblNewLabel\_3);

lblTotal = new JLabel("0");

lblTotal.setFont(new Font("Tahoma", Font.BOLD, 12));

lblTotal.setBounds(51, 140, 233, 14);

panel\_3.add(lblTotal);

lblerror = new JLabel("");

lblerror.setBounds(497, 140, 181, 14);

panel\_3.add(lblerror);

btnAddItem = new JButton("Add Item");

btnAddItem.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

DefaultTableModel model = (DefaultTableModel) table.getModel();

model = (DefaultTableModel) table.getModel();

Vector row = new Vector();

row.add("");

row.add("");

row.add("");

row.add("");

row.add("");

row.add("");

model.addRow(row);

table.setModel(model);

}

});

btnAddItem.setFont(new Font("Tahoma", Font.BOLD, 12));

btnAddItem.setBounds(315, 481, 104, 23);

panel.add(btnAddItem);

btnSaveOrder = new JButton("Save Order");

btnSaveOrder.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

if (validateClientDetails()==true)

{

getItemsDetails();

getClientsDetails();

SSalesID();

clear();

SalesID();

}

else

{

}

}

});

btnSaveOrder.setFont(new Font("Tahoma", Font.BOLD, 14));

btnSaveOrder.setBounds(296, 515, 154, 51);

panel.add(btnSaveOrder);

panel\_8 = new JPanel();

tabbedPane.addTab("INVOICE", null, panel\_8, null);

panel\_8.setLayout(null);

JPanel panel\_9 = new JPanel();

panel\_9.setBounds(379, 5, 1, 1);

panel\_9.setLayout(null);

panel\_9.setBorder(new EmptyBorder(5, 5, 5, 5));

panel\_8.add(panel\_9);

JSeparator separator = new JSeparator();

separator.setForeground(Color.BLACK);

separator.setBackground(Color.BLACK);

separator.setBounds(10, 147, 653, 15);

panel\_9.add(separator);

JLabel label = new JLabel("<html><u>INVOICE</u></html>");

label.setFont(new Font("Gabriola", Font.BOLD, 19));

label.setBounds(297, -17, 74, 50);

panel\_9.add(label);

JLabel label\_1 = new JLabel("<html><u>NUMERO UNO INTERIORS</U></html>");

label\_1.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_1.setBounds(20, 39, 168, 20);

panel\_9.add(label\_1);

JLabel label\_2 = new JLabel("A-67 ,Redmond Road");

label\_2.setBounds(20, 65, 118, 15);

panel\_9.add(label\_2);

JLabel label\_3 = new JLabel("1st Floor,Mckinsey Building");

label\_3.setBounds(20, 83, 147, 15);

panel\_9.add(label\_3);

JLabel label\_4 = new JLabel("Redling Boulevard");

label\_4.setBounds(20, 100, 100, 15);

panel\_9.add(label\_4);

JLabel label\_5 = new JLabel("CA95064");

label\_5.setBounds(20, 121, 128, 15);

panel\_9.add(label\_5);

JLabel label\_6 = new JLabel("");

label\_6.setBounds(500, 39, 63, 50);

panel\_9.add(label\_6);

JLabel label\_7 = new JLabel("Sales ID:");

label\_7.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_7.setBounds(20, 173, 63, 15);

panel\_9.add(label\_7);

JLabel label\_9 = new JLabel("");

label\_9.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_9.setBounds(80, 173, 108, 15);

panel\_9.add(label\_9);

JLabel label\_10 = new JLabel("Order Date:");

label\_10.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_10.setBounds(336, 168, 82, 20);

panel\_9.add(label\_10);

JLabel label\_11 = new JLabel("");

label\_11.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_11.setBounds(438, 172, 89, 15);

panel\_9.add(label\_11);

JLabel label\_12 = new JLabel("Billing Address:");

label\_12.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_12.setBounds(20, 199, 100, 20);

panel\_9.add(label\_12);

JTextArea textArea = new JTextArea();

textArea.setBounds(130, 199, 100, 93);

panel\_9.add(textArea);

JLabel label\_13 = new JLabel("Shipping Address:");

label\_13.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_13.setBounds(346, 199, 118, 15);

panel\_9.add(label\_13);

JTextArea textArea\_1 = new JTextArea();

textArea\_1.setText("");

textArea\_1.setBounds(474, 198, 100, 93);

panel\_9.add(textArea\_1);

JLabel label\_14 = new JLabel("Contact Number:");

label\_14.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_14.setBounds(20, 298, 108, 20);

panel\_9.add(label\_14);

textField = new JTextField();

textField.setColumns(10);

textField.setBounds(140, 303, 100, 20);

panel\_9.add(textField);

JPanel panel\_10 = new JPanel();

panel\_10.setLayout(null);

panel\_10.setBounds(22, 344, 641, 155);

panel\_9.add(panel\_10);

JScrollPane scrollPane\_2 = new JScrollPane();

scrollPane\_2.setBounds(0, 0, 641, 136);

panel\_10.add(scrollPane\_2);

JLabel label\_15 = new JLabel("Total:");

label\_15.setFont(new Font("Tahoma", Font.BOLD, 13));

label\_15.setBounds(0, 134, 56, 25);

panel\_10.add(label\_15);

JLabel label\_16 = new JLabel("<Auto-Filled Sum of Price columns>");

label\_16.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_16.setBounds(51, 140, 233, 14);

panel\_10.add(label\_16);

JLabel label\_17 = new JLabel("Other terms and conditions:");

label\_17.setBounds(20, 546, 173, 20);

panel\_9.add(label\_17);

JTextArea textArea\_2 = new JTextArea();

textArea\_2.setText("All Items are non-exchangeable and non-returnable.\r\nAll Items are delivered Within 14 days of the sales order.\r\n<Edit>");

textArea\_2.setBounds(224, 521, 415, 57);

panel\_9.add(textArea\_2);

JButton button = new JButton("Print Invoice");

button.setFont(new Font("Tahoma", Font.BOLD, 12));

button.setBounds(251, 597, 120, 23);

panel\_9.add(button);

JPanel panel\_11 = new JPanel();

panel\_11.setLayout(null);

panel\_11.setBorder(new EmptyBorder(5, 5, 5, 5));

panel\_11.setBounds(10, 5, 739, 621);

panel\_8.add(panel\_11);

JSeparator separator\_3 = new JSeparator();

separator\_3.setForeground(Color.BLACK);

separator\_3.setBackground(Color.BLACK);

separator\_3.setBounds(10, 147, 719, 15);

panel\_11.add(separator\_3);

JLabel label\_18 = new JLabel("<html><u>INVOICE</u></html>");

label\_18.setFont(new Font("Gabriola", Font.BOLD, 19));

label\_18.setBounds(297, -17, 74, 50);

panel\_11.add(label\_18);

JLabel label\_19 = new JLabel("<html><u>NUMERO UNO INTERIORS</U></html>");

label\_19.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_19.setBounds(20, 39, 168, 20);

panel\_11.add(label\_19);

JLabel label\_20 = new JLabel("A-67 ,Redmond Road");

label\_20.setBounds(20, 65, 118, 15);

panel\_11.add(label\_20);

JLabel label\_21 = new JLabel("1st Floor,Mckinsey Building");

label\_21.setBounds(20, 83, 147, 15);

panel\_11.add(label\_21);

JLabel label\_22 = new JLabel("Redling Boulevard");

label\_22.setBounds(20, 100, 100, 15);

panel\_11.add(label\_22);

JLabel label\_23 = new JLabel("CA95064");

label\_23.setBounds(20, 121, 128, 15);

panel\_11.add(label\_23);

JLabel label\_24 = new JLabel("");

label\_24.setBounds(500, 39, 63, 50);

panel\_11.add(label\_24);

JLabel label\_25 = new JLabel("Sales ID:");

label\_25.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_25.setBounds(20, 173, 63, 15);

panel\_11.add(label\_25);

label\_26 = new JLabel("");

label\_26.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_26.setBounds(80, 173, 108, 15);

panel\_11.add(label\_26);

JLabel label\_27 = new JLabel("Order Date:");

label\_27.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_27.setBounds(346, 170, 82, 20);

panel\_11.add(label\_27);

label\_28 = new JLabel("");

label\_28.setEnabled(false);

label\_28.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_28.setBounds(474, 173, 89, 15);

panel\_11.add(label\_28);

JLabel label\_29 = new JLabel("Billing Address:");

label\_29.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_29.setBounds(20, 199, 100, 20);

panel\_11.add(label\_29);

JTextArea textArea\_3 = new JTextArea();

textArea\_3.setBounds(130, 199, 100, 93);

panel\_11.add(textArea\_3);

JLabel label\_30 = new JLabel("Shipping Address:");

label\_30.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_30.setBounds(346, 199, 118, 15);

panel\_11.add(label\_30);

JTextArea textArea\_4 = new JTextArea();

textArea\_4.setText("");

textArea\_4.setBounds(474, 198, 100, 93);

panel\_11.add(textArea\_4);

JLabel label\_31 = new JLabel("Contact Number:");

label\_31.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_31.setBounds(20, 298, 108, 20);

panel\_11.add(label\_31);

textField\_2 = new JTextField();

textField\_2.setColumns(10);

textField\_2.setBounds(140, 303, 100, 20);

panel\_11.add(textField\_2);

JPanel panel\_12 = new JPanel();

panel\_12.setLayout(null);

panel\_12.setBounds(22, 344, 707, 155);

panel\_11.add(panel\_12);

JScrollPane scrollPane\_3 = new JScrollPane();

scrollPane\_3.setBounds(0, 0, 707, 136);

panel\_12.add(scrollPane\_3);

table\_2 = new JTable();

table\_2.setModel(new DefaultTableModel(

new Object[][] {

},

new String[] {

"Item ID", "Item Name", "Item Description", "Rate", "Quantity", "Price"

}

));

scrollPane\_3.setViewportView(table\_2);

JLabel label\_32 = new JLabel("Total:");

label\_32.setFont(new Font("Tahoma", Font.BOLD, 13));

label\_32.setBounds(0, 134, 56, 25);

panel\_12.add(label\_32);

label\_33 = new JLabel("<Auto-Filled Sum of Price columns>");

label\_33.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_33.setBounds(51, 140, 233, 14);

panel\_12.add(label\_33);

JLabel label\_34 = new JLabel("Other terms and conditions:");

label\_34.setBounds(20, 546, 173, 20);

panel\_11.add(label\_34);

JTextArea textArea\_5 = new JTextArea();

textArea\_5.setText("All Items are non-exchangeable and non-returnable.\r\nAll Items are delivered Within 14 days of the sales order.\r\n<Edit>");

textArea\_5.setBounds(224, 521, 415, 57);

panel\_11.add(textArea\_5);

JButton button\_1 = new JButton("Print Invoice");

button\_1.setFont(new Font("Tahoma", Font.BOLD, 12));

button\_1.setBounds(251, 597, 120, 23);

panel\_11.add(button\_1);

JPanel panel\_2 = new JPanel();

panel\_2.setBounds(0, 0, 764, 82);

contentPane.add(panel\_2);

panel\_8.hide();

SalesID();

Items();

}

void SalesID()

{

try

{

BufferedReader br = new BufferedReader(new FileReader("SalesID.txt"));

String str;

String id = null;

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

{

id = br.readLine();

}

int n =Integer.parseInt(id);

SalesID = n + 1;

String ns =String.valueOf(SalesID);

lblSID.setText("S"+ns);

TempID =ns;

br.close();

}

catch(Exception e)

{

}

}

void SSalesID()

{

String Sa = TempID;

try {

BufferedWriter bw = new BufferedWriter(new FileWriter("SalesID.txt"));

bw.newLine();

bw.write(Sa);

bw.close();

System.out.println("done");

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

void Items()

{

String[] ItemsID = new String[15];

for(int i = 0;i < 15;i++)

{

int n = i+1;

ItemsID [i] = "T "+String.valueOf(n);

}

//TableColumn itemsID = table.getColumnModel().getColumn(0);

TableColumn itemsID = table.getColumnModel().getColumn(0);

JComboBox itemsIDComboBox = new JComboBox(ItemsID);

itemsID.setCellEditor(new DefaultCellEditor(itemsIDComboBox));

}

void validatetable()

{

DefaultTableModel model = (DefaultTableModel) table.getModel();

model = (DefaultTableModel)table.getModel();

int n =model.getRowCount() - 1;

String itemid = (String) model.getValueAt(n, 0);

switch(itemid)

{

case "T 1":

model.setValueAt("7000", n, 3);

model.setValueAt("Made with Oak", n, 2);

model.setValueAt("Doors", n, 1);

btnAddItem.setEnabled(true);

btnSaveOrder.setEnabled(true);

lblerror.setText("");

break;

case "T 2":

model.setValueAt("5000", n, 3);

model.setValueAt("made with pine",n,2);

model.setValueAt("Chairs", n, 1);

btnAddItem.setEnabled(true);

btnSaveOrder.setEnabled(true);

lblerror.setText("");

break;

case "T 3":

model.setValueAt("Bag of Nails", n, 1);

model.setValueAt("500", n, 3);

model.setValueAt("200 pices of nails", n, 2);

btnAddItem.setEnabled(true);

btnSaveOrder.setEnabled(true);

lblerror.setText("");

break;

case "T 4":

model.setValueAt("Hammer", n , 1);

model.setValueAt("2500", n, 3);

model.setValueAt("general hammer", n, 2);

btnAddItem.setEnabled(true);

btnSaveOrder.setEnabled(true);

lblerror.setText("");

break;

case "T 5":

model.setValueAt("Plier",n, 1);

model.setValueAt("1000", n, 3);

model.setValueAt("Strong and durable", n, 2);

btnAddItem.setEnabled(true);

btnSaveOrder.setEnabled(true);

lblerror.setText("");

break;

case "T 6":

model.setValueAt("Tape Measure", n, 1);

model.setValueAt("650",n,3);

model.setValueAt("dueable", n, 2);

btnAddItem.setEnabled(true);

btnSaveOrder.setEnabled(true);

lblerror.setText("");

break;

case "T 7":

model.setValueAt("Squares", n, 1);

model.setValueAt("1200", n,3);

model.setValueAt("made in sweeden", n, 2);

btnAddItem.setEnabled(true);

btnSaveOrder.setEnabled(true);

lblerror.setText("");

break;

case "T 8":

model.setValueAt("Buble levels", n, 1);

model.setValueAt("500", n, 3);

model.setValueAt("straight and durable", n, 2);

btnAddItem.setEnabled(true);

btnSaveOrder.setEnabled(true);

lblerror.setText("");

break;

case "T 9":

model.setValueAt("Utlity Knife", n, 1);

model.setValueAt("700", n, 3);

model.setValueAt("Sharp and durable", n, 2);

btnAddItem.setEnabled(true);

btnSaveOrder.setEnabled(true);

lblerror.setText("");

break;

case "T 10":

model.setValueAt("Bag of Marking Chalk", n, 1);

model.setValueAt("500", n, 3);

model.setValueAt("100 pieces", n, 2);

btnAddItem.setEnabled(true);

btnSaveOrder.setEnabled(true);

lblerror.setText("");

break;

case "T 11":

model.setValueAt("Screw Driver", n, 1);

model.setValueAt("1200", n, 3);

model.setValueAt("combination of all avalable mouth and sizes", n, 2);

btnAddItem.setEnabled(true);

btnSaveOrder.setEnabled(true);

lblerror.setText("");

break;

case "T 12":

model.setValueAt("Saw", n, 1);

model.setValueAt("2500", n, 3);

model.setValueAt("Sharp and durable hark saw", n, 2);

btnAddItem.setEnabled(true);

btnSaveOrder.setEnabled(true);

lblerror.setText("");

break;

case "T 13":

model.setValueAt("Drill", n, 1);

model.setValueAt("7000", n, 3);

model.setValueAt("strong", n, 2);

btnAddItem.setEnabled(true);

btnSaveOrder.setEnabled(true);

lblerror.setText("");

break;

case "T 14":

model.setValueAt("Mallet", n, 1);

model.setValueAt("2000", n, 3);

model.setValueAt("very Strong", n, 2);

btnAddItem.setEnabled(true);

btnSaveOrder.setEnabled(true);

lblerror.setText("");

break;

case "T 15":

model.setValueAt("Chisel", n, 1);

model.setValueAt("1300", n, 3);

model.setValueAt("strong and durable", n, 2);

btnAddItem.setEnabled(true);

btnSaveOrder.setEnabled(true);

lblerror.setText("");

break;

default :

btnAddItem.setEnabled(false);

btnSaveOrder.setEnabled(false);

lblerror.setText("please select an item");

break;

}

}

void calculatePrice()

{

DefaultTableModel model = (DefaultTableModel) table.getModel();

model = (DefaultTableModel)table.getModel();

int n =model.getRowCount() - 1;

int qnt = 0;

int rate = Integer.parseInt((String) model.getValueAt(n, 3));

qnt = Integer.parseInt((String) model.getValueAt(n, 4));

int price = rate\*qnt;

model.setValueAt(String.valueOf(price), n, 5);

int CP = Integer.parseInt(lblTotal.getText());

int Total = CP + price;

lblTotal.setText(String.valueOf(Total));

}

void getClientsDetails()

{

String SaleID, Name,Address,Phone;

Name = txtName.getText();

Address = txtAddress.getText();

Phone = txtPhone.getText();

SaleID = String.valueOf(SalesID) ;

try {

BufferedWriter bw = new BufferedWriter(new FileWriter("Clientdetails.txt",true));

bw.write(SaleID +"/");

bw.write(Name + "/");

bw.write(Address + "/");

bw.write(Phone +"/");

bw.newLine();

bw.close();

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

void getItemsDetails()

{

DefaultTableModel model = (DefaultTableModel) table.getModel();

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

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

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

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

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

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

String SalesID = lblSID.getText();

String Total = lblTotal.getText();

for(int c = 0 ; c< model.getRowCount() ; c++)

{

itemID.add((String) model.getValueAt(c, 0));

itemName.add((String) model.getValueAt(c,1));

itemDescription.add((String) model.getValueAt(c, 2));

Rate.add((String) model.getValueAt(c, 3));

Quantity.add((String) model.getValueAt(c, 4));

Price.add((String) model.getValueAt(c, 5));

}

try {

BufferedWriter bw = new BufferedWriter(new FileWriter("ItemsBought.txt",true));

for(int n = 0 ; n < itemID.size();n++)

{

bw.write(TempID + "/");

bw.write(itemID.get(n) + "/");

bw.write(itemName.get(n) + "/");

bw.write(itemDescription.get(n) + "/");

bw.write(Rate.get(n) + "/");

bw.write(Quantity.get(n) + "/");

bw.write(Price.get(n)+"/");

bw.newLine();

}

bw.close();

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

try {

BufferedWriter w = new BufferedWriter(new FileWriter("Sales",true));

w.newLine();

w.write(SalesID+"/");

w.write(Total+"/");

w.close();

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

boolean validateClientDetails()

{

String ClientName, ClientAddress, ClientPhone;

ClientName = txtName.getText();

ClientAddress = txtAddress.getText();

ClientPhone = txtPhone.getText();

boolean validate;

if(ClientName.isEmpty())

{

validate = false;

lblerror.setText("Please enter Client name");

}

else

validate = true;

if(ClientAddress.isEmpty())

{

validate = false;

lblerror.setText("Please enter a client Address");

}

else

validate = true;

if(ClientPhone.isEmpty())

{

validate = false;

lblerror.setText("Please enter a client Phone");

}

else

validate = true;

return validate;

}

void clear()

{

DefaultTableModel model = (DefaultTableModel) table.getModel();

txtName.setText("");

txtAddress.setText("");

txtPhone.setText("");

lblTotal.setText("");

model.setRowCount(0);

SalesID();

}

void cleartabel()

{

DefaultTableModel model = (DefaultTableModel) table\_1.getModel();

model.setRowCount(0);

}

void search()

{

String SaID =txtSalesID.getText();

String SalID = null;

String Name = null;

String Address = null;

String Phone = null;

try {

BufferedReader br = new BufferedReader(new FileReader("ClientDetails.txt"));

String str;

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

{

StringTokenizer st = new StringTokenizer(str,"/");

SalID = "S" + st.nextToken();

Name = st.nextToken();

Address = st.nextToken();

Phone = st.nextToken();

if(SaID.equals(SalID))

{

result.setText("Sales Found");

lblresultID.setText(SaID);

SalesNumber.setText(SalID);

ClientName.setText(Name);

ClientPhone.setText(Phone);

ClientAddress.setText(Address);

label\_26.setText(SaID);

textField\_2.setText(Phone);

break;

}

else

{

result.setText("Not found");

lblresultID.setText(SaID +" not found");

}

}

br.close();

}

catch ( IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

try

{

BufferedReader br=new BufferedReader(new FileReader("ItemsBought.txt"));

String str;

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

{

StringTokenizer itemstoken = new StringTokenizer(str,"/");

String FSalesID = "S"+itemstoken.nextToken();

String FItemsID = itemstoken.nextToken();

String FItemsName = itemstoken.nextToken();

String FItemsDesc = itemstoken.nextToken();

String FRate = itemstoken.nextToken();

String FQuantity = itemstoken.nextToken();

String FPrice = itemstoken.nextToken();

Vector row = new Vector();

if(SaID.equals(FSalesID))

{

DefaultTableModel model = (DefaultTableModel) table\_1.getModel();

row.add(FItemsID);

row.add(FItemsName);

row.add(FItemsDesc);

row.add(FRate);

row.add(FQuantity);

row.add(FPrice);

model.addRow(row);

table\_1.setModel(model);

}

if(SaID.equals(FSalesID))

{

Vector rw = new Vector();

DefaultTableModel ma = (DefaultTableModel) table\_2.getModel();

rw.add(FItemsID);

rw.add(FItemsName);

rw.add(FItemsDesc);

rw.add(FRate);

rw.add(FQuantity);

rw.add(FPrice);

ma.addRow(rw);

table\_2.setModel(ma);

}

}

br.close();

}

catch(Exception e)

{

e.printStackTrace();

}

try

{

BufferedReader r = new BufferedReader(new FileReader("Sales.txt"));

String s;

while((s =r.readLine())!=null)

{

StringTokenizer t = new StringTokenizer(s,"/");

String salesid = t.nextToken();

String total = t.nextToken();

if(SaID.equals(salesid))

{

Total1.setText(total);

label\_33.setText(total);

break;

}

}

r.close();

} catch (IOException e) {

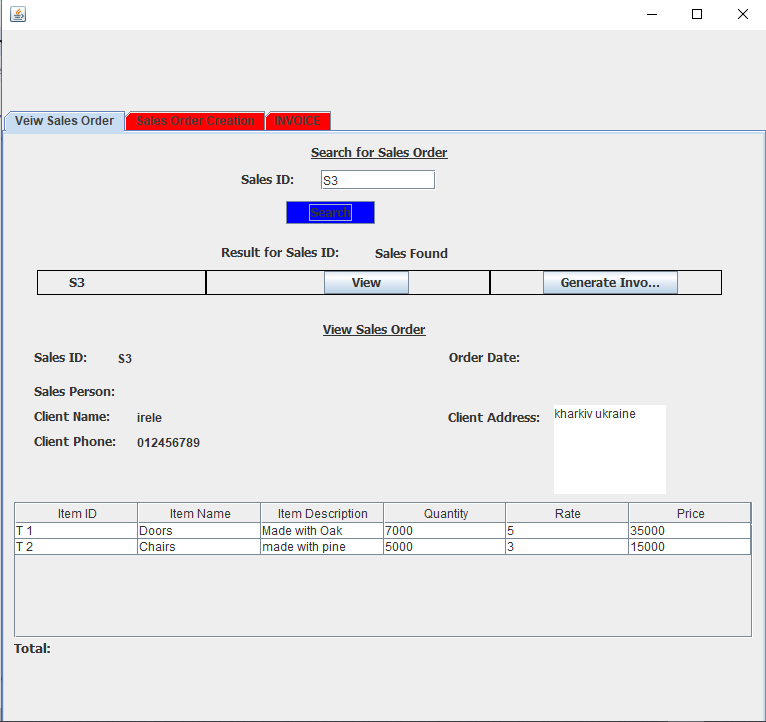
// TODO Auto-generated catch block

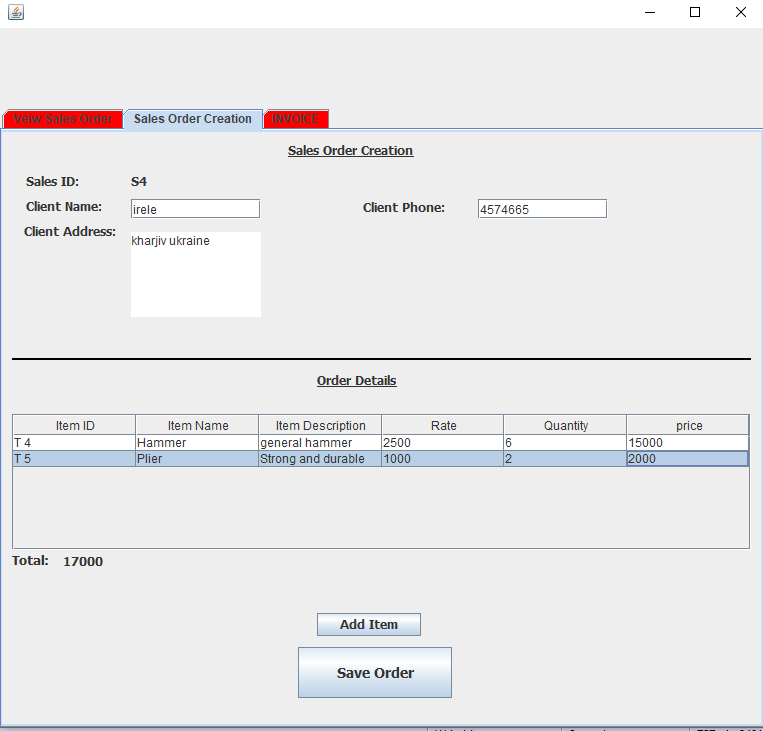
e.printStackTrace();

}

}

}





**Invoice.java :**

import java.awt.EventQueue;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.border.EmptyBorder;

import javax.swing.JSeparator;

import java.awt.Color;

import javax.swing.JLabel;

import java.awt.Font;

import java.io.BufferedReader;

import java.io.FileReader;

import java.io.IOException;

import java.util.StringTokenizer;

import java.util.Vector;

import javax.swing.JTextArea;

import javax.swing.JTextField;

import javax.swing.JScrollPane;

import javax.swing.JTable;

import javax.swing.table.DefaultTableModel;

import javax.swing.JButton;

public class INVOICE extends JFrame {

private JPanel contentPane;

public JTextField textField;

public static JTable table;

public JLabel lblNewLabel;

public JTextArea textArea;

public JLabel label\_2;

public JLabel lblNewLabel\_1;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

INVOICE frame = new INVOICE();

frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the frame.

\*/

public INVOICE() {

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setBounds(100, 100, 720, 659);

contentPane = new JPanel();

contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

setContentPane(contentPane);

contentPane.setLayout(null);

JSeparator separator = new JSeparator();

separator.setForeground(Color.BLACK);

separator.setBackground(Color.BLACK);

separator.setBounds(10, 147, 653, 15);

contentPane.add(separator);

JLabel lblInvoice = new JLabel("<html><u>INVOICE</u></html>");

lblInvoice.setFont(new Font("Gabriola", Font.BOLD, 19));

lblInvoice.setBounds(297, -17, 74, 50);

contentPane.add(lblInvoice);

JLabel lblNumeroUnoInteriors = new JLabel("<html><u>NUMERO UNO INTERIORS</U></html>");

lblNumeroUnoInteriors.setFont(new Font("Tahoma", Font.BOLD, 12));

lblNumeroUnoInteriors.setBounds(20, 39, 168, 20);

contentPane.add(lblNumeroUnoInteriors);

JLabel lblAredmondRoad = new JLabel("A-67 ,Redmond Road");

lblAredmondRoad.setBounds(20, 65, 118, 15);

contentPane.add(lblAredmondRoad);

JLabel lblstFloormckinseyBuilding = new JLabel("1st Floor,Mckinsey Building");

lblstFloormckinseyBuilding.setBounds(20, 83, 147, 15);

contentPane.add(lblstFloormckinseyBuilding);

JLabel lblRedlingBoulevard = new JLabel("Redling Boulevard");

lblRedlingBoulevard.setBounds(20, 100, 100, 15);

contentPane.add(lblRedlingBoulevard);

JLabel lblCa = new JLabel("CA95064");

lblCa.setBounds(20, 121, 128, 15);

contentPane.add(lblCa);

JLabel label = new JLabel("");

label.setBounds(500, 39, 63, 50);

contentPane.add(label);

JLabel lblSalesId = new JLabel("Sales ID:");

lblSalesId.setFont(new Font("Tahoma", Font.BOLD, 12));

lblSalesId.setBounds(20, 173, 63, 15);

contentPane.add(lblSalesId);

lblNewLabel = new JLabel("");

lblNewLabel.setFont(new Font("Tahoma", Font.BOLD, 12));

lblNewLabel.setBounds(80, 173, 108, 15);

contentPane.add(lblNewLabel);

JLabel lblOrderDate = new JLabel("Order Date:");

lblOrderDate.setFont(new Font("Tahoma", Font.BOLD, 12));

lblOrderDate.setBounds(336, 168, 82, 20);

contentPane.add(lblOrderDate);

lblNewLabel\_1 = new JLabel("");

lblNewLabel\_1.setFont(new Font("Tahoma", Font.BOLD, 12));

lblNewLabel\_1.setBounds(438, 172, 89, 15);

contentPane.add(lblNewLabel\_1);

JLabel lblBillingAddress = new JLabel("Billing Address:");

lblBillingAddress.setFont(new Font("Tahoma", Font.BOLD, 12));

lblBillingAddress.setBounds(20, 199, 100, 20);

contentPane.add(lblBillingAddress);

textArea = new JTextArea();

textArea.setBounds(130, 199, 100, 93);

contentPane.add(textArea);

JLabel lblShippingAddress = new JLabel("Shipping Address:");

lblShippingAddress.setFont(new Font("Tahoma", Font.BOLD, 12));

lblShippingAddress.setBounds(346, 199, 118, 15);

contentPane.add(lblShippingAddress);

JTextArea textArea\_1 = new JTextArea();

textArea\_1.setText("");

textArea\_1.setBounds(474, 198, 100, 93);

contentPane.add(textArea\_1);

JLabel lblContactNumber = new JLabel("Contact Number:");

lblContactNumber.setFont(new Font("Tahoma", Font.BOLD, 12));

lblContactNumber.setBounds(20, 298, 108, 20);

contentPane.add(lblContactNumber);

textField = new JTextField();

textField.setBounds(140, 303, 100, 20);

contentPane.add(textField);

textField.setColumns(10);

JPanel panel = new JPanel();

panel.setLayout(null);

panel.setBounds(22, 344, 641, 155);

contentPane.add(panel);

JScrollPane scrollPane = new JScrollPane();

scrollPane.setBounds(0, 0, 641, 136);

panel.add(scrollPane);

table = new JTable();

table.setModel(new DefaultTableModel(

new Object[][] {

},

new String[] {

"Item ID", "Item Name", "Item Description", "Quantity", "Rate", "Price"

}

));

scrollPane.setViewportView(table);

JLabel label\_1 = new JLabel("Total:");

label\_1.setFont(new Font("Tahoma", Font.BOLD, 13));

label\_1.setBounds(0, 134, 56, 25);

panel.add(label\_1);

label\_2 = new JLabel("<Auto-Filled Sum of Price columns>");

label\_2.setFont(new Font("Tahoma", Font.BOLD, 12));

label\_2.setBounds(51, 140, 233, 14);

panel.add(label\_2);

JLabel lblOtherTermsAnd = new JLabel("Other terms and conditions:");

lblOtherTermsAnd.setBounds(20, 546, 173, 20);

contentPane.add(lblOtherTermsAnd);

JTextArea txtrAllItemsAre = new JTextArea();

txtrAllItemsAre.setText("All Items are non-exchangeable and non-returnable.\r\nAll Items are delivered Within 14 days of the sales order.\r\n<Edit>");

txtrAllItemsAre.setBounds(224, 521, 415, 57);

contentPane.add(txtrAllItemsAre);

JButton btnPrintInvoice = new JButton("Print Invoice");

btnPrintInvoice.setFont(new Font("Tahoma", Font.BOLD, 12));

btnPrintInvoice.setBounds(251, 597, 120, 23);

contentPane.add(btnPrintInvoice);

}

void search()

{

MainForm m = new MainForm();

String SaID = MainForm.txtSalesID.getText();

try

{

BufferedReader br=new BufferedReader(new FileReader("ItemsBought.txt"));

String str;

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

{

StringTokenizer itemstoken = new StringTokenizer(str,"/");

String FSalesID = itemstoken.nextToken();

String FItemsID = itemstoken.nextToken();

String FItemsName = itemstoken.nextToken();

String FItemsDesc = itemstoken.nextToken();

String FRate = itemstoken.nextToken();

String FQuantity = itemstoken.nextToken();

String FPrice = itemstoken.nextToken();

Vector row = new Vector();

if(SaID.equals(FSalesID))

{

DefaultTableModel model = (DefaultTableModel) table.getModel();

row.add(FItemsID);

row.add(FItemsName);

row.add(FItemsDesc);

row.add(FRate);

row.add(FQuantity);

row.add(FPrice);

model.addRow(row);

table.setModel(model);

}

}

br.close();

}

catch(Exception e)

{

e.printStackTrace();

}

try

{

BufferedReader r = new BufferedReader(new FileReader("Sales.txt"));

String s;

while((s =r.readLine())!=null)

{

StringTokenizer t = new StringTokenizer(s,"/");

String salesid = t.nextToken();

String total = t.nextToken();

if(SaID.equals(salesid))

{

label\_2.setText(total);

break;

}

}

r.close();

} catch (IOException e) {

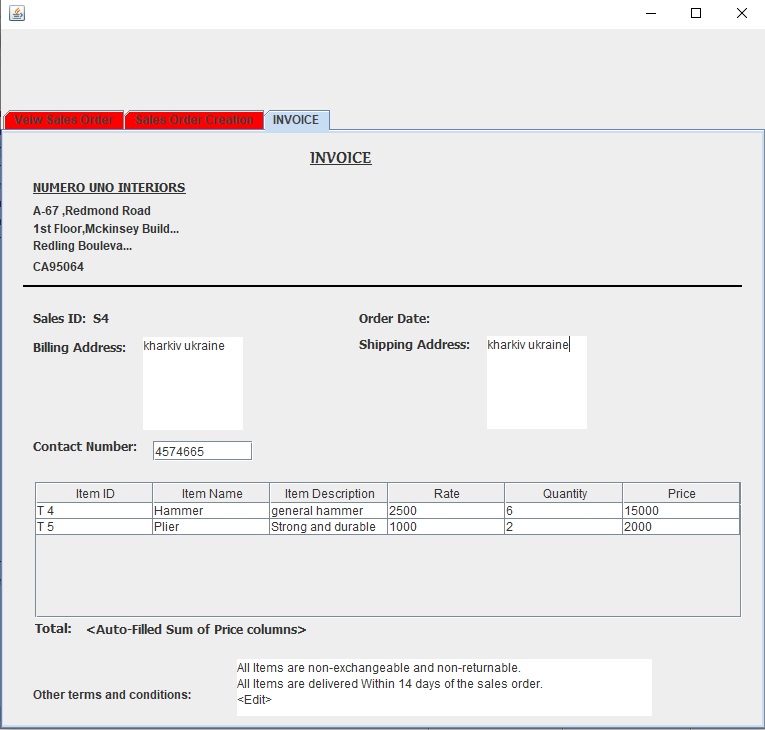
// TODO Auto-generated catch block

e.printStackTrace();

}

}

}



**Login.java:**

import java.awt.Color;

import java.awt.Component;

import java.awt.Dialog.ModalExclusionType;

import java.awt.EventQueue;

import java.awt.FocusTraversalPolicy;

import java.awt.Font;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.BufferedReader;

import java.io.File;

import java.io.FileNotFoundException;

import java.io.FileReader;

import java.io.IOException;

import java.util.StringTokenizer;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JPanel;

import javax.swing.JPasswordField;

import javax.swing.JTextField;

import javax.swing.border.EmptyBorder;

public class Login extends JFrame {

private JPanel contentPane;

private JTextField unname;

private JLabel lblUsername;

private JLabel lblNumeroUno;

private JLabel lblNewLabel;

private JPasswordField passwordField;

private JLabel lblerror;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

Login frame = new Login();

frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the frame.

\*/

public Login() {

setUndecorated(true);

setModalExclusionType(ModalExclusionType.APPLICATION\_EXCLUDE);

setVisible(true);

setRootPaneCheckingEnabled(false);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setBounds(100, 100, 307, 245);

contentPane = new JPanel();

contentPane.setBackground(new Color(25, 25, 112));

contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

setContentPane(contentPane);

contentPane.setLayout(null);

unname = new JTextField();

unname.setToolTipText("<html>\r\ngjjj\r\n</html>");

unname.setBounds(21, 78, 225, 28);

contentPane.add(unname);

unname.setColumns(10);

lblUsername = new JLabel("UserName :");

lblUsername.setForeground(Color.GREEN);

lblUsername.setFont(new Font("Tahoma", Font.BOLD, 12));

lblUsername.setBounds(21, 63, 69, 14);

contentPane.add(lblUsername);

lblNewLabel = new JLabel("PassWord :");

lblNewLabel.setForeground(Color.GREEN);

lblNewLabel.setFont(new Font("Tahoma", Font.BOLD, 12));

lblNewLabel.setBounds(21, 117, 69, 14);

contentPane.add(lblNewLabel);

passwordField = new JPasswordField();

passwordField.setBounds(21, 134, 225, 28);

contentPane.add(passwordField);

JButton btnLogin = new JButton("Login");

btnLogin.setForeground(Color.BLACK);

btnLogin.setBackground(Color.GREEN);

btnLogin.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

try

{

BufferedReader br = new BufferedReader(new FileReader("login.txt"));

String str;

String u= unname.getText();

String p= passwordField.getSelectedText();

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

{

StringTokenizer st = new StringTokenizer(str,"/");

String uname = st.nextToken();

String pass = st.nextToken();

if(u.equals(uname)&&(p.equals(pass)))

{

MainForm m = new MainForm();

m.show();

}

else

{

lblerror.enable();

}

}

br.close();

}

catch(IOException ioe)

{

}

}

});

btnLogin.setFont(new Font("Tahoma", Font.BOLD, 13));

btnLogin.setBounds(21, 189, 89, 23);

contentPane.add(btnLogin);

lblerror = new JLabel("invalid name, username or pass word");

lblerror.setForeground(Color.RED);

lblerror.setBounds(21, 223, 194, 14);

contentPane.add(lblerror);

lblerror.disable();

JPanel panel = new JPanel();

panel.setBackground(Color.GREEN);

panel.setBounds(0, 0, 305, 42);

contentPane.add(panel);

panel.setLayout(null);

lblNumeroUno = new JLabel("<html>\r\n<u>Numero Uno</u>\r\n</html>");

lblNumeroUno.setBounds(85, -23, 183, 65);

lblNumeroUno.setForeground(Color.BLACK);

panel.add(lblNumeroUno);

lblNumeroUno.setFont(new Font("Gabriola", Font.BOLD | Font.ITALIC, 25));

JButton btnCancel = new JButton("Cancel");

btnCancel.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

System.exit(ABORT);

}

});

btnCancel.setBackground(Color.GREEN);

btnCancel.setForeground(Color.BLACK);

btnCancel.setFont(new Font("Tahoma", Font.BOLD, 12));

btnCancel.setBounds(157, 189, 89, 23);

contentPane.add(btnCancel);

//setFocusTraversalPolicy(new FocusTraversalPolicy(new Component[]{contentPane, lblName, name, unname, lblUsername, lblNewLabel, passwordField, btnLogin, lblerror, panel, lblNumeroUno}));

}

}

