**HOME INVENTORY PROJECT**

HomeInventory.java

PROGRAM:

package HomeInventory;

import static HomeInventory.HomeInventory.lastPage;

import static HomeInventory.HomeInventory.numberEntries;

import javax.swing.\*;

import javax.swing.filechooser.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.beans.\*;

import java.io.\*;

import java.util.\*;

import java.text.\*;

import java.awt.print.\*;

private void initComponents() {

jTextField3 = new javax.swing.JTextField();

jToolBar1 = new javax.swing.JToolBar();

jButton1 = new javax.swing.JButton();

jButton2 = new javax.swing.JButton();

jButton3 = new javax.swing.JButton();

jButton4 = new javax.swing.JButton();

jButton5 = new javax.swing.JButton();

jButton6 = new javax.swing.JButton();

jButton7 = new javax.swing.JButton();

jLabel1 = new javax.swing.JLabel();

jLabel2 = new javax.swing.JLabel();

jTextField1 = new javax.swing.JTextField();

jComboBox1 = new javax.swing.JComboBox();

jCheckBox1 = new javax.swing.JCheckBox();

jLabel3 = new javax.swing.JLabel();

jTextField2 = new javax.swing.JTextField();

jLabel4 = new javax.swing.JLabel();

jLabel5 = new javax.swing.JLabel();

jLabel6 = new javax.swing.JLabel();

jLabel7 = new javax.swing.JLabel();

jTextField4 = new javax.swing.JTextField();

jTextField5 = new javax.swing.JTextField();

jTextField6 = new javax.swing.JTextField();

jScrollPane1 = new javax.swing.JScrollPane();

jTextArea1 = new javax.swing.JTextArea();

jTextField7 = new javax.swing.JTextField();

jLabel8 = new javax.swing.JLabel();

jButton8 = new javax.swing.JButton();

jPanel1 = new javax.swing.JPanel();

jButton9 = new javax.swing.JButton();

jButton10 = new javax.swing.JButton();

jButton11 = new javax.swing.JButton();

jButton12 = new javax.swing.JButton();

jButton13 = new javax.swing.JButton();

jButton14 = new javax.swing.JButton();

jButton15 = new javax.swing.JButton();

jButton16 = new javax.swing.JButton();

jButton17 = new javax.swing.JButton();

jButton18 = new javax.swing.JButton();

jButton19 = new javax.swing.JButton();

jButton20 = new javax.swing.JButton();

jButton21 = new javax.swing.JButton();

jButton22 = new javax.swing.JButton();

jButton23 = new javax.swing.JButton();

jButton24 = new javax.swing.JButton();

jButton25 = new javax.swing.JButton();

jButton26 = new javax.swing.JButton();

jButton27 = new javax.swing.JButton();

jButton28 = new javax.swing.JButton();

jButton29 = new javax.swing.JButton();

jButton30 = new javax.swing.JButton();

jButton31 = new javax.swing.JButton();

jButton32 = new javax.swing.JButton();

jButton33 = new javax.swing.JButton();

jButton34 = new javax.swing.JButton();

jPanel2 = new javax.swing.JPanel();

jSeparator1 = new javax.swing.JSeparator();

jButton35 = new javax.swing.JButton();

jTextField3.setText("jTextField3");

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

jToolBar1.setBackground(new java.awt.Color(0, 51, 255));

jToolBar1.setFloatable(false);

jToolBar1.setOrientation(javax.swing.SwingConstants.VERTICAL);

jToolBar1.setRollover(true);

jToolBar1.setAlignmentY(0.0F);

jButton1.setFont(new java.awt.Font("Arial", 0, 12)); // NOI18N

jButton1.setText("New");

jButton1.setFocusable(false);

jButton1.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jButton1.setMaximumSize(new java.awt.Dimension(70, 50));

jButton1.setPreferredSize(new java.awt.Dimension(70, 50));

jButton1.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);

jButton1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

jToolBar1.add(jButton1);

jButton2.setFont(new java.awt.Font("Arial", 0, 12));

jButton2.setText("Delete");

jButton2.setFocusable(false);

jButton2.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jButton2.setMaximumSize(new java.awt.Dimension(70, 50));

jButton2.setPreferredSize(new java.awt.Dimension(70, 50));

jButton2.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);

jButton2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton2ActionPerformed(evt);

}

});

jToolBar1.add(jButton2);

jButton3.setText("Save");

jButton3.setFocusable(false);

jButton3.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jButton3.setMaximumSize(new java.awt.Dimension(70, 50));

jButton3.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);

jButton3.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton3ActionPerformed(evt);

}

});

jToolBar1.add(jButton3);

jButton4.setText("Previous");

jButton4.setFocusable(false);

jButton4.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jButton4.setMaximumSize(new java.awt.Dimension(70, 50));

jButton4.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);

jButton4.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton4ActionPerformed(evt);

}

});

jToolBar1.add(jButton4);

jButton5.setText("Next");

jButton5.setFocusable(false);

jButton5.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jButton5.setMaximumSize(new java.awt.Dimension(70, 50));

jButton5.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);

jButton5.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton5ActionPerformed(evt);

}

});

jToolBar1.add(jButton5);

jButton6.setText("Print");

jButton6.setFocusable(false);

jButton6.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jButton6.setMaximumSize(new java.awt.Dimension(70, 50));

jButton6.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);

jButton6.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton6ActionPerformed(evt);

}

});

jToolBar1.add(jButton6);

jButton7.setText("Exit");

jButton7.setFocusable(false);

jButton7.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jButton7.setMaximumSize(new java.awt.Dimension(70, 50));

jButton7.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);

jButton7.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton7ActionPerformed(evt);

}

});

jToolBar1.add(jButton7);

jLabel1.setFont(new java.awt.Font("Arial", 1, 12));

jLabel1.setText("Inventory Item");

jLabel1.setToolTipText("");

jLabel1.setAlignmentX(1.0F);

jLabel2.setFont(new java.awt.Font("Arial", 1, 12));

jLabel2.setText("Location");

jLabel2.setAlignmentX(1.0F);

jLabel2.setAlignmentY(1.0F);

jTextField1.setFont(new java.awt.Font("Arial", 0, 12));

jTextField1.setAlignmentX(2.0F);

jTextField1.setAlignmentY(0.0F);

jTextField1.setPreferredSize(new java.awt.Dimension(400, 25));

jComboBox1.setEditable(true);

jComboBox1.setFont(new java.awt.Font("Arial", 0, 12));

jComboBox1.setModel(new javax.swing.DefaultComboBoxModel(new String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

jComboBox1.setAlignmentX(2.0F);

jComboBox1.setAlignmentY(1.0F);

jComboBox1.setPreferredSize(new java.awt.Dimension(270, 25));

jCheckBox1.setFont(new java.awt.Font("Arial", 1, 11));

jCheckBox1.setText("Marked?");

jCheckBox1.setAlignmentX(5.0F);

jCheckBox1.setAlignmentY(1.0F);

jLabel3.setFont(new java.awt.Font("Arial", 1, 12));

jLabel3.setText("Serial Number");

jLabel3.setAlignmentX(1.0F);

jLabel3.setAlignmentY(2.0F);

jLabel3.setAutoscrolls(true);

jTextField2.setFont(new java.awt.Font("Arial", 0, 12));

jTextField2.setAlignmentX(2.0F);

jTextField2.setAlignmentY(2.0F);

jTextField2.setPreferredSize(new java.awt.Dimension(270, 25));

jLabel4.setFont(new java.awt.Font("Arial", 1, 12));

jLabel4.setText("Purchase Price");

jLabel4.setAlignmentX(1.0F);

jLabel4.setAlignmentY(3.0F);

jLabel5.setFont(new java.awt.Font("Arial", 1, 12));

jLabel5.setText("Date Purchased");

jLabel5.setAlignmentX(4.0F);

jLabel5.setAlignmentY(3.0F);

jLabel6.setFont(new java.awt.Font("Arial", 1, 12));

jLabel6.setText("Store/Website");

jLabel6.setAlignmentX(1.0F);

jLabel6.setAlignmentY(4.0F);

jLabel7.setFont(new java.awt.Font("Arial", 1, 12));

jLabel7.setText("Note");

jLabel7.setAlignmentX(1.0F);

jLabel7.setAlignmentY(5.0F);

jTextField4.setFont(new java.awt.Font("Arial", 0, 12));

jTextField4.setPreferredSize(new java.awt.Dimension(160, 25));

jTextField5.setFont(new java.awt.Font("Arial", 0, 12));

jTextField5.setText("jTextField5");

jTextField5.setAlignmentX(5.0F);

jTextField5.setAlignmentY(3.0F);

jTextField5.setPreferredSize(new java.awt.Dimension(120, 25));

jTextField6.setFont(new java.awt.Font("Arial", 0, 12));

jTextField6.setAlignmentX(2.0F);

jTextField6.setAlignmentY(4.0F);

jTextField6.setPreferredSize(new java.awt.Dimension(400, 25));

jTextArea1.setEditable(false);

jTextArea1.setBackground(new java.awt.Color(255, 255, 192));

jTextArea1.setColumns(20);

jTextArea1.setFont(new java.awt.Font("Arial", 0, 12));

jTextArea1.setLineWrap(true);

jTextArea1.setRows(5);

jTextArea1.setWrapStyleWord(true);

jTextArea1.setAlignmentX(2.0F);

jTextArea1.setAlignmentY(6.0F);

jTextArea1.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jScrollPane1.setViewportView(jTextArea1);

jTextField7.setFont(new java.awt.Font("Arial", 0, 12));

jTextField7.setPreferredSize(new java.awt.Dimension(400, 25));

jLabel8.setFont(new java.awt.Font("Arial", 1, 12));

jLabel8.setText("Photo");

jLabel8.setAlignmentX(1.0F);

jLabel8.setAlignmentY(6.0F);

jButton8.setFont(new java.awt.Font("Arial", 1, 12);

jButton8.setText("...");

jButton8.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton8ActionPerformed(evt);

}

});

jPanel1.setBackground(new java.awt.Color(255, 255, 51));

jPanel1.setToolTipText("");

jPanel1.setAlignmentX(1.0F);

jPanel1.setAlignmentY(7.0F);

jPanel1.setFont(new java.awt.Font("Arial", 1, 12));

jPanel1.setName("Item Search"); // NOI18N

jPanel1.setLayout(null);

jButton9.setFont(new java.awt.Font("Arial", 1, 12));

jButton9.setText("A");

jPanel1.add(jButton9);

jButton9.setBounds(20, 10, 41, 23);

jButton10.setFont(new java.awt.Font("Arial", 1, 12));

jButton10.setText("B");

jPanel1.add(jButton10);

jButton10.setBounds(60, 10, 41, 23);

jButton11.setFont(new java.awt.Font("Arial", 1, 12));

jButton11.setText("C");

jPanel1.add(jButton11);

jButton11.setBounds(100, 10, 41, 23);

jButton12.setFont(new java.awt.Font("Arial", 1, 12));

jButton12.setText("D");

jPanel1.add(jButton12);

jButton12.setBounds(140, 10, 41, 23);

jButton13.setFont(new java.awt.Font("Arial", 1, 12));

jButton13.setText("E");

jPanel1.add(jButton13);

jButton13.setBounds(180, 10, 39, 23);

jButton14.setFont(new java.awt.Font("Arial", 1, 12));

jButton14.setText("F");

jPanel1.add(jButton14);

jButton14.setBounds(219, 10, 40, 23);

jButton15.setFont(new java.awt.Font("Arial", 1, 12));

jButton15.setText("G");

jPanel1.add(jButton15);

jButton15.setBounds(20, 30, 41, 23);

jButton16.setFont(new java.awt.Font("Arial", 1, 12));

jButton16.setText("H");

jPanel1.add(jButton16);

jButton16.setBounds(60, 30, 41, 23);

jButton17.setFont(new java.awt.Font("Arial", 1, 12));

jButton17.setText("I");

jPanel1.add(jButton17);

jButton17.setBounds(100, 30, 40, 23);

jButton18.setFont(new java.awt.Font("Arial", 1, 12));

jButton18.setText("J");

jPanel1.add(jButton18);

jButton18.setBounds(140, 30, 40, 23);

jButton19.setFont(new java.awt.Font("Arial", 1, 12));

jButton19.setText("K");

jPanel1.add(jButton19);

jButton19.setBounds(180, 30, 41, 23);

jButton20.setFont(new java.awt.Font("Arial", 1, 12));

jButton20.setText("L");

jPanel1.add(jButton20);

jButton20.setBounds(220, 30, 39, 23);

jButton21.setFont(new java.awt.Font("Arial", 1, 12));

jButton21.setText("M");

jPanel1.add(jButton21);

jButton21.setBounds(20, 50, 43, 23);

jButton22.setFont(new java.awt.Font("Arial", 1, 12));

jButton22.setText("N");

jPanel1.add(jButton22);

jButton22.setBounds(60, 50, 41, 23);

jButton23.setFont(new java.awt.Font("Arial", 1, 12));

jButton23.setText("O");

jPanel1.add(jButton23);

jButton23.setBounds(100, 50, 41, 23);

jButton24.setFont(new java.awt.Font("Arial", 1, 12));

jButton24.setText("P");

jPanel1.add(jButton24);

jButton24.setBounds(140, 50, 41, 23);

jButton25.setFont(new java.awt.Font("Arial", 1, 12));

jButton25.setText("Q");

jPanel1.add(jButton25);

jButton25.setBounds(180, 50, 41, 23);

jButton26.setFont(new java.awt.Font("Arial", 1, 12));

jButton26.setText("R");

jPanel1.add(jButton26);

jButton26.setBounds(220, 50, 41, 23);

jButton27.setFont(new java.awt.Font("Arial", 1, 12));

jButton27.setText("S");

jPanel1.add(jButton27);

jButton27.setBounds(20, 70, 41, 23);

jButton28.setFont(new java.awt.Font("Arial", 1, 12));

jButton28.setText("T");

jPanel1.add(jButton28);

jButton28.setBounds(60, 70, 40, 23);

jButton29.setFont(new java.awt.Font("Arial", 1, 12));

jButton29.setText("U");

jPanel1.add(jButton29);

jButton29.setBounds(100, 70, 41, 23);

jButton30.setFont(new java.awt.Font("Arial", 1, 12));

jButton30.setText("V");

jPanel1.add(jButton30);

jButton30.setBounds(140, 70, 40, 23);

jButton31.setFont(new java.awt.Font("Arial", 1, 12));

jButton31.setText("W");

jPanel1.add(jButton31);

jButton31.setBounds(180, 70, 45, 23);

jButton32.setFont(new java.awt.Font("Arial", 1, 12));

jButton32.setText("X");

jPanel1.add(jButton32);

jButton32.setBounds(220, 70, 41, 23);

jButton33.setFont(new java.awt.Font("Arial", 1, 12));

jButton33.setText("Y");

jPanel1.add(jButton33);

jButton33.setBounds(20, 90, 40, 23);

jButton34.setFont(new java.awt.Font("Arial", 1, 12));

jButton34.setText("Z");

jPanel1.add(jButton34);

jButton34.setBounds(59, 90, 40, 23);

jPanel2.setBackground(new java.awt.Color(255, 102, 102));

javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);

jPanel2.setLayout(jPanel2Layout);

jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGap(0, 240, Short.MAX\_VALUE)

);

jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGap(0, 160, Short.MAX\_VALUE)

);

jButton35.setFont(new java.awt.Font("Arial", 0, 12));

jButton35.setText("Item Search");

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGap(56, 56, 56)

.addComponent(jToolBar1, javax.swing.GroupLayout.PREFERRED\_SIZE, 94, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(40, 40, 40)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel1)

.addComponent(jLabel2))

.addGap(62, 62, 62)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGroup(layout.createSequentialGroup()

.addComponent(jComboBox1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(26, 26, 26)

.addComponent(jCheckBox1))))

.addGroup(layout.createSequentialGroup()

.addGap(2, 2, 2)

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 342, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(39, 39, 39)

.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel3)

.addComponent(jLabel4)

.addComponent(jLabel6)

.addGroup(layout.createSequentialGroup()

.addGap(2, 2, 2)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel7, javax.swing.GroupLayout.PREFERRED\_SIZE, 35, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel8))))

.addGap(23, 23, 23))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()

.addComponent(jButton35)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)))

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(18, 18, 18)

.addComponent(jLabel5)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(jTextField5, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jTextField6, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGroup(layout.createSequentialGroup()

.addComponent(jTextField7, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(124, 124, 124)

.addComponent(jSeparator1, javax.swing.GroupLayout.PREFERRED\_SIZE, 50, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(layout.createSequentialGroup()

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 350, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(18, 18, 18)

.addComponent(jButton8)))))

.addContainerGap(192, Short.MAX\_VALUE))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGap(30, 30, 30)

.addComponent(jToolBar1, javax.swing.GroupLayout.PREFERRED\_SIZE, 374, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(layout.createSequentialGroup()

.addGap(35, 35, 35)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel1)

.addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel2)

.addComponent(jComboBox1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jCheckBox1))

.addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel3)

.addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel4)

.addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel5)

.addComponent(jTextField5, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jTextField6, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel6))

.addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel7)

.addComponent(jTextField7, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addComponent(jSeparator1, javax.swing.GroupLayout.PREFERRED\_SIZE, 10, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 35, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jButton8))

.addGap(28, 28, 28))

.addGroup(layout.createSequentialGroup()

.addComponent(jLabel8)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton35)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)))

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 160, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))))

.addContainerGap(120, Short.MAX\_VALUE))

);

jPanel1.getAccessibleContext().setAccessibleName("Item Search");

pack();

}

public class HomeInventory extends javax.swing.JFrame {

JDateChooser dateDateChooser = new JDateChooser();

static final int maximumEntries = 300;

static int numberEntries;

static InventoryItem[] myInventory = new InventoryItem[maximumEntries];

int currentEntry;

static final int entriesPerPage = 2;

static int lastPage;

// frame constructor

public void HomeInventory(){

setTitle("Home Inventory Manager");

setResizable(false);

setDefaultCloseOperation(JFrame.DO\_NOTHING\_ON\_CLOSE);

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent evt)

{

exitForm(evt);

}

});

getContentPane().setLayout(new GridBagLayout());

GridBagConstraints gridConstraints;

jToolBar1.setFloatable(false);

jToolBar1.setBackground(Color.BLUE);

jToolBar1.setOrientation(SwingConstants.VERTICAL);

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 0;

gridConstraints.gridy = 0;

gridConstraints.gridheight = 8;

gridConstraints.fill = GridBagConstraints.VERTICAL;

getContentPane().add(jToolBar1, gridConstraints);

jToolBar1.addSeparator();

Dimension bSize = new Dimension(70, 50);

jButton1.setText("New");

sizeButton(jButton1, bSize);

jButton1.setToolTipText("Add New Item");

jButton1.setHorizontalTextPosition(SwingConstants.CENTER);

jButton1.setVerticalTextPosition(SwingConstants.BOTTOM);

jButton1.setFocusable(false);

jToolBar1.add(jButton2);

jButton2.setText("Delete");

sizeButton(jButton2, bSize);

jButton2.setToolTipText("Delete Current Item");

jButton2.setHorizontalTextPosition(SwingConstants.CENTER);

jButton2.setVerticalTextPosition(SwingConstants.BOTTOM);

jButton2.setFocusable(false);

jToolBar1.add(jButton2);

jButton3.setText("Save");

sizeButton(jButton3, bSize);

jButton3.setToolTipText("Save Current Item");

jButton3.setHorizontalTextPosition(SwingConstants.CENTER);

jButton3.setVerticalTextPosition(SwingConstants.BOTTOM);

jButton3.setFocusable(false);

jToolBar1.add(jButton3);

jToolBar1.addSeparator();

jButton4.setText("Previous");

sizeButton(jButton4, bSize);

jButton4.setToolTipText("Display Previous Item");

jButton4.setHorizontalTextPosition(SwingConstants.CENTER);

jButton4.setVerticalTextPosition(SwingConstants.BOTTOM);

jButton4.setFocusable(false);

jToolBar1.add(jButton4);

jButton5.setText("Next");

sizeButton(jButton5, bSize);

jButton5.setToolTipText("Display Next Item");

jButton5.setHorizontalTextPosition(SwingConstants.CENTER);

jButton5.setVerticalTextPosition(SwingConstants.BOTTOM);

jButton5.setFocusable(false);

jToolBar1.add(jButton5);

jToolBar1.addSeparator();

jButton6.setText("Print");

sizeButton(jButton6, bSize);

jButton6.setToolTipText("Print Inventory List");

jButton6.setHorizontalTextPosition(SwingConstants.CENTER);

jButton6.setVerticalTextPosition(SwingConstants.BOTTOM);

jButton6.setFocusable(false);

jToolBar1.add(jButton6);

jButton7.setText("Exit");

sizeButton(jButton7, bSize);

jButton7.setToolTipText("Exit Program");

jButton7.setFocusable(false);

jToolBar1.add(jButton7);

jLabel1.setText("Inventory Item");

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 1;

gridConstraints.gridy = 0;

gridConstraints.insets = new Insets(10, 10, 0, 10);

gridConstraints.anchor = GridBagConstraints.EAST;

getContentPane().add(jLabel1, gridConstraints);

jTextField1.setPreferredSize(new Dimension(400, 25));

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 2;

gridConstraints.gridy = 0;

gridConstraints.gridwidth = 5;

gridConstraints.insets = new Insets(10, 0, 0, 10);

gridConstraints.anchor = GridBagConstraints.WEST;

getContentPane().add(jTextField1, gridConstraints);

jTextField1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jTextField1ActionPerformed(evt);

}

private void jTextField1ActionPerformed(ActionEvent evt) {

throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

});

jCheckBox1.setText("Marked?");

jCheckBox1.setFocusable(false);

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 5;

gridConstraints.gridy = 1;

gridConstraints.insets = new Insets(10, 10, 0, 0);

gridConstraints.anchor = GridBagConstraints.WEST;

getContentPane().add(jCheckBox1, gridConstraints);

jLabel2.setText("Location");

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 1;

gridConstraints.gridy = 1;

gridConstraints.insets = new Insets(10, 10, 0, 10);

gridConstraints.anchor = GridBagConstraints.EAST;

getContentPane().add(jLabel2, gridConstraints);

jComboBox1.setPreferredSize(new Dimension(270, 25));

jComboBox1.setFont(new Font("Arial", Font.PLAIN, 12));

jComboBox1.setEditable(true);

jComboBox1.setBackground(Color.WHITE);

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 2;

gridConstraints.gridy = 1;

gridConstraints.gridwidth = 3;

gridConstraints.insets = new Insets(10, 0, 0, 10);

gridConstraints.anchor = GridBagConstraints.WEST;

getContentPane().add(jComboBox1, gridConstraints);

jComboBox1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jComboBox1ActionPerformed(evt);

}

private void jComboBox1ActionPerformed(ActionEvent evt) {

throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

});

jLabel3.setText("Serial Number");

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 1;

gridConstraints.gridy = 2;

gridConstraints.insets = new Insets(10, 10, 0, 10);

gridConstraints.anchor = GridBagConstraints.EAST;

getContentPane().add(jLabel3, gridConstraints);

jTextField2.setPreferredSize(new Dimension(270, 25));

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 2;

gridConstraints.gridy = 2;

gridConstraints.gridwidth = 3;

gridConstraints.insets = new Insets(10, 0, 0, 10);

gridConstraints.anchor = GridBagConstraints.WEST;

getContentPane().add(jTextField2, gridConstraints);

jTextField2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jTextField2ActionPerformed(evt);

}

private void jTextField2ActionPerformed(ActionEvent evt) {

throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

});

jLabel4.setText("Purchase Price");

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 1;

gridConstraints.gridy = 3;

gridConstraints.insets = new Insets(10, 10, 0, 10);

gridConstraints.anchor = GridBagConstraints.EAST;

getContentPane().add(jLabel4, gridConstraints);

jTextField4.setPreferredSize(new Dimension(160, 25));

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 2;

gridConstraints.gridy = 3;

gridConstraints.gridwidth = 2;

gridConstraints.insets = new Insets(10, 0, 0, 10);

gridConstraints.anchor = GridBagConstraints.WEST;

getContentPane().add(jTextField4, gridConstraints);

jTextField4.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jTextField4ActionPerformed(evt);

}

private void jTextField4ActionPerformed(ActionEvent evt) {

throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

});

jLabel5.setText("Date Purchased");

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 4;

gridConstraints.gridy = 3;

gridConstraints.insets = new Insets(10, 10, 0, 0);

gridConstraints.anchor = GridBagConstraints.WEST;

getContentPane().add(jLabel5, gridConstraints);

dateDateChooser.setPreferredSize(new Dimension(120, 25));

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 5;

gridConstraints.gridy = 3;

gridConstraints.gridwidth = 2;

gridConstraints.insets = new Insets(10, 0, 0, 10);

gridConstraints.anchor = GridBagConstraints.WEST;

dateDateChooser.addPropertyChangeListener(new PropertyChangeListener()

{

public void propertyChange(PropertyChangeEvent evt)

{

dateDateChooserPropertyChange(evt);

}

});

jLabel6.setText("Store/Website");

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 1;

gridConstraints.gridy = 4;

gridConstraints.insets = new Insets(10, 10, 0, 10);

gridConstraints.anchor = GridBagConstraints.EAST;

getContentPane().add(jLabel6, gridConstraints);

jTextField6.setPreferredSize(new Dimension(400, 25));

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 2;

gridConstraints.gridy = 4;

gridConstraints.gridwidth = 5;

gridConstraints.insets = new Insets(10, 0, 0, 10);

gridConstraints.anchor = GridBagConstraints.WEST;

getContentPane().add(jTextField6, gridConstraints);

jTextField6.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jTextField6ActionPerformed(evt);

}

private void jTextField6ActionPerformed(ActionEvent evt) {

throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

});

jLabel7.setText("Note");

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 1;

gridConstraints.gridy = 5;

gridConstraints.insets = new Insets(10, 10, 0, 10);

gridConstraints.anchor = GridBagConstraints.EAST;

getContentPane().add(jLabel7, gridConstraints);

jTextField7.setPreferredSize(new Dimension(400, 25));

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 2;

gridConstraints.gridy = 5;

gridConstraints.gridwidth = 5;

gridConstraints.insets = new Insets(10, 0, 0, 10);

gridConstraints.anchor = GridBagConstraints.WEST;

getContentPane().add(jTextField7, gridConstraints);

jTextField7.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jTextField7ActionPerformed(evt);

}

private void jTextField7ActionPerformed(ActionEvent evt) {

throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

});

jLabel8.setText("Photo");

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 1;

gridConstraints.gridy = 6;

gridConstraints.insets = new Insets(10, 10, 0, 10);

gridConstraints.anchor = GridBagConstraints.EAST;

getContentPane().add(jLabel8, gridConstraints);

jTextArea1.setPreferredSize(new Dimension(350, 35));

jTextArea1.setFont(new Font("Arial", Font.PLAIN, 12));

jTextArea1.setEditable(false);

jTextArea1.setLineWrap(true);

jTextArea1.setWrapStyleWord(true);

jTextArea1.setBackground(new Color(255, 255, 192));

jTextArea1.setBorder(BorderFactory.createLineBorder(Color.BLACK));

jTextArea1.setFocusable(false);

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 2;

gridConstraints.gridy = 6;

gridConstraints.gridwidth = 4;

gridConstraints.insets = new Insets(10, 0, 0, 10);

gridConstraints.anchor = GridBagConstraints.WEST;

getContentPane().add(jTextArea1, gridConstraints);

jButton8.setText("...");

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 6;

gridConstraints.gridy = 6;

gridConstraints.insets = new Insets(10, 0, 0, 10);

gridConstraints.anchor = GridBagConstraints.WEST;

getContentPane().add(jButton8, gridConstraints);

jPanel1.setPreferredSize(new Dimension(240, 160));

jPanel1.setBorder(BorderFactory.createTitledBorder("Item Search"));

jPanel1.setLayout(new GridBagLayout());

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 1;

gridConstraints.gridy = 7;

gridConstraints.gridwidth = 3;

gridConstraints.insets = new Insets(10, 0, 10, 0);

gridConstraints.anchor = GridBagConstraints.CENTER;

getContentPane().add(jPanel1, gridConstraints);

int x = 0, y = 0;

// create and position 26 buttons

for (int i = 0; i < 26; i++)

{

JButton jButton35 = new JButton();

// set text property

jButton35.setText(String.valueOf((char) (65 + i)));

jButton35.setFont(new Font("Arial", Font.BOLD, 12));

jButton35.setMargin(new Insets(-10, -10, -10, -10));

sizeButton(jButton35, new Dimension(37, 27));

jButton35.setBackground(Color.YELLOW);

jButton35.setFocusable(false);

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = x;

gridConstraints.gridy = y;

jPanel1.add(jButton35, gridConstraints);

x++;

// six buttons per row

if (x % 6 == 0)

{

x = 0;

y++;

}

}

jPanel2.setPreferredSize(new Dimension(240, 160));

gridConstraints = new GridBagConstraints();

gridConstraints.gridx = 4;

gridConstraints.gridy = 7;

gridConstraints.gridwidth = 3;

gridConstraints.insets = new Insets(10, 0, 10, 10);

gridConstraints.anchor = GridBagConstraints.CENTER;

getContentPane().add(jPanel2, gridConstraints);

pack();

Dimension screenSize =Toolkit.getDefaultToolkit().getScreenSize();

setBounds((int) (0.5 \* (screenSize.width - getWidth())), (int) (0.5 \* (screenSize.height -getHeight())), getWidth(), getHeight());

int n;

// open file for entries

try

{

BufferedReader inputFile = new BufferedReader(new FileReader("inventory.txt"));

numberEntries =Integer.valueOf(inputFile.readLine()).intValue();

if (numberEntries != 0)

{

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

{

myInventory[i] = new InventoryItem();

myInventory[i].description = inputFile.readLine();

myInventory[i].location = inputFile.readLine();

myInventory[i].serialNumber = inputFile.readLine();

myInventory[i].marked =Boolean.valueOf(inputFile.readLine()).booleanValue();

myInventory[i].purchasePrice =inputFile.readLine();

myInventory[i].purchaseDate = inputFile.readLine();

myInventory[i].purchaseLocation =inputFile.readLine();

myInventory[i].note = inputFile.readLine();

myInventory[i].photoFile = inputFile.readLine();

}

}

// read in combo box elements

n = Integer.valueOf(inputFile.readLine()).intValue();

if (n != 0)

{

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

{

jComboBox1.addItem(inputFile.readLine());

}

}

inputFile.close();

currentEntry = 1;

showEntry(currentEntry);

}

catch (Exception ex)

{

numberEntries = 0;

currentEntry = 0;

}

if (numberEntries == 0)

{

jButton1.setEnabled(false);

jButton2.setEnabled(false);

jButton5.setEnabled(false);

jButton4.setEnabled(false);

jButton6.setEnabled(false);

}

}

public static void main(String args[]) {

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new HomeInventory().setVisible(true);

}

});

}

private void exitForm(WindowEvent evt)

{

if (JOptionPane.showConfirmDialog(null, "Any unsaved changes will be lost.\nAre yousure you want to exit?", "Exit Program", JOptionPane.YES\_NO\_OPTION,JOptionPane.QUESTION\_MESSAGE) == JOptionPane.NO\_OPTION)

return;

// write entries back to file

try

{

try (PrintWriter outputFile = new PrintWriter(new BufferedWriter(newFileWriter("inventory.txt")))) {

outputFile.println(numberEntries);

if (numberEntries != 0)

{

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

{

outputFile.println(myInventory[i].description);

outputFile.println(myInventory[i].location);

outputFile.println(myInventory[i].serialNumber);

outputFile.println(myInventory[i].marked);

outputFile.println(myInventory[i].purchasePrice);

outputFile.println(myInventory[i].purchaseDate);

outputFile.println(myInventory[i].purchaseLocation);

outputFile.println(myInventory[i].note);

outputFile.println(myInventory[i].photoFile);

}

}

// write combo box entries

outputFile.println(jComboBox1.getItemCount());

if (jComboBox1.getItemCount() != 0)

{

for (int i = 0; i < jComboBox1.getItemCount(); i++)

outputFile.println(jComboBox1.getItemAt(i));

} }

}

catch (Exception ex)

{

}

System.exit(0);

}

private void jButton1ActionPerformed(ActionEvent evt)

{

checkSave();

blankValues();

}

private void jButton2ActionPerformed(ActionEvent evt)

{

if (JOptionPane.showConfirmDialog(null, "Are you sure you want to delete this item?",

"Delete Inventory Item", JOptionPane.YES\_NO\_OPTION,

JOptionPane.QUESTION\_MESSAGE) == JOptionPane.NO\_OPTION)

return;

deleteEntry(currentEntry);

if (numberEntries == 0)

{

currentEntry = 0;

blankValues();

}

else

{

currentEntry--;

if (currentEntry == 0)

currentEntry = 1;

showEntry(currentEntry);

}

}

private void jButton3ActionPerformed(ActionEvent evt)

{

// check for description

jTextField1.setText(jTextField1.getText().trim());

if (jTextField1.getText().equals(""))

{

JOptionPane.showConfirmDialog(null, "Must have item description.", "Error",

JOptionPane.DEFAULT\_OPTION, JOptionPane.ERROR\_MESSAGE);

jTextField1.requestFocus();

return;

}

if (jButton1.isEnabled())

{

// delete edit entry then resave

deleteEntry(currentEntry);

}

// capitalize first letter

String s = jTextField1.getText();

jTextField1.setText(s.substring(0, 1).toUpperCase() + s.substring(1));

numberEntries++;

// determine new current entry location based on description

currentEntry = 1;

if (numberEntries != 1)

{

do

{

if

(jTextField1.getText().compareTo(myInventory[currentEntry - 1].description) < 0)

break;

currentEntry++;

}

while (currentEntry < numberEntries);

}

// move all entries below new value down one position unless at end

if (currentEntry != numberEntries)

{

for (int i = numberEntries; i >= currentEntry + 1; i--)

{

myInventory[i - 1] = myInventory[i - 2];

myInventory[i - 2] = new InventoryItem();

}

}

myInventory[currentEntry - 1] = new InventoryItem();

myInventory[currentEntry - 1].description = jTextField1.getText();

myInventory[currentEntry - 1].location =

jComboBox1.getSelectedItem().toString();

myInventory[currentEntry - 1].marked = jCheckBox1.isSelected();

myInventory[currentEntry - 1].serialNumber = jTextField2.getText();

myInventory[currentEntry - 1].purchasePrice = jTextField4.getText();

myInventory[currentEntry - 1].purchaseDate =

dateToString(dateDateChooser.getDate());

myInventory[currentEntry - 1].purchaseLocation = jTextField6.getText();

myInventory[currentEntry - 1].photoFile = jTextArea1.getText();

myInventory[currentEntry - 1].note = jTextField7.getText();

showEntry(currentEntry);

if (numberEntries < maximumEntries)

jButton1.setEnabled(true);

else

jButton1.setEnabled(false);

jButton2.setEnabled(true);

jButton6.setEnabled(true);

}

private void jButton4ActionPerformed(ActionEvent evt)

{

checkSave();

currentEntry--;

showEntry(currentEntry);

}

private void jButton5ActionPerformed(ActionEvent evt)

{

checkSave();

currentEntry++;

showEntry(currentEntry);

}

private void jButton6ActionPerformed(ActionEvent evt)

{

lastPage = (int) (1 + (numberEntries - 1) / entriesPerPage);

PrinterJob inventoryPrinterJob = PrinterJob.getPrinterJob();

if (inventoryPrinterJob.printDialog())

{

try

{

inventoryPrinterJob.print();

}

catch (PrinterException ex)

{

JOptionPane.showConfirmDialog(null, ex.getMessage(), "Print Error",JOptionPane.DEFAULT\_OPTION, JOptionPane.ERROR\_MESSAGE);

}

}

}

private void jButton7ActionPerformed(ActionEvent evt)

{

exitForm(null);

}

private void jButton8ActionPerformed(ActionEvent evt)

{

JFileChooser openChooser = new JFileChooser();

openChooser.setDialogType(JFileChooser.OPEN\_DIALOG);

openChooser.setDialogTitle("Open Photo File");

openChooser.addChoosableFileFilter(new FileNameExtensionFilter("Photo Files","jpg"));

if (openChooser.showOpenDialog(this) == JFileChooser.APPROVE\_OPTION)

showPhoto(openChooser.getSelectedFile().toString());

}

private void jButton35ActionPerformed(ActionEvent evt)

{

int i;

if (numberEntries == 0)

return;

// search for item letter

String letterClicked = evt.getActionCommand();

i = 0;

do

{

if (myInventory[i].description.substring(0, 1).equals(letterClicked))

{

currentEntry = i + 1;

showEntry(currentEntry);

return;

}

i++;

}

while (i < numberEntries);

JOptionPane.showConfirmDialog(null, "No " + letterClicked + " inventory items.","None Found", JOptionPane.DEFAULT\_OPTION,JOptionPane.INFORMATION\_MESSAGE);

}

private void blankValues()

{

// blank input screen

jButton1.setEnabled(false);

jButton2.setEnabled(false);

jButton3.setEnabled(true);

jButton4.setEnabled(false);

jButton5.setEnabled(false);

jButton6.setEnabled(false);

jTextField1.setText("");

jComboBox1.setSelectedItem("");

jCheckBox1.setSelected(false);

jTextField2.setText("");

jTextField4.setText("");

dateDateChooser.setDate(new Date());

jTextField5.setText("");

jTextField7.setText("");

jTextArea1.setText("");

jPanel2.repaint();

}

private void jTextField1ActionPerformed(ActionEvent evt)

{

jComboBox1.requestFocus();

}

private void jComboBox1ActionPerformed(ActionEvent evt)

{

// If in list - exit method

if (jComboBox1.getItemCount() != 0)

{

for (int i = 0; i < jComboBox1.getItemCount(); i++)

{

if (jComboBox1.getSelectedItem().toString().equals(jComboBox1.getItemAt(i).toString()))

{

jTextField2.requestFocus();

return;

}

}

}

// If not found, add to list box

jComboBox1.addItem(jComboBox1.getSelectedItem());

jTextField2.requestFocus();

}

private void jTextField2ActionPerformed(ActionEvent evt)

{

jTextField4.requestFocus();

}

private void jTextField4ActionPerformed(ActionEvent evt)

{

dateDateChooser.requestFocus();

}

private void dateDateChooserPropertyChange(PropertyChangeEvent evt)

{

jTextField6.requestFocus();

}

private void jTextField6ActionPerformed(ActionEvent evt)

{

jTextField7.requestFocus();

}

private void jTextField7ActionPerformed(ActionEvent evt)

{

jButton8.requestFocus();

}

private void sizeButton(JButton b, Dimension d)

{

b.setPreferredSize(d);

b.setMinimumSize(d);

b.setMaximumSize(d);

}

private void showEntry(int j)

{

// display entry j (1 to numberEntries)

jTextField1.setText(myInventory[j - 1].description);

jComboBox1.setSelectedItem(myInventory[j - 1].location);

jCheckBox1.setSelected(myInventory[j - 1].marked);

jTextField2.setText(myInventory[j - 1].serialNumber);

jTextField4.setText(myInventory[j - 1].purchasePrice);

dateDateChooser.setDate(stringToDate(myInventory[j - 1].purchaseDate));

jTextField5.setText(myInventory[j - 1].purchaseLocation);

jTextField7.setText(myInventory[j - 1].note);

showPhoto(myInventory[j - 1].photoFile);

jButton5.setEnabled(true);

jButton4.setEnabled(true);

if (j == 1)

jButton4.setEnabled(false);

if (j == numberEntries)

jButton5.setEnabled(false);

jTextField1.requestFocus();

}

private Date stringToDate(String s)

{

int m = Integer.parseInt(s.substring(0, 2)) - 1;

int d = Integer.parseInt(s.substring(3, 5));

int y = Integer.parseInt(s.substring(6)) - 1900;

return(new Date(y, m, d));

}

private String dateToString(Date dd)

{

String yString = String.valueOf(dd.getYear() + 1900);

int m = dd.getMonth() + 1;

String mString = new DecimalFormat("00").format(m);

int d = dd.getDate();

String dString = new DecimalFormat("00").format(d);

return(mString + "/" + dString + "/" + yString);

}

private void showPhoto(String photoFile)

{

if (!photoFile.equals(""))

{

try

{

jTextArea1.setText(photoFile);

}

catch (Exception ex)

{

jTextArea1.setText("");

}

}

else

{

jTextArea1.setText("");

}

jPanel2.repaint();

}

private void deleteEntry(int j)

{

// delete entry j

if (j != numberEntries)

{

// move all entries under j up one level

for (int i = j; i < numberEntries; i++)

{

myInventory[i - 1] = new InventoryItem();

myInventory[i - 1] = myInventory[i];

}

}

numberEntries--;

}

private void checkSave()

{

boolean edited = false;

if (!myInventory[currentEntry - 1].description.equals(jTextField1.getText()))

edited = true;

else if (!myInventory[currentEntry -1].location.equals(jComboBox1.getSelectedItem().toString()))

edited = true;

else if (myInventory[currentEntry - 1].marked != jCheckBox1.isSelected())

edited = true;

else if (!myInventory[currentEntry - 1].serialNumber.equals(jTextField2.getText()))

edited = true;

else if (!myInventory[currentEntry - 1].purchasePrice.equals(jTextField4.getText()))

edited = true;

else if (!myInventory[currentEntry -1].purchaseDate.equals(dateToString(dateDateChooser.getDate())))

edited = true;

else if (!myInventory[currentEntry -1].purchaseLocation.equals(jTextField6.getText()))

edited = true;

else if (!myInventory[currentEntry - 1].note.equals(jTextField7.getText()))

edited = true;

else if (!myInventory[currentEntry - 1].photoFile.equals(jTextArea1.getText()))

edited = true;

if (edited)

{

if (JOptionPane.showConfirmDialog(null, "You have edited this item. Do you want tosave the changes?", "Save Item", JOptionPane.YES\_NO\_OPTION,JOptionPane.QUESTION\_MESSAGE) == JOptionPane.YES\_OPTION)

jButton3.doClick();

}

}

private Writer newFileWriter(String inventorytxt) {

throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

private void initComponents() {

throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

}

class jPanel1 extends JPanel

{

public void paintComponent(Graphics g)

{

Graphics2D g2D = (Graphics2D) g;

super.paintComponent(g2D);

// draw border

g2D.setPaint(Color.BLACK);

g2D.draw(new Rectangle2D.Double(0, 0, getWidth() - 1, getHeight() - 1));

Image photoImage;

photoImage = new ImageIcon(HomeInventory.jTextArea1.getText()).getImage();

int w = getWidth();

int h = getHeight();

double rWidth = (double) getWidth() / (double) photoImage.getWidth(null);

double rHeight = (double) getHeight() / (double) photoImage.getHeight(null);

if (rWidth > rHeight)

{

// leave height at display height, change width by amount height is changed

w = (int) (photoImage.getWidth(null) \* rHeight);

}

else

{

// leave width at display width, change height by amount width is changed

h = (int) (photoImage.getHeight(null) \* rWidth);

}

// center in panel

g2D.drawImage(photoImage, (int) (0.5 \* (getWidth() - w)), (int) (0.5 \* (getHeight() -h)), w, h, null);

g2D.dispose();

}

}

class InventoryDocument implements Printable

{

public int print(Graphics g, PageFormat pf, int pageIndex)

{

Graphics2D g2D = (Graphics2D) g;

if ((pageIndex + 1) > HomeInventory.lastPage)

{

return NO\_SUCH\_PAGE;

}

int i, iEnd;

g2D.setFont(new Font("Arial", Font.BOLD, 14));

g2D.drawString("Home Inventory Items - Page " + String.valueOf(pageIndex + 1),

(int) pf.getImageableX(), (int) (pf.getImageableY() + 25));

// get starting y

int dy = (int) g2D.getFont().getStringBounds("S",

g2D.getFontRenderContext()).getHeight();

int y = (int) (pf.getImageableY() + 4 \* dy);

iEnd = HomeInventory.entriesPerPage \* (pageIndex + 1);

if (iEnd > HomeInventory.numberEntries)

iEnd = HomeInventory.numberEntries;

for (i = 0 + HomeInventory.entriesPerPage \* pageIndex; i < iEnd; i++)

{

// dividing line

Line2D.Double dividingLine = new Line2D.Double(pf.getImageableX(), y, pf.getImageableX() + pf.getImageableWidth(), y);

g2D.draw(dividingLine);

y += dy;

g2D.setFont(new Font("Arial", Font.BOLD, 12));

g2D.drawString(HomeInventory.myInventory[i].description, (int) pf.getImageableX(), y);

y += dy;

g2D.setFont(new Font("Arial", Font.PLAIN, 12));

g2D.drawString("Location: " + HomeInventory.myInventory[i].location, (int)(pf.getImageableX() + 25), y);

y += dy;

if (HomeInventory.myInventory[i].marked)

g2D.drawString("Item is marked with identifying information.", (int)(pf.getImageableX() + 25), y);

else

g2D.drawString("Item is NOT marked with identifying information.", (int)(pf.getImageableX() + 25), y);

y += dy;

g2D.drawString("Serial Number: " +HomeInventory.myInventory[i].serialNumber, (int) (pf.getImageableX() + 25), y);

y += dy;

g2D.drawString("Price: $" + HomeInventory.myInventory[i].purchasePrice + ",Purchased on: " + HomeInventory.myInventory[i].purchaseDate, (int) (pf.getImageableX() +25), y);

y += dy;

g2D.drawString("Purchased at: " +HomeInventory.myInventory[i].purchaseLocation, (int) (pf.getImageableX() + 25), y);

y += dy;

g2D.drawString("Note: " + HomeInventory.myInventory[i].note, (int)(pf.getImageableX() + 25), y);

y += dy;

try

{

// maintain original width/height ratio

Image inventoryImage = new ImageIcon(HomeInventory.myInventory[i].photoFile).getImage();

double ratio = (double) (inventoryImage.getWidth(null)) / (double)

inventoryImage.getHeight(null);

g2D.drawImage(inventoryImage, (int) (pf.getImageableX() + 25), y, (int) (100 \*ratio), 100, null);

}

catch (Exception ex)

{

// have place to go in case image file doesn't open

}

y += 2 \* dy + 100;

}

return PAGE\_EXISTS;

}

}

// Variables declaration - do not modify

private javax.swing.JButton jButton1;

private javax.swing.JButton jButton10;

private javax.swing.JButton jButton11;

private javax.swing.JButton jButton12;

private javax.swing.JButton jButton13;

private javax.swing.JButton jButton14;

private javax.swing.JButton jButton15;

private javax.swing.JButton jButton16;

private javax.swing.JButton jButton17;

private javax.swing.JButton jButton18;

private javax.swing.JButton jButton19;

private javax.swing.JButton jButton2;

private javax.swing.JButton jButton20;

private javax.swing.JButton jButton21;

private javax.swing.JButton jButton22;

private javax.swing.JButton jButton23;

private javax.swing.JButton jButton24;

private javax.swing.JButton jButton25;

private javax.swing.JButton jButton26;

private javax.swing.JButton jButton27;

private javax.swing.JButton jButton28;

private javax.swing.JButton jButton29;

private javax.swing.JButton jButton3;

private javax.swing.JButton jButton30;

private javax.swing.JButton jButton31;

private javax.swing.JButton jButton32;

private javax.swing.JButton jButton33;

private javax.swing.JButton jButton34;

private javax.swing.JButton jButton35;

private javax.swing.JButton jButton4;

private javax.swing.JButton jButton5;

private javax.swing.JButton jButton6;

private javax.swing.JButton jButton7;

private javax.swing.JButton jButton8;

private javax.swing.JButton jButton9;

private javax.swing.JCheckBox jCheckBox1;

private javax.swing.JComboBox jComboBox1;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JLabel jLabel5;

private javax.swing.JLabel jLabel6;

private javax.swing.JLabel jLabel7;

private javax.swing.JLabel jLabel8;

private javax.swing.JPanel jPanel1;

private javax.swing.JPanel jPanel2;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JSeparator jSeparator1;

private javax.swing.JTextArea jTextArea1;

private javax.swing.JTextField jTextField1;

private javax.swing.JTextField jTextField2;

private javax.swing.JTextField jTextField3;

private javax.swing.JTextField jTextField4;

private javax.swing.JTextField jTextField5;

private javax.swing.JTextField jTextField6;

private javax.swing.JTextField jTextField7;

private javax.swing.JToolBar jToolBar1;

// End of variables declaration

OUTPUT:

