

```

1 package project3;
2
3 import javax.swing.*;
4 import java.awt.*;
5 import java.awt.event.ActionEvent;
6 import java.awt.event.ActionListener;
7
8 /*****
9  *
10 * Maintains the GUI1024 for the red box rental store
11 *
12 *****/
13 public class GUICarDealer extends JFrame implements ActionListener{
14     /** Holds menu bar */
15     private JMenuBar menus;
16
17     /** menus in the menu bar */
18     private JMenu fileMenu;
19     private JMenu actionMenu;
20
21     /** menu items in each of the menus */
22     private JMenuItem openSerItem;
23     private JMenuItem exitItem;
24     private JMenuItem saveSerItem;
25     private JMenuItem saveTextItem;
26     private JMenuItem boughtCarItem;
27     private JMenuItem boughtTruckItem;
28
29     private JMenuItem saveText;
30     private JMenuItem openText;
31     private JMenuItem boughtScreen;
32     private JMenuItem soldScreen;
33     private JMenuItem overDueScreen;
34
35
36     private JMenuItem soldItem;
37
38     /** Holds the list engine */
39     private ListEngine DList;
40     private JPanel panel;
41
42     /** Holds JListArea */
43     private JTable jListArea;
44
45     /** Scroll pane */
46     //private JScrollPane scrollList;
47
48     /*****
49     *
50     * A constructor that starts a new GUI1024 for the rental store
51     *
52     *****/
53     public GUICarDealer(){
54         //adding menu bar and menu items
55         menus = new JMenuBar();
56         fileMenu = new JMenu("File");

```

```

57     actionMenu = new JMenu("Action");
58     openSerItem = new JMenuItem("Open File");
59     exitItem = new JMenuItem("Exit");
60     saveSerItem = new JMenuItem("Save File");
61     boughtTruckItem = new JMenuItem("Bought at auction a Truck");
62     boughtCarItem = new JMenuItem("Bought at auction a Car");
63
64
65     soldItem = new JMenuItem("Sold Car or Truck");
66     openText = new JMenuItem("Open Text");
67     saveText = new JMenuItem("Save Text");
68     boughtScreen = new JMenuItem("Bought Screen");
69     soldScreen = new JMenuItem("Sold Screen");
70     overDueScreen = new JMenuItem("30 Days overDue Screen");
71
72     //adding items to bar
73     fileMenu.add(openSerItem);
74     fileMenu.add(saveSerItem);
75     fileMenu.add(openText);
76     fileMenu.add(saveText);
77     fileMenu.add(exitItem);
78     fileMenu.add(boughtScreen);
79     fileMenu.add(soldScreen);
80     fileMenu.add(overDueScreen);
81
82     actionMenu.add(boughtTruckItem);
83     actionMenu.add(boughtCarItem);
84     actionMenu.add(soldItem);
85
86     menus.add(fileMenu);
87     menus.add(actionMenu);
88
89     //adding ActionListener
90     openSerItem.addActionListener(this);
91     saveSerItem.addActionListener(this);
92     exitItem.addActionListener(this);
93     boughtTruckItem.addActionListener(this);
94     boughtCarItem.addActionListener(this);
95     soldItem.addActionListener(this);
96     openText.addActionListener(this);
97     saveText.addActionListener(this);
98     boughtScreen.addActionListener(this);
99     soldScreen.addActionListener(this);
100    overDueScreen.addActionListener(this);
101
102    setJMenuBar(menus);
103    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
104
105    panel = new JPanel();
106    DList = new ListEngine();
107    jListArea = new JTable(DList);
108    JScrollPane scrollList = new JScrollPane(jListArea);
109    scrollList.setPreferredSize(new Dimension(800,300));
110    panel.add(scrollList);
111
112    add(panel, BorderLayout.CENTER);

```

```

113
114     setVisible(true);
115     setSize(950,450);
116 }
117
118 /*****
119  *
120  * This method handles event-handling code for the GUI1024
121  *
122  * @param e - Holds the action event parameter
123  *****/
124 public void actionPerformed(ActionEvent e) {
125
126     Object comp = e.getSource();
127
128     if (saveSerItem == comp || saveTextItem == comp) {
129         JFileChooser chooser = new JFileChooser();
130         int status = chooser.showSaveDialog(null);
131         if (status == JFileChooser.APPROVE_OPTION) {
132             String filename = chooser.getSelectedFile().getAbsolutePath();
133             if (saveSerItem == e.getSource())
134                 DList.saveDatabase(filename);
135         }
136     }
137
138     //MenuBar options
139     if(e.getSource() == boughtTruckItem){
140         Auto auto = new Truck();
141         boughtOnDialogTruck dialog = new boughtOnDialogTruck(this, auto);
142         if(dialog.getCloseStatus() == boughtOnDialogCar.OK){
143             DList.add(auto);
144         }
145         DList.updateDisplay(DList.displayValue);
146     }
147
148     if(boughtCarItem == e.getSource()){
149         Auto auto1 = new Car();
150         boughtOnDialogCar dialog = new boughtOnDialogCar(this, auto1);
151         if(dialog.getCloseStatus() == boughtOnDialogCar.OK)
152             DList.add(auto1);
153         DList.updateDisplay(DList.displayValue);
154     }
155
156     if (soldItem == e.getSource()) {
157         int index = jListArea.getSelectedRow();
158         if (index >= 0) {
159             Auto unit = DList.get(index);
160             soldOnDialog dialog = new soldOnDialog(this, unit);
161             DList.updateDisplay(DList.displayValue);
162         }
163         else
164             JOptionPane.showMessageDialog(null, "Please select an auto to sell");
165     }
166
167     if (openText == e.getSource()){
168         DList.loadFromText("saved");

```

```
169     }
170
171     if (saveText == e.getSource()){
172         DList.saveAsText("saved");
173     }
174
175     if (boughtScreen == e.getSource()){
176         DList.updateDisplay(DList.boughtScreen);
177         boughtCarItem.setEnabled(true);
178         boughtTruckItem.setEnabled(true);
179         soldItem.setEnabled(true);
180     }
181
182     if (soldScreen == e.getSource()){
183         DList.updateDisplay(DList.soldScreen);
184         boughtCarItem.setEnabled(false);
185         boughtTruckItem.setEnabled(false);
186         soldItem.setEnabled(false);
187     }
188
189     if (overDueScreen == e.getSource()){
190         DList.updateDisplay(DList.overdueScreen);
191         boughtCarItem.setEnabled(false);
192         boughtTruckItem.setEnabled(false);
193         soldItem.setEnabled(true);
194     }
195     if (exitItem == e.getSource())
196         System.exit(-1);
197 }
198
199
200 public static void main(String[] args) {
201     new GUICarDealer();
202 }
203 }
```