

```

1 package com.mycompany.lifelinebloodbank;
2
3 import java.awt.HeadlessException;
10
12 * To change this license header, choose License Headers in Project Properties.
16
17 /**
18 *
19 * @author hp
20 */
21 public class Hospital extends javax.swing.JFrame {
22
23     /**
24      * Creates new form Hospital
25      */
26     public Hospital() {
27         initComponents();
28     }
29
30     /**
31      * This method is called from within the constructor to initialize the form.
32      * WARNING: Do NOT modify this code. The content of this method is always
33      * regenerated by the Form Editor.
34      */
35     @SuppressWarnings("unchecked")
36     // <editor-fold defaultstate="collapsed" desc="Generated Code">
37     private void initComponents() {
38
39         jPanel2 = new javax.swing.JPanel();
40         buyBlood = new javax.swing.JLabel();
41         jPanel5 = new javax.swing.JPanel();
42         jLabel23 = new javax.swing.JLabel();
43         recipientBloodType = new javax.swing.JComboBox<>();
44         jPanel11 = new javax.swing.JPanel();
45         checkAvailability = new javax.swing.JButton();
46
47         setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
48
49         jPanel2.setBackground(new java.awt.Color(74, 31, 61));
50
51         buyBlood.setFont(new java.awt.Font("Tahoma", 1, 24)); // NOI18N
52         buyBlood.setForeground(new java.awt.Color(255, 255, 255));
53         buyBlood.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
54         buyBlood.setText("Hospital Cart");
55
56         javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);
57         jPanel2.setLayout(jPanel2Layout);
58         jPanel2Layout.setHorizontalGroup(
59             jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
60                 .addGroup(jPanel2Layout.createSequentialGroup()
61                     .addGap(75, 75, 75)
62                     .addComponent(buyBlood, javax.swing.GroupLayout.PREFERRED_SIZE, 399,
63                         javax.swing.GroupLayout.PREFERRED_SIZE)
64                     .addGap(10, 10, 10)
65                     .addComponent(checkAvailability, javax.swing.GroupLayout.PREFERRED_SIZE, 69,
66                         javax.swing.GroupLayout.PREFERRED_SIZE)
67                 )
68             .addContainerGap(10, Short.MAX_VALUE)
69         );
70     }
71 }

```

```

70         .addContainerGap())
71     );
72
73     jPanel15.setBackground(new java.awt.Color(217, 103, 74));
74
75     jLabel23.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
76     jLabel23.setForeground(new java.awt.Color(255, 255, 255));
77     jLabel23.setText("What is Recipient's Blood Type");
78
79     javax.swing.GroupLayout jPanel15Layout = new javax.swing.GroupLayout(jPanel15);
80     jPanel15.setLayout(jPanel15Layout);
81     jPanel15Layout.setHorizontalGroup(
82         jPanel15Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
83             .addGroup(jPanel15Layout.createSequentialGroup()
84                 .addContainerGap()
85                 .addComponent(jLabel23)
86                 .addContainerGap(30, Short.MAX_VALUE))
87         );
88     jPanel15Layout.setVerticalGroup(
89         jPanel15Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
90             .addGroup(jPanel15Layout.createSequentialGroup()
91                 .addContainerGap()
92                 .addComponent(jLabel23)
93                 .addContainerGap(162, Short.MAX_VALUE))
94         );
95
96     recipientBloodType.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] {
97         "+A", "-A", "+O", "-O", "+B", "-B", "+AB", "-AB" }));
98     recipientBloodType.addActionListener(new java.awt.event.ActionListener() {
99         public void actionPerformed(java.awt.event.ActionEvent evt) {
100             recipientBloodTypeActionPerformed(evt);
101         }
102     });
103
104     checkAvailability.setFont(new java.awt.Font("Tahoma", 1, 18)); // NOI18N
105     checkAvailability.setForeground(new java.awt.Color(255, 51, 0));
106     checkAvailability.setText("Check Availability");
107     checkAvailability.addActionListener(new java.awt.event.ActionListener() {
108         public void actionPerformed(java.awt.event.ActionEvent evt) {
109             checkAvailabilityActionPerformed(evt);
110         }
111     });
112
113     javax.swing.GroupLayout jPanel11Layout = new javax.swing.GroupLayout(jPanel11);
114     jPanel11.setLayout(jPanel11Layout);
115     jPanel11Layout.setHorizontalGroup(
116         jPanel11Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
117             .addGroup(jPanel11Layout.createSequentialGroup()
118                 .addContainerGap(59, Short.MAX_VALUE)
119                 .addComponent(checkAvailability, javax.swing.GroupLayout.PREFERRED_SIZE,
120                     181, javax.swing.GroupLayout.PREFERRED_SIZE)
121                 .addGap(43, 43, 43))
122         );
123     jPanel11Layout.setVerticalGroup(
124         jPanel11Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
125             .addGroup(jPanel11Layout.createSequentialGroup()
126                 .addGroup(jPanel11Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
127                     .addGroup(jPanel11Layout.createSequentialGroup()
128                         .addGap(15, 15, 15)
129                         .addComponent(checkAvailability, javax.swing.GroupLayout.PREFERRED_SIZE,
130                             45, javax.swing.GroupLayout.PREFERRED_SIZE)
131                         .addContainerGap(40, Short.MAX_VALUE))
132                     .addContainerGap(40, Short.MAX_VALUE))
133             );

```

```

128
129     javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
130     getContentPane().setLayout(layout);
131     layout.setHorizontalGroup(
132         layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
133         .addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT_SIZE,
134             javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
135         .addGroup(layout.createSequentialGroup()
136             .addComponent(jPanel5, javax.swing.GroupLayout.PREFERRED_SIZE,
137                 javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
138             .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
139                 .addGroup(layout.createSequentialGroup()
140                     .addPreferredGap(
141                         javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE,
142                         Short.MAX_VALUE)
143                     .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
144                         javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
145                     .addContainerGap(56, Short.MAX_VALUE))
146                 .addGroup(layout.createSequentialGroup()
147                     .addGap(48, 48, 48)
148                     .addComponent(recipientBloodType,
149                         javax.swing.GroupLayout.PREFERRED_SIZE, 242, javax.swing.GroupLayout.PREFERRED_SIZE)
150                     .addContainerGap(70, Short.MAX_VALUE))))
151     );
152     layout.setVerticalGroup(
153         layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
154         .addGroup(layout.createSequentialGroup()
155             .addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED_SIZE,
156                 javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
157             .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
158                 .addGroup(layout.createSequentialGroup()
159                     .addGroup(layout.createSequentialGroup()
160                         .addGroup(layout.createSequentialGroup()
161                             .addGap(18, 18, 18)
162                             .addComponent(recipientBloodType,
163                                 javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
164                                 javax.swing.GroupLayout.PREFERRED_SIZE)
165                             .addGap(13, 13, 13)
166                             .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
167                                 javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
168                             .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
169                                 Short.MAX_VALUE))
170                         .addGroup(layout.createSequentialGroup()
171                             .addPreferredGap(
172                                 javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE,
173                                 Short.MAX_VALUE)
174                             .addComponent(jPanel5, javax.swing.GroupLayout.PREFERRED_SIZE,
175                                 javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
176                             .addContainerGap()))
177                     .addGroup(layout.createSequentialGroup()
178                         .addPreferredGap(
179                             javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE,
180                             Short.MAX_VALUE)
181                         .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
182                             javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
183                         .addContainerGap(56, Short.MAX_VALUE)))
184                 .addGroup(layout.createSequentialGroup()
185                     .addGap(48, 48, 48)
186                     .addComponent(recipientBloodType,
187                         javax.swing.GroupLayout.PREFERRED_SIZE, 242, javax.swing.GroupLayout.PREFERRED_SIZE)
188                     .addContainerGap(70, Short.MAX_VALUE))))
189     );
190     pack();
191 } // </editor-fold>
192
193 private void checkAvailabilityActionPerformed(java.awt.event.ActionEvent evt) {
194     // TODO add your handling code here:
195     String bloodType = (String) recipientBloodType.getSelectedItem();
196     final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
197     final String DB_URL = "jdbc:mysql://localhost:3306/lifelinebloodbank?
198         autoReconnect=true&useSSL=false";
199     final String user = "root";
200     final String password = "Mysql@123";

```

```

173     String query;
174     Connection con = null;
175     PreparedStatement psmt = null;
176     int n = 0;
177     try{
178         try {
179             Class.forName(JDBC_DRIVER);
180
181         } catch (ClassNotFoundException ex) {
182             // Logger.getLogger(DonorPage.class.getName()).log
183             (Level.SEVERE, null, ex);
184             JOptionPane.showMessageDialog(null,ex);
185         } catch (Exception e) {
186             JOptionPane.showMessageDialog(null,e);
187         }
188         con = DriverManager.getConnection(DB_URL,user,password);
189         query = "select count(donorBloodType) from donor where donorBloodType =
190         '"+bloodType+"' GROUP BY donorBloodType;";
191         psmt = con.prepareStatement(query);
192         psmt.executeUpdate();
193         ResultSet rs = psmt.executeQuery();
194         if(rs.next()) {
195             n = rs.getInt("count(donorBloodType)");
196         }
197         if (n > 0) {
198             Object[] options1 = {"Yes","No"};
199
200             int selection = JOptionPane.showOptionDialog(null,
201                 "Yes it is available. Do you want to buy blood packets?",
202                 "Blood cart",
203                 JOptionPane.YES_NO_OPTION,
204                 JOptionPane.PLAIN_MESSAGE,
205                 null,
206                 options1,
207                 null);
208             if(selection == JOptionPane.YES_OPTION){
209                 BloodCart bc = new BloodCart();
210                 bc.run();
211             } else {
212                 JOptionPane.showMessageDialog(null,"No. of packets available: "+ n);
213             }
214
215         }catch (HeadlessException | SQLException e) {
216             JOptionPane.showMessageDialog(null,e);
217         }
218     }
219
220
221     private void recipientBloodTypeActionPerformed(java.awt.event.ActionEvent evt) {
222         // TODO add your handling code here:
223     }
224
225     /**
226     * @param args the command line arguments
227     */
228     public static void run() {
229         /* Set the Nimbus look and feel */
230         //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code
231         (optional) ">
232         /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look

```

```
and feel.
232     * For details see
    http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
233     */
234     try {
235         for (javax.swing.UIManager.LookAndFeelInfo info :
    javax.swing.UIManager.getInstalledLookAndFeels()) {
236             if ("Nimbus".equals(info.getName())) {
237                 javax.swing.UIManager.setLookAndFeel(info.getClassName());
238                 break;
239             }
240         }
241     } catch (ClassNotFoundException ex) {
242         java.util.logging.Logger.getLogger(Hospital.class.getName()).log
    (java.util.logging.Level.SEVERE, null, ex);
243     } catch (InstantiationException ex) {
244         java.util.logging.Logger.getLogger(Hospital.class.getName()).log
    (java.util.logging.Level.SEVERE, null, ex);
245     } catch (IllegalAccessException ex) {
246         java.util.logging.Logger.getLogger(Hospital.class.getName()).log
    (java.util.logging.Level.SEVERE, null, ex);
247     } catch (javax.swing.UnsupportedLookAndFeelException ex) {
248         java.util.logging.Logger.getLogger(Hospital.class.getName()).log
    (java.util.logging.Level.SEVERE, null, ex);
249     }
250     //</editor-fold>
251
252     /* Create and display the form */
253     java.awt.EventQueue.invokeLater(new Runnable() {
254         public void run() {
255             new Hospital().setVisible(true);
256         }
257     });
258 }
259
260 // Variables declaration - do not modify
261 private javax.swing.JLabel buyBlood;
262 private javax.swing.JButton checkAvailability;
263 private javax.swing.JLabel jLabel23;
264 private javax.swing.JPanel jPanel1;
265 private javax.swing.JPanel jPanel2;
266 private javax.swing.JPanel jPanel5;
267 private javax.swing.JComboBox<String> recipientBloodType;
268 // End of variables declaration
269 }
270
```