REFERENCES

[1] Bluetooth official web sites – bluetooth.com and bluetooth.org

[2] The Bluetooth Special Interest Group **(**SIG**)** - **www.**bluetooth**.com/**Bluetooth/SIG/

[3] Netbeans Official web site - http://www.netbeans.org

[4] NetBeans 5.5 IDE - http://www.netbeans.org/products/ide/

[5] Tutorial to connect MySql with NetBeans IDE - http://www.netbeans.org/kb/articles/mysql-client.html

[6] MySql Database and Java Connector - http://dev.mysql.com/downloads/

[7] MIDP 2.0 (JSR 118): MIDP 2.0 - java.sun.com/products/midp/

[8] NetBeans Mobility Pack - http://www.netbeans.org/products/mobility/

[9] *Using the Java APIs for Bluetooth Wireless Technology*, Parts 1 and 2 C. Enrique Ortiz, December 2004, February 2005 http://developers.sun.com/techtopics/mobility/apis/articles/bluetoothintro/ http://developers.sun.com/techtopics/mobility/apis/articles/bluetoothcore/

[10] *Introduction to Bluetooth and J2ME*, Parts 1 and 2 Jason Lam, December 2004 -http://wirelesspronews.com/wirelesspronews-14-20041213IntroductiontoBluetoothandJ2ME.html http://www.jasonlam604.com/articles_introduction_to_bluetooth_and_j2me_part2.php

[11] JSR 82 specification Java Community Page - www.jcp.org/en/jsr/detail?id=82

[12] J2ME MIDP 2.0 specification – http://jcp.org/aboutJava/communityprocess/final/jsr118/index.html

[13] *Programming Java 2 Micro Edition on Symbian OS: A Developer's Guide to MIDP 2.0* Martin de Jode, John Wiley, July 2004 http://www.symbian.com/books/j2me/j2me_info.html

[14] *Beginning J2ME: From Novice to Professional*, 3rd By Edition Sing Li and Jonathan Knudsen, Apress, May 2005 http://www.apress.com/book/bookDisplay.html?bID=426

[15] *Bluetooth for Java By* Bruce Hopkins and Ranjith Antony, Apress, March 2003 http://www.apress.com/book/bookDisplay.html?bID=139 and http://www.javabluetooth.com/

[16] J2ME Connected, Limited Device Configuration (JSR-30), Sun Microsystems, Inc. http://www.jcp.org/jsr/detail/30.jsp

[17] Interference in the 2.4 GHz ISM Band: Impact on the Bluetooth Access control Performance by Nada Golmie and Frederic Mouveaux National Institute of Standards and Technology Gaithersburg, Maryland 20899

[18] J. Bray and C. Sturman, *Bluetooth 1.1, Connect Without Cables*, Second Edition, Prentice Hall, 2002.

[19] Kumar et. al., Bluetooth Application Programming with the Java APIs, First Edition, Morgan Kaufmann, 2004.

[20] Vietcovo Community Forum - http://forum.vietcovo.com/

[21] Java Bluetooth Community Forum - http://community.java.net/projects/community/communications

[22] Open Source Web Site - http://sourceforge.net/

[23] Java Swing Tutorial - http://www.java2s.com/Tutorial/Java/CatalogJava.htm

[24] Avetana Bluetooth JSR-82 API – http://www.avetana-gmbh.de/avetana-gmbh/produkte/jsr82.eng.xm

APPENDICES

APPENDIX I

PC SIDE SOURCE CODE

```
package ui;

import java.awt.Dimension;
import java.awt.Point;
import java.awt.Rectangle;
import java.awt.event.ActionListener;
import javax.swing.JDialog;
import javax.swing.JFrame;

public class About extends JDialog {

    /** Creates new form About */
    public About(JFrame parent) {
        super(parent,true);
        initComponents();
        pack();
        Rectangle parentBounds = parent.getBounds();
        Dimension size = getSize();
        // Center in the parent
        int x = Math.max(0, parentBounds.x + (parentBounds.width -
size.width) / 2);
        int y = Math.max(0, parentBounds.y + (parentBounds.height -
size.height) / 2);
        setLocation(new Point(x, y));
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method
is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc=" Generated Code
">//GEN-BEGIN:initComponents
    private void initComponents() {
        java.awt.GridBagConstraints gridBagConstraints;

        mainPanel = new javax.swing.JPanel();
        copyrightTextArea = new javax.swing.JTextArea();
        closeButton = new javax.swing.JButton();

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


setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE
);
        setTitle("About Bluetooth Bus Info");
        mainPanel.setLayout(new java.awt.GridBagLayout());


mainPanel.setBorder(javax.swing.BorderFactory.createEmptyBorder(11,
11, 12, 12));

copyrightTextArea.setBackground(javax.swing.UIManager.getDefaults().g
etColor("Panel.background"));
        copyrightTextArea.setColumns(25);
        copyrightTextArea.setEditable(false);
        copyrightTextArea.setLineWrap(true);
        copyrightTextArea.setRows(8);
```

```
        copyrightTextArea.setText("Bluetooth Bus Info\n\nAuthor: Mohd
Suhkri Yasri\nLanguage: Java\nPlatform: Windows\nOrganization:
Unimap");
        copyrightTextArea.setWrapStyleWord(true);
        copyrightTextArea.setBorder(null);
        copyrightTextArea.setFocusable(false);
        gridBagConstraints = new java.awt.GridBagConstraints();
        gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;
        gridBagConstraints.anchor =
java.awt.GridBagConstraints.NORTH;
        gridBagConstraints.weightx = 1.0;
        gridBagConstraints.weighty = 1.0;
        gridBagConstraints.insets = new java.awt.Insets(24, 0, 24,
0);
        mainPanel.add(copyrightTextArea, gridBagConstraints);

        closeButton.setMnemonic('C');
        closeButton.setText("Close");
        closeButton.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                closeButtonActionPerformed(evt);
            }
        });

        gridBagConstraints = new java.awt.GridBagConstraints();
        gridBagConstraints.gridx = 0;
        gridBagConstraints.gridy = 1;
        gridBagConstraints.anchor =
java.awt.GridBagConstraints.SOUTHEAST;
        mainPanel.add(closeButton, gridBagConstraints);

        gridBagConstraints = new java.awt.GridBagConstraints();
        gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;
        gridBagConstraints.weightx = 1.0;
        gridBagConstraints.weighty = 1.0;
        getContentPane().add(mainPanel, gridBagConstraints);

    }// </editor-fold>//GEN-END:initComponents

    private void
closeButtonActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_closeButtonActionPerformed
        setVisible(false);
        dispose();
    }//GEN-LAST:event_closeButtonActionPerformed


    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton closeButton;
    private javax.swing.JTextArea copyrightTextArea;
    private javax.swing.JPanel mainPanel;
    // End of variables declaration//GEN-END:variables

}
```

```
package ui;

import javax.swing.table.AbstractTableModel;
import java.sql.*;
import java.util.ArrayList;

public class ATableModel extends AbstractTableModel {

    private int colnum=6;
    private int rownum;
    private String[] colNames={
        "BluetoothID","Company","Regno","Origin","Time
Arrive","Platform"
    };
    private  ArrayList<String[]> ResultSets;

    public ATableModel(ResultSet aa) {

      ResultSets=new ArrayList<String[]>();

      try{
        while(aa.next()){

              String[] row={

aa.getString("abluetoothID"),aa.getString("acompany"),
aa.getString("aregno"),aa.getString("origin"),aa.getString("timearriv
e"),aa.getString("platform")

            };
            ResultSets.add(row);

         }
      }
      catch(Exception e){
          System.out.println("Exception in CarTableModel");
            }

    }

    public Object getValueAt(int rowindex, int columnindex) {

       String[] row=ResultSets.get(rowindex);
       return row[columnindex];

    }

    public int getRowCount() {
        return ResultSets.size();
    }

    public int getColumnCount() {
        return colnum;
    }

    public String getColumnName(int param) {

       return colNames[param];
    }

}
```

```
package ui;

import java.sql.*;

public class MyDBConnection {

    private Connection myConnection;
    private MyDBConnection mdbc;
    private java.sql.Statement stmt;

    /** Creates a new instance of MyDBConnection */
    public MyDBConnection() {

    }

    public void init(){

       try{

        Class.forName("com.mysql.jdbc.Driver");
        myConnection=DriverManager.getConnection(
                "jdbc:mysql://localhost/bluetooth","root", "qwerty"
                );
        }
        catch(Exception e){
            System.out.println("Failed to get connection");
            e.printStackTrace();
        }
    }


    public Connection getMyConnection(){
        return myConnection;
    }


    public void close(ResultSet rs){

        if(rs !=null){
            try{
               rs.close();
            }
            catch(Exception e){}

        }
    }

     public void close(java.sql.Statement stmt){

        if(stmt !=null){
            try{
               stmt.close();
            }
            catch(Exception e){}

        }
    }

  public void destroy(){

    if(myConnection !=null){
```

```
            try{
                  myConnection.close();
               }
               catch(Exception e){}


      }
    }

}
```

```
package ui;

import javax.swing.table.AbstractTableModel;
import java.sql.*;
import java.util.ArrayList;

public class scanTable extends AbstractTableModel {

    private int colnum=6;
    private int rownum;
    private String[] colNames={
        "BluetoothID","Company","Regno","Destination","Time
Depart","Platform"
    };
    private  ArrayList<String[]> ResultSets;

    public scanTable(ResultSet rs) {

      ResultSets=new ArrayList<String[]>();

      try{
        while(rs.next()){

              String[] row={
                rs.getString("dbluetoothID"),
rs.getString("company"),
rs.getString("regno"),rs.getString("destination"),rs.getString("timed
epart"),rs.getString("platform")

            };
            ResultSets.add(row);

         }
      }
      catch(Exception e){
          System.out.println("Exception in CarTableModel");
            }

    }

    public Object getValueAt(int rowindex, int columnindex) {

       String[] row=ResultSets.get(rowindex);
       return row[columnindex];

    }

    public int getRowCount() {
        return ResultSets.size();
    }

    public int getColumnCount() {
        return colnum;
    }

    public String getColumnName(int param) {

       return colNames[param];
    }

}
```

```
package ui;

import java.awt.event.KeyEvent;
import java.util.Calendar;
import javax.swing.JTextArea;
import java.awt.BorderLayout;
import java.awt.Dimension;
import java.awt.event.KeyAdapter;
import java.awt.event.KeyEvent;
import java.util.Calendar;

import javax.swing.BorderFactory;
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;
import javax.swing.JTextField;
import javax.swing.SwingUtilities;
import java.sql.*;
import ui.SwingUI;

public class UI extends javax.swing.JFrame implements
ConnectionProtocol.EventHandler {

    public UI() {
        mdbc=new MyDBConnection();
        mdbc.init();
        Connection conn=mdbc.getMyConnection();
        try {
            stmt= conn.createStatement();
        } catch (SQLException ex) {
            ex.printStackTrace();
        }
        initComponents();
    }

    // <editor-fold defaultstate="collapsed" desc=" Generated Code
">//GEN-BEGIN:initComponents
    private void initComponents() {
        jTabbedPane1 = new javax.swing.JTabbedPane();
        jPanel1 = new javax.swing.JPanel();
        jPanel12 = new javax.swing.JPanel();
        jScrollPane1 = new javax.swing.JScrollPane();
        jTextArea1 = new javax.swing.JTextArea();
        jScrollPane2 = new javax.swing.JScrollPane();
        jTextArea2 = new javax.swing.JTextArea();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jPanel14 = new javax.swing.JPanel();
        jScrollPane5 = new javax.swing.JScrollPane();
        jTextArea3 = new javax.swing.JTextArea();
        jScrollPane6 = new javax.swing.JScrollPane();
        jTextArea4 = new javax.swing.JTextArea();
        jTextField1 = new javax.swing.JTextField();
        jTextField2 = new javax.swing.JTextField();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jPanel2 = new javax.swing.JPanel();
        jTabbedPane2 = new javax.swing.JTabbedPane();
        jPanel4 = new javax.swing.JPanel();
        jButton4 = new javax.swing.JButton();
```

```
jButton5 = new javax.swing.JButton();
jPanel5 = new javax.swing.JPanel();
jLabel3 = new javax.swing.JLabel();
jTextField4 = new javax.swing.JTextField();
jLabel4 = new javax.swing.JLabel();
jLabel5 = new javax.swing.JLabel();
jTextField5 = new javax.swing.JTextField();
jTextField6 = new javax.swing.JTextField();
jLabel6 = new javax.swing.JLabel();
jLabel7 = new javax.swing.JLabel();
jTextField7 = new javax.swing.JTextField();
jTextField8 = new javax.swing.JTextField();
jLabel8 = new javax.swing.JLabel();
jTextField29 = new javax.swing.JTextField();
jPanel3 = new javax.swing.JPanel();
jPanel6 = new javax.swing.JPanel();
jLabel11 = new javax.swing.JLabel();
jLabel10 = new javax.swing.JLabel();
jTextField9 = new javax.swing.JTextField();
jTextField10 = new javax.swing.JTextField();
jLabel9 = new javax.swing.JLabel();
jTextField25 = new javax.swing.JTextField();
jLabel29 = new javax.swing.JLabel();
jTextField26 = new javax.swing.JTextField();
jLabel30 = new javax.swing.JLabel();
jTextField27 = new javax.swing.JTextField();
jLabel31 = new javax.swing.JLabel();
jTextField28 = new javax.swing.JTextField();
jButton3 = new javax.swing.JButton();
jButton6 = new javax.swing.JButton();
jPanel7 = new javax.swing.JPanel();
jLabel12 = new javax.swing.JLabel();
jLabel13 = new javax.swing.JLabel();
jLabel14 = new javax.swing.JLabel();
jTextField11 = new javax.swing.JTextField();
jTextField12 = new javax.swing.JTextField();
jButton7 = new javax.swing.JButton();
jButton8 = new javax.swing.JButton();
jScrollPane3 = new javax.swing.JScrollPane();
jTable1 = new javax.swing.JTable();
jPanel8 = new javax.swing.JPanel();
jTabbedPane3 = new javax.swing.JTabbedPane();
jPanel9 = new javax.swing.JPanel();
jLabel16 = new javax.swing.JLabel();
jLabel17 = new javax.swing.JLabel();
jLabel18 = new javax.swing.JLabel();
jLabel19 = new javax.swing.JLabel();
jLabel20 = new javax.swing.JLabel();
jLabel21 = new javax.swing.JLabel();
jTextField14 = new javax.swing.JTextField();
jTextField15 = new javax.swing.JTextField();
jTextField16 = new javax.swing.JTextField();
jTextField17 = new javax.swing.JTextField();
jTextField18 = new javax.swing.JTextField();
jTextField19 = new javax.swing.JTextField();
jButton11 = new javax.swing.JButton();
jButton12 = new javax.swing.JButton();
jPanel10 = new javax.swing.JPanel();
jPanel13 = new javax.swing.JPanel();
jLabel23 = new javax.swing.JLabel();
jLabel24 = new javax.swing.JLabel();
```

```
        jTextField20 = new javax.swing.JTextField();
        jTextField21 = new javax.swing.JTextField();
        jLabel22 = new javax.swing.JLabel();
        jLabel32 = new javax.swing.JLabel();
        jLabel33 = new javax.swing.JLabel();
        jLabel34 = new javax.swing.JLabel();
        jTextField3 = new javax.swing.JTextField();
        jTextField30 = new javax.swing.JTextField();
        jTextField31 = new javax.swing.JTextField();
        jTextField32 = new javax.swing.JTextField();
        jButton13 = new javax.swing.JButton();
        jButton14 = new javax.swing.JButton();
        jPanel11 = new javax.swing.JPanel();
        jLabel25 = new javax.swing.JLabel();
        jLabel26 = new javax.swing.JLabel();
        jLabel27 = new javax.swing.JLabel();
        jTextField22 = new javax.swing.JTextField();
        jTextField23 = new javax.swing.JTextField();
        jButton15 = new javax.swing.JButton();
        jButton16 = new javax.swing.JButton();
        jScrollPane4 = new javax.swing.JScrollPane();
        jTable2 = new javax.swing.JTable();
        jPanel15 = new javax.swing.JPanel();
        jPanel16 = new javax.swing.JPanel();
        jScrollPane7 = new javax.swing.JScrollPane();
        jTextArea5 = new javax.swing.JTextArea();
        jTextField13 = new javax.swing.JTextField();
        jTextField33 = new javax.swing.JTextField();
        jLabel15 = new javax.swing.JLabel();
        jLabel28 = new javax.swing.JLabel();
        jButton9 = new javax.swing.JButton();
        jPanel17 = new javax.swing.JPanel();
        jScrollPane8 = new javax.swing.JScrollPane();
        jTextArea6 = new javax.swing.JTextArea();
        jTextField24 = new javax.swing.JTextField();
        jLabel36 = new javax.swing.JLabel();
        jTextField34 = new javax.swing.JTextField();
        jLabel35 = new javax.swing.JLabel();
        jButton10 = new javax.swing.JButton();
        jMenuBar1 = new javax.swing.JMenuBar();
        jMenu1 = new javax.swing.JMenu();
        jMenuItem1 = new javax.swing.JMenuItem();
        jMenu2 = new javax.swing.JMenu();
        jMenuItem2 = new javax.swing.JMenuItem();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        addWindowListener(new java.awt.event.WindowAdapter() {
            public void windowClosing(java.awt.event.WindowEvent evt)
{
                formWindowClosing(evt);
            }
        });


jPanel1.setBorder(javax.swing.BorderFactory.createTitledBorder(""));

jPanel12.setBorder(javax.swing.BorderFactory.createTitledBorder("Blue
tooth Devices"));
        jTextArea1.setColumns(20);
        jTextArea1.setRows(5);
```

```
        jScrollPane1.setViewportView(jTextArea1);

        jTextArea2.setColumns(20);
        jTextArea2.setRows(5);
        jScrollPane2.setViewportView(jTextArea2);


jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
        jLabel1.setText("Bluetooth Address");


jLabel2.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
        jLabel2.setText("Time and Date Detected");

        javax.swing.GroupLayout jPanel12Layout = new
javax.swing.GroupLayout(jPanel12);
        jPanel12.setLayout(jPanel12Layout);
        jPanel12Layout.setHorizontalGroup(

jPanel12Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
            .addGroup(jPanel12Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel12Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING)
                    .addComponent(jScrollPane1,
javax.swing.GroupLayout.DEFAULT_SIZE, 250, Short.MAX_VALUE)
                    .addComponent(jLabel1,
javax.swing.GroupLayout.DEFAULT_SIZE, 250, Short.MAX_VALUE))
                .addGap(20, 20, 20)

.addGroup(jPanel12Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.TRAILING)
                    .addComponent(jScrollPane2,
javax.swing.GroupLayout.DEFAULT_SIZE, 255, Short.MAX_VALUE)
                    .addComponent(jLabel2,
javax.swing.GroupLayout.DEFAULT_SIZE, 255, Short.MAX_VALUE))
                .addContainerGap())
        );
        jPanel12Layout.setVerticalGroup(

jPanel12Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel12Layout.createSequentialGroup()

.addGroup(jPanel12Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jLabel1)
                    .addComponent(jLabel2))

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

.addGroup(jPanel12Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING)
                    .addComponent(jScrollPane2,
javax.swing.GroupLayout.DEFAULT_SIZE, 173, Short.MAX_VALUE)
                    .addComponent(jScrollPane1,
javax.swing.GroupLayout.DEFAULT_SIZE, 173, Short.MAX_VALUE))
                .addContainerGap())
```

```
        );


jPanel14.setBorder(javax.swing.BorderFactory.createTitledBorder("Buss
es Information"));
        jTextArea3.setColumns(20);
        jTextArea3.setRows(5);
        jScrollPane5.setViewportView(jTextArea3);

        jTextArea4.setColumns(20);
        jTextArea4.setRows(5);
        jScrollPane6.setViewportView(jTextArea4);

        jButton1.setText("Departure Information");
        jButton1.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jButton1ActionPerformed(evt);
            }
        });

        jButton2.setText("Arriving Information");
        jButton2.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jButton2ActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout jPanel14Layout = new
javax.swing.GroupLayout(jPanel14);
        jPanel14.setLayout(jPanel14Layout);
        jPanel14Layout.setHorizontalGroup(

jPanel14Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
            .addGroup(jPanel14Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel14Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING)
                    .addComponent(jTextField1,
javax.swing.GroupLayout.DEFAULT_SIZE, 256, Short.MAX_VALUE)
                    .addGroup(jPanel14Layout.createSequentialGroup()
                        .addComponent(jScrollPane6,
javax.swing.GroupLayout.DEFAULT_SIZE, 256, Short.MAX_VALUE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED))
                    .addComponent(jButton2,
javax.swing.GroupLayout.DEFAULT_SIZE, 256, Short.MAX_VALUE))
                .addGap(14, 14, 14)

.addGroup(jPanel14Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING, false)
                    .addComponent(jButton1,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
```

```
                    .addComponent(jTextField2,
javax.swing.GroupLayout.Alignment.TRAILING)
                    .addComponent(jScrollPane5,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, 255, Short.MAX_VALUE))
                .addContainerGap())
        );
        jPanel14Layout.setVerticalGroup(

jPanel14Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
            .addGroup(jPanel14Layout.createSequentialGroup()

.addGroup(jPanel14Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING)
                    .addComponent(jScrollPane6,
javax.swing.GroupLayout.DEFAULT_SIZE, 217, Short.MAX_VALUE)
                    .addComponent(jScrollPane5,
javax.swing.GroupLayout.DEFAULT_SIZE, 217, Short.MAX_VALUE))

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

.addGroup(jPanel14Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jTextField1,
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))

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

.addGroup(jPanel14Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jButton2)
                    .addComponent(jButton1)))
        );

        javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
        jPanel1.setLayout(jPanel1Layout);
        jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.TRAILING)
                    .addComponent(jPanel12,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                    .addComponent(jPanel14,
javax.swing.GroupLayout.Alignment.LEADING,
```

```
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
                .addContainerGap())
        );
        jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup()
                .addComponent(jPanel12,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jPanel14,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        );
        jTabbedPane1.addTab("Device Scanner", jPanel1);


jTabbedPane2.setBorder(javax.swing.BorderFactory.createTitledBorder("
Departure Database"));
        jTabbedPane2.setName("Add");

jPanel4.setBorder(javax.swing.BorderFactory.createTitledBorder("Add
Departure Data"));
        jButton4.setText("Add");
        jButton4.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jButton4ActionPerformed(evt);
            }
        });

        jButton5.setText("Clear Field");
        jButton5.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jButton5ActionPerformed(evt);
            }
        });


jPanel5.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 0, 0)));
        jLabel3.setText("Bluetooth ID :");

        jLabel4.setText("Company :");

        jLabel5.setText("Registration Number :");

        jLabel6.setText("Destination :");

        jLabel7.setText("Time Depart :");

        jLabel8.setText("Platform :");
```

```
        javax.swing.GroupLayout jPanel5Layout = new
javax.swing.GroupLayout(jPanel5);
        jPanel5.setLayout(jPanel5Layout);
        jPanel5Layout.setHorizontalGroup(

jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(jPanel5Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
                    .addComponent(jLabel5)
                    .addComponent(jLabel4)
                    .addComponent(jLabel6)
                    .addComponent(jLabel7)
                    .addComponent(jLabel8)
                    .addComponent(jLabel3))

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

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
                    .addComponent(jTextField8,
javax.swing.GroupLayout.DEFAULT_SIZE, 387, Short.MAX_VALUE)
                    .addComponent(jTextField7,
javax.swing.GroupLayout.DEFAULT_SIZE, 387, Short.MAX_VALUE)
                    .addComponent(jTextField6,
javax.swing.GroupLayout.DEFAULT_SIZE, 387, Short.MAX_VALUE)
                    .addComponent(jTextField5,
javax.swing.GroupLayout.DEFAULT_SIZE, 387, Short.MAX_VALUE)
                    .addComponent(jTextField4,
javax.swing.GroupLayout.DEFAULT_SIZE, 387, Short.MAX_VALUE)
                    .addComponent(jTextField29,
javax.swing.GroupLayout.DEFAULT_SIZE, 387, Short.MAX_VALUE))
                .addContainerGap())
        );
        jPanel5Layout.setVerticalGroup(

jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(jPanel5Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jLabel3)
                    .addComponent(jTextField29,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jLabel4)
                    .addComponent(jTextField4,
javax.swing.GroupLayout.PREFERRED_SIZE,
```

```
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jLabel5)
                    .addComponent(jTextField5,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jLabel6)
                    .addComponent(jTextField6,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
                    .addGroup(jPanel5Layout.createSequentialGroup()
                        .addGap(26, 26, 26)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                            .addComponent(jLabel8)
                            .addComponent(jTextField8,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                        .addComponent(jLabel7)
                        .addComponent(jTextField7,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
                .addContainerGap())
        );

        javax.swing.GroupLayout jPanel4Layout = new
javax.swing.GroupLayout(jPanel4);
        jPanel4.setLayout(jPanel4Layout);
        jPanel4Layout.setHorizontalGroup(

jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(jPanel4Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
```

```
.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel4Layout.createSequentialGroup()
                        .addComponent(jButton5)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                        .addComponent(jButton4))
                    .addComponent(jPanel5,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
                .addContainerGap())
        );
        jPanel4Layout.setVerticalGroup(

jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(jPanel4Layout.createSequentialGroup()
                .addComponent(jPanel5,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

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

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jButton4)
                    .addComponent(jButton5)))
        );
        jTabbedPane2.addTab("Add", jPanel4);


jPanel3.setBorder(javax.swing.BorderFactory.createTitledBorder("Edit
Departure Data"));

jPanel6.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 0, 0)));
        jLabel11.setText("Edit Company :");

        jLabel10.setText("Enter Bluetooth ID :");

        jLabel9.setText("Edit Registration No:");

        jLabel29.setText("Edit Destination :");

        jLabel30.setText("Edit Time Depart :");

        jLabel31.setText("Edit Platform :");

        javax.swing.GroupLayout jPanel6Layout = new
javax.swing.GroupLayout(jPanel6);
        jPanel6.setLayout(jPanel6Layout);
        jPanel6Layout.setHorizontalGroup(

jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(jPanel6Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
```

```
                    .addGroup(jPanel6Layout.createSequentialGroup()

.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
                            .addComponent(jLabel11)
                            .addComponent(jLabel10)
                            .addComponent(jLabel9)
                            .addComponent(jLabel29)
                            .addComponent(jLabel30))

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

.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
                            .addComponent(jTextField9,
javax.swing.GroupLayout.DEFAULT_SIZE, 393, Short.MAX_VALUE)
                            .addComponent(jTextField10,
javax.swing.GroupLayout.DEFAULT_SIZE, 393, Short.MAX_VALUE)
                            .addComponent(jTextField25,
javax.swing.GroupLayout.DEFAULT_SIZE, 393, Short.MAX_VALUE)
                            .addComponent(jTextField26,
javax.swing.GroupLayout.DEFAULT_SIZE, 393, Short.MAX_VALUE)
                            .addComponent(jTextField27,
javax.swing.GroupLayout.DEFAULT_SIZE, 393, Short.MAX_VALUE)))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel6Layout.createSequentialGroup()
                        .addComponent(jLabel31)
                        .addGap(35, 35, 35)
                        .addComponent(jTextField28,
javax.swing.GroupLayout.DEFAULT_SIZE, 393, Short.MAX_VALUE)))
                .addContainerGap())
        );
        jPanel6Layout.setVerticalGroup(

jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(jPanel6Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jLabel10)
                    .addComponent(jTextField9,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jLabel11)
                    .addComponent(jTextField10,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

```
.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jLabel9)
                    .addComponent(jTextField25,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jLabel29)
                    .addComponent(jTextField26,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jLabel30)
                    .addComponent(jTextField27,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jTextField28,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel31))
                .addContainerGap())
        );

        jButton3.setText("Update");
        jButton3.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jButton3ActionPerformed(evt);
            }
        });

        jButton6.setText("Clear Field");
        jButton6.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jButton6ActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout jPanel3Layout = new
javax.swing.GroupLayout(jPanel3);
```

```
        jPanel3.setLayout(jPanel3Layout);
        jPanel3Layout.setHorizontalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel3Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.TRAILING)
                    .addComponent(jPanel6,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                    .addGroup(jPanel3Layout.createSequentialGroup()
                        .addComponent(jButton6)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                        .addComponent(jButton3)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED))
)
                .addContainerGap())
        );
        jPanel3Layout.setVerticalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel3Layout.createSequentialGroup()
                .addComponent(jPanel6,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
12, Short.MAX_VALUE)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jButton6)
                    .addComponent(jButton3))
                .addContainerGap())
        );
        jTabbedPane2.addTab("Edit", jPanel3);


jPanel7.setBorder(javax.swing.BorderFactory.createTitledBorder("Delet
e Departure Data By:"));
        jLabel12.setText("Bluetooth ID :");


jLabel13.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
        jLabel13.setText("OR");

        jLabel14.setText("Registration Number :");

        jButton7.setText("Delete");
        jButton7.addActionListener(new
java.awt.event.ActionListener() {
```

```
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jButton7ActionPerformed(evt);
            }
        });

        jButton8.setText("Clear Field");
        jButton8.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jButton8ActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout jPanel7Layout = new
javax.swing.GroupLayout(jPanel7);
        jPanel7.setLayout(jPanel7Layout);
        jPanel7Layout.setHorizontalGroup(

jPanel7Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(jPanel7Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel7Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
                    .addGroup(jPanel7Layout.createSequentialGroup()

.addGroup(jPanel7Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)

.addGroup(jPanel7Layout.createSequentialGroup()

.addGroup(jPanel7Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
                                    .addComponent(jLabel12)
                                    .addComponent(jLabel14))

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

.addGroup(jPanel7Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
                                    .addComponent(jTextField11,
javax.swing.GroupLayout.DEFAULT_SIZE, 409, Short.MAX_VALUE)
                                    .addComponent(jTextField12,
javax.swing.GroupLayout.DEFAULT_SIZE, 409, Short.MAX_VALUE)))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel7Layout.createSequentialGroup()
                                .addComponent(jButton8)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                                .addComponent(jButton7)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED))
)
                        .addGap(10, 10, 10))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel7Layout.createSequentialGroup()
```

```
                        .addComponent(jLabel13,
javax.swing.GroupLayout.DEFAULT_SIZE, 518, Short.MAX_VALUE)
                        .addContainerGap())))
        );
        jPanel7Layout.setVerticalGroup(

jPanel7Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel7Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel7Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jLabel12)
                    .addComponent(jTextField11,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jLabel13)

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

.addGroup(jPanel7Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jLabel14)
                    .addComponent(jTextField12,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
120, Short.MAX_VALUE)

.addGroup(jPanel7Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jButton8)
                    .addComponent(jButton7)))
        );
        jTabbedPane2.addTab("Delete", jPanel7);

        ResultSet rs=getResultFromdepart1();
        jTable1.setModel(new scanTable(rs));
        mdbc.close(rs);
        jScrollPane3.setViewportView(jTable1);

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

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel2Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.TRAILING)
```

```
                    .addComponent(jScrollPane3,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, 567, Short.MAX_VALUE)
                    .addComponent(jTabbedPane2,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, 567, Short.MAX_VALUE))
                .addGap(12, 12, 12))
        );
        jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(jPanel2Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jTabbedPane2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jScrollPane3,
javax.swing.GroupLayout.DEFAULT_SIZE, 219, Short.MAX_VALUE)
                .addContainerGap())
        );
        jTabbedPane1.addTab("Departure Database", jPanel2);


jTabbedPane3.setBorder(javax.swing.BorderFactory.createTitledBorder("
Arriving Database"));

jPanel9.setBorder(javax.swing.BorderFactory.createTitledBorder("Add
Arrive Data"));
        jLabel16.setText("Bluetooth ID :");

        jLabel17.setText("Company :");

        jLabel18.setText("Registration Number :");

        jLabel19.setText("Origin :");

        jLabel20.setText("Time Arrive:");

        jLabel21.setText("Platform :");

        jButton11.setText("Add");
        jButton11.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jButton11ActionPerformed(evt);
            }
        });

        jButton12.setText("Clear Field");
        jButton12.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jButton12ActionPerformed(evt);
            }
        });
```

```
        javax.swing.GroupLayout jPanel9Layout = new
javax.swing.GroupLayout(jPanel9);
        jPanel9.setLayout(jPanel9Layout);
        jPanel9Layout.setHorizontalGroup(

jPanel9Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(jPanel9Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel9Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
                    .addGroup(jPanel9Layout.createSequentialGroup()

.addGroup(jPanel9Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
                            .addComponent(jLabel18)
                            .addComponent(jLabel19)
                            .addComponent(jLabel20)
                            .addComponent(jLabel21)
                            .addComponent(jLabel17)
                            .addComponent(jLabel16))

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

.addGroup(jPanel9Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
                            .addComponent(jTextField15,
javax.swing.GroupLayout.DEFAULT_SIZE, 411, Short.MAX_VALUE)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel9Layout.createSequentialGroup()

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                                .addComponent(jTextField19,
javax.swing.GroupLayout.DEFAULT_SIZE, 411, Short.MAX_VALUE))
                            .addComponent(jTextField18,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, 411, Short.MAX_VALUE)
                            .addComponent(jTextField17,
javax.swing.GroupLayout.DEFAULT_SIZE, 411, Short.MAX_VALUE)
                            .addComponent(jTextField16,
javax.swing.GroupLayout.DEFAULT_SIZE, 411, Short.MAX_VALUE)
                            .addComponent(jTextField14,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, 411, Short.MAX_VALUE)))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel9Layout.createSequentialGroup()
                        .addComponent(jButton12)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                        .addComponent(jButton11)))
                .addContainerGap())
        );
        jPanel9Layout.setVerticalGroup(

jPanel9Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(jPanel9Layout.createSequentialGroup()
```

```
                .addContainerGap()

.addGroup(jPanel9Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jLabel16)
                    .addComponent(jTextField14,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel9Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jLabel17)
                    .addComponent(jTextField15,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel9Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jLabel18)
                    .addComponent(jTextField16,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel9Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jLabel19)
                    .addComponent(jTextField17,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel9Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jLabel20)
                    .addComponent(jTextField18,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel9Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
                    .addComponent(jLabel21)
                    .addComponent(jTextField19,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
17, Short.MAX_VALUE)

.addGroup(jPanel9Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                    .addComponent(jButton11)
                    .addComponent(jButton12)))
        );
        jTabbedPane3.addTab("Add", jPanel9);


jPanel10.setBorder(javax.swing.BorderFactory.createTitledBorder("Edit
Depart Data"));
        jPanel10.setName("Edit");

jPanel13.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 0, 0)));
        jLabel23.setText("Enter Bluetooth ID :");

        jLabel24.setText("Edit Company :");

        jLabel22.setText("Edit Registration No :");

        jLabel32.setText("Edit Origin :");

        jLabel33.setText("Edit Time Arrive :");

        jLabel34.setText("Edit Platform :");

        javax.swing.GroupLayout jPanel13Layout = new
javax.swing.GroupLayout(jPanel13);
        jPanel13.setLayout(jPanel13Layout);
        jPanel13Layout.setHorizontalGroup(

jPanel13Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
            .addGroup(jPanel13Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel13Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING)

.addGroup(jPanel13Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel13Layout.createSequentialGroup()
                            .addComponent(jLabel23)

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

.addGroup(jPanel13Layout.createSequentialGroup()
                            .addComponent(jLabel24)
                            .addGap(9, 9, 9)))
                    .addComponent(jLabel22)
                    .addComponent(jLabel32)
                    .addComponent(jLabel33)
                    .addComponent(jLabel34))

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

```
.addGroup(jPanel13Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING)
                    .addComponent(jTextField20,
javax.swing.GroupLayout.DEFAULT_SIZE, 392, Short.MAX_VALUE)
                    .addComponent(jTextField21,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, 392, Short.MAX_VALUE)
                    .addComponent(jTextField3,
javax.swing.GroupLayout.DEFAULT_SIZE, 392, Short.MAX_VALUE)
                    .addComponent(jTextField30,
javax.swing.GroupLayout.DEFAULT_SIZE, 392, Short.MAX_VALUE)
                    .addComponent(jTextField31,
javax.swing.GroupLayout.DEFAULT_SIZE, 392, Short.MAX_VALUE)
                    .addComponent(jTextField32,
javax.swing.GroupLayout.DEFAULT_SIZE, 392, Short.MAX_VALUE))
                .addContainerGap())
        );
        jPanel13Layout.setVerticalGroup(

jPanel13Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
            .addGroup(jPanel13Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel13Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jLabel23)
                    .addComponent(jTextField21,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel13Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jLabel24)
                    .addComponent(jTextField20,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel13Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jLabel22)
                    .addComponent(jTextField3,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel13Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jLabel32)
                    .addComponent(jTextField30,
javax.swing.GroupLayout.PREFERRED_SIZE,
```

```
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel13Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jLabel33)
                    .addComponent(jTextField31,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel13Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jLabel34)
                    .addComponent(jTextField32,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        );

        jButton13.setText("Update");
        jButton13.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jButton13ActionPerformed(evt);
            }
        });

        jButton14.setText("Clear Field");
        jButton14.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jButton14ActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout jPanel10Layout = new
javax.swing.GroupLayout(jPanel10);
        jPanel10.setLayout(jPanel10Layout);
        jPanel10Layout.setHorizontalGroup(

jPanel10Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel10Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel10Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.TRAILING)
                    .addComponent(jPanel13,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
```

```
                    .addGroup(jPanel10Layout.createSequentialGroup()
                        .addComponent(jButton14)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                        .addComponent(jButton13)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED))
)
                .addContainerGap())
        );
        jPanel10Layout.setVerticalGroup(

jPanel10Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel10Layout.createSequentialGroup()
                .addComponent(jPanel13,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

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

.addGroup(jPanel10Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jButton14)
                    .addComponent(jButton13))
                .addContainerGap())
        );
        jTabbedPane3.addTab("Edit", jPanel10);


jPanel11.setBorder(javax.swing.BorderFactory.createTitledBorder("Dele
te Arrive Data By "));
        jLabel25.setText("Bluetooth ID :");


jLabel26.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
        jLabel26.setText("OR");

        jLabel27.setText("Registration Number :");

        jButton15.setText("Delete");
        jButton15.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jButton15ActionPerformed(evt);
            }
        });

        jButton16.setText("Clear Field");
        jButton16.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jButton16ActionPerformed(evt);
            }
        });
```

```
        javax.swing.GroupLayout jPanel11Layout = new
javax.swing.GroupLayout(jPanel11);
        jPanel11.setLayout(jPanel11Layout);
        jPanel11Layout.setHorizontalGroup(

jPanel11Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
            .addGroup(jPanel11Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel11Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING)
                    .addComponent(jLabel26,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, 520, Short.MAX_VALUE)
                    .addGroup(jPanel11Layout.createSequentialGroup()
                        .addComponent(jLabel25)
                        .addGap(43, 43, 43)
                        .addComponent(jTextField22,
javax.swing.GroupLayout.DEFAULT_SIZE, 410, Short.MAX_VALUE))
                    .addGroup(jPanel11Layout.createSequentialGroup()
                        .addComponent(jLabel27)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                        .addComponent(jTextField23,
javax.swing.GroupLayout.DEFAULT_SIZE, 411, Short.MAX_VALUE))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel11Layout.createSequentialGroup()
                        .addComponent(jButton16)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                        .addComponent(jButton15)))
                .addContainerGap())
        );
        jPanel11Layout.setVerticalGroup(

jPanel11Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
            .addGroup(jPanel11Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel11Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jLabel25)
                    .addComponent(jTextField22,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jLabel26)

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

.addGroup(jPanel11Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jLabel27)
                    .addComponent(jTextField23,
javax.swing.GroupLayout.PREFERRED_SIZE,
```

```
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
101, Short.MAX_VALUE)

.addGroup(jPanel11Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jButton15)
                    .addComponent(jButton16)))
        );
        jTabbedPane3.addTab("Delete", jPanel11);

        ResultSet aa=getresultfromarrive();
        jTable2.setModel(new ATableModel(aa));
        mdbc.close(aa);
        jScrollPane4.setViewportView(jTable2);

        javax.swing.GroupLayout jPanel8Layout = new
javax.swing.GroupLayout(jPanel8);
        jPanel8.setLayout(jPanel8Layout);
        jPanel8Layout.setHorizontalGroup(

jPanel8Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel8Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel8Layout.createParallelGroup(javax.swing.GroupLayout.A
lignment.TRAILING)
                    .addComponent(jScrollPane4,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, 569, Short.MAX_VALUE)
                    .addComponent(jTabbedPane3,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, 569, Short.MAX_VALUE))
                .addContainerGap())
        );
        jPanel8Layout.setVerticalGroup(

jPanel8Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(jPanel8Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jTabbedPane3,
javax.swing.GroupLayout.PREFERRED_SIZE, 283,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jScrollPane4,
javax.swing.GroupLayout.DEFAULT_SIZE, 238, Short.MAX_VALUE)
                .addContainerGap())
        );
        jTabbedPane1.addTab("Arriving Database", jPanel8);


jPanel16.setBorder(javax.swing.BorderFactory.createTitledBorder("Depa
rture Information"));
        jTextArea5.setColumns(20);
        jTextArea5.setRows(5);
```

```
        jScrollPane7.setViewportView(jTextArea5);

        jTextField13.addKeyListener(new java.awt.event.KeyAdapter() {
            public void keyPressed(java.awt.event.KeyEvent evt) {
                jTextField13KeyPressed(evt);
            }
        });

        jLabel15.setText("Company :");

        jLabel28.setText("Destination :");

        jButton9.setText("Search");
        jButton9.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jButton9ActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout jPanel16Layout = new
javax.swing.GroupLayout(jPanel16);
        jPanel16.setLayout(jPanel16Layout);
        jPanel16Layout.setHorizontalGroup(

jPanel16Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel16Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel16Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.TRAILING)
                    .addComponent(jScrollPane7,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, 537, Short.MAX_VALUE)

.addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanel16Layout.createSequentialGroup()

.addGroup(jPanel16Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING, false)
                            .addComponent(jLabel28,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                            .addComponent(jLabel15,
javax.swing.GroupLayout.DEFAULT_SIZE, 82, Short.MAX_VALUE))

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

.addGroup(jPanel16Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING)
                            .addComponent(jTextField13,
javax.swing.GroupLayout.DEFAULT_SIZE, 361, Short.MAX_VALUE)
                            .addComponent(jTextField33,
javax.swing.GroupLayout.DEFAULT_SIZE, 361, Short.MAX_VALUE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                        .addComponent(jButton9,
javax.swing.GroupLayout.DEFAULT_SIZE, 84, Short.MAX_VALUE)))
```

```
                .addContainerGap())
        );
        jPanel16Layout.setVerticalGroup(

jPanel16Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel16Layout.createSequentialGroup()
                .addComponent(jScrollPane7,
javax.swing.GroupLayout.DEFAULT_SIZE, 156, Short.MAX_VALUE)

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

.addGroup(jPanel16Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jLabel15)
                    .addComponent(jTextField33,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel16Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jLabel28)
                    .addComponent(jTextField13,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jButton9))
                .addContainerGap())
        );


jPanel17.setBorder(javax.swing.BorderFactory.createTitledBorder("Arri
ving Information"));
        jTextArea6.setColumns(20);
        jTextArea6.setRows(5);
        jScrollPane8.setViewportView(jTextArea6);

        jLabel36.setText("Origin :");

        jLabel35.setText("Company :");

        jButton10.setText("Search");
        jButton10.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jButton10ActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout jPanel17Layout = new
javax.swing.GroupLayout(jPanel17);
        jPanel17.setLayout(jPanel17Layout);
        jPanel17Layout.setHorizontalGroup(

jPanel17Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
```

```
            .addGroup(jPanel17Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel17Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING)
                    .addComponent(jScrollPane8,
javax.swing.GroupLayout.DEFAULT_SIZE, 537, Short.MAX_VALUE)
                    .addGroup(jPanel17Layout.createSequentialGroup()

.addGroup(jPanel17Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING, false)
                            .addComponent(jLabel36,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                            .addComponent(jLabel35,
javax.swing.GroupLayout.PREFERRED_SIZE, 82,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel17Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING)
                            .addComponent(jTextField24,
javax.swing.GroupLayout.DEFAULT_SIZE, 358, Short.MAX_VALUE)
                            .addComponent(jTextField34,
javax.swing.GroupLayout.DEFAULT_SIZE, 358, Short.MAX_VALUE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                        .addComponent(jButton10,
javax.swing.GroupLayout.DEFAULT_SIZE, 87, Short.MAX_VALUE)))
                .addContainerGap())
        );
        jPanel17Layout.setVerticalGroup(

jPanel17Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
            .addGroup(jPanel17Layout.createSequentialGroup()
                .addComponent(jScrollPane8,
javax.swing.GroupLayout.PREFERRED_SIZE, 175,
javax.swing.GroupLayout.PREFERRED_SIZE)

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

.addGroup(jPanel17Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jLabel35)
                    .addComponent(jTextField24,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

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

.addGroup(jPanel17Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jLabel36)
                    .addComponent(jTextField34,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jButton10))
```

```
                .addContainerGap())
        );

        javax.swing.GroupLayout jPanel15Layout = new
javax.swing.GroupLayout(jPanel15);
        jPanel15.setLayout(jPanel15Layout);
        jPanel15Layout.setHorizontalGroup(

jPanel15Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel15Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel15Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.TRAILING)
                    .addComponent(jPanel17,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                    .addComponent(jPanel16,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
                .addContainerGap())
        );
        jPanel15Layout.setVerticalGroup(

jPanel15Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
            .addGroup(jPanel15Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jPanel16,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jPanel17,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(15, Short.MAX_VALUE))
        );
        jTabbedPane1.addTab("Phone Request", jPanel15);

        jMenu1.setText("File");
        jMenuItem1.setText("Exit");
        jMenuItem1.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jMenuItem1ActionPerformed(evt);
            }
        });

        jMenu1.add(jMenuItem1);

        jMenuBar1.add(jMenu1);

        jMenu2.setText("About");
```

```
        jMenuItem2.setText("About");
        jMenuItem2.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent
evt) {
                jMenuItem2ActionPerformed(evt);
            }
        });

        jMenu2.add(jMenuItem2);

        jMenuBar1.add(jMenu2);

        setJMenuBar(jMenuBar1);

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

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jTabbedPane1,
javax.swing.GroupLayout.DEFAULT_SIZE, 594, Short.MAX_VALUE)
        );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jTabbedPane1,
javax.swing.GroupLayout.DEFAULT_SIZE, 577, Short.MAX_VALUE)
        );
        pack();
    }// </editor-fold>//GEN-END:initComponents

    private void jButton10ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jButton10ActionPerformed
    jTextArea6.setText("");
    getResultsArriveInfo();
    jTextField24.setText("");
    jTextField34.setText("");
    }//GEN-LAST:event_jButton10ActionPerformed

    private void jButton9ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jButton9ActionPerformed
    jTextArea5.setText("");
    getResultsDepartInfo();
    jTextField33.setText("");
    jTextField13.setText("");
    }//GEN-LAST:event_jButton9ActionPerformed

    private void jTextField13KeyPressed(java.awt.event.KeyEvent evt)
{//GEN-FIRST:event_jTextField13KeyPressed
    if (evt.getKeyCode() == KeyEvent.VK_ENTER) {
      protocol.broadcastMessage(jTextField13.getText());
        jTextField13.setText("");
      }
    }//GEN-LAST:event_jTextField13KeyPressed

    private void jMenuItem2ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jMenuItem2ActionPerformed
    new About(this).setVisible(true);
    }//GEN-LAST:event_jMenuItem2ActionPerformed
```

```
    private void jButton2ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jButton2ActionPerformed
        jTextArea4.setText("");
        getResultsArrive();
        jTextField1.setText("");
    }//GEN-LAST:event_jButton2ActionPerformed

    private void jButton1ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jButton1ActionPerformed
        jTextArea3.setText("");
        getResultsDepart();
        jTextField2.setText("");
    }//GEN-LAST:event_jButton1ActionPerformed

    private void jButton4ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jButton4ActionPerformed
        String dbluetoothID=jTextField29.getText();
        String company=jTextField4.getText();
        String regno=jTextField5.getText();
        String destination=jTextField6.getText();
        String timedepart=jTextField7.getText();
        String platform=jTextField8.getText();

        String insertStr="";

        try{

            insertStr="insert into depart (dbluetoothID, company,
regno, destination, timedepart, platform ) values("
                    +quotate(dbluetoothID)+","
                    +quotate(company)+","
                    +quotate(regno)+","
                    +quotate(destination)+","
                    +quotate(timedepart)+","
                    +quotate(platform)
                    +")";

            int done=stmt.executeUpdate(insertStr);
            getContentPane().removeAll();
            initComponents();

        } catch(Exception e){
            e.printStackTrace();
        }
    }//GEN-LAST:event_jButton4ActionPerformed

    private void jButton5ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jButton5ActionPerformed
        jTextField29.setText("");
        jTextField4.setText("");
        jTextField5.setText("");
        jTextField6.setText("");
        jTextField7.setText("");
        jTextField8.setText("");
    }//GEN-LAST:event_jButton5ActionPerformed

    private void jButton3ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jButton3ActionPerformed
        String dbluetoothID=jTextField9.getText();
        String company=jTextField10.getText();
```

```
        String regno=jTextField25.getText();
        String destination=jTextField26.getText();
        String timedepart=jTextField27.getText();
        String platform=jTextField28.getText();

        String insertStr="";

        try{
            insertStr="update depart set " +
                    "company="+quotate(company)+
                    ",regno=" +quotate(regno)+
                    ",destination="+quotate(destination)+
                    ",timedepart="+quotate(timedepart)+
                    ",platform="+quotate(platform)+
                    "where dbluetoothID ="+quotate(dbluetoothID)+";";

            int done=stmt.executeUpdate(insertStr);
            getContentPane().removeAll();
            initComponents();

        } catch(Exception e){
            e.printStackTrace();
        }
    }//GEN-LAST:event_jButton3ActionPerformed

    private void jButton6ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jButton6ActionPerformed
        jTextField9.setText("");
        jTextField10.setText("");
    }//GEN-LAST:event_jButton6ActionPerformed

    private void jButton7ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jButton7ActionPerformed
        String dbluetoothID=jTextField11.getText();
        String regno=jTextField12.getText();


        String insertStr="";
        String insertStr1="";

        try{
            insertStr="delete from depart where dbluetoothID ="
                    +quotate(dbluetoothID)
                    +";";

            insertStr1="delete from depart where regno ="
                    +quotate(regno)
                    +";";

            int done=stmt.executeUpdate(insertStr);
            int done1=stmt.executeUpdate(insertStr1);

            getContentPane().removeAll();
            initComponents();

        } catch(Exception e){
            e.printStackTrace();
        }
    }//GEN-LAST:event_jButton7ActionPerformed
```

```
    private void jButton8ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jButton8ActionPerformed
        jTextField11.setText("");
        jTextField12.setText("");
    }//GEN-LAST:event_jButton8ActionPerformed

    private void jButton11ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jButton11ActionPerformed
        String abluetoothID=jTextField14.getText();
        String acompany=jTextField15.getText();
        String aregno=jTextField16.getText();
        String origin=jTextField17.getText();
        String timearrive=jTextField18.getText();
        String platform=jTextField19.getText();

        String insertStr="";

        try{

            insertStr="insert into arrive (abluetoothID, acompany,
aregno, origin, timearrive, platform ) values("
                    +quotate(abluetoothID)+","
                    +quotate(acompany)+","
                    +quotate(aregno)+","
                    +quotate(origin)+","
                    +quotate(timearrive)+","
                    +quotate(platform)
                    +")";

            int done=stmt.executeUpdate(insertStr);
            getContentPane().removeAll();
            initComponents();

        } catch(Exception e){
            e.printStackTrace();
        }
    }//GEN-LAST:event_jButton11ActionPerformed

    private void jButton12ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jButton12ActionPerformed
        jTextField14.setText("");
        jTextField15.setText("");
        jTextField16.setText("");
        jTextField17.setText("");
        jTextField18.setText("");
        jTextField19.setText("");
    }//GEN-LAST:event_jButton12ActionPerformed

    private void jButton13ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jButton13ActionPerformed
        String abluetoothID=jTextField21.getText();
        String acompany=jTextField20.getText();
        String aregno=jTextField3.getText();
        String origin=jTextField30.getText();
        String timearrive=jTextField31.getText();
        String platform=jTextField32.getText();


        String insertStr="";
```

```
        try{
            insertStr="update arrive set " +
                    "acompany="+quotate(acompany)+
                    ",aregno=" +quotate(aregno)+
                    ",origin="+quotate(origin)+
                    ",timearrive="+quotate(timearrive)+
                    ",platform="+quotate(platform)+
                    "where abluetoothID ="+quotate(abluetoothID)+";";

            int done=stmt.executeUpdate(insertStr);
            getContentPane().removeAll();
            initComponents();

        } catch(Exception e){
            e.printStackTrace();
        }
    }//GEN-LAST:event_jButton13ActionPerformed

    private void jButton14ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jButton14ActionPerformed
        jTextField21.setText("");
        jTextField20.setText("");
    }//GEN-LAST:event_jButton14ActionPerformed

    private void jButton15ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jButton15ActionPerformed
        String abluetoothID=jTextField22.getText();
        String aregno=jTextField23.getText();

        String insertStr="";
        String insertStr1="";

        try{
            insertStr="delete from arrive where abluetoothID ="
                    +quotate(abluetoothID)
                    +";";

            insertStr1="delete from arrive where aregno ="
                    +quotate(aregno)
                    +";";

            int done=stmt.executeUpdate(insertStr);
            int done1=stmt.executeUpdate(insertStr1);
            getContentPane().removeAll();
            initComponents();

        } catch(Exception e){
            e.printStackTrace();
        }
    }//GEN-LAST:event_jButton15ActionPerformed

    private void jButton16ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jButton16ActionPerformed
        jTextField22.setText("");
        jTextField23.setText("");
    }//GEN-LAST:event_jButton16ActionPerformed

    private void formWindowClosing(java.awt.event.WindowEvent evt)
{//GEN-FIRST:event_formWindowClosing
    mdbc.close(stmt);
    mdbc.destroy();
```

```
    }//GEN-LAST:event_formWindowClosing

    private void jMenuItem1ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jMenuItem1ActionPerformed
    System.exit(0);
    }//GEN-LAST:event_jMenuItem1ActionPerformed

    private void addMessage(String userName, String msg) {
            Log.log("Chat msg from user: " + userName + ", " + msg);
            StringBuffer buff = new StringBuffer();
            Calendar cal = Calendar.getInstance();
            int h = cal.get(Calendar.HOUR_OF_DAY);
            int m = cal.get(Calendar.MINUTE);
            buff.append("<");
            if (h < 10)
                  buff.append("0");
            buff.append(h);
            buff.append(":");
            if (m < 10)
                  buff.append("0");
            buff.append(m);
            buff.append("> ");
            if (userName != null) {
                  buff.append(userName);
                  buff.append(": ");
            }
            buff.append(msg);
            buff.append("\n");
            jTextArea5.insert(buff.toString(),
jTextArea5.getDocument().getLength());
            // TODO: move scroll pane window.
      }

        public void chatMessage(String userName, String msg) {
            addMessage(userName, msg);
      }

      public void chatLeave(String userName) {
            Log.log("Chatter leaving: " + userName);
            addMessage(null, userName + " is leaving.");
      }

      public void chatEnter(String userName) {
            Log.log("Chatter entering: " + userName);
            addMessage(null, userName + " has joined.");
      }

      public void setProtocol(ConnectionProtocol protocol) {
            this.protocol = protocol;
      }

    public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                while(true){
                new UI().setVisible(true);
                }
            }
        });
    }
```

```
    public ResultSet getResultFromdepart1() {
        ResultSet rs=null;
        try{
            rs=stmt.executeQuery("Select * from depart");
        }
        catch(SQLException e){}
        return rs;
    }

    public ResultSet getresultfromarrive() {

        ResultSet aa=null;

        try{
            aa=stmt.executeQuery("Select * from arrive");

        }
        catch(SQLException e){}

        return aa;
    }

    public void getResultsDepart() {
      String dbluetoothID= jTextField2.getText();

         try {
             //System.out.println("I am here");
             Class.forName("com.mysql.jdbc.Driver").newInstance();
           }
           catch (Exception e) {
             System.out.println(
               "Unable to register the JDBC Driver.\n" +
               "Make sure the JDBC driver is in the\n" +
               "classpath.\n");
           }

           // This URL specifies we are connecting with a database
server
           // on localhost.
           String url = "jdbc:mysql://localhost/bluetooth";

           // The username / password to connect under.
           String username = "root";
           String password = "qwerty";

           // Make a connection with the database.
           Connection con;
           try {
             con = DriverManager.getConnection(url, username,
password);
           }
           catch (SQLException e) {
             System.out.println(
               "Unable to make a connection to the database.\n" +
               "The reason: " + e.getMessage());
             return;
           }

           try {

           Statement stmt = con.createStatement();
```

```
            // using executeQuery():
            ResultSet rs = stmt.executeQuery(
                "Select * from depart where
dbluetoothID="+quotate(dbluetoothID)+"");


            // moving forward in the result set:
            while (rs.next()) {
                  String Company = rs.getString("company");
              String Regno = rs.getString("regno");
              String Destination = rs.getString("destination");
              String TimeDepart = rs.getString("timedepart");
              String Platform = rs.getString("platform");
              jTextArea3.append("Company:"
+Company+""+"\n\n"+"Registration No:"
+Regno+""+"\n\n"+"Destination:"+Destination+""+"\n\n");
              jTextArea3.append("Time Depart:"
+TimeDepart+""+"\n\n"+"Platform:" +Platform+"");
              }

              // Close the connection when finished
              con.close();

            }
            catch (SQLException e) {
              System.out.println(
                "An error occured\n" +
                "The SQLException message is: " + e.getMessage());
              return;
          }
    }

    public void getResultsArrive() {
      String abluetoothID= jTextField1.getText();

          try {
              //System.out.println("I am here");
              Class.forName("com.mysql.jdbc.Driver").newInstance();
            }
            catch (Exception e) {
              System.out.println(
                "Unable to register the JDBC Driver.\n" +
                "Make sure the JDBC driver is in the\n" +
                "classpath.\n");
            }

            // This URL specifies we are connecting with a database
server
            // on localhost.
            String url = "jdbc:mysql://localhost/bluetooth";

            // The username / password to connect under.
            String username = "root";
            String password = "qwerty";

            // Make a connection with the database.
            Connection con;
            try {
              con = DriverManager.getConnection(url, username,
password);
```

```
            }
            catch (SQLException e) {
              System.out.println(
                "Unable to make a connection to the database.\n" +
                "The reason: " + e.getMessage());
              return;
            }

            try {

            Statement stmt = con.createStatement();

            // using executeQuery():
            ResultSet rs = stmt.executeQuery(
                "Select * from arrive where
abluetoothID="+quotate(abluetoothID)+"");


            // moving forward in the result set:
            while (rs.next()) {
                  String Company = rs.getString("acompany");
              String Regno = rs.getString("aregno");
              String Origin = rs.getString("origin");
              String TimeArrive = rs.getString("timearrive");
              String Platform = rs.getString("platform");
              jTextArea4.append("Company:"
+Company+""+"\n\n"+"Registration No:"
+Regno+""+"\n\n"+"Origin:"+Origin+""+"\n\n");
            jTextArea4.append("Time Arrive:"
+TimeArrive+""+"\n\n"+"Platform:" +Platform+"");
            }

              // Close the connection when finished
              con.close();

            }
            catch (SQLException e) {
              System.out.println(
                "An error occured\n" +
                "The SQLException message is: " + e.getMessage());
              return;
          }
    }

    public void getResultsDepartInfo() {
      String company= jTextField33.getText();
        String destination = jTextField13.getText();

         try {
             //System.out.println("I am here");
             Class.forName("com.mysql.jdbc.Driver").newInstance();
           }
           catch (Exception e) {
             System.out.println(
               "Unable to register the JDBC Driver.\n" +
               "Make sure the JDBC driver is in the\n" +
               "classpath.\n");
           }

           // This URL specifies we are connecting with a database
server
```

```
            // on localhost.
            String url = "jdbc:mysql://localhost/bluetooth";

            // The username / password to connect under.
            String username = "root";
            String password = "qwerty";

            // Make a connection with the database.
            Connection con;
            try {
              con = DriverManager.getConnection(url, username,
password);
            }
            catch (SQLException e) {
              System.out.println(
                "Unable to make a connection to the database.\n" +
                "The reason: " + e.getMessage());
              return;
            }

            try {

            Statement stmt = con.createStatement();

            // using executeQuery():
            ResultSet rs = stmt.executeQuery(
                "Select * from depart where
company="+quotate(company)+"AND
destination="+quotate(destination)+"");
              System.out.println(quotate(destination));
              System.out.println(quotate(company));
         //SELECT * FROM depart WHERE company = "unimap" AND
destination = 'kangar';

            // moving forward in the result set:
            while (rs.next()) {
                  String Company = rs.getString("company");
              String Regno = rs.getString("regno");
              String Destination = rs.getString("destination");
              String TimeDepart = rs.getString("timedepart");
              String Platform = rs.getString("platform");
              jTextArea5.append("Company:"
+Company+""+"\n\n"+"Registration No:"
+Regno+""+"\n\n"+"Destination:"+Destination+""+"\n\n");
              jTextArea5.append("Time Depart:"
+TimeDepart+""+"\n\n"+"Platform:" +Platform+"");
              }

              // Close the connection when finished
              con.close();

            }
            catch (SQLException e) {
              System.out.println(
                "An error occured\n" +
                "The SQLException message is: " + e.getMessage());
              return;
          }
    }

    public void getResultsArriveInfo() {
```

```
      String acompany= jTextField24.getText();
        String origin = jTextField34.getText();

         try {
             //System.out.println("I am here");
             Class.forName("com.mysql.jdbc.Driver").newInstance();
           }
           catch (Exception e) {
             System.out.println(
               "Unable to register the JDBC Driver.\n" +
               "Make sure the JDBC driver is in the\n" +
               "classpath.\n");
           }

           // This URL specifies we are connecting with a database
server
           // on localhost.
           String url = "jdbc:mysql://localhost/bluetooth";

           // The username / password to connect under.
           String username = "root";
           String password = "qwerty";

           // Make a connection with the database.
           Connection con;
           try {
             con = DriverManager.getConnection(url, username,
password);
           }
           catch (SQLException e) {
             System.out.println(
               "Unable to make a connection to the database.\n" +
               "The reason: " + e.getMessage());
             return;
           }

           try {

           Statement stmt = con.createStatement();

           // using executeQuery():
           ResultSet rs = stmt.executeQuery(
               "Select * from arrive where
acompany="+quotate(acompany)+"AND origin="+quotate(origin)+"");
                     //SELECT * FROM depart WHERE company = "unimap"
AND destination = 'kangar';

           // moving forward in the result set:
           while (rs.next()) {
                 String Company = rs.getString("acompany");
             String Regno = rs.getString("aregno");
             String Origin = rs.getString("origin");
             String TimeArrive = rs.getString("timearrive");
             String Platform = rs.getString("platform");
             jTextArea6.append("Company:"
+Company+""+"\n\n"+"Registration No:"
+Regno+""+"\n\n"+"Origin:"+Origin+""+"\n\n");
             jTextArea6.append("Time Arrive:"
+TimeArrive+""+"\n\n"+"Platform:" +Platform+"");
             }
```

```
          // Close the connection when finished
          con.close();

        }
        catch (SQLException e) {
          System.out.println(
            "An error occured\n" +
            "The SQLException message is: " + e.getMessage());
          return;
      }
}

public String quotate(String content){

    return "'"+content+"'";
}
// Variables declaration - do not modify//GEN-BEGIN:variables
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 jButton2;
private javax.swing.JButton jButton3;
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.JLabel jLabel1;
private javax.swing.JLabel jLabel10;
private javax.swing.JLabel jLabel11;
private javax.swing.JLabel jLabel12;
private javax.swing.JLabel jLabel13;
private javax.swing.JLabel jLabel14;
private javax.swing.JLabel jLabel15;
private javax.swing.JLabel jLabel16;
private javax.swing.JLabel jLabel17;
private javax.swing.JLabel jLabel18;
private javax.swing.JLabel jLabel19;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel20;
private javax.swing.JLabel jLabel21;
private javax.swing.JLabel jLabel22;
private javax.swing.JLabel jLabel23;
private javax.swing.JLabel jLabel24;
private javax.swing.JLabel jLabel25;
private javax.swing.JLabel jLabel26;
private javax.swing.JLabel jLabel27;
private javax.swing.JLabel jLabel28;
private javax.swing.JLabel jLabel29;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel30;
private javax.swing.JLabel jLabel31;
private javax.swing.JLabel jLabel32;
private javax.swing.JLabel jLabel33;
private javax.swing.JLabel jLabel34;
```

```
private javax.swing.JLabel jLabel35;
private javax.swing.JLabel jLabel36;
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.JLabel jLabel9;
private javax.swing.JMenu jMenu1;
private javax.swing.JMenu jMenu2;
private javax.swing.JMenuBar jMenuBar1;
private javax.swing.JMenuItem jMenuItem1;
private javax.swing.JMenuItem jMenuItem2;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel10;
private javax.swing.JPanel jPanel11;
private javax.swing.JPanel jPanel12;
private javax.swing.JPanel jPanel13;
private javax.swing.JPanel jPanel14;
private javax.swing.JPanel jPanel15;
private javax.swing.JPanel jPanel16;
private javax.swing.JPanel jPanel17;
private javax.swing.JPanel jPanel2;
private javax.swing.JPanel jPanel3;
private javax.swing.JPanel jPanel4;
private javax.swing.JPanel jPanel5;
private javax.swing.JPanel jPanel6;
private javax.swing.JPanel jPanel7;
private javax.swing.JPanel jPanel8;
private javax.swing.JPanel jPanel9;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JScrollPane jScrollPane2;
private javax.swing.JScrollPane jScrollPane3;
private javax.swing.JScrollPane jScrollPane4;
private javax.swing.JScrollPane jScrollPane5;
private javax.swing.JScrollPane jScrollPane6;
private javax.swing.JScrollPane jScrollPane7;
private javax.swing.JScrollPane jScrollPane8;
private javax.swing.JTabbedPane jTabbedPane1;
private javax.swing.JTabbedPane jTabbedPane2;
private javax.swing.JTabbedPane jTabbedPane3;
private javax.swing.JTable jTable1;
private javax.swing.JTable jTable2;
public static javax.swing.JTextArea jTextArea1;
public static javax.swing.JTextArea jTextArea2;
public static javax.swing.JTextArea jTextArea3;
public static javax.swing.JTextArea jTextArea4;
public static javax.swing.JTextArea jTextArea5;
private javax.swing.JTextArea jTextArea6;
public static javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField10;
private javax.swing.JTextField jTextField11;
private javax.swing.JTextField jTextField12;
public static javax.swing.JTextField jTextField13;
private javax.swing.JTextField jTextField14;
private javax.swing.JTextField jTextField15;
private javax.swing.JTextField jTextField16;
private javax.swing.JTextField jTextField17;
private javax.swing.JTextField jTextField18;
private javax.swing.JTextField jTextField19;
public static javax.swing.JTextField jTextField2;
```

```
    private javax.swing.JTextField jTextField20;
    private javax.swing.JTextField jTextField21;
    private javax.swing.JTextField jTextField22;
    private javax.swing.JTextField jTextField23;
    private javax.swing.JTextField jTextField24;
    private javax.swing.JTextField jTextField25;
    private javax.swing.JTextField jTextField26;
    private javax.swing.JTextField jTextField27;
    private javax.swing.JTextField jTextField28;
    private javax.swing.JTextField jTextField29;
    private javax.swing.JTextField jTextField3;
    private javax.swing.JTextField jTextField30;
    private javax.swing.JTextField jTextField31;
    private javax.swing.JTextField jTextField32;
    public static javax.swing.JTextField jTextField33;
    private javax.swing.JTextField jTextField34;
    private javax.swing.JTextField jTextField4;
    private javax.swing.JTextField jTextField5;
    private javax.swing.JTextField jTextField6;
    private javax.swing.JTextField jTextField7;
    private javax.swing.JTextField jTextField8;
    private javax.swing.JTextField jTextField9;
    // End of variables declaration//GEN-END:variables
    private ConnectionProtocol protocol;
    private MyDBConnection mdbc;
    private java.sql.Statement stmt;
    private Connection connection;
}
```

```
package ui;
```

```
import javax.bluetooth.LocalDevice;
import javax.microedition.io.Connector;
import javax.microedition.io.StreamConnectionNotifier;

public class ServerThread extends Thread {

	private ConnectionProtocol protocol;

	public ServerThread(ConnectionProtocol protocol) {
		this.protocol = protocol;
	}

	public void run() {
		try {
			LocalDevice device = LocalDevice.getLocalDevice();

	device.setDiscoverable(BluetoothSettings.DISCOVERY_MODE);
			String url = "btspp://localhost:" +
BluetoothSettings.UUID
					+ ";name=Bluetooth Bus Info";

			Log.log("Create server by uri: " + url);
			StreamConnectionNotifier notifier =
(StreamConnectionNotifier) Connector
					.open(url);

			while (true) { // infinite loop to accept
connections.
				Log.log("Waiting for connection...");
				protocol.handleServerConnection(new
WorkaroundStreamConnection(notifier.acceptAndOpen()));
			}
		} catch (Exception e) {
			Log.log("ServerThread-Exception", e);
		}
	}

}
```

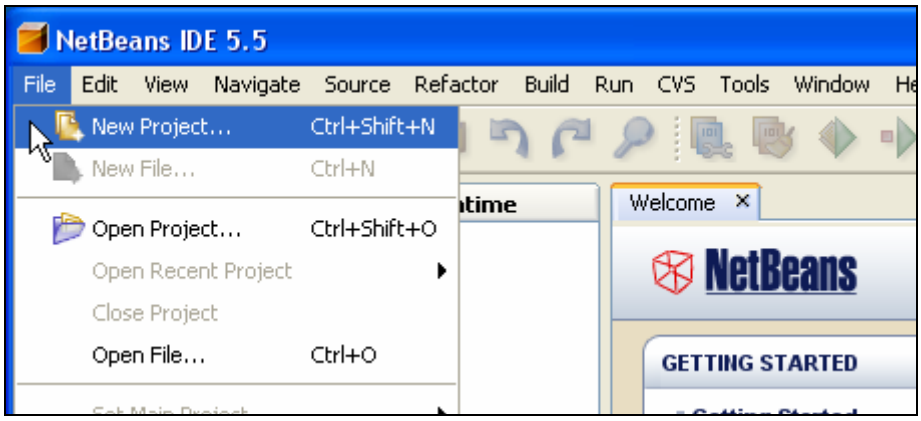
APPENDIX II

STEP TO DEVELOP JAVA APPLICATION IN NETBEANS 5.5

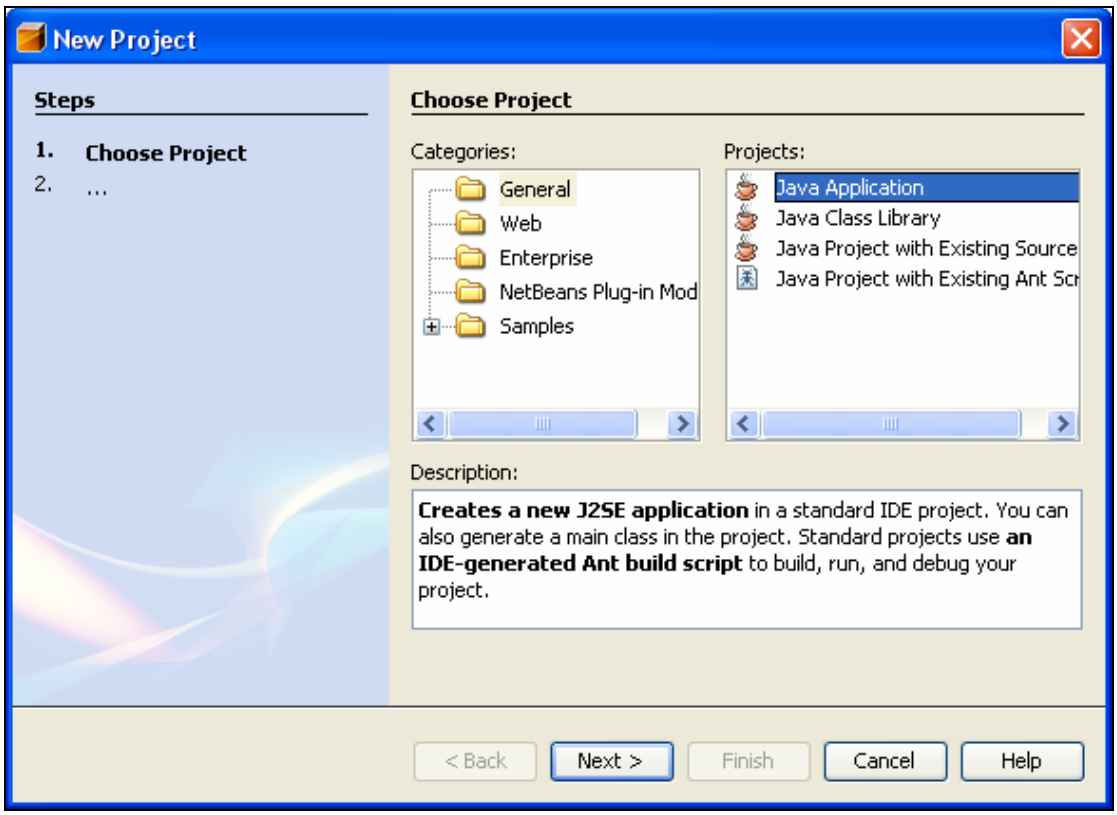
Create Java General Application in NetBeans IDE 5.5

1. **Start NetBeans IDE.**

2. **In the IDE, choose File > New Project, as shown in the figure below.**

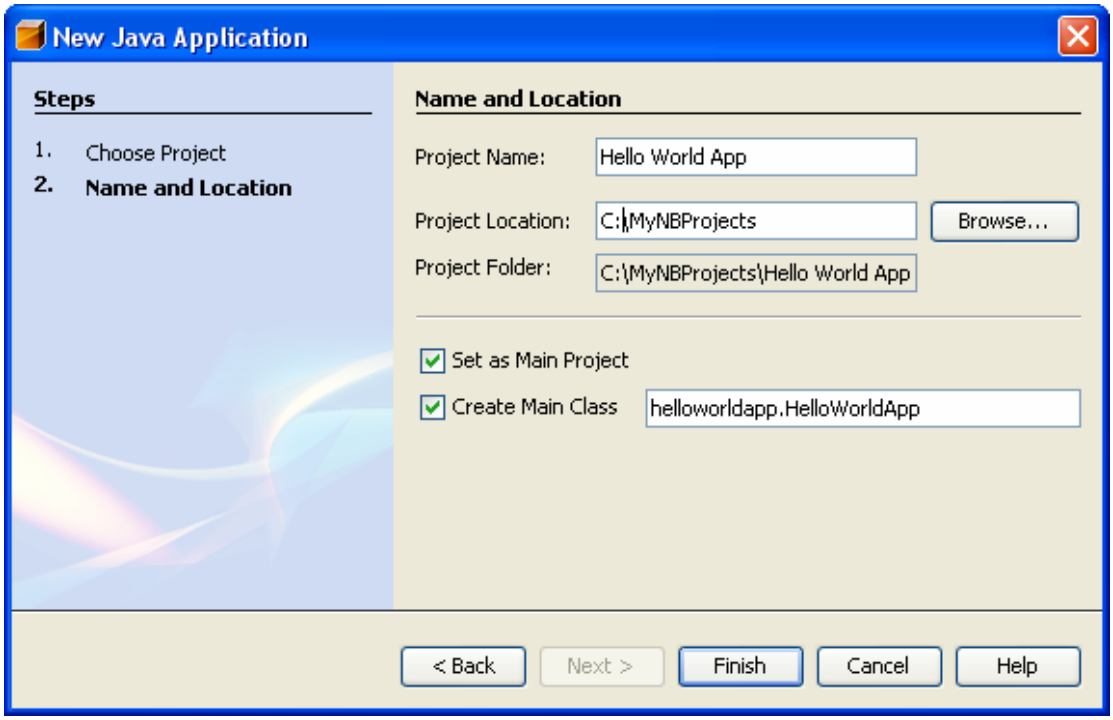


3. **In the New Project wizard, expand the General category and select Java Application as shown in the figure below. Then click Next.**



4. **In the Name and Location page of the wizard, do the following (as shown in the figure below):**

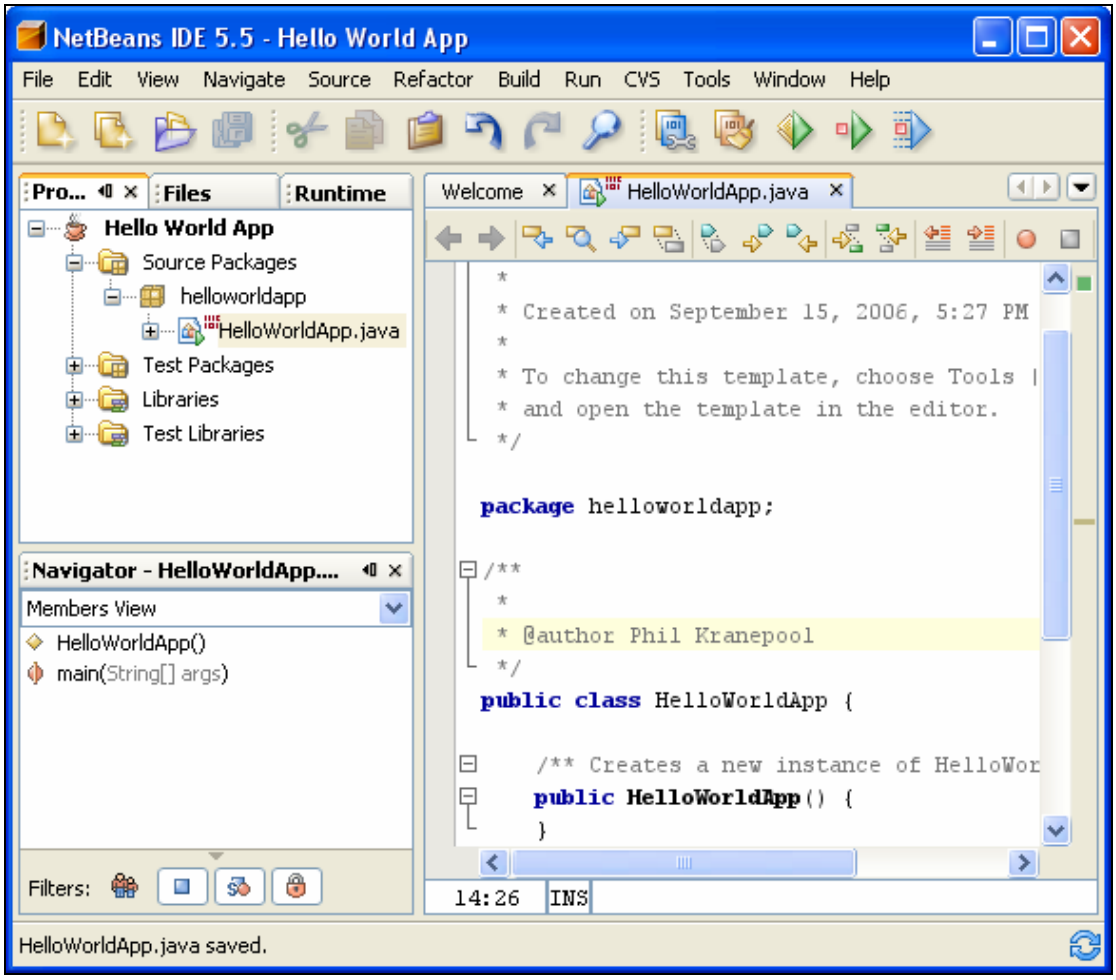
- In the Project Name field, type `Hello World App`.
- In the Create Main Class field, type `helloworldapp.HelloWorldApp`.
- Leave the Set as Main Project checkbox selected.



5. **Click Finish.**

The project is created and opened in the IDE. You should see the following components:

- The Projects window, which contains a tree view of the components of the project, including source files, libraries that your code depends on, and so on.
- The Source Editor window with a file called `HelloWorldApp` open.
- The Navigator window, which you can use to quickly navigate between elements within the selected class.



6. Adding Code to the Generated Source File

Because you have left the Create Main Class checkbox selected in the New Project wizard, the IDE has created a skeleton class for you. You can add the "Hello World!" message to the skeleton code by replacing the line:

```
// TODO code application logic here
```

with the line:

```
System.out.println("Hello World!");
```

Save the change by choosing File > Save.

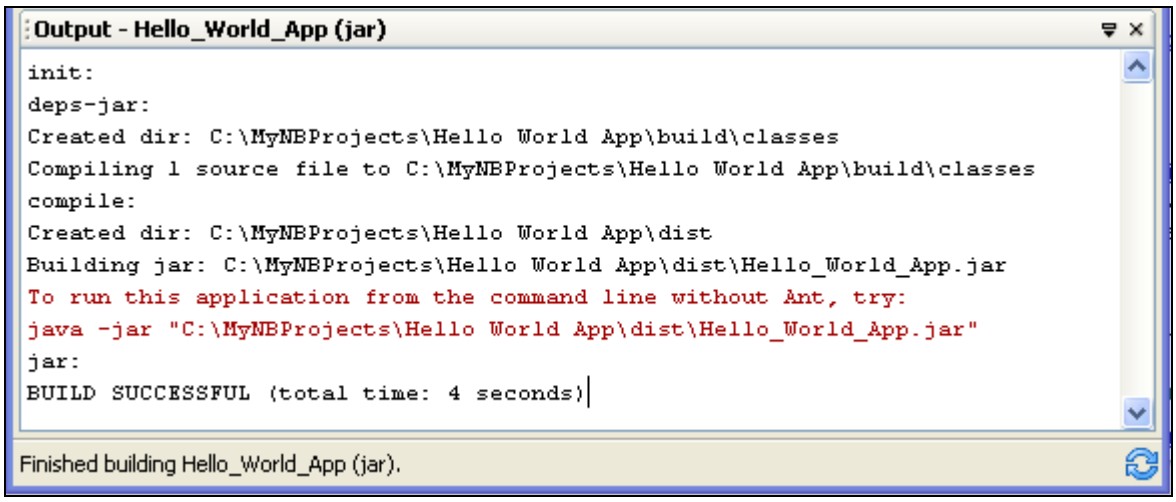
The file should look something like the following:

```
/*
* HelloWorldApp.java
*
* Created on September 15, 2006, 5:27 PM
*
* To change this template, choose Tools > Template Manager
* and open the template in the editor.
*/

package helloworldapp;

/**
* The HelloWorldApp class implements an application that
* simply displays "Hello World!" to the standard output.
*/
public class HelloWorldApp {

    /** Creates a new instance of HelloWorldApp */
    public HelloWorldApp() {
    }

    /**
    * @param args the command line arguments
    */
    public static void main(String[] args) {
        //Display "Hello World!"
        System.out.println("Hello World!");
    }

}
```

7. Compiling the Source File

To compile your source file, choose Build > Build Main Project from the IDE's main menu.

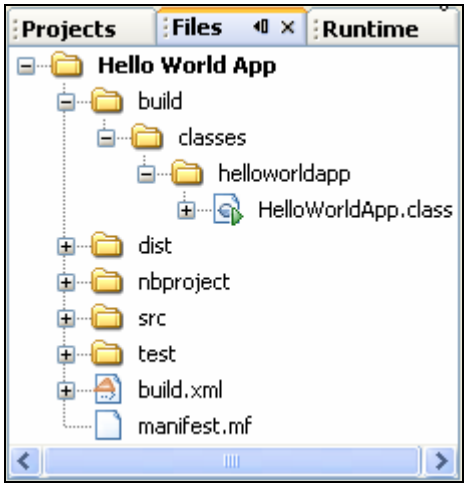
The Output window opens and displays output similar to what you see in the following figure.



If the build output concludes with the statement `BUILD SUCCESSFUL`, You have successfully compiled your program.

If the build output concludes with the statement `BUILD FAILED`, you probably have a syntax error in your code. Errors are reported in the Output window as hyper-linked text. You double-click such a hyper-link to navigate to the source of an error. You can then fix the error and once again choose Build > Build Main Project.

When you build the project, the byte code file `HelloWorldApp.class` is generated. You can see where the new file is generated by opening the Files window and expanding the `Hello World App/build/classes/helloworldapp` node as shown in the following figure.

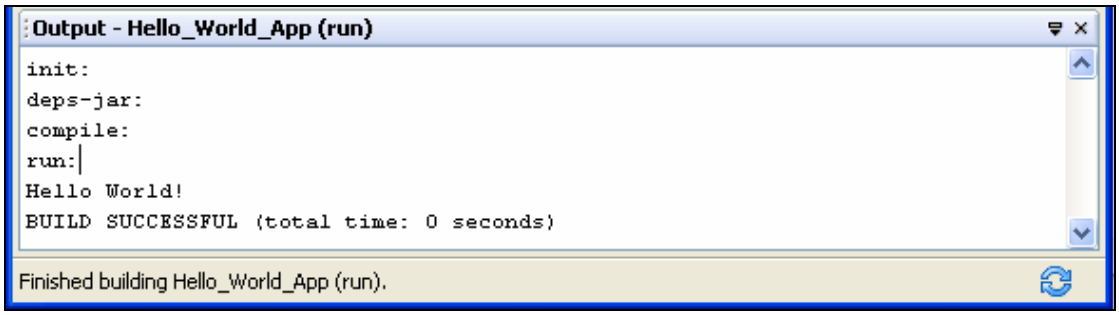


Now that you have built the project, you can run your program.

8. Running the Program

From the IDE's menu bar, choose Run > Run Main Project.

The next figure shows what you should now see.



APPENDIX III

MAKING A SIMPLE MYSQL CLIENT IN NETBEANS

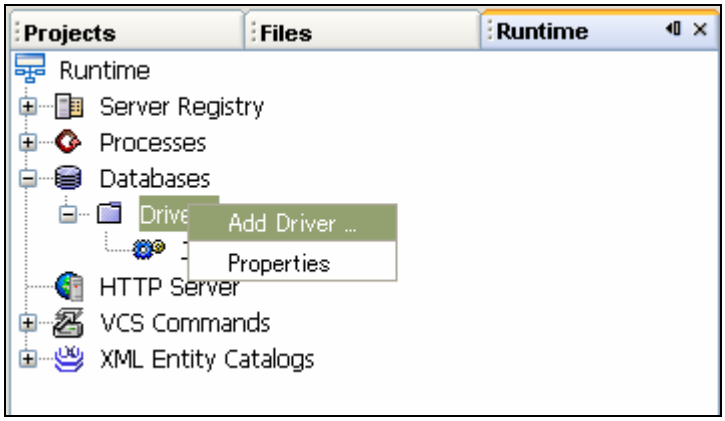
1. Connecting to MySQL database

I expect that the readers already have a suitable environment of MySQL database server working, having a database with an enough grant to a given user. In this sample,

database name: nonidb
user: noniko
password: (blank) <- because it's a merely a sample at all. If you have some important data, please set a proper password!

Also, you would get a JDBC driver file for MySQL. I prepared **mysql-connector-java-3.1.6-bin.jar** file through MySQL developer's web site.

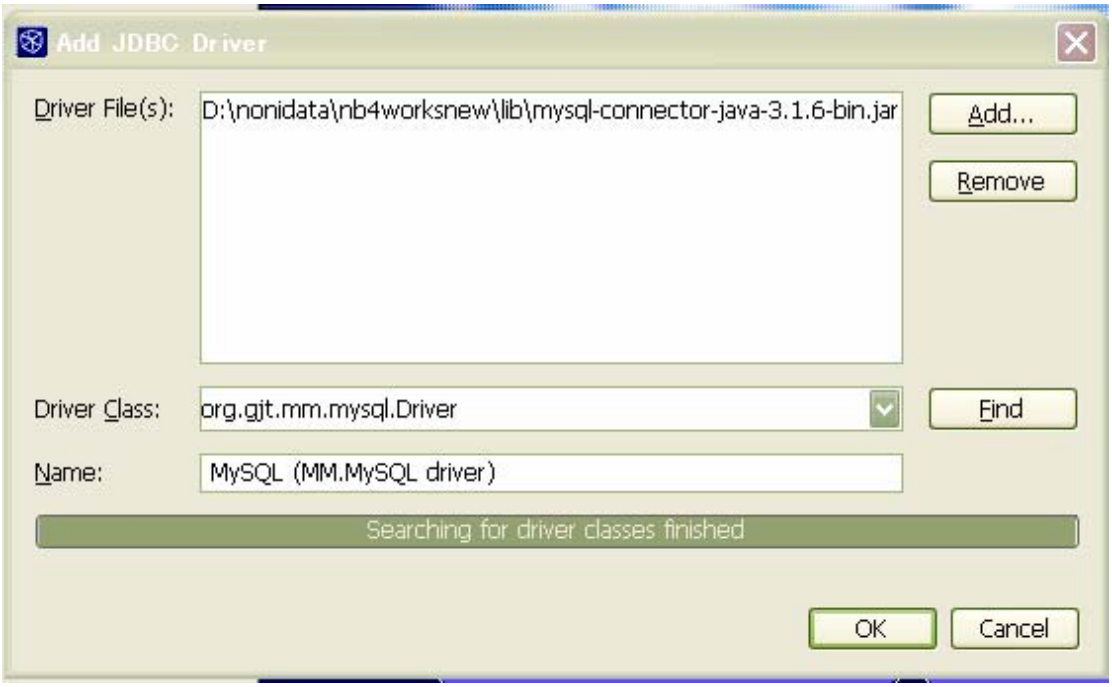
Now, let's go to NetBeans. Click the "Runtime" tab to switch the window you may usually see "Projects" or "Files" window.



2. Select "Add Driver" in Runtime Window

You will find "Databases" node here. open this node to get "Drivers" node under it. If you open more to show the content of "Drivers", you will find only "JDBC-ODBC". The first thing to do is add MySQL driver. Right-click the "Drivers" node to show context menu and select "Add Driver".

Then the setting window appears.

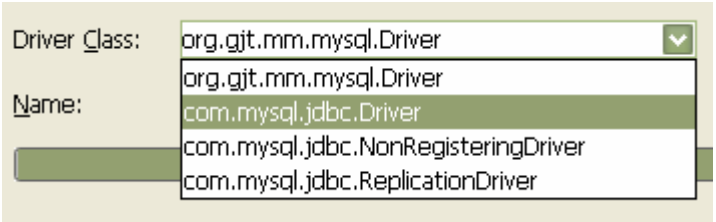


3. Find the location of JDBC driver file.

By clicking "Add" button you can search the driver file graphically. As the case above, I prepared D:\nonidata\nb4worksnew folder for contain project folders and lib subfolder to contain library files.

If you select the driver file's location, Driver Class and Name are set automatically. In the case of MySql JDBC driver, the old class name is set first. But click "Find" button, then the progress bar below became active and finally "Searching for driver classes finished" is shown.

Then open the combo of "Driver Class" setting. You can select the current class name of the JDBC Driver.

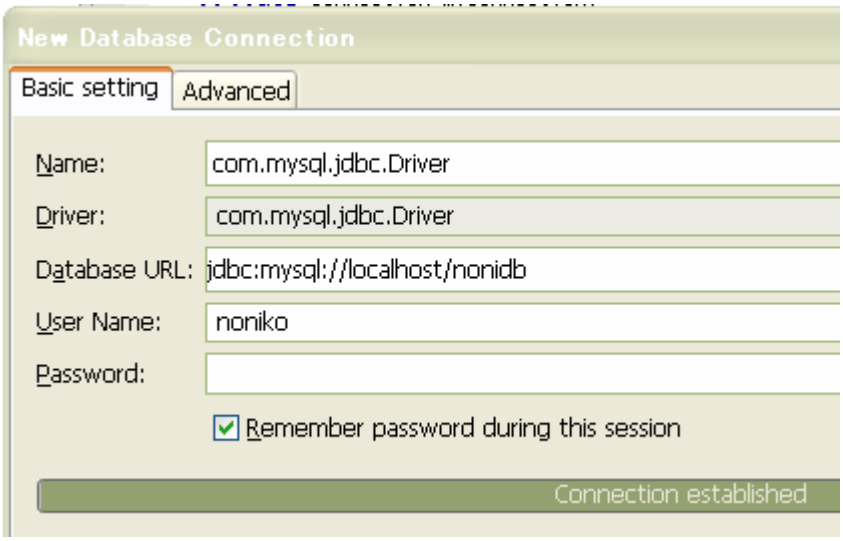


3. Change class name to that of current version.

Click "OK", then the new driver node is added under "Driver" node. Right-click it to select "Connect Using.." from context menu.

4. Select "Connect Using..." from context menu

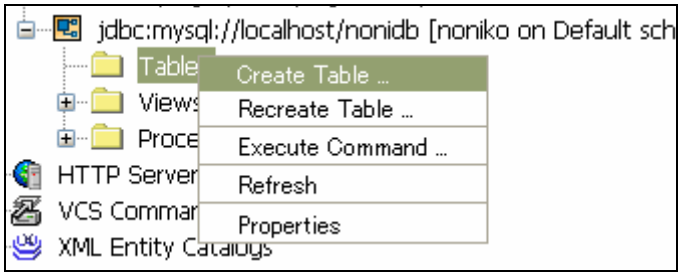
The setting window appears. Here Database URL, User Name and Password are needed (In this case Password is blank).Click OK then you could see "Connection established" message in the progress bar below. The page might be switched to "Advanced", but in this case there is no advanced setting needed.



After the connection established, a node for this connection is appeared. Let's proceed to next step .

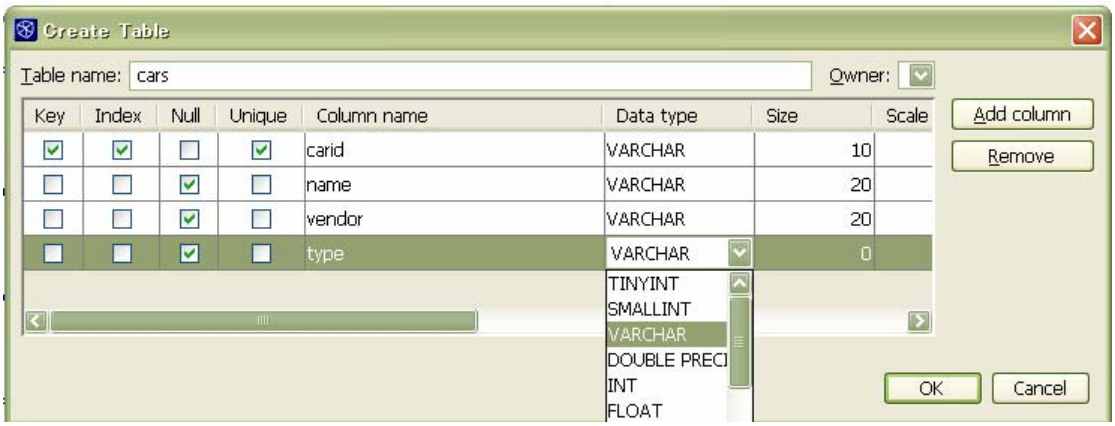
5. Creating a sample table

Right-Click the node for the connection established above, Select "Create Table...".



6. Select "Create Table" from context menu

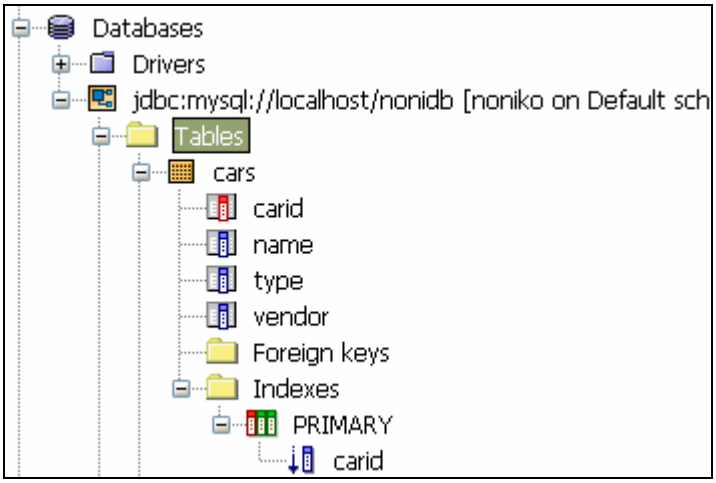
Then a window appears for setting the table to create. To display the required setting this window might have to be expanded manually. The important settings are Key, Column name, Data type and Size, depending on the Data type.



7. Creating a table in NetBeans

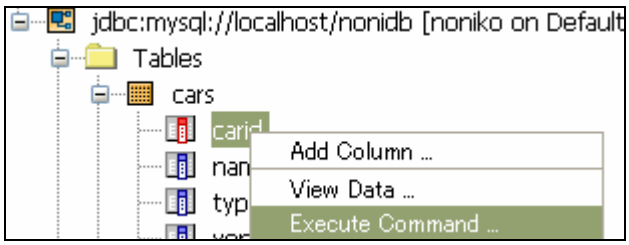
In this case I made a sample table "**cars**" having **carid**, **name**, **vendor** and **type** as fields. the field carid is the primary key. All the fields have data type of VARCHAR. Only carid has 10 for size and all others have 20.

Once the table created, you can see its node under the connection node. The structure of the table "cars" are displayed.

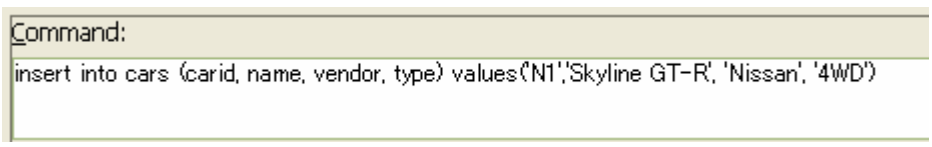


8. Insert the first data

Now let's insert the first row by sending SQL command. Right-Click any node under the connection node to display "Execute Commands..."menu item

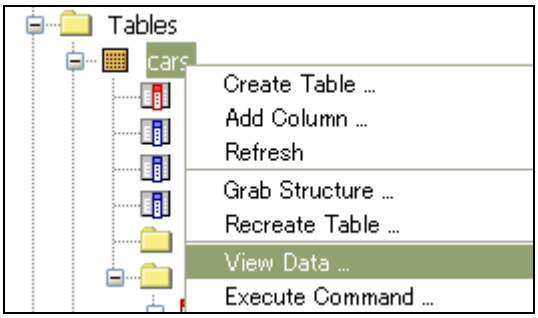


Then the large window of operating command appears at the same place of source editor. Too large, so I will not show the screenshot of the whole appearance. The important spots are, first of all, the text area for inputting command**.**



9. Input a whole command sentence.

Click "Execute" button and, if the command is valid, you can see the "Command successfully excuted" message below. More to make sure, right-click the "cars" node and select "View data".



Right-Click the "cars" node to view data.

Results:

carid	name	vendor	type
N1	Skyline GT-R	Nissan	4WD

10. First data inserted

Yeah! The first data were successfully inserted. But you might think the whole sentence "insert into..." bothering. I did. So I made up mind to build an application to insert data by more graphical operation.

11. Creating a new Java Project

Let's create a new Java Project for the application. Select from Menu bar "File"->"New Project". In the new project wizard, select "General"->"Java Application" as the project type. In my case the project name was "mysqloperation" and the location was "d:\nonidata\nb4worksnew\mysqloperation". Check off "create main project".

After creating the project, create new package. In my case, "mysqlop".

12. Creating a utility class for connecting database

Operation of database in Java is very complicated, requiring connection, statement, resultset and many try&catch. So I used a utility class dealing with such operation.

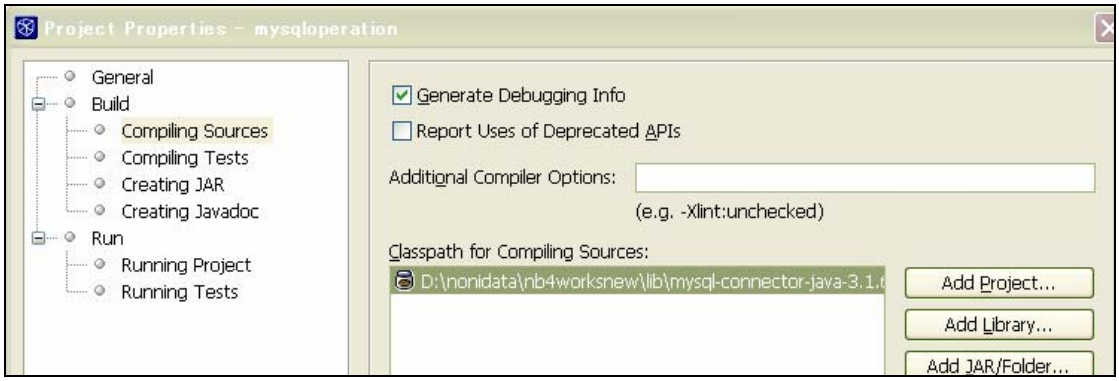
The whole code is shown here, MyDBConnection.java.

init() method is for connecting MySQL database.

getConnection() method is to be called after init()method, to give to the other class the established connection.

two **close()** methods and **destroy()** methods are for settling the end of operation. Closing connection, statement or resultset requires try&catch so they have been extracted here.

To use this file, classpath to the JDBC driver must be set for this project. Right-click the "**mysqloperation**" project and select "Properties..." from the context menu. In the "Project Properties" window, select "Build"->"Compiling Sources".Click "Add JAR/Folder..." button near the "Classpath for Compiling Source" setting , so you can search the location of the JDBC jar file.



Adding JDBC jar file to classpath

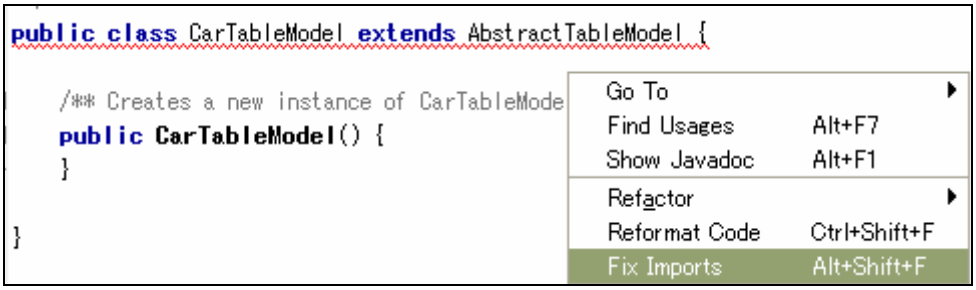
13. Creating the Table Model

Then I prepared a cusomized Table Model for the JTable to use in this app.

The whole code is shown here, CarTableModel.java.

To write this code I used some nice tools of NetBeans. First, create a new Java Class file named CarTableModel.java. It has a very plain and empty template, just with package and class declaration.

Edit manually the class declaration to extend "AbstractTableModel" class. The editor soon show the red and waved underline to show errors. One cause of this is lack of importing "javax.swing.table.AbstractTableModel" class. This error is fixed by right-clicking somewhere of the code and select "Fix Imports".



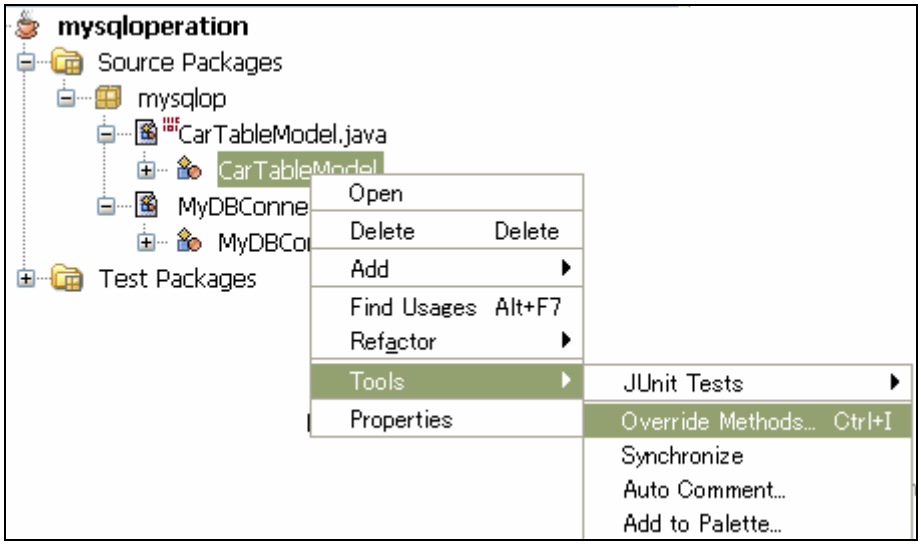
Use "Fix imports" tool

You can check the declaration

import javax.swing.table.AbstractTableModel;

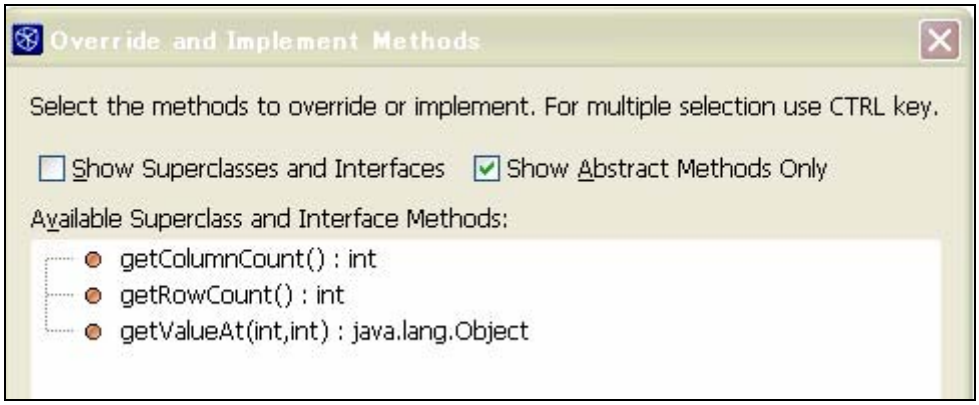
has been inserted near the top of the source code.

But still the red underline remains. The remaining cause of this is lack of implementation of proper abstract methods.
This is fixed by "overriding method" tool invoked this way...



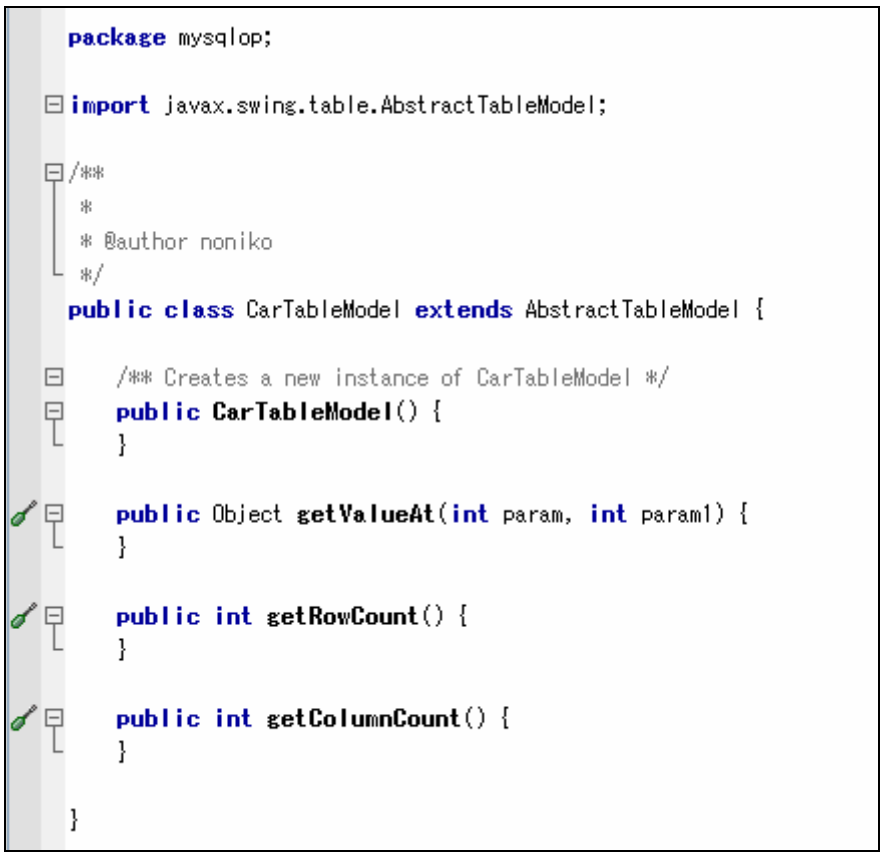
13. Use "Overrice Methods" tool

The required implementations are extracted checking "Show Abstract Methods Only".



14. Required implementations are listed

Select all of the three methods by clicking the node and "OK".



```
package mysqlop;

import javax.swing.table.AbstractTableModel;

/**
 *
 * @author noniko
 */
public class CarTableModel extends AbstractTableModel {

    /** Creates a new instance of CarTableModel */
    public CarTableModel() {
    }

    public Object getValueAt(int param, int param1) {
    }

    public int getRowCount() {
    }

    public int getColumnCount() {
    }

}
```

Required implementations are inserted automatically

Now the red line is gone!

In fact, we also need to override getColumnName(int param) method, otherwise we couldn't get customized headders of the table.

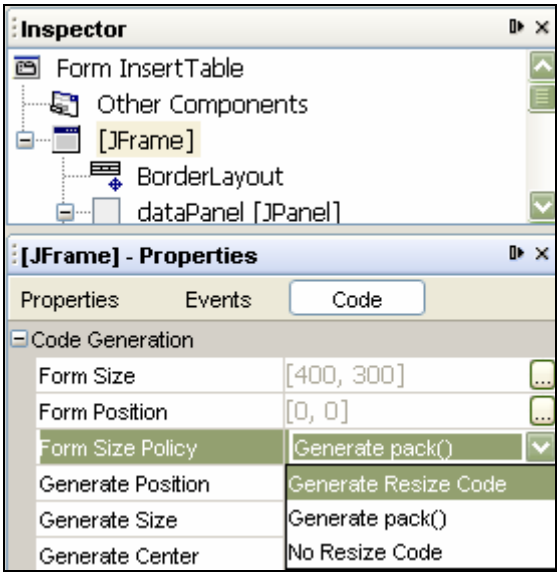
Let's fill the contents.
The code is specified to access the specific table "cars", so the number of column is fixed to 4.
Each column's name is also known and can be set directly.

This time I used Generics, a new feature of JDK5.0, for the first time. It worked successfully in NB (Clap hands!)

15. Creating the JFrame Form class-the Form design

Now let's create the JFrame Form Application, The main class of this project! "JFrame Form" can be selected in the new wizard. The name is "InsertTable" in my case, which created a source file "InsertTable.java".

First, design the form. It should be take a vertically long window, so I fixed the form size instead of use "packing".
Select "JFrame" node in the Inspector window to show its properties. Change "Form Size Policy" from default "Generate pack()" to "Generate Resize Code".



16. To fix the Form size.

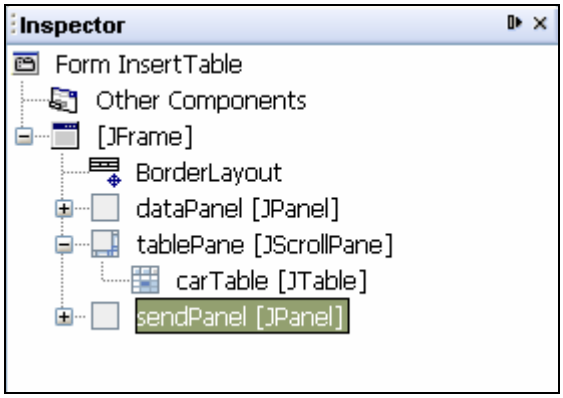
Then "Form Size" property becomes editable. Edit it as:

17. The Form size edited.

This form will be separated vertically into three parts:

(1)JTextfields or JComboBox to input the new data.
(2)JButton to send data and show message.
(3)The JTable.

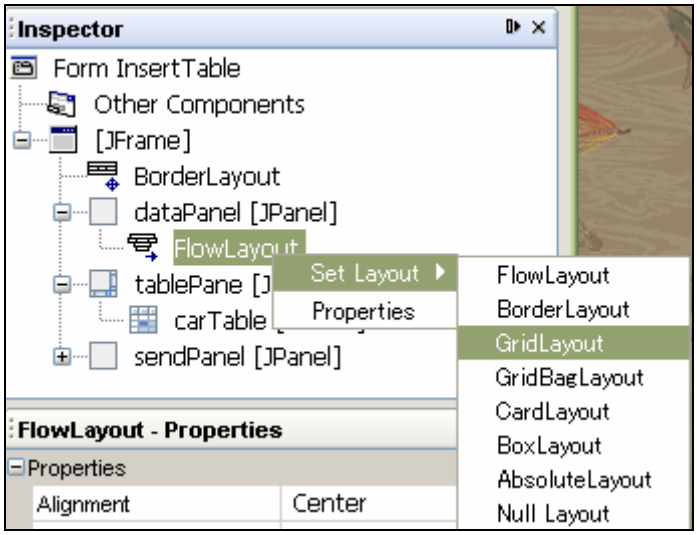
The first two should be located on eachJPanel, and the JTable on a JScrollPane. This is the basic parts.



18. JPanels and JScrollPane to locate the components.

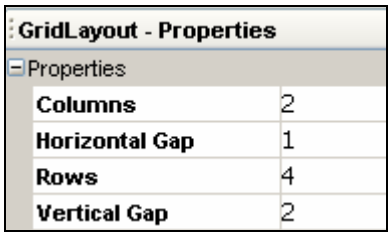
Notice the vertical positions of the nodes don't mean their actual locations. I mean, dataPanel, sendPanel and tablePane are located to North, Center and South of BorderLayout, respectively.

On dataPanel, 3JTextFields and one JComboBox to input data and 4JLabels for explanations should be located. I decided GridLayout is the best to set these locations. The default layout onJPanel is FlowLayout in NetBeans. It must be changed:

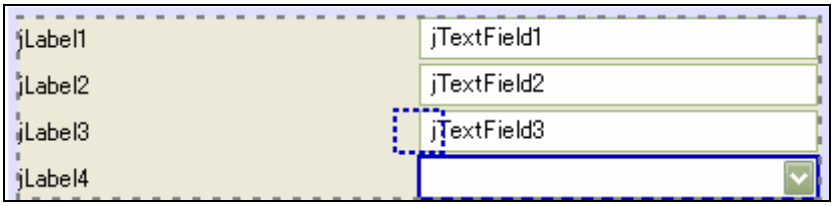


19. Change to GridLayout

Then properties of this Grid Layout are displayed. Set column number to 2, row number to 4.

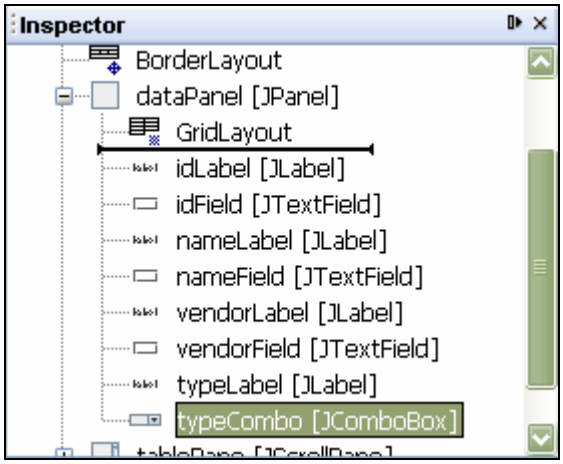


Then locate the components. They could be moved by drag and drop on the form, if you are not satisfied with the location.



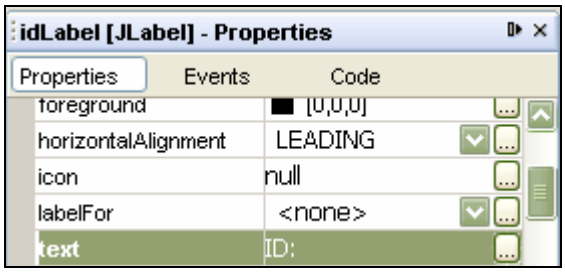
20. Moving the JComboBox within the GridLayout.

The component might escape to a very wrong position, as a different panel. Don't worry; you can move its location by drag and drop within the Inspector window, too.



21. Moving the JComboBox within the Inspector window.

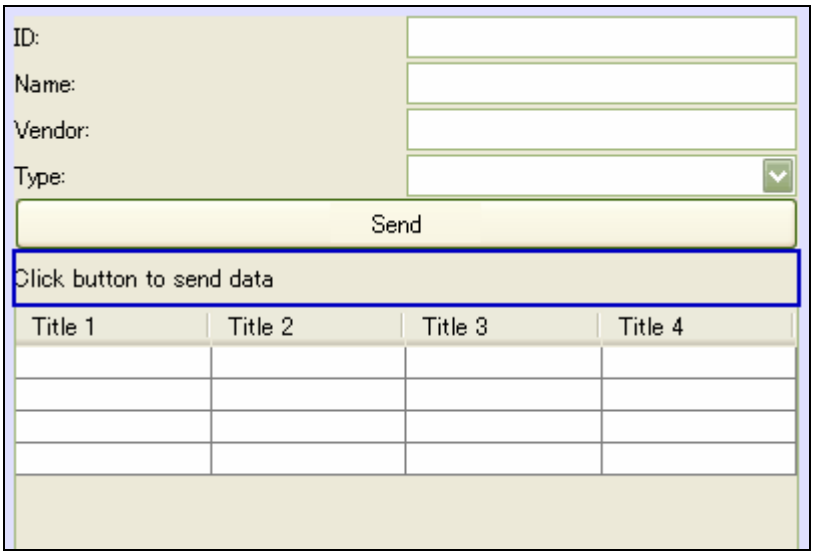
After all components are set, customize their name to show their roles.
Also customize text property of each JLabel, and eraze that of each JTextField. Here is an example of editing "text" property of JLabel:



22. Customizing "text" property to show "ID:"as JLabel

Then sendPanel to locate JButton and and JLabel. Also GridLayout is set for this JPanel, with 1 column and two rows. The name and "text" property are customized.

23. In this way the components, other than JTable, arearranged as below:

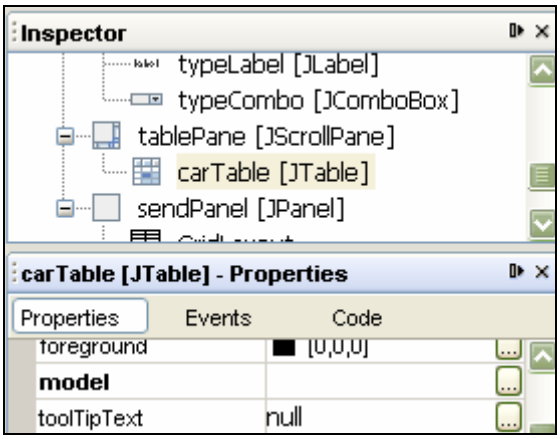


Components without JTable have been customized.

24. Customizing JTable

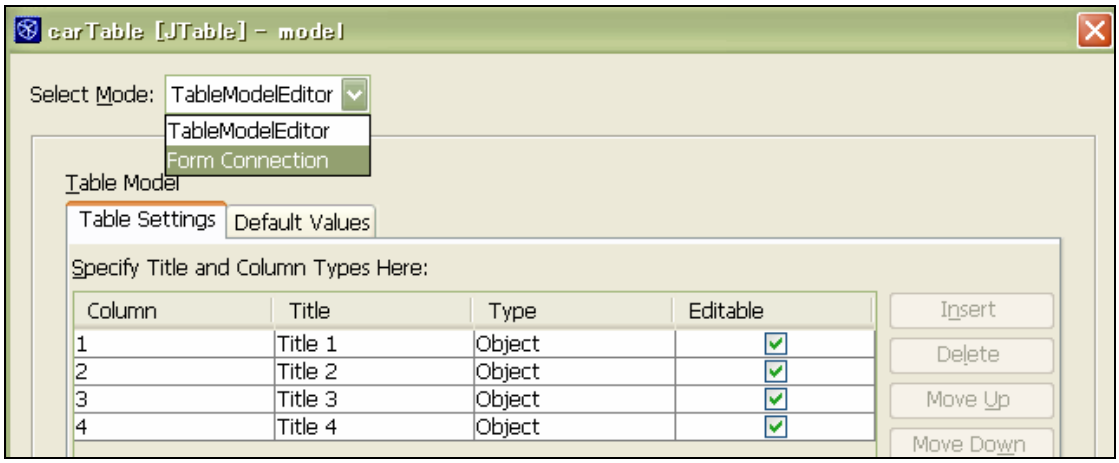
As shown above, at this point the JTable has 4x4 structure and empty data. Let's customize it to show the content of the table "cars" of MySQL database.

Select the JTable now is named "carTable" in the Inspector window to show its properties. Find the "model" property and click the "..." button at the right edge.



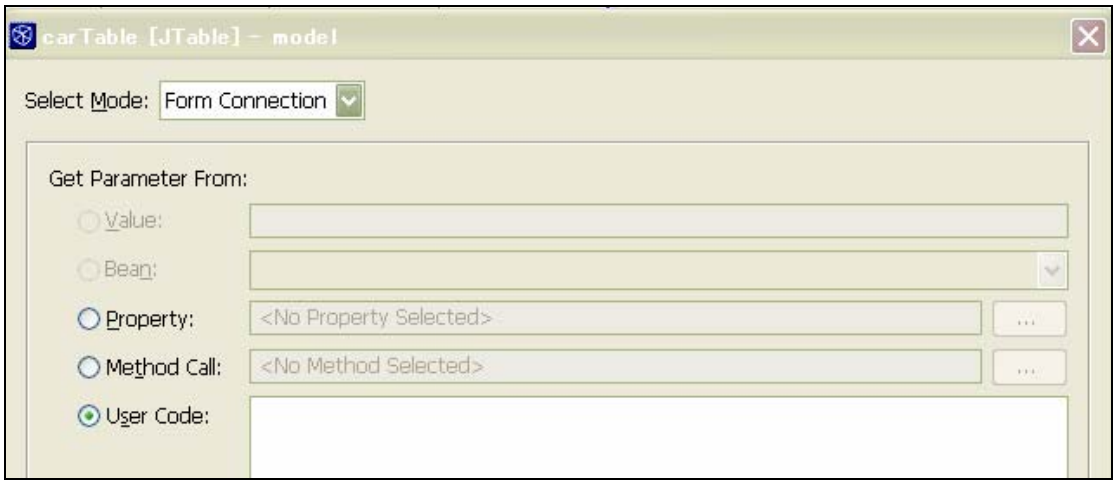
carTable and its "model" property

Then the editor window appears. The default 4x4 structure is set in this window. Let's change "**Select Mode**" from "TableModelEditor" to "**FormConnection**".



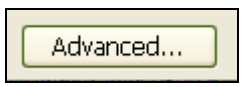
25. Switching from default Table Model Editor to Form Connection mode.

In the Form Connection mode, click the radio of "User Code" in the option window.



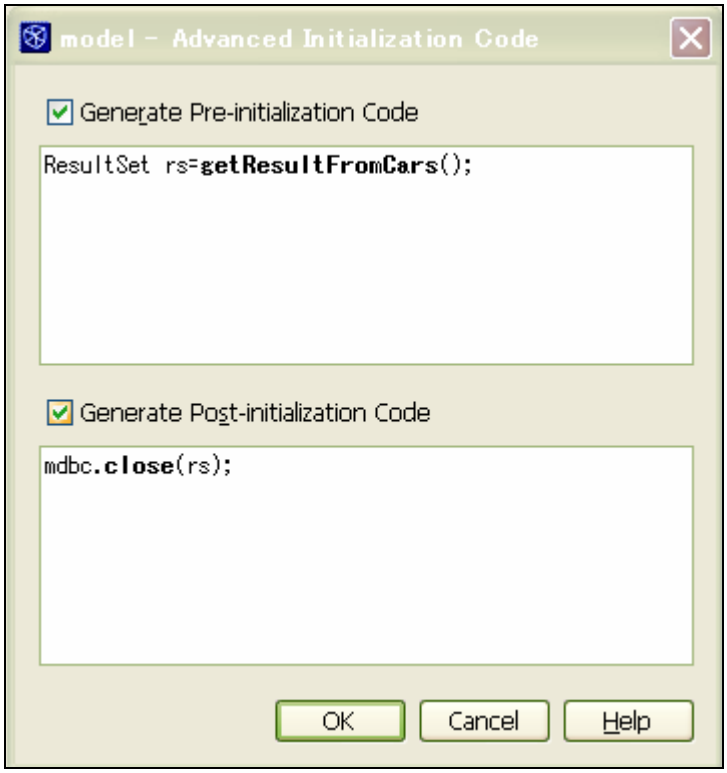
26. Form Connection mode

Then look at the bottom of this window. The "Advanced" button is there. Click it!



"Advanced" button at the bottom of the window

A window appears to edit before and after the code of setting the TableModel for the JTable. This time check both to edit.



The method named getResultFromCars should be created in the source code of "InsertTable.java" manually like

```
public ResultSet getResultFromCars() {

        ResultSet rs=null;

        try{
            rs=stmt.executeQuery("Select * from cars");

        }
        catch(SQLException e){}

        return rs;

    }
```

Click OK to return the "User Code" to fill.

Note that here semicolon";" to terminate the sentence SHOUD NOT BE input. Why? check the source code. Check the InitComponents() **method. Here!**

```
ResultSet rs=getResultFromCars();
carTable.setModel(new CarTableModel(rs));
mdbc.close(rs);
```

Light-blue "Protected""Uneditable" area? Who said so? Ho-ho-ho! You can edit frrreely by this means prepared by NetBeans!

26. First running of InsertTable

To make a first test run of this customized JTable, we still have something to do. First, declaration.

```
private MyDBConnection mdbc;
private java.sql.Statement stmt;
```

They are to be written somewhere in the source. Near the EOF, close to other variables, would be nice.

Then the constructor. Note that we should add throw declaration to it

```
public InsertTable() throws  Exception{
```

to avoid that annoying compile error "java.sql.SQLException is not reported...".

The whole content of the constructor is :

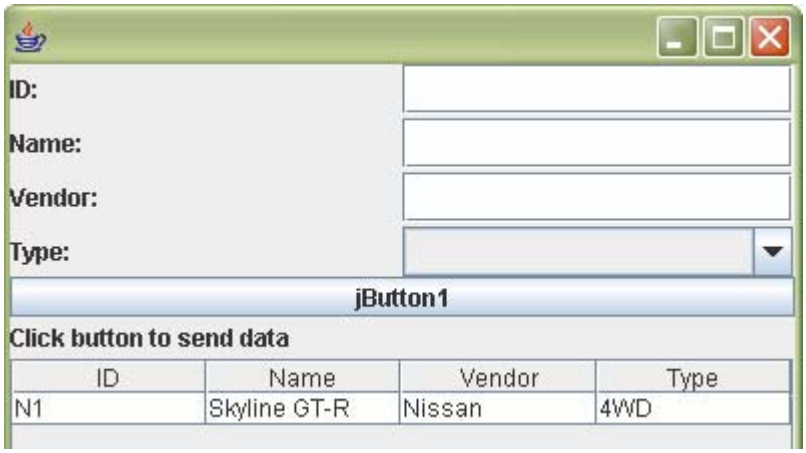
```
public InsertTable() throws  Exception{

         mdbc=new MyDBConnection();
         mdbc.init();
         Connection conn=mdbc.getMyConnection();
         stmt= conn.createStatement();

        initComponents();
    }
```

One more to avoid the same compile error. Edit the main method prepared by NetBeans:

```
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            try{
                  new InsertTable().setVisible(true);
               }
            catch(Exception e){
            }
        }
    });
}
```

Add try-catch to the main method

Now let's run the project! An window to select main class might be appear. In the list you will find only this "InsertTable.class". Select it.



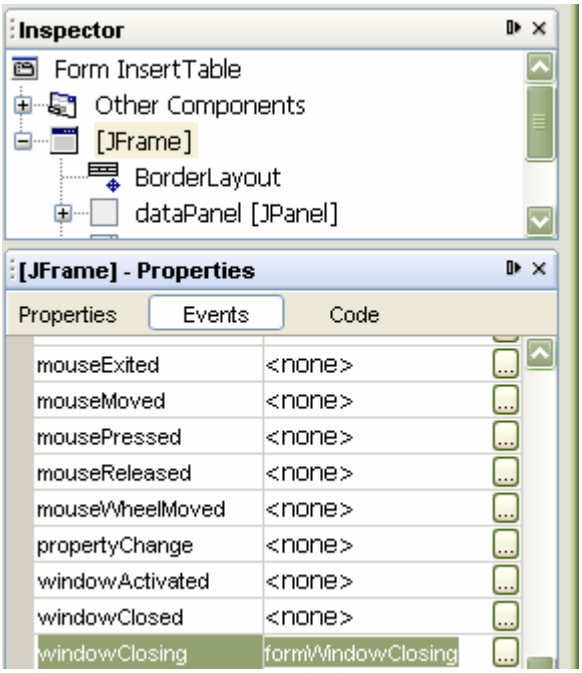
The tabledata of MySQL was shown

The only one row existing in "cars" table of MySQL database has been displayed!

27. Taking a little care...

Now that we could display existing data from MySQL to JTable, Let's feel settled and take a little care for better application.

One thing is a proper termination of this application, All the sessions with MySQL(connection, statement, resultsets) must be closed.
This application is terminated by closing the window. So we should add an event processing method for WindowClosing event.
Select "JFrame" in the Inspector window to show its properties window, and then click "events" to switch the page. Find "windowClosing" event. It is set to be <none> by default, but when you click the word <none>, it would be changed automatically to "formWindowClosing"...press the Enter key while this row is selected.



Setting windowClosing event of JFrame

Then the source code appears to enable to edit the method.

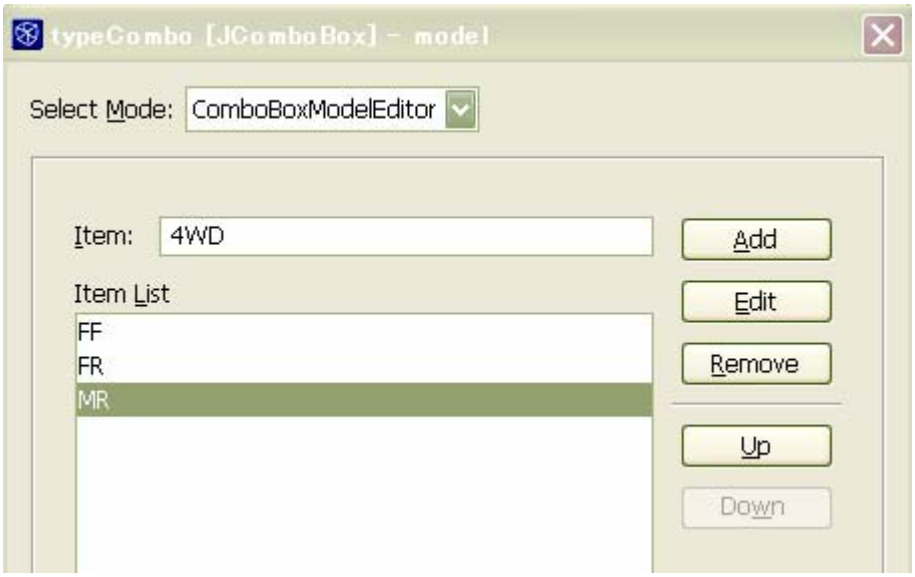
```
private void formWindowClosing(java.awt.event.WindowEvent evt) {
    // TODO add your handling code here:
    mdbc.close(stmt);
    mdbc.destroy();
}
```

Edit "formWindowClosing" method

Close the statement and the connection.

The other thing is to set items of the JComboBox named **typeCombo**. Yes, it should be equipped to select drive type of a car.

Select "typeCombo" node in the Inspector Window to show its properties. Find "**model**" property and click "..." button on the right edge, just like we did for setting the TableModel for JTable.
A window appears, This time we use the default **ComboBoxModelEditor**.



Input a new item in "Item:" field and click "Add" button. That's easy! I prepared the items "FF", "FR", "MR" and"4WD" as drive type. Are there anything else?

28. Inserting data to MySQL

The finishing spurt! Inserting data to MySQL should be done at the clicking "Send" button(the JButton named "**sendButton**").

In NetBeans' GUI Editor, it's very easy to make event processing method for JButton. Just double-click the JButton itself on the form.

The **"sendButtonActionPerformed"** method is created in the source file. Let's fill the content as this:

```
private void sendButtonActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_sendButtonActionPerformed
        // TODO add your handling code here:

        String carid=idField.getText();
        String name=nameField.getText();
        String vendor=vendorField.getText();
        String type=(String)typeCombo.getSelectedItem();

        String insertStr="";

        try{

            insertStr="insert into cars (carid, name, vendor, type)
values("
                    +quotate(carid)+","
                    +quotate(name)+","
                    +quotate(vendor)+","
                    +quotate(type)
                    +")";

            int done=stmt.executeUpdate(insertStr);

            commentLabel.setText("1 row inserted");

            getContentPane().removeAll();
            initComponents();

        }
        catch(Exception e){
            commentLabel.setText("Error occurred in inserting data");
            e.printStackTrace();
        }

    }//GEN-LAST:event_sendButtonActionPerformed
```

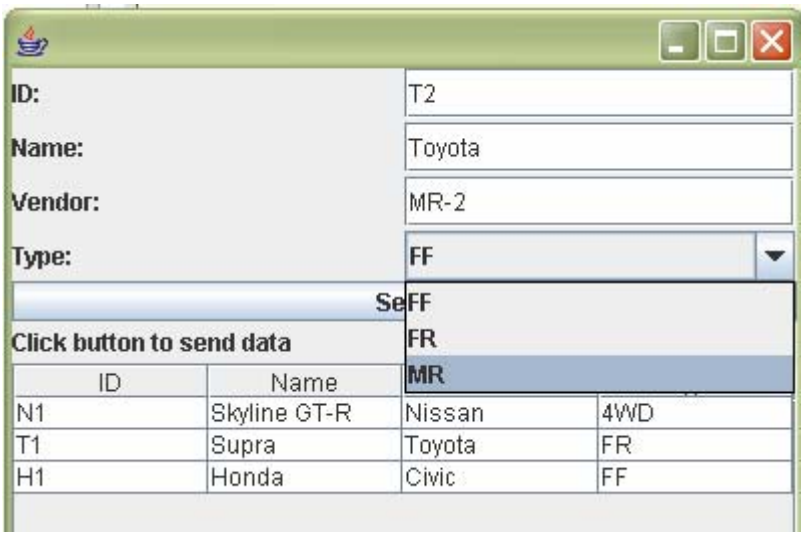
Note that **initComponents** method is called after inserting data, and after **removing the current comtent pane**.It's necessary to refresh the JTable's appearance...If I were skilled more, I could refresh only JTable, instead of the whole container. This is, as is called in Japanese a way like "Chopping tofu with the axe". Sorry!

Also note here the **quotate** method is called. I created it not to hit so many quotation marks:

```
public String quotate(String content){

        return "'"+content+"'";
    }
```

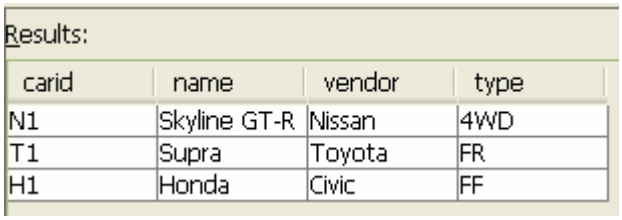
Now the whole code for InsertTable.java.

Run the project! Input data and click the "Send" button. Now you can see another row just inserted in the JTable. Repeat inserting a few more!



Operating the completed application

...And let's make sure these data will be viewed also from NetBeans' utility.



APPENDIX IV

PHONE SIDE SOURCE CODE

```
package ui;
```

```
import javax.bluetooth.BluetoothStateException;
import javax.bluetooth.LocalDevice;
import javax.microedition.midlet.MIDlet;
import javax.microedition.midlet.MIDletStateChangeException;

public class BluetoothBusInfo extends MIDlet {

	protected void startApp() throws MIDletStateChangeException {
		LcduiUI ui = new LcduiUI(this);
		ConnectionProtocol protocol = new ConnectionProtocol(ui);

		try {

	protocol.setUserName(LocalDevice.getLocalDevice().getFriendlyNa
me());
		} catch (BluetoothStateException e) {
			protocol.setUserName("user");
		}

		new ServerThread(protocol).start();
		new DiscoveryThread(protocol).start();

		ui.setCurrent();
	}

	protected void pauseApp() {
	}

	protected void destroyApp(boolean arg0) throws
MIDletStateChangeException {
		Log.clear();
	}

}
```

```
/*
 * BluetoothSettings.java
 *
 * Created on April 11, 2007, 4:00 PM
 *
 * To change this template, choose Tools | Template Manager
 * and open the template in the editor.
 */

package ui;

import javax.bluetooth.DiscoveryAgent;
import javax.bluetooth.ServiceRecord;
import javax.bluetooth.UUID;

public interface BluetoothSettings {

      /**
       * A constant UUID for Serial Port Profile.
       */
      static public final UUID UUID = new UUID("1101", true);

      /**
       * A constant discovery mode. In global mode no paring is
needed.
       */
      static public final int DISCOVERY_MODE = DiscoveryAgent.GIAC;

      /**
       * @see javax.bluetooth.ServiceRecord#getConnectionURL(int
requiredSecurity,
       *      boolean mustBeMaster)
       */
      public static final int SECURITY_OPTIONS =
ServiceRecord.NOAUTHENTICATE_NOENCRYPT;

      /**
       * @see javax.bluetooth.ServiceRecord#getConnectionURL(int
requiredSecurity,
       *      boolean mustBeMaster)
       */
      public static final boolean MUST_BE_MASTER = false;

      /**
       * A workaround for a bug in bluecove. BlueCove may crash if
getFriendlyName
       * method is used.
       *
       * @see javax.bluetooth.RemoteDevice#getFriendlyName(boolean
alwaysAsk)
       */
      public static final boolean WORKAROUND_BLUECOVE = true;

      /**
       * A constant number to search services from remote device.
Services are
       * searched couple of times from a device if service search
returned with
       * search error.
       *
```

```
         * @see
javax.bluetooth.DiscoveryListener#serviceSearchCompleted(int
         *      transID, int respCode)
         * @see javax.bluetooth.DiscoveryListener#SERVICE_SEARCH_ERROR
         */
        public static final int MAX_TRY_COUNT_TO_SEARCH_SERVICE = 2;

        /**
         * A constant number which tells how many seconds are slept
between
         * discoveries in discovery thread.
         */
        public static final int SLEEP_TIME_BEFORE_NEW_DISCOVERY = 20;

}
```

```
package ui;

import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.io.IOException;
import java.util.Enumeration;
import java.util.Hashtable;

import javax.bluetooth.BluetoothStateException;
import javax.bluetooth.LocalDevice;
import javax.microedition.io.Connection;
import javax.microedition.io.StreamConnection;

/**
 * A class that handles connections to remote chatters and the
protocol used in
 * messaging.
 *
 */
public class ConnectionProtocol {

	/**
	 * An interface for UI event handler.
	 */
	static public interface EventHandler {
		public void chatMessage(String userName, String msg);

		public void chatLeave(String userName);

		public void chatEnter(String userName);

		public void setProtocol(ConnectionProtocol protocol);
	}

	class RemoteChatter {
		Connection conn;

		DataOutputStream out;

		DataInputStream in;

		String userName;

		String btAddress;

		RemoteChatter(String btAddress, Connection conn,
DataInputStream in,
				DataOutputStream out, String uName) {
			this.btAddress = btAddress;
			this.conn = conn;
			this.in = in;
			this.out = out;
			this.userName = uName;
		}

		/**
		 * A method that starts the message reading thread which
contains the
		 * main chat loop. When the main loop starts the remote
chatter enters
```

```
             * in chat. Messages are read from the connection while
it's open. When
             * the connection closes chatter leaves from the chat.
             */
            void startMessageReadingThread() {
                  new Thread() {
                        public void run() {
                              ui.chatEnter(userName);
                              while (true) { // infinite loop to read
messages.
                                    try {
                                          ui.chatMessage(userName,
readString(in));
                                    } catch (IOException e) {
                                          ui.chatLeave(userName);
                                          close();
                                          break;
                                    }
                              }
                        }
                  }.start();
            }

            void close() {
                  synchronized (btAddress2remoteChatters) {
                        Log.log("Close chat connection.");
                        try {
                              conn.close();
                        } catch (IOException e) {
                        }
                        btAddress2remoteChatters.remove(btAddress);
                  }
            }
      }

      protected Hashtable btAddress2remoteChatters = new Hashtable();

      protected EventHandler ui;

      protected String userName;

      public ConnectionProtocol(EventHandler ui) {
            this.ui = ui;
            ui.setProtocol(this);
      }

      public void handleServerConnection(final StreamConnection
connection) {
            // A server connections shall be handled immediately.
Hence, a new
            // thread is started to do the actual protocol handling.
            new Thread() {
                  public void run() {
                        DataInputStream in;
                        try { // to read client's bluetooth address
                              in = connection.openDataInputStream();
                              Log.log("Read bt address.");
                              String btAddress = readString(in);
                              Log.log("bt addres is: " + btAddress);
                              handleConnection(connection, btAddress,
in, connection
```

```
                                          .openDataOutputStream());
                        } catch (Exception e) {

      Log.log("ConnectionProtocol.handleServerConnection-
Exception",e);
                        }
                  }
            }.start();
      }

      public void handleClientConnection(StreamConnection connection,
                  String btAddress) {
            DataOutputStream out;
            try { // to write my bluetooth address to server
                  out = connection.openDataOutputStream();
                  Log.log("Write bt address.");
                  writeString(out,
LocalDevice.getLocalDevice().getBluetoothAddress());

                  handleConnection(connection, btAddress, connection
                              .openDataInputStream(), out);
            } catch (Exception e) {
                  Log.log("ConnectionProtocol.handleClientConnection-
Exception", e);
                  try {
                        connection.close();
                  } catch (IOException e1) {
                  }
            }
      }

      protected void handleConnection(StreamConnection connection,
                  String btAddress, DataInputStream in,
DataOutputStream out)
                  throws Exception {

            btAddress = btAddress.toLowerCase();

            // The synchronization is very important here. Otherwise
there might be
            // two connection between devices A and B if both create
a connection to
            // each other at the same time.
            synchronized (btAddress2remoteChatters) {
                  if
(!btAddress2remoteChatters.containsKey(btAddress)) {
                        Log.log("Write user name.");
                        writeString(out, userName);
                        Log.log("Read user name.");
                        String userName = readString(in);
                        Log.log("userName: " + userName);
                        RemoteChatter chatConn = new
RemoteChatter(btAddress,
                                    connection, in, out, userName);
                        btAddress2remoteChatters.put(btAddress,
chatConn);
                        Log.log("Start message read thread");
                        chatConn.startMessageReadingThread();
                  } else {
                        connection.close();
                  }
```

```
            }
      }

      /**
       * @return true if connection is made already or the address is
local
       *         Bluetooth address.
       */
      public boolean hasConnection(String bluetoothAddress) {
            try {
                  return
btAddress2remoteChatters.containsKey(bluetoothAddress
                              .toLowerCase())
                              ||
bluetoothAddress.equalsIgnoreCase(LocalDevice

      .getLocalDevice().getBluetoothAddress());
            } catch (BluetoothStateException e) {
                  Log.log("ConnectionProtocol.hasConnection()", e);
                  return true;
            }
      }

      public void setUserName(String userName) {
            this.userName = userName;
            ui.chatEnter(userName);
      }

      /**
       * Broadcasts given message to other chatters and then sends
the message to
       * user's own chat UI.
       */
      public void broadcastMessage(String msg) {
            Log.log("Broadcast message: " + msg);
            for (Enumeration e = btAddress2remoteChatters.elements();
e
                        .hasMoreElements();) {
                  RemoteChatter chat = (RemoteChatter)
e.nextElement();
                  try {
                        writeString(chat.out, msg);
                  } catch (IOException e1) {
                        chat.close();
                  }
            }

            ui.chatMessage(userName, msg);
      }

      /*-
       *
       *  ----  Util methods ----
       */

      protected void writeString(DataOutputStream out, String str)
                  throws IOException {
            out.writeUTF(str);
            out.flush();
      }
```

```
	protected String readString(DataInputStream in) throws
IOException {
		return in.readUTF();
	}

}
```

```
/*
 * DiscoveryThread.java
 *
 * Created on April 11, 2007, 3:58 PM
 *
 * To change this template, choose Tools | Template Manager
 * and open the template in the editor.
 */

package ui;

import java.io.IOException;
import java.util.Vector;

import javax.bluetooth.DeviceClass;
import javax.bluetooth.DiscoveryAgent;
import javax.bluetooth.DiscoveryListener;
import javax.bluetooth.LocalDevice;
import javax.bluetooth.RemoteDevice;
import javax.bluetooth.ServiceRecord;
import javax.bluetooth.UUID;
import javax.microedition.io.Connector;
import javax.microedition.io.StreamConnection;

/**
 * A class that discovers new connections to other Bluetooth
ChatApplications in
 * remote devices.
 */
public class DiscoveryThread extends Thread implements
DiscoveryListener {

      private Object waitMutex = new Object();

      private ConnectionProtocol protocol;

      private Vector foundDevices = new Vector();

      private RemoteDevice currentDevice;

      private boolean serviceSearchInError;

      public DiscoveryThread(ConnectionProtocol protocol) {
            this.protocol = protocol;
      }

      public void run() {
            try {
                  while (true) { // infinite loop to discover new
connections.

                        foundDevices.removeAllElements();

                        DiscoveryAgent agent =
LocalDevice.getLocalDevice()
                                    .getDiscoveryAgent();

                        // This thread starts an inquiry method and
then waits for
                        // a notification from mutex. A Bluetooth
thread will invoke
```

```
                        // the inquiryCompleted method and sends a
notification to
                        // mutex.
                        //
                        // @see http://en.wikipedia.org/wiki/Mutex
                        synchronized (waitMutex) {
                              Log.log("Start inquiry method to found
devices.");

      agent.startInquiry(BluetoothSettings.DISCOVERY_MODE, this);
                              waitMutex.wait();
                        }

                        UUID uuids[] = new UUID[] {
BluetoothSettings.UUID };
                        for (int i = 0; i < foundDevices.size(); i++)
{
                              serviceSearchInError = false;
                              for (int t = 0; t <
BluetoothSettings.MAX_TRY_COUNT_TO_SEARCH_SERVICE; t++) {

                                    // This thread starts a service
search method and then
                                    // waits for a notification from
mutex. A Bluetooth
                                    // thread will invoke the
serviceSearchCompleted method
                                    // and sends a notification to
mutex.
                                    synchronized (waitMutex) {
                                          currentDevice =
(RemoteDevice) foundDevices
                                                .elementAt(i);
                                          Log
                                                .log("Start to
search the Serial Port Profile(SPP) service from "
                                                      +
deviceString(currentDevice));
                                          agent.searchServices(null,
uuids, currentDevice,
                                                this);
                                          waitMutex.wait();
                                    }
                                    if (!serviceSearchInError)
                                          break;
                                    else
                                          // sleep few seconds before
next service search.
                                          Thread.sleep(2 * 1000);
                              }
                        }
                        Thread

      .sleep(BluetoothSettings.SLEEP_TIME_BEFORE_NEW_DISCOVERY *
1000);
                  }
            } catch (Exception e) {
                  Log.log("DiscoveryThread-Exception", e);
            }
      }
```

```
	/*-
	 *
	 *  --- Implement DiscoveryListener ---
	 *
	 * @see javax.bluetooth.DiscoveryListener
	 */

	public void inquiryCompleted(int arg0) {
		synchronized (waitMutex) {
			waitMutex.notify();
		}
	}

	public void deviceDiscovered(RemoteDevice dev, DeviceClass
clazz) {
		Log.log("Device found: " + deviceString(dev));
		if (!protocol.hasConnection(dev.getBluetoothAddress()))
			foundDevices.addElement(dev);
	}

	public void servicesDiscovered(int transId, ServiceRecord[]
records) {
		Log.log("Service discovered.");
		for (int i = 0; i < records.length; i++) {

			if (records[i] == null)
				continue;

			String url = records[i].getConnectionURL(
					BluetoothSettings.SECURITY_OPTIONS,
					BluetoothSettings.MUST_BE_MASTER);
			Log.log("Connecting to url: "+url);
			try {
				StreamConnection stream = (StreamConnection)
Connector
						.open(url);
				stream = new
WorkaroundStreamConnection(stream);
				protocol.handleClientConnection(stream,
currentDevice
						.getBluetoothAddress());
				break;
			} catch (IOException e) {
				Log.log("DiscoveryThread-Exception", e);
			}
		}
	}

	public void serviceSearchCompleted(int transId, int respCode) {
		serviceSearchInError = false;
		String msg = null;
		switch (respCode) {
		case SERVICE_SEARCH_COMPLETED:
			msg = "the service search completed normally";
			break;
		case SERVICE_SEARCH_TERMINATED:
			msg = "the service search request was cancelled by
a call to DiscoveryAgent.cancelServiceSearch()";
			break;
		case SERVICE_SEARCH_ERROR:
```

```
                  msg = "an error occurred while processing the
request";
                  serviceSearchInError = true;
                  break;
            case SERVICE_SEARCH_NO_RECORDS:
                  msg = "no records were found during the service
search";
                  break;
            case SERVICE_SEARCH_DEVICE_NOT_REACHABLE:
                  msg = "the device specified in the search request
could not be reached or the local device could not establish a
connection to the remote device";
                  break;
            }
            Log.log("Service search completed - " + msg);

            synchronized (waitMutex) {
                  waitMutex.notify();
            }
      }

      /*-
       *
       * --- Utility method ---
       */

      private String deviceString(RemoteDevice dev) {
            String ret = null;
            try {
                  if (BluetoothSettings.WORKAROUND_BLUECOVE)
                        ret = "";
                  else
                        ret = dev.getFriendlyName(false);
            } catch (IOException e) {
                  ret = "none";
            }
            ret += " - " + dev.getBluetoothAddress();
            return ret;
      }

}
```

```
/*
 * LcduiUI.java
 *
 * Created on April 11, 2007, 4:39 PM
 */

package ui;

import java.util.Date;
import java.util.Vector;

import javax.microedition.lcdui.Command;
import javax.microedition.lcdui.CommandListener;
import javax.microedition.lcdui.CustomItem;
import javax.microedition.lcdui.Display;
import javax.microedition.lcdui.Displayable;
import javax.microedition.lcdui.Font;
import javax.microedition.lcdui.Form;
import javax.microedition.lcdui.Graphics;
import javax.microedition.lcdui.TextField;
import javax.microedition.midlet.MIDlet;

public class LcduiUI extends Form implements CommandListener,
            ConnectionProtocol.EventHandler {

      /**
       * A custom item that draws last messages to screen.
       */
      class MessageArea extends CustomItem {
            private int negHeight;

            protected MessageArea(String label, int negHeight) {
                  super(label);
                  if (negHeight > LcduiUI.this.getHeight())
                        this.negHeight = LcduiUI.this.getHeight() /
4;
                  else
                        this.negHeight = negHeight;
            }

            protected int getMinContentWidth() {
                  return LcduiUI.this.getWidth();
            }

            protected int getMinContentHeight() {
                  return LcduiUI.this.getHeight() - negHeight;
            }

            protected int getPrefContentWidth(int arg0) {
                  return getMinContentWidth();
            }

            protected int getPrefContentHeight(int arg0) {
                  return getMinContentHeight();
            }

            protected void paint(Graphics g, int w, int h) {

                  g.setColor(0xFFFFA0); // yellow-white
                  g.fillRect(0, 0, w, h);
```

```
                  Font f = Font.getDefaultFont();
                  g.setColor(0); // black
                  g.setFont(f);

                  // paint last messages in screen.
                  synchronized (messages) {
                        int rowsLeft = getMinimumHeight() /
f.getHeight();
                        if ((getMinimumHeight() % f.getHeight()) > 0)
                              rowsLeft--;
                        for (int msgIdx = messages.size() - 1;
rowsLeft >= 0
                                    && msgIdx >= 0; msgIdx--) {
                              rowsLeft -= drawMessage(g, f,
messages.elementAt(msgIdx)
                                          .toString(), rowsLeft);
                        }
                  }

            }

            /**
             * Assumes non empty message.
             *
             * @return used row count.
             */
            private int drawMessage(Graphics g, Font f, String msg,
int rowsLeft) {
                  int maxWidth = LcduiUI.this.getWidth();
                  int begin = 0, end = 0;
                  int rows = getApproximationOfRowCount(f, msg);
                  for (int rowIdx = 0; rowIdx < rows; rowIdx++) {
                        for (; end <= msg.length()
                                    && f.substringWidth(msg, begin,
end - begin) < maxWidth; end++)
                              ;
                        end--;
                        g.drawSubstring(msg, begin, end - begin, 0,
                                    (rowsLeft - rows + rowIdx) *
f.getHeight(),
                                    Graphics.TOP | Graphics.LEFT);
                        begin = end;
                  }
                  return rows;
            }

            private int getApproximationOfRowCount(Font f, String
msg) {
                  int maxWidth = LcduiUI.this.getWidth();
                  int strWidth = f.stringWidth(msg);
                  int rows = strWidth / maxWidth;
                  if ((strWidth % maxWidth) > 0)
                        rows++;
                  return rows;
            }

            public void refresh() {
                  repaint();
            }
      }
```

```
	class Message {
		Date date;

		String userName;

		String msg;

		Message(String userName, String msg) {
			this.date = new Date();
			this.userName = userName;
			this.msg = msg;
		}

		public String toString() {
			if (userName == null)
				return msg;

			StringBuffer buf = new StringBuffer();
			buf.append(userName);
			buf.append(": ");
			buf.append(msg);
			return buf.toString();
		}
	}

	private MIDlet midlet;

	private Vector messages = new Vector();

	private ConnectionProtocol protocol;

	private TextField field;

	private MessageArea messageArea;

	private Command send = new Command("Send message",
Command.SCREEN, 1);

	private Command log = new Command("Log screen", Command.SCREEN,
2);

	public LcduiUI(MIDlet m) {
		super("Bluetooth Bus Info");
		this.midlet = m;

		field = new TextField("Msg:", "Application Started", 128,
TextField.ANY);
		messageArea = new MessageArea("Message Area",
field.getMinimumHeight());

		append(messageArea);
		append(field);

		addCommand(send);
		addCommand(log);

		setCommandListener(this);
	}

	public void chatMessage(String userName, String msg) {
		messages.addElement(new Message(userName, msg));
```

```
            messageArea.refresh();
      }

      public void chatLeave(String userName) {
            messages.addElement(new Message(null, userName + " is
leaving."));
            messageArea.refresh();
      }

      public void chatEnter(String userName) {
            messages.addElement(new Message(null, userName + " has
joined."));
            messageArea.refresh();
      }

      public void setCurrent() {
            Display.getDisplay(midlet).setCurrent(this);
            messageArea.refresh();
      }

      public void setProtocol(ConnectionProtocol protocol) {
            this.protocol = protocol;
      }

      public void commandAction(Command cmd, Displayable d) {
            if (cmd == send) {
                  protocol.broadcastMessage(field.getString());
            } else if (cmd == log) {
                  Log.setCurrent(midlet);
            }

      }

}
```

```
/*
 * Log.java
 *
 * Created on April 11, 2007, 3:58 PM
 *
 * To change this template, choose Tools | Template Manager
 * and open the template in the editor.
 */

package ui;

import java.util.Vector;

import javax.microedition.lcdui.Command;
import javax.microedition.lcdui.CommandListener;
import javax.microedition.lcdui.Display;
import javax.microedition.lcdui.Displayable;
import javax.microedition.lcdui.Form;
import javax.microedition.lcdui.StringItem;
import javax.microedition.midlet.MIDlet;

public class Log extends Form implements CommandListener {

      static private Log instance;

      static Vector logs = new Vector();

      private MIDlet midlet;

      private Displayable previous;

      private Command back = new Command("Back", Command.BACK, 1);

      private Command next = new Command("Next Page", Command.ITEM,
2);

      private Command prev = new Command("Prev. Page", Command.ITEM,
2);

      private int currentPage = 0;

      static private int LINES_IN_PAGE = 5;

      // private Command refresh = new Command("Refresh",
Command.SCREEN, 1);

      private Log() {
            super("Log");
            addCommand(back);
            // addCommand(refresh);
            addCommand(prev);
            addCommand(next);
            setCommandListener(this);
            instance = this;
      }

      public static void setCurrent(MIDlet midlet) {
            if (instance == null)
                  instance = new Log();

            instance.midlet = midlet;
```

```
            instance.previous =
Display.getDisplay(midlet).getCurrent();

            Display.getDisplay(midlet).setCurrent(instance);

            instance.refresh();
      }

      public void commandAction(Command cmd, Displayable d) {
            if (cmd == back) {
                  Display.getDisplay(midlet).setCurrent(previous);
            } else if (cmd == prev) {
                  if (currentPage > 0)
                        currentPage--;
                  refresh();
            } else if (cmd == next) {
                  if (currentPage < getPages())
                        currentPage++;
                  refresh();
            }
      }

      private int getPages() {
            int pages = logs.size() / LINES_IN_PAGE;
            return pages;
      }

      private void refresh() {
            synchronized (logs) {
                  if (size() == 0)
                        append(new StringItem("", ""));

                  if (!(get(0) instanceof StringItem))
                        return;

                  StringItem str = (StringItem) get(0);

                  StringBuffer sb = new StringBuffer();
                  for (int i = currentPage * LINES_IN_PAGE; i <
(currentPage + 1)
                              * LINES_IN_PAGE
                              && i < logs.size(); i++) {
                        sb.append(logs.elementAt(i).toString());
                        sb.append("\n");
                  }

                  sb.insert(0, "Page (" + currentPage + "/" +
getPages() + "):\n");
                  str.setText(sb.toString());
            }
      }

      static public void log(String string) {
            logs.addElement(string);
            System.out.println(string);

            if (instance != null)
                  instance.refresh();
      }
```

```
	static public void log(String location, Exception e) {
		log(location);
		log(e.getMessage());
		e.printStackTrace();
	}

	static public void clear() {
		logs.removeAllElements();
	}

}
```

```
/*
 * ServerThread.java
 *
 * Created on April 11, 2007, 3:57 PM
 *
 * To change this template, choose Tools | Template Manager
 * and open the template in the editor.
 */

package ui;

import javax.bluetooth.LocalDevice;
import javax.microedition.io.Connector;
import javax.microedition.io.StreamConnectionNotifier;

/**
 * A class that acts as a server to wait for new Bluetooth
connections.
 */
public class ServerThread extends Thread {

      private ConnectionProtocol protocol;

      public ServerThread(ConnectionProtocol protocol) {
            this.protocol = protocol;
      }

      public void run() {
            try {
                  LocalDevice device = LocalDevice.getLocalDevice();

      device.setDiscoverable(BluetoothSettings.DISCOVERY_MODE);
                  String url = "btspp://localhost:" +
BluetoothSettings.UUID
                              + ";name=ChatApplication";

                  Log.log("Create server by uri: " + url);
                  StreamConnectionNotifier notifier =
(StreamConnectionNotifier) Connector
                              .open(url);

                  while (true) { // infinite loop to accept
connections.
                        Log.log("Waiting for connection...");
                        protocol.handleServerConnection(new
WorkaroundStreamConnection(notifier.acceptAndOpen()));
                  }
            } catch (Exception e) {
                  Log.log("ServerThread-Exception", e);
            }
      }

}
```

```
/*
```

```
 * WorkaroundStreamConnection.java
 *
 * Created on April 11, 2007, 3:56 PM
 *
 * To change this template, choose Tools | Template Manager
 * and open the template in the editor.
 */

package ui;

import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;

import javax.microedition.io.StreamConnection;

/**
 * A workaround class for S60 3rd edition and S80 devices. This class
will slow
 * down the stream.
 *
 */
public class WorkaroundStreamConnection extends Object implements
            StreamConnection {

      private static final int SLOW_DOWN = 150; // ms

      private StreamConnection conn;

      public WorkaroundStreamConnection(StreamConnection stream) {
            this.conn = stream;
      }

      public InputStream openInputStream() throws IOException {
            return new InputStream() {
                  InputStream in = conn.openInputStream();

                  public synchronized int read() throws IOException {
                        int ret = -1;
                        while (true) {
                              ret = in.read();
                              if (ret < 0) {
                                    try {
                                          Thread.sleep(SLOW_DOWN);
                                    } catch (InterruptedException e)
{
                                    }
                                    continue;
                              }
                              break;
                        }
                        try {
                              Thread.sleep(SLOW_DOWN);
                        } catch (InterruptedException e) {
                        }
                        return ret;
                  }
            };
      }
```

```
	public DataInputStream openDataInputStream() throws IOException
{
		return new DataInputStream(openInputStream());
	}

	public void close() throws IOException {
		conn.close();
	}

	public OutputStream openOutputStream() throws IOException {
		return new OutputStream() {
			OutputStream out = conn.openOutputStream();

			public synchronized void write(int value) throws
IOException {
				out.write(value);
				out.flush();
				try {
					Thread.sleep(SLOW_DOWN);
				} catch (InterruptedException e) {
				}
			}

			public void flush() throws IOException {
			}
		};
	}

	public DataOutputStream openDataOutputStream() throws
IOException {
		return new DataOutputStream(openOutputStream());
	}

}
```

APPENDIX V

STEP TO DEVELOP MIDP PROGRAM IN NETBEANS IDE 5.5

Creating a MIDP Application Using the Visual Mobile Designer

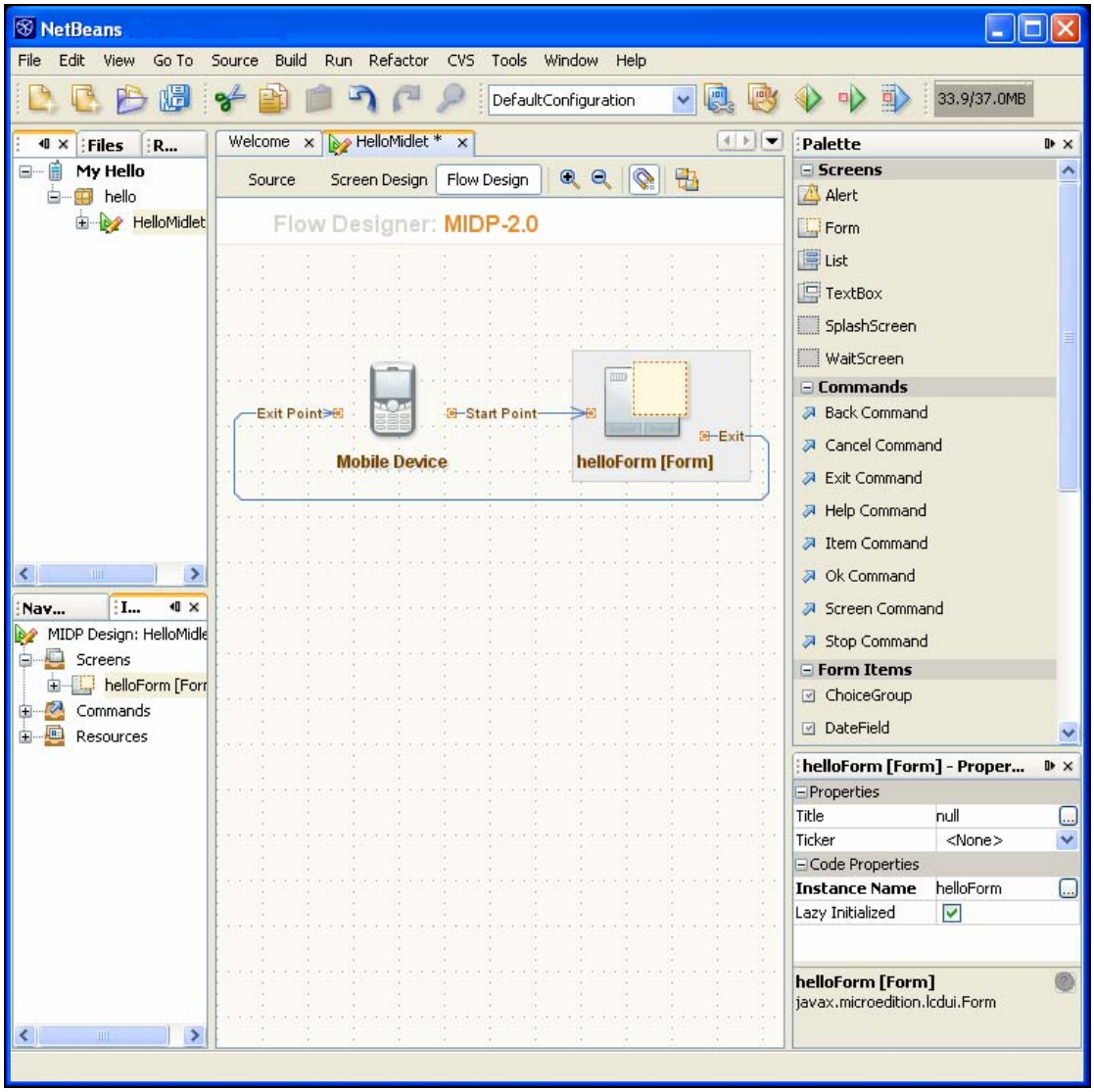
The NetBeans IDE provides a wizard that enables you to quickly create a MIDP project. When creating the project, you can choose to develop your application in the Visual Mobile Designer (VMD) or in the Source Code Editor.

Using the Visual Mobile Designer gives you the ability to graphically plan out the flow of the application and design the screens the application will use. The designer automatically creates the code for the application.

Creating a MIDP/CLDC Application

1. Choose File > New Project (Ctrl-Shift-N). Under Categories, select Mobile. Under Projects, select Mobile Application and click Next.
2. Enter `MyHello` in the Project Name field. Change the Project Location to any directory on your system. From now on, we will refer to this directory as `$PROJECTHOME`.
3. Check the Set as Main Project and Create Hello MIDlet checkboxes (both are checked by default). Click Next.
4. Leave the J2ME Wireless Toolkit as the selected Target Platform.
5. Click Finish. The IDE creates the `$PROJECTHOME/MyHello` project folder. The project folder contains all of your sources and project metadata, such as the project Ant script.

   The application itself is displayed in the Flow Design window of the Visual Mobile Designer.



Editing the Java Source Code

Now let's edit the text displayed by the MIDlet.

1. Click on Screen Design.

   This opens the Screen Designer window, and displays the `helloForm[Form]` screen, which is the only screen available in the application.
2. Double-click on the "Hello world!" text and type in some new text.

Compiling and Running the Project

1. Choose Run > Run Main Project (F6) from the Run menu. Double-click the Output window to maximize it so you can see all the output. Note that the `HelloMIDlet.java` file is built before it is executed. A device emulator opens to display the results of the executed MIDlet. The default device emulator is `DefaultColorPhone`.
2. In the device emulator window, click on the button below the Launch command. The device emulator launches the MIDlet and displays the text you entered in the source code.
3. Click on the button below Exit to close the MIDlet. Then click on the button in the upper right corner of the device to close the emulator window.