

refreshing a window

Source: <http://coding.derkeiler.com/Archive/Java/comp.lang.java.programmer/2006-03/msg00677.html>

- *From:* abenm613@xxxxxxx
 - *Date:* 4 Mar 2006 19:18:23 -0800
-

I am doing a program that is supposed to do the following. As a query is being processed, the results are to be displayed in a JTable at the bottom. I have already made sure the queries run okay. However, after the query is processed, the results do not come up at the bottom until I minimize (or maximize) the window. I want to know how to make the window update itself automatically without waiting for the user to minimize/maximize it. If you need some code to answer my question, here is the listByReqeust class that does the querying:

```
import java.awt.*;
import java.awt.event.*;
import java.sql.*;
import javax.swing.*;
import javax.swing.event.*;
import javax.swing.text.JTextComponent;
import java.io.*;
import java.util.*;
import javax.swing.table.*;
import java.text.*;
import java.awt.print.*;
import java.lang.reflect.*;
//import java.pension.util.*;

public class listByRequest extends JFrame implements ActionListener{
private ScrollingPanel fields;
private JTextField output, input, input1, input2, input3;
private Connection connection;
private String searchKey, searchStart, query, listQuery, whatIsInput,
social;
private JList resultList;
private JPanel listPanel, topPanel, mainPanel;
ResultSet resultSet;
JTable table[] = new JTable[100];
JPanel bottomPanel;
boolean flag[] = new boolean[20];
int c = 0, contCount = 0;
private JLabel lname, fname, gnumber, pcode;
private JButton close, print;
```

refreshing a window

```
Container[] cont = new Container[100];

public listByRequest(Connection c, ScrollingPanel f, JPanel bp, JTable
listTable)
{

fields = f;
connection = c;
bottomPanel = bp;
table[contCount] = listTable;

}

public void actionPerformed(ActionEvent e)
{
lname = new JLabel();
fname = new JLabel();
gnumber = new JLabel();
pcode = new JLabel();
topPanel = new JPanel();

try{
Statement statement = connection.createStatement();

query = "SELECT ssec, lastname, firstname, streetaddress, city,
state, zipcode, groupnumber FROM students WHERE";

System.out.println("I'm about to start constructing query");

if(!fields.lastName.getText().equals("")){
query = query+" lastname = '"+fields.lastName.getText()+"'";
flag[0] = true;
if((!fields.firstName.getText().equals("")) ||
(!fields.groupNumber.getText().equals("")) ||
(!fields.programCode.getText().equals("")))
query = query+" AND";
//System.out.println(query);
lname.setText("Last name = "+fields.lastName.getText());
topPanel.add(lname);

}
if(!fields.firstName.getText().equals("")){
flag[1]=true;
System.out.println("The first name appears as
"+fields.firstName.getText());
query = query+" firstname = '"+fields.firstName.getText()+"'";;
if((!fields.groupNumber.getText().equals("")) ||
(!fields.programCode.getText().equals("")))

```

refreshing a window

refreshing a window

```
query = query+" AND";
fname.setText("First name = "+fields.firstName.getText());
topPanel.add(fname);
}
if(!fields.groupNumber.getText().equals("")){
flag[3]=true;
System.out.println("The group number is
"+fields.groupNumber.getText());
query = query+" groupnumber =
"+fields.groupNumber.getText()+"";
if(!fields.programCode.getText().equals(""))
query = query+" AND";
gnumber.setText("Group number =
"+fields.groupNumber.getText());
topPanel.add(gnumber);
}
if(!fields.programCode.getText().equals("")){
flag[4]=true;
System.out.println("The program code is
"+fields.programCode.getText());
query = query+" programcode = "+fields.programCode.getText()+"";
pcode.setText("Program code = "+fields.programCode.getText());
topPanel.add(pcode);
}
System.out.println(query);
if(query == "SELECT * FROM students WHERE"){
JOptionPane.showMessageDialog(null, "Invalid query", "Invalid
query", JOptionPane.ERROR_MESSAGE);
System.exit(1);
}
System.out.println("\nSending query: "+query+"\n");

//table = new JTable(12, 12);

//Container cont = getContentPane();

//cont.add(table);

resultSet = statement.executeQuery(query);
System.out.println("\nResults found...");

showList(resultSet);

System.out.println("\nQuery successful");
resultSet.close();

//setSize(1050, 1050);

}
```

refreshing a window

refreshing a window

```
catch(SQLException sqllex){
System.out.println("Something wrong with SQL finding the record");
sqllex.printStackTrace();
System.out.println(sqllex.toString());
}
}
```

```
public void showList(ResultSet rs){
contCount++;
cont[contCount] = getContentPane();
listPanel = new JPanel();
Vector columnHeads = new Vector();
Vector rows = new Vector();
cont[contCount].setLayout(new BorderLayout());
//frame = new JFrame();

try{
boolean moreRecords = rs.next();

bottomPanel.remove(table[contCount-1]);

//listPanel = new JPanel();

if(!moreRecords){
JOptionPane.showMessageDialog(this, "ResultSet contained no
records");
setTitle("No records to display");
return;
}
boolean nextExists;
ResultSetMetaData rsmd = rs.getMetaData();

for(int i=1; i<=rsmd.getColumnCount(); ++i){
columnHeads.addElement(rsmd.getColumnName(i));
//columnHeads.elementAt(i).setWidth(5);
}

do{
rows.addElement(getNextRow(rs, rsmd));
nextExists = rs.next();
//if(nextExists)
// System.out.println("Next exists");
//else
// System.out.println("No next");

}while(nextExists);
table[contCount] = new JTable(rows, columnHeads);
ExcelAdapter ea = new ExcelAdapter(table[contCount]);
```

refreshing a window

```
bottomPanel.add(table[contCount]);

validate();

}

catch(SQLException sqlEx){
System.out.println("SQL Exception");
sqlEx.printStackTrace();
}

catch(Exception ex){
ex.printStackTrace();
System.out.println(ex.toString());
}

}

public Vector getNextRow(ResultSet rs, ResultSetMetaData rsmd) throws
SQLException{
Vector currentRow = new Vector();
for (int i=1; i<=rsmd.getColumnCount(); ++i){
switch(rsmd.getColumnType(i)){
case Types.VARCHAR:
case Types.LONGVARCHAR:
currentRow.addElement(rs.getString(i));
break;
case Types.INTEGER:
currentRow.addElement(new Long(rs.getLong(i)));
break;
default:
//System.out.println("Type was: "+rsmd.getColumnTypeName(i));
}
}
return currentRow;
}

}
```