

ReaderWindow

```
import java.awt.Button;
import java.awt.DisplayMode;
import java.awt.Frame;
import java.awt.GraphicsConfiguration;
import java.awt.GraphicsDevice;
import java.awt.TextArea;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.WindowEvent;
import java.awt.event.WindowListener;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import javax.comm.CommPortIdentifier;
//erweitert thread, somit können wir in der klasse einen neuen thread starten
public class ReaderWindow extends Thread{
//variablendeklarationen
private static final long serialVersionUID = 1L;
private static Button b1;
private static TextArea ta;
private static Frame frame;
private static CommPortIdentifier cid=null;
private static Thread readThread;
public ReaderWindow (CommPortIdentifier ci)
{
//den identifier global zur verfügung stellen
cid=ci;
/*
 * Fensterschnulli */
frame= new Frame("PortReader");
//was passiert wenn das Fenster irgendwie geschlossen wird
frameaddWindowListener(new WindowListener(){
public void windowClosing(WindowEvent e)
{
frame.dispose();
}
try{
SimpleReaderSerialPort().close();
}catch(Exception ex){}
System.exit(0);
}

@Override public void windowActivated(WindowEvent arg0) {}
@Override public void windowClosed(WindowEvent arg0) {}
@Override public void windowDeactivated(WindowEvent arg0) {}
@Override public void windowDeiconified(WindowEvent arg0) {}
@Override public void windowIconified(WindowEvent arg0) {}
```

```

@Override public void windowOpened(WindowEvent arg0) {}
});
framesetLayout(null); // "Null-Layout" setzen
ta= new TextArea();
t.setBounds(0, 0, 299, 120); // x, y, breite, höhe
b1= new Button();
b1.setBounds(110,125,80,20);
b1.setLabel("schliessen");
b1.setName("Fenster schliessen");
b1.addActionListener(buttonListener);
frame.add(ta);
frame.add(b1);
DisplayMode dMode= null;
GraphicsConfiguration gc = frame.getGraphicsConfiguration();
GraphicsDevice gd = gc.getDevice();
dMode= gd.getDisplayMode();
int width = dMode.getWidth()/2-150;
int height = dMode.getHeight()/2-77;
framesetLocation(width,height);
framesetSize(300,150);
framesetResizable(false);
framesetUndecorated(true);
framesetVisible(true);
/*
 * ENDE Fensterschnulli */
//hier wird der thread gestartet (unten bei run() gehts weiter :))
readThread= new Thread(this);
readThread.start();
}
//was passiert wenn wir "schließen" drücken
ActionListener buttonListener = new ActionListener() {
public void actionPerformed(ActionEvent event) {
//fenster zerstören
frame.dispose();
try{
//versuchen den port zu schließen
SimpleRegdtSerialPort().close();
}catch(Exception ex){}
System.exit(0);
}
};
//wir brauchen einen neuen Thread, sonst reagiert nix mehr. hier wird er gestartet
public void run(){
//Zeiterzeugung
Calendar cal = Calendar.getInstance();
SimpleDateFormat sdf=new SimpleDateFormat("HH:mm:ss\n");
ReaderWindow.getTa().append("Programmstart um: "+sdf.format(cal.getTime()));
//ENDE Zeiterzeugung

```

```

try {
//versuchen den reader zu starten
new SimpleRead(cid);
} catch (Exception e) {
    printStackTrace();
}
}
/*
 * GETTERS AND SETTERS
 * also wir geben anderen klassen die möglichkeit objekte in der klasse zu verä
 */
public static TextArea getTa() {
return ta;
}
public static void setTa(TextArea ta) {
    ReaderWindow.ta = ta;
}
}

```

Wird von [ComPortTester](#) aufgerufen.

Danach wird [PortReader](#) zum Testen des Ports aufgerufen.