

# Android-Apps

## Lösen einer quadratischen Gleichung

### I/O in einer Activity

# quadrGleichung (I)

I  
N  
F  
O  
R  
M  
A  
T  
I  
K

Java - quadrGleichung-1/res/layout/main.xml - Eclipse Platform

File Edit Run Navigate Search Project Refactor Window Help

Package Explorer: amando, BigBrother-Master, BigBrother-Slave, canvasBitmap, canvasDemo, Gleichungsloeser, Graphen, Guthabenabfrage, hangman, Hello Android, Kugel Final, LGS\_22, Maps\_show\_11, Maps\_show\_12, Maps\_show\_13, MatheApp\_1

quadr\_g1.java strings.xml main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:orientation="vertical"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent">

  <TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/txt_pqFormel_intro" />

  <LinearLayout
    android:orientation="horizontal"
    android:layout_height="wrap_content"
    android:layout_width="wrap_content"
    android:layout_gravity="center_horizontal">
    <EditText android:id="@+id/edt_a2"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:inputType="numberSigned|numberDecimal"
    />
    <TextView
      android:layout_width="fill_parent"
      android:layout_height="wrap_content"
      android:text=" x^2 + "
    />
    <EditText android:id="@+id/edt_a1"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
    />
  </LinearLayout>
</LinearLayout>
```

alles untereinander angeordnet

alles nebeneinander angeordnet

alles wird zentriert

nur Dezimalzahl mit Vorzeichen

Die View zur Texteingabe, Abfrage über die id R.id.edt\_a2

Neues Projekt anlegen

quadrGleichung-1

- src
- gen [Generated Java Files]
- Android 2.1-update1
- assets
- res
  - drawable-hdpi
  - drawable-ldpi
  - drawable-mdpi
  - layout

# quadrGleichung (I)

```

        android:layout_height="wrap_content"
        android:text=" = 0"
    />
</LinearLayout>

<Button
    android:id="@+id/pqf_btn_loesen"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/txt_btn_loesen"
    android:onClick="onClickLoesen"
/>

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/txt_ergebnis_intro" />

```

strings.xml nicht vergessen!

so heißt die vom Buttonclick aufgerufene Procedure

```

<TableLayout
    android:layout_height="wrap_content"
    android:layout_width="wrap_content"
    android:layout_gravity="center_horizontal">
    <TableRow>
        <TextView android:id="@+id/txt_xlinks1"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
        />
        <TextView android:id="@+id/txt_xrechts1"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
        />
    </TableRow>
    <TableRow>
        <TextView android:id="@+id/txt_xlinks2"

```

zwei Lösungen möglich

```

        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
    />
</TableRow>
</TableLayout>

<TextView android:id="@+id/pqf_txt_info"
    android:textColor="#FF0000"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"/>
</LinearLayout>

```

Textfarbe(rgb in Hex)

so breit wie die Eltern-View, so hoch wie der Inhalt (flexibel)

Table mit mehreren Zeilen

Die View zur Textausgabe, Ansteuerung über die id `R.id.txt_xrechts1`

# quadrGleichung (I)

```
quadr_gl.java x strings.xml main.xml
package eh.beispiele.pqf;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;

public class quadr_gl extends Activity {

    // Die drei Parameter
    private float a2,a1,a0;

    // Die beiden Unbekannten
    private double x1,x2;

    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

Muss nicht eingetippt werden.  
Wenn die zu importierenden Klassen nicht  
automatisch erscheinen, dann  
Strg + Shift + „O“ drücken!

# quadrGleichung (I)

```

/* Testet, ob eine Zahl eingegeben wurde und wandelt den String in float um.
 * In der XML-Datei ist bereits angegeben, dass nur Zahlen, 1 Komma(also der Punkt)
 * und das Minuszeichen eingegeben werden dürfen. */
private static float secure(String test){
    float ok = 0.0f; ohne „f“ kapiert java häufig nicht, dass es sich um float handelt!
    if ((test.length() != 0) && (test != ".")){
        ok = Float.parseFloat(test); Umwandlung von String zu float
    }
    return ok;
}

```

```

/* Einlesen der Daten und Übergabe an die Ergebnis-Activity */
public void onClickLoesen(final View sfNormal) {
    /* Die Eingabefelder werden über ihre id gefunden und der Inhalt
     * der EditText den Fließkommazahl-Variablen zugewiesen. */
    final EditText txt_a2 = (EditText) findViewById(R.id.edt_a2); finden und casten
    a2 = secure(txt_a2.getText().toString()); Den Text der View in einen String umwandeln
    final EditText txt_a1 = (EditText) findViewById(R.id.edt_a1);
    a1 = secure(txt_a1.getText().toString());
    final EditText txt_a0 = (EditText) findViewById(R.id.edt_a0);
    a0 = secure(txt_a0.getText().toString());

    /* Ergebnis wird berechnet, in msg wird ggf. Fehlermeldung geliefert */
    String msg = berechne();
}

```

Berechnung folgt

# quadrGleichung (I)

```

/* Testet, ob eine Zahl eingegeben wurde und wandelt den String in float um.
 * In der XML-Datei ist bereits angegeben, dass nur Zahlen, 1 Komma(also der Punkt)
 * und das Minuszeichen eingegeben werden dürfen. */
private static float secure(String test){
    float ok = 0.0f;
    if ((test.length() != 0) && (test != "-")){
        ok = Float.
    }
    return ok;
}

```

Die Hilfe erscheint nach dem Eintippen des Punktes, bzw. wenn die Maus über dem gewünschten Inhalt positioniert wird.

```

/* Einlesen der Daten
public void onClick() {
    /* Die Eingabefelder
    * der EditTexts
    final EditText
    a2 = secure(txt2);
    final EditText
    a1 = secure(txt1);
    final EditText
    a0 = secure(txt0);

    /* Ergebnis wird
    String msg = be

    // Views über i
    final TextView
    final TextView

```

MAX\_VALUE float - Float  
 MIN\_VALUE float - Float  
 NaN float - Float  
 NEGATIVE\_INFINITY float - Float  
 POSITIVE\_INFINITY float - Float  
 intBitsToFloat(int bits) float - Float  
 parseFloat(String string) float - Float  
 SIZE int - Float  
 compare(float float1, float float2) int - Float  
 floatToIntBits(float value) int - Float  
 floatToRawIntBits(float value) int - Float  
 class Class<java/lang/Float>  
 TYPE Class<java.lang.Float> - Float  
 isInfinite(float f) boolean - Float  
 isNaN(float f) boolean - Float  
 toHexString(float f) String - Float  
 toString(float f) String - Float

Press 'Ctrl+Space' to show Template Proposals

**public static float parseFloat (String string)**

Since: [API Level 1](#)

Parses the specified string as a float value.

#### Parameters

**string** the string representation of a float value.

#### Returns

- the primitive float value represented by string.

#### Throws

[NumberFormatException](#) if string is null, has a length zero or can not be parsed as a value.

#### See Also

- [valueOf\(String\)](#)

# quadrGleichung (I)

```
/* berechnet die beiden Unbekannten */
private String berechne() {
    /* Annahme: wird Fehlerfrei sein. Annahme: es gibt zwei Lösungen */
    String msg = "2";
    /* sind genügend Parameter angegeben? */
    if (a2 == 0) {
        if (a1 == 0) {
            msg = "Bitte Werte eingeben!";
        }
        else { // Lineare Gl. mit einer Lösung
            x1 = -a0 / a1;
            msg = "1";
        }
    }
    else { // Umwandeln in p-q-Gl.
        double p = a1/a2;
        double q = a0/a2;
        double disk = p/2*p/2-q;
        if (disk > 0) {
            x1 = -p/2-Math.sqrt(disk);
            x2 = -p/2+Math.sqrt(disk);
        }
        else {
            if (disk == 0) {
                x1 = -p/2;
                msg = "1";
            }
            else {
                msg = "keine Lösung";
            }
        }
    }
}
```

Vergleichsoperator in java

Die Bedingung *immer* in Klammern setzen

Zuweisung wie Vergleich in Delphi

# quadrGleichung (I)

finden und casten

```
// Views über id finden
final TextView txt_ergx1_links = (TextView) findViewById(R.id.txt_xlinks1);
final TextView txt_ergx1_rechts = (TextView) findViewById(R.id.txt_xrechts1);
final TextView txt_ergx2_links = (TextView) findViewById(R.id.txt_xlinks2);
final TextView txt_ergx2_rechts = (TextView) findViewById(R.id.txt_xrechts2);
final TextView txt_ergx = (TextView) findViewById(R.id.pqf_txt_info);
/* falls kein Fehler gemeldet wird, wird TextView per id gesucht und
 * das Ergebnis formatiert geschrieben. */
if (msg == "2") {
    txt_ergx1_links.setText("x1 = ");
    txt_ergx1_rechts.setText(String.format("%.3f%n", x1));
    txt_ergx2_links.setText("x2 = ");
    txt_ergx2_rechts.setText(String.format("%.3f%n", x2));
    txt_ergx.setText(""); // keine Fehlermeldung
}
else {
    if (msg == "1") {
        txt_ergx1_links.setText("x = ");
        txt_ergx1_rechts.setText(String.format("%.3f%n", x1));
        txt_ergx2_links.setText(""); // nur eine Lösung
        txt_ergx2_rechts.setText("");
        txt_ergx.setText(""); // keine Fehlermeldung
    }
    else {
        /* falls Fehler, Ausgabe der Fehlermeldung */
        txt_ergx1_links.setText("");
        txt_ergx1_rechts.setText("");
        txt_ergx2_links.setText(""); // keine Lösung
        txt_ergx2_rechts.setText("");
        txt_ergx.setText(msg);
    }
}
}
```

String der View zuweisen

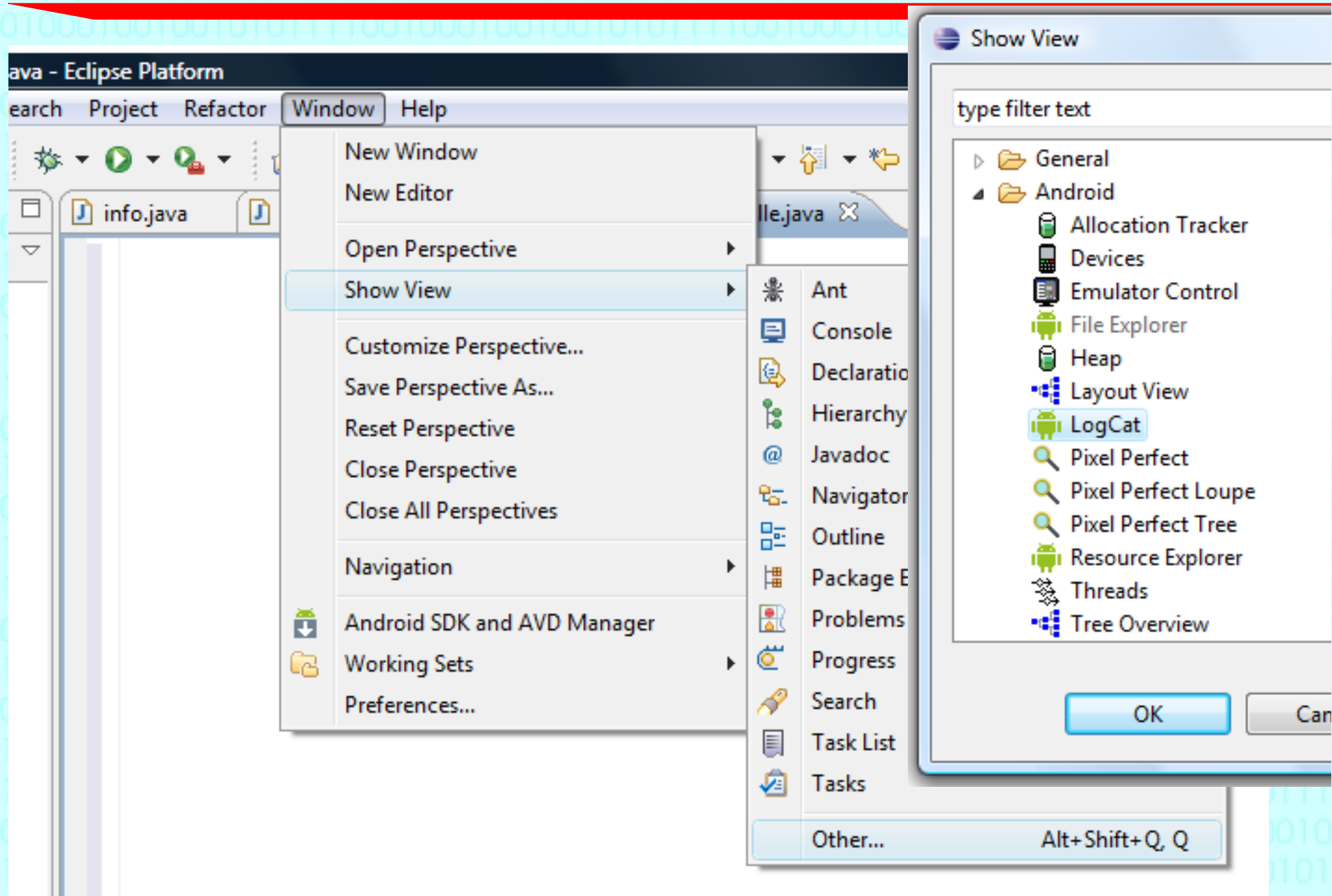
Zahl mit drei  
Nachkommastellen  
Formatieren  
(und Zeilenabschluss)



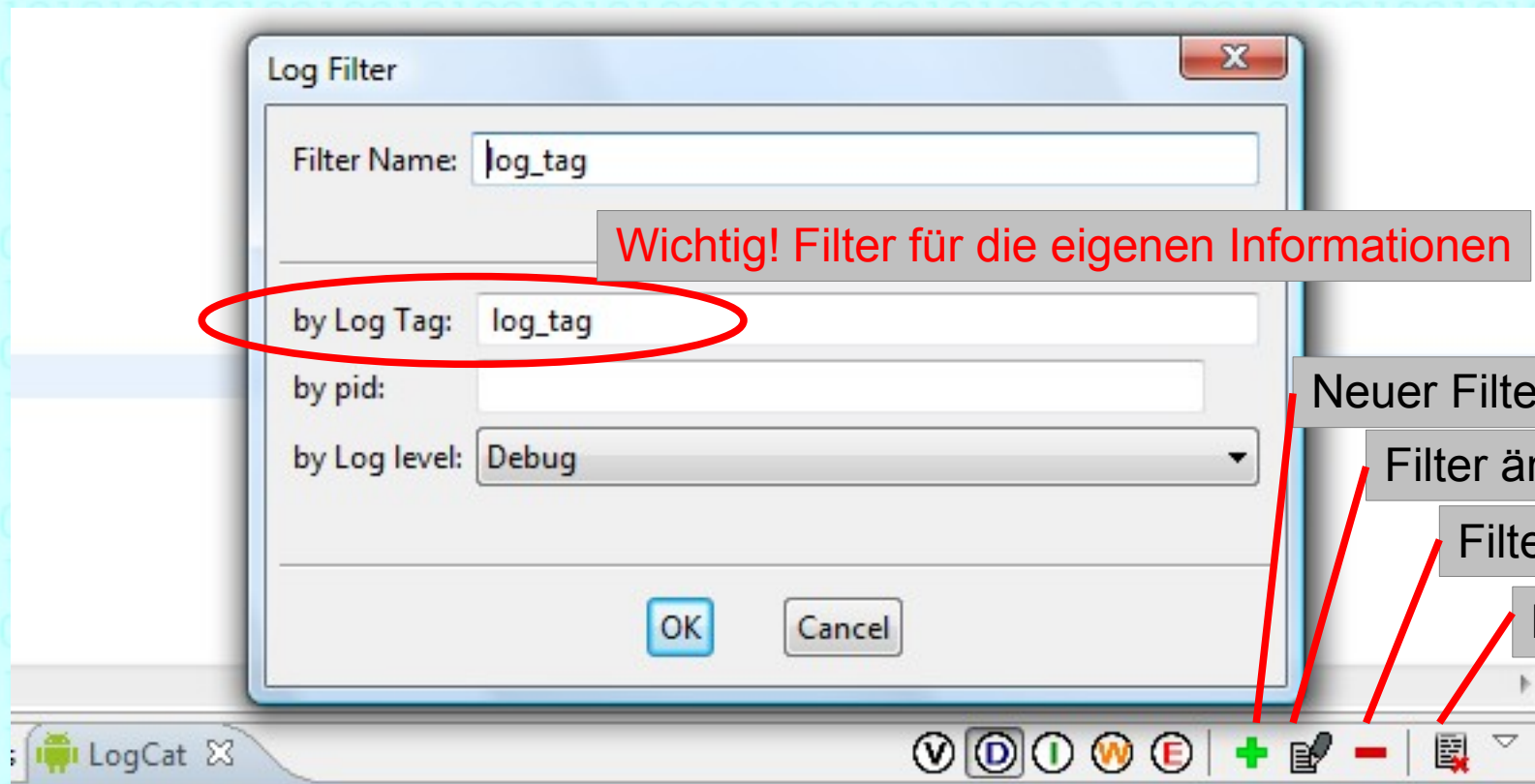
# Android-Apps

## Debuggen mit der LogCat

# Debuggen mit der LogCat



# Debuggen mit der LogCat



Wichtigkeit der Meldung / Schweregrad des Fehlers

V – verbose

D – debug

I – info

W – warn

E – error

Teilmengen

# Debuggen mit der LogCat

```
else { // Umwandeln in p-q-GL.
```

```
double p = a1/a2;
```

```
double q = a0/a2;
```

```
double disk = p/2*p/2-q;
```

```
Log.d("log tag", "disk: "+String.valueOf(disk));
```

```
if (disk > 0) {
```

```
    x1 = -p/2-Math.sqrt(disk);
```

```
    x2 = -p/2+Math.sqrt(disk);
```

```
}
```

```
else {
```

```
import android.util.Log;
```

Problems Search Console File Explorer Progress LogCat X

Log	DebugMeldungen	log_tag			
Time		pid	tag	Message	
04-23 14:17...	I	29	logwrapper	/system/bin/tc terminated by exit(255)	
04-23 14:17...	E	136	logwrapper	executing /system/bin/tc failed: No such file or directory	
04-23 14:17...	I	29	logwrapper	/system/bin/tc terminated by exit(255)	
04-23 14:17...	E	137	logwrapper	executing /system/bin/tc failed: No such file or directory	
04-23 14:17...	I	29	logwrapper	/system/bin/tc terminated by exit(255)	
04-23 14:17...	W	57	ActivityManager	Unable to start service Intent { act=@0 }: not found	
04-23 14:17...	W	57	ActivityManager	Unable to start service Intent { act=@0 }: not found	
04-23 14:17...	D	37	qemud	fdhandler_accept_event: accepting on fd 10	
04-23 14:17...	D	37	qemud	created client 0xc088 listening on fd 14	
04-23 14:17...	I	114	ActivityThread	Publishing provider com.android.launcher2.settings: com.android.lau	
04-23 14:17...	I	110	ActivityThread	Publishing provider icc: com.android.phone.IccProvider	
04-23 14:17...	D	106	dalvikvm	No JNI_OnLoad found in /system/lib/libwvndict.so 0x43e37588, skipping	
04-23 14:17...	I	110	ActivityThread	Publishing provider mms-sms: com.android.providers.telephony.MmsSms	
04-23 14:17...	D	37	qemud	client_fd_receive: attempting registration for service 'gps'	
04-23 14:17...	D	37	qemud	client_fd_receive: -> received channel id 7	

# Debuggen mit der LogCat

```

else { // Umwandeln in p-q-Gl.
    double p = a1/a2;
    double q = a0/a2;
    double disk = p/2*p/2-q;
    Log.d("log_tag", "disk: "+String.valueOf(disk));
    if (disk > 0) {
        x1 = -p/2-Math.sqrt(disk);
        x2 = -p/2+Math.sqrt(disk);
    }
    else {
        if (disk == 0) {
            x1 = -p/2;
            msg = "1";
        }
        else {
            msg = "keine Lösung";
            Log.d("log_tag", "msg "+msg);
        }
    }
}
}

```

Problems Search Console File Explorer Progress LogCat

Time	pid	tag	Message
04-23 14:23...	D 298	<u>log_tag</u>	disk: -1.4375
04-23 14:23...	D 298	log_tag	msg keine Lösung

# Android-Apps

## mehrere Activities und Parameterübergabe per Intent

# QuadrGleichung (II)

I  
N  
F  
O  
R  
M  
A  
T  
I  
K

Java - quadrGleichung-2/src/eh/beispiele/quadrgl/quadr\_gl\_berechne.java - Eclipse Platform

File Edit Refactor Run Source Navigate Search Project Window Help

Package Ex Navigator

quadrGleichung-2 Manifest quadr\_gl\_berechne.java quadr\_gl\_eingabe.java quadr\_gl\_

```

package eh.beispiele.quadrgl;

/* Diese Klasse ist nur für die Berechnungen zuständig */
public class quadr_gl_berechne {

    // Die drei Parameter
    public float a2,a1,a0;

    // Die beiden Unbekannten
    public double x1,x2;

    /* berechnet die beiden Unbekannten */
    public String berechne() {
        /* Annahme: wird Fehlerfrei sein. Annahme: es gibt zwei Lösungen */
        String msg = "2";
        /* sind genügend Parameter angegeben? */
        if (a2 == 0) {
            if (a1 == 0) {
                msg = "Bitte Werte eingeben!";
            }
            else { // Lineare Gl. mit einer Lösung
                x1 = -a0 / a1;
                msg = "1";
            }
        }
        else { // Umwandeln in p-q-Gl.
            double p = a1/a2;
            double q = a0/a2;
            double disk = p/2*p/2-q;

```

amando  
BigBrother-Master  
BigBrother-Slave  
canvasBitmap  
canvasDemo  
Gleichungsloeser  
Guthabenabfrage  
hangman  
Pythagoras  
quadrGleichung-1  
quadrGleichung-2  
src  
eh.beispiele.quadrgl  
quadr\_gl\_berechne.java  
quadr\_gl\_eingabe.java  
gen [Generated Java Files]  
Android 2.1-update1

Neues Projekt anlegen

Eigene Klasse nur für die Berechnungen hinzufügen

Dazu Kontextmenü des Packages mit der rechten Maustaste aufrufen

# quadrGleichung (II)

I  
N  
F  
O  
R  
M  
A  
T  
I  
K

```
quadrGleichung-2 Manifest | quadr_gl_berechne.java | quadr_gl_eingabe_layout.xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >

    <TextView
        style="@style/TextStyleUeberschrift"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="@string/txt_quadr_gl_eingabe_intro" />

    <TableLayout
        android:layout_height="wrap_content"
        android:layout_width="wrap_content"
        android:layout_gravity="center_horizontal">
        <TableRow>
            <EditText android:id="@+id/edt_a2"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:inputType="numberSigned|numberDecimal"
            />
            <TextView
                android:layout_width="fill_parent"
                android:layout_height="wrap_content"
                android:text=" x^2 + "
            />
            <EditText android:id="@+id/edt_a1"
```

Eigenes Layout für die Eingabe,  
main.xml umbenennen (F2)

Style-Definitionen auslagern



# quadrGleichung (II)

```

quadrGleichung-2 Manifest | strings.xml | styles.xml X | colors.xml
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <style name="MathTheme">
    <item name="android:windowBackground">@color/hintergrund</item>
    <item name="android:textViewStyle">@style/TextStyleNormal</ite
    <item name="android:buttonStyle">@style/Schaltflaeche</item>
  </style>

  <style name="TextStyleNormal">
    <item name="android:textSize">18sp</item>
    <item name="android:textColor">@color/textfarbe</item>
  </style>

  <style name="TextStyleUeberschrift"
    parent="@style/TextStyleNormal">
    <item name="android:textSize">24sp</item>
    <item name="android:gravity">center</item>
    <item name="android:paddingTop">10dp</item>
    <item name="android:paddingBottom">20dp</item>
  </style>

  <style name="TextStyleWarnung"
    parent="@style/TextStyleNormal">
    <item name="android:textSize">20sp</item>
    <item name="android:textColor">@color/textfarbe_warnung</item>
    <item name="android:gravity">center</item>
  </style>

  <style name="SchaltflaecheText"
    parent="android:Widget.TextView">

```

Farbdefinition wird auch ausgelagert

Selbstbezug möglich

sp: scale-independent Pixels

auch möglich:  
px: pixel  
mm: Millimeter

dp: density-independent Pixels

```

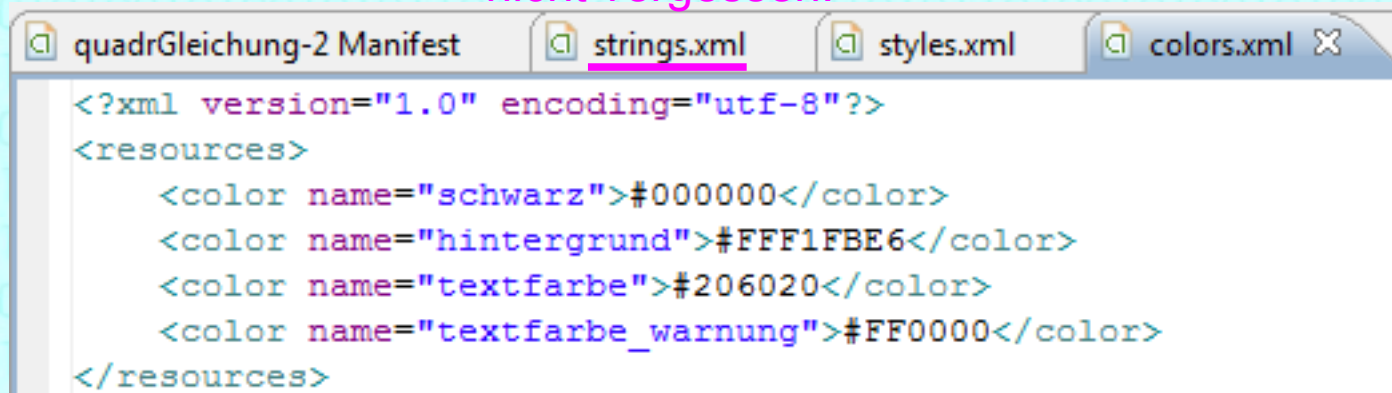
<style name="SchaltflaecheText"
  parent="android:Widget.TextView">
  <item name="android:textSize">18sp</item>
  <item name="android:textStyle">bold</item>
</style>

<style name="Schaltflaeche"
  parent="android:Widget.Button">
  <item name="android:textAppearance">@style/SchaltflaecheText</item>
</style>
</resources>

```

# quadrGleichung (II)

nicht vergessen!



```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <color name="schwarz">#000000</color>
  <color name="hintergrund">#FFF1FBE6</color>
  <color name="textfarbe">#206020</color>
  <color name="textfarbe_warnung">#FF0000</color>
</resources>
```

Farbdefinition mit rgb (rot|grün|blau)-Anteil in Hex-Code  
bzw.  
mit vorangestelltem Alphakanal (Transparenz) → argb (siehe *hintergrund*)

styles.xml und colors.xml liegen genau wie strings.xml im Ordner /res/values,  
diese werden dort im Kontextmenü mit *New File* erzeugen

# quadrGleichung (II)

```
quadrGleichung-2 Manifest  *quadr_gl_eingabe.java X
package eh.beispiele.quadrGl;

import android.app.Activity;

public class quadr_gl_eingabe extends Activity {

    /* Definition der Bezeichnungen für die Variablen,
     * die per Intent an die Ergebnis-Activity übergeben werden. */
    static final String A2 = "a2",
    static final String A1 = "a1";
    static final String A0 = "a0";

    /* Wird direkt nach der Erzeugung der Activity aufgerufen. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        /* zeigt das bereits compilierte Layout der Seite an
         * die XML-Seite dazu heisst res/layout/quadr_gl_eingabe_lay.xml */
        setContentView(R.layout.quadr_gl_eingabe_lay);
    }

    /* Testet, ob eine Zahl eingegeben wurde und wandelt den String in float um.
     * In der XML-Datei ist bereits angegeben, dass nur Zahlen, 1 Komma (also der Punkt)
     * und das Minuszeichen eingegeben werden dürfen. */
    private static float secure(String test) {
        float ok = 0.0f;
        if ((test.length() != 0) && (test != ".")) {
            ok = Float.parseFloat(test);
        }
        return ok;
    }
}
```

Konstanten für die Übergabe per Intent

Anpassen!

# quadrGleichung (II)

```

/* Einlesen der Daten und Übergabe an die Ergebnis-Activity */
public void onClickLoesen(final View sfNormal) {
    /* Die Eingabefelder werden wieder über ihre id gefunden und der Inhalt
    * der EditText den Fließkommazahl-Variablen zugewiesen. */
    final EditText txt_a2 = (EditText) findViewById(R.id.edt_a2);
    final float a2 = secure(txt_a2.getText().toString());
    final EditText txt_a1 = (EditText) findViewById(R.id.edt_a1);
    final float a1 = secure(txt_a1.getText().toString());
    final EditText txt_a0 = (EditText) findViewById(R.id.edt_a0);
    final float a0 = secure(txt_a0.getText().toString());

    /* Erzeugen des Intents für die Datenübergabe */
    final Intent i = new Intent(this, quadr_gl_ergebnis.class);

    /* die drei Parameter werden an den Intent übergeben */
    i.putExtra(A2, a2);
    i.putExtra(A1, a1);
    i.putExtra(A0, a0);

    /* starten der Ergebnis-Activity */
    startActivity(i);
}
}

```

Bezeichner zum  
Wiederfinden in  
Zielactivity

Übergabe an Intent

übergebener Wert

bei der Definition angegebene Activity wird gestartet

Expliziter Intent ist  
noch fehlerhaft:  
Activity für die  
Ergebnisausgabe  
muss noch angelegt  
werden

# quadrGleichung (II)

I  
N  
F  
O  
R  
M  
A  
T  
I  
K

```
quadrGleichung-2 Manifest  quadr_gl_eingabe.java  quadr_gl_ergebnis_lay.xml X
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:orientation="vertical"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent" >

  <TextView
    style="@style/TextStyleUeberschrift"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/txt_ergebnis_intro" />

  <TableLayout
    android:layout_height="wrap_content"
    android:layout_width="wrap_content"
    android:layout_gravity="center_horizontal">
    <TableRow>
      <TextView android:id="@+id/txt_xlinks1"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
      />
      <TextView android:id="@+id/txt_xrechts1"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
      />
    </TableRow>
    <TableRow>
```

Eigenes Layout für die Ausgabe

# quadrGleichung (II)

```
quadrGleichung-2 Manifest  quadr_gl_ergebnis.java x
package eh.beispiele.quadrgl;

//mit Strg Shift "O" werden die benötigten Klassen automatisch eingebunden
import android.app.Activity;

public class quadr_gl_ergebnis extends Activity{

    /* Definition der Bezeichnungen für die Variablen,
     * die per Intent von der Eingabe-Activity übergeben werden.
     * Muss identisch sein!*/
    static final String A2 = "a2";
    static final String A1 = "a1";
    static final String A0 = "a0";

    /* Wird direkt nach der Erzeugung der Activity aufgerufen. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        /* zeigt das bereits compilierte Layout der Seite an
         * die XML-Seite dazu heisst res/layout/quadr_gl_ergebnis_lay.xml */
        setContentView(R.layout.quadr_gl_ergebnis_lay);

        /* Der komplette Inhalt des Intents wird eingelesen */
        final Bundle extras = getIntent().getExtras();
        if (extras != null) {
            /* Instanz der Klasse berechnung (extra Datei) */
            final quadr_gl_berechne erg = new quadr_gl_berechne();

            /* Zuweisen der sechs Parameter */
            erg.a2 = extras.getFloat(A2);
```

Bezeichner zum Wiederfinden  
der Übergabeparameter

# quadrGleichung (II)

```
/* Wird direkt nach der Erzeugung der Activity aufgerufen. */
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    /* zeigt das bereits compilierte Layout der Seite an
     * die XML-Seite dazu heisst res/layout/quadr_gl_ergebnis_lay.xml */
    setContentView(R.layout.quadr_gl_ergebnis_lay);

    /* Der komplette Inhalt des Intents wird eingelesen */
    final Bundle extras = getIntent().getExtras();    alle Parameter einlesen
    if (extras != null) {
        /* Instanz der Klasse berechnung (extra Datei) */
        final quadr_gl_berechne erg = new quadr_gl_berechne();    ausgelagerte
                                                                    Berechnung
                                                                    instanzieren

        /* Zuweisen der sechs Parameter */
        erg.a2 = extras.getFloat(A2);
        erg.a1 = extras.getFloat(A1);
        erg.a0 = extras.getFloat(A0);    Parameter identifizieren und zuweisen

        /* Ergebnis berechnen und anzeigen (s.u.) */
        zeigeErgebnis(erg);    Berechnung durchführen und Ergebnisse anzeigen
    }
}
```

# quadrGleichung (II)

```

/* Ergebnis berechnen und Anzeigen */
private void zeigeErgebnis(quadr_gl_berechne erg) {
    /* Titel der Activity festlegen */
    setTitle("MatheApp"); Titel der Activity kann geändert werden

    /* Ergebnis wird berechnet, in msg wird ggf. Fehlermeldung geliefert */
    String msg = erg.berechne();

    /* falls kein Fehler gemeldet wird, wird TextView per id gesucht und
     * das Ergebnis formatiert geschrieben. */
    if (msg == "2") {
        final TextView txt_ergx1_links = (TextView) findViewById(R.id.txt_xlinks1);
        txt_ergx1_links.setText("x1 = ");
        final TextView txt_ergx1_rechts = (TextView) findViewById(R.id.txt_xrechts1);
        txt_ergx1_rechts.setText(String.format("%.3f%n", erg.x1));
        final TextView txt_ergx2_links = (TextView) findViewById(R.id.txt_xlinks2);
        txt_ergx2_links.setText("x2 = ");
        final TextView txt_ergx2_rechts = (TextView) findViewById(R.id.txt_xrechts2);
        txt_ergx2_rechts.setText(String.format("%.3f%n", erg.x2));
    }
    else {
        if (msg == "1") {
            final TextView txt_ergx1_links = (TextView) findViewById(R.id.txt_xlinks1);
            txt_ergx1_links.setText("x = ");
            final TextView txt_ergx1_rechts = (TextView) findViewById(R.id.txt_xrechts1);
            txt_ergx1_rechts.setText(String.format("%.3f%n", erg.x1));
        }
        else {
            /* falls Fehler, Ausgabe der Fehlermeldung */
            final TextView txt_ergx = (TextView) findViewById(R.id.quagl_txt_info);
            txt_ergx.setText(msg);
        }
    }
}

```

Löschen alter Werte kann entfallen,  
da Activity jedes Mal neu initialisiert wird.



# quadrGleichung (II)

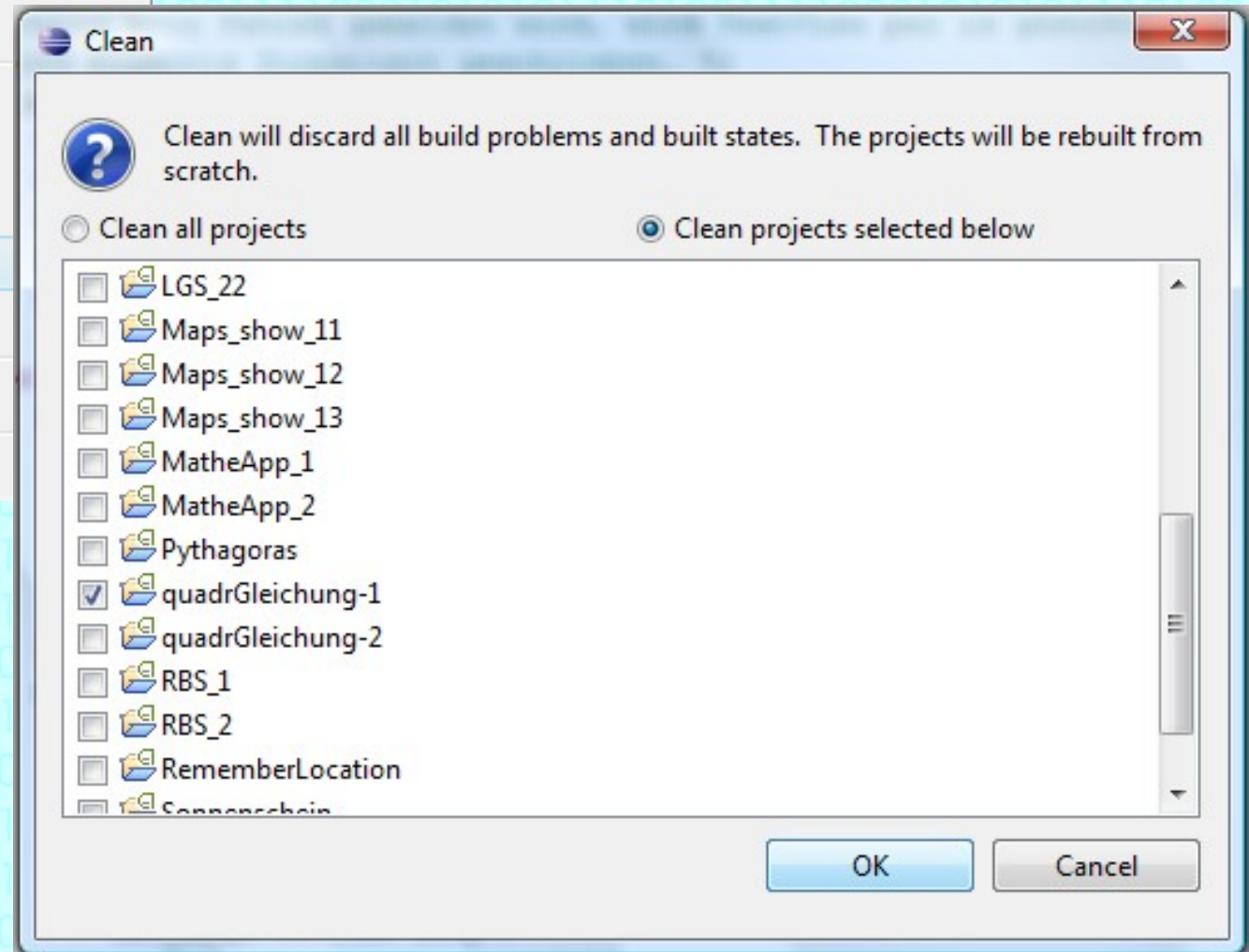
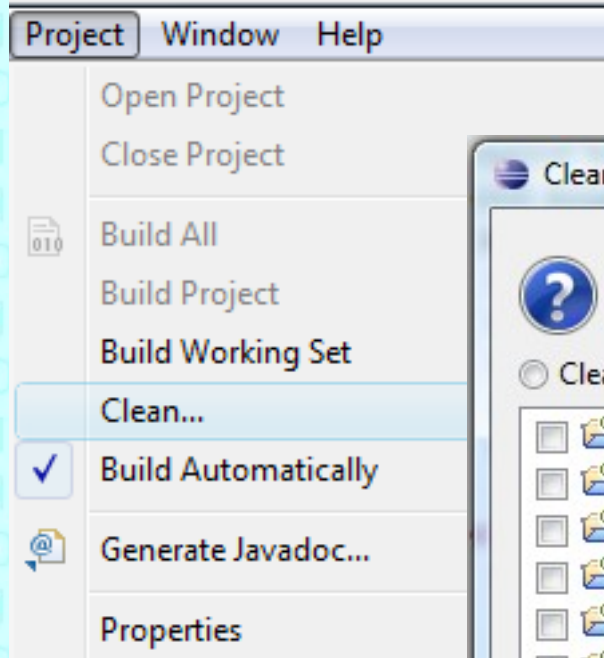
```
quadrGleichung-2 Manifest x | quadr_gl_ergebnis.java | quadr_gl_eingabe.java | strings.xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="eh.beispiele.quadrgl"
    android:versionCode="1"
    android:versionName="1.0">
    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".quadr_gl_eingabe"
            android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".quadr_gl_ergebnis" />
    </application>
    <uses-sdk android:minSdkVersion="7" />
</manifest>
```

**WICHTIG !!!**

Werden weitere Activities nicht im Manifest vereinbart, stürzt die App noch beim Start *ohne Fehlermeldung* ab.

# wenn gar nichts mehr hilft ...

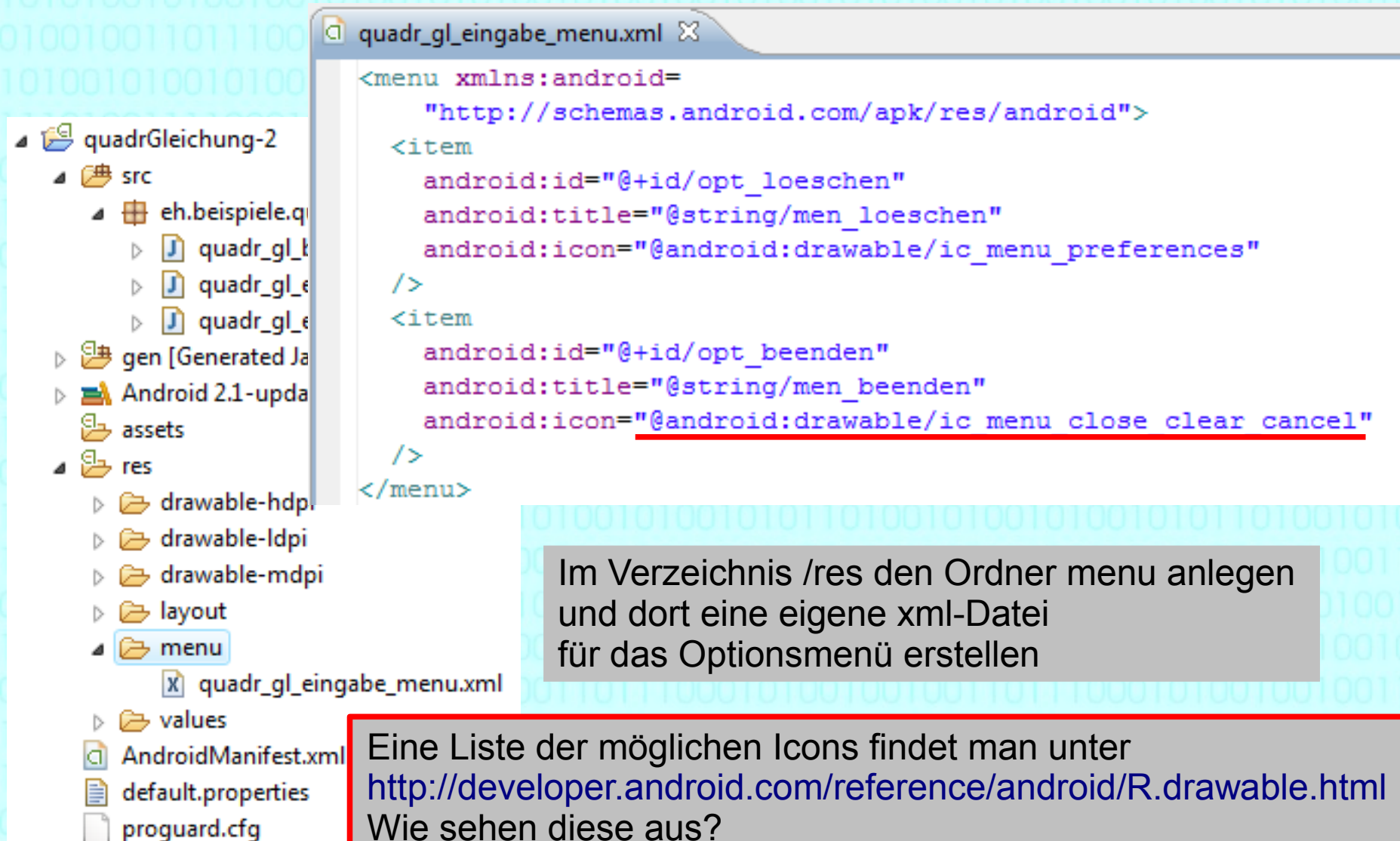
I  
N  
F  
O  
R  
M  
A  
T  
I  
K



# Android-Apps

## Das Optionsmenü

# Das Optionsmenü



```

<menu xmlns:android=
  "http://schemas.android.com/apk/res/android">
  <item
    android:id="@+id/opt_loeschen"
    android:title="@string/men_loeschen"
    android:icon="@android:drawable/ic_menu_preferences"
  />
  <item
    android:id="@+id/opt_beenden"
    android:title="@string/men_beenden"
    android:icon="@android:drawable/ic_menu_close_clear_cancel"
  />
</menu>

```

Im Verzeichnis /res den Ordner menu anlegen und dort eine eigene xml-Datei für das Optionsmenü erstellen

Eine Liste der möglichen Icons findet man unter <http://developer.android.com/reference/android/R.drawable.html>  
Wie sehen diese aus?  
<http://androiddrawableexplorer.appspot.com/>

# Das Optionsmenü

```

/* Erzeugt das OptionsMenü
 * die XML-Seite dazu heisst res/menu/eingabe_menu.xml */
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.quadr_gl_eingabe_menu, menu);
    return super.onCreateOptionsMenu(menu);
}

/* wird bei der Wahl eines Menüpunktes des Optionsmenü
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
        case R.id.opt_loeschen:
            /* die drei Parameter werden gelöscht */
            final EditText edtxt_x2 = (EditText) findV
            edtxt_x2.setText("");
            final EditText edtxt_x1 = (EditText) findV
            edtxt_x1.setText("");
            final EditText edtxt_x0 = (EditText) findV
            edtxt_x0.setText("");
            return true;
        case R.id.opt_beenden:
            finish(); // Activity beenden (wird aber
            return true;
        default: super.onOptionsItemSelected(item);
    }
    return false;
}

```

xml-Datei in  
Menü-Daten  
umwandeln



in quadr\_gl\_eingabe\_menu.xml vereinbarte id

Ohne *return*  
läuft die App in  
den nächsten  
Fall ...

strings.xml nicht vergessen!

# Übung

I

N

F

O

R

M

A

T

I

K

Erstellen Sie eine App mit zwei Activities und einem Optionsmenü zu einem mathematischen Problem nach Wahl.

Z.B.:

- LGS (2|2)
- ggT
- kgV
- Binominalkoeffizient
- Binominalverteilung
- ...