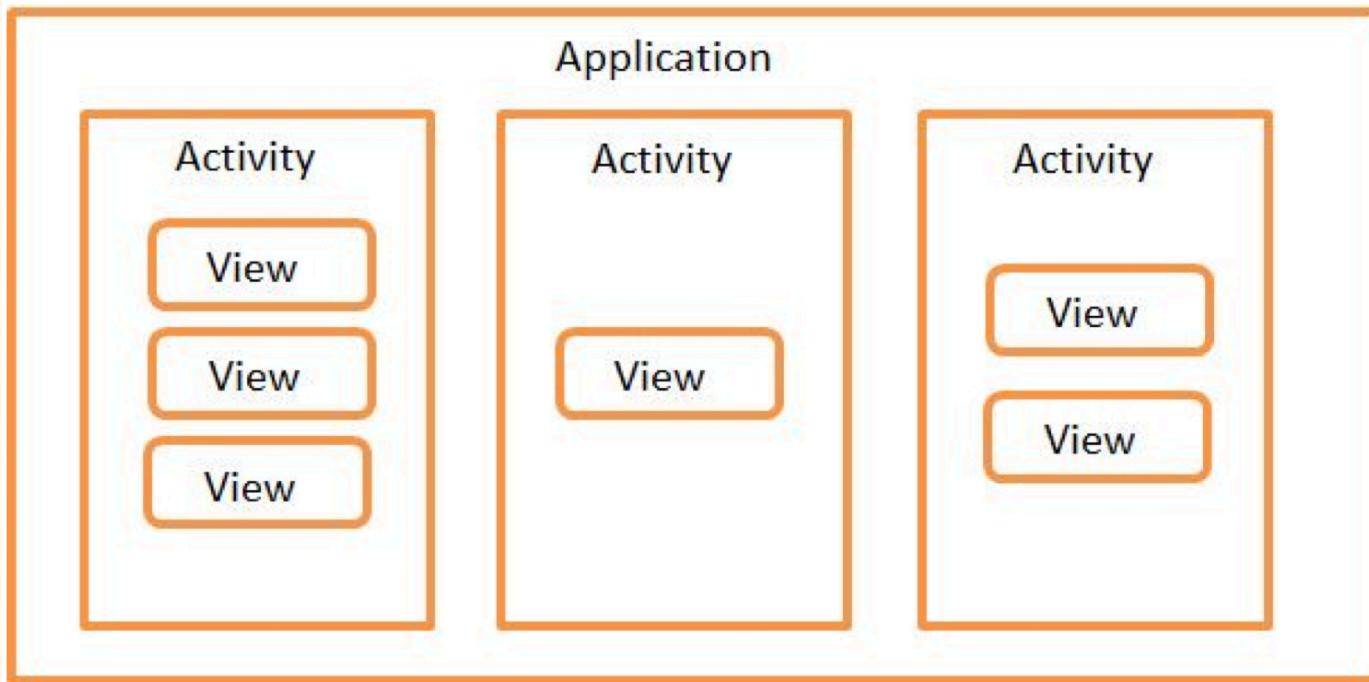


Android App UI

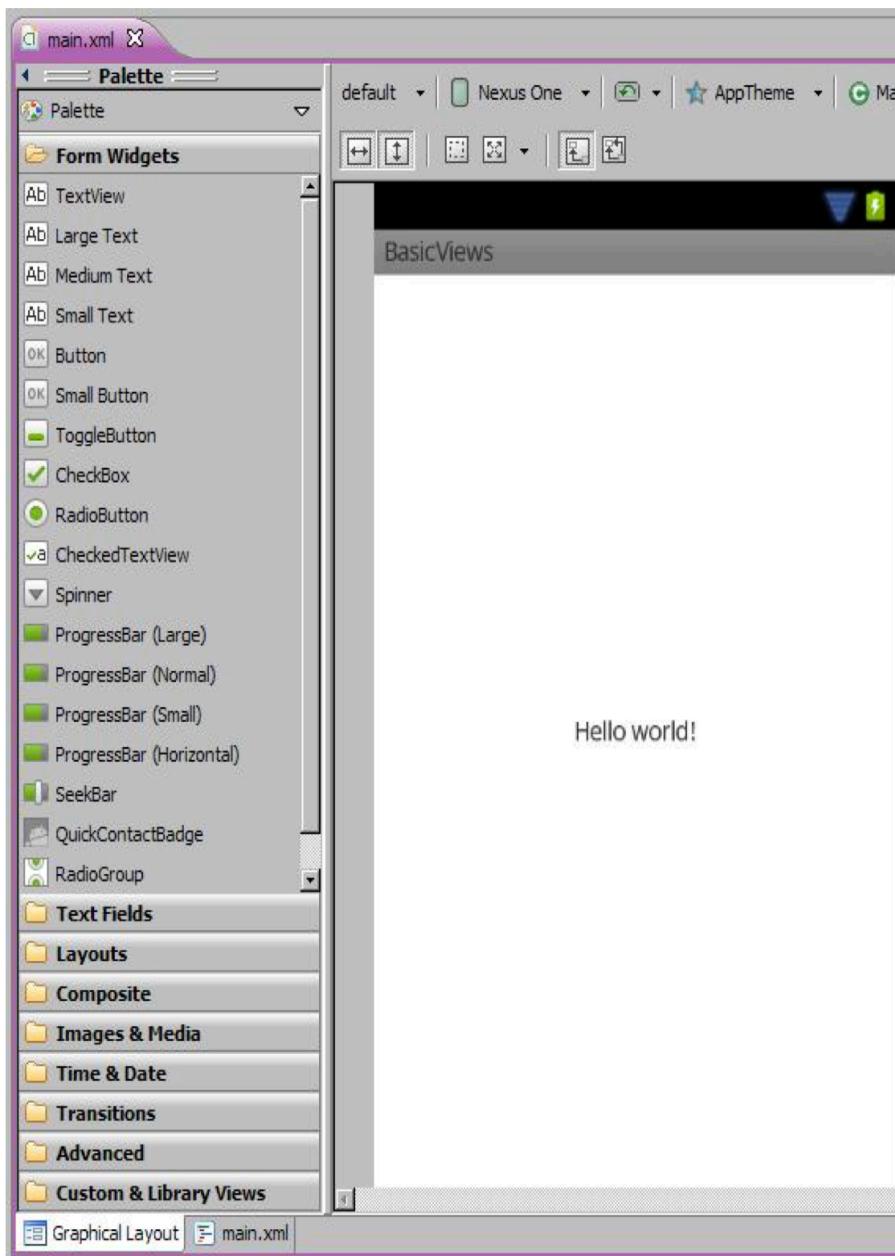
Prof. Dr.-Ing. V.Brovkov

App UI in Android OS



- App: eine oder Mehrere Activities
- z-Fenstersystem (Was ist es?)
- UI: Ein oder mehrere Views
- Layout (xml) & Activity (Java) & Strings & Colors ... (Wie?)
- Views anordnen (Wie?)
- Views Abmessungen (Wie?)
- Verschiedene Geräte, verschiedene Displays (Wie?)
- Display Anordnung
- ...

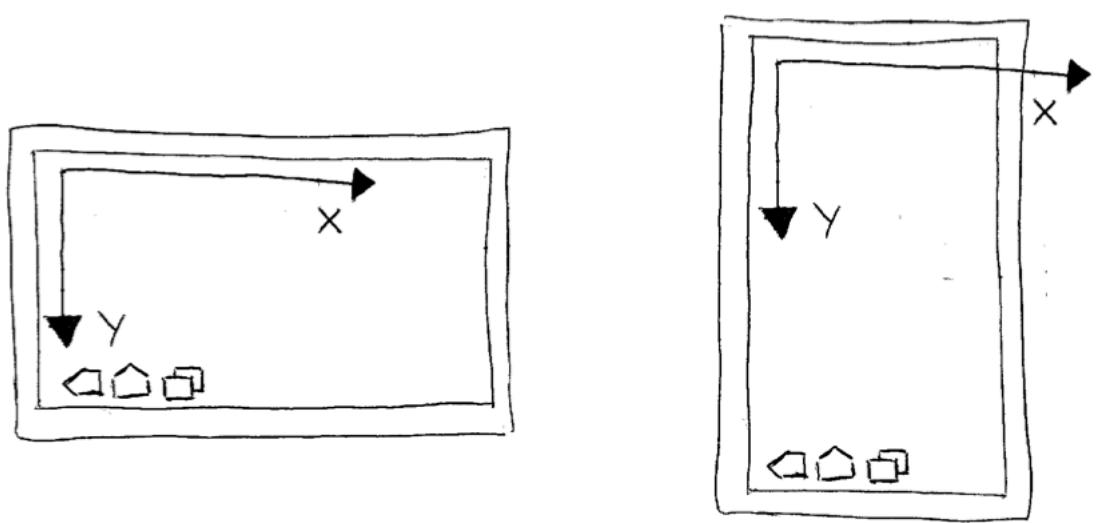
Vorhandene Views



- Views:
 - TextView
 - Button
 - ToggleButton
 - CheckBox
 - RadioButton
 - ...
- ViewGroups
 - LinearLayout
 - TableLayout
 - RelativeLayout
 - AbsoluteLayout
 - ...

Bildschirm Koordinatensystem

- Bildschirm Koordinatensystem
- Problem:
Ausrichtung/Orientation?
- Noch ein Problem:
Koordinatensysteme
Bildschirm / Gyroskop /
Beschleunigungssensor



Folgende Stichwörter sind wichtig im Zusammenhang mit dem Bildschirm:

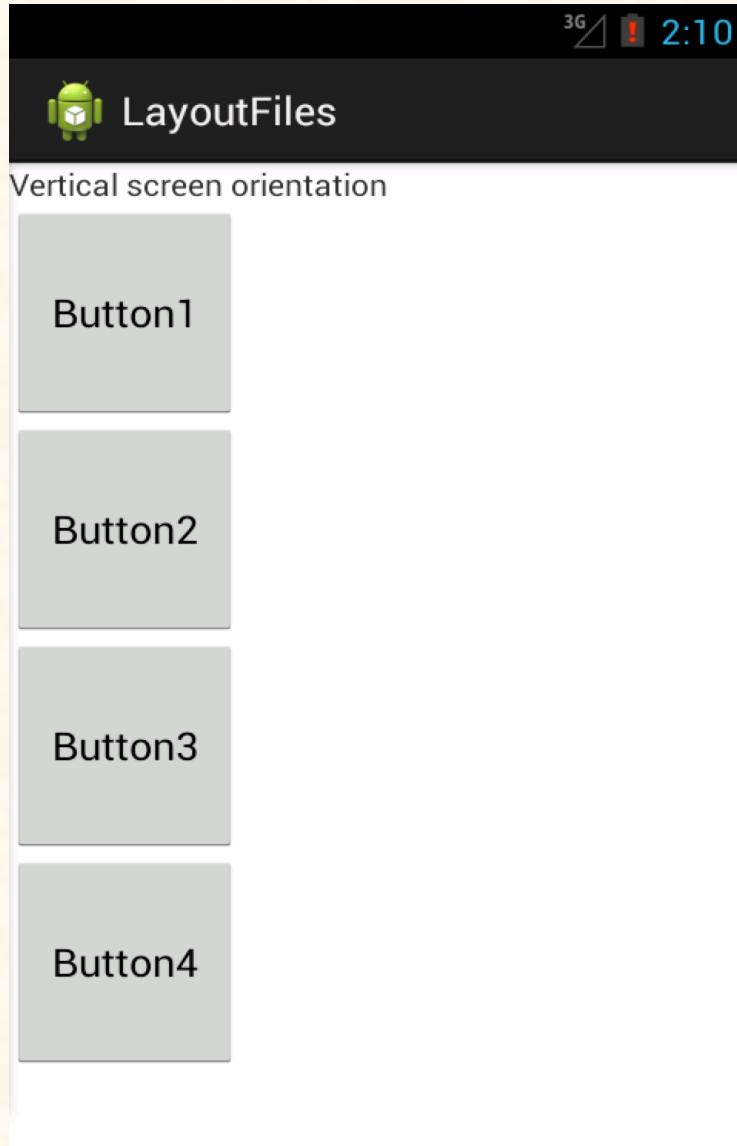
1. `Display.getMetrics(DisplayMetrics);`
2. `getwindowManager().getDefaultDisplay();` //innerhalb einer Activity
3. `getwindowManager().getDefaultDisplay().getOrientation();` //innerhalb einer Activity
4. `getresources().getconfiguration();` //innerhalb einer Activity
5. `configuration.screenlayout, configuration.uiMode`
6. `<supports-screens />` im `AndroidManifest.xml`-File

Bildschirm Ausrichtung

- Projekt Beispiel: LayoutFiles

Bildschirm-Ausrichtung

Nehmen wir an, dass eine Layout-Datei entworfen wurde:



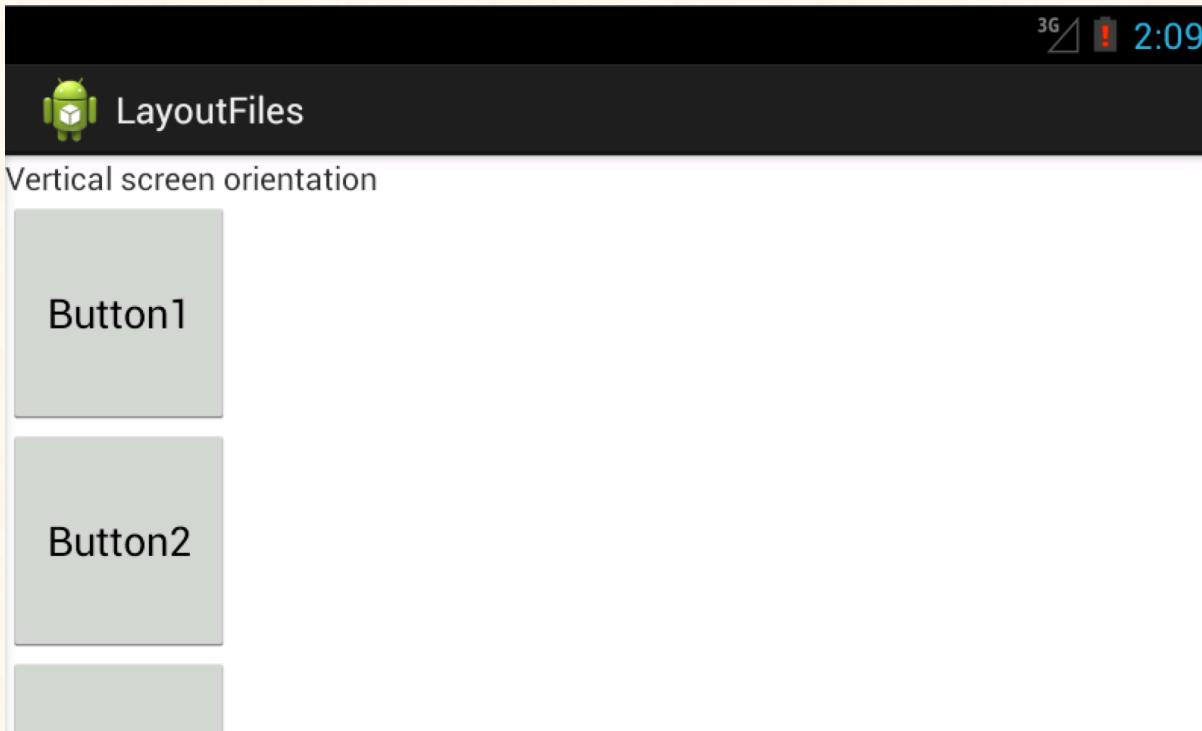
```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     android:orientation="vertical"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent">
7     <TextView
8         android:id="@+id/textView1"
9         android:layout_width="wrap_content"
10        android:layout_height="wrap_content"
11        android:text="Vertical screen orientation">
12    </TextView>
13    <LinearLayout
14        android:layout_height="wrap_content"
15        android:layout_width="match_parent"
16        android:id="@+id/linearLayout1"
17        android:orientation="vertical">
18        <Button
19            android:text="Button1"
20            android:id="@+id/button1"
21            android:layout_height="100dp"
22            android:layout_width="100dp">
23        </Button>
24        <Button
25            android:text="Button2"
26            android:id="@+id/button2"
27            android:layout_height="100dp"
28            android:layout_width="100dp">
29        </Button>
```

Bildschirm-Ausrichtung

```
7 public class MainActivity extends Activity {  
8     @Override  
9     public void onCreate(Bundle savedInstanceState) {  
10         super.onCreate(savedInstanceState);  
11         setContentView(R.layout.activity_main);  
12     }  
13     @Override  
14     public boolean onCreateOptionsMenu(Menu menu) {  
15         getMenuInflater().inflate(R.menu.activity_main, menu);  
16         return true;  
17     }  
18 }
```

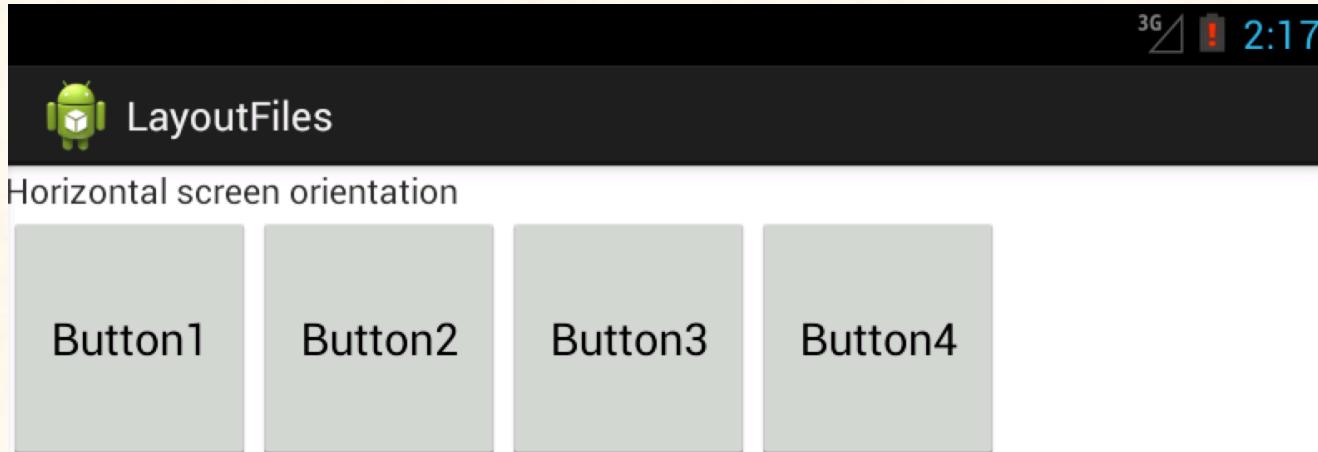
- Die Layout-Datei wird mit entsprechender Activity während dessen Erstellung verbunden

Bildschirm-Ausrichtung



- Beim Drehen des Gerätes wird das Bildschirm aber nicht mehr verwendbar
- Das Drehen ist eine sehr häufige Benutzer-Aktion

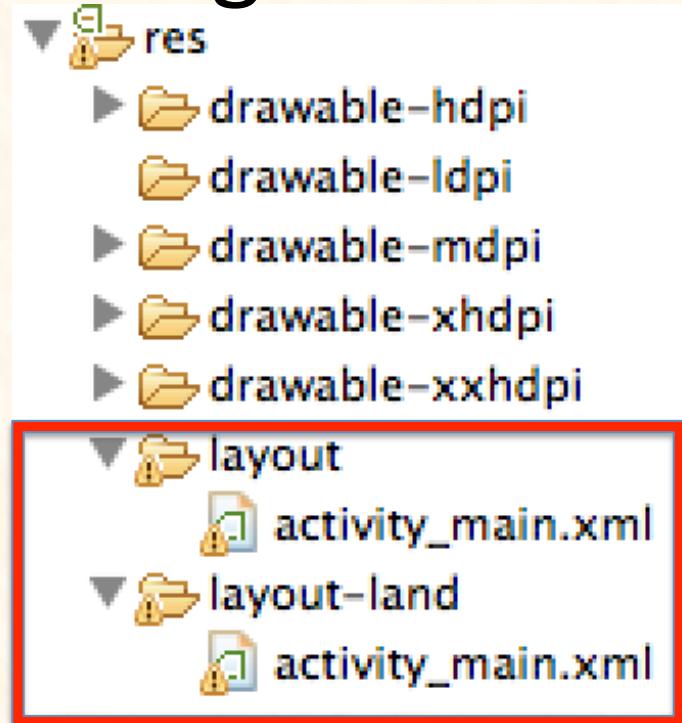
Bildschirm-Ausrichtung



- Man kann eine entsprechende Layout-Datei bauen, die das passende Aussehen des Bildschirmes in Landscape-Ausrichtung beschreibt. Das wird aber der Porträt-Ausrichtung nicht passen
- Beide Layouts sind erforderlich

Bildschirm-Ausrichtung

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     android:orientation="vertical"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent">
7     <TextView
8         android:id="@+id/textView1"
9         android:layout_width="wrap_content"
10        android:layout_height="wrap_content"
11        android:text="Horizontal screen orientation">
12    </TextView>
13    <LinearLayout
14        android:layout_height="wrap_content"
15        android:layout_width="match_parent"
16        android:id="@+id/linearLayout1"
17        android:orientation="horizontal">
18        <Button
19            android:text="Button1"
20            android:id="@+id/button1"
21            android:layout_height="100dp"
22            android:layout_width="100dp">
23        </Button>
24        <Button
25            android:text="Button2"
26            android:id="@+id/button2"
27            android:layout_height="100dp"
28            android:layout_width="100dp">
29        </Button>
30        <Button
31            android:text="Button3"
32            android:id="@+id/button3"
33            android:layout_height="100dp"
34            android:layout_width="100dp">
35        </Button>
```

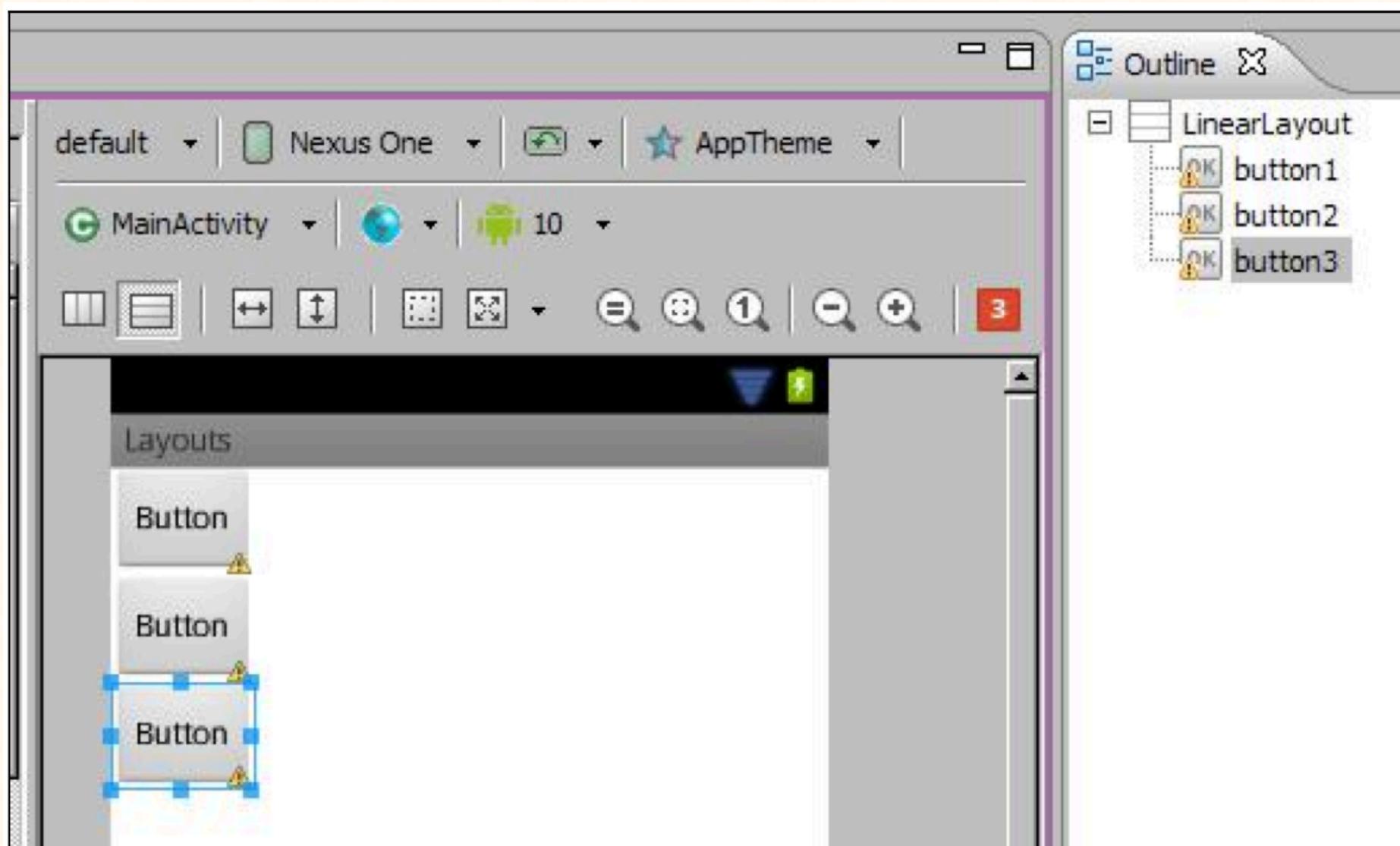


Die Lösung:
die beiden Layout-
Dateien zu behalten!

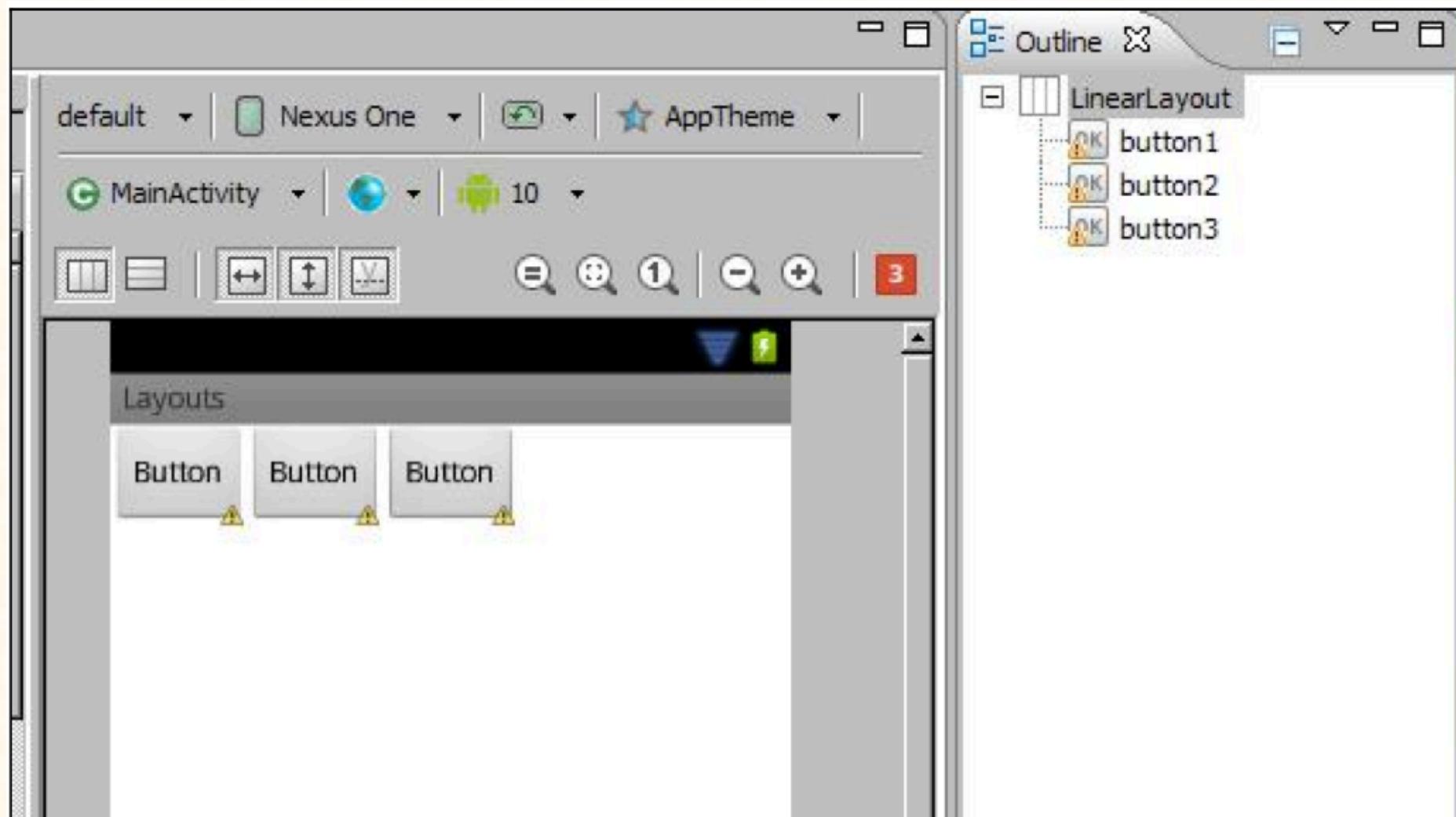
ViewGroups

- die gleiche Idee wie in Java Layout-Managers
- dient als Container für Views
- steuert die Lage und die Abmessungen von Views
- kann die untergeordnete ViewGroups beibehalten

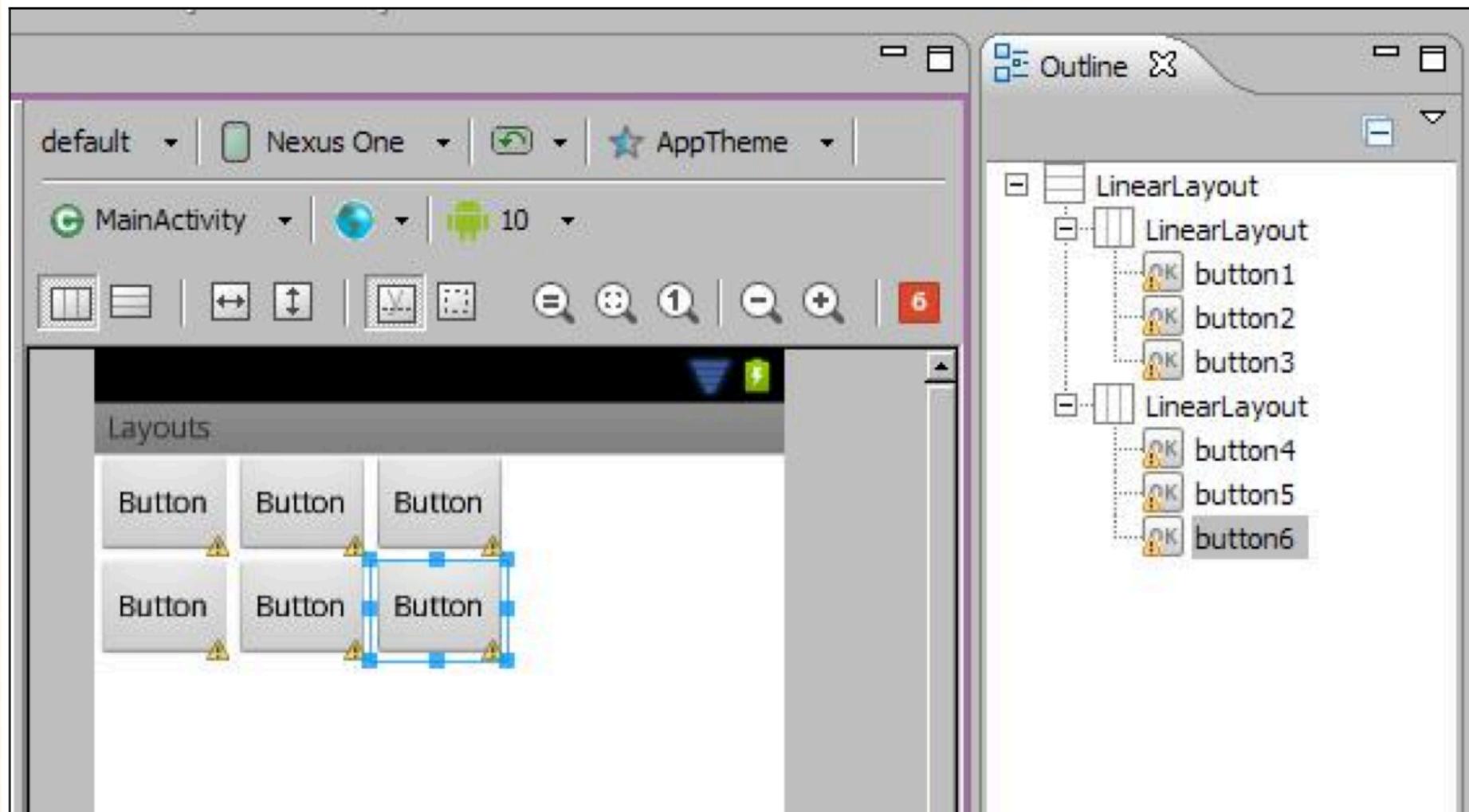
LinearLayout



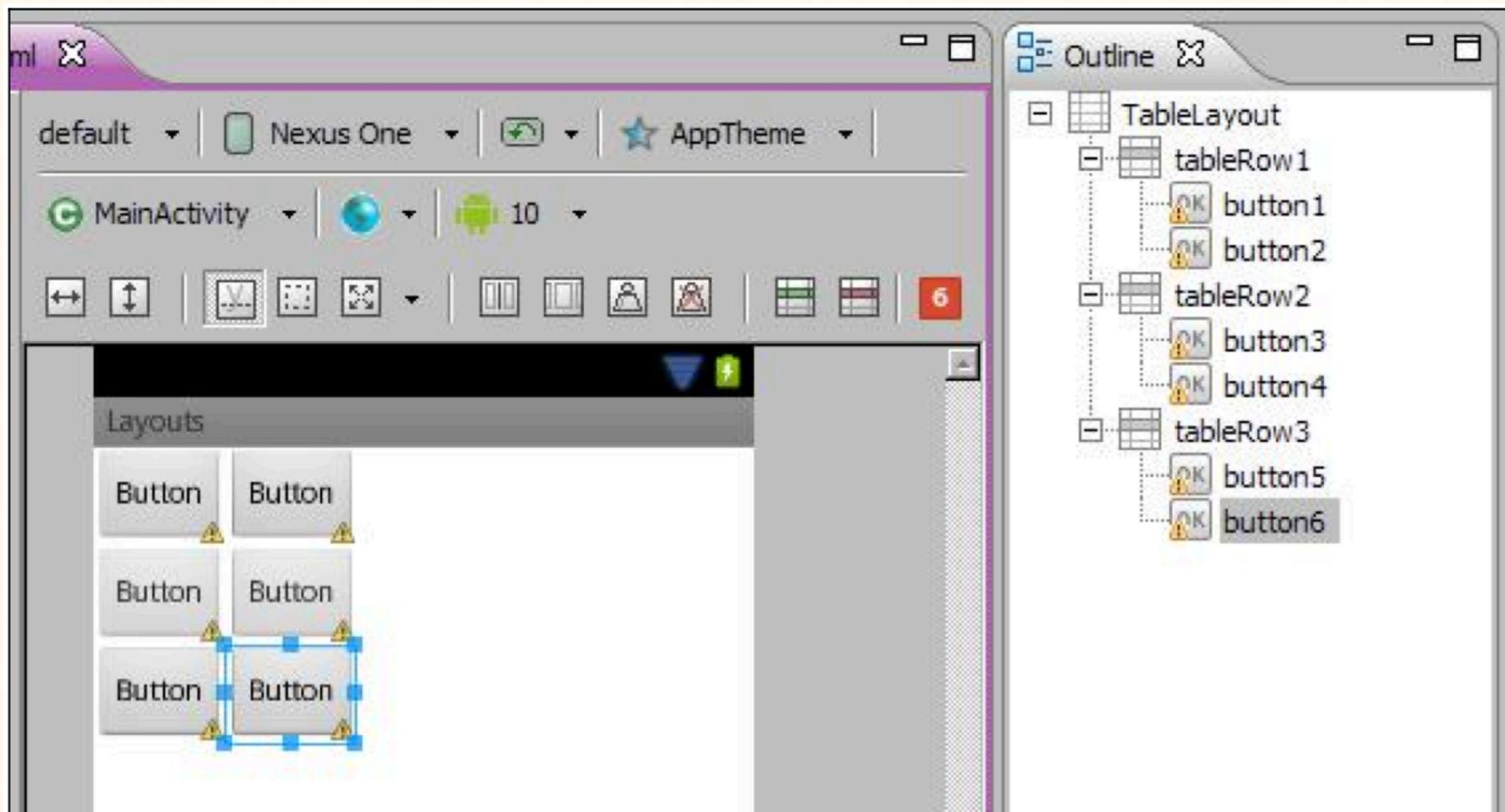
LinearLayout



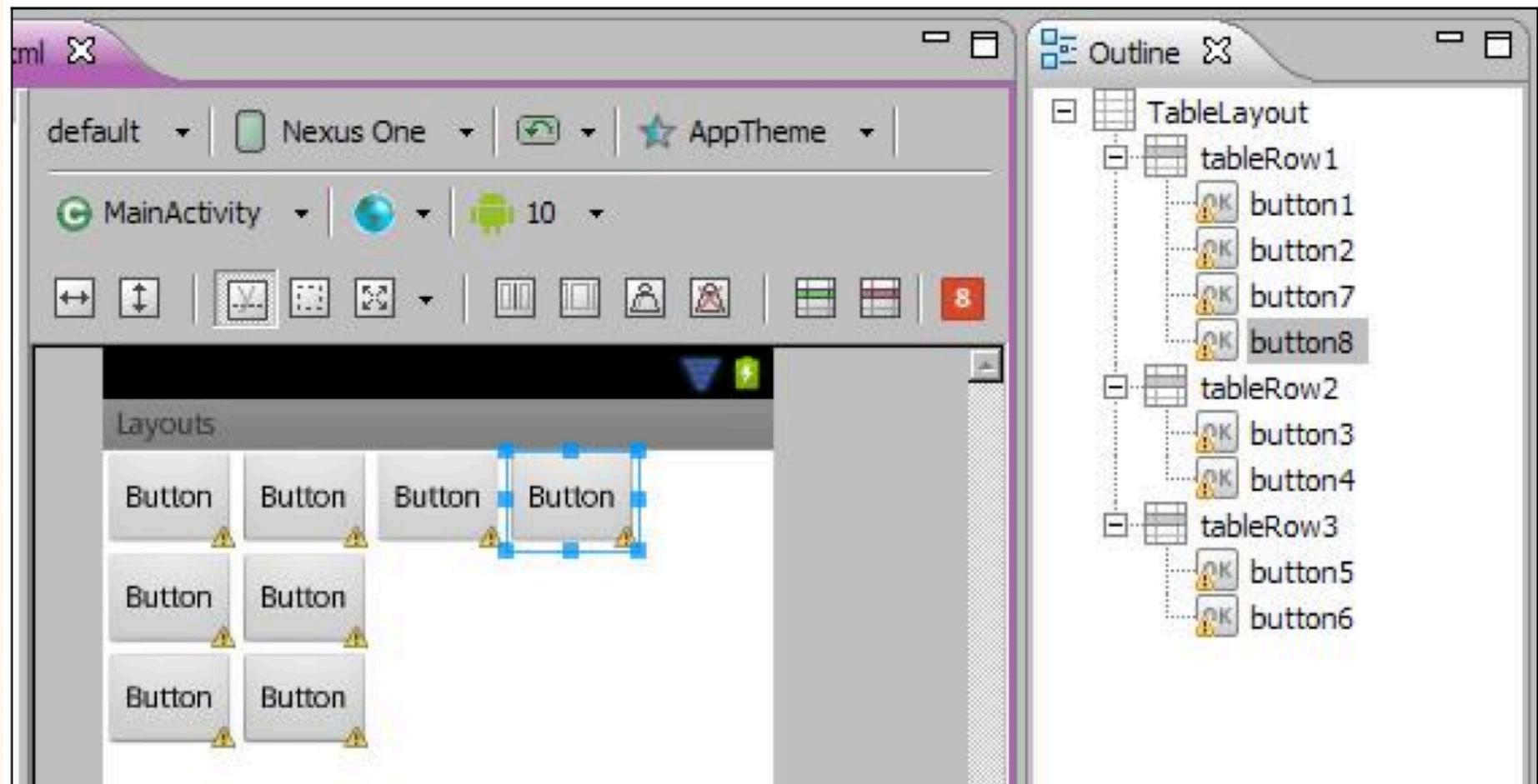
LinearLayout



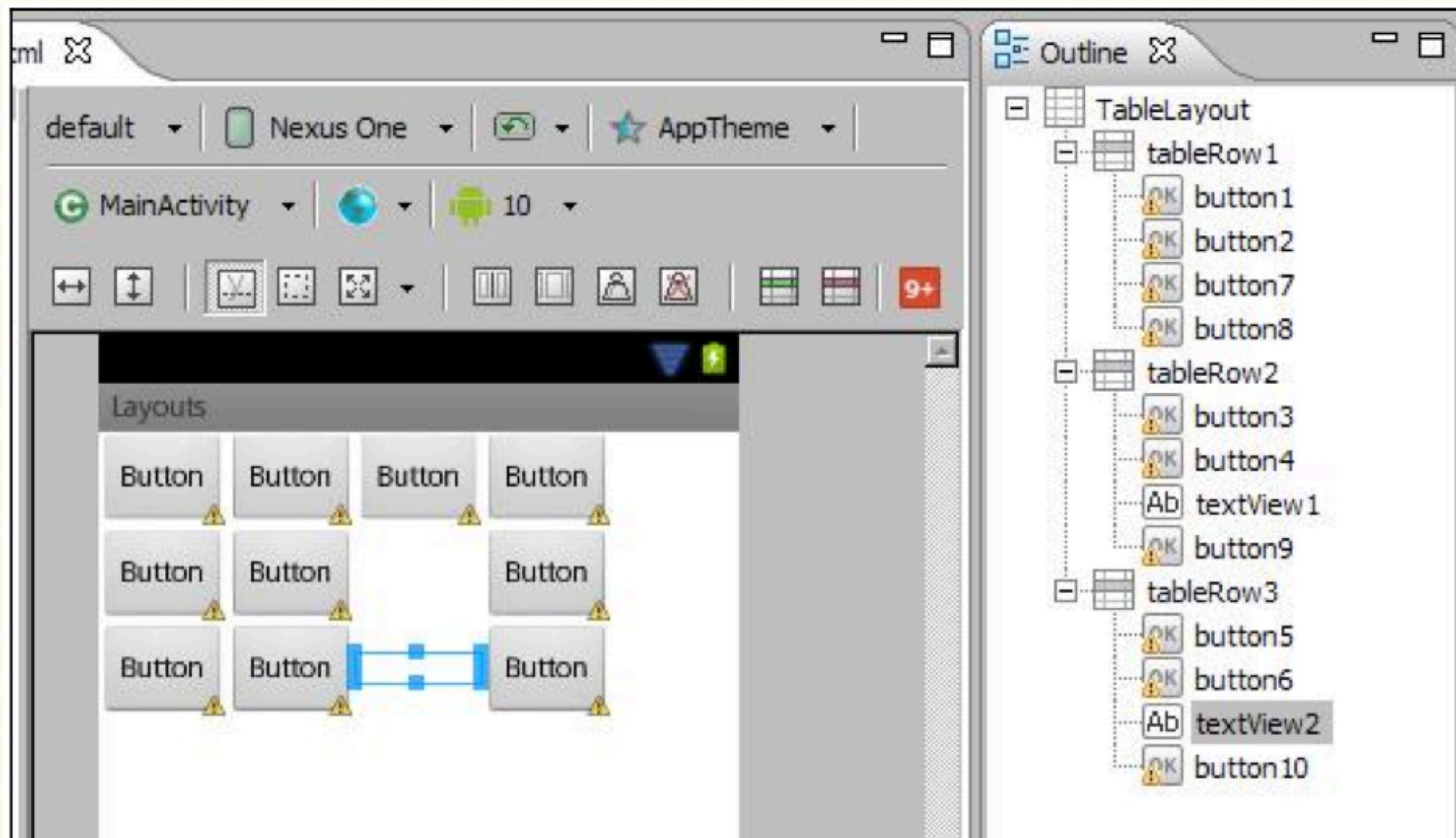
TableLayout



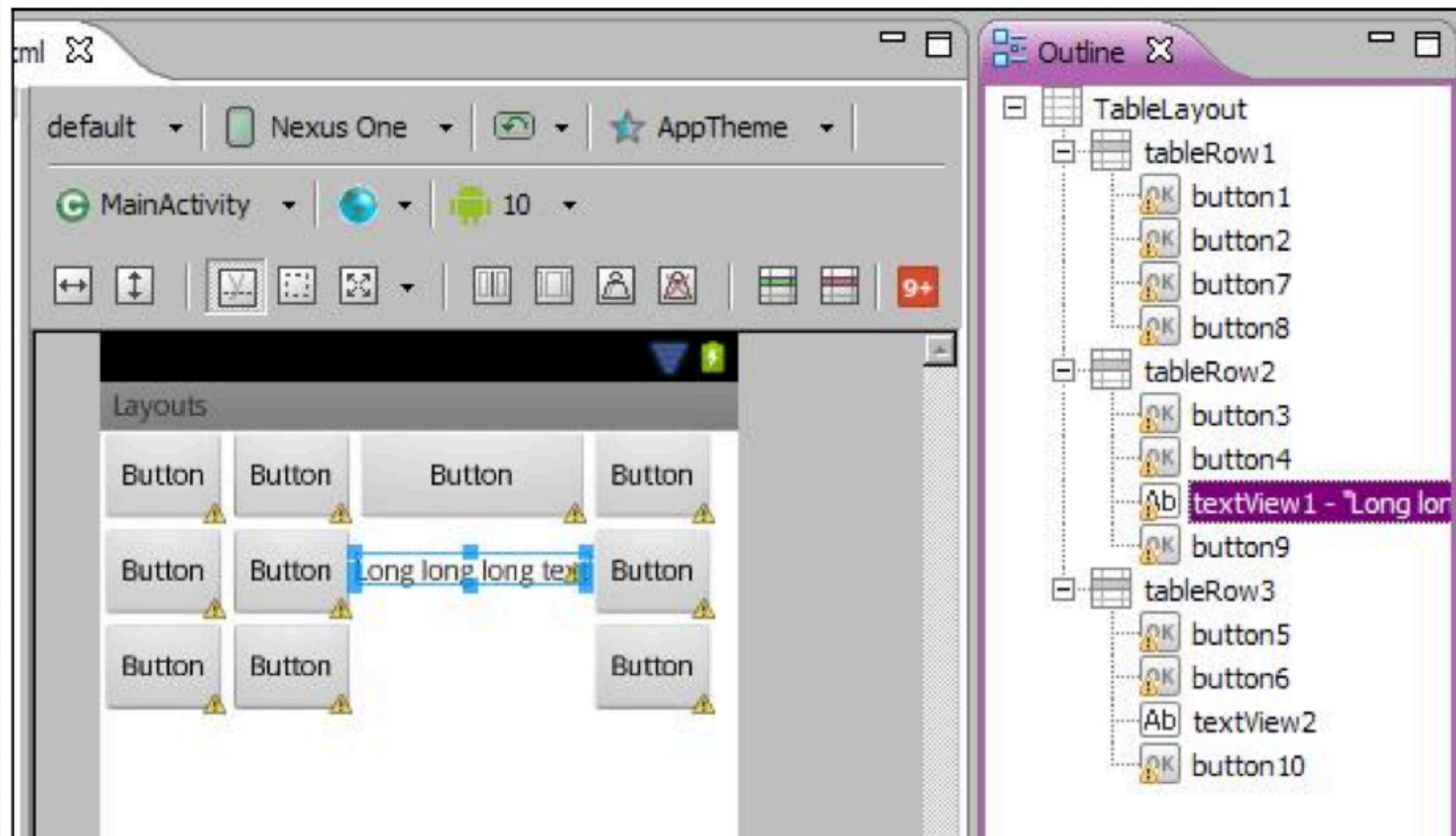
TableLayout



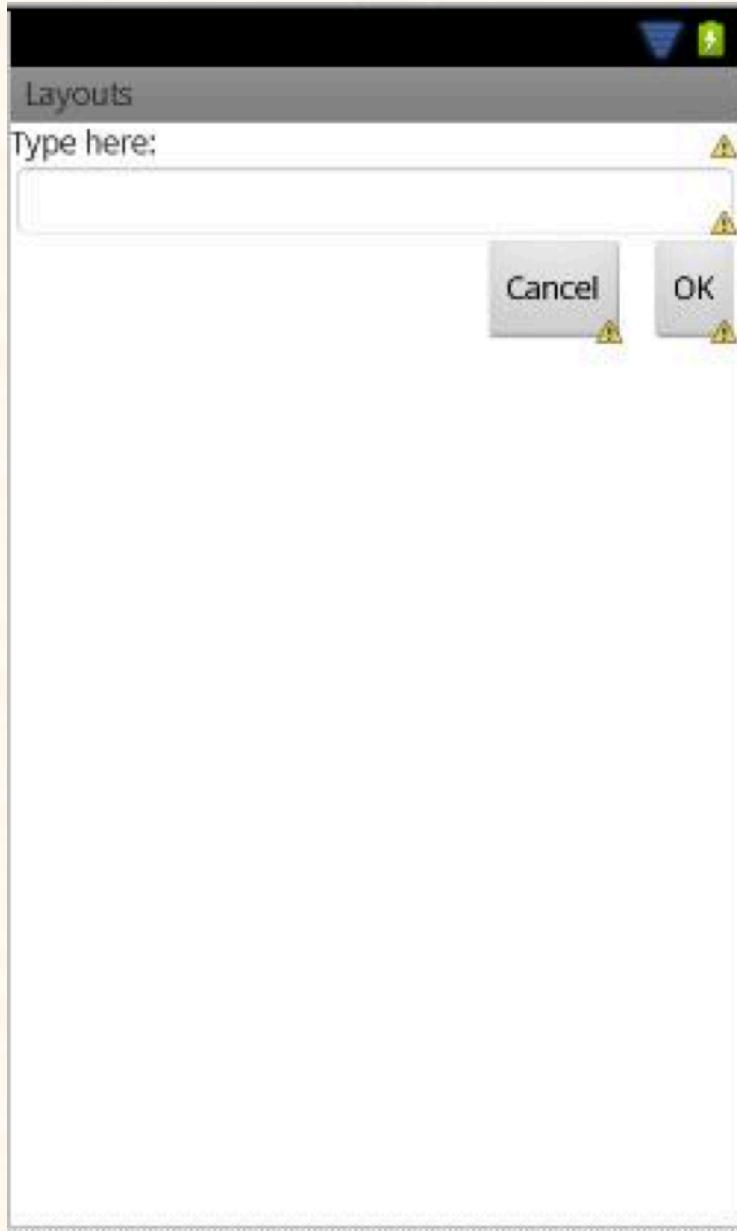
TableLayout



TableLayout



RelativeLayout



```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TextView
        android:id="@+id/label"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Type here:"/>
    <EditText
        android:id="@+id/entry"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/label"
        android:background="@android:drawable/editbox_background"/>
    <Button
        android:id="@+id/ok"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentRight="true"
        android:layout_below="@+id/entry"
        android:layout_marginLeft="10dip"
        android:text="OK"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignTop="@+id/ok"
        android:layout_toLeftOf="@+id/ok"
        android:text="Cancel"/>
</RelativeLayout>
```

RelativeLayout

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TextView
        android:id="@+id/label"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Type here:">
    </TextView>
    <EditText
        android:id="@+id/entry"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/label"
        android:background="@android:drawable/editbox_background">
    </EditText>
```

RelativeLayout

```
<Button  
    android:id="@+id/ok"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_alignParentRight="true"  
    android:layout_below="@+id/entry"  
    android:layout_marginLeft="10dip"  
    android:text="OK">  
</Button>  
<Button  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_alignTop="@+id/ok"  
    android:layout_toLeftOf="@+id/ok"  
    android:text="Cancel">  
</Button>  
</RelativeLayout>
```

RelativeLayout

- xml Layout-Datei Records:
 - android:layout_width="match_parent"
 - android:layout_height="wrap_content"
 - android:id="@+id/entry"
 - ...
- android bedeutet namespace
- id – View ID
- layout_width und layout_height – die Breite und die Hohe von View
 - kann als absolute Grosse gegeben werden
 - match_parent oder fill_parent – maximal mögliche Grosse im Container
 - wrap_content – entspricht dem Content

RelativeLayout

- **TextView**
 - android:id="@+id/**label**" - ID
 - android:layout_width="match_parent" – die volle Breite des Containers
 - android:layout_height="wrap_content" – die Höhe entsprechend dem Inhalt;
 - keine Verbindung mit anderen Komponenten
- **EditText**
 - android:id="@+id/**entry**" - ID
 - android:layout_width="match_parent" - die volle Breite des Containers
 - android:layout_height="wrap_content" - die Höhe entsprechend dem Inhalt
 - android:layout_below="@id/**label**" – **unter** dem TextView (link mit ID)

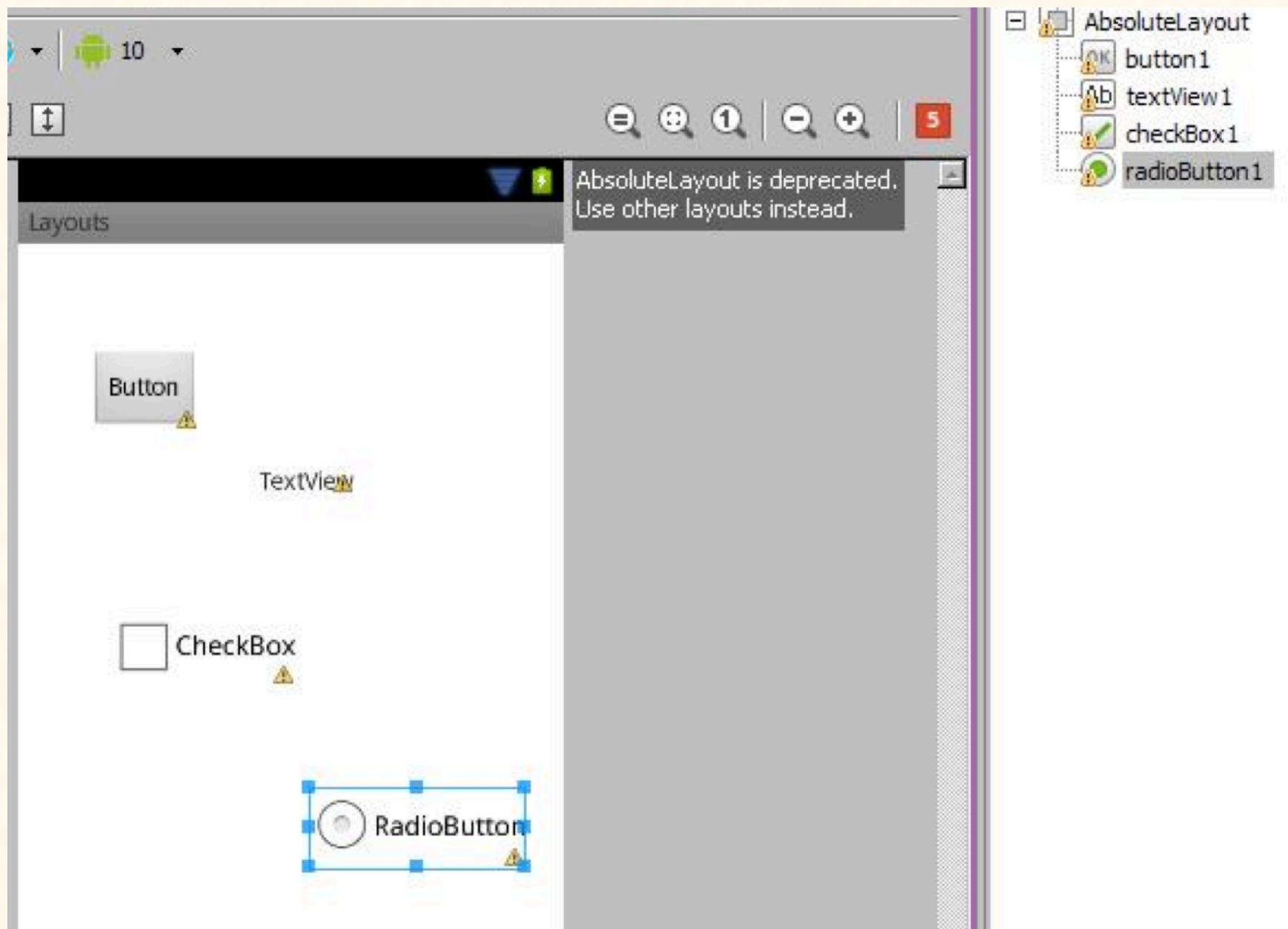
RelativeLayout

- **Button_OK**
 - android:id="@+id/**ok**" – ID
 - android:layout_width="wrap_content" - die Breite entsprechend dem Inhalt
 - android:layout_height="wrap_content" – die Höhe entsprechend dem Inhalt
 - android:layout_below="@id/entry" – **unter** dem EditText
 - android:layout_alignParentRight="true" – Ausrichtung zur rechten Seite von Parent
 - android:layout_marginLeft="10dip" – hat eine Lücke links (eine Lücke mit Button_Cancel)

RelativeLayout

- **Button_Cancel**
 - android:layout_width="wrap_content" - die Breite entsprechend dem Inhalt
 - android:layout_height="wrap_content" – die Höhe entsprechend dem Inhalt
 - android:layout_toLeftOf="@+id/ok" – links von Button_OK
 - android:layout_alignTop="@+id/ok" – ausgerichtet am oberen Rand von Button_OK

AbsoluteLayout



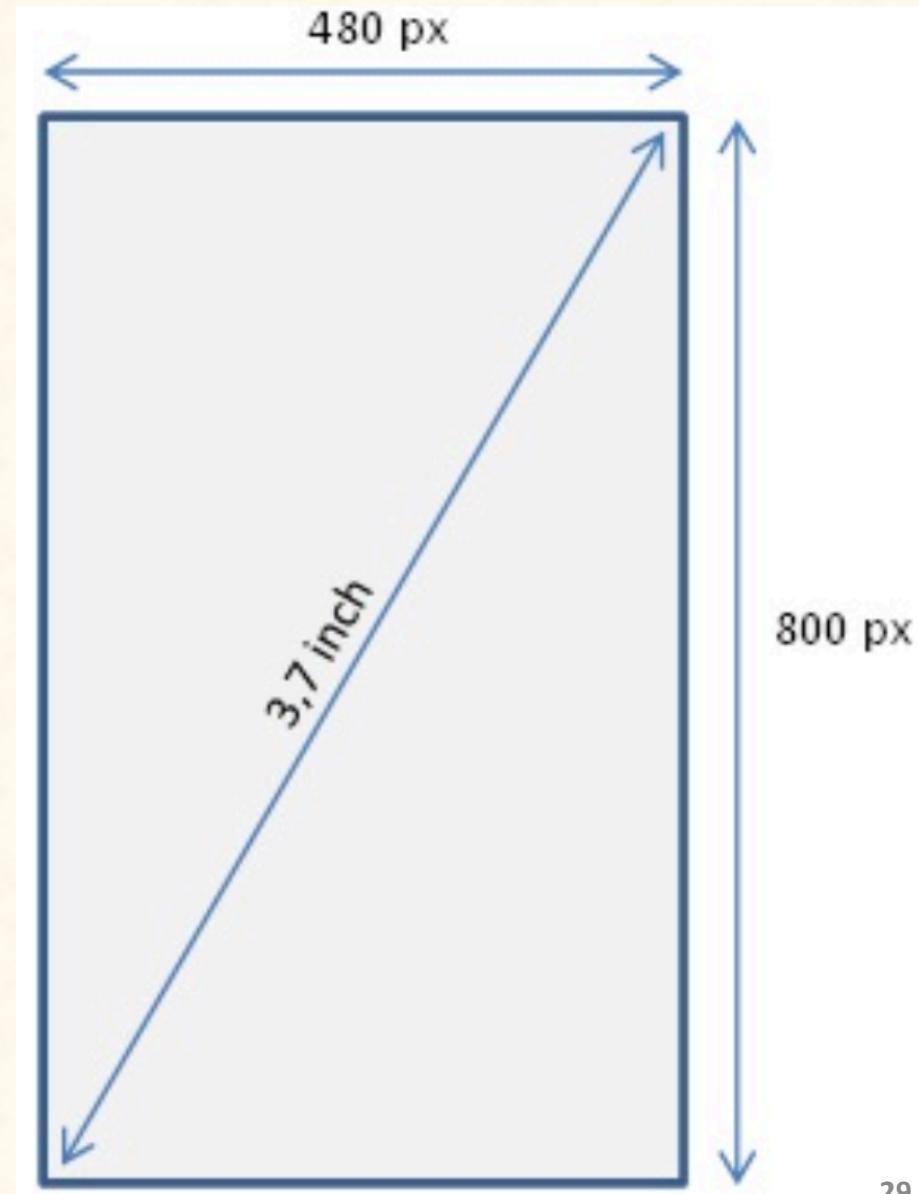
AbsoluteLayout

```
<?xml version="1.0" encoding="utf-8"?>
<AbsoluteLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_x="42dp"
        android:layout_y="62dp"
        android:text="Button">
    </Button>
    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_x="142dp"
        android:layout_y="131dp"
        android:text="TextView">
    </TextView>
    <CheckBox
        android:id="@+id/checkBox1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_x="55dp"
        android:layout_y="212dp"
        android:text="CheckBox">
    </CheckBox>
    <RadioButton
        android:id="@+id/radioButton1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_x="171dp"
        android:layout_y="318dp"
        android:text="RadioButton">
    </RadioButton>
</AbsoluteLayout>
```

View Parameter

View Parameter

- HTC Desfire Screen
- $c = 3,7 * \text{dpi}$
 $(3,7 * \text{dpi})^2 = 480^2 + 800^2$
 $\text{dpi}^2 = 870400 / 13,69 =$
63579
dpi = 252.



Bildschirm Maßeinheiten

Maßeinheit	Beschreibung
dp (oder dip)	Von der Auflösung unabhängige Bildpunkte (Density-independent Pixels). Diese Einheit verhält sich relativ zu 160 dpi. 160 dp sind immer ein Zoll, egal auf welchem Bildschirm. Diese Maßeinheit wird hauptsächlich für Maßangaben in Layouts verwendet. Diese Maßeinheit ist geräteunabhängig.
sp	Von der Skalierung unabhängige Pixel (Scale-independent Pixel). Diese Maßangaben verhalten sich wie die dp-Maßangaben, berücksichtigen aber den Schriftskalierungsfaktor, den der Benutzer eingestellt hat. Wird hauptsächlich für Schriftgrößen verwendet, sodass die aktuelle Bildschirmauflösung und die Benutzereinstellungen berücksichtigt werden. Diese Maßeinheit ist geräteunabhängig.
pt	Point, 1/72 eines Zolls. Diese Maßeinheit ist geräteunabhängig.
px	Pixel. Ein Pixel korrespondiert mit einem Bildpunkt des Bildschirms. Diese Maßeinheit ist geräteabhängig und deshalb nicht zur Verwendung empfohlen.
mm	Millimeter. Diese Maßeinheit ist geräteunabhängig.
in	Inches (Zoll). Diese Maßeinheit ist geräteunabhängig.

View Parameter

Misc	
Layout gravity	
Layout height	wrap_content
Layout margin	
Layout margin bottom	
Layout margin left	
Layout margin right	
Layout margin top	
Layout weight	
Layout width	wrap_content

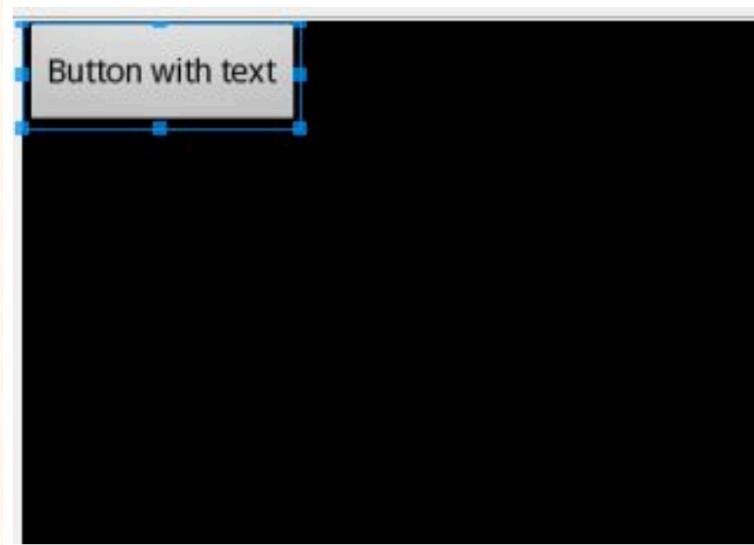
- **Layout width, Layout height:**
 - Absolute Größe in dp, sp, pt, px, mm, in
 - Constant als match_parent (fill_parent), wrap_content
- **Layout weight**
- **Layout gravity**
- **Layout margin**
- ...

Layout width



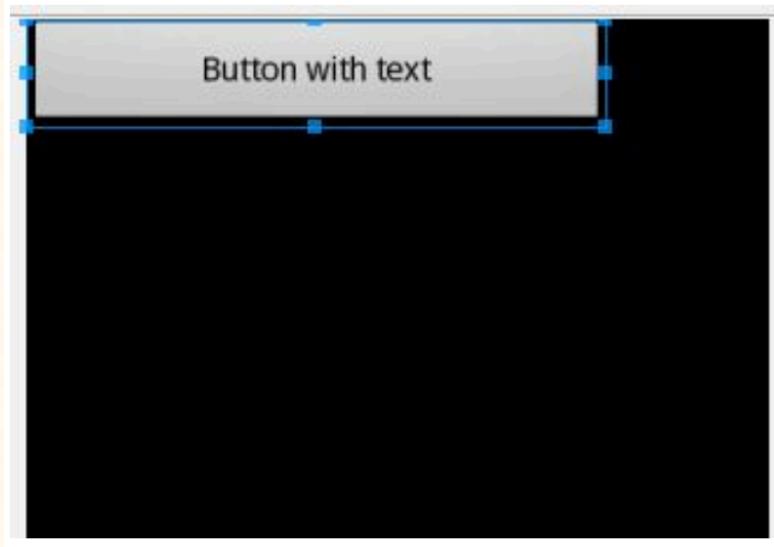
- in einer Layout-Datei:
 - LinearLayout: horizontal
 - Button
 - Layout width: **wrap_content**
 - Layout height: **wrap_content**
 - Text: **Button**

Layout width



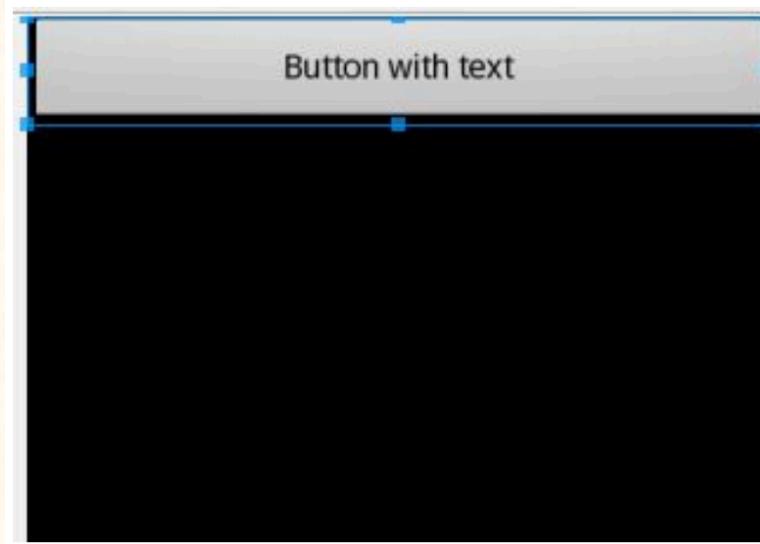
- **in der Layout-Datei:**
 - **LinearLayout: horizontal**
 - **Button**
 - Layout width: **wrap_content**
 - Layout height: **wrap_content**
 - Text: **Button with text**

Layout width



- **in der Layout-Datei:**
 - **LinearLayout: horizontal**
 - **Button**
 - **Layout width: 250 dp**
 - **Layout height: wrap_content**
 - **Text: Button with text**

Layout width



- **in der Layout-Datei:**
 - **LinearLayout: horizontal**
 - **Button**
 - Layout width: **match_parent**
 - Layout height: **wrap_content**
 - Text: **Button with text**

Layout weight



– Button

- Layout width: wrap_content
- Layout height: wrap_content
- Text: B1

– Button

- Layout width: wrap_content
- Layout height: wrap_content
- Text: B2

Layout weight



- **Button**
 - Layout width: wrap_content
 - Layout height: wrap_content
 - Text: B1
 - weight : 1

- **Button**
 - Layout width: wrap_content
 - Layout height: wrap_content
 - Text: B2
 - weight : 1

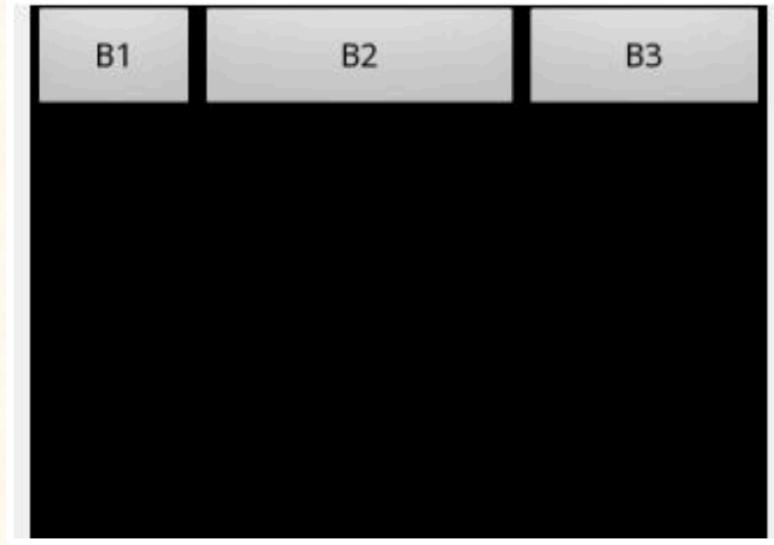
Layout weight



- **Button**
 - Layout width: wrap_content
 - Layout height: wrap_content
 - Text: B1
 - weight : 1

- **Button**
 - Layout width: wrap_content
 - Layout height: wrap_content
 - Text: B2
 - weight : 3

Layout weight



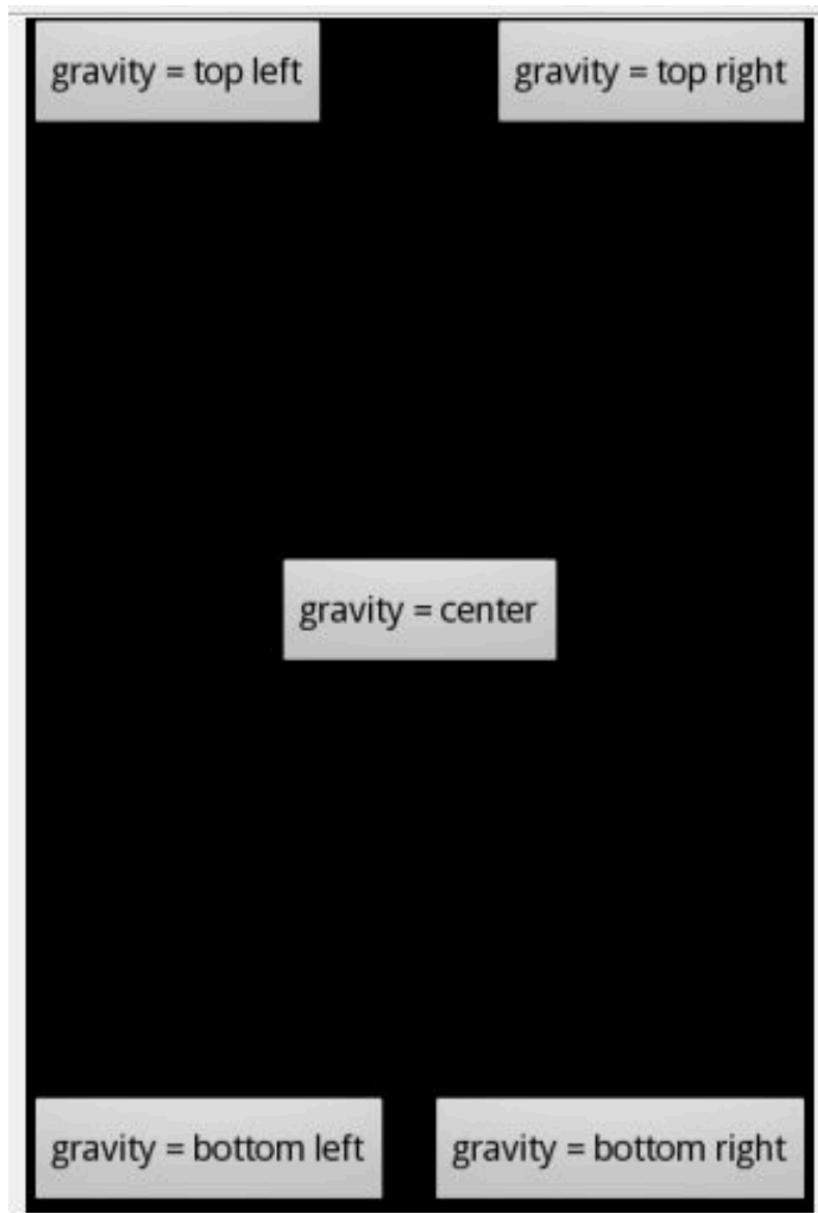
- **Button**
 - Layout width: wrap_content
 - Layout height: wrap_content
 - Text: B1
 - weight : 1
- **Button**
 - Layout width: wrap_content
 - Layout height: wrap_content
 - Text: B2
 - weight : 3
- **Button**
 - Layout width: wrap_content
 - Layout height: wrap_content
 - Text: B3
 - weight : 2

Layout weight



- **Button**
 - Layout width: wrap_content
 - Layout height: wrap_content
 - Text: B1
 - weight : 1
- **Button**
 - Layout width: wrap_content
 - Layout height: wrap_content
 - Text: B2
 - weight : 0
- **Button**
 - Layout width: wrap_content
 - Layout height: wrap_content
 - Text: B3
 - weight : 0

Layout gravity



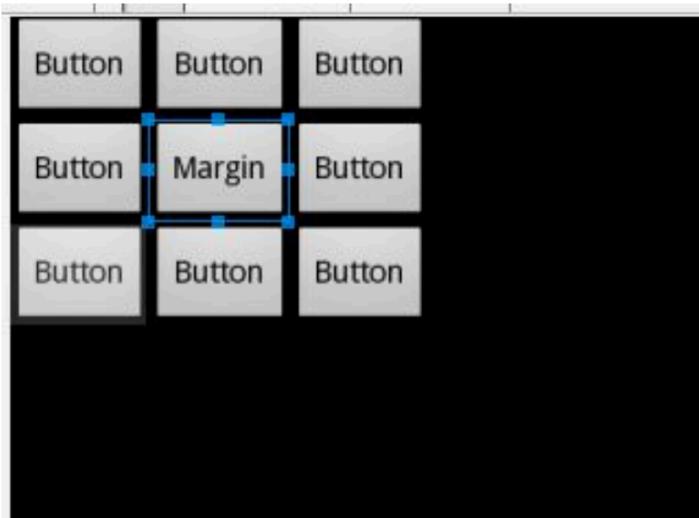
- **LinearLayout as a Parent**
- **FrameLayout as a Child (alle Objekte werden default oben links positioniert)**
- **5 Buttons in a FrameLayout:**
 - layout_width="wrap_content"
 - layout_height="wrap_content"
 - layout_gravity="top|left"
(layout_gravity="top|right")
(layout_gravity="bottom|left")
(layout_gravity="bottom|right")
(layout_gravity="center")

Layout gravity

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <FrameLayout
        android:id="@+id/frameLayout1"
        android:layout_width="match_parent"
        android:layout_height="match_parent">
        <Button
            android:id="@+id/button1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="top|left"
            android:text="gravity = top left">
        </Button>
        <Button
            android:id="@+id/button2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="top|right"
            android:text="gravity = top right">
        </Button>
    </FrameLayout>
</LinearLayout>
```

```
<Button
    android:id="@+id/button3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="bottom|left"
    android:text="gravity = bottom left">
</Button>
<Button
    android:id="@+id/button4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="bottom|right"
    android:text="gravity = bottom right">
</Button>
<Button
    android:id="@+id/button5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:text="gravity = center">
</Button>
</FrameLayout>
</LinearLayout>
```

Layout margin



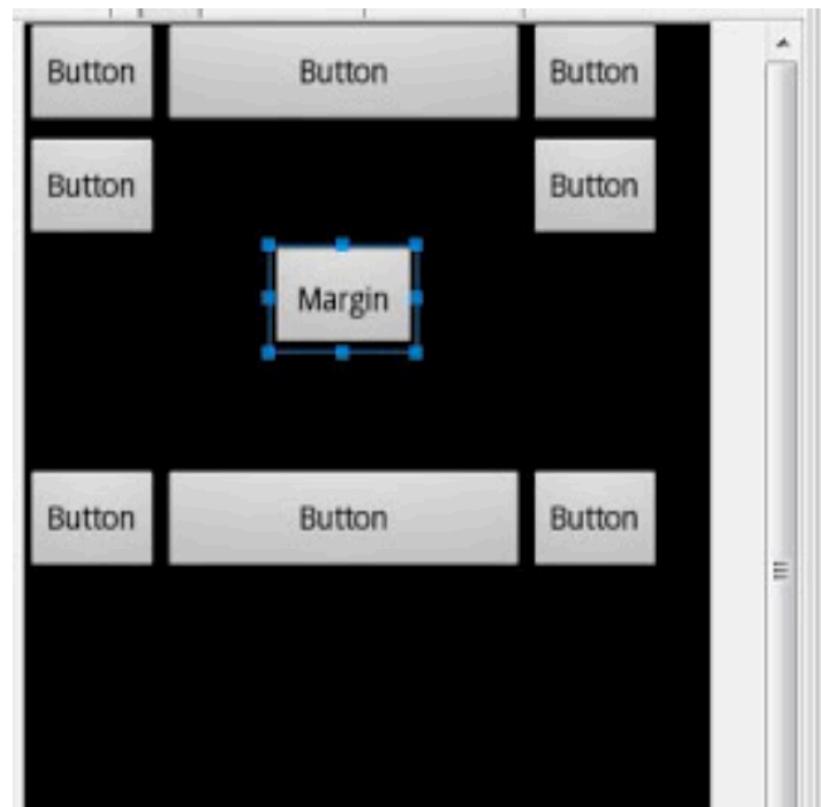
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TableLayout
        android:layout_height="wrap_content"
        android:layout_width="match_parent"
        android:id="@+id/tableLayout1">
        <TableRow
            android:id="@+id/tableRow1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content">
            <Button
                android:text="Button"
                android:id="@+id/button1"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content">
            </Button>
            <Button
                android:text="Button"
                android:id="@+id/button2"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content">
            </Button>
            <Button
                android:text="Button"
                android:id="@+id/button3"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content">
            </Button>
        </TableRow>
        <TableRow
            android:id="@+id/tableRow2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content">
            <Button
                android:text="Button"
                android:id="@+id/button4"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content">
            </Button>
            <Button
                android:text="Button"
                android:id="@+id/button5"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content">
            </Button>
            <Button
                android:text="Button"
                android:id="@+id/button6"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content">
            </Button>
        </TableRow>
    </TableLayout>
</LinearLayout>
```

Layout margin

```
<TableRow
    android:id="@+id/tableRow2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content">
    <Button
        android:text="Button"
        android:id="@+id/button4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content">
    </Button>
    <Button
        android:text="Margin"
        android:id="@+id/button5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content">
    </Button>
    <Button
        android:text="Button"
        android:id="@+id/button6"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content">
    </Button>
</TableRow>
```

```
<TableRow
    android:id="@+id/tableRow3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content">
    <Button
        android:text="Button"
        android:id="@+id/button7"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content">
    </Button>
    <Button
        android:text="Button"
        android:id="@+id/button8"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content">
    </Button>
    <Button
        android:text="Button"
        android:id="@+id/button9"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content">
    </Button>
</TableRow>
</TableLayout>
</LinearLayout>
```

Layout margin



button3
tableRow2
button4
button5 - "Margin"
button6
tableRow3
button7
button8
button9

Text scale X
Text select handle
Text select handle left
Text select handle right

Text size

Text style

Typeface

Visibility

Width

Misc

Layout gravity

Layout height wrap_content

Layout margin 50dp

Layout margin bottom

Layout margin left

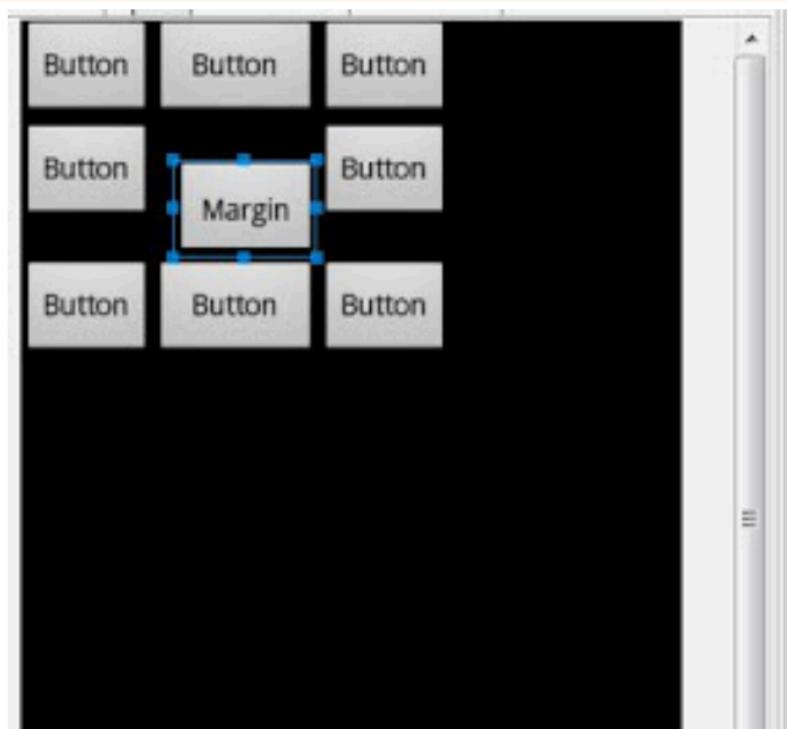
Layout margin right

Layout margin top

Layout weight

Layout width wrap_content

Layout margin



button3	Text scale X
tableRow2	Text select handle
button4	Text select handle left
button5 - "Margin"	Text select handle right
button6	Text size
tableRow3	Text style
button7	Typeface
button8	Visibility
button9	Width
▲ Misc	
Layout gravity	
Layout height	wrap_content
Layout margin	
Layout margin bottom	
Layout margin left	10dp
Layout margin right	
Layout margin top	20dp
Layout weight	
Layout width	wrap_content

Layout margin

The screenshot shows the Android Studio layout editor. On the left, a 3x3 grid of buttons is displayed. The central button in the second row and second column has a white background with the word "Margin" written on it. The entire grid is set against a black background. To the right of the layout is a detailed properties panel.

Properties Panel:

- Text scale X
- Text select handle
- Text select handle left
- Text select handle right
- Text size
- Text style
- Typeface
- Visibility
- Width
- Misc**
- Layout gravity
- Layout height `wrap_content`
- Layout margin
- Layout margin bottom `40dp`**
- Layout margin left
- Layout margin right `30dp`
- Layout margin top
- Layout weight
- Layout width `wrap_content`

Fragen?

Android App UI