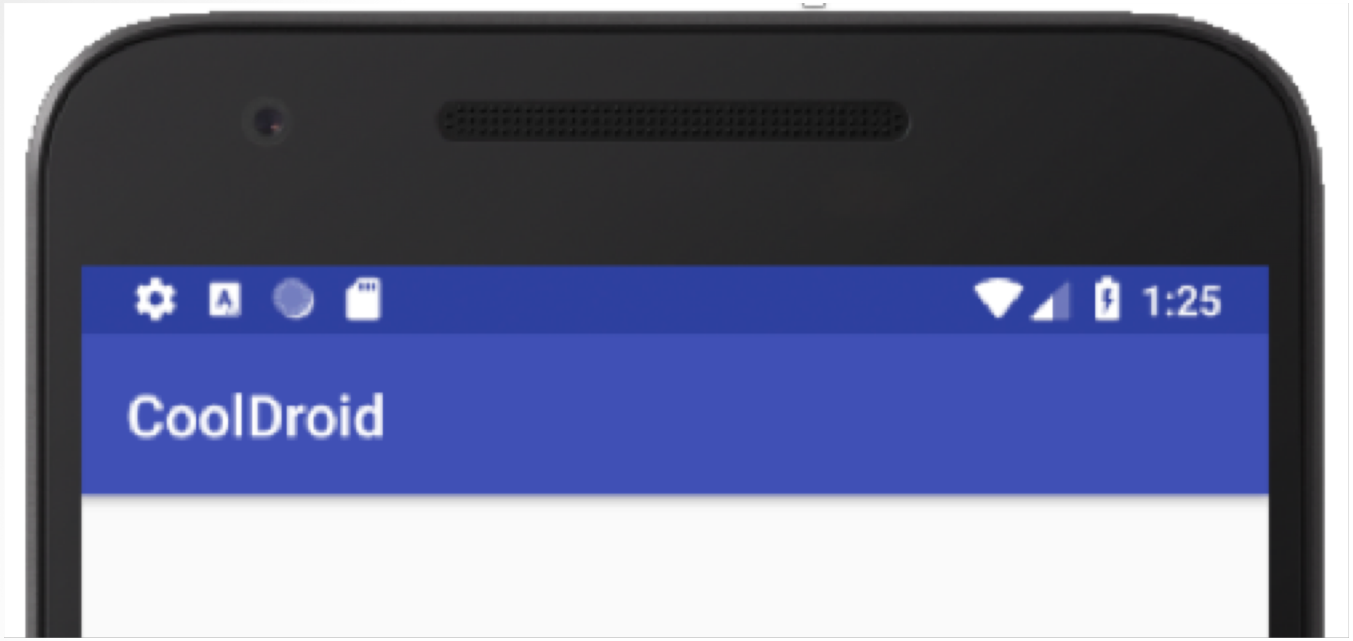
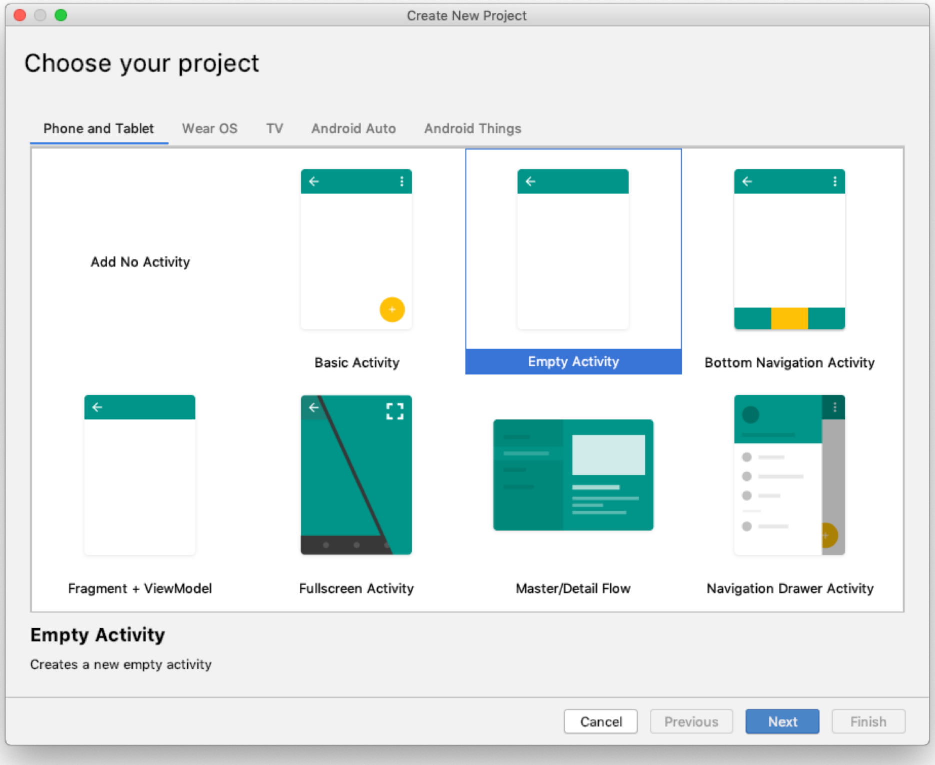
My First Android Project



Übung

Typ der Haupt-Activity festlegen

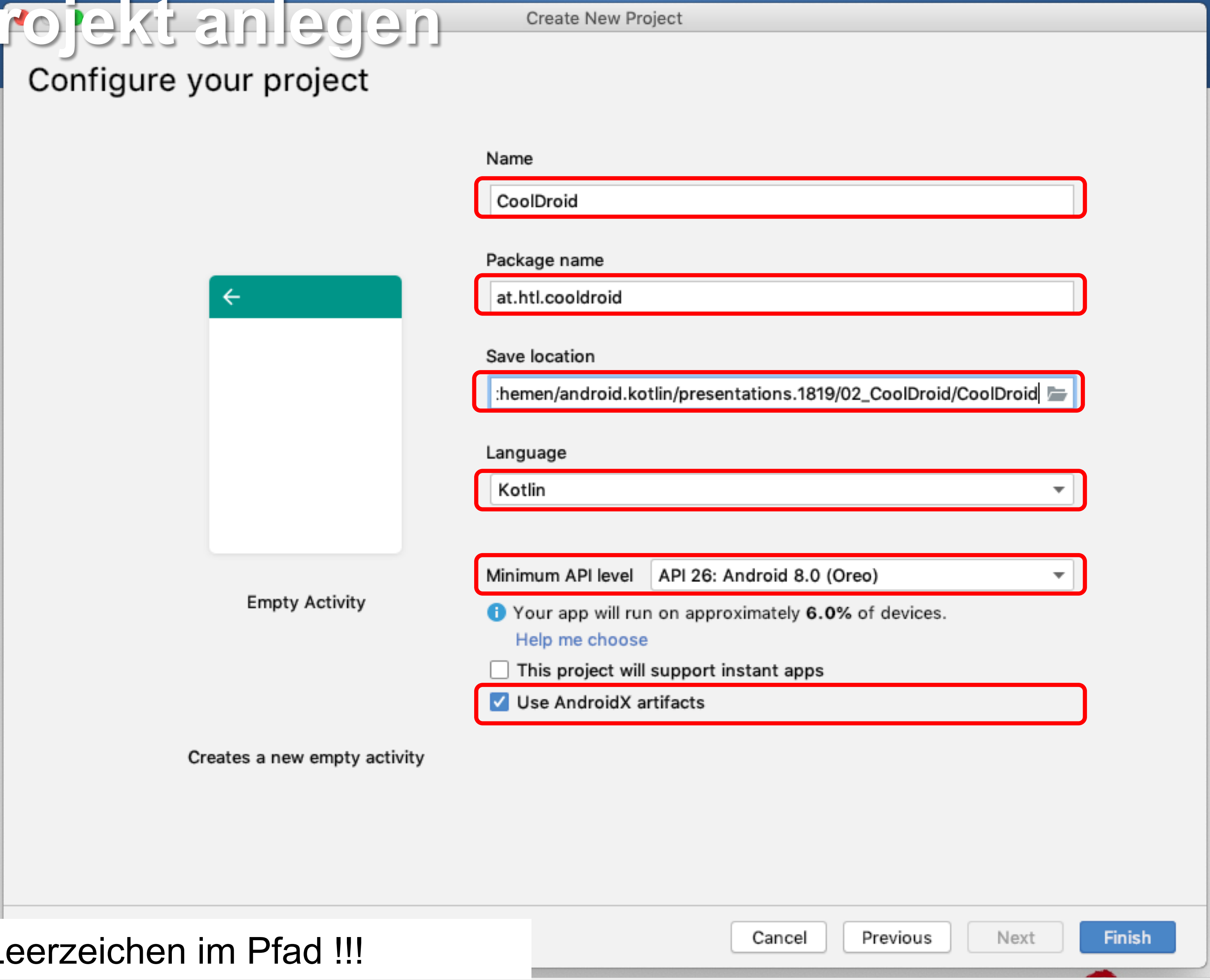
- Activity ist DIE GUI-Komponente in Android
- Templates erleichtern den Einstieg
- Wir verwenden eine Empty Activity für das erste Beispiel



Empty Activity

Creates a new empty activity

Projekt anlegen



Create New Project

Configure your project

Name

CoolDroid

Package name

at.htl.cooldroid

Save location

:hemen/android.kotlin/presentations.1819/02_CoolDroid/CoolDroid

Language

Kotlin

Minimum API level API 26: Android 8.0 (Oreo)

Your app will run on approximately **6.0%** of devices.

Help me choose

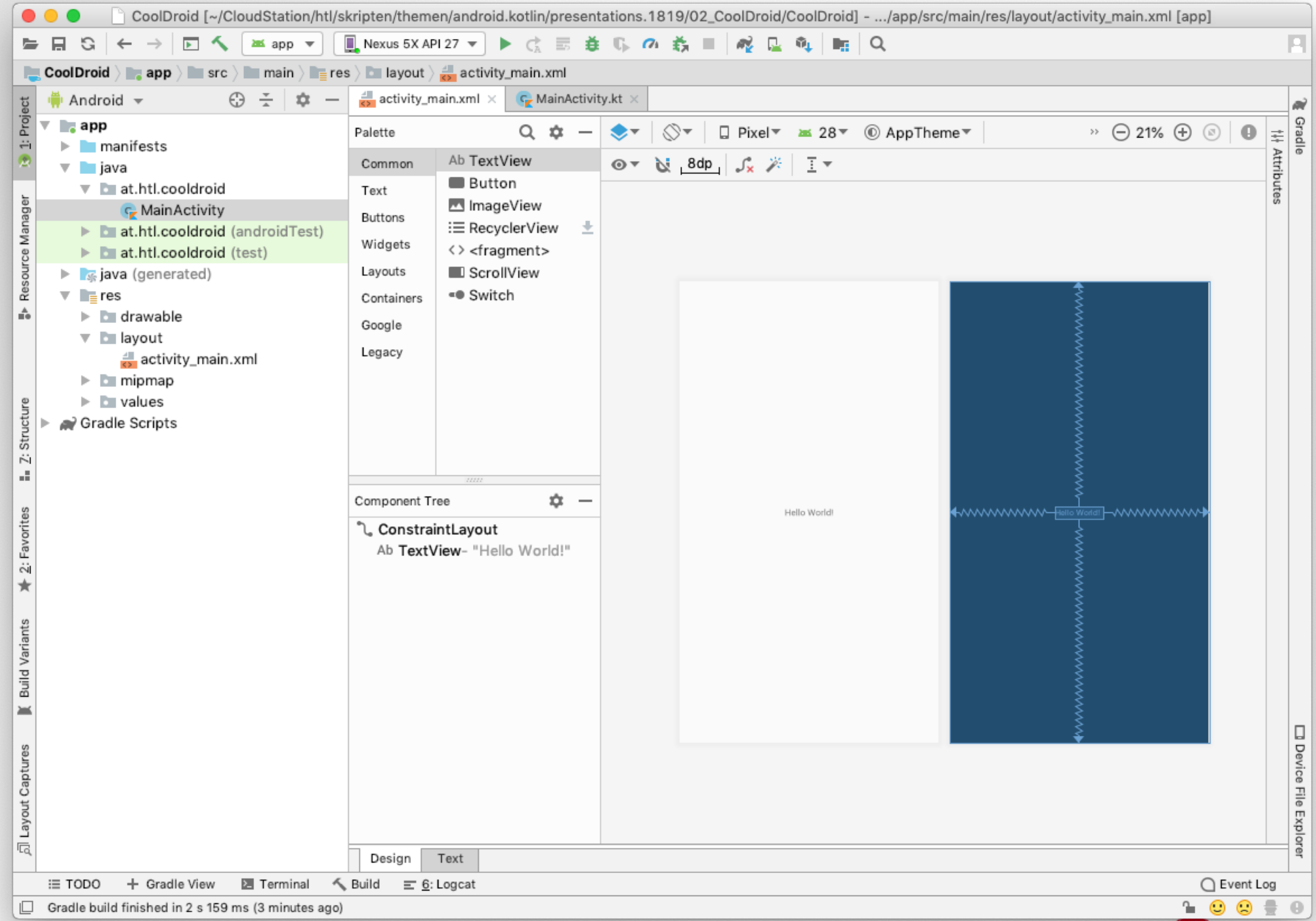
☐ This project will support instant apps

☑ Use AndroidX artifacts

Keine Leerzeichen im Pfad !!!

Your project location contains whitespace. This can cause problems on some platforms and is not recommended.

Projekt in der IDE

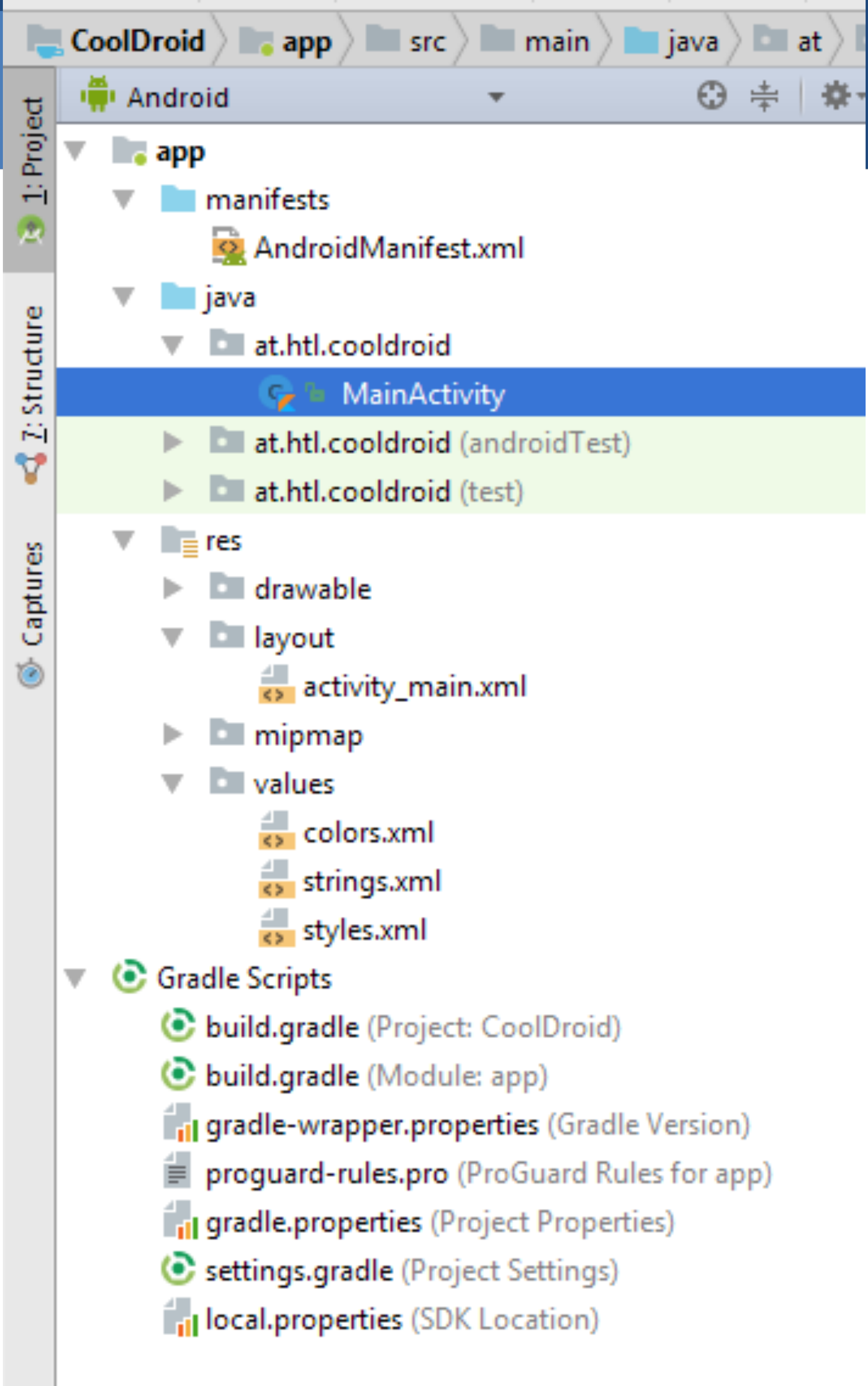


CoolDroid [~/CloudStation/htl/skripten/themen/android.kotlin/presentations.1819/02_CoolDroid/CoolDroid] - .../app/src/main/res/layout/activity_main.xml [app]

Gradle build finished in 2 s 159 ms (3 minutes ago)

Project-Window Alt-1

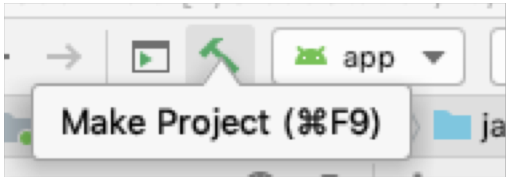
- Java-Quelltexte
- Ressourcen
- Layouts
- Menüs
- Manifest
- …

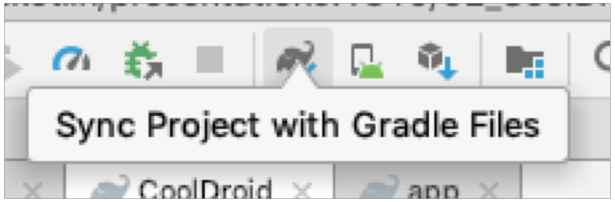


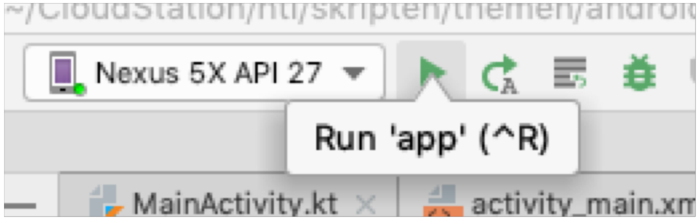
```
▼ app
    ▼ manifests
        AndroidManifest.xml
    ▼ java
        ▼ at.htl.cooldroid
            MainActivity
        ▶ at.htl.cooldroid (androidTest)
        ▶ at.htl.cooldroid (test)
    ▼ res
        ▶ drawable
        ▼ layout
            activity_main.xml
        ▶ mipmap
        ▼ values
            colors.xml
            strings.xml
            styles.xml
▼ Gradle Scripts
    build.gradle (Project: CoolDroid)
    build.gradle (Module: app)
    gradle-wrapper.properties (Gradle Version)
    proguard-rules.pro (ProGuard Rules for app)
    gradle.properties (Project Properties)
    settings.gradle (Project Settings)
    local.properties (SDK Location)
```

Wichtige Symbole

- Make Project
- Sync
- Run

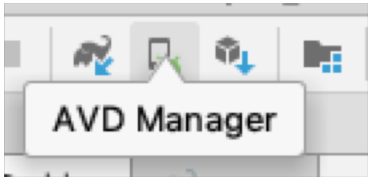

Make Project (⌘F9)


Sync Project with Gradle Files

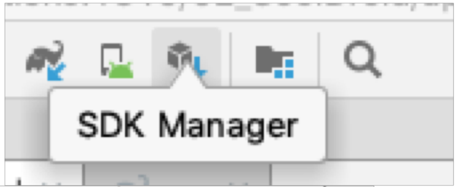

Run 'app' (^R)

Zugriff auf Tools

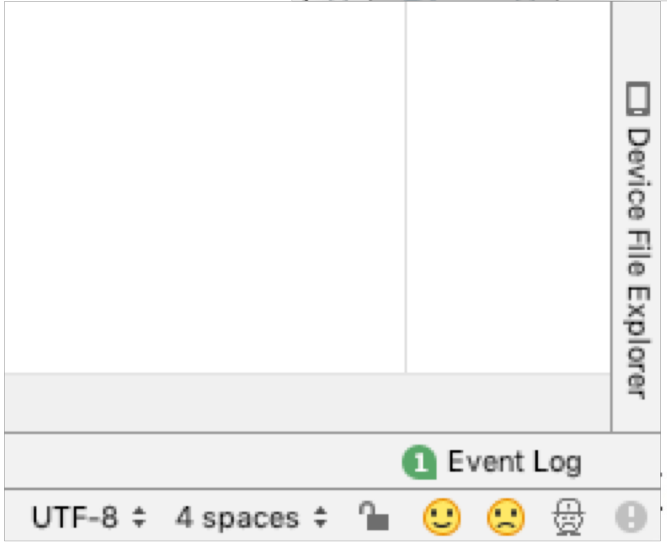
- Verwaltung der Android-Virtual-Devices (Emulatoren)



- SDK mit allen Tools und Android-Versionen

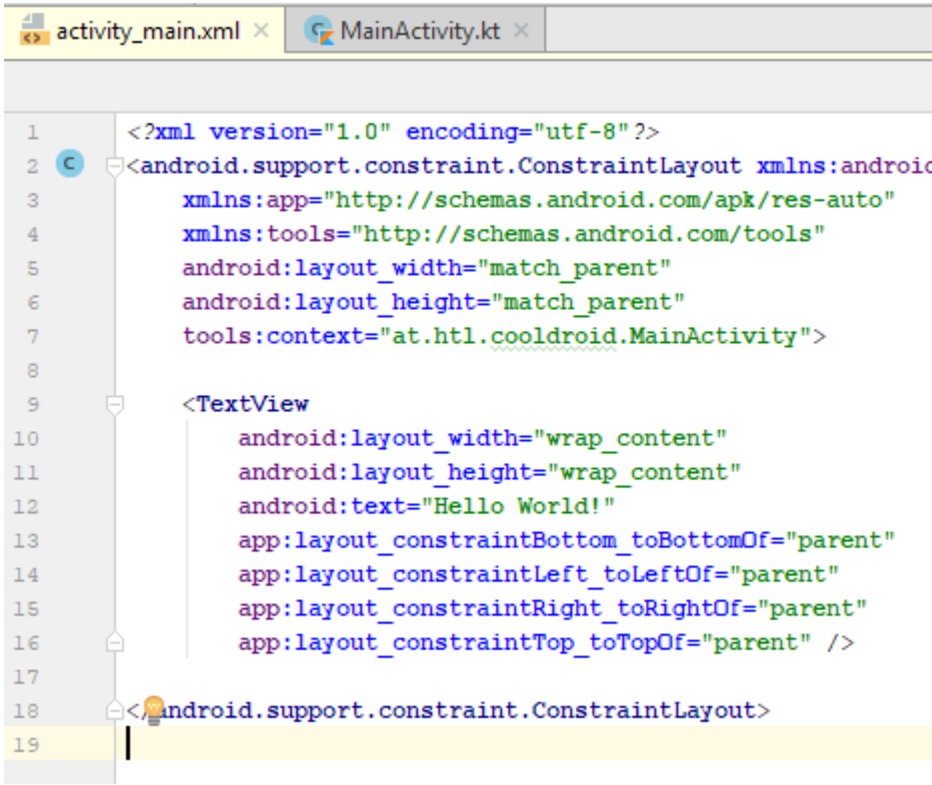


- Zugriff auf das Device (Filetransfer, ….)



Layout der Activity in XML

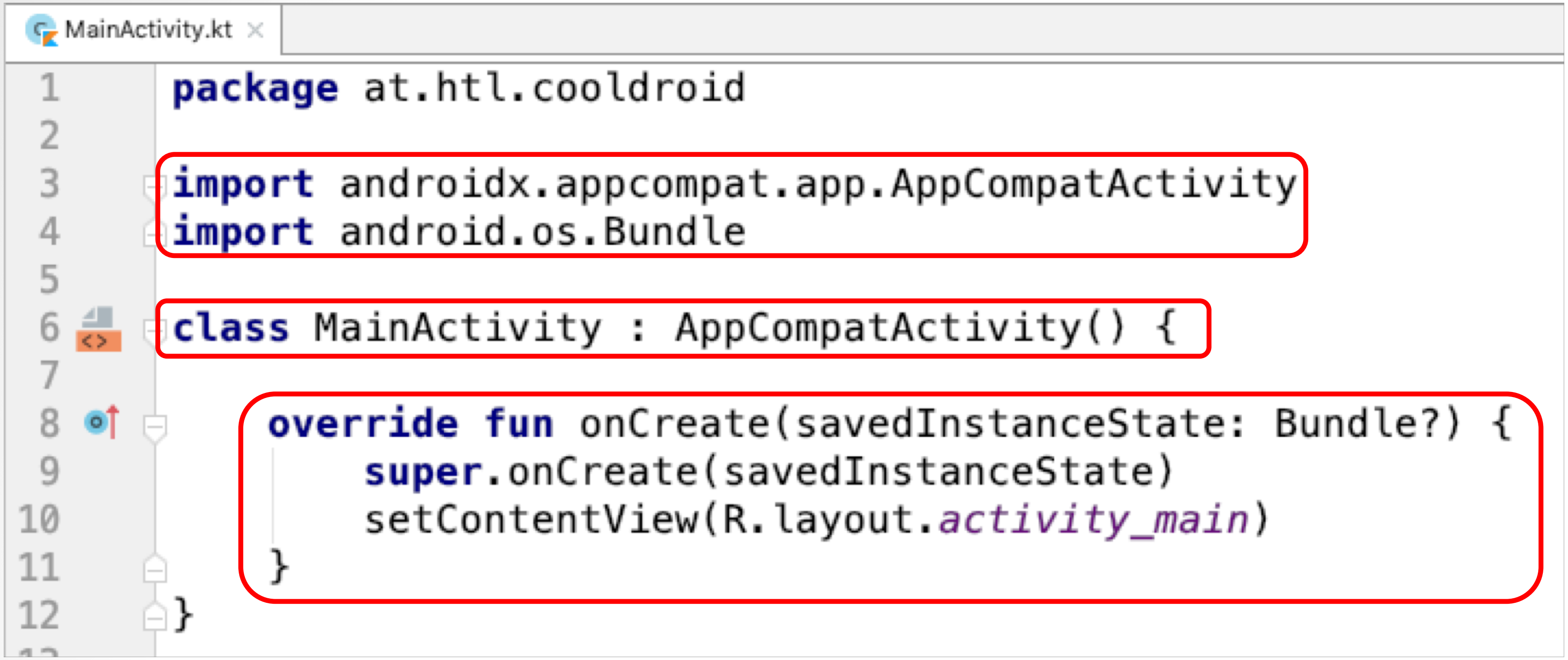
- Strikte Trennung zwischen Design und Code
- MVC
  - Design in XML definiert (View)
  - Javacode der Activity fungiert als Controller
  - Daten werden in Entitätsklassen verwaltet (Model)



```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout xmlns:androi
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="at.htl.cooldroid.MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</android.support.constraint.ConstraintLayout>

```

Code der Activity

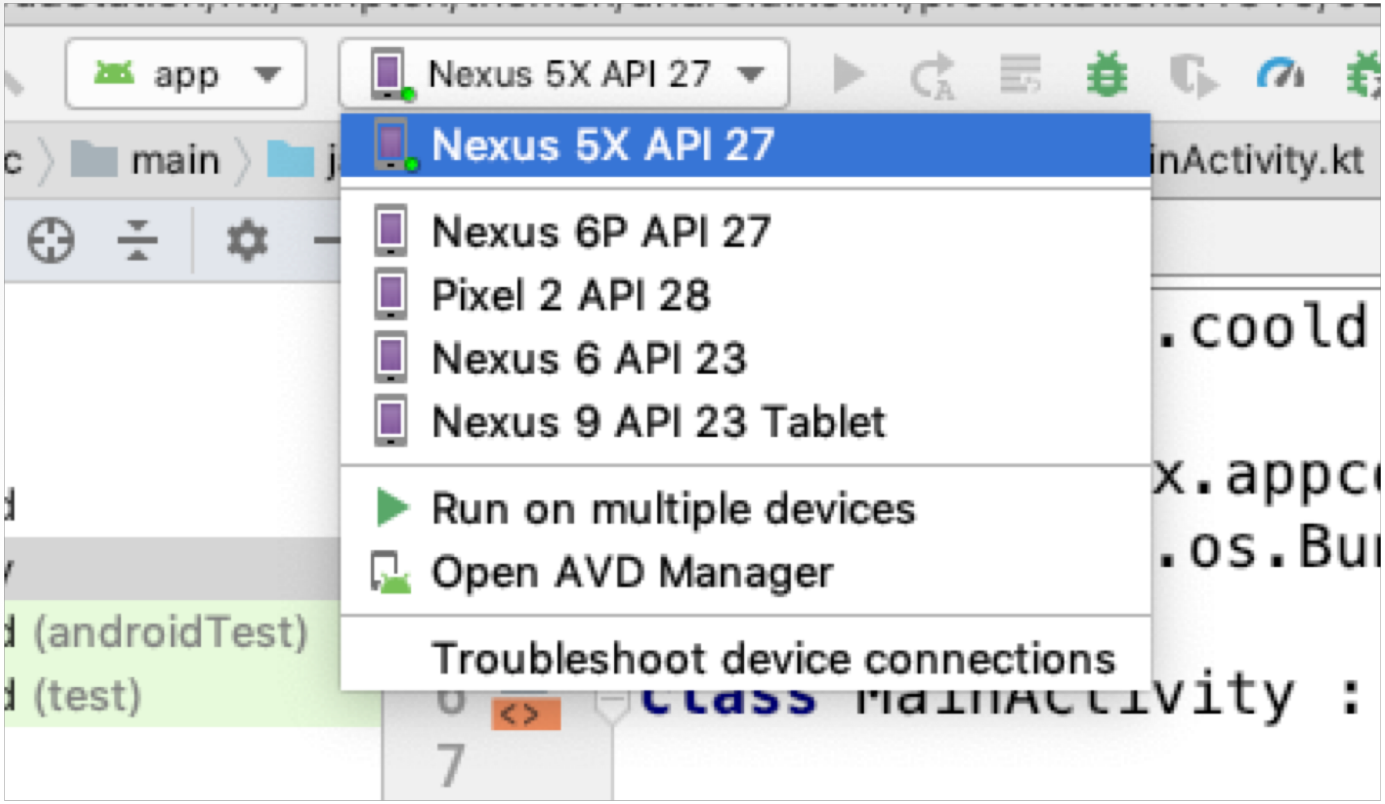
- Verwendet mächtige Funktionalitäten von Basisklassen
  - Deltaprogrammierung



```kotlin
package at.htl.cooldroid

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle

class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
    }
}
```

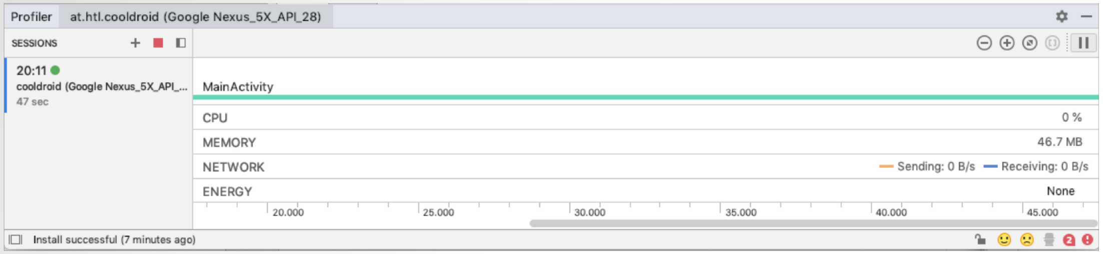
App laufen lassen

- Z.B. auf Handy per USB
- Sonst im Emulator



HTL LEONDING

Screenshot des Handys → DeviceMonitor



Profiler at.htl.cooldroid (Google Nexus_5X_API_28)

SESSIONS

20:11 ●
cooldroid (Google Nexus_5X_API_...
47 sec

MainActivity

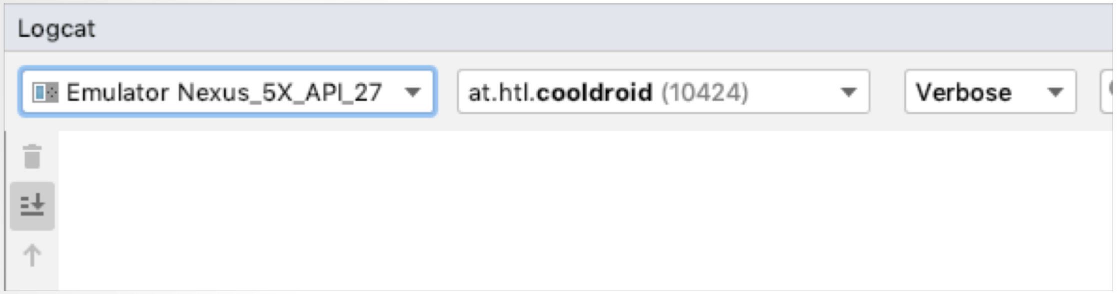
CPU 0 %

MEMORY 46.7 MB

NETWORK — Sending: 0 B/s — Receiving: 0 B/s

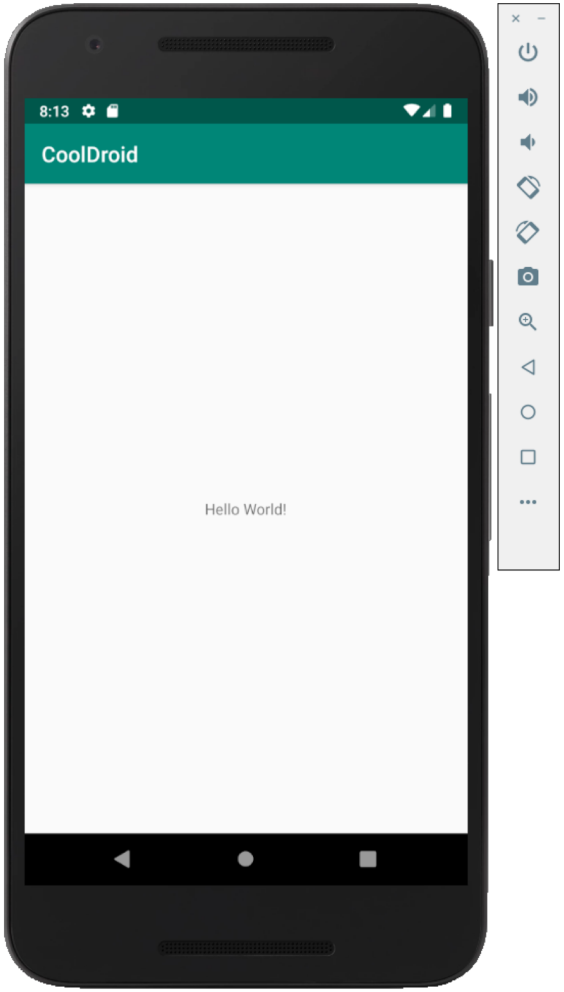
ENERGY None

Install successful (7 minutes ago)



Logcat

Emulator Nexus_5X_API_27 ▾ | at.htl.cooldroid (10424) ▾ | Verbose ▾



8:13

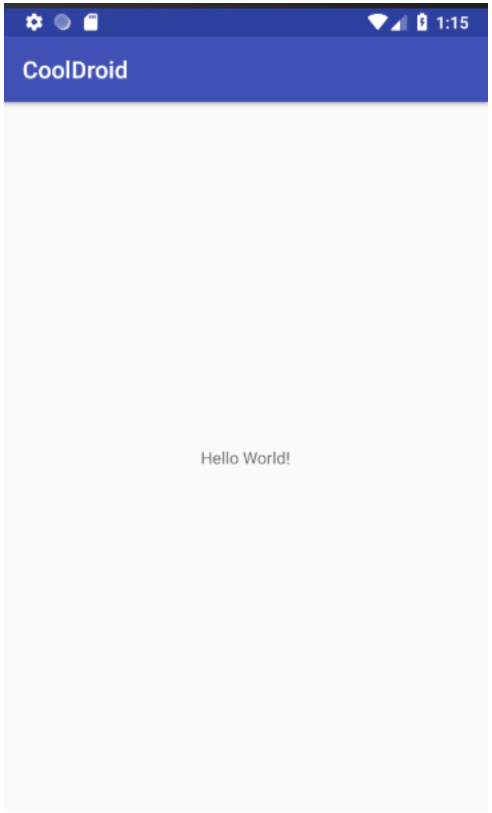
CoolDroid

Hello World!

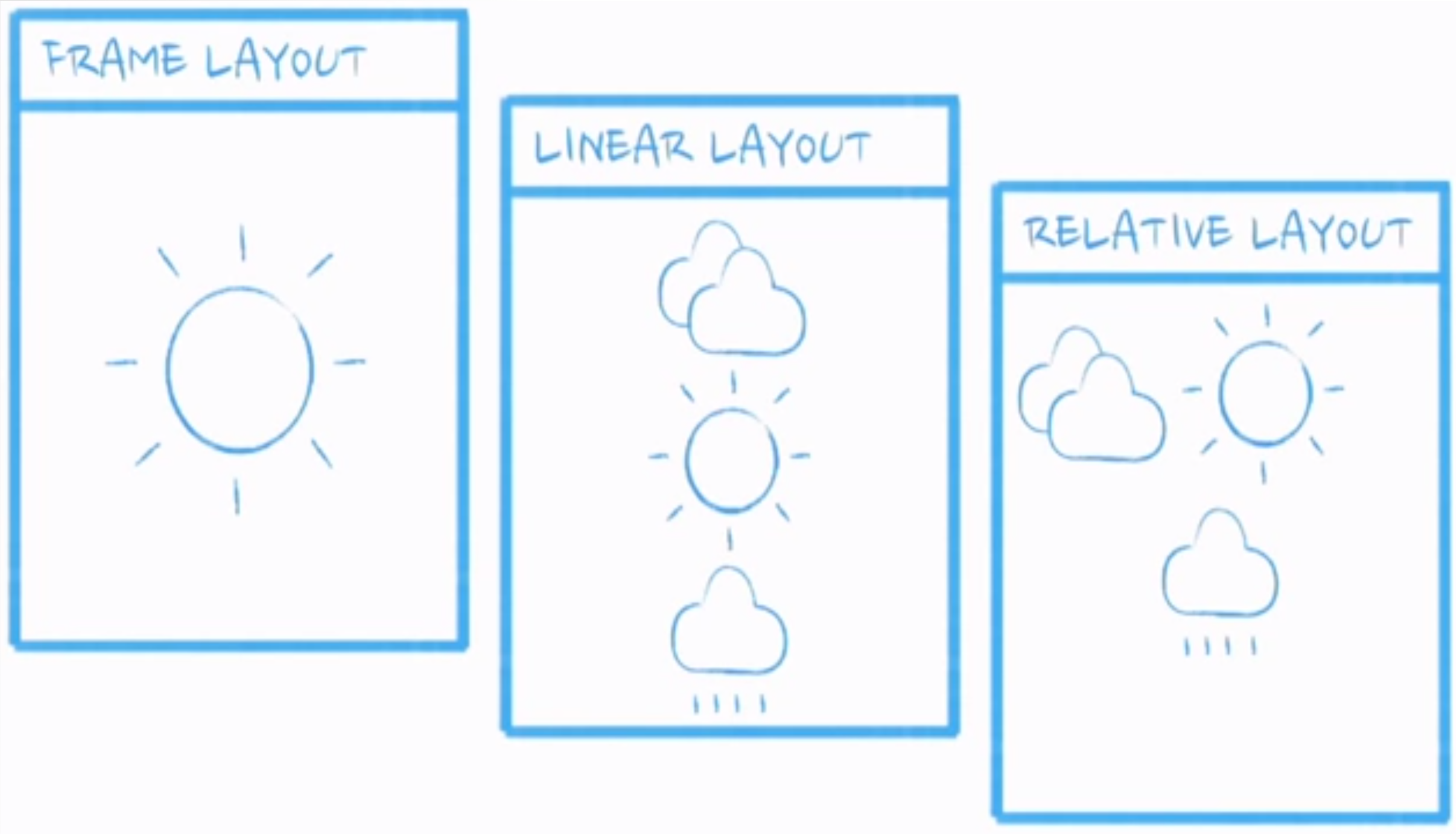
Emulator

- Hyper-V darf nicht aktiv sein
- Haxm installieren
- Emulator läuft halbwegs flott

- EMULATOR NICHT BEENDEN, SONST DAUERT DER PROGRAMMSTART LANGE



Layouts – Kurzer Überblick

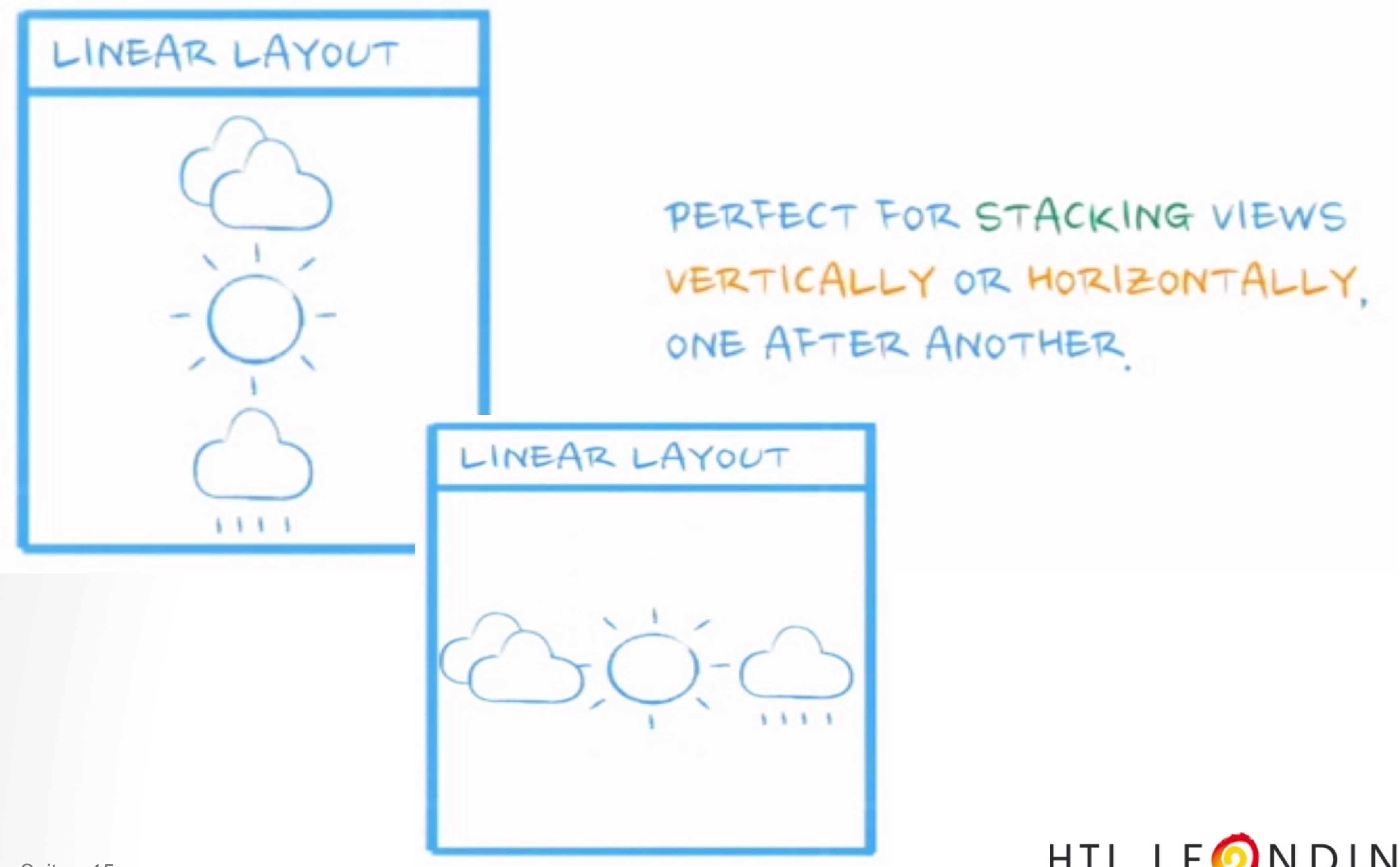


Layouts - FrameLayout

FRAME LAYOUT

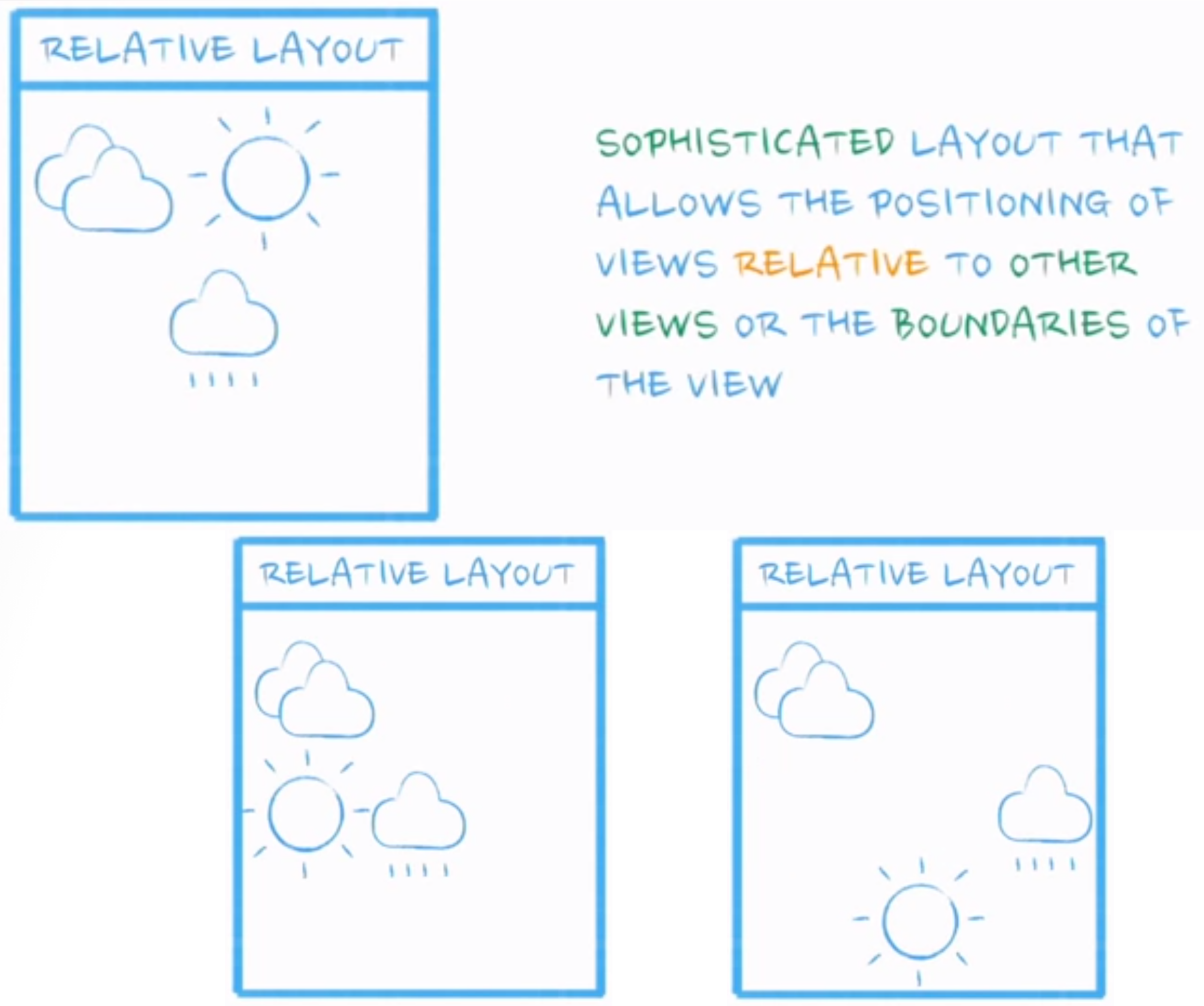
USEFUL FOR SIMPLE LAYOUTS. WITH A SINGLE VIEW OR STACK OR VIEWS. VIEWS ARE ALL ALIGNED AGAINST THE FRAME BOUNDARIES ONLY

Layout - LinearLayout



PERFECT FOR STACKING VIEWS VERTICALLY OR HORIZONTALLY, ONE AFTER ANOTHER.

Layout - RelativeLayout



SOPHISTICATED LAYOUT THAT ALLOWS THE POSITIONING OF VIEWS RELATIVE TO OTHER VIEWS OR THE BOUNDARIES OF THE VIEW

NEU: Constraint-Layout

- Flaches Layout ➔ performanter als verschachtelte Layouts
- Constraints beschreiben Beziehungen zwischen den Views

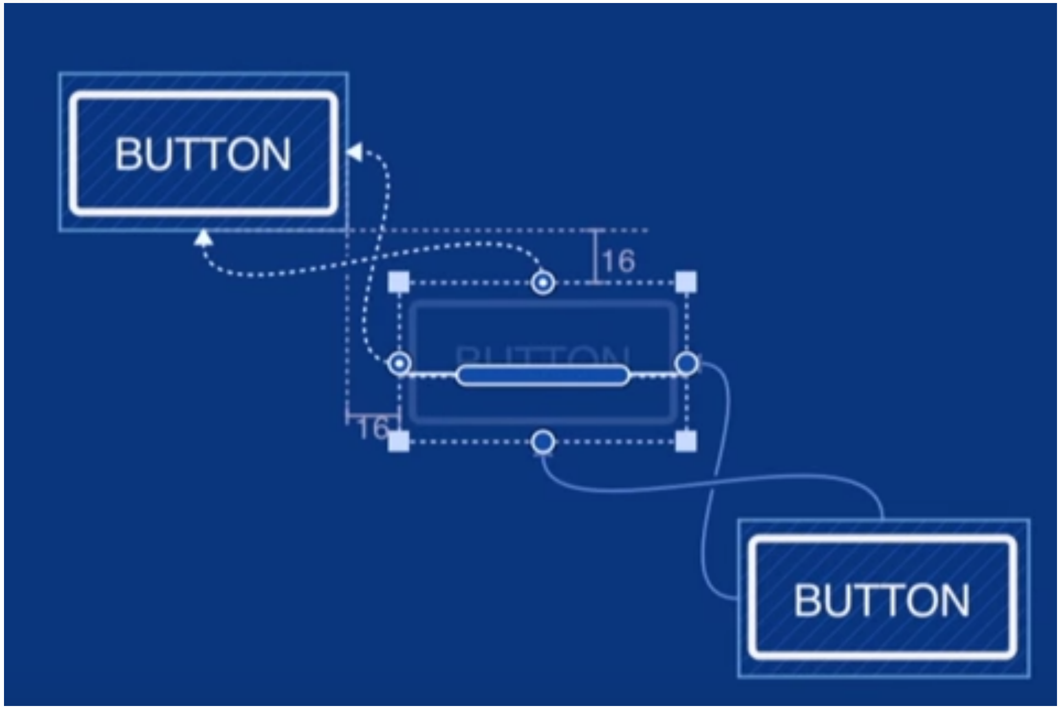
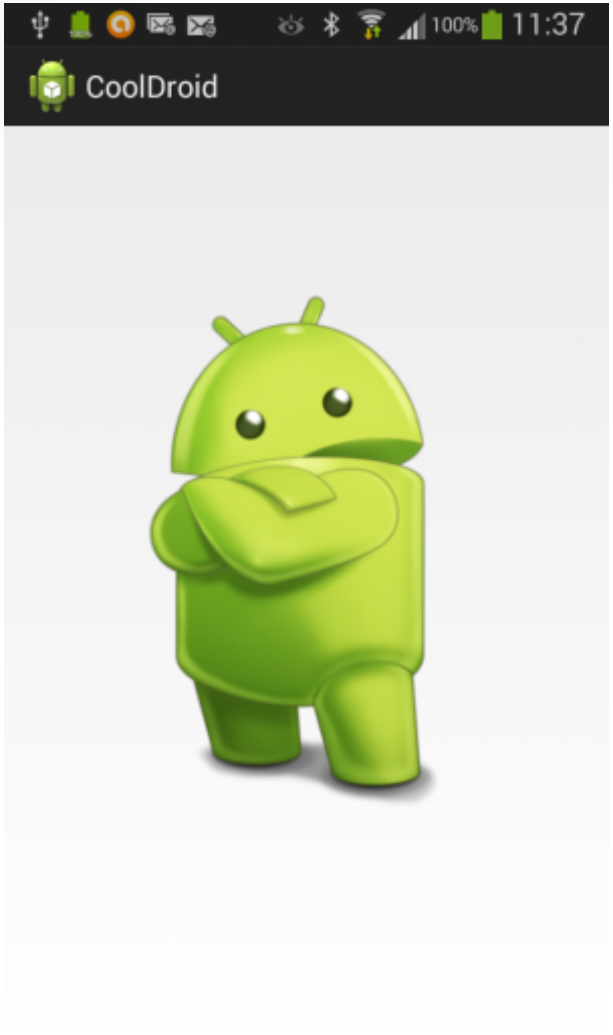


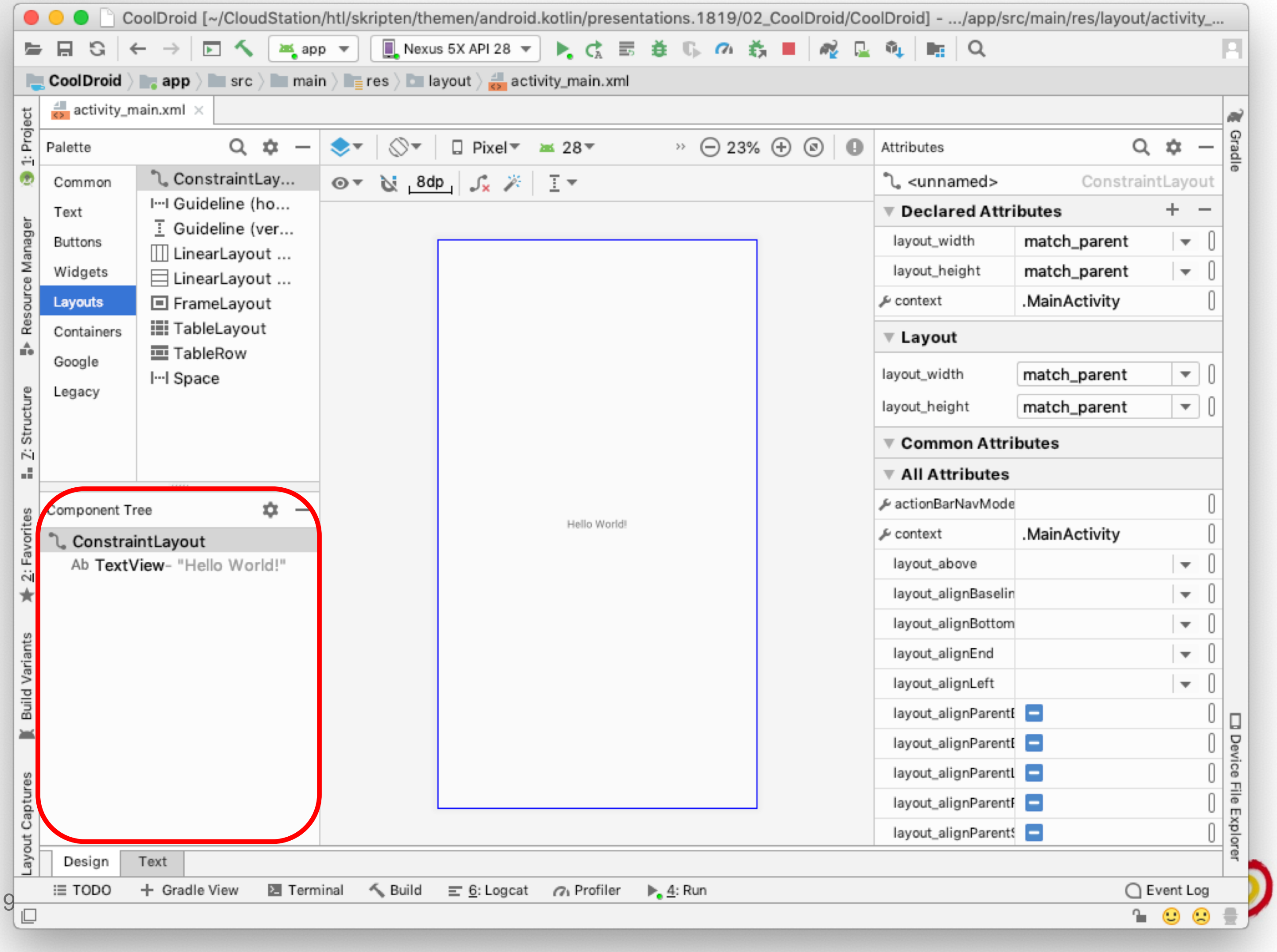
Bild anzeigen lassen



Übung

Arbeiten im Design-Mode

- Component Tree für guten Überblick



CoolDroid [~/CloudStation/htl/skripten/themen/android.kotlin/presentations.1819/02_CoolDroid/CoolDroid] - .../app/src/main/res/layout/activity_...

CoolDroid 〉 app 〉 src 〉 main 〉 res 〉 layout 〉 activity_main.xml

Attributes

<unnamed> ConstraintLayout

Declared Attributes

layout_width	match_parent
layout_height	match_parent
context	.MainActivity

Layout

layout_width	match_parent
layout_height	match_parent

Common Attributes

All Attributes

actionBarNavMode	
context	.MainActivity
layout_above	
layout_alignBaselin	
layout_alignBottom	
layout_alignEnd	
layout_alignLeft	
layout_alignParentI	
layout_alignParentE	
layout_alignParentL	
layout_alignParentR	
layout_alignParentS	

Component Tree

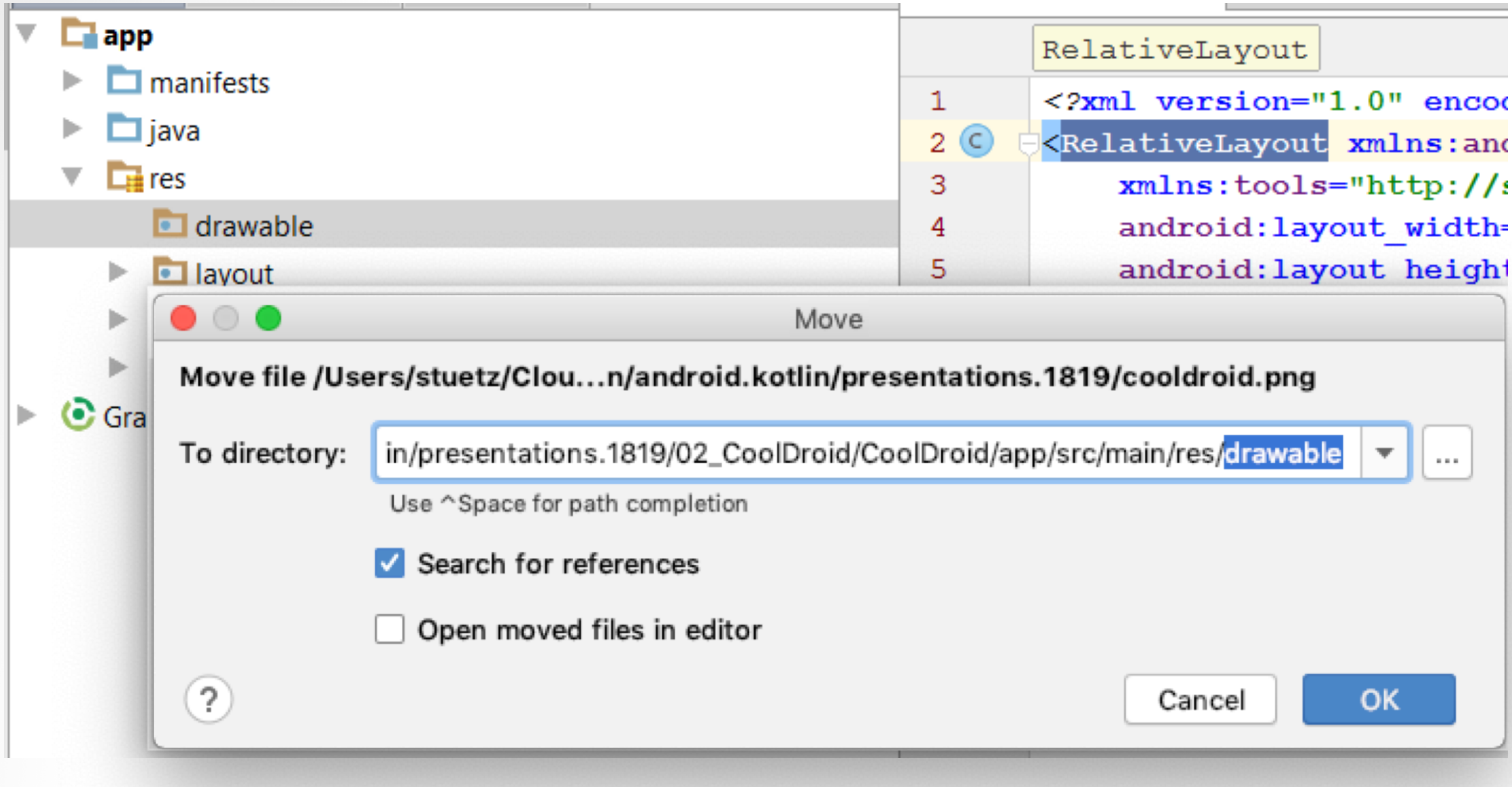
- ConstraintLayout
  - Ab TextView- "Hello World!"

Bild einfügen

1. Bild in das Ressourcenverzeichnis „**drawable**“ kopieren (mit Drag and Drop vom FileExplorer kopieren)
2. ImageView - Element ins Layout einfügen
3. Ressource an ImageView binden

Datei über Copy & Paste in Verzeichnis

- Strg-C + Strg-V
- Drag and Drop funktioniert nicht



```
<?xml version="1.0" encod
<RelativeLayout xmlns:and
    xmlns:tools="http://s
    android:layout_width=
    android:layout_height
```

Move

Move file /Users/stuetz/Clou...n/android.kotlin/presentations.1819/cooldroid.png

To directory: in/presentations.1819/02_CoolDroid/CoolDroid/app/src/main/res/drawable

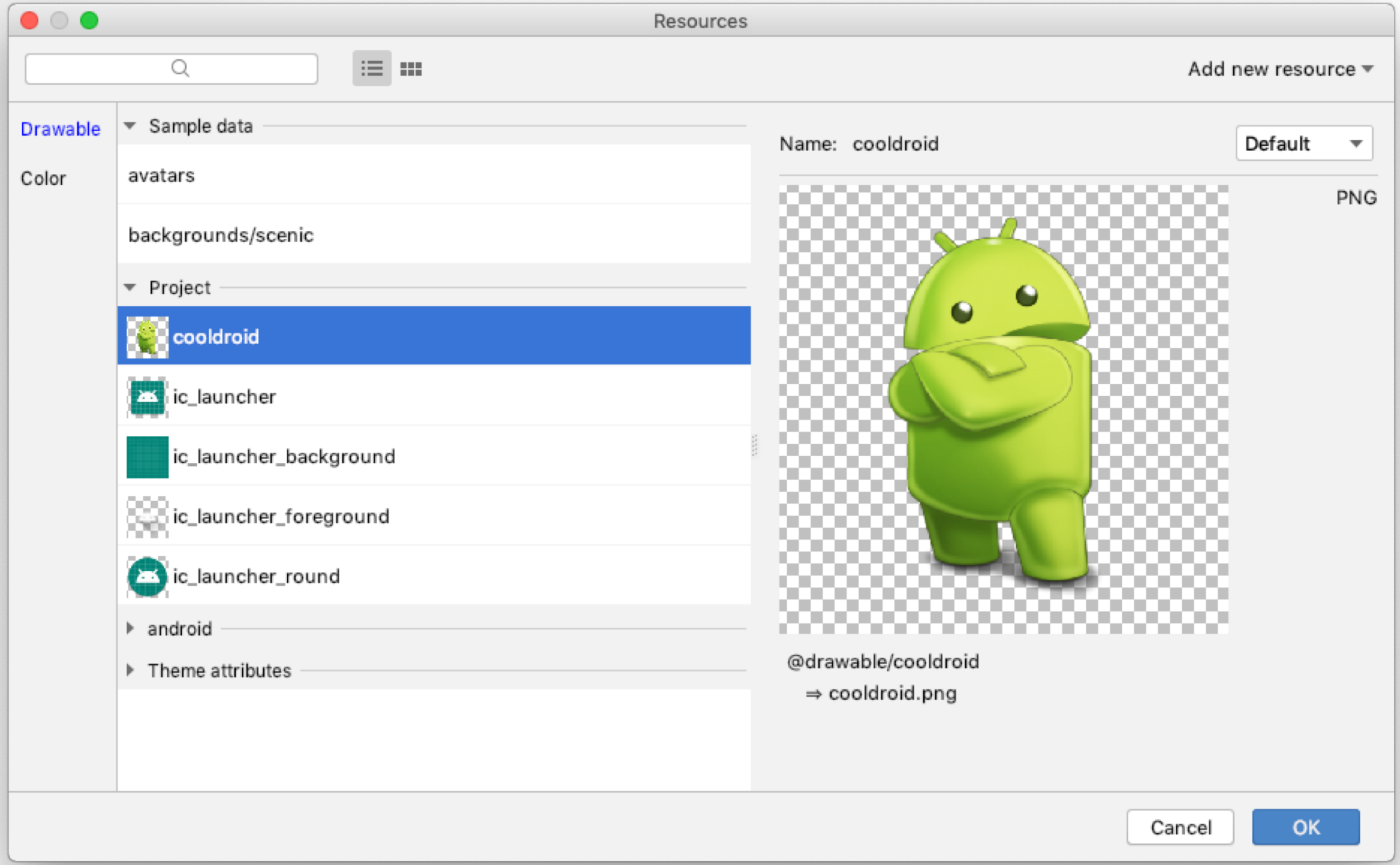
Use ^Space for path completion

☑ Search for references

□ Open moved files in editor

ImageView einfügen

- Im Designer
- Zuordnung des Images einfach
- TextView entfernen

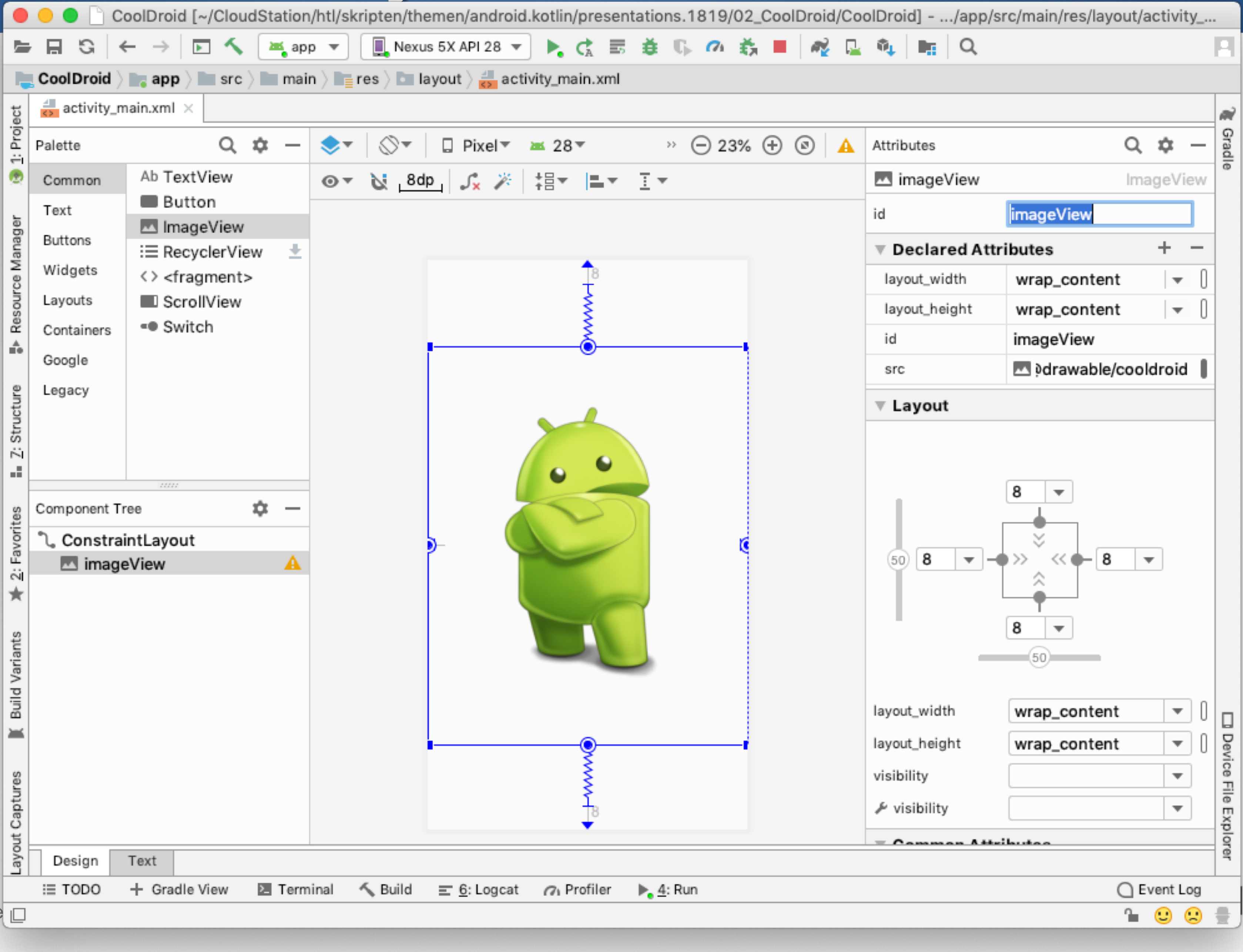


Name: cooldroid

@drawable/cooldroid
⇒ cooldroid.png

DING

ConstraintLayout



CoolDroid [~/CloudStation/htl/skripten/themen/android.kotlin/presentations.1819/02_CoolDroid/CoolDroid] - .../app/src/main/res/layout/activity_...

CoolDroid 〉 app 〉 src 〉 main 〉 res 〉 layout 〉 activity_main.xml

Component Tree

- ConstraintLayout
  - imageView

Attributes

imageView — ImageView

id: imageView

Declared Attributes

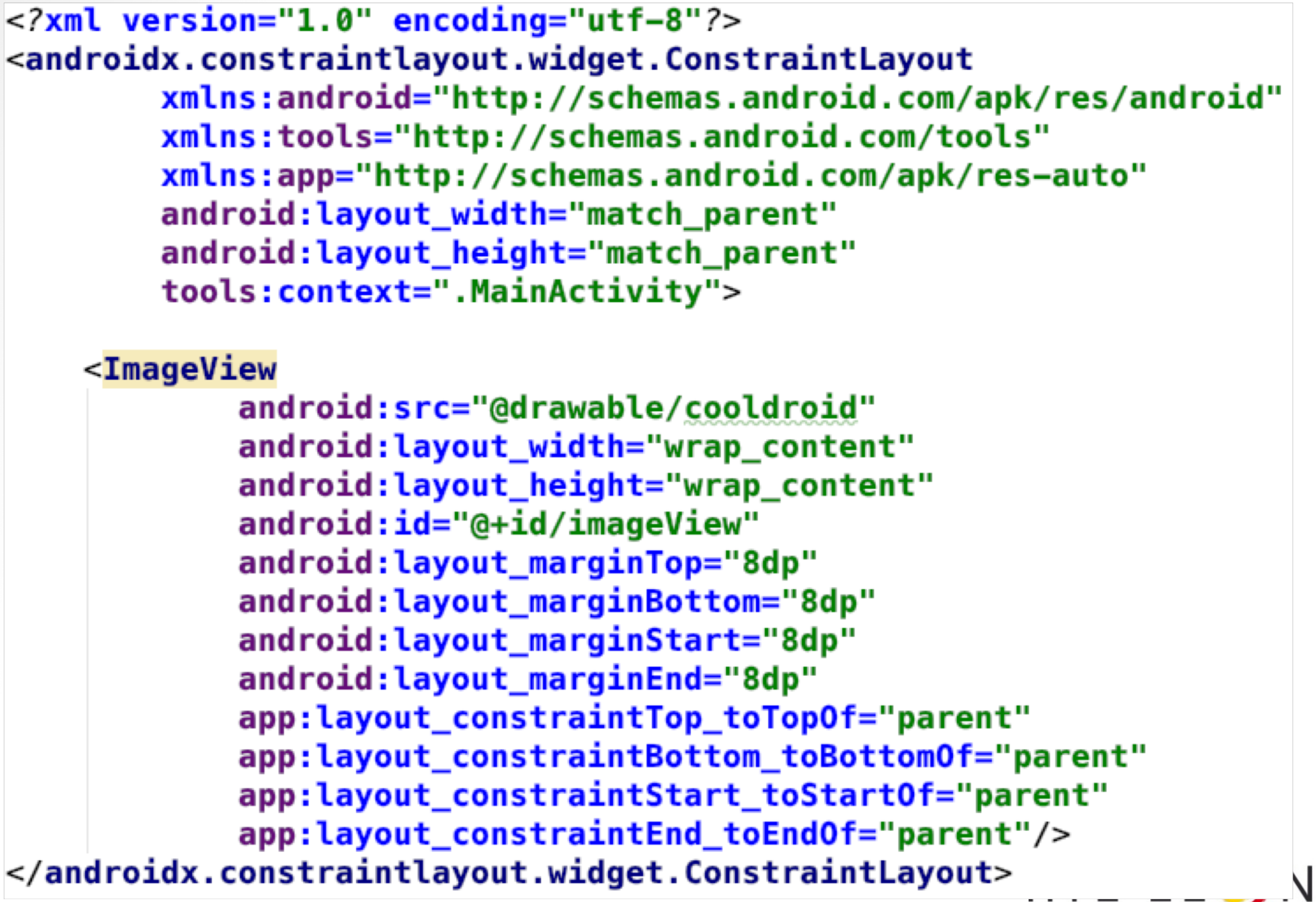
layout_width	wrap_content
layout_height	wrap_content
id	imageView
src	@drawable/cooldroid

Layout

layout_width	wrap_content
layout_height	wrap_content
visibility	
visibility	

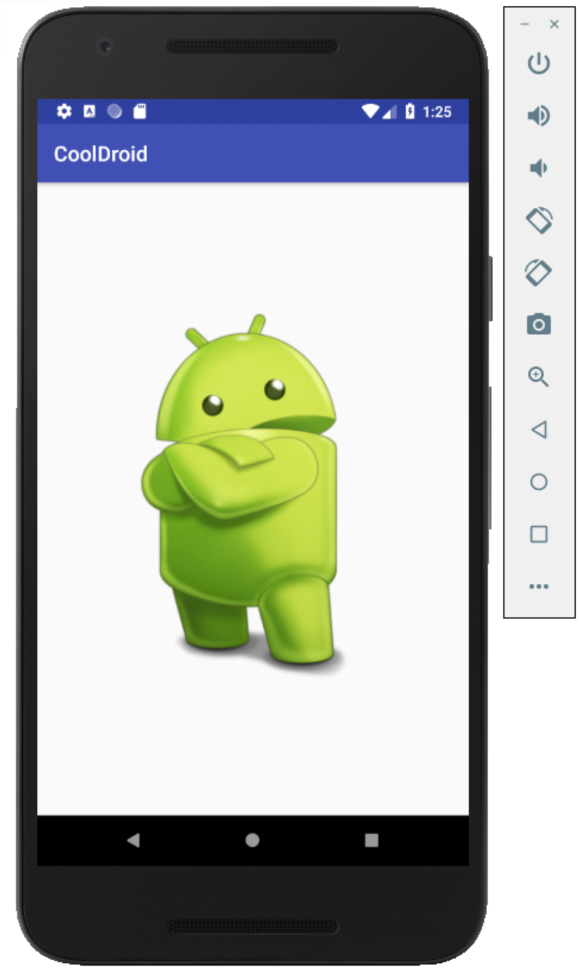
XML-Layout kontrollieren

- Etwaige Warnings beheben
  - Rechtschreibprüfung wird hier aber ignoriert



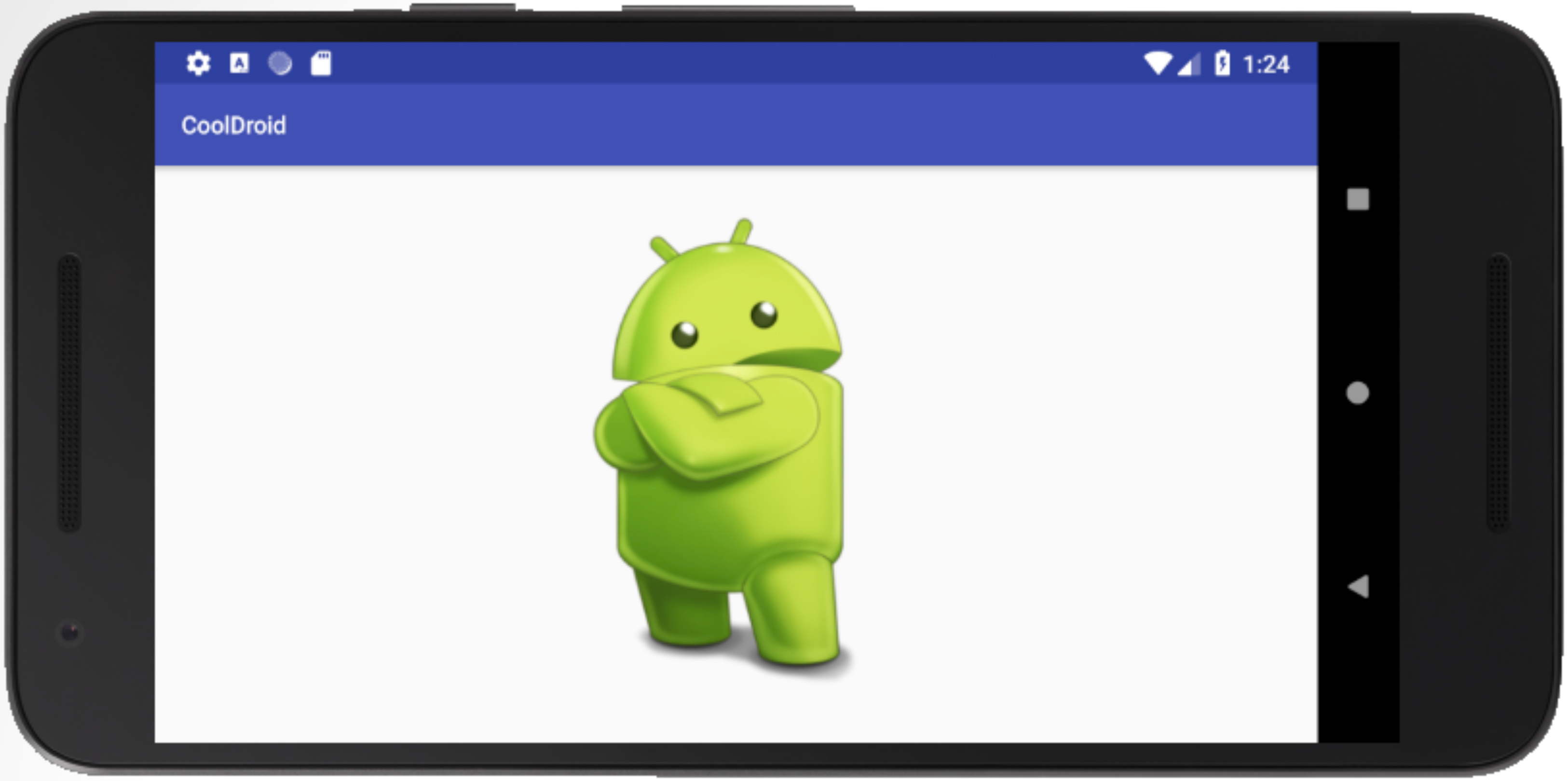
```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:tools="http://schemas.android.com/tools"
        xmlns:app="http://schemas.android.com/apk/res-auto"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        tools:context=".MainActivity">

    <ImageView
            android:src="@drawable/cooldroid"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:id="@+id/imageView"
            android:layout_marginTop="8dp"
            android:layout_marginBottom="8dp"
            android:layout_marginStart="8dp"
            android:layout_marginEnd="8dp"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintEnd_toEndOf="parent"/>
</androidx.constraintlayout.widget.ConstraintLayout>
```

App laufen lassen



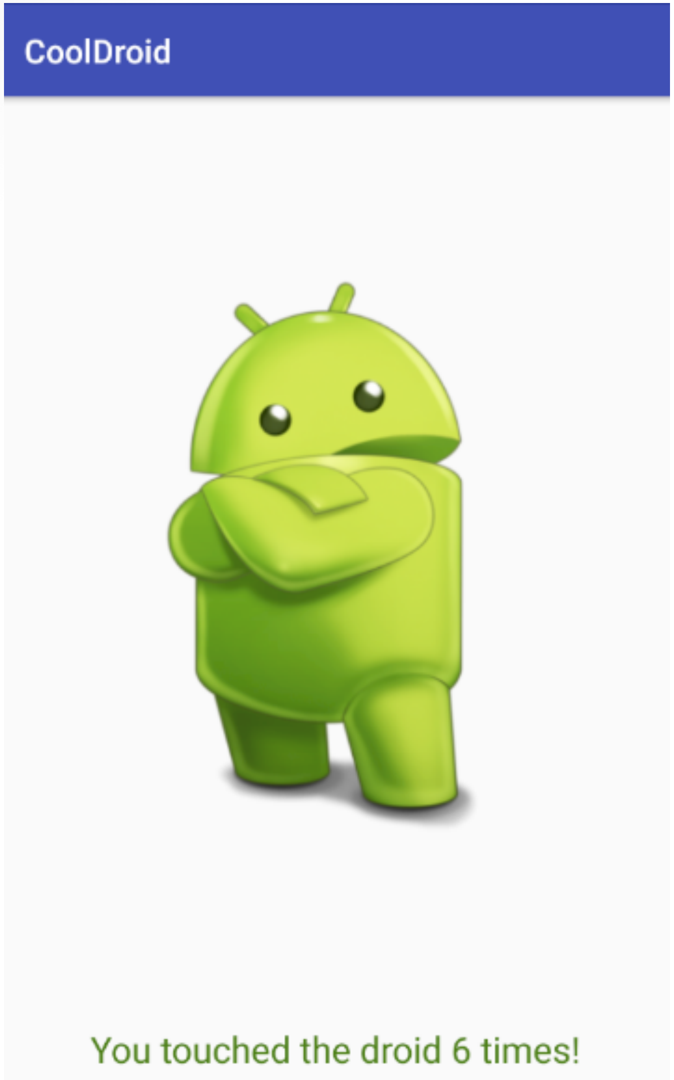
Achten Sie darauf, dass der HAXM läuft

Auch im Landscape-Mode



CoolDroid

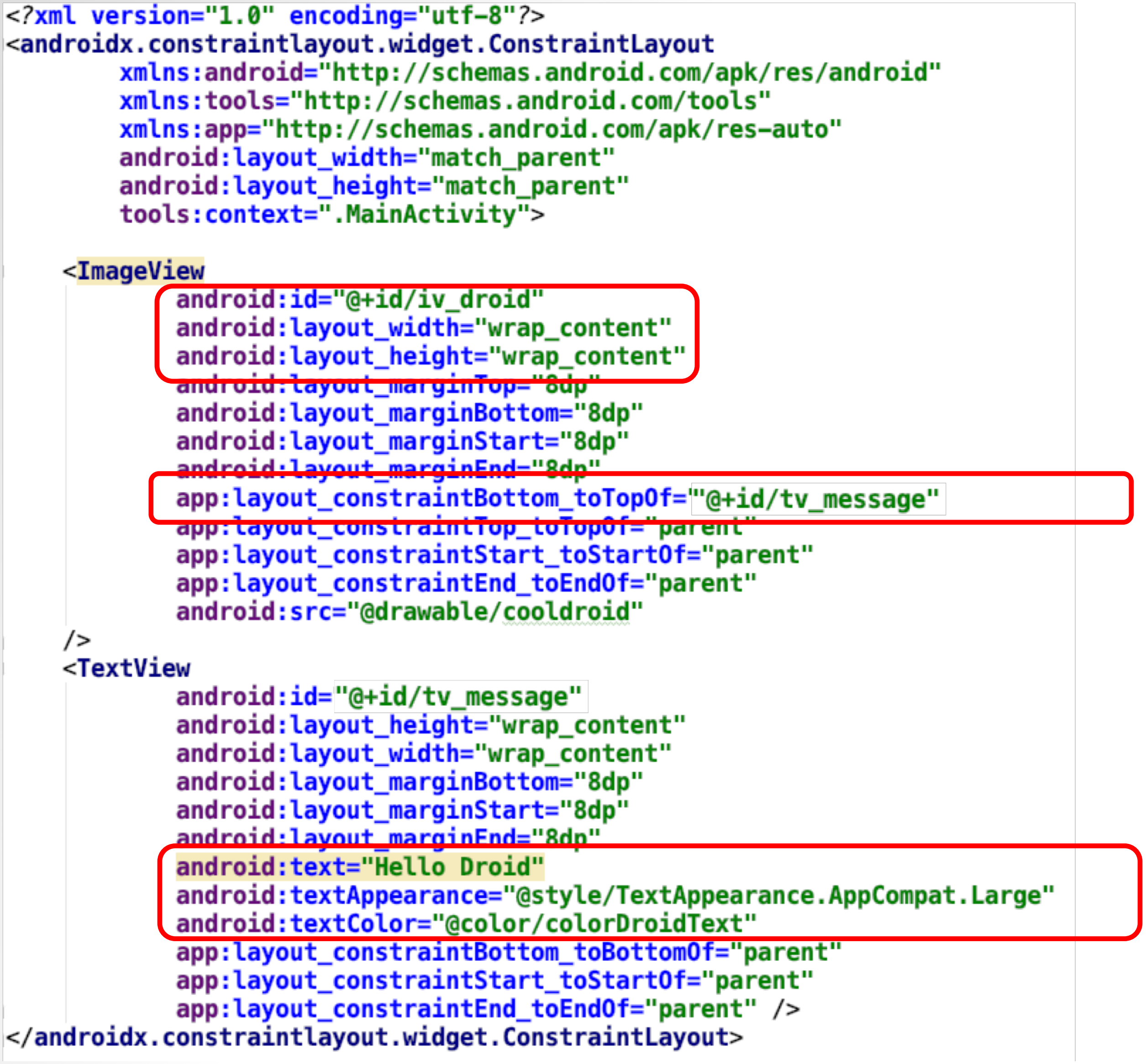
App reagiert auf Touch-Event



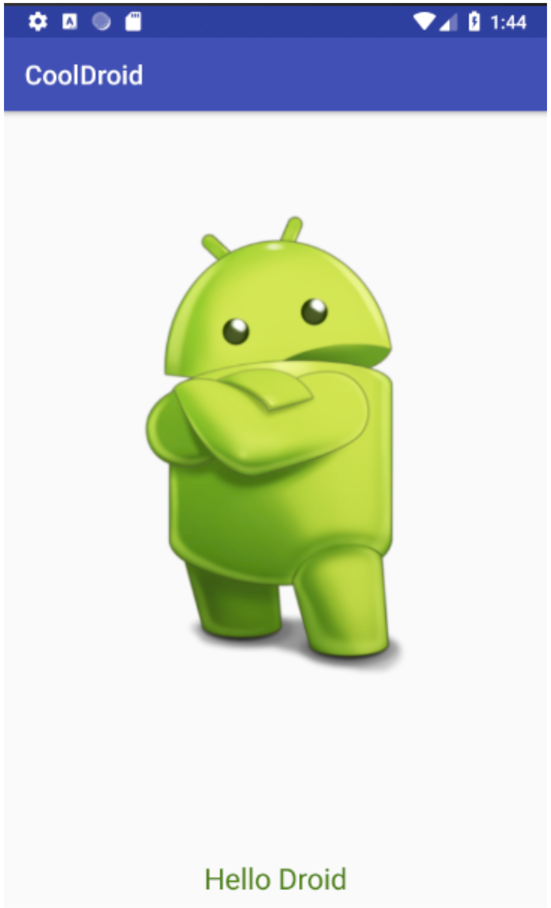
You touched the droid 6 times!

Übung

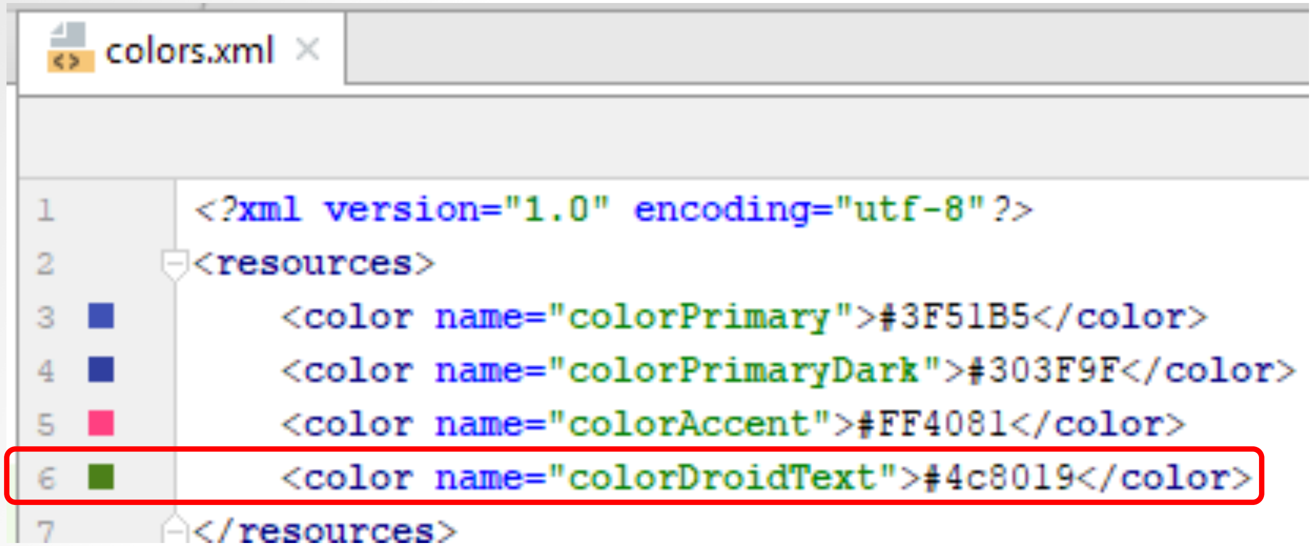
Viewdefinition



```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:tools="http://schemas.android.com/tools"
        xmlns:app="http://schemas.android.com/apk/res-auto"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        tools:context=".MainActivity">

    <ImageView
            android:id="@+id/iv_droid"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="8dp"
            android:layout_marginBottom="8dp"
            android:layout_marginStart="8dp"
            android:layout_marginEnd="8dp"
            app:layout_constraintBottom_toTopOf="@+id/tv_message"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            android:src="@drawable/cooldroid"
    />
    <TextView
            android:id="@+id/tv_message"
            android:layout_height="wrap_content"
            android:layout_width="wrap_content"
            android:layout_marginBottom="8dp"
            android:layout_marginStart="8dp"
            android:layout_marginEnd="8dp"
            android:text="Hello Droid"
            android:textAppearance="@style/TextAppearance.AppCompat.Large"
            android:textColor="@color/colorDroidText"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintEnd_toEndOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```



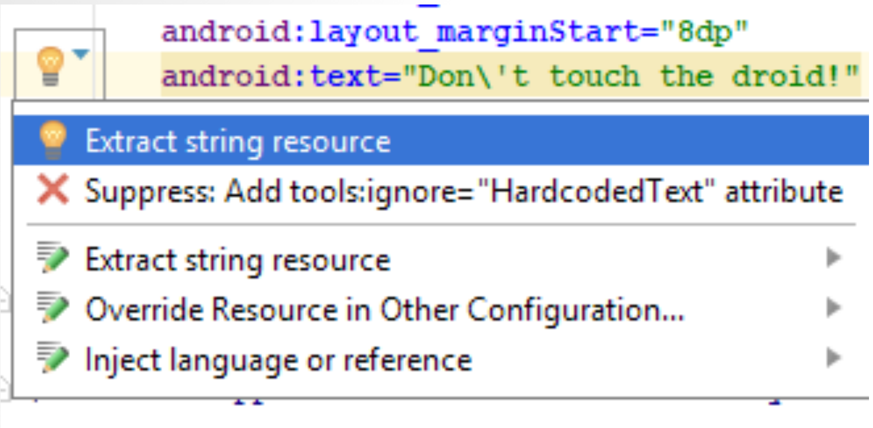
Colors sind als Ressource definiert



```xml
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#3F51B5</color>
    <color name="colorPrimaryDark">#303F9F</color>
    <color name="colorAccent">#FF4081</color>
    <color name="colorDroidText">#4c8019</color>
</resources>
```

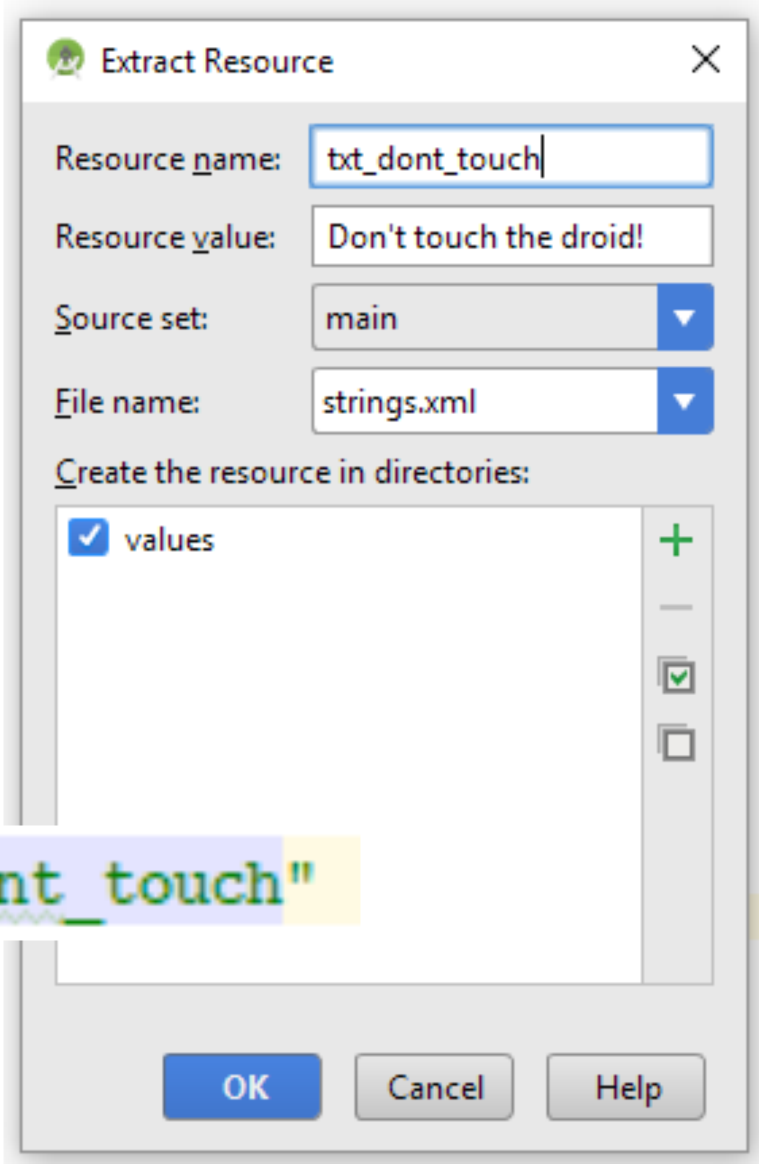
Keine literalen Texte im Code oder XML

- In Stringressourcen auslagern



```
android:layout_marginStart="8dp"
android:text="Don\'t touch the droid!"
```

Extract string resource
Suppress: Add tools:ignore="HardcodedText" attribute
Extract string resource
Override Resource in Other Configuration...
Inject language or reference



Resource name: txt_dont_touch

Resource value: Don't touch the droid!

Source set: main

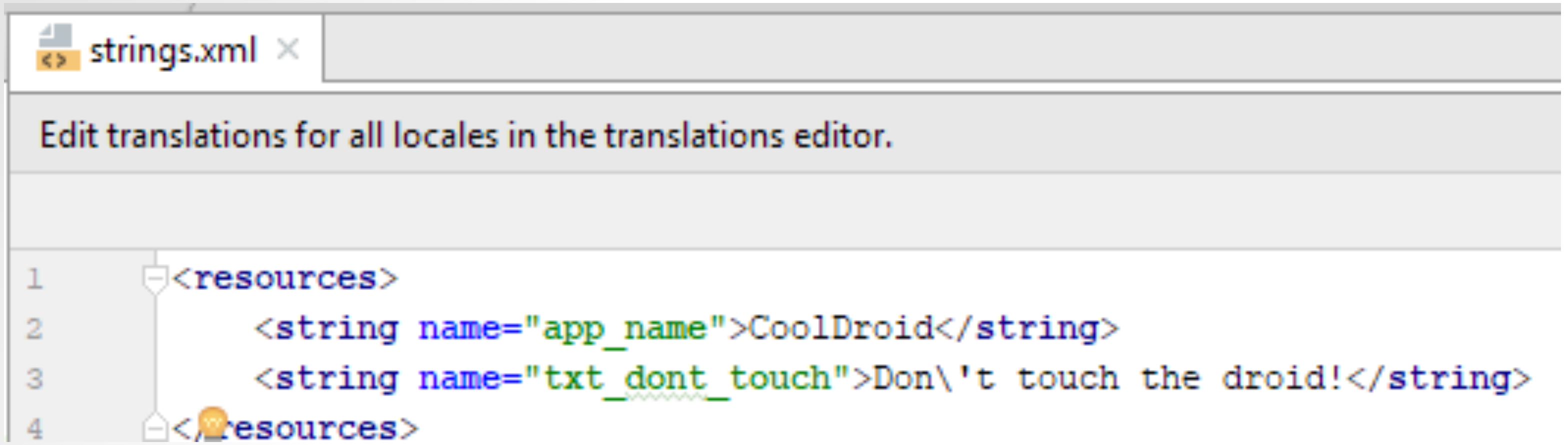
File name: strings.xml

Create the resource in directories:

☑ values

```
android:text="@string/txt_dont_touch"
```

Texte als Stringressourcen

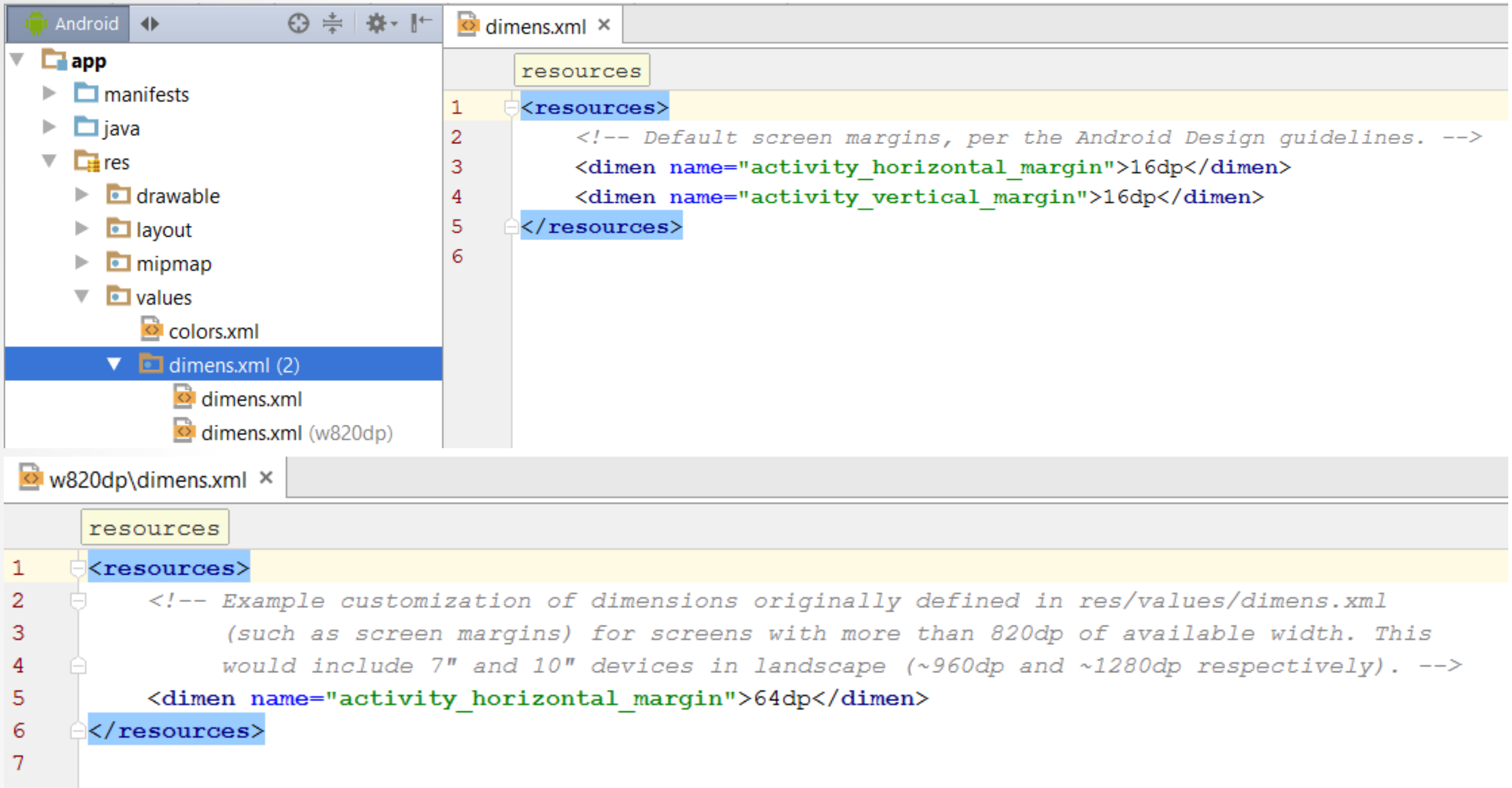


Edit translations for all locales in the translations editor.

```
<resources>
    <string name="app_name">CoolDroid</string>
    <string name="txt_dont_touch">Don\'t touch the droid!</string>
</resources>
```

Abstände in dimens.xml

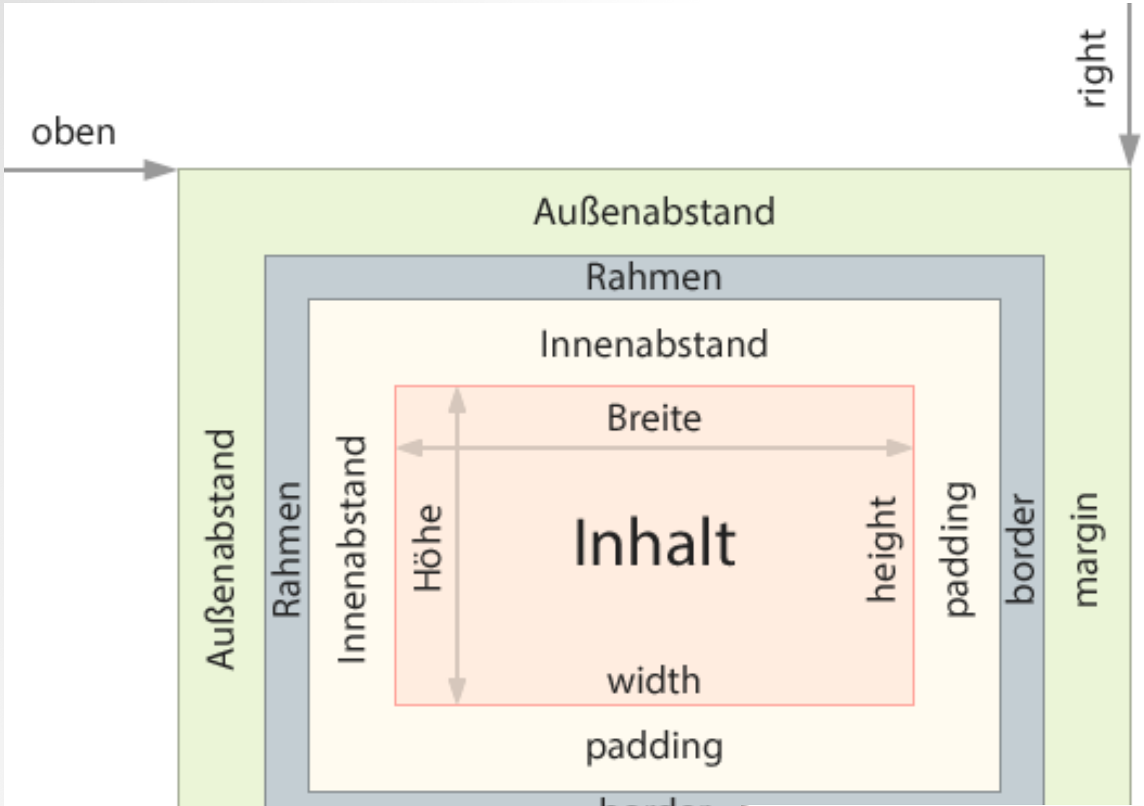
- Unterschiedliche Abstände abhängig von der Gerätegröße

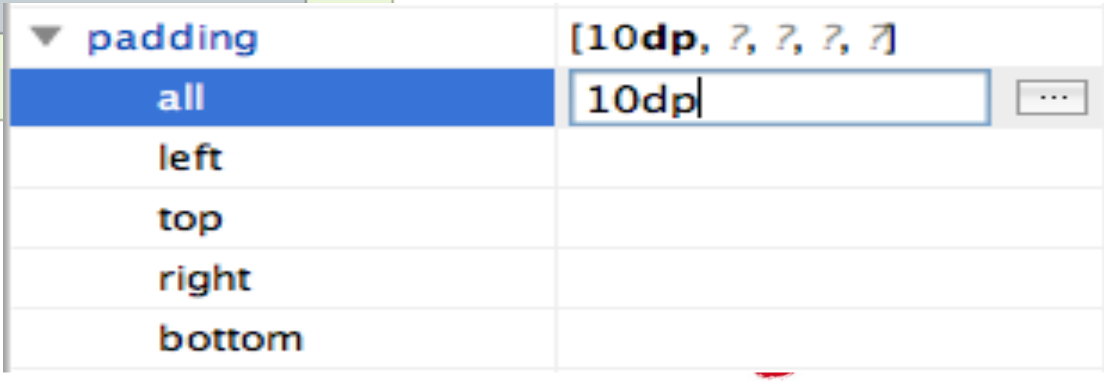


```xml
<resources>
    <!-- Default screen margins, per the Android Design guidelines. -->
    <dimen name="activity_horizontal_margin">16dp</dimen>
    <dimen name="activity_vertical_margin">16dp</dimen>
</resources>
```

```xml
<resources>
    <!-- Example customization of dimensions originally defined in res/values/dimens.xml
         (such as screen margins) for screens with more than 820dp of available width. This
         would include 7" and 10" devices in landscape (~960dp and ~1280dp respectively). -->
    <dimen name="activity_horizontal_margin">64dp</dimen>
</resources>
```

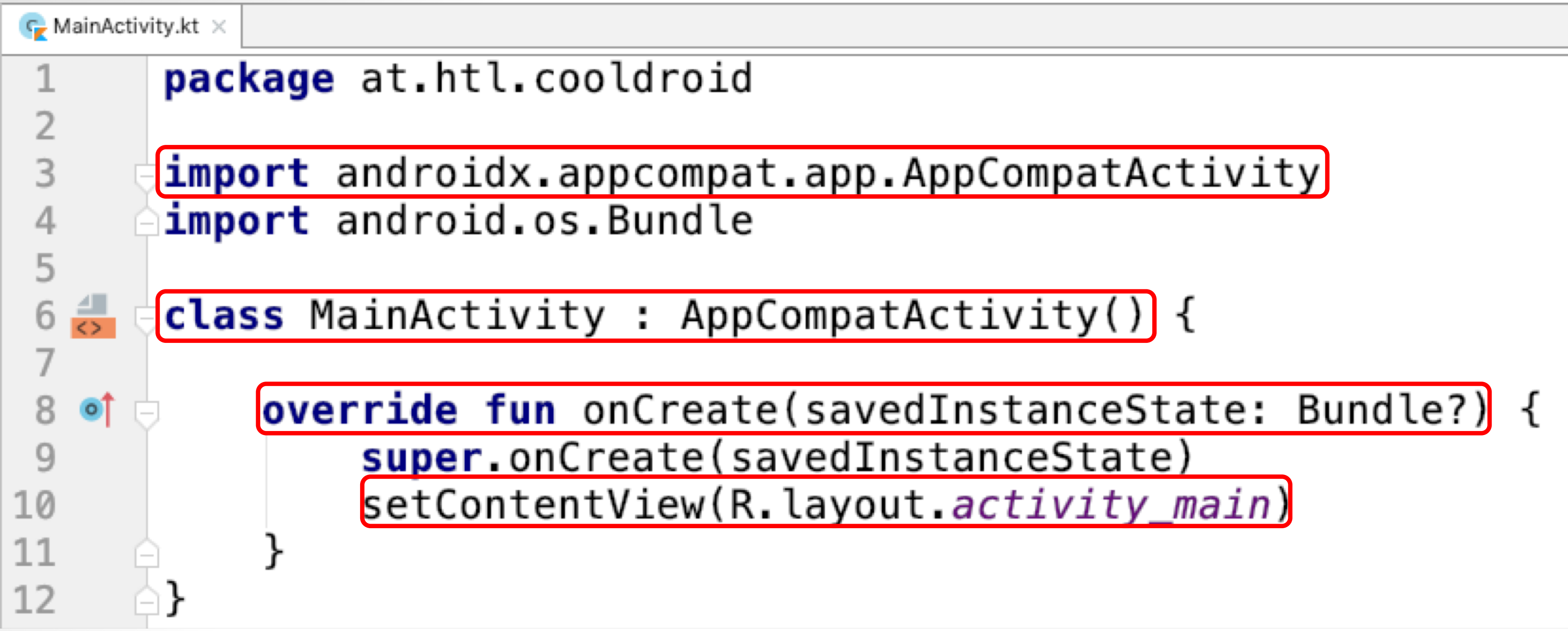
Padding und Margin wie in HTML





▼ padding	[10dp, ?, ?, ?, ?]
all	10dp
left	
top	
right	
bottom	

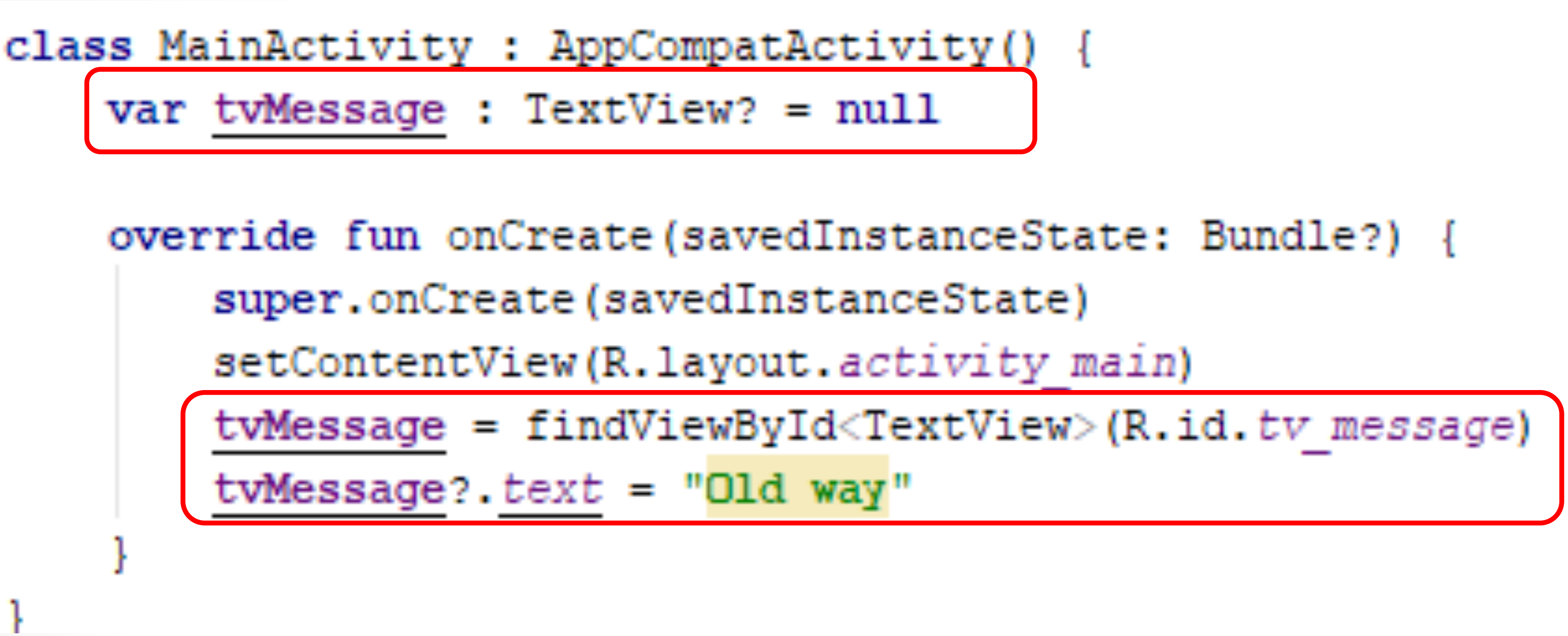
Endlich Kotlin → Activity als Controller



```kotlin
package at.htl.cooldroid

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle

class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
    }
}
```

Auf Viewelemente in Controller zugreifen

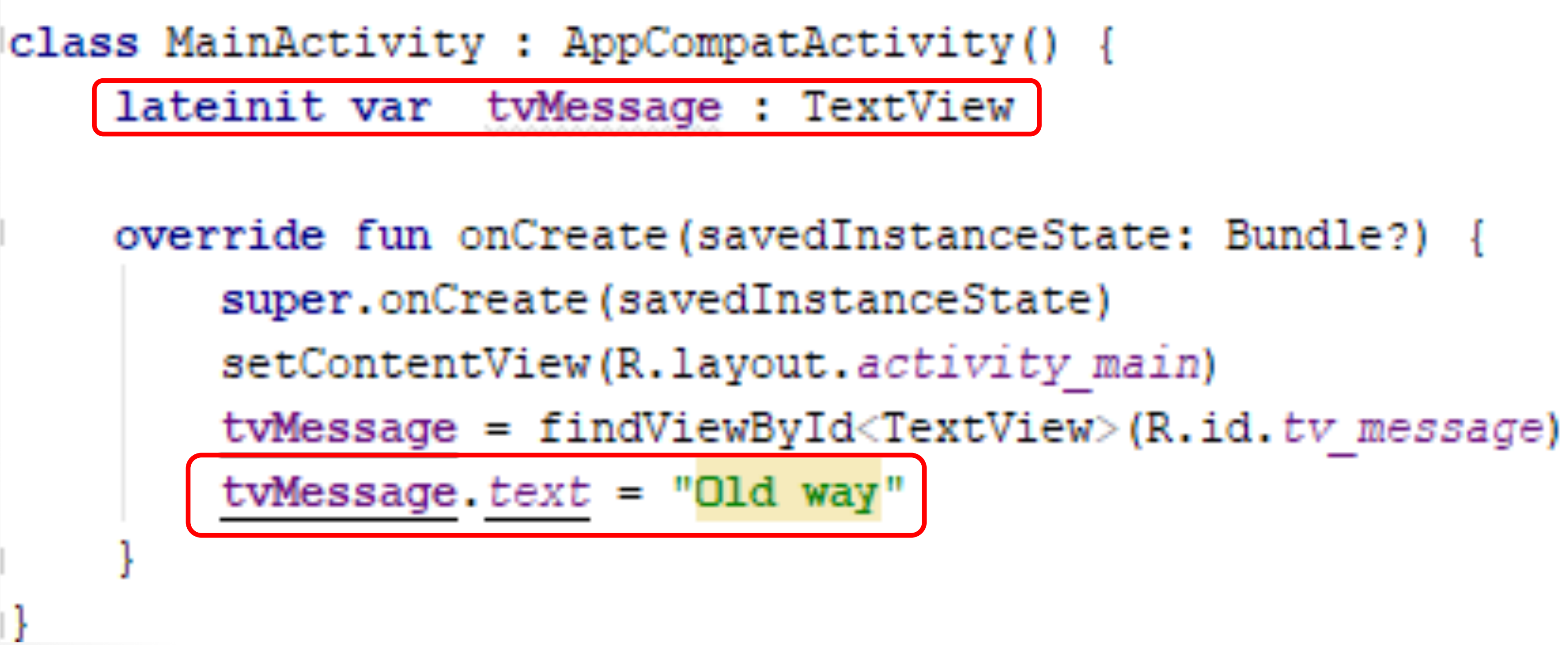
- Java-Way in Kotlin
  - tvMessage ist var statt val
  - Muss am Anfang mit null initialisiert werden
  - Immer wieder findViewById()
  - Zumindest der Cast kann weggelassen werden



```
class MainActivity : AppCompatActivity() {
    var tvMessage : TextView? = null

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        tvMessage = findViewById<TextView>(R.id.tv_message)
        tvMessage?.text = "Old way"
    }
}
```

Kleine Erleichterung - lateinit

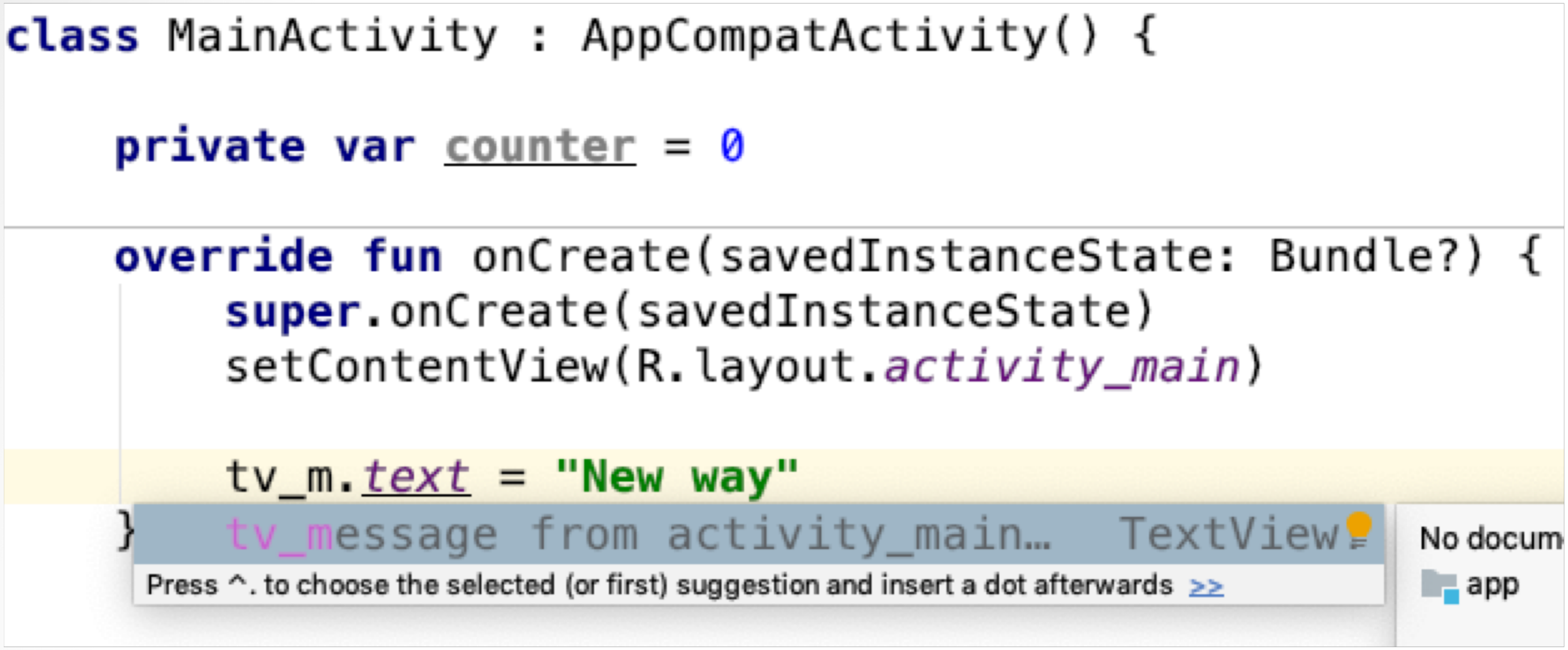
- Das Nullable-Problem wird behoben
- Zugriff ohne null-check ist möglich



```
class MainActivity : AppCompatActivity() {
    lateinit var  tvMessage : TextView

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        tvMessage = findViewById<TextView>(R.id.tv_message)
        tvMessage.text = "Old way"
    }
}
```

Kotlin-Way

- Import der Kotlinextension notwendig
- Codierrichtline camelCase verletzt
  - Dafür ist tv_droid sofort als GUI-Control erkennbar

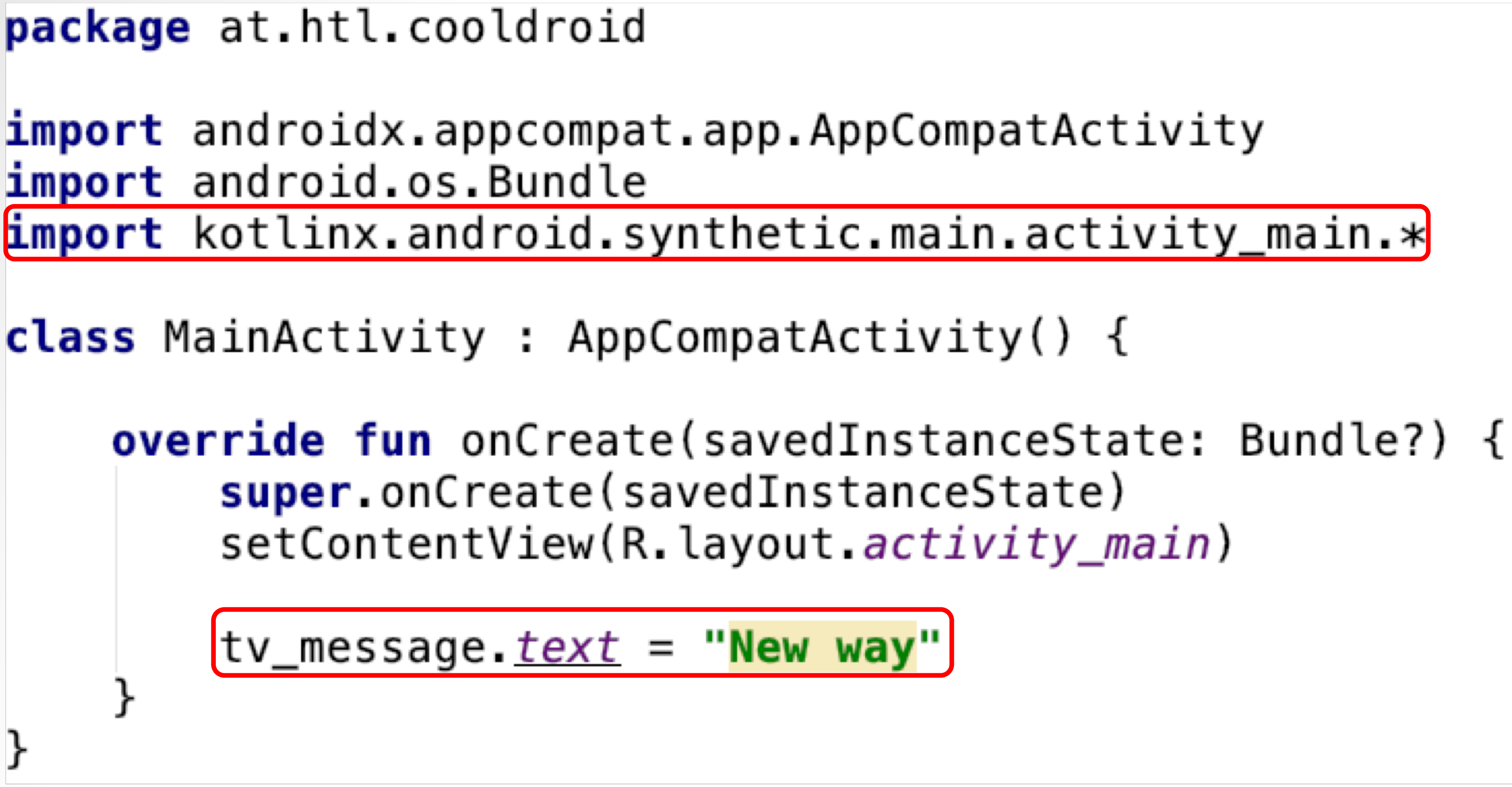


```
class MainActivity : AppCompatActivity() {

    private var counter = 0

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        tv_m.text = "New way"
    }
```

```
tv_message from activity_main...   TextView
Press ^. to choose the selected (or first) suggestion and insert a dot afterwards >>
```

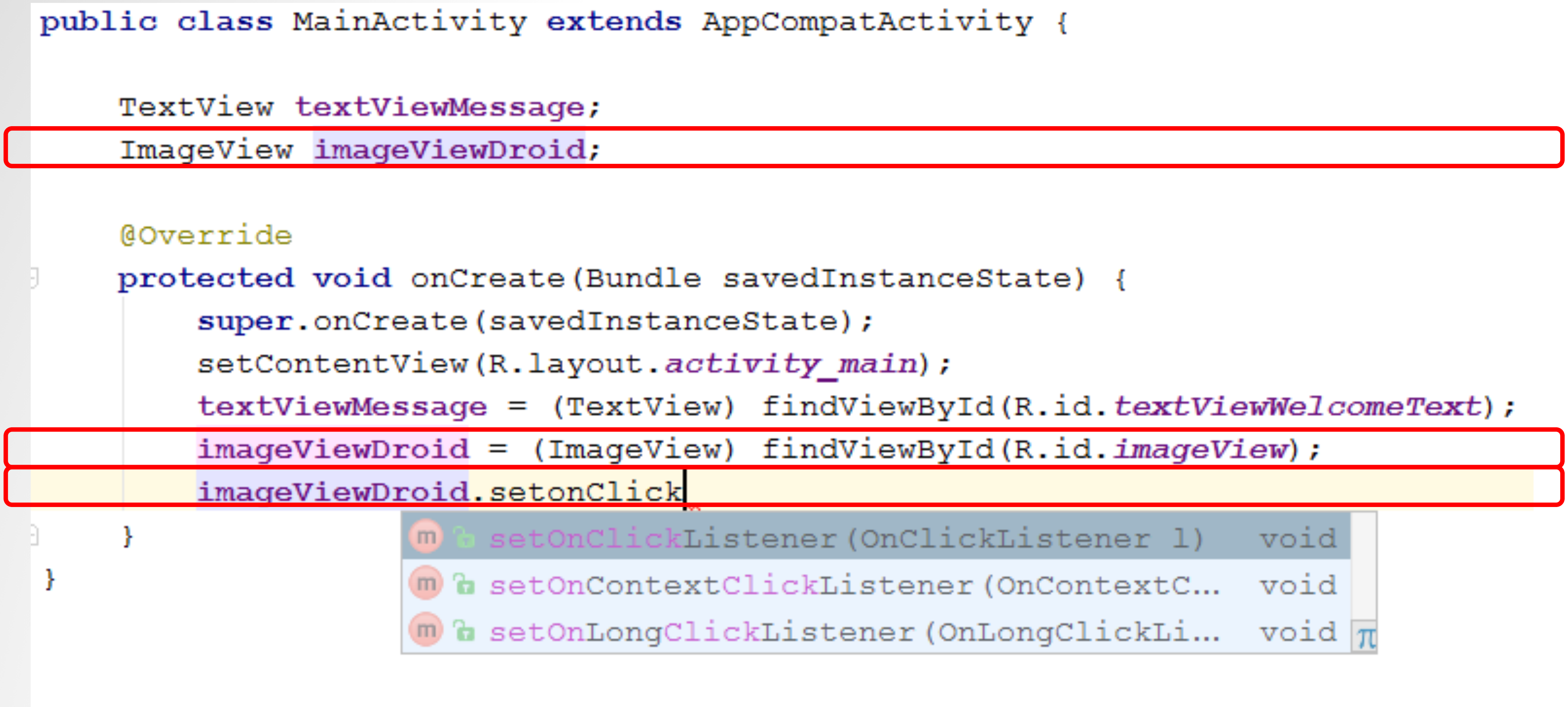
```
No docum
app
```

Optimal lesbarer Code



```
package at.htl.cooldroid

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import kotlinx.android.synthetic.main.activity_main.*

class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        tv_message.text = "New way"
    }
}
```

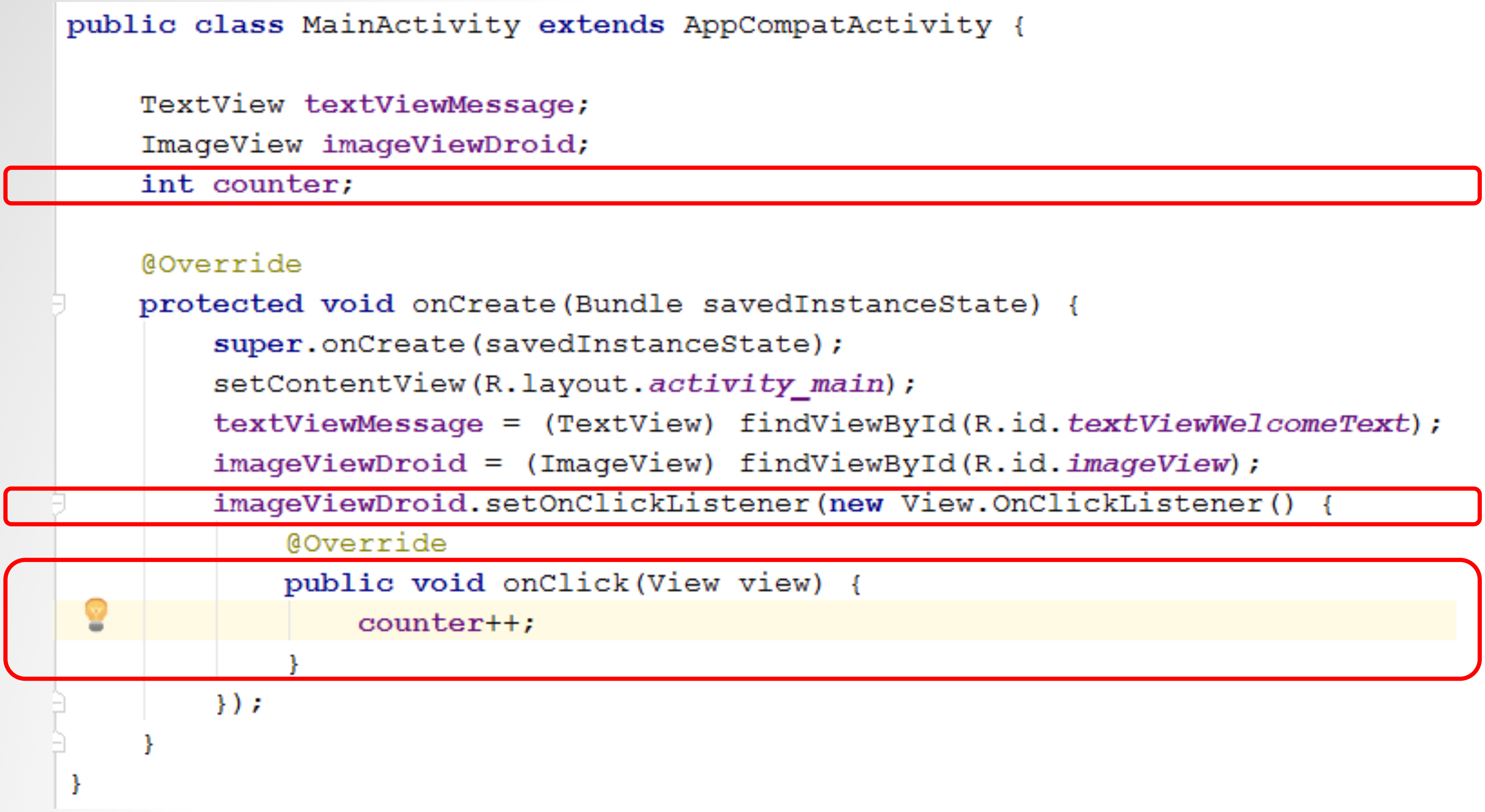
Java: OnClickListener auf ImageView setzen



```
public class MainActivity extends AppCompatActivity {

    TextView textViewMessage;
    ImageView imageViewDroid;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        textViewMessage = (TextView) findViewById(R.id.textViewWelcomeText);
        imageViewDroid = (ImageView) findViewById(R.id.imageView);
        imageViewDroid.setonClick
    }
}
```

m setOnClickListener(OnClickListener l) void
m setOnContextClickListener(OnContextC... void
m setOnLongClickListener(OnLongClickLi... void
π

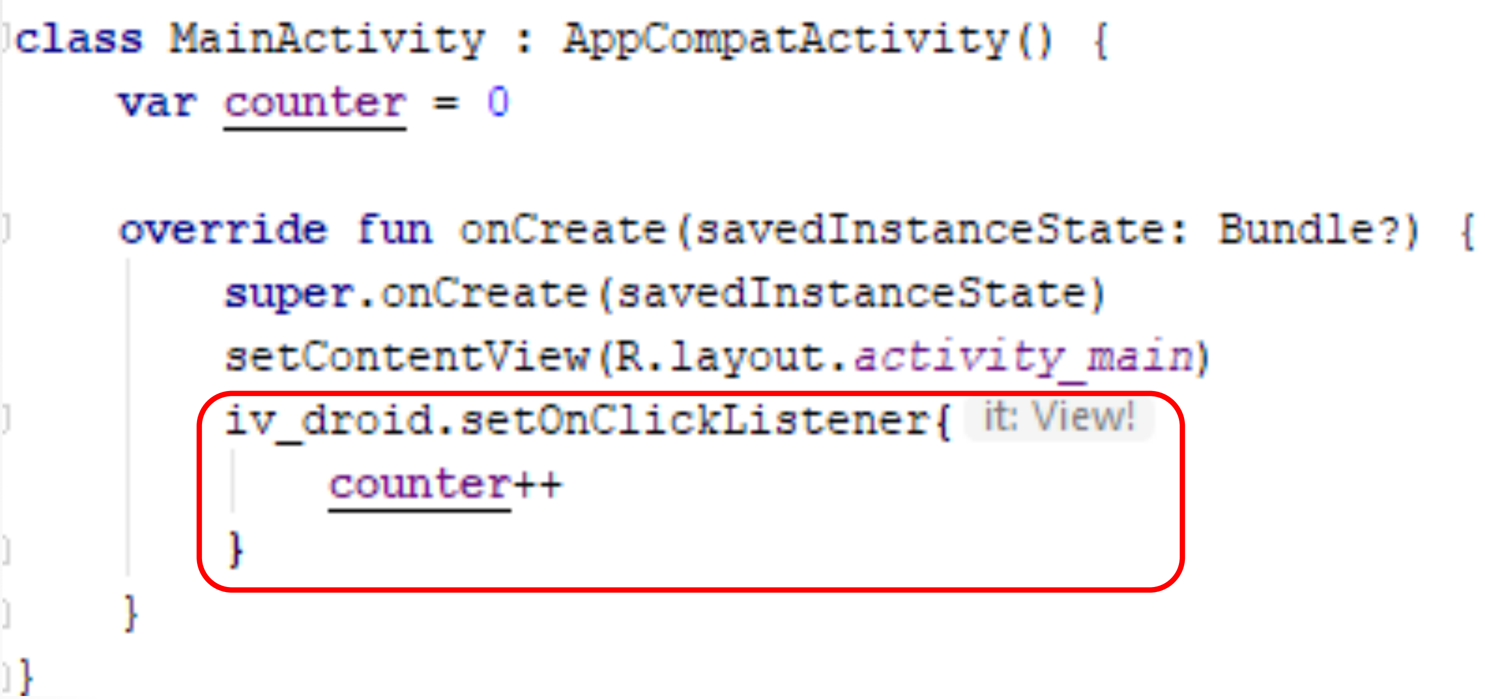
Java: Als anonyme innere Klasse



```
public class MainActivity extends AppCompatActivity {

    TextView textViewMessage;
    ImageView imageViewDroid;
    int counter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        textViewMessage = (TextView) findViewById(R.id.textViewWelcomeText);
        imageViewDroid = (ImageView) findViewById(R.id.imageView);
        imageViewDroid.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                counter++;
            }
        });
    }
}
```

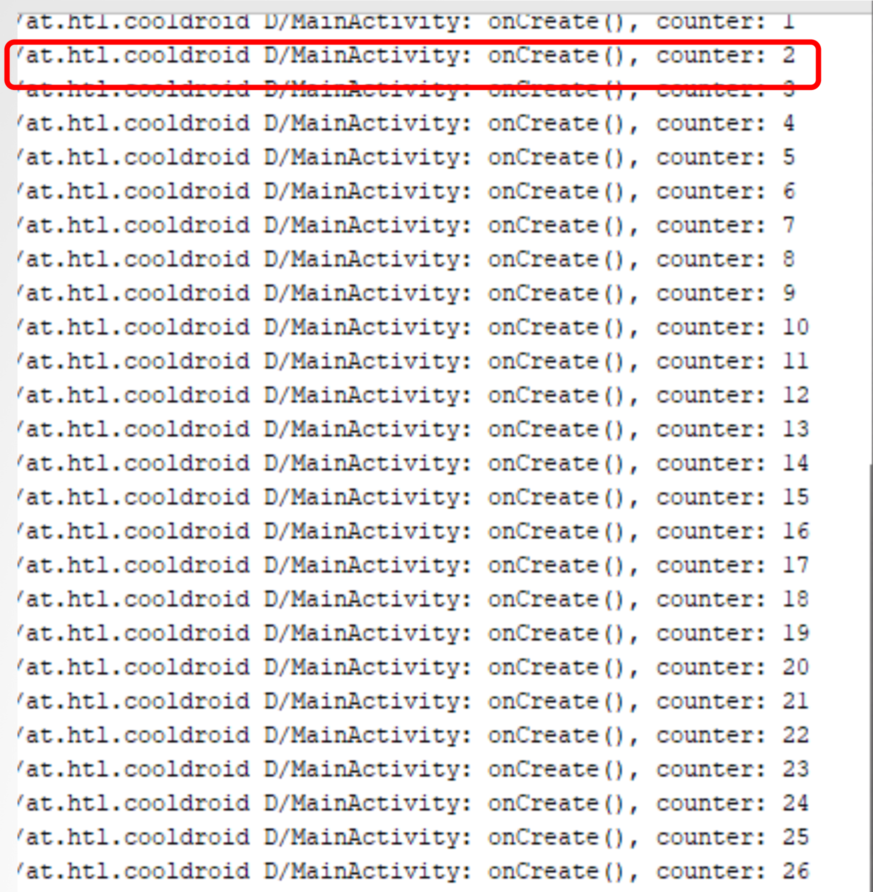
In Kotlin mit LambdaExpression

- Lambda ist letzter Parameter ➔ {} – Syntax
  - it wird nicht benötigt

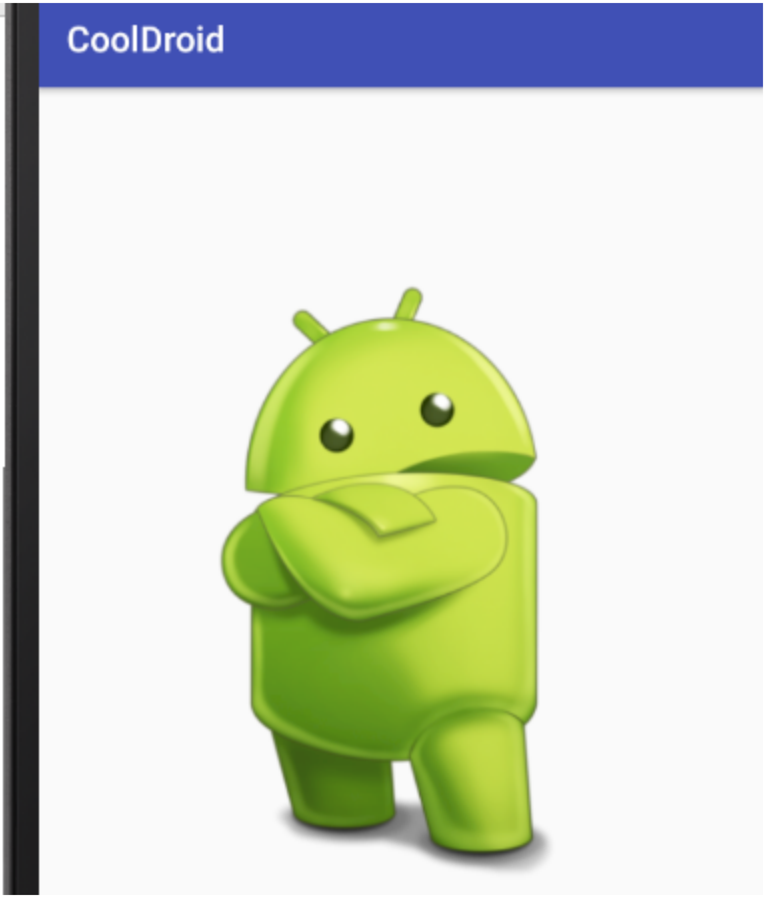


```
class MainActivity : AppCompatActivity() {
    var counter = 0

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        iv_droid.setOnClickListener{ it: View!
            counter++
        }
    }
}
```

Ausgabe des Zählers per LogCat

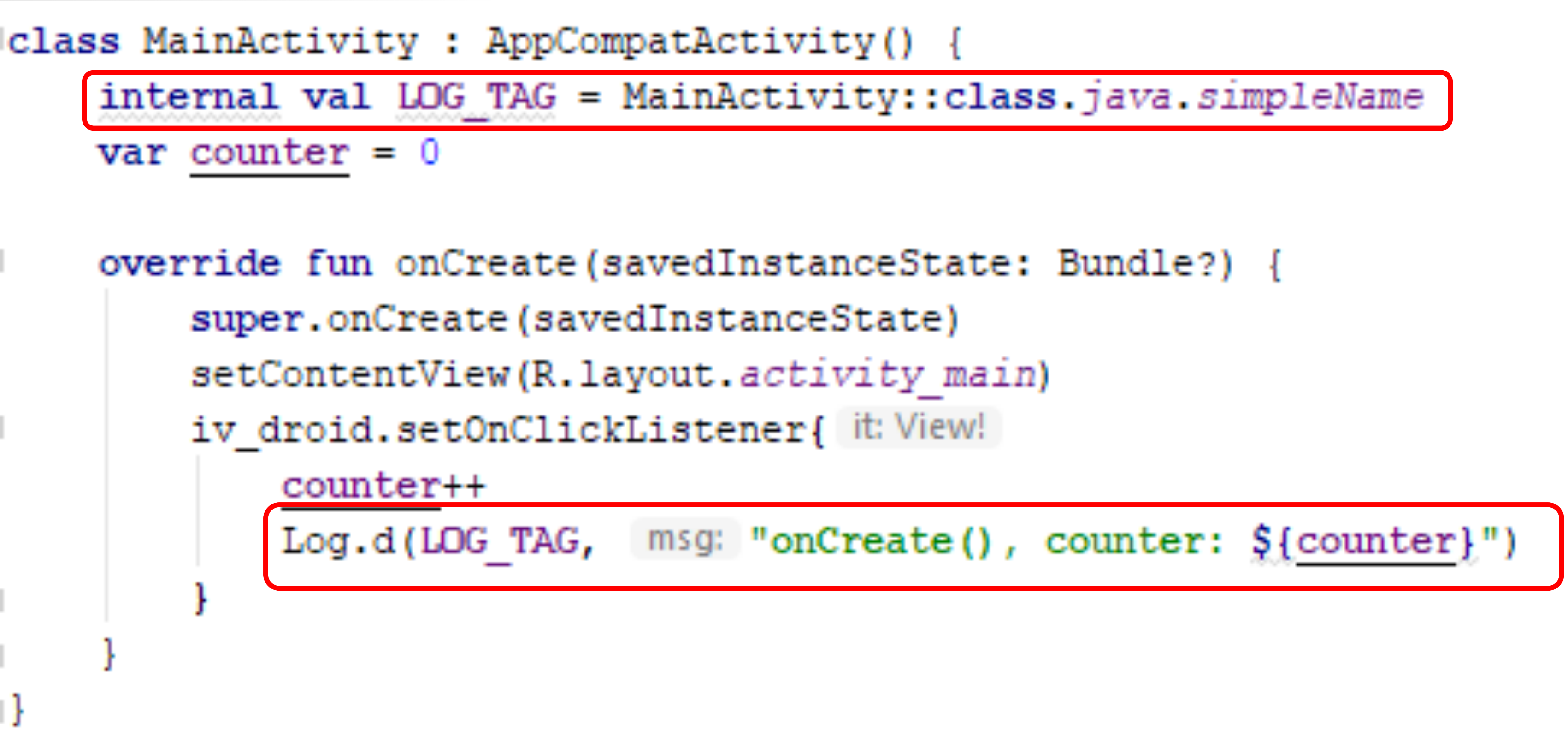


```
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 1
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 2
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 3
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 4
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 5
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 6
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 7
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 8
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 9
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 10
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 11
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 12
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 13
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 14
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 15
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 16
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 17
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 18
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 19
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 20
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 21
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 22
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 23
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 24
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 25
/at.htl.cooldroid D/MainActivity: onCreate(), counter: 26
```



Übung

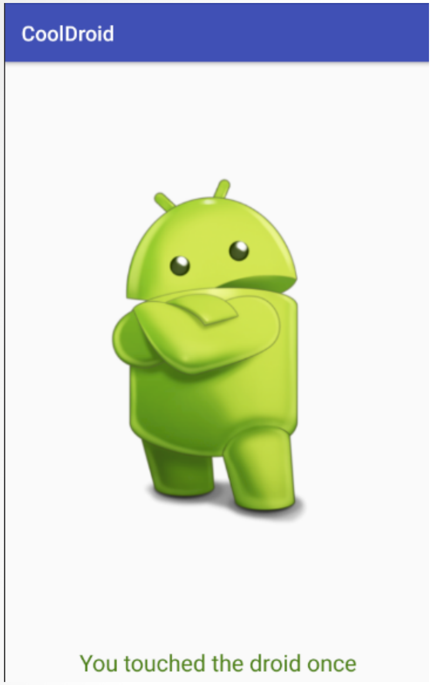
Implementierung - Logging



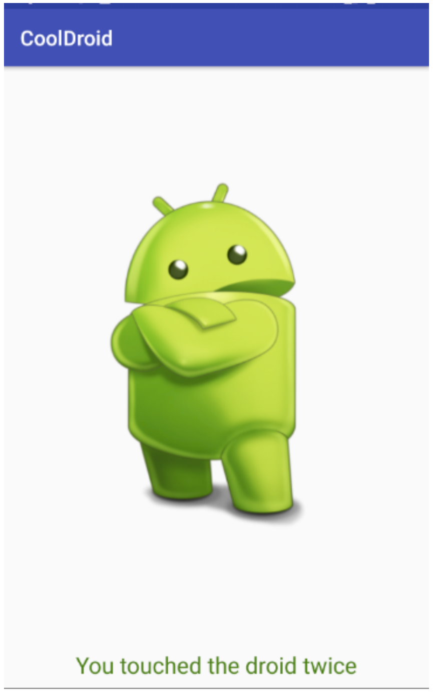
```
class MainActivity : AppCompatActivity() {
    internal val LOG_TAG = MainActivity::class.java.simpleName
    var counter = 0

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        iv_droid.setOnClickListener{ it: View!
            counter++
            Log.d(LOG_TAG, msg: "onCreate(), counter: ${counter}")
        }
    }
}
```

Texte abhängig vom Zählerstand ausgeben

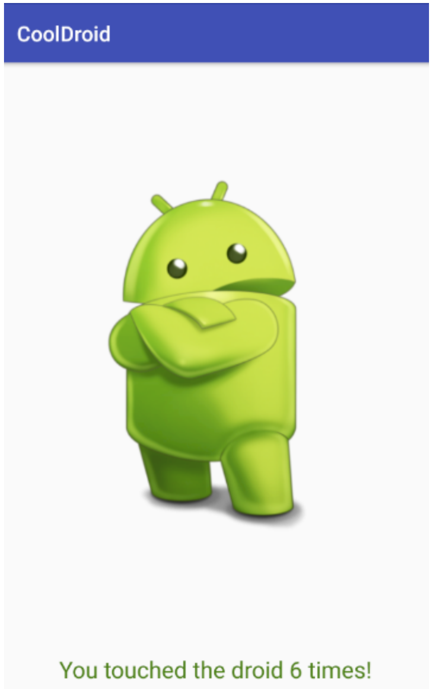
- Texte in Stringressource mit Platzhaltern
- once, twice für 1* und 2*
- dann 3 times, 4 times …



You touched the droid once



You touched the droid twice

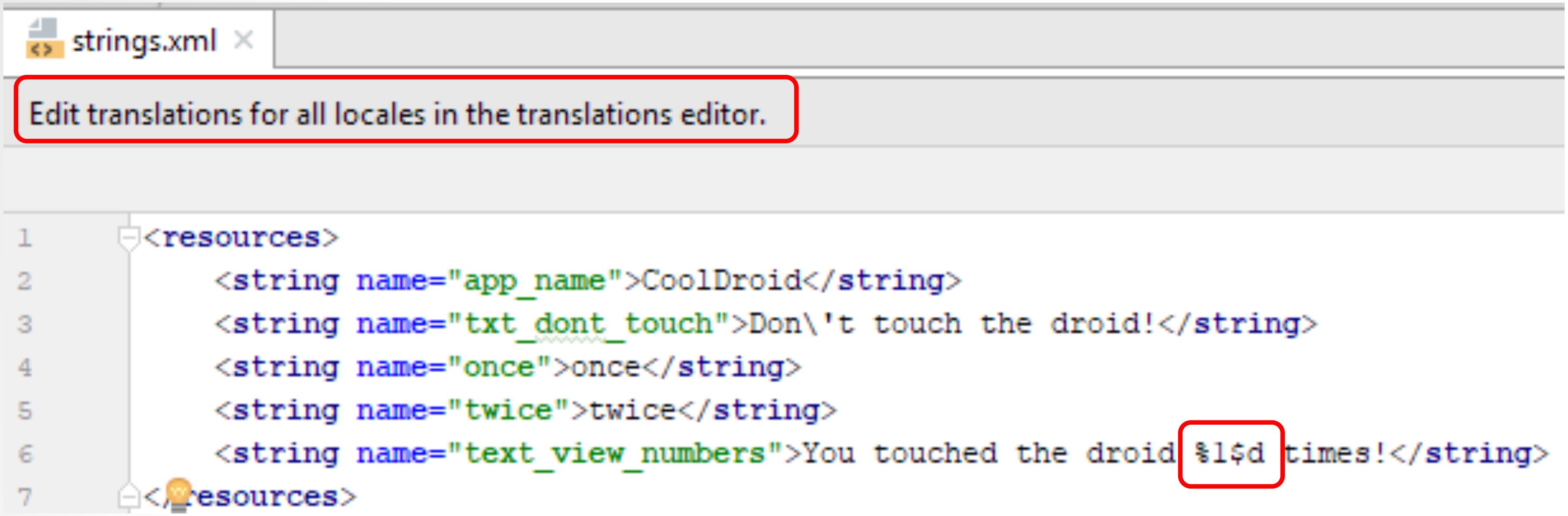


You touched the droid 6 times!

Übung

Stringressourcen mit Platzhaltern

- Hinweis auf Mehrsprachigkeit
- Platzhalter im letzten Fall



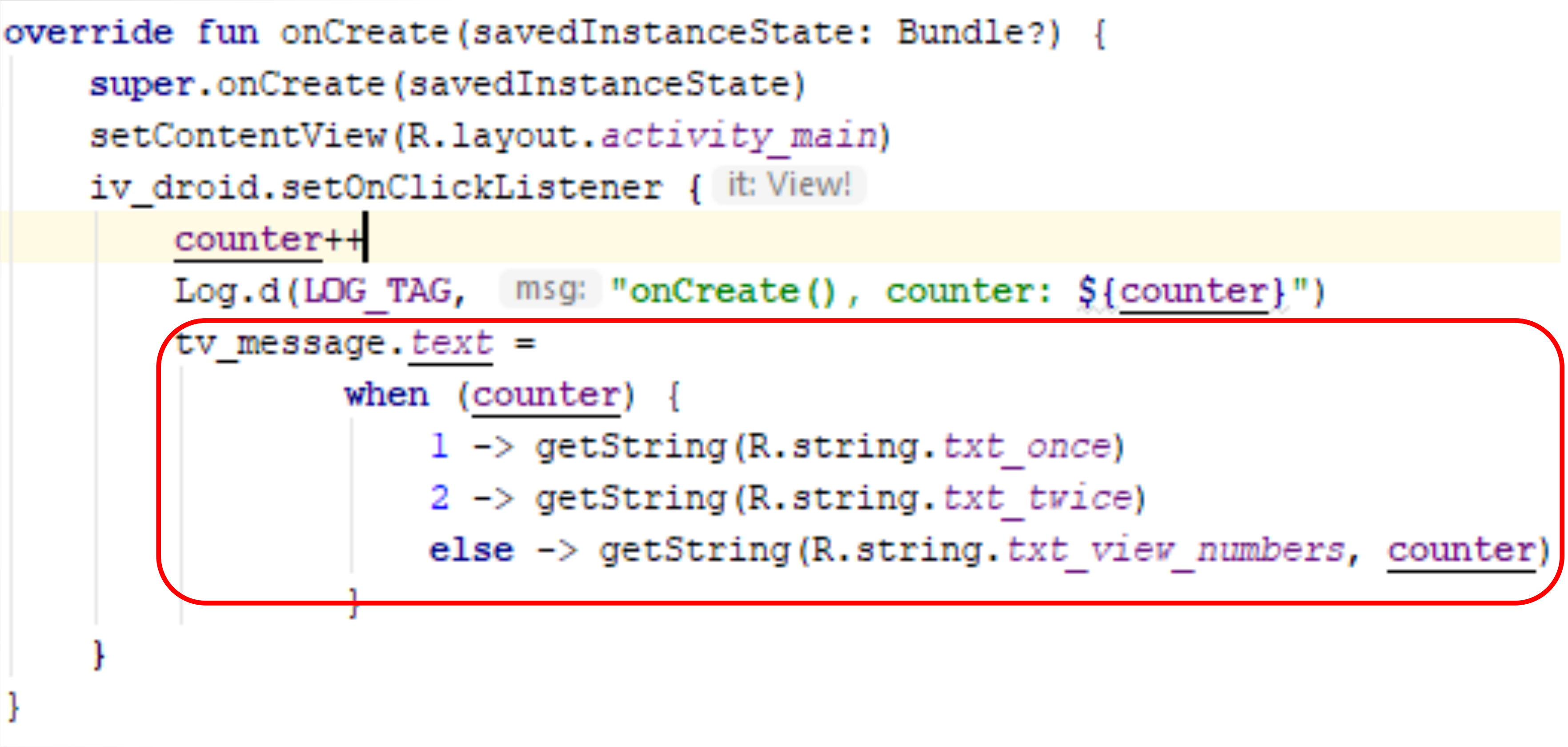
strings.xml

Edit translations for all locales in the translations editor.

```
<resources>
    <string name="app_name">CoolDroid</string>
    <string name="txt_dont_touch">Don\'t touch the droid!</string>
    <string name="once">once</string>
    <string name="twice">twice</string>
    <string name="text_view_numbers">You touched the droid %1$d times!</string>
</resources>
```

Kompakter Code

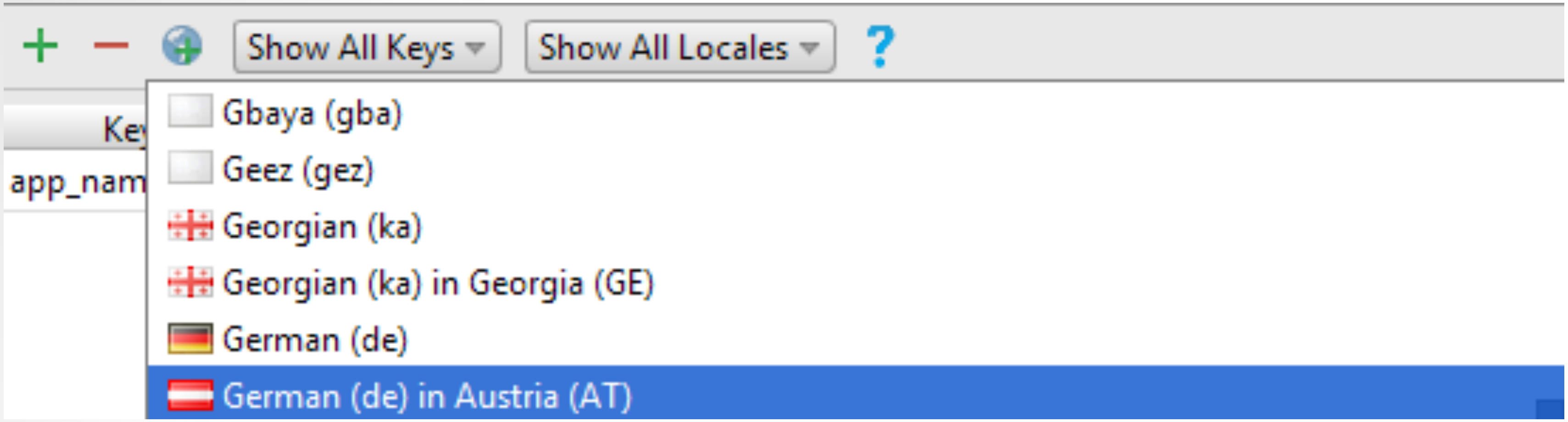
- War in Java eigene Methode



```
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_main)
    iv_droid.setOnClickListener { it: View!
        counter++
        Log.d(LOG_TAG, msg: "onCreate(), counter: ${counter}")
        tv_message.text =
                when (counter) {
                    1 -> getString(R.string.txt_once)
                    2 -> getString(R.string.txt_twice)
                    else -> getString(R.string.txt_view_numbers, counter)
                }
    }
}
```

Mehrsprachigkeit implementieren

- Ressourcendatei sprachspezifisch

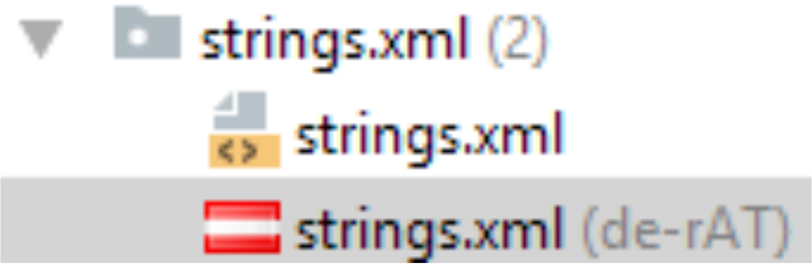


Show All Keys ▾ Show All Locales ▾ ?

Key

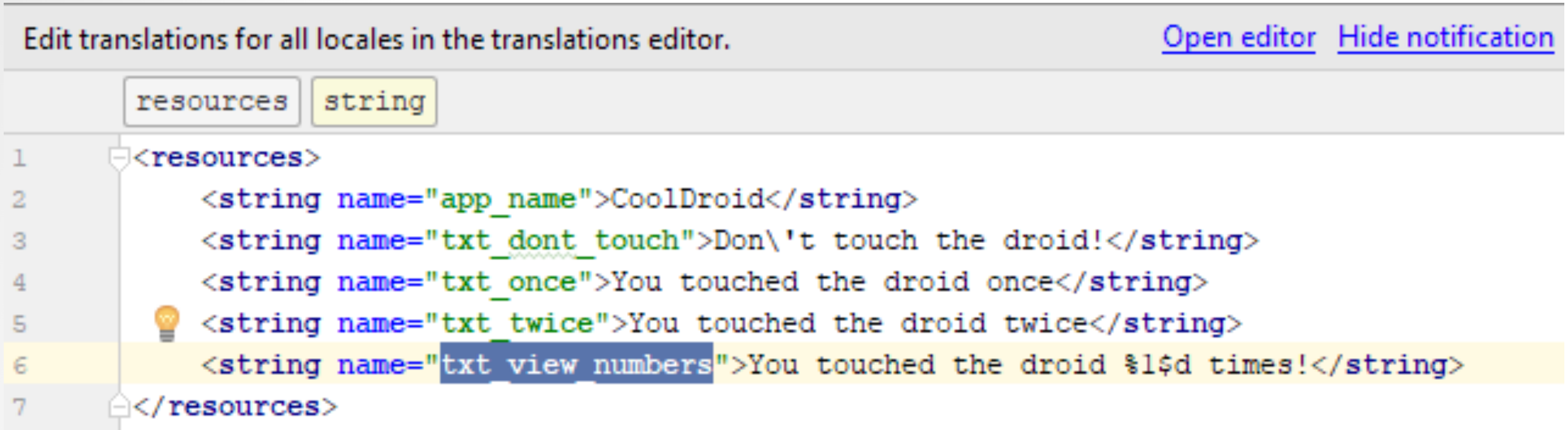
app_nam

- Gbaya (gba)
- Geez (gez)
- Georgian (ka)
- Georgian (ka) in Georgia (GE)
- German (de)
- German (de) in Austria (AT)



```
▼ strings.xml (2)
    strings.xml
    strings.xml (de-rAT)
```

Mehrsprachigkeit

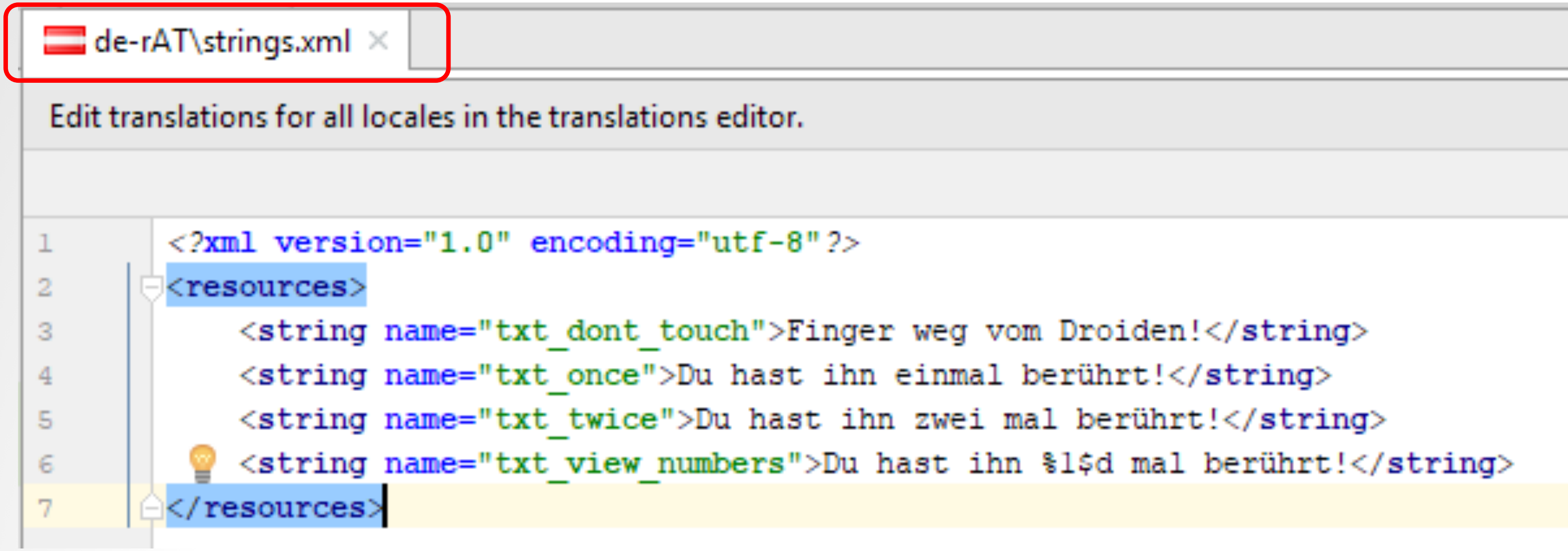


Edit translations for all locales in the translations editor.

```
<resources>
    <string name="app_name">CoolDroid</string>
    <string name="txt_dont_touch">Don\'t touch the droid!</string>
    <string name="txt_once">You touched the droid once</string>
    <string name="txt_twice">You touched the droid twice</string>
    <string name="txt_view_numbers">You touched the droid %1$d times!</string>
</resources>
```

Mehrsprachigkeit über Ressourcen

- Stringressourcen kopieren und übersetzen

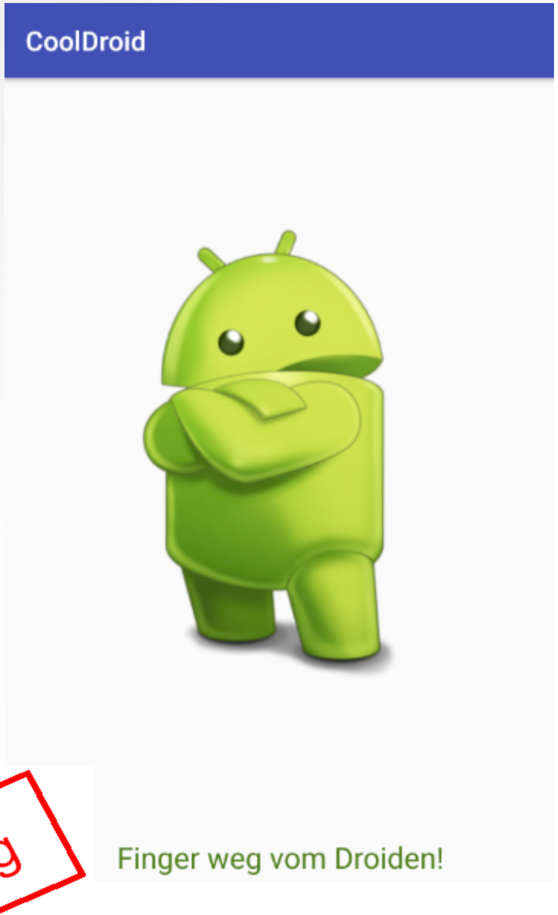


de-rAT\strings.xml ×

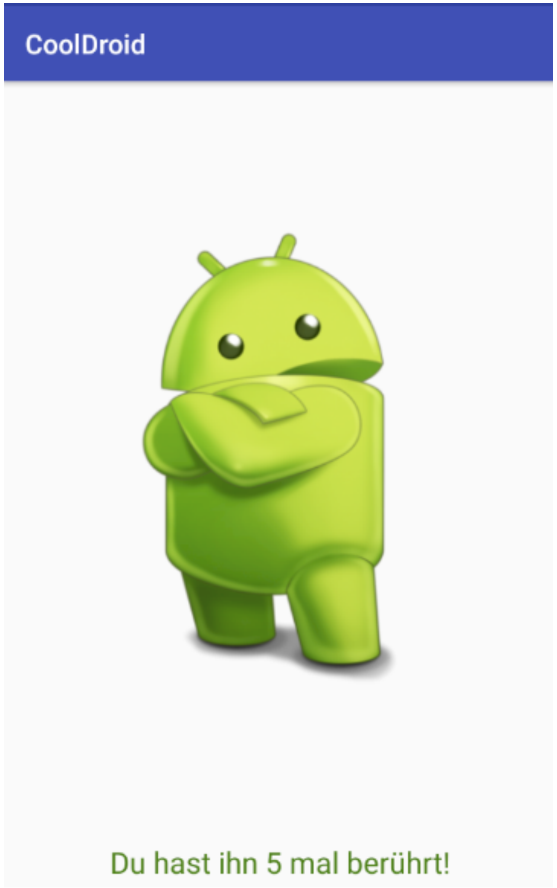
Edit translations for all locales in the translations editor.

```xml
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="txt_dont_touch">Finger weg vom Droiden!</string>
    <string name="txt_once">Du hast ihn einmal berührt!</string>
    <string name="txt_twice">Du hast ihn zwei mal berührt!</string>
    <string name="txt_view_numbers">Du hast ihn %1$d mal berührt!</string>
</resources>
```

Nun in deutscher Sprache


CoolDroid

Finger weg vom Droiden!

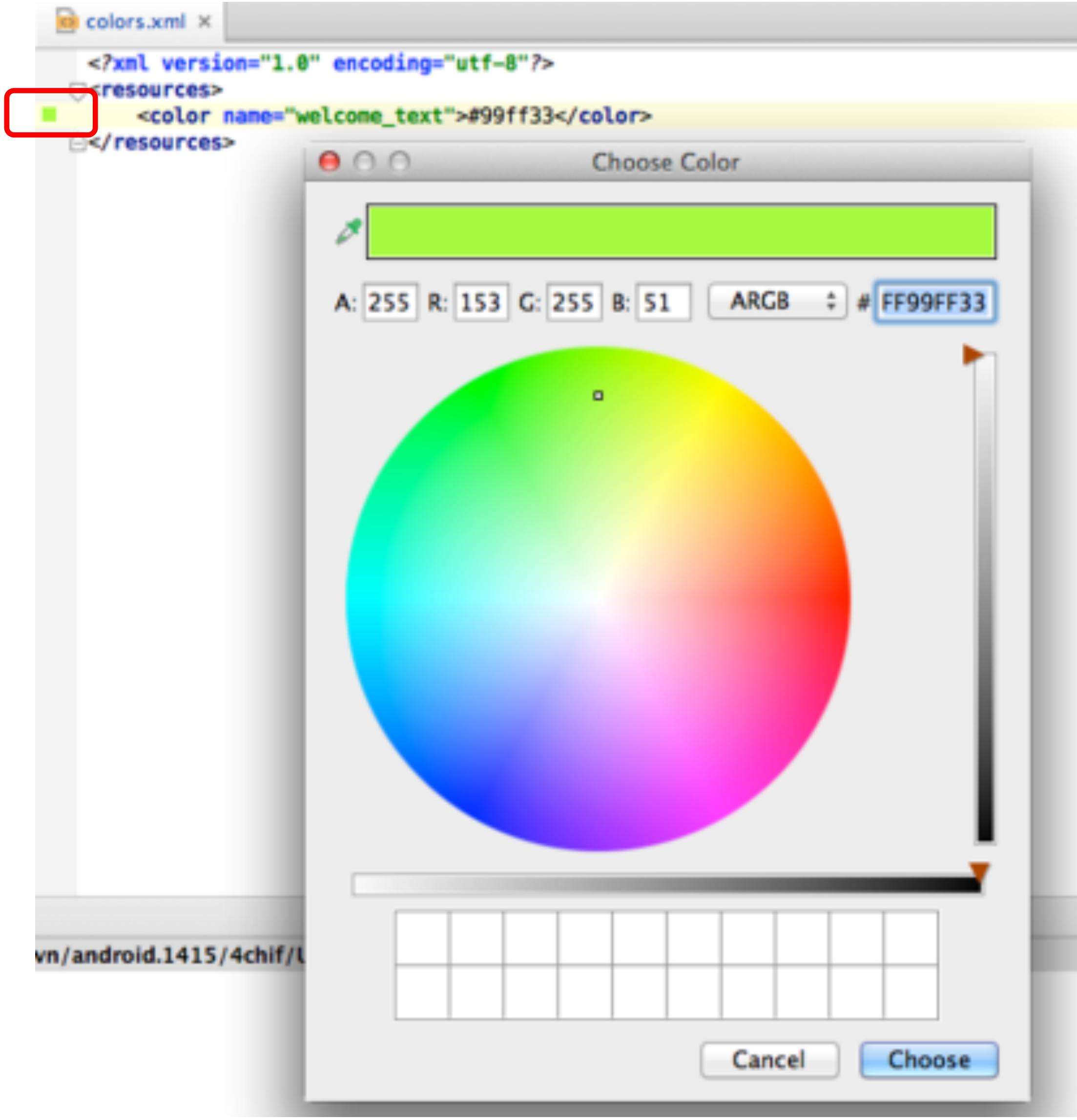

CoolDroid

Du hast ihn 5 mal berührt!

Übung

Farbauswahl in AndroidStudio

- Zugriff über Farbvorschau



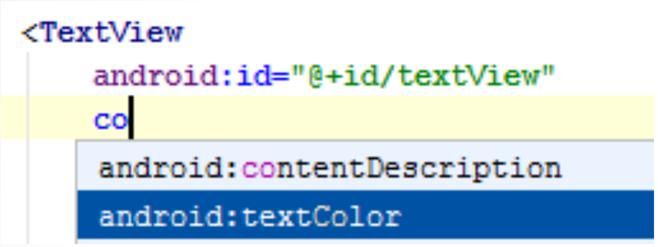
```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="welcome_text">#99ff33</color>
</resources>
```

A: 255 R: 153 G: 255 B: 51 ARGB # FF99FF33

vn/android.1415/4chif/l

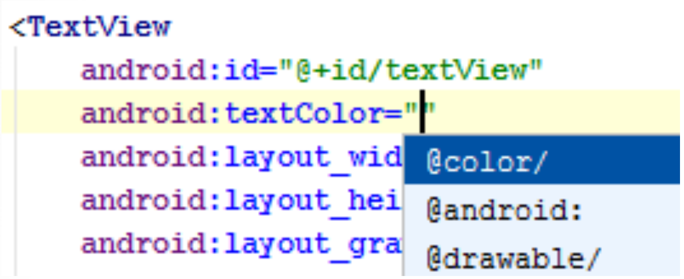
Zuweisung der Farbe in XML

- AutoComplete nutzen: „co“ reicht

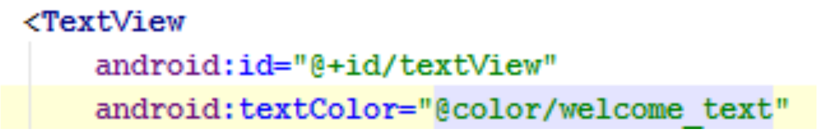


```
<TextView
    android:id="@+id/textView"
    co
     android:contentDescription
     android:textColor
```

- Auswahl der Möglichkeiten mit „Strg-Leertaste“



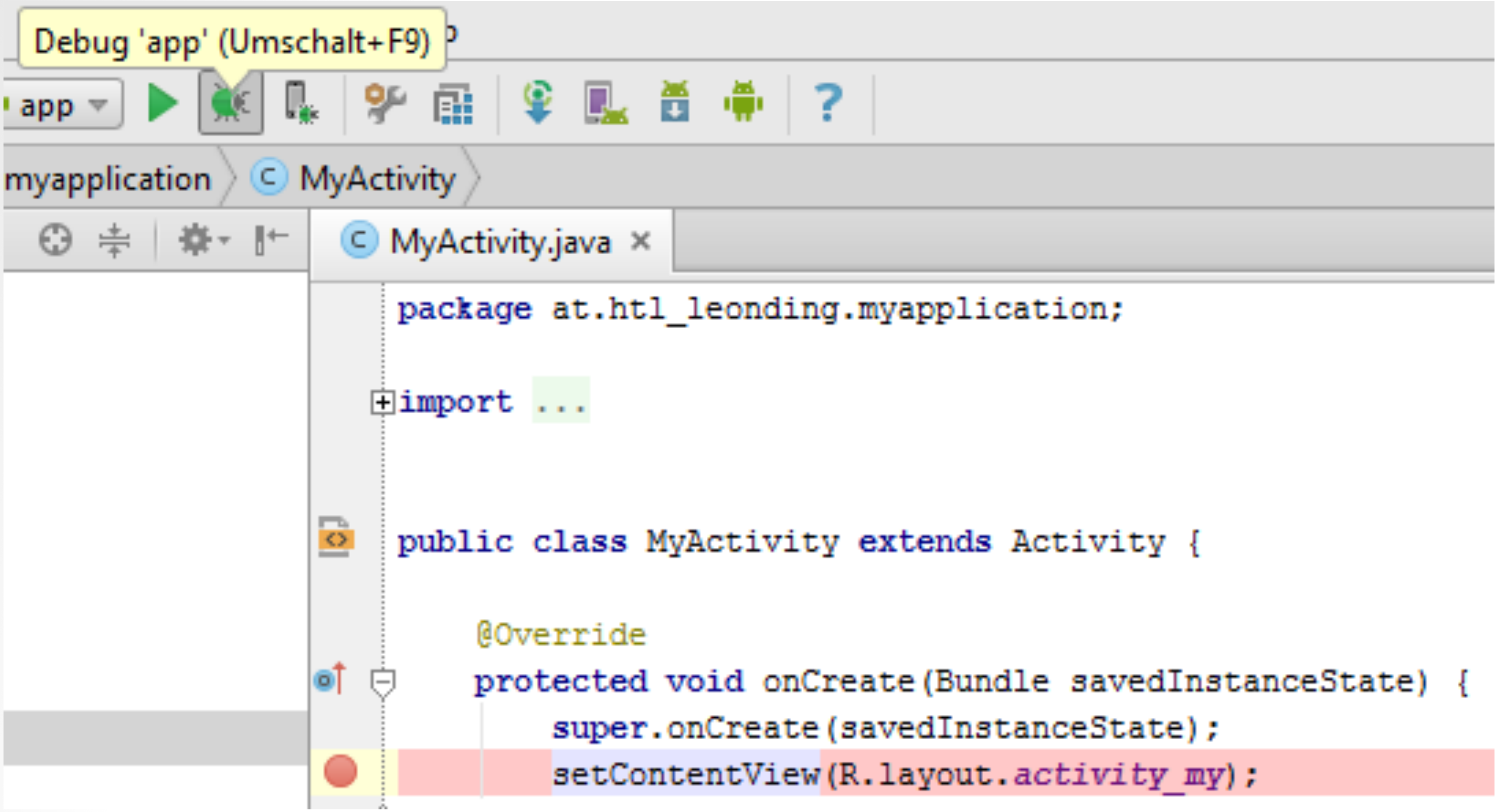
```
<TextView
    android:id="@+id/textView"
    android:textColor=""
    android:layout_wid @color/
    android:layout_hei @android:
    android:layout_gra @drawable/
```



```
<TextView
    android:id="@+id/textView"
    android:textColor="@color/welcome_text"
```

Debugger verwenden

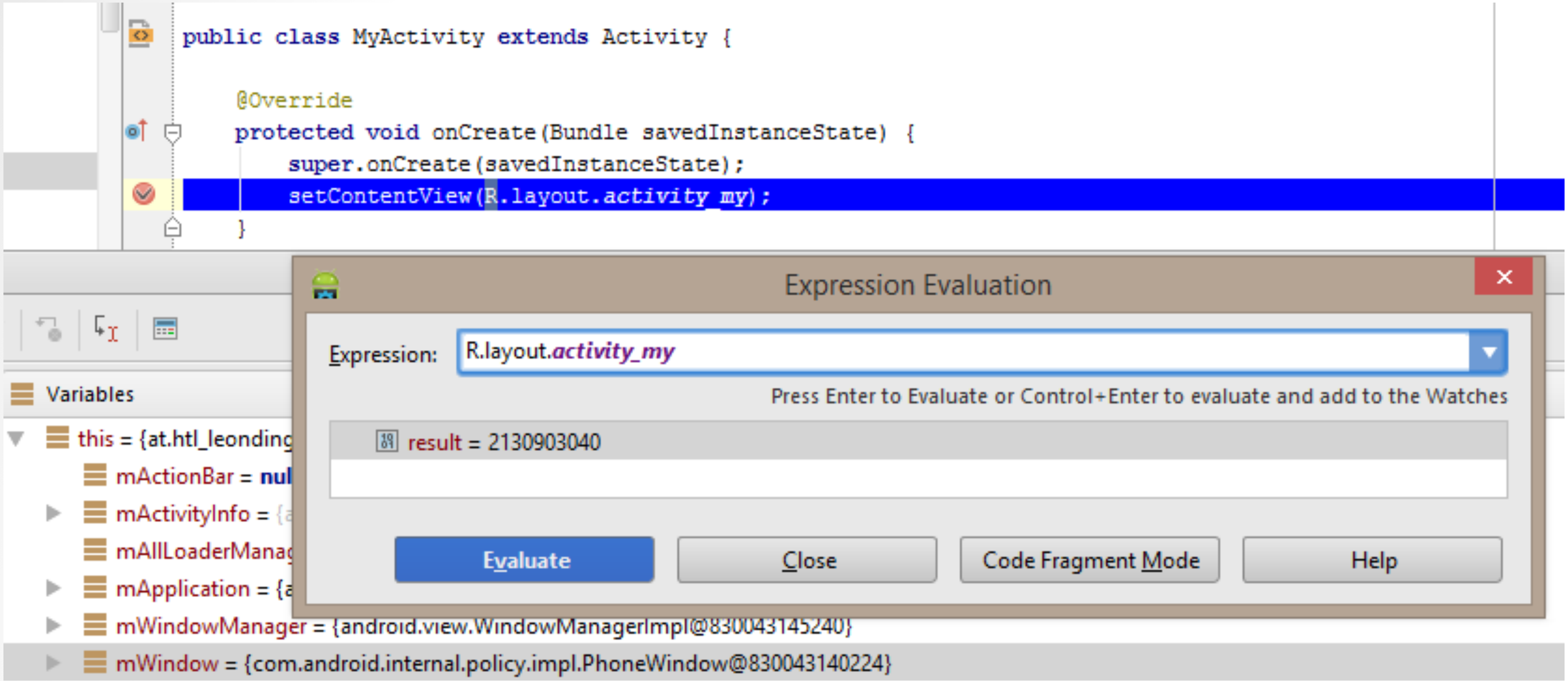
- Breakpoint setzen und Debugsession starten



```
package at.htl_leonding.myapplication;

import ...


public class MyActivity extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_my);
```

Variablen inspizieren

- Ausdrücke auswerten



```
public class MyActivity extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_my);
    }
```

Expression Evaluation

Expression: R.layout.activity_my

Press Enter to Evaluate or Control+Enter to evaluate and add to the Watches

result = 2130903040

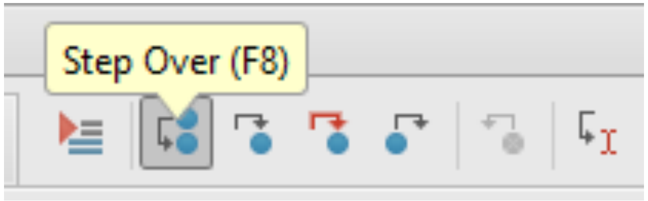
```
this = {at.htl_leonding
    mActionBar = nul
    mActivityInfo = {a
    mAllLoaderManag
    mApplication = {a
    mWindowManager = {android.view.WindowManagerImpl@830043145240}
    mWindow = {com.android.internal.policy.impl.PhoneWindow@830043140224}
```

HTL LEONDING

Debugger steuern

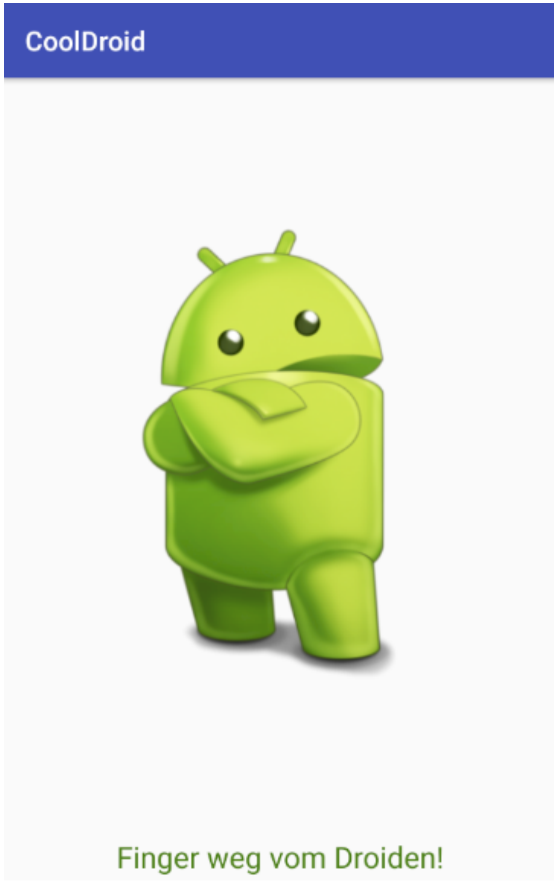
- F7: Step Into
- F8: Step Over
- Shift F8: Step Out
- F9: Continue

- Strg F2: Stop



Wechsel Portrait ⇔ Landscape

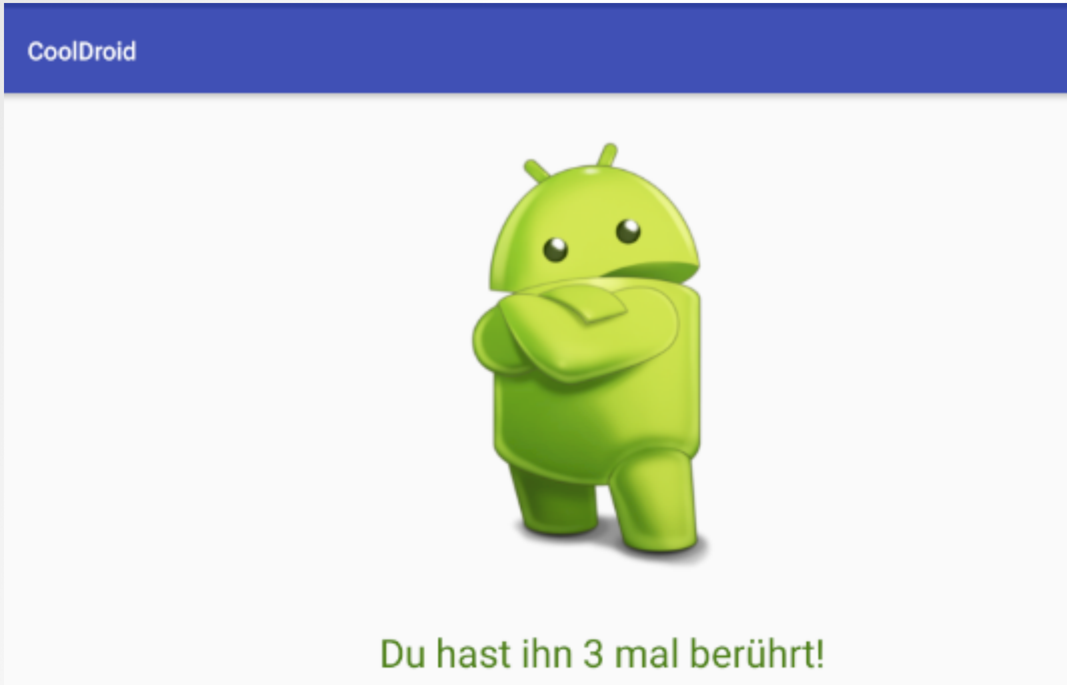
- Status geht verloren
  - Zähler beginnt wieder bei 0
- Behebung
  - Spezialistenaufgabe
  - Activities und deren Lebenszyklus



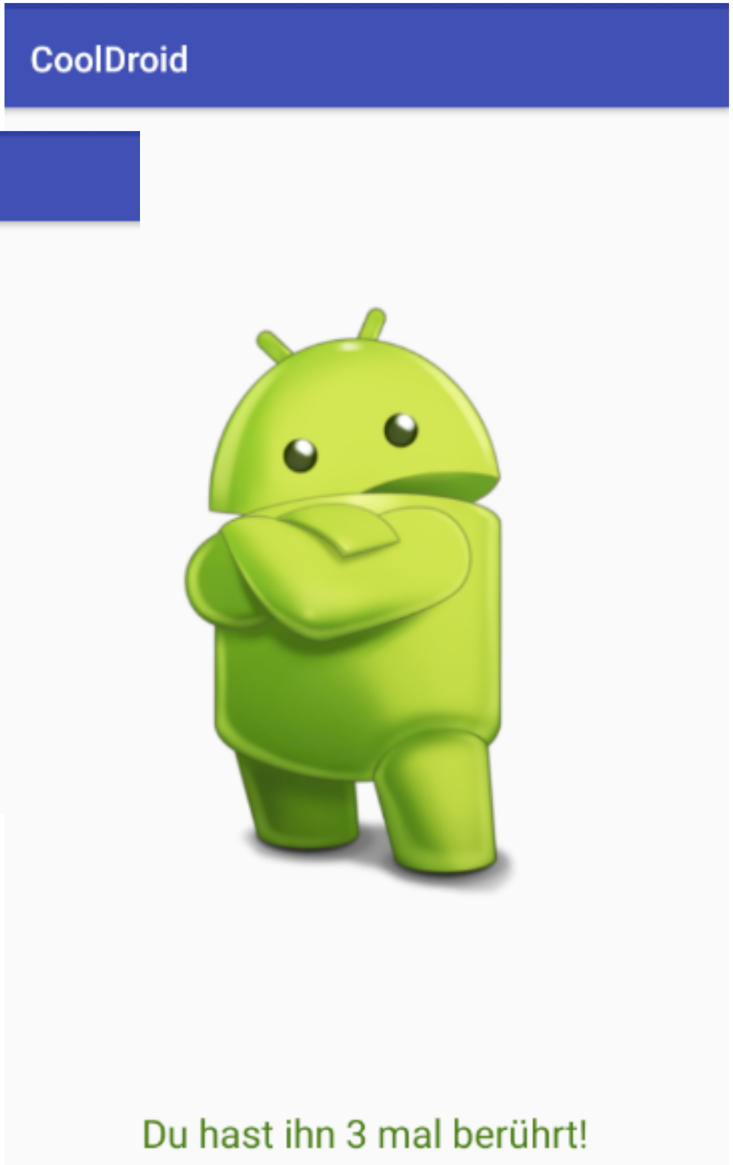
CoolDroid

Finger weg vom Droiden!

Übung



Du hast ihn 3 mal berührt!

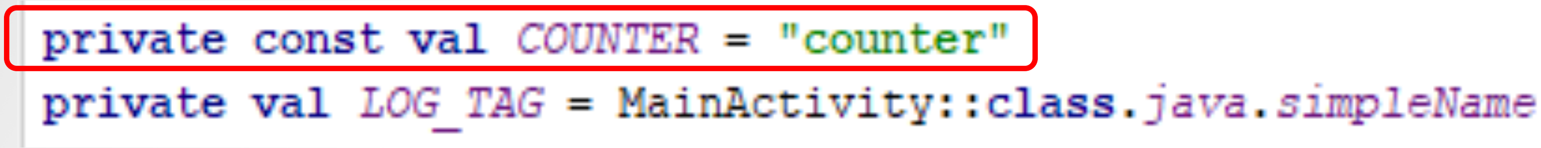


Du hast ihn 3 mal berührt!

CoolDroid Ende

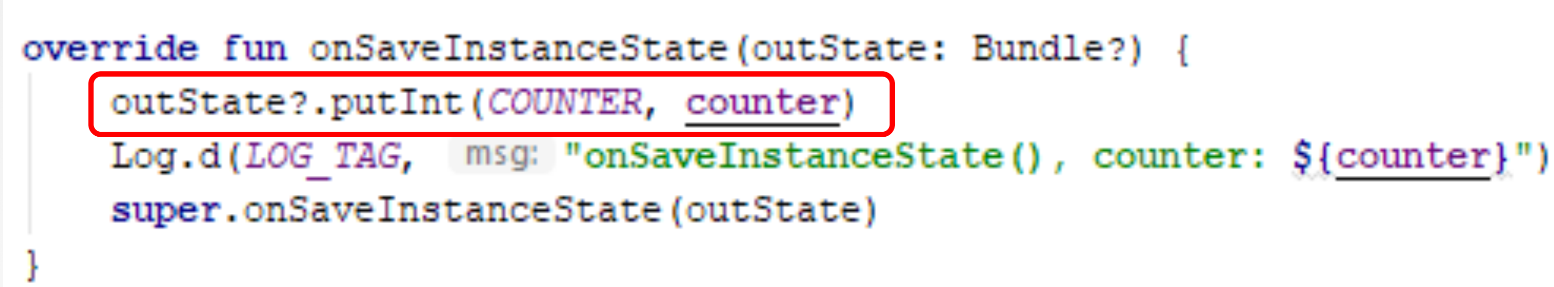
Status sichern und wiederherstellen

- Keys für Bundle und LOG_TAG



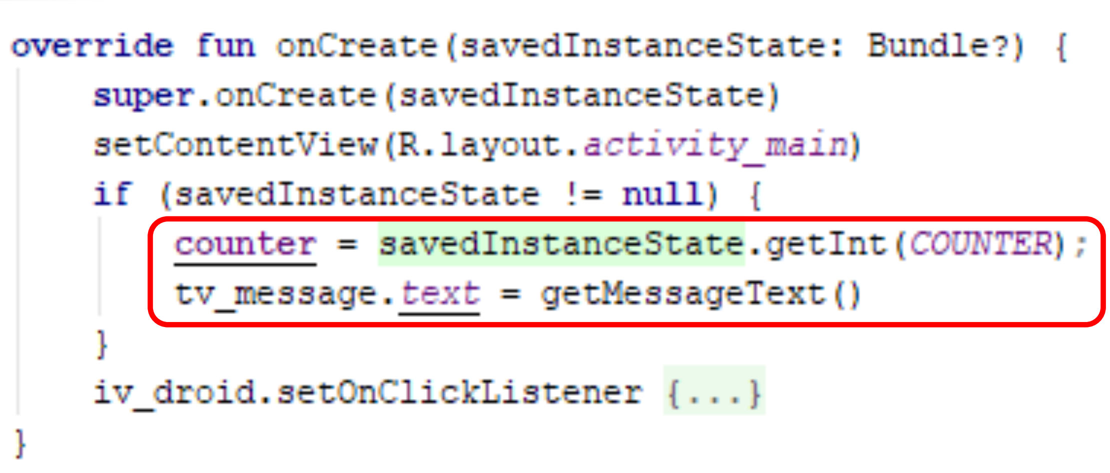
```
private const val COUNTER = "counter"
private val LOG_TAG = MainActivity::class.java.simpleName
```

- Bei Bedarf Status sichern



```
override fun onSaveInstanceState(outState: Bundle?) {
    outState?.putInt(COUNTER, counter)
    Log.d(LOG_TAG,  msg: "onSaveInstanceState(), counter: ${counter}")
    super.onSaveInstanceState(outState)
}
```

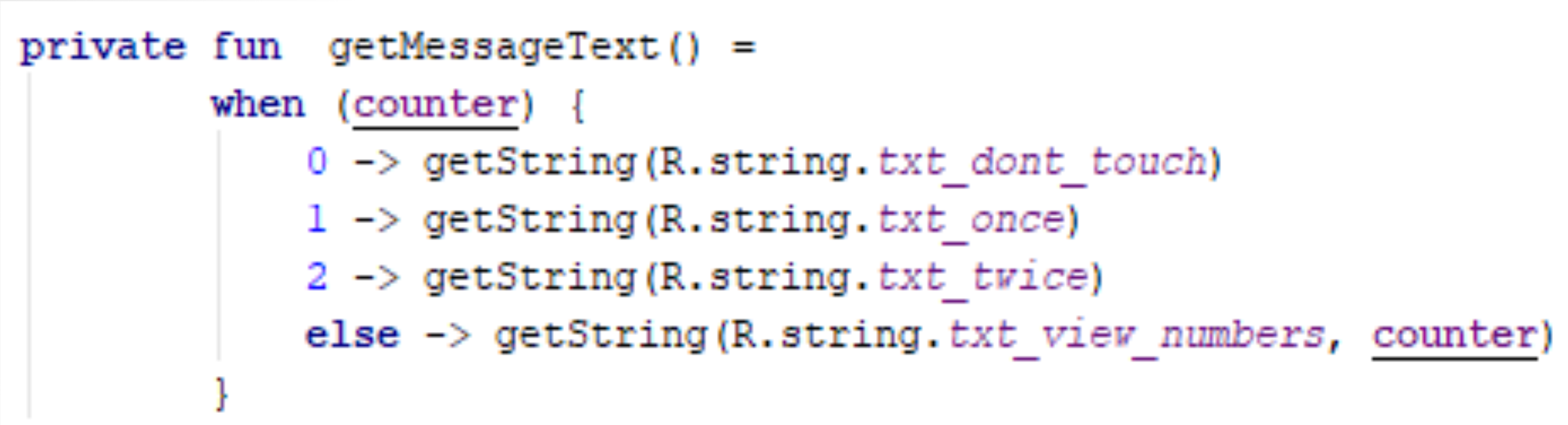
- … und wieder laden



```
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_main)
    if (savedInstanceState != null) {
        counter = savedInstanceState.getInt(COUNTER);
        tv_message.text = getMessageText()
    }
    iv_droid.setOnClickListener {...}
}
```

Codeverdopplung vermeiden

- Function mit einer Expression



```
private fun  getMessageText() =
        when (counter) {
            0 -> getString(R.string.txt_dont_touch)
            1 -> getString(R.string.txt_once)
            2 -> getString(R.string.txt_twice)
            else -> getString(R.string.txt_view_numbers, counter)
        }
```