



Mobile Programming

Catatan Kuliah #10

Alauddin Maulana Hirzan, M. Kom

0607069401

The background features a diagonal split between a teal upper-left section and a light gray lower-right section, with a white central area where the text is located.

Sumber Daya, **Style**, dan **Theme**



Sumber Daya, **Style**, dan **Theme**

Apa itu **Sumber Daya** #1

Sumber daya Android adalah berbagai jenis konten seperti gambar, file audio, string, warna, tata letak, animasi, dan banyak lagi yang digunakan dalam aplikasi Android. Sumber daya Android disimpan di direktori `res/` pada proyek Android dan diatur berdasarkan jenisnya.

Sumber daya adalah file tambahan dan konten statis yang digunakan kode, seperti bitmap, definisi tata letak, string antarmuka pengguna, instruksi animasi, dan banyak lagi.



Sumber Daya, **Style**, dan **Theme**

Apa itu **Sumber Daya** #2

Berikut adalah beberapa jenis sumber daya Android yang umum:

- ▶ Layout
- ▶ Drawable
- ▶ String
- ▶ Color
- ▶ Style
- ▶ Animation
- ▶ Menu
- ▶ Value



Sumber Daya, **Style**, dan **Theme**

Apa itu **Sumber Daya** #3

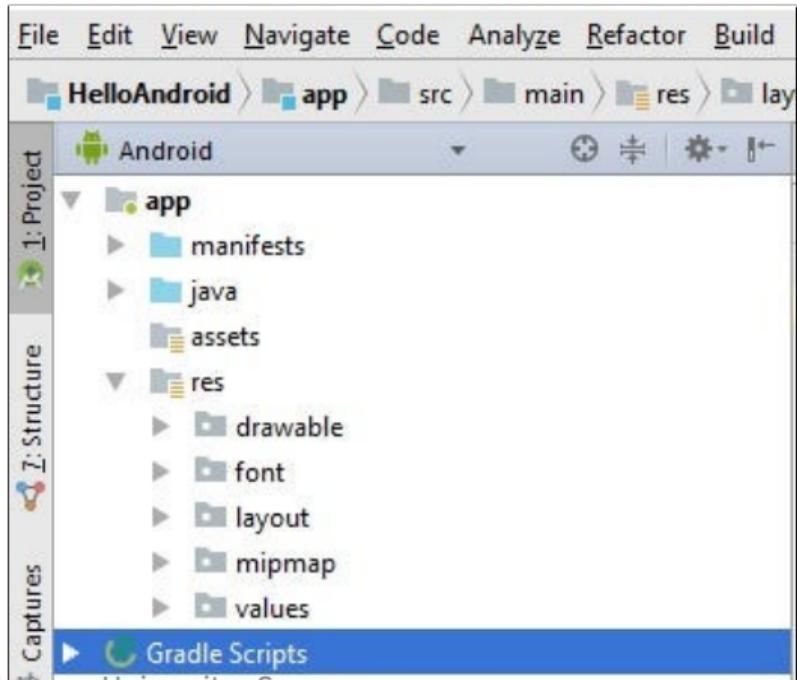
Programmer harus selalu mengeksternalisasi sumber daya aplikasi seperti gambar dan string dari kode, sehingga programmer dapat memeliharanya secara mandiri. Programmer juga harus menyediakan sumber daya alternatif untuk konfigurasi perangkat tertentu, dengan mengelompokkannya dalam direktori sumber daya yang diberi nama khusus. Pada saat runtime, Android menggunakan sumber daya yang sesuai berdasarkan konfigurasi saat ini.

Dalam contoh ini, direktori `res/` berisi semua sumber daya (dalam subdirektori): sumber daya gambar, dua sumber daya tata letak, direktori `mipmap/` untuk ikon peluncur, dan file sumber daya string. Nama direktori sumber daya adalah penting dan dijelaskan pada slide berikutnya



Sumber Daya, **Style**, dan **Theme**

Tampilan Sumber Daya





Sumber Daya, **Style**, dan **Theme**

Sumber Daya - Layout

Sumber daya ini merupakan hal yang paling vital dalam pembuatan aplikasi Android. Tanpa adanya sumber daya ini aplikasi tidak akan berjalan dengan baik.

Dalam pengembangan Android, tata letak digunakan untuk menentukan antarmuka pengguna (UI) aplikasi. Tata letak terdiri dari berbagai komponen, seperti tampilan dan grup tampilan, yang digunakan untuk membuat elemen visual aplikasi.

Layout biasanya dibuat dengan menggunakan Bahasa XML



Sumber Daya, **Style**, dan **Theme**

Sumber Daya - Drawable

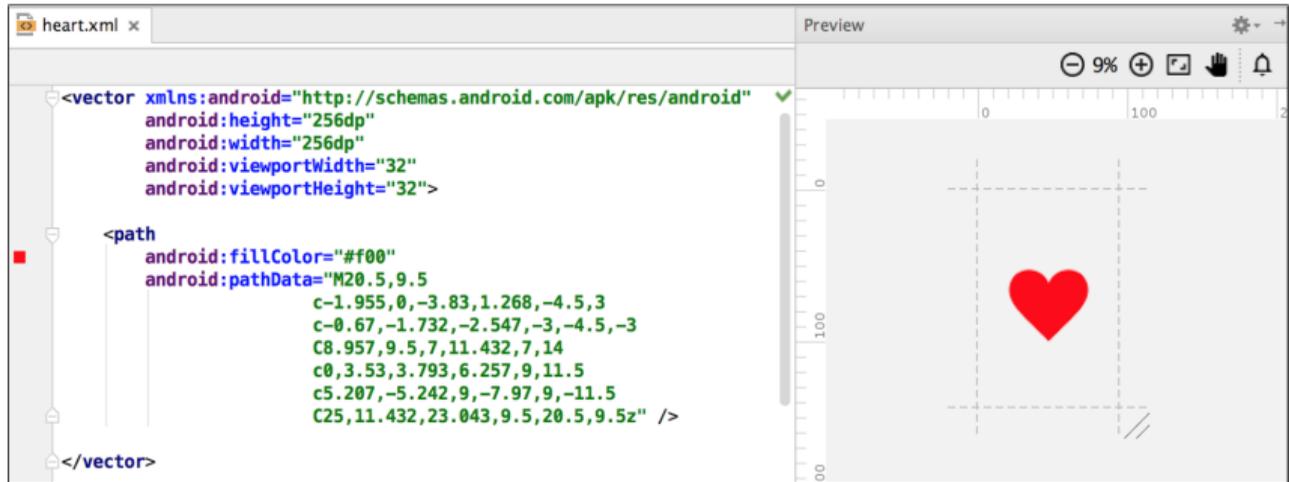
Dalam pengembangan Android, sumber daya yang dapat digambar adalah file XML atau bitmap yang digunakan untuk menentukan tampilan elemen visual dalam aplikasi. Drawable resource dapat digunakan untuk menentukan gambar latar belakang, ikon, tombol, dan elemen grafis lainnya.

Berbagai jenis **drawable**:

- ▶ Bitmap (Gambar Raster)
- ▶ Shape (XML Gambar Geometris)
- ▶ Vector (XML Gambar Vektor)
- ▶ Layer (XML Gambar Berlapis)
- ▶ State List (XML State Gambar)

Sumber Daya, **Style**, dan **Theme**

Tampilan Sumber Daya





Sumber Daya, **Style**, dan **Theme**

Sumber Daya - String #1

Dalam pengembangan Android, sumber daya string adalah string teks yang digunakan untuk mendefinisikan elemen UI seperti label, judul, dan pesan dalam aplikasi Android. Sumber daya string disimpan dalam file XML di direktori **res/values** pada proyek aplikasi.

Sumber daya string memiliki beberapa keunggulan dibandingkan dengan string yang dikodekan langsung ke dalam kode aplikasi.

Informasi

Hardcode Teks merujuk ke sebuah Teks yang dituliskan langsung ke Objek. Sedangkan **Softcode Teks** merujuk ke sebuah Teks yang dipanggil oleh Objek untuk ditampilkan



Sumber Daya, **Style**, dan **Theme**

Sumber Daya - String #2

Keuntungan menggunakan **Softcode Teks**:

- ▶ Pelokalan: Sumber daya string dapat dengan mudah diterjemahkan ke dalam berbagai bahasa dengan membuat file XML terpisah untuk setiap bahasa
- ▶ Konsistensi: Dengan menggunakan sumber daya string, pengembang dapat memastikan bahwa teks yang sama digunakan secara konsisten di seluruh aplikasi.
- ▶ Pemeliharaan: Sumber daya string dapat diperbarui dengan mudah hanya dengan mengedit file XML.



Sumber Daya, **Style**, dan **Theme**

Sumber Daya - String #3

Contoh sederhana memanggil String:

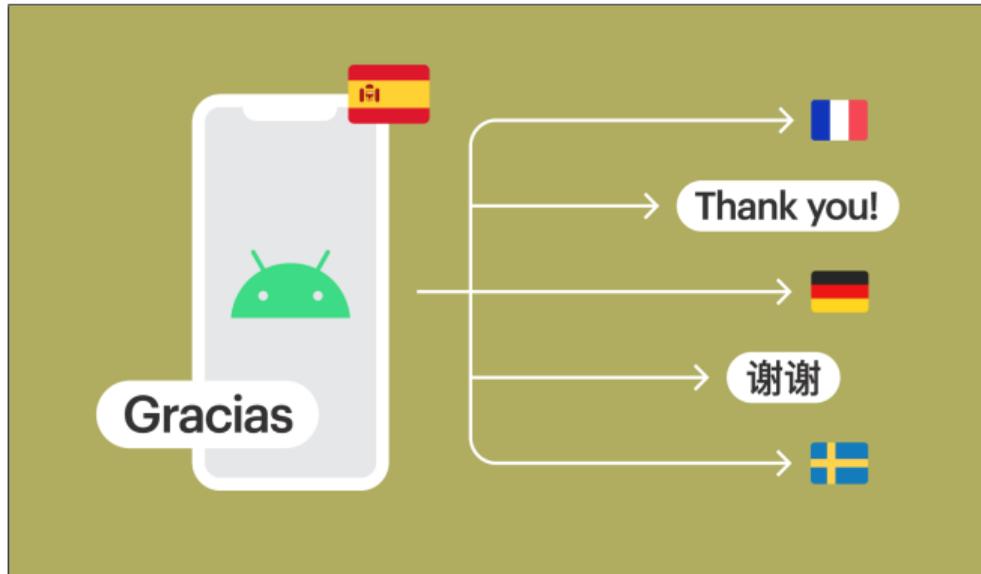
```
<TextView  
android:id="@+id/hello_world"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:text="@string/hello_world_text" />
```

File yang dipanggil berisikan

```
<resources>  
  
<string name="hello_world_text">Hello, world!</string>  
  
</resources>
```

Sumber Daya, **Style**, dan **Theme**

Tampilan Pelokalan





Sumber Daya, **Style**, dan **Theme**

Sumber Daya - Style & Color #1

Style adalah kumpulan atribut yang menentukan tampilan elemen UI dalam aplikasi Android. Gaya ditentukan dalam file XML di direktori `res/values` pada proyek aplikasi, dan dapat diterapkan pada setiap elemen UI atau seluruh aktivitas atau tema aplikasi.

Colors adalah sumber daya penting lainnya dalam pengembangan Android. Warna didefinisikan dalam file XML di direktori `res/values` pada proyek aplikasi, dan dapat direferensikan oleh sumber daya lain seperti `style`, `drawables`, dan `layout`.



Sumber Daya, **Style**, dan **Theme**

Sumber Daya - Style & Color #2

Contoh pembuatan sumber daya Colors dan Style:

```
<resources>  
<color name="my_color">#FF0000</color>  
<style name="MyButtonStyle">  
<item name="android:background">@color/my_color</item>  
</style>  
</resources>
```



Sumber Daya, **Style**, dan **Theme**

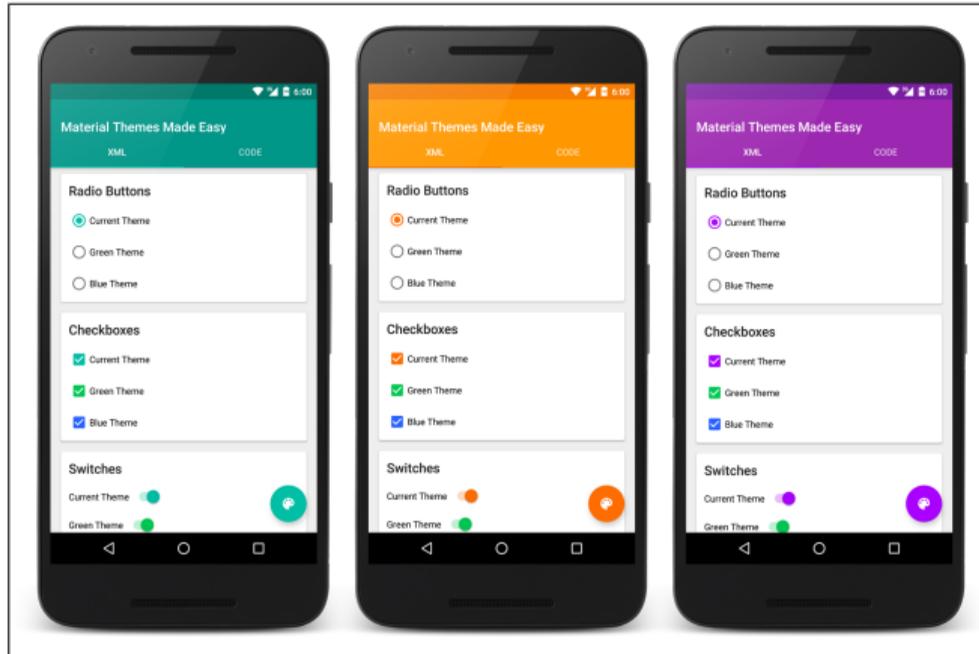
Sumber Daya - Style & Color #3

Implementasi sumber daya Colors dan Style:

```
<Button  
android:id="@+id/my_button"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:text="Click me!"/>
```

Sumber Daya, **Style**, dan **Theme**

Tampilan Colors dan Style





Sumber Daya, **Style**, dan **Theme**

Sumber Daya - Animation

Dalam pengembangan Android, animasi dapat digunakan untuk menambah daya tarik visual dan meningkatkan pengalaman pengguna aplikasi. Animasi dapat didefinisikan sebagai sumber daya dalam file XML di direktori res/anim pada proyek aplikasi.

Ada dua cara yang dapat digunakan untuk menganimasikan di aplikasi Android:

1. Property Animation
2. Frame Animation



Sumber Daya, **Style**, dan **Theme**

Sumber Daya - Property Animation #1

Jenis animasi yang memungkinkan pengembang untuk menganimasikan properti sebuah objek dari waktu ke waktu. Kelas `ObjectAnimator` digunakan untuk membuat animasi properti.

```
<set  
  xmlns:android="http://schemas.android.com/apk/res/android">  
  <objectAnimator android:propertyName="alpha"  
    android:duration="1000"  
    android:valueFrom="0"  
    android:valueTo="1" />  
</set>
```



Sumber Daya, **Style**, dan **Theme**

Sumber Daya - Property Animation #2

Dalam contoh ini, satu set animasi didefinisikan dengan satu `objectAnimator`. Atribut `propertyName` menentukan properti yang akan dianimasikan (dalam hal ini, `alpha` untuk mengontrol transparansi), dan atribut `duration` menentukan panjang animasi dalam milidetik. Atribut `valueFrom` dan `valueTo` menentukan nilai awal dan akhir properti.

Diimplementasikan dalam bentuk:

```
Animation myAnimation = AnimationUtils.loadAnimation(this,  
R.anim.my_animation);  
myView.startAnimation(myAnimation);
```



Sumber Daya, **Style**, dan **Theme**

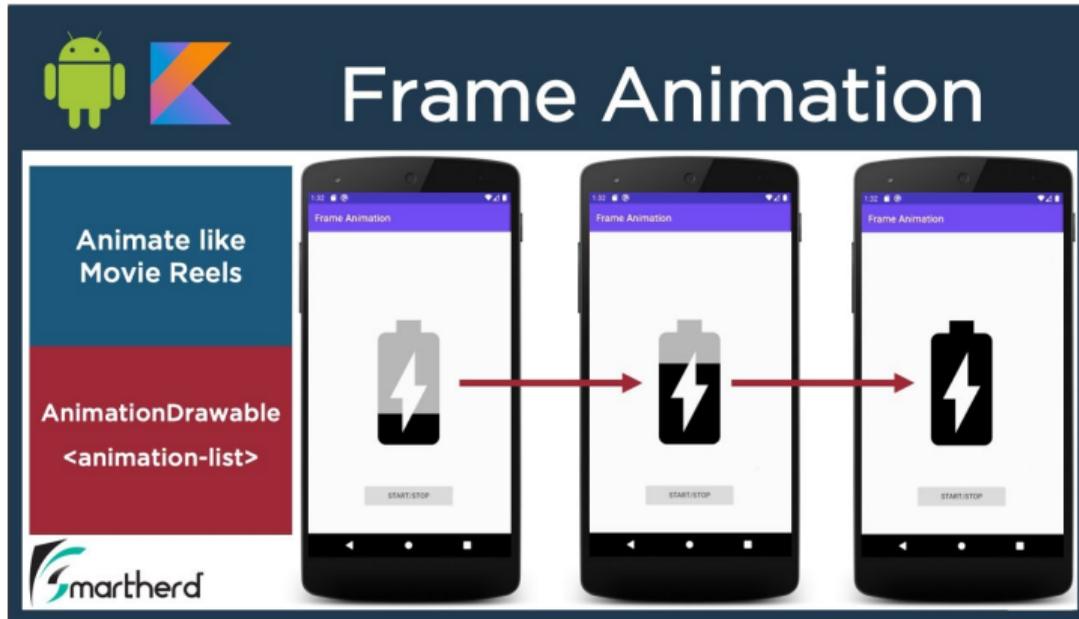
Sumber Daya - Frame Animation #1

Animasi bingkai adalah jenis animasi yang melibatkan tampilan urutan gambar secara berurutan untuk menciptakan ilusi gerakan. Animasi bingkai dapat didefinisikan sebagai serangkaian file gambar dalam direktori `res/drawable` pada proyek aplikasi, atau sebagai file XML dalam direktori `res/anim` yang mereferensikan serangkaian file gambar.

```
<animation-list xmlns:android="http://schemas.android.com/apk/res/android"
  android:oneshot="false">
  <item android:drawable="@drawable/frame1" android:duration="100" />
  <item android:drawable="@drawable/frame2" android:duration="100" />
  <item android:drawable="@drawable/frame3" android:duration="100" />
</animation-list>
```

Sumber Daya, **Style**, dan **Theme**

Frame Animation



The diagram illustrates frame animation on an Android device. It features a dark blue header with the Android logo and a colorful 'K' icon on the left, and the text 'Frame Animation' in white on the right. Below the header, there are three smartphone screens showing a battery icon with a lightning bolt. The battery level increases from left to right, as indicated by red arrows between the screens. Each screen has a 'START/STOP' button at the bottom. To the left of the screens is a red box containing the text 'Animate like Movie Reels' and 'AnimationDrawable <animation-list>'. At the bottom left of the diagram is the 'smartherd' logo.

