

应用资源

2018年5月23日 9:09

- Android应用的资源和作用
- Android应用的资源的存储方式
- 在XML布局文件中使用资源
- 在Java程序中使用资源
- 使用字符串资源
- 使用颜色资源
- 使用尺寸资源
- 使用数组资源
- 使用图片资源
- 使用各种Drawable资源
- 使用原始XML资源
- 使用布局资源
- 使用菜单资源
- 使用样式和主题资源
- 使用属性资源
- 使用原始资源
- 使用资源进行程序国际化
- 自适应不同屏幕的资源

源代码：

- 界面布局文件：xml，文件中每个标签对应于相应的View标签
- Java源文件：四大组件（Activity、Service、BroadcastReceiver、ContentProvider）
- 资源文件：以xml文件为主，还包括图片资源

应用资源：

- assets目录：无法通过R资源清单类访问的原生资源
- res目录：可通过R资源清单类访问的资源

1、在Java代码中使用资源清单项

[<package_name>].R.<resource_type>.<resource_name>

2、在Java代码中访问实际资源

“Android资源访问总管家” —— Resources类

getXxx (int id) ：根据资源清单ID来获取实际资源

getAssets () ：获取访问/assets/目录下资源的AssetManager对象

3、在XML文件中使用资源

@[<package_name>:]<resource_type>/<resource_name>

字符串、颜色、尺寸资源： (/res/values/)

根元素： <resources...>

1、字符串资源

```
<!-- 定义一个字符串， 名称为 hello, 字符串值为 Hello World, ValuesResTest! -->
<string name="hello">Hello World, ValuesResTest!</string>
```

2、颜色资源

```
<!-- 定义一个颜色， 名称为 cl, 颜色为红色 -->
<color name="cl">#F00</color>
```

3、尺寸资源

```
<!-- 定义一个尺寸， 名称为 spacing, 尺寸大小为8dp-->
<dimen name="spacing">8dp</dimen>
```

数组 (Array) 资源：

采用位于/res/values目录下的arrays.xml文件来定义数组资源

Java类： [<package_name>].R.array.array_name

XML文件： @ [<package_name>:]array/array_name

Drawable资源： (/res/drawable)

Java类： [<package_name>].R.drawable.<file_name>

XML文件： @ [<package_name>:]drawable/file_name

- stateListDrawable资源

根元素为<selector.../>， 包含多个<item.../>元素 {让各种组件的背景、前景随状态的变化而切换}

- LayerDrawable资源

根元素为<layer-list.../>,包含多个<item.../>元素{系统会按这些Drawable对象的数组顺序来绘制}

- shapeDrawable资源

根元素为<shape.../>， 用于定义一个基本的几何图形

- clipDrawable资源

根元素为<clip.../>,代表从其他位图上截取的一个“图片片段”

- AnimationDrawable资源

根元素为<set.../>,表示一个动画

Property Animation (属性动画) 资源：

- * <set.../>: 父元素， 该元素定义的资源代表AnimatorSet对象
- * <objectAnimator.../>:用于定义ObjectAnimator动画
- * <animator.../>:用于定义valueAnimator动画

原始XML资源：

Android应用的初始化配置信息、应用相关的数据资源

布局 (Layout) 资源:

根元素是各种布局管理器, 如LinearLayout、TableLayout、FrameLayout等, 接着在该布局管理器中定义各种View组件

菜单 (Menu) 资源:

根元素是< menu.../>元素, 无需指定任何属性

样式 (Style) 和主题 (Theme) 资源:

- * 样式资源文件的根元素是<resources.../>元素, 包含多个<style.../>子元素 (定义一个样式 {name和parent属性}), <style.../>包含多个<item.../>子元素 (定义一个格式项)
- * 主题资源与样式资源相似

属性 (Attribute) 资源:

- attr子元素: 定义一个属性
- declare-styleable元素: 定义一个styleable对象, 每个styleable对象就是一组attr属性的集合

Res文件夹:

res类资源放在工程根目录的res子目录下, 它里面保存的文件大多数都会被编译, 并且都会被赋予资源ID。这样我们就可以在程序中通过ID来访问res类的资源。res类资源按照不同的用途可以进一步划分为以下9种子类型:

--animator。这类资源以XML文件保存在res/animator目录下, 用来描述属性动画。属性动画通过改变对象的属性来实现动画效果, 例如, 通过不断地修改对象的坐标值来实现对象移动动画, 又如, 通过不断地修改对象的Alpha通道值来实现对象的渐变效果。

--anim。这类资源以XML文件保存在res/anim目录下, 用来描述补间动画。补间动画和属性动画不同, 它不是通过修改对象的属性来实现, 而是在对象的原来形状或者位置的基础上实现一个变换来得到的, 例如, 对对象施加一个旋转变换, 就可以获得一个旋转动画, 又如, 对对象实施一个缩放变换, 就可以获得一个缩放动画。从数学上来讲, 就是在对象的原来形状或者位置的基础上施加一个变换矩阵来实现动画效果。注意, 在动画的执行过程中, 对象的属性是始终保持不变的, 我们看到的只不过是它的一个变形副本。

--color。这类资源以XML文件保存在res/color目录下, 用描述对象颜色状态选择子。例如, 我们可以定义一个选择子, 规定一个对象在不同状态下显示不同的颜色。对象的状态可以划分为pressed、focused、selected、checkable、checked、enabled和window_focused等7种。

--drawable。这类资源以XML或者Bitmap文件保存在res/drawable目录下, 用来描述可绘制对象。例如, 我们可以在里面放置一些图片 (.png, .9.png, .jpg, .gif), 来作为程序界面视图的背景图。注意, 保存在这个目录中的Bitmap文件在打包的过程中, 可能会被优化

的。例如，一个不需要多于256色的真彩色PNG文件可能会被转换成一个只有8位调色板的PNG面板，这样就可以无损地压缩图片，以减少图片所占用的内存资源。

--layout。这类资源以XML文件保存在res/layout目录下，用来描述应用程序界面布局。

--menu。这类资源以XML文件保存在res/menu目录下，用来描述应用程序菜单，例如，Options Menu、Context Menu和Sub Menu。

--raw。这类资源以任意格式的文件保存在res/raw目录下，它们和assets类资源一样，都是原装不动地打包在apk文件中的，不过它们会被赋予资源ID，这样我们就可以在程序中通过ID来访问它们。例如，假设在res/raw目录下有一个名称为filename的文件，并且它在编译的过程，被赋予的资源ID为R.raw.filename，那么就可以使用以下代码来访问它：

--values。这类资源以XML文件保存在res/values目录下，用来描述一些简单值，例如，数组、颜色、尺寸、字符串和样式值等，一般来说，这六种不同的值分别保存在名称为arrays.xml、colors.xml、dimens.xml、strings.xml和styles.xml文件中。

--xml。这类资源以XML文件保存在res/xml目录下，一般就是用来描述应用程序的配置信息。