Demo55-基本动画Demo

1. activity\_splash.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:id="@+id/rl\_splash\_root"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="@drawable/splash\_bg\_newyear"**>  
  
  
 <**ImageView  
 android:id="@+id/img\_sheep"  
 android:layout\_centerInParent="true"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:src="@drawable/splash\_sheep\_newyear"** />  
  
  
</**RelativeLayout**>

1. SplashActivity.java

**import** android.app.Activity;  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.view.animation.AlphaAnimation;  
**import** android.view.animation.Animation;  
**import** android.view.animation.AnimationSet;  
**import** android.view.animation.RotateAnimation;  
**import** android.view.animation.ScaleAnimation;  
**import** android.widget.ImageView;  
**import** android.widget.RelativeLayout;  
**import** com.example.newbeijing.activity.GuideActivity;  
**import** com.example.newbeijing.activity.MainActivity;  
**import** com.example.newbeijing.utils.CacheUtil;  
  
*/\*\*  
 \* 面向对象的本质：构造对象，调用象方  
 \*/***public class** SplashActivity **extends** Activity {  
  
  
 *//静态常量做key，可以保持不变，刚好符合key的需求，且多处可用，所以使用静态* **public static final** String ***START\_MAIN*** = **"start\_main"**;  
  
 *//view声明* **private** RelativeLayout **rl\_splash\_root**;  
 **private** ImageView **img\_sheep**;  
  
 *//继承重写Activity的生命周期方法：onCreate()，且super先调用父类的onCreate做一些初始化操作吧（可查一下）* @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
  
 *//初始化view* initView();  
  
 *//初始化Data(准备Data + 把Data填充到view显示出来)* initData();  
  
  
  
  
  
 }  
  
  
 *//初始化view* **private void** initView() {  
  
 *//将布局activity\_splash.xml初始化成主view* setContentView(R.layout.***activity\_splash***);  
  
 *//初始化view* **rl\_splash\_root** = (RelativeLayout) findViewById(R.id.***rl\_splash\_root***);  
 **img\_sheep** = (ImageView) findViewById(R.id.***img\_sheep***);  
 }  
  
  
  
 *//初始化Data（构造动画对象），填充数据显示出来* **private void** initData() {  
  
  
 */\*\*  
 \* 构造AlphaAnimation渐变动画对象，目的是后期构造AnimationSet对象  
 \* 类似Builder建造特殊对象的思想  
 \* 渐变就是在可见与不可见之间慢慢切换  
 \*/  
 //构造器（起始透明度0不可见，末态透明度1完全可见）* AlphaAnimation aa=**new** AlphaAnimation(0,1);  
 *//setter设置动画对象播放完后停在 播完的状态* aa.setFillAfter(**true**);  
  
  
 */\*\*  
 \* 构造ScaleAnimation缩放动画对象，目的是后期构造AnimationSet对象  
 \* 类似Builder建造特殊对象的思想  
 \* 缩放就是从大到小，或从小到大  
 \*/  
 //构造器：一般是fromX，Y一样，然后toX，Y一样，这样才不会变形，0.5,2,0.5，2，即从本身的0.5倍变成2倍，后面是旋转中心的位置，下面这个旋转中心值常用* ScaleAnimation sa=**new** ScaleAnimation(0,1,0,1, Animation.***RELATIVE\_TO\_SELF***,0.5f,Animation.***RELATIVE\_TO\_SELF***,0.5f);  
 *//setter设置动画对象播放完后停在 播完的状态* sa.setFillAfter(**true**);  
  
  
 */\*\*  
 \* 构造ScaleAnimation缩放动画对象，目的是后期构造AnimationSet对象  
 \* 类似Builder建造特殊对象的思想  
 \* 缩放就是从大到小，或从小到大  
 \*/  
 //构造器：从0度开始旋转360度结束，旋转中心跟上面一样，X轴，左到右1，Y轴，上到1下* RotateAnimation ra=**new** RotateAnimation(0,360,Animation.***RELATIVE\_TO\_SELF***,0.5f,Animation.***RELATIVE\_TO\_SELF***,0.5f);  
 *//setter设置动画对象播放完后停在 播完的状态* ra.setFillAfter(**true**);  
  
 */\*\*  
 \* 构造AnimationSet对象，叠加动画效果  
 \* setDuration只是设置了整个动画的播放时间，  
 \* 但是整个过程的播放速率，例如先快后慢，先慢后快，匀速，  
 \* 是需要插值器Interpolator来设置的，不设置的话，默认匀速  
 \*/* AnimationSet set=**new** AnimationSet(**false**);  
 *//setter调用象方初始化内部属性对象值，两个都可理解为setter* set.addAnimation(aa);  
 set.addAnimation(sa);  
 set.addAnimation(ra);  
 set.setDuration(1000);  
 ra.setFillAfter(**true**);  
  
 */\*\*  
 \* 属于将Data填充到view显示出来，通过调用象方  
 \*/  
 //开启动画* **rl\_splash\_root**.startAnimation(set);  
 *//动画状态监听触发方法是写在set类内部的，所以这里的监听器对象本质上并不是监听器  
 //而是一个监听结果处理类的对象，会被在set内部的触发方法里调用此对象的象方，所以叫回调* set.setAnimationListener(**new** MyAnimationListener());  
 }  
  
  
  
 */\*\*  
 \* 重写动画监听结果处理类，并不是监听触发类，是结果处理类来的，所以才叫回调  
 \*/* **class** MyAnimationListener **implements** Animation.AnimationListener{  
  
 */\*\*  
 \* 监听动画的接口，动画开始时，动画结束时，动画重复播放时，学会看方法名，一看就知道  
 \** ***@param animation*** *\*/* @Override  
 **public void** onAnimationStart(Animation animation) {  
  
 }  
  
 @Override  
 **public void** onAnimationEnd(Animation animation) {  
  
 **boolean** isStartMain= CacheUtil.*getBoolean*(SplashActivity.**this**, ***START\_MAIN***);  
 Intent intent;  
 **if** (isStartMain){  
  
 *//进去过主页面，则直接进入主页面  
 // 跳转到MainActivity* intent=**new** Intent(SplashActivity.**this**, MainActivity.**class**);  
  
 }**else** {  
 *//没进去过主页面，则先进入引导页面* intent=**new** Intent(SplashActivity.**this**, GuideActivity.**class**);  
  
 }  
 startActivity(intent);  
 finish();  
 }  
  
 @Override  
 **public void** onAnimationRepeat(Animation animation) {  
  
 }  
 }  
  
  
  
  
}