<http://blog.csdn.net/xiaxiazaizai01/article/details/52232846>

1. 在values下的styles.xml下进行dialog样式设计

<!-- dialog样式 -->

<style name="MyDialog" parent="@android:style/Theme.Dialog">

<item name="android:windowFrame">@null</item>

<item name="android:windowIsFloating">true</item>

<item name="android:windowIsTranslucent">true</item>

<item name="android:windowNoTitle">true</item>

<item name="android:windowBackground">@android:color/transparent</item>

<item name="android:backgroundDimEnabled">true</item>

<item name="android:backgroundDimAmount">0.6</item>

</style>

1. 自定义MyDialog类

public class MyDialog extends Dialog {

private Context context;

private static MyDialog dialog;

private ImageView ivProgress;

public MyDialog(Context context) {

super(context);

this.context = context;

}

public MyDialog(Context context, int themeResId) {

super(context, themeResId);

this.context = context;

}

//显示dialog的方法

public static MyDialog showDialog(Context context){

dialog = new MyDialog(context, R.style.MyDialog);//dialog样式

dialog.setContentView(R.layout.dialog\_layout);//dialog布局文件

dialog.setCanceledOnTouchOutside(false);//点击外部不允许关闭dialog

return dialog;

}

@Override

public void onWindowFocusChanged(boolean hasFocus) {

super.onWindowFocusChanged(hasFocus);

if(hasFocus && dialog != null){

ivProgress = (ImageView) dialog.findViewById(R.id.ivProgress);

Animation animation = AnimationUtils.loadAnimation(context, R.anim.dialog\_progress\_anim);

ivProgress.startAnimation(animation);

}

}

}

1. 添加dialog布局文件dialog\_layout

<?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"

>

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:gravity="center"

android:padding="15dp"

android:layout\_centerInParent="true"

android:background="@drawable/shape\_dialog\_bg"

android:orientation="vertical">

<RelativeLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center">

<ImageView

android:id="@+id/ivBackGround"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:background="@drawable/loading\_background"

android:layout\_centerInParent="true"

/>

<ImageView

android:id="@+id/ivProgress"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:background="@drawable/loading\_progress"

android:layout\_centerInParent="true"

/>

</RelativeLayout>

<TextView

android:id="@+id/tvText"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="加载中，请稍后..."

android:textSize="16sp"

android:layout\_marginTop="5dp"

/>

</LinearLayout>

</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"

>

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:gravity="center"

android:padding="15dp"

android:layout\_centerInParent="true"

android:background="@drawable/shape\_dialog\_bg"

android:orientation="vertical">

<RelativeLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center">

<ImageView

android:id="@+id/ivBackGround"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:background="@drawable/loading\_background"

android:layout\_centerInParent="true"

/>

<ImageView

android:id="@+id/ivProgress"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:background="@drawable/loading\_progress"

android:layout\_centerInParent="true"

/>

</RelativeLayout>

<TextView

android:id="@+id/tvText"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="加载中，请稍后..."

android:textSize="16sp"

android:layout\_marginTop="5dp"

/>

</LinearLayout>

</RelativeLayout>

1. 穿件动画文件anim，设置dialog的动画效果

<?xml version="1.0" encoding="utf-8"?>

<set xmlns:android="http://schemas.android.com/apk/res/android"

android:shareInterpolator="false">

<!-- 如果不想共享一个interpolator(动画变化的速率)，则设置android:shareInterpolator="false"，并且需要在每一个动画效果处添加interpolator -->

<rotate

android:fromDegrees="0"

android:toDegrees="+360"

android:pivotX="50%"

android:pivotY="50%"

android:repeatCount="-1"

android:repeatMode="restart"

android:interpolator="@android:anim/linear\_interpolator"

android:duration="800"

/>

<!--

fromDegrees 动画开始时的角度

toDegrees 动画结束时物件的旋转角度,正代表顺时针

pivotX 属性为动画相对于物件的X坐标的开始位置

pivotY 属性为动画相对于物件的Y坐标的开始位置

repeatCount 动画重复次数，-1表示无限执行

repeatMode 重复类型，有两个值，reverse表示倒序回放，restart表示从头播放

interpolator 动画变化速率，这里是匀速

-->

</set>

1. 调用MyDialog

public class MainActivity extends AppCompatActivity {

private Button btn;

MyDialog dialog;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

btn = (Button) findViewById(R.id.btnClick);

btn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

dialog = MyDialog.showDialog(MainActivity.this);

dialog.show();

}

});

}

}