使用StatusBarUtil 来实现沉浸式

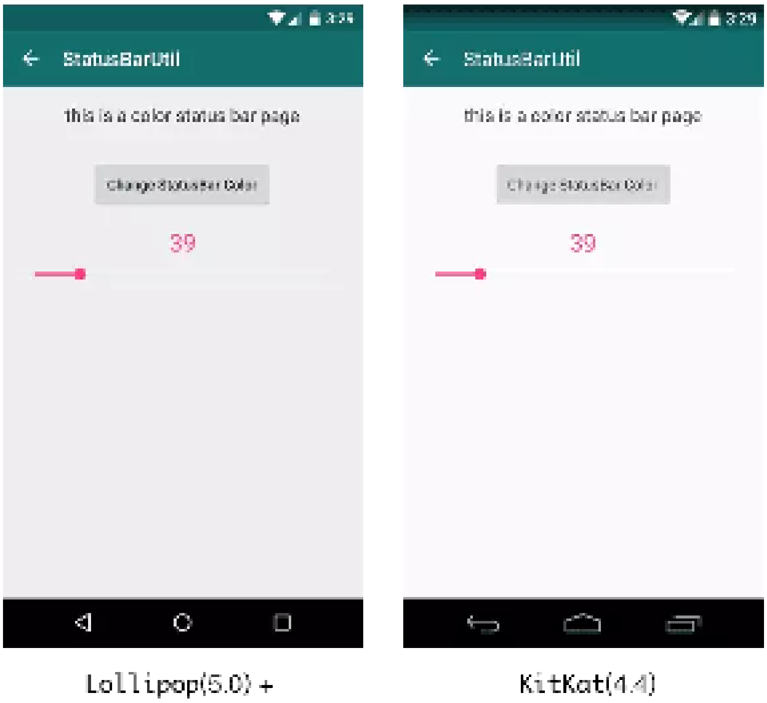
李庆雪 (/u/d21787b52bdc) + 关注
0.9 2017.10.30 18:15* 字数 575 阅读 5214 评论 0 喜欢 4
(/u/d21787b52bdc)

文章参考StatusBarUtil 状态栏工具类（实现沉浸式状态栏/变色状态栏）(https://link.jianshu.com?t=https://jaeger.itscoder.com/android/2016/03/27/statusbar-util.html)

StatusBaStatusBarUtil 状态栏工具类（实现沉浸式状态栏/变色状态栏）
这是一个为Android App 设置状态栏的工具类， 可以在4.4及其以上系统中实现 沉浸式状态栏/状态栏变色，支持设置状态栏透明度
项目 GitHub 地址 (https://link.jianshu.com?t=https://github.com/laobie/StatusBarUtil)
Sample 下载
下载 StatusBarUtil-Demo(APK) (https://link.jianshu.com?t=http://fir.im/5mnp)

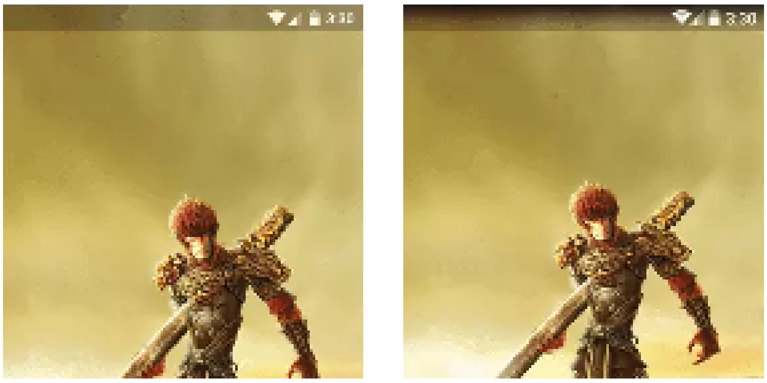
- 设置状态栏颜色

```
StatusBarUtil.setColor(Activity activity, int color)
```



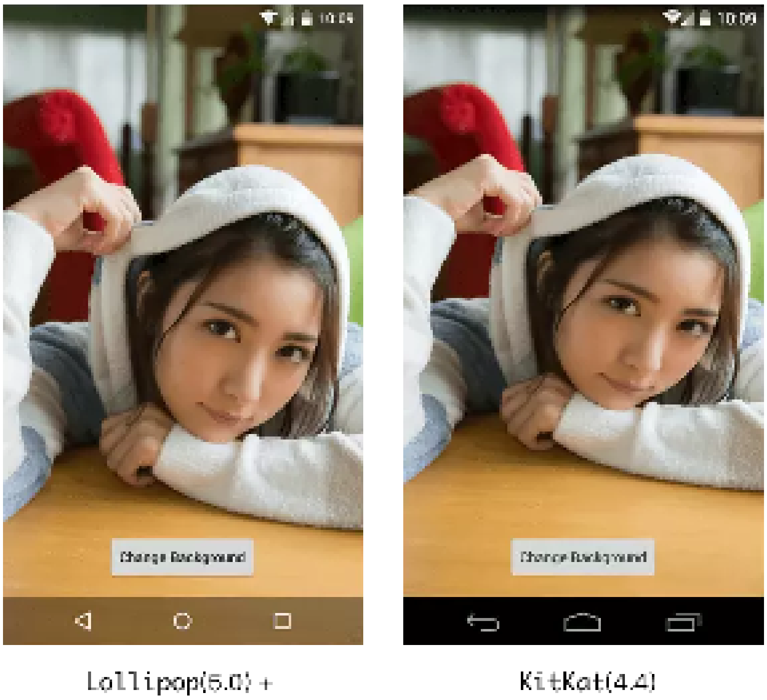
- 设置状态栏半透明

```
StatusBarUtil.setTranslucent(Activity activity, int statusBarAlpha)
```



- 设置状态栏全透明

```
StatusBarUtil.setTransparent(Activity activity)
```

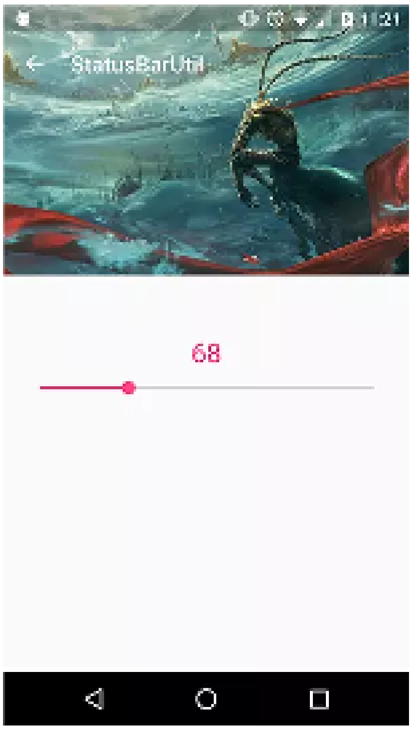


- 包含 DrawerLayout的界面设置状态栏颜色（也可以设置半透明和全透明）

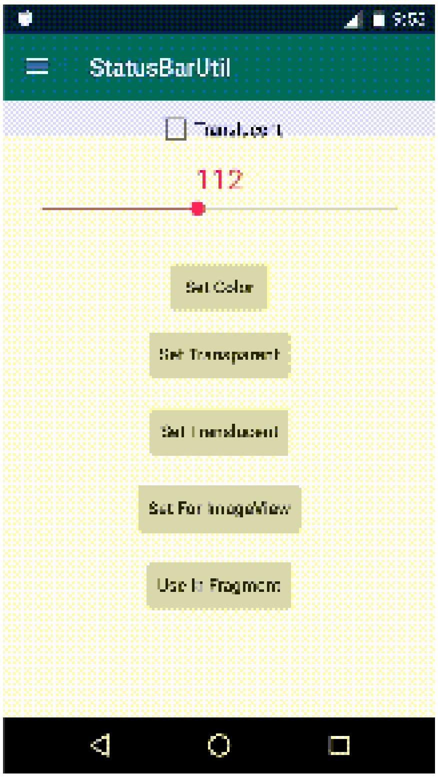
```
StatusBarUtil.setColorForDrawerLayout(Activity activity, DrawerLayout drawerLayout, int color)
```

- 使用 ImageView 作为头部的界面设置状态栏透明

```
StatusBarUtil.setTranslucentForImageView(Activity activity, int statusBarAlpha,
View needOffsetView)
```



- 在 Fragment 中使用 ImageView 作为头部的界面的时候



当一个Activity界面下有4个同级的Fragment的时候,首页的Fragment顶部是图片的时候,设置了透明状态栏之后就会造成其他界面的标题顶到状态栏中,这个时候只需要设置一下标题布局在父控件的LayoutParams,LayoutParams 是 子控件 用来告诉 父控件 自己要如何布局时用的,也就是说设置一下顶部控件的marginTop为状态栏的高度就可以了

```
override fun onCreateView(inflater: LayoutInflater?, container: ViewGroup?, savedInst
        val view = inflater!!.inflate(R.layout.fragment_me, container, false)
        val tvMePress = view.findViewById<TextView>(R.id.tvMePress)
        val layoutParams = tvMePress.layoutParams as LinearLayout.LayoutParams
        layoutParams.topMargin = StatusBarUtil.getStatusBarHeight(context)
        return view
    }
```

参考从LayoutParams说起到代码动态布局 (https://www.jianshu.com/p/0d6f753fdd92)

通过传入 statusBarAlpha 参数，可以改变状态栏的透明度值，默认值是112。

使用在 build.gradle 文件中添加依赖, StatusBarUtil 已经发布在 JCenter:

`compile 'com.jaeger.statusbarutil:library:1.4.0'`

在 setContentView()
之后调用你需要的方法，例如:

```
setContentView(R.layout.main_activity);
...
StatusBarUtil.setColor(MainActivity.this, mColor);
```

- 如果你在一个包含 DrawerLayout
  的界面中使用, 你需要在布局文件中为 DrawerLayout添加
  `android:fitsSystemWindows="true"`
  属性:

```
<android.support.v4.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk
        android:id="@+id/drawer_layout"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:fitsSystemWindows="true">
        ...
    </android.support.v4.widget.DrawerLayout>
```

- 当你设置了 statusBarAlpha值时，该值需要在 0 ~ 255 之间

为了更好的根据自己需求更改所以直接将工具类拷贝到自己项目中就可以

- 在app的build.gradle中dependencies节点下加入
  `compile 'com.android.support:design:26.1.0'`
- 在res中的values目录下创建values.xml,内容如下:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <item name="statusbarutil_fake_status_bar_view" type="id"/>
    <item name="statusbarutil_translucent_view" type="id"/>
</resources>
```

- 在项目下创建util文件夹,并拷贝StatusBarUtil.class

```
package 包名.util;

import android.annotation.TargetApi;
import android.app.Activity;
import android.content.Context;
import android.graphics.Color;
import android.os.Build;
import android.support.annotation.ColorInt;
import android.support.annotation.IntRange;
import android.support.design.widget.CoordinatorLayout;
import android.support.v4.widget.DrawerLayout;
import android.view.View;
import android.view.ViewGroup;
import android.view.WindowManager;
import android.widget.LinearLayout;


/**
 * @author GitHub: https://github.com/laobie
 */
public class StatusBarUtil {
    public static final String TAG = "StatusBarUtil";
    private static final int DEFAULT_STATUS_BAR_ALPHA = 112;
    private static final int FAKE_STATUS_BAR_VIEW_ID = com.lqx.fragmentdemo.R.id.stat
    private static final int FAKE_TRANSLUCENT_VIEW_ID = com.lqx.fragmentdemo.R.id.sta
    private static final int TAG_KEY_HAVE_SET_OFFSET = -123;

    /**
     * 设置状态栏颜色
     *
     * @param activity 需要设置的 activity
     * @param color    状态栏颜色值
     */
    public static void setColor(Activity activity, @ColorInt int color) {
        setColor(activity, color, DEFAULT_STATUS_BAR_ALPHA);
    }

    /**
     * 设置状态栏颜色
     * API等级19:Android 4.4 KitKat   API等级20:Android 4.4W   API等级21:Android 5.0 L
     * API等级22:Android 5.1 Lollipop  API等级23:Android 6.0 Marshmallow   API等级24:A
     *
     * @param activity       需要设置的activity
     * @param color          状态栏颜色值
     * @param statusBarAlpha 状态栏透明度
     */
    public static void setColor(Activity activity, @ColorInt int color, @IntRange(fro
        //21以上
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
            activity.getWindow().addFlags(WindowManager.LayoutParams.FLAG_DRAWS_SYSTE
            activity.getWindow().clearFlags(WindowManager.LayoutParams.FLAG_TRANSLUCE
            activity.getWindow().setStatusBarColor(calculateStatusColor(color, status
            //19到21
        } else if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.KITKAT) {
            activity.getWindow().addFlags(WindowManager.LayoutParams.FLAG_TRANSLUCENT
            //DecorView是顶级View，内部有titlebar和contentParent两个子元素，
            // contentParent的id是content，而我们设置的main.xml布局则是contentParent里面
            ViewGroup decorView = (ViewGroup) activity.getWindow().getDecorView();
            View fakeStatusBarView = decorView.findViewById(FAKE_STATUS_BAR_VIEW_ID);
            if (fakeStatusBarView != null) {
                if (fakeStatusBarView.getVisibility() == View.GONE) {
                    fakeStatusBarView.setVisibility(View.VISIBLE);
                }
                fakeStatusBarView.setBackgroundColor(calculateStatusColor(color, stat
            } else {
                //第一次加载会先创建并添加
                decorView.addView(createStatusBarView(activity, color, statusBarAlpha
            }
            setRootView(activity);
        }
    }

    /**
     * 为滑动返回界面设置状态栏颜色
     *
     * @param activity 需要设置的activity
     * @param color    状态栏颜色值
     */
    public static void setColorForSwipeBack(Activity activity, int color) {
        setColorForSwipeBack(activity, color, DEFAULT_STATUS_BAR_ALPHA);
    }
```

```
/**
 * 为滑动返回界面设置状态栏颜色
 *
 * @param activity       需要设置的activity
 * @param color          状态栏颜色值
 * @param statusBarAlpha 状态栏透明度
 */
public static void setColorForSwipeBack(Activity activity, @ColorInt int color,
                                        @IntRange(from = 0, to = 255) int statusB
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.KITKAT) {
        ViewGroup contentView = ((ViewGroup) activity.findViewById(android.R.id.c
        View rootView = contentView.getChildAt(0);
        int statusBarHeight = getStatusBarHeight(activity);
        if (rootView != null && rootView instanceof CoordinatorLayout) {
            final CoordinatorLayout coordinatorLayout = (CoordinatorLayout) rootV
            if (Build.VERSION.SDK_INT < Build.VERSION_CODES.LOLLIPOP) {
                coordinatorLayout.setFitsSystemWindows(false);
                contentView.setBackgroundColor(calculateStatusColor(color, status
                boolean isNeedRequestLayout = contentView.getPaddingTop() < statu
                if (isNeedRequestLayout) {
                    contentView.setPadding(0, statusBarHeight, 0, 0);
                    coordinatorLayout.post(new Runnable() {
                        @Override
                        public void run() {
                            coordinatorLayout.requestLayout();
                        }
                    });
                }
            } else {
                coordinatorLayout.setStatusBarBackgroundColor(calculateStatusColo
            }
        } else {
            contentView.setPadding(0, statusBarHeight, 0, 0);
            contentView.setBackgroundColor(calculateStatusColor(color, statusBarA
        }
        setTransparentForWindow(activity);
    }
}

/**
 * 设置状态栏纯色 不加半透明效果
 *
 * @param activity 需要设置的 activity
 * @param color    状态栏颜色值
 */
public static void setColorNoTranslucent(Activity activity, @ColorInt int color)
    setColor(activity, color, 0);
}

/**
 * 设置状态栏颜色(5.0以下无半透明效果,不建议使用)
 *
 * @param activity 需要设置的 activity
 * @param color    状态栏颜色值
 */
@Deprecated
public static void setColorDiff(Activity activity, @ColorInt int color) {
    if (Build.VERSION.SDK_INT < Build.VERSION_CODES.KITKAT) {
        return;
    }
    transparentStatusBar(activity);
    ViewGroup contentView = (ViewGroup) activity.findViewById(android.R.id.conten
    // 移除半透明矩形,以免叠加
    View fakeStatusBarView = contentView.findViewById(FAKE_STATUS_BAR_VIEW_ID);
    if (fakeStatusBarView != null) {
        if (fakeStatusBarView.getVisibility() == View.GONE) {
            fakeStatusBarView.setVisibility(View.VISIBLE);
        }
        fakeStatusBarView.setBackgroundColor(color);
    } else {
        contentView.addView(createStatusBarView(activity, color));
    }
    setRootView(activity);
}

/**
 * 使状态栏半透明
 * <p>
 * 适用于图片作为背景的界面,此时需要图片填充到状态栏
 *
 * @param activity 需要设置的activity
 */
public static void setTranslucent(Activity activity) {
    setTranslucent(activity, DEFAULT_STATUS_BAR_ALPHA);
}

/**
 * 使状态栏半透明
 * <p>
 * 适用于图片作为背景的界面,此时需要图片填充到状态栏
 *
 * @param activity       需要设置的activity
 * @param statusBarAlpha 状态栏透明度
 */
public static void setTranslucent(Activity activity, @IntRange(from = 0, to = 255
    if (Build.VERSION.SDK_INT < Build.VERSION_CODES.KITKAT) {
        return;
    }
    setTransparent(activity);
    addTranslucentView(activity, statusBarAlpha);
}

/**
 * 针对根布局是 CoordinatorLayout, 使状态栏半透明
 * <p>
 * 适用于图片作为背景的界面,此时需要图片填充到状态栏
 *
 * @param activity       需要设置的activity
 * @param statusBarAlpha 状态栏透明度
 */
public static void setTranslucentForCoordinatorLayout(Activity activity, @IntRang
    if (Build.VERSION.SDK_INT < Build.VERSION_CODES.KITKAT) {
        return;
    }
    transparentStatusBar(activity);
    addTranslucentView(activity, statusBarAlpha);
}

/**
 * 设置状态栏全透明
 *
 * @param activity 需要设置的activity
 */
public static void setTransparent(Activity activity) {
    if (Build.VERSION.SDK_INT < Build.VERSION_CODES.KITKAT) {
        return;
    }
    transparentStatusBar(activity);
    setRootView(activity);
}

/**
 * 使状态栏透明(5.0以上半透明效果,不建议使用)
 * <p>
 * 适用于图片作为背景的界面,此时需要图片填充到状态栏
 *
 * @param activity 需要设置的activity
 */
@Deprecated
public static void setTranslucentDiff(Activity activity) {
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.KITKAT) {
        // 设置状态栏透明
        activity.getWindow().addFlags(WindowManager.LayoutParams.FLAG_TRANSLUCENT
        setRootView(activity);
    }
}

/**
 * 为DrawerLayout 布局设置状态栏变色
 *
 * @param activity     需要设置的activity
 * @param drawerLayout DrawerLayout
 * @param color        状态栏颜色值
 */
public static void setColorForDrawerLayout(Activity activity, DrawerLayout drawer
    setColorForDrawerLayout(activity, drawerLayout, color, DEFAULT_STATUS_BAR_ALP
}

/**
 * 为DrawerLayout 布局设置状态栏颜色,纯色
 *
 * @param activity     需要设置的activity
 * @param drawerLayout DrawerLayout
 * @param color        状态栏颜色值
 */
public static void setColorNoTranslucentForDrawerLayout(Activity activity, Drawer
    setColorForDrawerLayout(activity, drawerLayout, color, 0);
}

/**
 * 为DrawerLayout 布局设置状态栏变色
 *
 * @param activity       需要设置的activity
 * @param drawerLayout   DrawerLayout
 * @param color          状态栏颜色值
 * @param statusBarAlpha 状态栏透明度
 */
public static void setColorForDrawerLayout(Activity activity, DrawerLayout drawer
                                           @IntRange(from = 0, to = 255) int stat
    if (Build.VERSION.SDK_INT < Build.VERSION_CODES.KITKAT) {
        return;
    }
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
        activity.getWindow().addFlags(WindowManager.LayoutParams.FLAG_DRAWS_SYSTE
        activity.getWindow().clearFlags(WindowManager.LayoutParams.FLAG_TRANSLUCE
        activity.getWindow().setStatusBarColor(Color.TRANSPARENT);
    } else {
        activity.getWindow().addFlags(WindowManager.LayoutParams.FLAG_TRANSLUCENT
    }
    // 生成一个状态栏大小的矩形
    // 添加 statusBarView 到布局中
    ViewGroup contentLayout = (ViewGroup) drawerLayout.getChildAt(0);
    View fakeStatusBarView = contentLayout.findViewById(FAKE_STATUS_BAR_VIEW_ID);
    if (fakeStatusBarView != null) {
        if (fakeStatusBarView.getVisibility() == View.GONE) {
            fakeStatusBarView.setVisibility(View.VISIBLE);
        }
        fakeStatusBarView.setBackgroundColor(color);
    } else {
        contentLayout.addView(createStatusBarView(activity, color), 0);
    }
    // 内容布局不是 LinearLayout 时,设置padding top
    if (!(contentLayout instanceof LinearLayout) && contentLayout.getChildAt(1) !
        contentLayout.getChildAt(1)
                .setPadding(contentLayout.getPaddingLeft(), getStatusBarHeight(ac
                        contentLayout.getPaddingRight(), contentLayout.getPadding
    }
    // 设置属性
    setDrawerLayoutProperty(drawerLayout, contentLayout);
    addTranslucentView(activity, statusBarAlpha);
}

/**
 * 设置 DrawerLayout 属性
 *
 * @param drawerLayout              DrawerLayout
 * @param drawerLayoutContentLayout DrawerLayout 的内容布局
 */
private static void setDrawerLayoutProperty(DrawerLayout drawerLayout, ViewGroup
    ViewGroup drawer = (ViewGroup) drawerLayout.getChildAt(1);
    drawerLayout.setFitsSystemWindows(false);
    drawerLayoutContentLayout.setFitsSystemWindows(false);
    drawerLayoutContentLayout.setClipToPadding(true);
    drawer.setFitsSystemWindows(false);
}

/**
 * 为DrawerLayout 布局设置状态栏变色(5.0以下无半透明效果,不建议使用)
 *
 * @param activity     需要设置的activity
 * @param drawerLayout DrawerLayout
 * @param color        状态栏颜色值
 */
@Deprecated
public static void setColorForDrawerLayoutDiff(Activity activity, DrawerLayout dr
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.KITKAT) {
        activity.getWindow().addFlags(WindowManager.LayoutParams.FLAG_TRANSLUCENT
        // 生成一个状态栏大小的矩形
        ViewGroup contentLayout = (ViewGroup) drawerLayout.getChildAt(0);
        View fakeStatusBarView = contentLayout.findViewById(FAKE_STATUS_BAR_VIEW_
        if (fakeStatusBarView != null) {
            if (fakeStatusBarView.getVisibility() == View.GONE) {
                fakeStatusBarView.setVisibility(View.VISIBLE);
            }
            fakeStatusBarView.setBackgroundColor(calculateStatusColor(color, DEFA
        } else {
            // 添加 statusBarView 到布局中
            contentLayout.addView(createStatusBarView(activity, color), 0);
        }
        // 内容布局不是 LinearLayout 时,设置padding top
        if (!(contentLayout instanceof LinearLayout) && contentLayout.getChildAt(
            contentLayout.getChildAt(1).setPadding(0, getStatusBarHeight(activity
        }
        // 设置属性
        setDrawerLayoutProperty(drawerLayout, contentLayout);
    }
}

/**
 * 为 DrawerLayout 布局设置状态栏透明
 *
 * @param activity     需要设置的activity
 * @param drawerLayout DrawerLayout
 */
public static void setTranslucentForDrawerLayout(Activity activity, DrawerLayout 
    setTranslucentForDrawerLayout(activity, drawerLayout, DEFAULT_STATUS_BAR_ALPH
}

/**
 * 为 DrawerLayout 布局设置状态栏透明
 *
 * @param activity     需要设置的activity
 * @param drawerLayout DrawerLayout
 */
public static void setTranslucentForDrawerLayout(Activity activity, DrawerLayout 
                                                 @IntRange(from = 0, to = 255) in
    if (Build.VERSION.SDK_INT < Build.VERSION_CODES.KITKAT) {
        return;
    }
    setTransparentForDrawerLayout(activity, drawerLayout);
    addTranslucentView(activity, statusBarAlpha);
}

/**
 * 为 DrawerLayout 布局设置状态栏透明
 *
 * @param activity     需要设置的activity
 * @param drawerLayout DrawerLayout
 */
public static void setTransparentForDrawerLayout(Activity activity, DrawerLayout 
    if (Build.VERSION.SDK_INT < Build.VERSION_CODES.KITKAT) {
        return;
    }
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
        activity.getWindow().addFlags(WindowManager.LayoutParams.FLAG_DRAWS_SYSTE
        activity.getWindow().clearFlags(WindowManager.LayoutParams.FLAG_TRANSLUCE
        activity.getWindow().setStatusBarColor(Color.TRANSPARENT);
    } else {
        activity.getWindow().addFlags(WindowManager.LayoutParams.FLAG_TRANSLUCENT
    }

    ViewGroup contentLayout = (ViewGroup) drawerLayout.getChildAt(0);
    // 内容布局不是 LinearLayout 时,设置padding top
    if (!(contentLayout instanceof LinearLayout) && contentLayout.getChildAt(1) !
        contentLayout.getChildAt(1).setPadding(0, getStatusBarHeight(activity), 0
    }

    // 设置属性
    setDrawerLayoutProperty(drawerLayout, contentLayout);
}

/**
 * 为 DrawerLayout 布局设置状态栏透明(5.0以上半透明效果,不建议使用)
 *
 * @param activity     需要设置的activity
 * @param drawerLayout DrawerLayout
 */
@Deprecated
public static void setTranslucentForDrawerLayoutDiff(Activity activity, DrawerLay
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.KITKAT) {
        // 设置状态栏透明
        activity.getWindow().addFlags(WindowManager.LayoutParams.FLAG_TRANSLUCENT
        // 设置内容布局属性
        ViewGroup contentLayout = (ViewGroup) drawerLayout.getChildAt(0);
        contentLayout.setFitsSystemWindows(true);
        contentLayout.setClipToPadding(true);
        // 设置抽屉布局属性
```

```
            ViewGroup vg = (ViewGroup) drawerLayout.getChildAt(1);
            vg.setFitsSystemWindows(false);
            // 设置 DrawerLayout 属性
            drawerLayout.setFitsSystemWindows(false);
        }
    }

    /**
     * 为头部是 ImageView 的界面设置状态栏全透明
     *
     * @param activity       需要设置的activity
     * @param needOffsetView 需要向下偏移的 View
     */
    public static void setTransparentForImageView(Activity activity, View needOffsetV
        setTranslucentForImageView(activity, 0, needOffsetView);
    }

    /**
     * 为头部是 ImageView 的界面设置状态栏透明(使用默认透明度)
     *
     * @param activity       需要设置的activity
     * @param needOffsetView 需要向下偏移的 View
     */
    public static void setTranslucentForImageView(Activity activity, View needOffsetV
        setTranslucentForImageView(activity, DEFAULT_STATUS_BAR_ALPHA, needOffsetVie
    }

    /**
     * 为头部是 ImageView 的界面设置状态栏透明
     *
     * @param activity       需要设置的activity
     * @param statusBarAlpha 状态栏透明度
     * @param needOffsetView 需要向下偏移的 View
     */
    public static void setTranslucentForImageView(Activity activity, @IntRange(from =
                                                  View needOffsetView) {
        if (Build.VERSION.SDK_INT < Build.VERSION_CODES.KITKAT) {
            return;
        }
        setTransparentForWindow(activity);
        addTranslucentView(activity, statusBarAlpha);
        if (needOffsetView != null) {
            Object haveSetOffset = needOffsetView.getTag(TAG_KEY_HAVE_SET_OFFSET);
            if (haveSetOffset != null && (Boolean) haveSetOffset) {
                return;
            }
            ViewGroup.MarginLayoutParams layoutParams = (ViewGroup.MarginLayoutParams
            layoutParams.setMargins(layoutParams.leftMargin, layoutParams.topMargin +
                    layoutParams.rightMargin, layoutParams.bottomMargin);
            needOffsetView.setTag(TAG_KEY_HAVE_SET_OFFSET, true);
        }
    }

    /**
     * 为 fragment 头部是 ImageView 的设置状态栏透明
     *
     * @param activity       fragment 对应的 activity
     * @param needOffsetView 需要向下偏移的 View
     */
    public static void setTranslucentForImageViewInFragment(Activity activity, View n
        setTranslucentForImageViewInFragment(activity, DEFAULT_STATUS_BAR_ALPHA, need
    }

    /**
     * 为 fragment 头部是 ImageView 的设置状态栏透明
     *
     * @param activity       fragment 对应的 activity
     * @param needOffsetView 需要向下偏移的 View
     */
    public static void setTransparentForImageViewInFragment(Activity activity, View n
        setTranslucentForImageViewInFragment(activity, 0, needOffsetView);
    }

    /**
     * 为 fragment 头部是 ImageView 的设置状态栏透明
     *
     * @param activity       fragment 对应的 activity
     * @param statusBarAlpha 状态栏透明度
     * @param needOffsetView 需要向下偏移的 View
     */
    public static void setTranslucentForImageViewInFragment(Activity activity, @IntRa
                                                            View needOffsetView) {
```

```
    setTranslucentForImageView(activity, statusBarAlpha, needOffsetView);
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.KITKAT && Build.VERSION.SDK_
        clearPreviousSetting(activity);
    }
}

/**
 * 隐藏伪状态栏 View
 *
 * @param activity 调用的 Activity
 */
public static void hideFakeStatusBarView(Activity activity) {
    ViewGroup decorView = (ViewGroup) activity.getWindow().getDecorView();
    View fakeStatusBarView = decorView.findViewById(FAKE_STATUS_BAR_VIEW_ID);
    if (fakeStatusBarView != null) {
        fakeStatusBarView.setVisibility(View.GONE);
    }
    View fakeTranslucentView = decorView.findViewById(FAKE_TRANSLUCENT_VIEW_ID);
    if (fakeTranslucentView != null) {
        fakeTranslucentView.setVisibility(View.GONE);
    }
}

/////////////////////////////////////////////////////////////////////////////////

@TargetApi(Build.VERSION_CODES.KITKAT)
private static void clearPreviousSetting(Activity activity) {
    ViewGroup decorView = (ViewGroup) activity.getWindow().getDecorView();
    View fakeStatusBarView = decorView.findViewById(FAKE_STATUS_BAR_VIEW_ID);
    if (fakeStatusBarView != null) {
        decorView.removeView(fakeStatusBarView);
        ViewGroup rootView = (ViewGroup) ((ViewGroup) activity.findViewById(andro
        rootView.setPadding(0, 0, 0, 0);
    }
}

/**
 * 添加半透明矩形条
 *
 * @param activity       需要设置的 activity
 * @param statusBarAlpha 透明值
 */
private static void addTranslucentView(Activity activity, @IntRange(from = 0, to
    ViewGroup contentView = (ViewGroup) activity.findViewById(android.R.id.conten
    View fakeTranslucentView = contentView.findViewById(FAKE_TRANSLUCENT_VIEW_ID)
    if (fakeTranslucentView != null) {
        if (fakeTranslucentView.getVisibility() == View.GONE) {
            fakeTranslucentView.setVisibility(View.VISIBLE);
        }
        fakeTranslucentView.setBackgroundColor(Color.argb(statusBarAlpha, 0, 0, 0
    } else {
        contentView.addView(createTranslucentStatusBarView(activity, statusBarAlp
    }
}

/**
 * 生成一个和状态栏大小相同的彩色矩形条
 *
 * @param activity 需要设置的 activity
 * @param color    状态栏颜色值
 * @return 状态栏矩形条
 */
private static View createStatusBarView(Activity activity, @ColorInt int color) {
    return createStatusBarView(activity, color, 0);
}

/**
 * 生成一个和状态栏大小相同的半透明矩形条
 *
 * @param activity 需要设置的activity
 * @param color    状态栏颜色值
 * @param alpha    透明值
 * @return 状态栏矩形条
 */
private static View createStatusBarView(Activity activity, @ColorInt int color, i
    // 绘制一个和状态栏一样高的矩形
    View statusBarView = new View(activity);
    LinearLayout.LayoutParams params =
            new LinearLayout.LayoutParams(ViewGroup.LayoutParams.MATCH_PARENT, ge
    statusBarView.setLayoutParams(params);
    statusBarView.setBackgroundColor(calculateStatusColor(color, alpha));
    statusBarView.setId(FAKE_STATUS_BAR_VIEW_ID);
    return statusBarView;
}

/**
 * 设置根布局参数
 */
private static void setRootView(Activity activity) {
    ViewGroup parent = (ViewGroup) activity.findViewById(android.R.id.content);
    for (int i = 0, count = parent.getChildCount(); i < count; i++) {
        View childView = parent.getChildAt(i);
        if (childView instanceof ViewGroup) {
            childView.setFitsSystemWindows(true);
            ((ViewGroup) childView).setClipToPadding(true);
        }
    }
}

/**
 * 设置透明
 */
private static void setTransparentForWindow(Activity activity) {
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
        activity.getWindow().setStatusBarColor(Color.TRANSPARENT);
        activity.getWindow()
                .getDecorView()
                .setSystemUiVisibility(View.SYSTEM_UI_FLAG_LAYOUT_STABLE | View.S
    } else if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.KITKAT) {
        activity.getWindow()
                .setFlags(WindowManager.LayoutParams.FLAG_TRANSLUCENT_STATUS, Wir
    }
}

/**
 * 使状态栏透明
 */
@TargetApi(Build.VERSION_CODES.KITKAT)
private static void transparentStatusBar(Activity activity) {
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
        activity.getWindow().addFlags(WindowManager.LayoutParams.FLAG_DRAWS_SYSTE
        activity.getWindow().clearFlags(WindowManager.LayoutParams.FLAG_TRANSLUCE
        activity.getWindow().addFlags(WindowManager.LayoutParams.FLAG_TRANSLUCENT
        activity.getWindow().setStatusBarColor(Color.TRANSPARENT);
    } else {
        activity.getWindow().addFlags(WindowManager.LayoutParams.FLAG_TRANSLUCENT
    }
}

/**
 * 创建半透明矩形 View
 *
 * @param alpha 透明值
 * @return 半透明 View
 */
private static View createTranslucentStatusBarView(Activity activity, int alpha)
    // 绘制一个和状态栏一样高的矩形
    View statusBarView = new View(activity);
    LinearLayout.LayoutParams params =
            new LinearLayout.LayoutParams(ViewGroup.LayoutParams.MATCH_PARENT, ge
    statusBarView.setLayoutParams(params);
    statusBarView.setBackgroundColor(Color.argb(alpha, 0, 0, 0));
    statusBarView.setId(FAKE_TRANSLUCENT_VIEW_ID);
    return statusBarView;
}

/**
 * 获取状态栏高度
 *
 * @param context context
 * @return 状态栏高度
 */
public static int getStatusBarHeight(Context context) {
    // 获得状态栏高度
    int resourceId = context.getResources().getIdentifier("status_bar_height", "c
    return context.getResources().getDimensionPixelSize(resourceId);
}

/**
 * 计算状态栏颜色
 *
 * @param color color值
 * @param alpha alpha值
 * @return 最终的状态栏颜色
```

```
     */
    private static int calculateStatusColor(@ColorInt int color, int alpha) {
        if (alpha == 0) {
            return color;
        }
        float a = 1 - alpha / 255f;
        int red = color >> 16 & 0xff;
        int green = color >> 8 & 0xff;
        int blue = color & 0xff;
        red = (int) (red * a + 0.5);
        green = (int) (green * a + 0.5);
        blue = (int) (blue * a + 0.5);
        return 0xff << 24 | red << 16 | green << 8 | blue;
    }
}
```

(/apps/redirect?utm_source=side-banner-click)

小礼物走一走，来简书关注我

赞赏支持

Android (/nb/11523553)

举报文章 © 著作权归作者所有

李庆雪 (/u/d21787b52bdc) ♂

写了 50169 字，被 64 人关注，获得了 149 个喜欢

(/u/d21787b52bdc)

＋关注

喜欢 | 4

更多分享

(/apps/redirect?utm_source=note-bottom-click)

登录 (/sign_in?utm_source=desktop&utm_medium=not-signed-in-comment-form)后发表评论

评论

智慧如你，不想发表一点想法 (/sign_in?utm_source=desktop&utm_medium=not-signed-in-nocomments-text)咩~

被以下专题收入，发现更多相似内容

第三方扩展 (/c/d90432b7a9c3?utm_source=desktop&utm_medium=notes-

included-collection)

Android黑科技 (/c/88a37138a7c0?
utm_source=desktop&utm_medium=notes-included-collection)

(/apps/redirect?
utm_source=side-
banner-click)

推荐阅读 更多精彩内容 > (/)

Android - 收藏集 (/p/dad51f6c9c4d?utm_campaign=maleskine&utm_co...

Android 自定义View的各种姿势1 Activity的显示之ViewRootImpl详解 Activity的显示之ViewRootImpl初探 Activity的显示之Window和View Android系统的创世之初以及Activity的生命周期 图解Andro...

passiontim (/u/e946d18f163c?
utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

源码众享最新教程 - 草稿 (/p/df8667ac1ec6?utm_campaign=maleskine&u...

¥开启¥ 【iAPP实现进入界面执行逐一显】 〖2017-08-25 15:22:14〗 《//首先开一个线程，因为你有这么多的stop要执行，其次必须使用ufusui（ ）,否则可能不会显示你的字体。切记:文字必须设置“text=”,否则很尴...

小菜c (/u/c5f2afa6ee0c?
utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

GitHub上受欢迎的Android UI Library (/p/da1ca645b95c?utm_campaign...

内容抽屉菜单ListViewWebViewSwitchButton按钮点赞按钮进度条TabLayout图标下拉刷新ViewPager图表(Chart)菜单(Menu)浮动菜单对话框空白页滑动删除手势操作RecyclerViewCardColorDrawableSpinner...

皇小弟 (/u/04dca94c9b8b?
utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

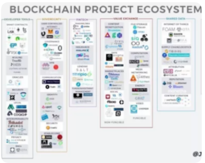
(/p/70ad4052f0bd?

utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation)
Android面试笔记（有福利） (/p/70ad4052f0bd?utm_campaign=maleski...

第一次修改和更新 内容:1.修正文章的排版问题2.更新一下在电子市场中这个项目经验 第二次更新说明 内容1.最近手上有项目需要开发，本人也有前端的内容需要学习和专研，可能就忽略更新，此处给Android开发...

LogosTravel (/u/49606254ed76?
utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

(/p/7ecb843485e7?



utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation)
（coindesk）“智能合约”的定义之争 (/p/7ecb843485e7?utm_campaign=...

2017-11-10 平行区块链 “智能合约”的定义之争原文：“Making Sense of Blockchain Smart Contracts” | 稿源：coindesk作者：Josh Stark ，区块链咨询公司和开发集团 Ledger Labs，律师兼运营和法律...

大圣2017 (/u/a78a1a2a847e?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

(/p/958ac6393a3d?

(/apps/redirect?utm_source=side-banner-click)

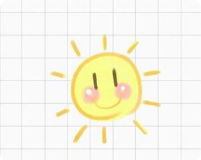
utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation)

时间管理100讲-16 时间管理的基础是精力管理 (/p/958ac6393a3d?utm_ca...

时间管理100讲-16 时间管理的基础是精力管理 习惯就是由我们潜意识在控制的各种行为，有好的，也有不好的。 好习惯和坏习惯一旦养成就很难改变。 那从这一讲开始，我们就将进入---习惯的部分。 今天就首先来...

张强用生命影响生命 (/u/f06d70cd1ac8?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

(/p/15f1dba4c4c7?



utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation)

爱一个人 (/p/15f1dba4c4c7?utm_campaign=maleskine&utm_content=n...

当你真的去爱一个人的时候，你会发现你在乎的是她的一切，尽管在别人眼里看来是幼稚的事情。

软软大人 (/u/2acec2a6e9a1?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

(/p/c203d52904fc?

utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation)

偶遇…… (/p/c203d52904fc?utm_campaign=maleskine&utm_content=n...

那天，你开着车，我坐在后排，打开车窗，秋风迎面吹来，有一丝丝凉意，但在日落的呵护下，却感觉很温暖。好难得，在雾霾笼罩的天空，在我们相识、相知、相爱的城市，偶遇了这美丽的日落。思绪开始渐渐拉..

歪歪不倒 (/u/15f825aeb924?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)