

# 换肤多主题方案

## 代码路径

<https://github.com/SanYuanAndy/OpenApi.git>

## 代码结构

- 1、多主题实现代码在multitheme模块；
- 2、多主题测试代码在app模块；

## 集成方法:

- 1、在Application的onCreate中调用如下代码，初始化多主题sdk

`MultiThemeSDK.getInstance().initial(app, mCurrThemeIndex);`// mCurrThemeIndex表示当前主题index

- 2、切换主题的时候调用如下代码

`MultiThemeSDK.getInstance().refreshTheme(mCurrThemeIndex);`// mCurrThemeIndex表示当前主题index

- 3、需要支持多主题的资源，名称命名前缀固定为theme\_，  
每个主题名(如果是文件则不包含文件类型后缀)后缀加上主题index，  
比如文本字体颜色，

主题0, theme\_main\_text\_color.xml;

主题1, theme\_main\_text\_color\_1.xml;

主题2, theme\_main\_text\_color\_2.xml;

比如View的背景颜色，

#ffdddddd

#ffffff

#ff777777

比如ImageView的图片资源，

主题0, theme\_image.png;

主题1, theme\_image\_1.png;

主题2, theme\_image\_2.png;

# 方案关键点

## 设置LayoutInflater的自定义Factory

```
void setFactory2(Factory2 factory)
```

目的: 自定义onCreateView方法

## Window的dispatchWindowFocusChanged

- 1、避免重新保存所有需要支持换肤的view
- 2、利用安卓原生的层级调用

## MultiThemeDrawable

使用自定义的Drawable,实现类似多状态Drawable的功能, 主题发生变化的时候触发自定义的Drawable重绘。

## 目前的问题

- 1、未适配Android6.0
- 2、暂时仅支持文字颜色、背景颜色、背景图片、ImageView内容图片换肤, 其他的需要继续开发
- 3、替换资源的时候, 由于使用了一次循环遍历, 有一定的性能消耗。