

直接讲解Palette的使用步骤和实例，补充说明Palette的几个注意点。

转载请注明链接：[https://blog.csdn.net/feather\\_wch/article/details/80066384](https://blog.csdn.net/feather_wch/article/details/80066384)

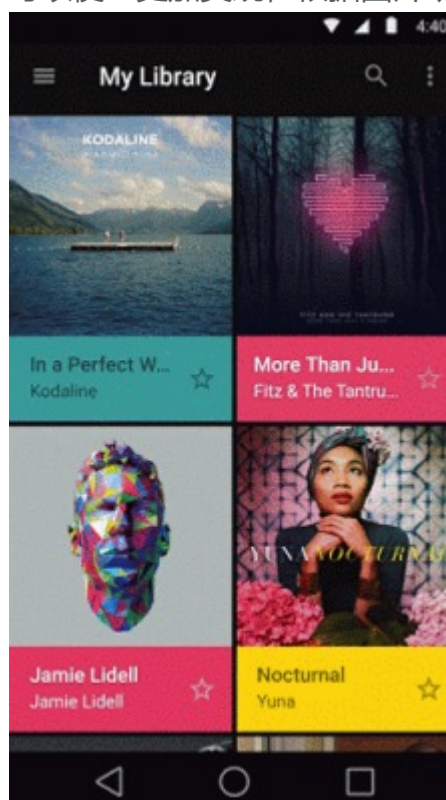
# Palette详解

版本：2018/4/24-1

- [Palette详解](#)
  - [注意点](#)
  - [参考资料](#)

## 1、Palette是什么？

1. Palette是一个可以从图片(Bitmap)中提取颜色的帮助类。
2. 可以使UI更加美观，根据图片动态的显示相应的颜色。





## 2、Palette的使用步骤

1、引入依赖包 `implementation 'com.android.support:palette-v7:26.1.0'`

2、获取调色板对象

// 同步

```
Palette p = Palette.from(bitmap).generate();
```

// 异步

```
Palette.from(bitmap).generate(new PaletteAsyncListener() {  
    public void onGenerated(Palette p) {  
        // Use generated instance  
    }  
});
```

3、根据调色板获取色样(Palette.Swatch可能为null)

//有活力的

```
Palette.Swatch vibrant = palette.getVibrantSwatch();
```

//有活力的，暗色

```
Palette.Swatch vibrantDark = palette.getDarkVibrantSwatch();
```

//有活力的，亮色

```
Palette.Swatch vibrantLight = palette.getLightVibrantSwatch();
```

//柔和的

```
Palette.Swatch muted = palette.getMutedSwatch();
```

//柔和的，暗色

```
Palette.Swatch mutedDark = palette.getDarkMutedSwatch();
```

//柔和的，亮色

```
Palette.Swatch mutedLight = palette.getLightMutedSwatch();
```

## 4、色样中获取颜色

```
//1、样本中的像素数量
int population = vibrant.getPopulation();
//2、颜色的RGB值
int rgb = vibrant.getRgb();
//3、颜色的HSL值
float[] hsl = vibrant.getHsl();
//4、主体文字的颜色值
int bodyTextColor = vibrant.getBodyTextColor();
//5、标题文字的颜色值
int titleTextColor = vibrant.getTitleTextColor();
```

## 3、Palette的简单实例(利用Glide获取Bitmap并且获取到Palette)

```
//获取图片的Bitmap
RequestBuilder<Bitmap> requestBuilder = Glide.with(getApplicationContext()).asBitmap();

requestBuilder.load(datas.get(mPosition).getHeadImg()).into(new SimpleTarget<Bitmap>() {
    @Override
    public void onResourceReady(@NonNull Bitmap bitmap, @Nullable Transition<? super Bitmap> tr

        //创建Palette(异步)
        Palette.from(bitmap).generate(new Palette.PaletteAsyncListener() {
            @Override
            public void onGenerated(Palette palette) {
                //1-获取 靓丽的活力色
                Palette.Swatch vibrant = palette.getVibrantSwatch();
                //2-不存在活力色，获取柔和色
                if(vibrant == null){
                    vibrant = palette.getDarkMutedSwatch();
                }
                //3-设置颜色
                if(vibrant != null){
                    int titleColor = vibrant.getRgb();
                    mCollapsingToolbarLayout.setContentScrimColor(titleColor);

                    mCollapsingToolbarLayout.setCollapsedTitleTextColor(vibrant.getBodyTextColo
                }
            }
        });
    });
});
```

## 注意点

### 1、Palette.Swatch获取为Null的解决办法

1. 柔和色`getLightMutedSwatch()` 基本上不会为 `NULL` 。
2. 活力色`getLightVibrantSwatch()` 相关API对于部分图片可能为 `NULL` 。
3. 可以参考上面实例，先获取 活力色 如果没有，则去获取 柔和色。确保最大概率获取到 颜色

## 2、颜色的选取

1. 建议 活力色使用一般色：`getVibrantSwatch` (不是亮色也不是深色)，并且 标题颜色 采用 `getBodyTextColor` ,一般为白色比较美观。
2. 建议 柔和色采用深色：`getDarkMutedSwatch` , 并且 标题颜色 使用 `getBodyTextColor` ,一般也是白色。

## 参考资料

1. [android Palette使用详解](#)