

暗黑模式(iOS多主题)完美设计实现



Tkoul

[关注](#)

💎 2 2020.06.09 16:08:47 字数 816 阅读 4,152

Introduce:

ios暗黑模式,多主题多皮肤设计,用于解决在ios上实现可跟随系统主题变更,也可不跟随实现自定义主题模式设计。本sdk可以支持任意多个主题。开发习惯上极度契合ios开发习惯,对于复杂的主题设置类,均可使用对象的tkThemeChangeBlock进行回调设置变更。优势:

- 1.代码无侵入,轻量级SDK。上手简单。完美解决多任务后台主题即时变更,屏幕快照变更主体。
- 2.全局控制,效率极高。用hashmap的形式管理block指针对象,轻量。
- 3.一切OC对象皆tkThemeChangeBlock主题回调,根据主题的目前的索引变更主题。
- 4.常用的UIKit的组件,赋予主题属性,主题方法等。直接设置主题颜色、图片数组即可。完全契合系统设置属性,方法习惯。
- 5.设计原理完美,弱引用的形式回调,不会对项目产生内存泄漏,内存占用不释放问题。UIKit-View层即用即注册,跟随对象释放即销毁。主题回调block即用即注册,跟随对象释放即销毁。

movie show

效果视频如下: (ps: 没办法,简书只支持优酷/腾讯视频 没法去广告 想看大视频,点链接:https://v.youku.com/v_show/id_XNDcwNTcxODMwNA==.html)

iOSDarkDemo.mp4

How To Get Started

1.导入 pod 'TKThemeConfig' // (ps: 本人会对sdk一直维护,放心使用)

2.在使用到的地方 #import <TKThemeConfig/TKThemeConfig.h>

程序加载完毕初始化



写字楼租赁



深圳共享办公室



如何出租写字楼



logo设计



Tkoul

[关注](#)

总资产3 (约0.28元)

万能API路由 (api任意调用, 组件完全解耦) - iOS

阅读 691

指针参数注意: 左值,右值

阅读 301

```

#import "TKAppDelegate.h"
#import "TKDemoViewCtrl.h"
#import <TKThemeConfig/TKThemeConfig.h>
@implementation TKAppDelegate
- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    //程序加载完毕 在其它组件含有UI元素前, 优先设置主题, 因为主题作用全局的UI
    //可以设置YES 也可设置NO 根据自己的项目自行选择
    [self changeThemeCofingFlowSystem:YES];
    return YES;
}

#pragma mark -- 程序加载完毕 在其它组件含有UI元素前, 优先设置主题, 因为主题作用全局的UI
- (void)changeThemeCofingFlowSystem:(BOOL)flowSystemTheme{
    if (flowSystemTheme) {

```

How To Usage

1.便捷用法

sdk封装可常见的view控件。对于这些常见的控件，比如CALayer, UIView, UIImageView, UIButton, UILabel等15个组件进行了贴合开发者的属性定制，在原属性上加前缀TKTheme。能满足80%以上的开发需求。剩下的用万能方法即可。有时间功夫的小伙伴可以加入一起完善便捷用法哦。示例如下： // UIButton like // UIImageView like // CALayer like。其它的可自行看头文件

```
layer.tkThemebackgroundColors = @[UIColor.brownColor,UIColor.darkGrayColor];
```

```
[themeSeting setTkThemebackgroundColors:@[UIColor.redColor,UIColor.brownColor]];
```

```
[imageViewsetTkThemeimages:@[[UIImage imageNamed:@"001.jpg"],[UIImage
imageNamed:@"002.jpg"]]];
```

```

- (void)viewDidLoad {
    [super viewDidLoad];

    //2.便捷用法,
    sdk封装可常见的view控件。对于这些常见的控件，比如CALayer, UIView, UIImageView, UIButton, UILabel等15个组件进行了贴合开发者的属性定制，在原属性上加前缀TKTheme。能满足80%以上的开发需求。剩下的用万能方法即可。有时间功夫的小伙伴可以加入一起完善便捷用法哦。
    //示例如下:
    //CALayer
    CALayer *layer = [[CALayer alloc] init];
    layer.frame = CGRectMake(10, 10, self.view.bounds.size.width-20, self.view.bounds.size.height/4);
    layer.tkThemebackgroundColors = @[UIColor.brownColor,UIColor.darkGrayColor];
    [self.view.layer addSublayer:layer];
    //UIButton
    UIButton *themeSeting = [[UIButton alloc] initWithFrame:CGRectMake(100, 100, 200, 40)];
    [themeSeting addTarget:self action:@selector(showSetingView:) forControlEvents:UIControlEventTouchUpInside];
    [themeSeting setTitle:@"打开设置" forState:UIControlStateNormal];
    [themeSeting setTkThemebackgroundColors:@[UIColor.redColor,UIColor.brownColor]];
    [self.view addSubview:themeSeting];
    //UIImageView
    UIImageView *imageView = [[UIImageView alloc] initWithFrame:CGRectMake(10, self.view.bounds.size.height/4+30, 180, 180)];
    [imageView setTkThemeimages:@[[UIImage imageNamed:@"001.jpg"],[UIImage imageNamed:@"002.jpg"]]];
    [self.view addSubview:imageView];
}

```

2.万能用法

一切皆object对象,一切对象具备tkThemeChangeBlock (NSObject+TKUpdate.h),变更主题会触发 任意对象的tkThemeChangeBlock回调，可以在这个回调做主题设置。本回调为主线程，进行UI主题变更，但是不要做耗时操作，耗时操作放到异步非主线程即可。示例如下： //view.tkThemeChangeBlock //navigationBar.navigationBar

```
self.view.tkThemeChangeBlock = ^(id _Nullable itself, NSInteger themeIndex) { //设置主题 }
```

```
self.navigationController.navigationBar.tkThemeChangeBlock = ^(id _Nullable itself,
NSInteger themeIndex) { //设置主题 }
```

```
-(void)viewDidLoad {
    [super viewDidLoad];
    //1.万能用法，一切皆对象。 一切对象具备tkThemeChangeBlock (NSObject+TKUpdate
    tkThemeChangeBlock。本回调为主线程，进行UI主题变更，但是不要做耗时操作，耗时操作放到异步非主线程即可。
```

简书

首页

下载APP

搜索

Q

Aa

beta

登录

注册

```
        color = UIColor.whiteColor;
    }else{
        color = UIColor.blackColor;
    }
    [(UIView*)itself setBackgroundColor:color];

    dispatch_async(dispatch_get_global_queue(0, 0), ^{
        //如果除了UI操作，还有其它耗时操作，放到其它非主线程队列
        // do something, while the long operation is underway.
    });
};
self.navigationController.navigationBar.tkThemeChangeBlock = ^(id _Nullable itself, NSInteger themeIndex) {
    UIImage *image = nil;
    if (themeIndex==0) {
        image = [UIImage imageNamed:@"005.png"];
    }else{
        image = [UIImage imageNamed:@"007.png"];
    }
    [(UINavigationController*)itself setBackgroundImage:image forBarMetrics:UIBarMetricsDefault];
```

赏

项目地址: <https://github.com/Tkoul/TKThemeConfig>

赞赏

本文为原创，啃了几天泡面搞出来的。转载注明出处。喜欢的给个小星星。对不足的地方，可以一起交流，提升。邮件: Tkoull@163.com

对简书编辑器玩不转，有点丑！设计思想后续补上。



44人点赞 >



iOS技术合集



"小礼物走一走，来简书关注我"

赞赏支持

还没有人赞赏，支持一下



Tkoul

总资产3 (约0.28元) 共写了3793字 获得52个赞 共29个粉丝

关注

2020年 #托福新变化# 托福®当场出分功能

托福®考试听力、阅读当场出分 助当场了解考试表现 更快决策考试及申请计划

ETS TOEFL

写下你的评论...

全部评论 9

只看作者

按时间倒序 按时间正序

推荐阅读

面试了一个 31 岁的 iOS 开发者，让我有所触动，30 岁以上的程序员该读 8,780

波波：硬核分析苹果 iOS 14 带来的 App Store 与 ASO 生态 4 大变化 阅读 459

做 iOS 6 年了 30 岁了工资不理想该怎么办？ 阅读 14,627

iOS 开发——“有点难”的 iOS 面试题 阅读 1,046

面试技巧攻克 (3) - OC 高级特性 阅读 309



写下你的评论...

 评论9 赞44

...

建议参考一下我去年实现的多主题方案，代码更少，接口更简单，内存管理更方便😁
<https://github.com/dennis-li/DLThemeAdapter> demo可以直接跑起来。

 赞  回复

Superman168

5楼 06.30 15:30

这个只能支持两种主题吗？如果要用很多个主题怎么办？我也不知道什么颜色，都是从后台请求动态获取的要怎么办？

 赞  回复Tkoul 作者

07.01 11:02

支持任意多种主题，只需要往数组里面添加，设置themeIndex即可。后台设置的颜色数据解析后add到数组即可。

 回复

Superman168

07.01 14:56

@Tkoul 我是说后台控制随时切换主题，你这个不是每次设置的属性颜色数组，怎么让设置的主题和后台的保持一致？对应起来？如果有十几个主题的话要加十几个颜色吗？

 回复 添加新评论

Superman168

4楼 06.30 14:26

2020.06.09 16:08:47

字数 818

阅读 618

说的感觉很牛逼的样子，忍不住看了看，研究一下

 赞  回复

神策新兵

3楼 06.13 00:58

我想知道整个项目适配暗黑模式，用您的库要多少行码

 赞  回复



灯红酒绿映不出的落寞

2楼 06.11 09:14

是个好方法，和DKNightVersion这个库思路差不多

👍 赞 💬 回复



灯红酒绿映不出的落寞

06.11 09:14

<https://github.com/draveness/DKNightVersion>

💬 回复



Tkoul 作者

06.11 09:51

去瞅了一眼，对外API设计大同小异，但是核心设计不一样。个人觉得通知量级重些，本sdk是通过轻量级block嗅探的方式完成最核心的主题变更，高效，轻量级。

💬 回复

✍️ 添加新评论

被以下专题收入，发现更多相似内容



iOS进阶专栏



iOS Dev...



iOS开发



iOS在路上



iOS开发



iOS学习专题



iOS开发者

展开更多 ▾