

ios下JS与原生OC互相调用(总结)



Haley_Wong (/u/a8cf6d63e889) [+ 关注](#)

2015.11.16 22:31* 字数 916 阅读 79676 评论 87 喜欢 252 阅读 79676 评论 87 喜欢 252

(/u/a8cf6d63e889)

ios开发免不了要与UIWebView打交道，然后就要涉及到JS与原生OC交互，今天总结一下JS与原生OC交互的两种方式。

JS调用原生OC篇

方式一

第一种方式是用JS发起一个假的URL请求，然后利用UIWebView的代理方法拦截这次请求，然后再做相应的处理。

我写了一个简单的HTML网页和一个btn点击事件用来与原生OC交互，HTML代码如下：

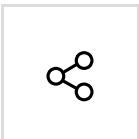
```
<html>
  <header>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <script type="text/javascript">
    function showAlert(message){
      alert(message);
    }

    function loadURL(url) {
      var iFrame;
      iFrame = document.createElement("iframe");
      iFrame.setAttribute("src", url);
      iFrame.setAttribute("style", "display:none;");
      iFrame.setAttribute("height", "0px");
      iFrame.setAttribute("width", "0px");
      iFrame.setAttribute("frameborder", "0");
      document.body.appendChild(iFrame);
      // 发起请求后这个 iFrame 就没用了，所以把它从 dom 上移除掉
      iFrame.parentNode.removeChild(iFrame);
      iFrame = null;
    }
    function firstClick() {
      loadURL("firstClick://shareClick?title=分享的标题&content=分享的内容
&url=链接地址&imagePath=图片地址");
    }
  </script>
</header>

<body>
  <h2> 这里是第一种方式 </h2>
  <br/>
  <br/>
  <button type="button" onclick="firstClick()">Click Me!</button>

</body>
</html>
```

然后在项目的控制器中实现UIWebView的代理方法：



```
#pragma mark - UIWebViewDelegate
- (BOOL)webView:(UIWebView *)webView shouldStartLoadWithRequest:(NSURLRequest *)request navigationType:(UIWebViewNavigationType)navigationType
{
    NSURL *url = [request URL];
    if ([[url scheme] isEqualToString:@"firstclick"]) {
        NSArray *params = [url.query componentsSeparatedByString:@"&"];

        NSMutableDictionary *tempDic = [NSMutableDictionary dictionary];
        for (NSString *paramStr in params) {
            NSArray *dicArray = [paramStr componentsSeparatedByString:@"="];
            if (dicArray.count > 1) {
                NSString *decodeValue = [dicArray[1] stringByReplacingPercentEscapesUsingEncoding:NSUTF8StringEncoding];
                [tempDic setObject:decodeValue forKey:dicArray[0]];
            }
        }

        UIAlertView *alertView = [[UIAlertView alloc] initWithTitle:@"方式一" message:@"这是OC原生的弹出窗" delegate:self cancelButtonTitle:@"收到" otherButtonTitles:nil];

        [alertView show];
        NSLog(@"tempDic:%@", tempDic);
        return NO;
    }

    return YES;
}
```

注意：1. JS中的firstClick,在拦截到的url scheme全都被转化为小写。

- 2.html中需要设置编码，否则中文参数可能会出现编码问题。
- 3.JS用打开一个iFrame的方式替代直接用document.location的方式，以避免多次请求，被替换覆盖的问题。

早期的JS与原生交互的开源库很多都是用得这种方式来实现的，例如：
PhoneGap、WebViewJavascriptBridge (<https://link.jianshu.com?t=https://github.com/marcuswestin/WebViewJavascriptBridge>)。关于这种方式调用OC方法，唐巧早期有篇文章有过介绍：
[关于UIWebView和PhoneGap的总结 \(https://link.jianshu.com?t=http://blog.devtang.com/blog/2012/03/24/talk-about-uiwebview-and-phonegap/\)](https://link.jianshu.com?t=http://blog.devtang.com/blog/2012/03/24/talk-about-uiwebview-and-phonegap/)

方式二

在iOS 7之后，apple添加了一个新的库JavaScriptCore，用来做JS交互，因此JS与原生OC交互也变得简单了许多。
首先导入JavaScriptCore库, 然后在OC中获取JS的上下文

```
JSContext *context = [self.webView valueForKeyPath:@"documentView.webView.mainFrame.javascriptContext"];
```

再然后定义好JS需要调用的方法，例如JS要调用share方法：
则可以在UIWebView加载url完成后，在其代理方法中添加要调用的share方法：

```
- (void)webViewDidFinishLoad:(UIWebView *)webView
{
    JSContext *context = [self.webView valueForKeyPath:@"documentView.webView.mainFrame.javascriptContext"];
    //定义好JS要调用的方法，share就是调用的share方法名
    context[@"share"] = ^() {
        NSLog(@"++++++Begin Log++++++");
        NSArray *args = [JSContext currentArguments];

        dispatch_async(dispatch_get_main_queue(), ^{
            UIAlertView *alertView = [[UIAlertView alloc] initWithTitle:@"方式二"
            message:@"这是OC原生的弹出窗" delegate:self cancelButtonTitle:@"收到" otherButtonTitles:nil];
            [alertView show];
        });

        for (JSValue *jsVal in args) {
            NSLog(@"%@", jsVal.toString);
        }

        NSLog(@"-----End Log-----");
    };
}
```

注意：

可能最新版本的iOS系统做了改动，现在（iOS9，Xcode 7.3，去年使用Xcode 6和iOS 8没有线程问题）中测试,block中是在子线程，因此执行UI操作，控制台有警告，需要回到主线程再操作UI。

其中相对应的html部分如下：

```
<html>
  <header>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <script type="text/javascript">

        function secondClick() {
            share( '分享的标题', '分享的内容', '图片地址' );
        }

        function showAlert(message){
            alert(message);
        }

    </script>
  </header>

  <body>
    <h2> 这里是第二种方式 </h2>
    <br/>
    <br/>
    <button type="button" onclick="secondClick()">Click Me!</button>

  </body>
</html>
```

JS部分确实要简单的多了。

OC调用JS篇

方式一

```
NSString *jsStr = [NSString stringWithFormat:@"showAlert('%@')",@"这里是JS中alert弹出的message"];
[_webView stringByEvaluatingJavaScriptFromString:jsStr];
```

注意：该方法会同步返回一个字符串，因此是一个同步方法，可能会阻塞UI。

方式二

继续使用JavaScriptCore库来做JS交互。

```
JSContext *context = [self.webView valueForKeyPath:@"documentView.webView.mainFrame.javascriptContext"];
NSString *textJS = @"showAlert('这里是JS中alert弹出的message')";
[context evaluateScript:textJS];
```

重点：

stringByEvaluatingJavaScriptFromString 是一个同步的方法，使用它执行JS方法时，如果JS 方法比较耗的时候，会造成界面卡顿。尤其是js 弹出alert 的时候。alert 也会阻塞界面，等待用户响应，而 stringByEvaluatingJavaScriptFromString 又会等待js执行完毕返回。这就造成了死锁。

官方推荐使用 WKWebView 的 evaluateJavaScript:completionHandler: 代替这个方法。其实我们也有另外一种方式，自定义一个延迟执行alert 的方法来防止阻塞，然后我们调用自定义的alert 方法。同理，耗时较长的js 方法也可以放到setTimeout 中。

```
function asyncAlert(content) {
    setTimeout(function(){
        alert(content);
    },1);
}
```

我也写了一个demo,包括JS与OC交互的两种方式。
JS_OC_summary (https://link.jianshu.com?t=https://github.com/Haley-Wong/JS_OC/tree/master/JS_OC_summary)

如果你看的还不尽兴，后面还有几篇JS相互调用的文章。

iOS下JS与OC互相调用（一）--UIWebView 拦截URL
(<https://www.jianshu.com/p/7151987f012d>)

iOS下JS与OC互相调用（二）--WKWebView 拦截URL
(<https://www.jianshu.com/p/99c3af6894f4>)

iOS下JS与OC互相调用（三）--MessageHandler
(<https://www.jianshu.com/p/433e59c5a9eb>)

iOS下JS与OC互相调用（四）--JavaScriptCore
(<https://www.jianshu.com/p/4db513ed2c1a>)

iOS下JS与OC互相调用（五）--UIWebView + WebViewJavascriptBridge
(<https://www.jianshu.com/p/2be213e3f673>)

iOS下JS与OC互相调用（六）--WKWebView + WebViewJavascriptBridge
(<https://www.jianshu.com/p/e951af9e5e74>)

iOS下JS与OC互相调用（七）--Cordova 基础
(<https://www.jianshu.com/p/78e486b31953>)

iOS下JS与OC互相调用（八）--Cordova详解+实战
(<https://www.jianshu.com/p/e74bc7abac8d>)

剩下的几篇未完待续。

小礼物走一走，来简书关注我

赞赏支持



Haley_Wong (/u/a8cf6d63e889)

写了 79307 字, 被 2076 人关注, 获得了 1488 个喜欢

(/u/a8cf6d63e889)写了 79307 字, 被 2076 人关注, 获得了 1488 个喜欢

+ 关注

Brighter! Brighter! Brighter!

♡ 喜欢 (/sign_in?utm_source=desktop&utm_medium=not-signed-in-like-button)

252







更多分享

(http://cwb.assets.jianshu.io/notes/images/2366399)



下载简书 App ▶

随时随地发现和创作内容



(/apps/download?utm_source=nbc)




后发表评论

(/sign_in?utm_source=desktop&utm_medium=not-signed-in-comment-form)

87条评论

只看作者

按喜欢排序 按时间正序 按时间倒序



小小聪明屋123 (/u/f9752836b891)

19楼 · 2016.10.18 11:10

(/u/f9752836b891)

你好，想问一下，网页加载完成，如何实现自动调用这个方法，而不是通过点击事件；我们后台那边回调我们iOS的方法时，无法实现自动调用，只能通过点击才能回调，安卓那边是可以自动调用的，请问这个问题该怎样解决？

👍 2人赞

💬 回复

Haley_Wong (/u/a8cf6d63e889): @小小聪明屋123 (/users/f9752836b891) 跟点击按钮一样的调用方式，没能自动调用成功，肯定是他们调用的时机不对，要在dom 加载完毕之后才行。

2016.10.18 13:37 💬 回复

冷碎音阶丿寒 (/u/5bf172817eb2): 我也遇到不自动调用的问题，请问怎么解决的？可以指教一下么？我是在webViewDidFinishLoad:(UIWebView *)webView方法里直接调用一个方法 但是发现根本不进block，其实我只是测试下会不会自动调用，因为我需要在完成加载后把app登陆信息返回过去，如果不能自动调用是不是意味着也不能把信息返回回去呢？

2017.05.09 17:49 💬 回复

我只是前端 (/u/937f2cd2bd28): 我也遇到这种情况，我看前端代码写在window.onload里的，window.onload = function(){log('测试');}，这样写应该在页面加载完之后自动调用log方法，但是我并没有收到，求助

2017.08.18 14:14 💬 回复

✎ 添加新评论

还有1条评论， 展开查看



无心可活 (/u/0aaa4562cc17)

2楼 · 2015.11.18 00:46

(/u/0aaa4562cc17)

嗯嗯

👍 1人赞

💬 回复



流浪在简书 (/u/c6866a2b7547)

3楼 · 2016.02.26 18:57

(/u/c6866a2b7547)

一段代码写完，用啥标签结束啊😄



赞



回复

Haley_Wong (/u/a8cf6d63e889): @流浪着的梦想家 (/users/c6866a2b7547)

两个``包起来就行了，代码放中间

2016.04.28 09:11 回复

流浪在简书 (/u/c6866a2b7547): @Halley_Wong (/users/a8cf6d63e889) 🍻

2016.04.29 23:28 回复

添加新评论



Xcode8 (/u/c1f545d5de1e)

4楼 · 2016.03.17 11:26

(/u/c1f545d5de1e)

很详细，谢谢



赞



回复



wosicuanqi (/u/e1a17979c440)

5楼 · 2016.04.26 18:45

(/u/e1a17979c440)

你说OC调用JS的方式一会造成阻塞UI，可是我在测试中没发现什么异常，反而方法二出现了几次卡死的现象。



赞



回复

Haley_Wong (/u/a8cf6d63e889): @wosicuanqi (/users/e1a17979c440) 你可以看下方式一的描述，而且该方法确实是个同步方法。另外因为示例demo中用的是alert弹出窗，UIWebView的AlertView弹出窗有时候会导致界面卡死，你遇到的卡死可能是这个原因。

2016.04.28 09:14 回复

无星... (/u/fef7696741c8): @Haley_Wong (/users/a8cf6d63e889)

UIWebView的AlertView弹出窗有时候会导致界面卡死，这个是原生的原因吗？

我在做ReactNative开发的时候遇到过，还以为是ReactNative的坑。。。

2017.11.15 09:21 回复

添加新评论



coderChrisLee (/u/18d86fcb7231)

6楼 · 2016.05.03 11:18

(/u/18d86fcb7231)

JS调用原生OC，我想用的是方式二，即iOS7之后像你所说的那样，提供一个接口（方法名）给后台，就是你例子中的share，我取名叫backToRootButtonClicked,但是后台开发人员说这个方法名，在他们那里会因为方法未定义找不到方法来源而报一个本地的错误，这个后端怎么解决呢？



赞




回复

Haley_Wong (/u/a8cf6d63e889): @coderChrisLee (/users/18d86fcb7231) 如果你下载了demo测试的话，应该是没有问题的。你们的后端开发人员的这个错误，可能是他们开发工具的自动检测导致的（如eclipse就会自动校验，可以在设置里关掉相关校验）。


2016.05.04 11:46 回复


镜湖渔叟 (/u/2ce7ffc1c8cf): @Haley_Wong (/users/a8cf6d63e889) js在调用的时候找不到ios提供的这个方法，在js中调用这个方法时候，直接写share ()

就行吗？不用加上什么对象什么的吗？

2016.12.29 14:56  回复

coderChrisLee (/u/18d86fcb7231): 后面我找到了其他的解决办法。通过继承自定义一个JSObject对象进行交互很方便。

2016.12.29 15:00  回复

 添加新评论 还有2条评论， [展开查看](#)



liyaoyao (/u/41847721a65f)


7楼 · 2016.05.09 13:44

(/u/41847721a65f)


楼主，你那边有没有试过通过JSExport协议，用js调oc端的方法？

 赞  回复


Haley_Wong (/u/a8cf6d63e889): @41847721a65f (/users/41847721a65f) 没试过啊，可以用JavaScriptCore库的吧


2016.05.09 17:02  回复

liyaoyao (/u/41847721a65f): @Halley_Wong (/users/a8cf6d63e889) 可以用，这几天在试这个框架，但是在js和ios的交互过程中遇到循环引用，然后按照网上找的资料用JSManagedValue 和 JSVirtualMachine去处理，但是没有效果，不知道楼主有没有遇到过？

2016.05.12 15:42  回复

海之号角 (/u/5b7642fe85d4): @41847721a65f (/users/41847721a65f) 同样遇到循环引用的问题

2016.07.18 17:27  回复

 添加新评论



d1425db57301 (/u/d1425db57301)

8楼 · 2016.05.12 09:55

(/u/d1425db57301)

楼主，我用方式2做的，有点问题// 隐藏标签栏

```
context[@"appGotoViewHideTabs"] = ^() {  
// NSArray *args = [JSContext currentArguments];  
// NSString *hideStr = args[0];  
NSLog(@"隐藏标签栏");  
[self hideView];  
};
```

可以断点到 context[@"appGotoViewHideTabs"] = ^() {


里面的打印进不去，请指导下，谢谢


 赞  回复

Haley_Wong (/u/a8cf6d63e889): @d1425db57301 (/users/d1425db57301) 里面的打印是在JS里调用appGotoViewHideTabs这个方法之后才会走的啊。

还有设置JS可以调用的原生方法得等webView加载完毕后再设置；

还有问题的话，尝试修改下JS调用的原生方法的方法名大小写，比如全用小写字母。

2016.05.12 11:57  回复

 添加新评论



小茗 (/u/1d25847912af)

9楼 · 2016.06.24 17:19

(/u/1d25847912af)

可以讲解一下WebViewJavascriptBridge吗,感觉这个很实用

 赞  回复



longdawei1988 (/u/13b4a08e7624)

10楼 · 2016.07.29 07:25

(/u/13b4a08e7624)

今天遇到WebViewJavascriptBridge与NJKWebViewProgressView冲突的问题，去掉了WebViewJavascriptBridge用原生的代替了，非常好用简洁，感谢楼主。



赞



回复



longdawei1988 (/u/13b4a08e7624)

11楼 · 2016.07.29 07:38

(/u/13b4a08e7624)

但是遇到了问题context["@xxx"]=^{

这里面操作ui会报警告，然后我在gcd中做的

dispatch_sync(dispatch_get_main_queue(), ^{}), 但是取参数放这里会取不到参数。

}



赞



回复

Haley_Wong (/u/a8cf6d63e889): @longdawei1988 (/users/13b4a08e7624)
你不要整个都放gcd里，只把操作ui的部分放gcd里，获取参数的部分还跟原来一样就好啦

2016.07.29 07:45 回复



添加新评论



啊哈哈哈哈哈群 (/u/0a5587655287)

12楼 · 2016.08.20 10:44

(/u/0a5587655287)

作者你好，我现在根据你的步骤来写 和JS也约定好了 为什么我获取不到他的数据啊 急！在线等 谢谢



赞



回复

Haley_Wong (/u/a8cf6d63e889): @啊哈哈哈哈哈群 (/users/0a5587655287)
你用的是那种方式？

2016.08.21 16:18 回复

啊哈哈哈哈哈群 (/u/0a5587655287): @Haley_Wong (/users/a8cf6d63e889)
第二种 你看到后请加个QQ595274189 跪谢啦

2016.08.22 11:44 回复

啊哈哈哈哈哈群 (/u/0a5587655287): @Haley_Wong (/users/a8cf6d63e889)
现在可以拿到 但是要点击一个Button 才会走NSArray *args = [JSContext currentArguments]; 什么情况啊？

2016.08.22 11:55 回复



添加新评论 | 还有1条评论， 展开查看



今天中午吃什么 (/u/c39f787762b4)

13楼 · 2016.09.01 09:05

(/u/c39f787762b4)

你好博主。我使用NSString *jsStr = [NSString

stringWithFormat:@"showAlert('%@')",@"这里是JS中alert弹出的message"];

[_webView stringByEvaluatingJavaScriptFromString...来传给h5让他弹窗，传递的内容是中文，为什么他那边收到的是乱码了...就是像没有utf-8过的字符串的样子、



赞



回复

Haley_Wong (/u/a8cf6d63e889): @今天中午吃什么 (/users/c39f787762b4)
H5里的编码格式可能不是UTF8导致的

2016.09.01 10:16 回复

今天中午吃什么 (/u/c39f787762b4): @Haley_Wong (/users/a8cf6d63e889)
现在我把一个数组转化为data再转成json字符串这样子传，他说我格式不对
2016.09.01 11:41 回复

添加新评论



dong_liang (/u/398b51c76cc9)
14楼 · 2016.09.03 03:23

(/u/398b51c76cc9)
function getId(){
alert(1)
}
self.context = [webView
valueForKeyPath:@"documentView.webView.mainFrame.javascriptContext"];
NSString *jsFuncStr = @"getId()";
[self.context evaluateScript:jsFuncStr];我这么写为什么不调js的方法呢?新手求教啊,谢谢

赞 回复



wokenshin (/u/63cd2480bff2)
15楼 · 2016.09.21 17:07

(/u/63cd2480bff2)
文章简单易懂 重实践。多谢楼主分享！👏

赞 回复

1

2

3

下一页

被以下专题收入，发现更多相似内容



iOS (/c/a226d0dde954?utm_source=desktop&utm_medium=notes-included-collection)



iOS开发 (/c/152889fde467?utm_source=desktop&utm_medium=notes-included-collection)



iOS-Vendor (/c/0e70dd8ea764?utm_source=desktop&utm_medium=notes-included-collection)



我的测试收藏 (/c/06ab95f30cd0?utm_source=desktop&utm_medium=notes-included-collection)



iOS开发专区 (/c/cb1fb209cddd?utm_source=desktop&utm_medium=notes-included-collection)



Reactiv... (/c/00959e944fcc?utm_source=desktop&utm_medium=notes-included-collection)



iOS开发笔录 (/c/86b2419c2451?utm_source=desktop&utm_medium=notes-included-collection)

展开更多

推荐阅读

更多精彩内容 > (/)

iOS下JS与OC互相调用（二）--WKWebView 拦截URL (/p/99c3af6894f...

在上篇文章中讲述了使用UIWebView拦截URL的方式来处理JS与OC交互。由于UIWebView比较耗内存，性能上不太好，而苹果在iOS 8中推出了WKWebView。同样的用WKWebView也可以拦截URL，做JS 与OC交

utm_campaign=maleskine&utm_content=user&utm_medium=pc_all_hots&utm_source=recommendation)

RunLoop 总结：RunLoop的应用场景（二）让Timer...

上一篇讲了使用RunLoop保证子线程的长时间存活，而不是执行完任务后就立刻销毁的应用场景。这一篇就讲述一下RunLoop如何保证NSTimer在视图滑动

Haley_Wong (/u/a8cf6d63e889?

utm_campaign=maleskine&utm_content=user&utm_medium=pc_all_hots&utm_source=recommendation)

读过最烂的二本学校，我也曾体验过你的堕落与挣扎 (/...

文/追风筝的哈桑 首语：研一过半，曾经的岁月对我意味着什么？这些年的挣扎与堕落，不断地自我救赎与渴望，正如帕斯卡尔曾言：人是一根脆弱的芦苇，唯

追风筝的哈桑 (/u/d5528e1b434e?

utm_campaign=maleskine&utm_content=user&utm_medium=pc_all_hots&utm_source=recommendation)

年薪10万，你就打败了90%的中国人 (/p/193136562a...

撰文 | 张帆 出品 | 网易浪潮工作室 如果你有一些爱分享职场鸡汤的朋友，那你一定会看到很多职场热文，比如《年薪10万和年薪100万的人差在哪里》、《为

浪潮工作室 (/u/ce9cdf041cb?

utm_campaign=maleskine&utm_content=user&utm_medium=pc_all_hots&utm_source=recommendation)

你不是因为优秀，才值得被爱 (/p/87f4d54d3123?ut...


文/韩大爷的杂货铺 1. 一次我问朋友：目前为止，你听到过的最让你感动的一句话，是什么？ 朋友说，还真有一句，出自他老婆。那年朋友大学毕业，初涉职

韩大爷的杂货铺 (/u/3e2c151e2c9d?

utm_campaign=maleskine&utm_content=user&utm_medium=pc_all_hots&utm_source=recommendation)

iOS下JS与OC互相调用（一）--UIWebView 拦截URL (/p/7151987f012d?...

最近准备把之前用UIWebView实现的JS与原生相互调用功能，用WKWebView来替换。顺便搜索整理了一下JS 与OC 交互的方式，非常之多啊。目前我已知的JS 与 OC 交互的处理方式： 1.在JS 中做一次URL跳转，

 Haley_Wong (/u/a8cf6d63e889?

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


(/p/4db513ed2c1a?

Objective-C	JavaScript	JSValue Convert	JSValue Constructor
nil	undefined		valueWithUndefinedContext
NSNumber	null		valueWithNullContext
NSString	string	toString	
NSNumber	number, boolean	toNumber toBoolean toInt32 toInt64	valueWithBooleanContext valueWithBooleanContext valueWithInt32Context valueWithInt64Context
NSDictionary	Object object	toDictionary	valueWithObjectContext
NSArray	Array object	toArray	valueWithNewArrayContext
NSDate	Date object	toDate	
NSBlock	Function object		
id	Wrapper object	toObject	valueWithObjectContext
Class	Constructor object		

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

iOS下JS与OC互相调用（四）--JavaScriptCore (/p/4db513ed2c1a?utm...


前面讲完拦截URL的方式实现JS与OC互相调用，终于到JavaScriptCore了。它是从iOS7开始加入的，用Objective-C 把 WebKit 的 JavaScript 引擎封装了一下，提供了简单快捷的方式与JavaScript交互。关于

 Haley_Wong (/u/a8cf6d63e889?

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

iOS 第三方库、插件、知名博客总结 (/p/fa0b6f594c36?utm_campaign=...

用到的组件1、通过CocoaPods安装项目名称项目信息 AFNetworking网络请求组件 FMDB本地数据库组件 SDWebImage多个缩略图缓存组件 UICKeyChainStore存放用户账号密码组件 Reachability监测网络状态

 大灰狼的小绵羊哥哥 (/u/524ab6e6e423?

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

[iOS]JS与原生OC互相调用(总结) (/p/3728275e4284?utm_campaign=m...

iOS开发免不了要与UIWebView打交道，然后就要涉及到JS与原生OC交互，今天总结一下JS与原生OC交互的两种方式。JS调用原生OC篇 方式一 第一种方式是用JS发起一个假的URL请求，然后利用UIWebView的

 德山_ (/u/46898049a33f?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

iOS下JS与OC互相调用（五）--UIWebView + WebViewJavascriptBridge...

WebViewJavascriptBridge是一个有点年代的JS与OC交互的库，使用该库的著名应用还挺多的，目前这个库有7000+star。我去翻看了它的第一版本已经是4年前了，在版本V4.1.4以及之前，该库只有一个类和一个

 Haley_Wong (/u/a8cf6d63e889?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

这些职场晋升法则你都知道了吗？ (/p/8ce272b17cc4?utm_campaign=m...

现今的职场早已充满“战火”，要想在竞争如此激烈的环境中脱颖而出获得晋升，就必须懂得并很好的运用职场晋升法则。虽说职场晋升需要看业绩，但更看重的是你如何做人。下面快搜搜小编就给大家讲讲职场的晋

 陈白白白 (/u/1de1cd4906ec?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

(/p/90e4257a1a91?



utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation)
一点点改变就够了 (/p/90e4257a1a91?utm_campaign=maleskine&utm_...

自打踏出校门，接手了我人生中第一份非兼职的工作之后，我的生物钟就发生了天翻地覆的变化了。以前在学校的时候，自认为寄几还是蛮乖的（可以理解为有点傻），每天差不多都是重复的动作，起床，上课，吃

 疯狂的拾年 (/u/3845d571d880?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

(/p/23ff343f741b?




utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation)
干货||新生基础护肤指南（男女都适用） (/p/23ff343f741b?utm_campaig...

关于护肤这件事走过很多弯路，毁过皮，用过很多鸡肋产品，终于总结出合适的方式，筛掉很多雷区。护肤作为一门必修课没有人来讲，但却一定要养成良好的护肤习惯。这篇文章写给走入护肤误区的童鞋们，也算

 Juno修行僧 (/u/a7ee7b64e8ac?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)


重读《白鹿原》 (/p/e87e2e2f069a?utm_campaign=maleskine&utm_co...

重读《白鹿原》，最欣赏书里的白嘉轩。鹿三死时，白嘉轩说白鹿原上最好的长工去了；朱先生死时，他说白鹿原上最好的先生去了；当他去世时，完全可以称得上白鹿原上腰杆子最直的人去了，即使后来他的腰被

 麦田的鸽子 (/u/6138db855894?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

《面对巨人》 (/p/d69a5eed3fd7?utm_campaign=maleskine&utm_cont...

20160429 虽说是一部超经典的影片，我却却是第一次听说，有点惭愧。魔力盒App果然很神奇，那些小片段大部分都选自经典影片，可以在短时间内接触到一部影片最精彩的部分，也可以通过它来了解更精彩的世界

 Annie_Baby (/u/23199ba040c2?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

