

UIWebView

webview用来展示网页，webview还可以与H5进行交互

博客链接1: [<http://blog.csdn.net/daiyelang/article/details/40989389>]

博客链接2: [<http://blog.csdn.net/luobo140716/article/details/45046909>]

一、基础

- 加载的三种方式

```
// 根据URL加载
(void)loadRequest:(NSURLRequest *)request;

// 根据加载HTML加载
(void)loadHTMLString:(NSString *)string baseURL:(nullable NSURL *)baseURL;

// 根据文件路径加载 (用来加载本地文件)
(void)loadData:(NSData *)data MIMEType:(NSString *)MIMEType textEncodingName:(NSString *)textEncodingName baseURL:(NSURL *)baseURL;
```

- 常用属性

```
@property (nonatomic, readonly, strong) UIScrollView *scrollView

@property (nullable, nonatomic, readonly, strong) NSURLRequest *request

// 常用方法
1. [webView goBack];           // 返回
2. [webView goForward];        // 向前
3. [webView reload];           // 重新加载数据 (重载)
4. [webView stopLoading];      // 停止加载数据 (取消载入内容)

自动对页面进行缩放以适应屏幕
_webview.scalesPageToFit = YES;
```

- Objective-C to JavaScript:

这个方法传入的就是JavaScript代码，直接在UIWebView上面执行js方法

```
(nullable NSString *)stringByEvaluatingJavaScriptFromString:(NSString *)script
```

注：JavaScript的执行时间被限定在10秒钟，如果执行时间超过10秒，那么页面就停止执行这个脚本。JavaScript的执行或许能够阻塞主线程，所以当脚本执行的时候不允许用户影响页面的加载。

JavaScript的内存分配被限制在10M，如果超出这个限制那么页面会发生异常。

- JavaScript to Objective-C

在这个代理方法里面截住请求，分析一下URL里面的参数，进行相应的处理。

```
- (BOOL)webView:(UIWebView *)webView shouldStartLoadWithRequest:(NSURLRequest *)request navigationType:(UIWebViewNavigationType)navigationType;
```

二、属性简介

- 如果你希望网页视图可以自动检测电话号码,让用户可以单击号码进行拨打,可以将该属性设置为YES

```
@property (nonatomic) BOOL detectsPhoneNumbers
```

- 可以设定使电话号码、网址、电子邮件和符合格式的日期等文字变为链接文字

```
@property (nonatomic) UIDataDetectorTypes dataDetectorTypes
```

- 这个值决定了用内嵌HTML5播放视频还是用本地的全屏控制

```
@property (nonatomic) BOOL allowsInlineMediaPlayback NS_AVAILABLE_IOS(4_0)
```

- 这个值决定了HTML5视频可以自动播放还是需要用户去启动播放

```
@property (nonatomic) BOOL mediaPlaybackRequiresUserAction
```

- 这个值决定了从这个页面是否可以Air Play

```
@property (nonatomic) BOOL mediaPlaybackAllowsAirPlay
```

- 这个值决定了是否在把内容全部加载到内存中再去渲染网页。

```
@property (nonatomic) BOOL suppressesIncrementalRendering NS_AVAILABLE_IOS(6_0)
```

- 如果设置为YES，用户必须明确的点击页面上的元素或者相关联的输入页面来显示键盘；如果设置为NO，一个元素的焦点事件导致输入视图的显示和自动关联这个元素。

```
@property (nonatomic) BOOL keyboardDisplayRequiresUserAction
```

- 设置页面分页模型选择

```
@property (nonatomic) UIWebPaginationMode paginationMode
```

- 这个属性决定了CSS属性是采用column-break 还是page-breaking样式

```
@property (nonatomic) UIWebPaginationBreakingMode paginationBreakingMode
```

- 分页的长度

```
@property (nonatomic) CGFloat pageLength NS_AVAILABLE_IOS(7_0);
```

- 多个页面之间差距值

```
@property (nonatomic) CGFloat gapBetweenPages NS_AVAILABLE_IOS(7_0);
```

- 分页的个数。

```
@property (nonatomic, readonly) NSUInteger pageCount NS_AVAILABLE_IOS(7_0);
```

- 允许图片在图片媒体播放。

@property (nonatomic) BOOL allowsPictureInPictureMediaPlayback

- 允许链接预览

@property (nonatomic) BOOL allowsLinkPreview