

# UITextView 显示网页内容(富文本)

1. 简介：富文本 NSAttributedString，可以设置特殊的文字，比如部分文字高亮、行间距、字间距等。这里不详细阐述，主要介绍一中思想：用富文本巧妙地显示 html 内容(如自定义聊天窗口数据入库是 html 字符串)，并且设置相关超链接，比如点击了 html 中的一段股票代码，将股票添加至行情列表。

2. 几个疑问：

(1) 为什么不用 label 呢？label 不是也可以设置富文本？

那是因为，label 设置富文本去展示 html 内容，但是不能进行交互。

(2) 为什么不用 webView 呢？webView 不就是用来展示 html 的嘛？然后 js 交互可以用 javascriptcore 框架实现。

那是因为，很多情况下，比如聊天界面单条数据这种展示 html 内容少并且 cell 又很多的，总不能创建多个 webview 去加载把？会不会很浪费资源？

3. 实现：

(1) 初始化 textView，设置代理

```
self.textView = [[UITextView alloc] initWithFrame:CGRectMake(0, 100, [[UIScreen mainScreen] bounds].size.width,
    [[UIScreen mainScreen] bounds].size.height)];
_textView.delegate = self;
_textView.dataDetectorTypes = UIDataDetectorTypeLink;
[self.view addSubview:_textView];
```

```
NSString *html_str = [NSString stringWithFormat:@"'的发生<a href='http://baidu.com'>001001</a> "];
```

```
NSMutableAttributedString *mutableAttri=[[NSMutableAttributedString alloc]initWithData:[html_str dataUsingEncoding:
    NSUnicodeStringEncoding] options:@{ NSDocumentTypeDocumentAttribute: NSHTMLTextDocumentType }
    documentAttributes:nil error:nil];
```

```
_textView.editable = NO;
```

```
[_textView setAttributedText:mutableAttri];
```

(2) 实现代理方法

```
#pragma mark UITextViewDelegate
```

```
- (BOOL)textView:(UITextView *)textView shouldInteractWithURL:(NSURL *)URL inRange:(NSRange)characterRange {
    NSString *string = [textView.text substringWithRange:characterRange];
    UIAlertController *alert = [UIAlertController alertControllerWithTitle:@"提示" message:[NSString stringWithFormat:@"点击的股票代码为: \n%@", string]
                             preferredStyle:UIAlertControllerStyleAlert];
    UIAlertAction *cancel = [UIAlertAction actionWithTitle:@"确定" style:UIAlertActionStyleCancel handler:^(UIAlertAction * _Nonnull action) {
    }];
    [alert addAction:cancel];
    [self presentViewController:alert animated:YES completion:nil];
    return NO;
}
```

#### 4. 讲解:

这里巧妙地利用了 UITextView 的自动内容检测功能, 将 `dataDetectorTypes` 设为 `UIDataDetectorTypeLink`, 即自动检测 url 链接, 然后给股票代码添加 `<a href=' ' http:// ' >股票代码<a>` 标签, 则当用户点击相应股票代码会触发代理方法, 拦截 url (代理方法返回 NO), 进行其他处理。

#### 5. 拓展:

假如不用 UITextView 的自动检索功能, 则自己需编写正则匹配出股票代码的数组 (含股票代码和相应字符串中的 range), 然后调用

```
[str addAttribute:NSLinkAttributeName value:[NSURL
URLWithString:[NSString stringWithFormat:@"stock:%@", [obj
objectForKey:@"string"]]] range:[[obj objectForKey:@"range"]
rangeValue]];
```

然后可以在代理方法里获取到自定义的 url 进行处理, 具体如下:

```
self.textView = [[UITextView alloc] initWithFrame:CGRectMake(0, 100, [[UIScreen mainScreen] bounds].size.width, [[UIScreen mainScreen] bounds].size.height)];
_textView.delegate = self;
[self.view addSubview:_textView];
```

```
NSString *html_str = [NSString stringWithFormat:@"<a href='http://my.oschina.net/duxinfeng'>%@</a> abcd300596abcd测试时间dcba002841dcba
%@" , @"新风作浪", @"2013-05-27"];
```

```
NSMutableAttributedString *str = [[NSMutableAttributedString alloc] initWithString:html_str attributes:nil];
_textView.editable = NO;
```

//匹配002或300开头的股票代码

```
[[self regexIsStockCode:html_str pattern:@"((002|300)(\\d){3})"] enumerateObjectsUsingBlock:^(NSDictionary *obj, NSUInteger idx, BOOL * _Nonnull stop) {
    [str addAttribute:NSLinkAttributeName value:[NSURL URLWithString:[NSString stringWithFormat:@"stock:%@", [obj objectForKey:@"string"]]] range:[[obj
    objectForKey:@"range"] rangeValue]];
}];
[_textView setAttributedText:str];
```

#pragma mark 正则匹配

```
- (NSArray *)regexIsStockCode:(NSString *)str pattern:(NSString *)pattern{
    NSMutableArray *result = [NSMutableArray array];

    NSRegularExpression *regular = [NSRegularExpression regularExpressionWithPattern:pattern options:NSRegularExpressionCaseInsensitive error:nil];

    NSTextCheckingResult *row;
    NSEnumerator *enumerator = [[regular matchesInString:str options:NSMatchingReportCompletion range:NSMakeRange(0, str.length)] objectEnumerator];
    while (row = [enumerator nextObject]) {
        NSString *string = [str substringWithRange:row.range];
        [result addObject:@{@"string":string, @"range":[NSValue valueWithRange:row.range]}];
    }
    return result;
}
```

#pragma mark UITextViewDelegate

```
- (BOOL)textView:(UITextView *)textView shouldInteractWithURL:(NSURL *)URL inRange:(NSRange)characterRange {
    UIAlertController *alert = [UIAlertController alertControllerWithTitle:@"提示" message:[NSString stringWithFormat:@"点击的链接为: \n%@", URL.absoluteString] preferredStyle:UIAlertControllerStyleAlert];
    UIAlertAction *cancel = [UIAlertAction actionWithTitle:@"确定" style:UIAlertActionStyleCancel handler:^(UIAlertAction * _Nonnull action) {
    }];
    [alert addAction:cancel];
    [self presentViewController:alert animated:YES completion:nil];
    return YES;
}
```