富文本

文字排版处理使用框架 CoreText ， TextKit ，其中coreText属于底层框架，TextKit在是对CoreText的再次封装。本文使用一个UIkit框架中的uiLabel，uitextField中的attributedString属性来处理文字

NSMutableAttributedString(可变文字) ，AttributedString(不可变)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*不使用字典的设置\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

///定义一个要编辑的文字

let Str = “Hello, Swift!”

/// 初始化对象

let orignStr = NSMutableAttributedString(string: Str)

/// 分段修改文字颜色和字体

orignStr.addAttribute(NSFontAttributeName, value: UIFont(name: "Arial", size: win\_c \* 17)!, range: NSMakeRange(0, (Str as NSString).length))

/// 分段控制，设置不同段字体颜色 从0开始后面5个设置为红色

orignStr.addAttribute(NSForegroundColorAttributeName, value: UIColor.redColor(), range: NSMakeRange(0, 5))

/// 添加设置后的属性到lable，textField

label.text = orignStr

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*使用字典的设置\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/// 定义要设置的字符串

let Str2: NSString = "dic,Swift!"

/// 设置字典 (颜色和字体)

let attrdic = [ NSForegroundColorAttributeName: UIColor.brownColor(),NSFontAttributeName: UIFont.boldSystemFontOfSize(18)]

/// 初始化对象

/// 从字符串第一个开始，字符串倒数第7个结束

let range = Str2.startIndex..<Str.endIndex.advancedBy(-7)

/// 增加修改的文字

let newOrign = NSMutableAttributedString(string: Str2[range], attributes: attrdic)

/// 最后同不使用字典方式 , 添加对象到label的attributedText属性。

anyLabel.attributedText = newOrign3

/// 封装函数

// MARK: - 富文本封装

// paramer: dic 传入修改字符的颜色、字体等

// paramer: str 传入字符串func canvertString(dic: [[String: UIColor]] , str: [String]) -> NSMutableAttributedString?{ }

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*最后介绍 attributedString的属性\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

***// NSFontAttributeName*** *设置字体属性，默认值：字体：Helvetica(Neue) 字号：12*

***// NSForegroundColorAttributeNam*** *设置字体颜色，取值为 UIColor对象，默认值为黑色*

***// NSBackgroundColorAttributeName*** *设置字体所在区域背景颜色，取值为 UIColor对象，默认值为nil, 透明色*

*// NSLigatureAttributeName             设置连体属性，取值为NSNumber 对象(整数)，0 表示没有连体字符，1 表示使用默认的连体字符*

*// NSKernAttributeName                 设定字符间距，取值为 NSNumber 对象（整数），正值间距加宽，负值间距变窄*

*// NSStrikethroughStyleAttributeName  设置删除线，取值为 NSNumber 对象（整数）*

*// NSStrikethroughColorAttributeName  设置删除线颜色，取值为 UIColor 对象，默认值为黑色*

*// NSUnderlineStyleAttributeName       设置下划线，取值为 NSNumber 对象（整数），枚举常量 NSUnderlineStyle中的值，与删除线类似*

*// NSUnderlineColorAttributeName       设置下划线颜色，取值为 UIColor 对象，默认值为黑色*

*// NSStrokeWidthAttributeName         设置笔画宽度，取值为 NSNumber 对象（整数），负值填充效果，正值中空效果*

*// NSStrokeColorAttributeName         填充部分颜色，不是字体颜色，取值为 UIColor 对象*

*// NSShadowAttributeName               设置阴影属性，取值为 NSShadow 对象*

*// NSTextEffectAttributeName          设置文本特殊效果，取值为 NSString 对象，目前只有图版印刷效果可用：*

*// NSBaselineOffsetAttributeName       设置基线偏移值，取值为 NSNumber （float）,正值上偏，负值下偏*

*// NSObliquenessAttributeName         设置字形倾斜度，取值为 NSNumber （float）,正值右倾，负值左倾*

*// NSExpansionAttributeName           设置文本横向拉伸属性，取值为 NSNumber （float）,正值横向拉伸文本，负值横向压缩文本*

*// NSWritingDirectionAttributeName    设置文字书写方向，从左向右书写或者从右向左书写*

*// NSVerticalGlyphFormAttributeName   设置文字排版方向，取值为 NSNumber 对象(整数)，0 表示横排文本，1 表示竖排文本*

*// NSLinkAttributeName                 设置链接属性，点击后调用[浏览器](http://www.2cto.com/os/liulanqi/" \t "_blank)打开指定URL地址*

*// NSAttachmentAttributeName          设置文本附件,取值为NSTextAttachment对象,常用于文字图片混排*

*// NSParagraphStyleAttributeName      设置文本段落排版格式，取值为 NSParagraphStyle 对象*