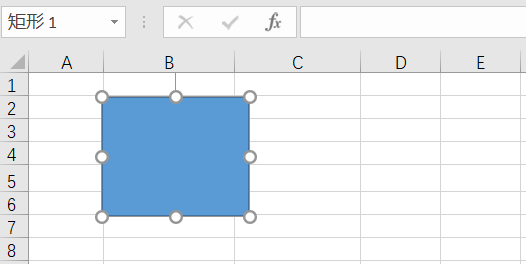
在上篇推文《操作Excel的Xlwings教程（三）》（添加链接）中重点介绍了Xlwings中的Sheet和Range等API知识点。本次推文将继续上次的知识点，将继续介绍Xlwings中API相关的知识，并展示一些小例子，让大家可以跟着教程进行学习：

我们先来看看xlwings中的Shape对象，Office开发文档对Shape进行了介绍，传送门如下：<https://docs.microsoft.com/zh-cn/office/vba/api/excel.shape>，即Shape对象代表绘图层中的对象，例如自选图形、任意多边形、OLE 对象或图片。

**一．Shapes**

假设我们有一个文件1.xlsx，在sheet1内有一个shape矩形，如下图所示：



我们进行运行以下的代码：

import xlwings as xw  
wb = xw.Book(**'1.xlsx'**)  
wb.sheets[0].shapes *# 输出Shapes([<Shape 'Rectangle 1' in <Sheet [1.xlsx]Sheet1>>])*

wb.sheets[0].shapes.count *# 返回集合中的对象数，输出1*

可以看出上述代码输出了这个Excel文件中的Shape对象，且计数了这个所选sheet页中形状的个数，所以在日常的操作中，我们可以使用count来对其中的shape总数进行统计。

**二．Shape**

跟Sheet对象一样，shape对象是[shapes](https://docs.xlwings.org/en/stable/api.html" \l "xlwings.main.Shapes" \o "xlwings.main.Shape)集合的成员：我们来看一下Shape中有哪些方法或属性，我们使用的Excel文件还是1.xlsx。先导入模块，打开文件，引用sheet1工作表：

import xlwings as xw  
wb = xw.Book(**'1.xlsx'**)  
sheet = wb.sheets[0]

方法或属性

xw.Shape(1).activate *# 激活里面的一个shape*

*# 或者 sheet.shapes[0].activate, 使用print打印可查看输出结果*

xw.Shape(1).delete *# 删除里面的一个shape*

xw.Shape(1).height *# height 返回或设置表示形状高度的点数。*

xw.Shape(1).left *# left 返回或设置表示形状水平位置的点数。*

sheet.shapes[0].top *# top 返回或设置表示形状垂直位置的点数*

sheet.shapes[0].width *# width 返回或设置表示形状宽度的点数。*

sheet.shapes[0].type *# type 返回形状的类型。输出 auto\_shape*

xw.Shape(1).name *# name 返回或设置形状的名称,此处输出 Rectangle 1*

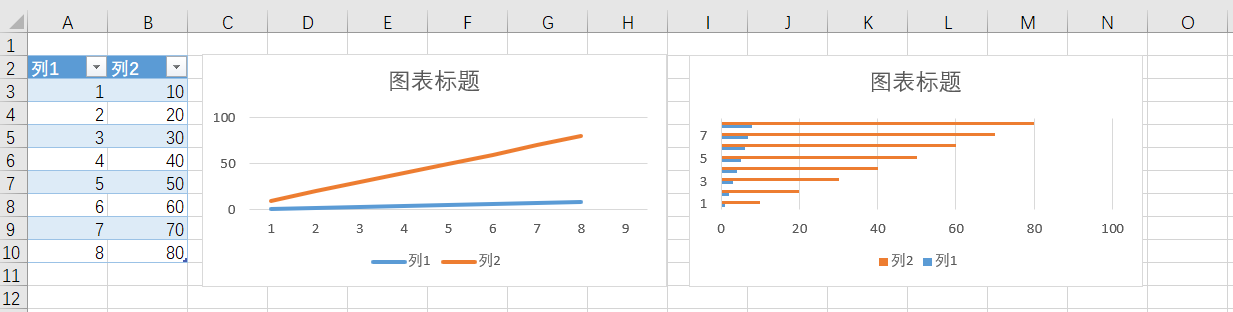
xw.Shape(1).parent *# parent 返回形状的父级,输出<Sheet [1.xlsx]Sheet1>*

Shape的相关操作就介绍到这里了，个人觉得不是很常用的功能，大家有兴趣的可以研究xlwings的main函数，来查看shape对象的使用方法。

**三．Charts**

图表类的对象是Excel中经常碰到的，xlwings中页存在很多操图表的方法，接下来我们介绍xlwings中的图表对象（Charts和Chart）。

假设现在我们有数据表2.xlsx,其sheet页的有一个表格数据生成的两张图表，接下来我们使用xlwings来进行相关的操作：



import xlwings as xw  
wb = xw.Book(**'2.xlsx'**)  
sheet = wb.sheets[0]  
s = sheet.charts  
print(s)

最后一行输出：Charts([<Chart **'Chart 1'** in <Sheet [2.xlsx]Sheet1>>, <Chart **'Chart 2'** in <Sheet [2.xlsx]Sheet1>>]) 即工作表上所有chart对象的集合。

如果想新增一个图表，可使用add()方法，其有四个参数可选left，top，width，height

sheet.charts.add(100,100) *# 会在所选sheet上(100,100)位置新建一个图表(空白表)*

*# left (float, default 0):left position in points  
# top (float, default 0):top position in points  
# width (float, default 355):width in points  
# height (float, default 211):height in points*

sheet.charts.count *# count属性可统计图表的个数，输出2*

**四．Chart**

跟之前介绍的Sheet等对象一样，chart对象是charts集合的成员。

那么chart中有哪些API是值的我们关注的呢？我们来逐个进行介绍：

**引用chart有两种方法**

sheet.charts[0] *# 法一，输出<Chart 'Chart 1' in <Sheet [2.xlsx]Sheet1>>*

sheet.charts[**'图表 2'**] *# 法二，输出<Chart 'Chart 2' in <Sheet [2.xlsx]Sheet1>>*

sheet.charts[0].api *#返回所使用引擎的本机对象*

sheet.charts[0].chart\_type *# 返回并设置图表的图表类型，输出line*sheet.charts[1].chart\_type *# 返回并设置图表的图表类型，输出bar\_clustered*

sheet.charts[0].delete() *# 使用索引删除图表1。*

sheet.charts[0].height *# 返回或设置代表图表高度*sheet.charts[0].left *# 返回或设置代表图表水平位置*sheet.charts[0].top *# 返回或设置代表图表垂直位置*sheet.charts[0].width *# 返回或设置代表图表宽度*

sheet.charts[0].name *# 返回或设置图表名称，输出Chart 1*

sheet.charts[0].parent *# 返回图表的父级对象，输出<Sheet [2.xlsx]Sheet1>*

sheet.charts[0].set\_source\_data() *# 设置图表的源数据范围。*

对于设置图表的源数据范围的方法set\_source\_data()，我们来看一个综合的例子：

我们打开一个已经存在的Excel文件3.xlsx，其sheet工作表有两列数据变量1和变量2，我们使用xlwings操作这个数据生成我们的图表，代码如下：

import xlwings as xw

sheet = xw.Book(**'4.xlsx'**).sheets[0]

chart = sheet.charts.add() *# 新增chart*

*# 数据源：sheet.range('A1:B7')，或者sheet.range('A1').expand()*

chart.set\_source\_data(sheet.range(**'A1'**).expand()) chart.chart\_type = **'line'** *# 设置图标的类型，此处为线型，具体的类型查看office官网VBA操作的手册*title=**'python知识学堂粉丝数'** *#标题名称*

chart.api[1].SetElement(2)

chart.api[1].ChartTitle.Text =title *#设置标题名称*

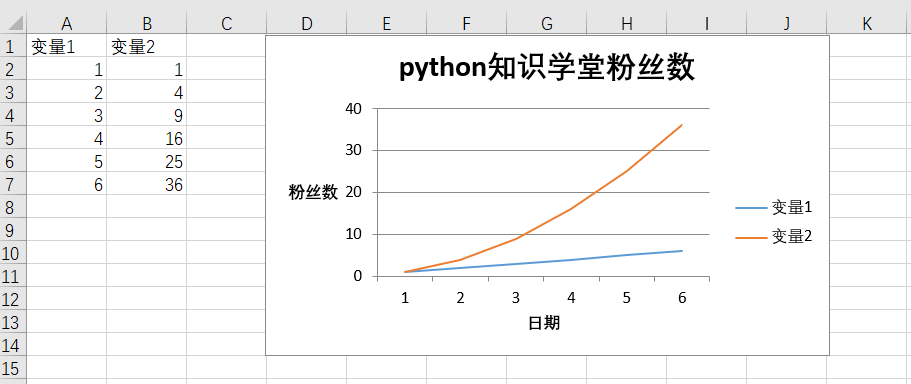
chart.api[1].SetElement(302) *# 横线*

chart.api[1].Axes(1).AxisTitle.Text = **"日期"** *#横轴标题名称*

chart.api[1].SetElement(311)

chart.api[1].Axes(2).AxisTitle.Text = **"粉丝数"** *#纵轴标题名称*

经上述一系列的操作，生成的图表如下，（为了展示图表是经过手动移动位置的，可通过height等属性进行设置）



**五．小结**

以上就是xlwings中Shape和Chart对象的相关知识，纵观全文，干货还是满满的，是不是觉得使用xlwings来操作Excel中的这些对象还是很方便呢？介绍了上述对象API的相关知识，相必你也跃跃欲试了，赶快操作起来吧！

哦，对了，感谢大家近期的关注，由于时间的原因，目前只能一周更新一篇原创推文，文中有什么不对的地方还请大家在公众号后台留言哈！有什么好的建议也欢迎提出。

预告：下期我们将继续讲解其他相关的API知识，如Picture（图片对象）。