一直以来，大家都熟悉WordVBA中可以创建正则对象，利用强大的正则表达式来处理字符，但是大家都熟悉Word文档都是含有诸多的格式、样式以及表格等，虽然这个正则对象是处理字符串之王，但是苦于只能处理“纯文本”，大家既爱又恨！

鉴于以上情况，本帖就是要解决WordVBA中大家既爱又恨正则对象，本帖解决在Word文档含有诸多的格式、样式以及表格等而不改变这些格式、样式及表格等前提下在WordVBA中创建正则来解决两大类型问题：1、修改格式，2、查找并替换字符

前条件：在Word文档含有诸多的格式、样式以及表格等，以下代码在不对原文档做修改下进行

'修改格式

Sub change()

     Dim i As Paragraph, mt, oRang As Range, n%, m%

    With CreateObject("vbscript.regexp")

        .Pattern = "(\[add\](.\*?)\[/add\])"

        .Global = True: .IgnoreCase = False: .MultiLine = True

        For Each i In ActiveDocument.Paragraphs

            For Each mt In .Execute(i.Range.Text)

                m = mt.FirstIndex: n = mt.Length

                Set oRang = ActiveDocument.Range(i.Range.Start + m, i.Range.Start + m + n)

                'MsgBox (oRang)

                oRang.Font.Color = wdColorBlue

                oRang.Font.Underline = wdUnderlineSingle

            Next

        Next

    End With

    With CreateObject("vbscript.regexp")

        .Pattern = "(\[del\](.\*?)\[/del\])"

        .Global = True: .IgnoreCase = False: .MultiLine = True

        For Each i In ActiveDocument.Paragraphs

            For Each mt In .Execute(i.Range.Text)

                m = mt.FirstIndex: n = mt.Length

                Set oRang = ActiveDocument.Range(i.Range.Start + m, i.Range.Start + m + n)

                'MsgBox (oRang)

                oRang.Font.Color = wdColorRed

                oRang.Font.StrikeThrough = True

            Next

        Next

    End With

End Sub

'查找替换

Sub Word文档中查找替换()

    Dim col As New Collection, k&

    Dim i%, mt, oRang As Range, n%, m%

    With CreateObject("vbscript.regexp")

        .Pattern = "" '这里各显其能输入正则表达式（根据自己的需求）

        .Global = True: .IgnoreCase = False: .MultiLine = True

        For Each i In ActiveDocument.Paragraphs

            For Each mt In .Execute(i.Range.Text)

                k = k + 1

                m = mt.FirstIndex: n = mt.Length

                Set oRang = ActiveDocument.Range(i.Range.Start + m, i.Range.Start + m + n)

                col.Add oRang, CStr(k)

            Next

        Next

    End With

    For j = 1 To col.Count

        col(j) = "" '替换为字符(根据自己需要)

    Next

End Sub

从根本上来看，Word 文档不过是一个巨大的字符流。人们倾向于认为文档是单词、句子和段落的集合。但实际上，文档就是一些字符。每个字符都有一定的作用。某些字符是字母、空格或制表符，另一些字符是段落标记或分页符。

但是，文档中如果有表格：（罪魁祸首：单元格结束符），正则中用 \x07 或者 \u0007 来表示这个字符。

1、把整个文档当做“目标文本”的时候，word中自身的find对象是找不到这个“罪魁祸首：单元格结束符”，因此word在统计字符的时候这个“单元格结束符”是“没有统计的”。但是正则又能查找到这个字符，那么就存在一个矛盾：range对象的Start属性是从文档（注意：这时候整个文档作为一个目标字符串）最开始0计算到整个文档未的数量 而通过正则查找到的对象属性FirstIndex的起始位置与range对象的Start属性得到的位置就不准确了，所以会出错！

2、把每一个段落当做“目标文本”的时候，情况就不一样了！那就定位准确了，不详述了。

3、所以在word中使用正则的时候注意：什么时候把word中什么对象（文档、段落、单元格等）作为正则的目标文本来进行查找是至关重要的！！！

4、本帖主要是阐述wordvba中怎么充分利用正则对象来扩展find对象的不足。比如这个单元格结束符find对象就无能为力，当然还有find对象无能为力的还比较多。