

WPS Office 2012 二次开发 FAQ

(基础部分)

1 二次开发的兼容性 FAQ.....	3
1.1 WPS Office 2012 的二次开发开发和微软的 Office 二次开发有什么异同.....	3
1.1.1 模块对象的区别.....	3
1.1.2 枚举常量的区别.....	3
1.1.3 commandbars 调用的区别.....	3
1.2 WPS Office2012 的接口类对象和对应的 DLL 文件.....	4
1.3 原有的与 WPS Office 2003 (WPS2003)、WPS Office 2002 (WPS2002)、WPS Office2005 (WPS2005)、WPS Office 2007 (2007)、WPS Office 2009 (2009) 结合的第三方程序，可以直接和 WPS Office2012 结合运行吗？.....	4
1.4 原有的与 WPS Office 2003 或者 WPS Office 2002 结合的第三方程序，需要做什么改动才能正常和 WPS Office2012 结合运行？.....	4
1.5 调用 WPS 接口 API 方法时要注意 Variant 参数类型的传递。.....	5
1.6 通过注册表修改宏代码的安全级别。.....	5
2 二次开发文字部分 FAQ.....	6
2.1 如何启动 WPS 文字模块？.....	6
2.2 如何退出 WPS 文字模块？.....	6
2.3 如何得到 WPS 中的工具栏数？.....	6
2.4 如何隐藏工具栏？.....	6
2.5 如何使工具栏的按钮失效？.....	7
2.6 如何获得一个文档？.....	7
2.7 如何打开 wps 文档？.....	7
2.8 如何关闭一个文档？.....	7
2.9 如何保存一个文档？.....	7
2.10 如何获得或修改当前用户名？.....	7
2.11 如何在当前文档中添加一个表格？.....	8
2.12 如何获取表格的行和列？.....	8
2.13 如何选中一个表？.....	8
2.14 如何修改一个表格的某个单元格的内容？.....	8
2.15 如何设置一个表格的单元格的高度和宽度？.....	8
2.16 表格增加一行之后，要获取增加表格的内容，必须重新给表格变量赋值.....	9
2.17 如何添加一个书签？.....	9
2.18 如何替换一个书签的内容？.....	10
2.19 如何获取和设置默认打印机？.....	10
2.20 如何打印一篇已打开的文档？.....	10
2.21 如何获取和设置某些文字的字体属性？.....	10
2.22 如何获取和设置段落的格式？.....	11

2.23	如何通过注册表获取 WPS 安装版本及安装路径？	11
2.24	如何在插入对象里添加 WPS 文档对象？	12
2.25	在 WPS 文字中使用 Find 的用法	12
2.26	去掉 Find 替换后出现的对话框	12
2.27	如何通过 MIME 设置实现 WPS 文字嵌入浏览器的方式打开服务器 DOC 或 WPS 文档	12
2.28	如何在破坏书签的情况下设置内容？	13
2.29	为什么在使用 Range.Text 为文档、书签，或用 FormField.Result 为窗体域设置文本内容时，回车符不见了？	13
2.30	如何判断文档的设计模式状态，如何退出设计模式？	13
2.31	如何隐藏表格的智能扩展标记？	13
2.32	如何通过 API 实现不显示表格虚线？	14
3	二次开发电子表格部分 FAQ	14
3.1	如何启动 WPS 电子表格模块？	14
3.2	如何退出电子表格？	14
3.3	如何打开一个电子表格文档？	14
3.4	如何关闭一个电子表格文档？	15
3.5	如何保存一个电子表格文档？	15
3.6	如何设置一个单元格的大小？	15
3.7	如何获取单元格中的内容？	15
3.8	如何设置单元格中的内容？	15
3.9	如何设置单元格中的公式？	16
3.10	如何设置单元格的边框？	16
3.11	如何读取批注的内容	16
3.12	如何添加批注？	16
3.13	如何设置单元格内字体的属性？	17
3.14	如何打印电子表格文档？	17
3.15	如何在插入对象里添加 WPS 表格对象？	17
3.16	WPS 表格中如何在默认位置插入图片	18
3.17	WPS 表格中如何插入图表	18
3.18	通过 MIME 设置 WPS 表格嵌入浏览器方式打开服务器的 et 或 xls 文档	18

1 二次开发的兼容性 FAQ

1.1 WPS Office 2012 的二次开发开发和微软的 Office 二次开发有什么异同

1.1.1 模块对象的区别

WPS Office2012 的二次开发接口体系和微软 Office 的二次开发接口体系基本保持一致。包括：对象树，即对象的组织构成、各对象代表的含义和微软 Office 对象均保持一致；接口函数，即各对象的方法属性的名称和含义均与微软 Office 保持一致。不同之处主要表现在两个方面：模块程序的对象名称和枚举常量的名称。

微软 Office 和 WPS Office 各模块的对象名称对应关系如下：

文字模块：	word.Application	——	wps.Application
表格模块：	Excel.Application	——	et.Application
演示模块：	PowerPoint.Application	——	wpp.Application

具体使用时，只需要替换对应的对象名称即可，如下代码：

下面的代码可以创建一个微软 word 进程

```
Set app = createObject("Word.Application")
```

下面的代码可以创建一个 WPS 文字进程

```
Set app = createObject("wps.Application")
```

1.1.2 枚举常量的区别

枚举常量通常出现在 API 方法的参数或 API 属性的赋值上，微软 Office 和 WPS Office 各模块的对象名称对应关系如下：

公用层：	mso	——	kso
文字模块：	wd	——	wps
表格模块：	xl	——	et
演示模块：	ppt	——	wpp

具体使用时，替换对应的名称或使用枚举常量对应的数值，也可以在 WPS 的 VBA 环境中查找对应的枚举常量确认其存在而且数值和微软 OFFICE 的一致

下面的代码可以打印当前页：

WORD 的代码：ActiveDocument.**PrintOut** Range:=wdPrintCurrentPage

WPS 的代码：ActiveDocument.**PrintOut** Range:=wpsPrintCurrentPage

也可以用对应的数值来同时支持 WORD 和 WPS，如下：

```
ActiveDocument.PrintOut Range:=2 ' 注释枚举常量 wdPrintCurrentPage = 2
```

1.1.3 commandbars 调用的区别

commandbars 接口 API 在微软 WORD 的二次开发中可以这样引用：
application.commandbars 也可以这样引用:document.commandbars。

commandbars 接口 API 在 WPS 文字的二次开发中只能这样引用：
application.commandbars,对于微软的 document.commandbars 引用，可以替换成
document.application.commandbars

1.2 WPS Office2012 的接口类对象和对应的 DLL 文件

对象	接口	对应文件（wps安装目录）
WPS 文字	WPS	\office6\wpscore.dll
WPS 表格	ET	\office6\applogic.dll
WPS 演示	WPP	\office6\wppcore.dll
公用层对象	KSO	\office6\kso.dll
add-in 机制	IDTExtensibility2	\office6\ksaddndr.dll

1.3 原有的与 WPS Office 2003 (WPS2003)、WPS Office 2002 (WPS2002)、 WPS Office2005 (WPS2005)、WPS Office 2007 (2007)、WPS Office 2009 (2009) 结合的第三方程序，可以直接和 WPS Office2012 结合运行 吗？

WPS2012 是与 WPS2005、WPS2007、WPS2009 完全兼容的版本，与 WPS2003、
WPS2002 不完全兼容。

WPS2003、WPS2002 的内核 “Kingsoft WPS 1.0 Object Library”、“Kingsoft ET 1.0
Object Library”、“Kingsoft WPP 2003 Object Library”

WPS Office2005 的内核 “Kingsoft WPS 2.0 Object Library”、“Kingsoft ET 2.0
Object Library”、“Kingsoft WPP 1.0 Object Library”

WPS Office2012、WPS Office2009、WPS Office2007 的内核 “Kingsoft WPS 2.0
Object Library”、“Kingsoft ET 2.0 Object Library”、“Kingsoft WPP 2.0 Object Library”

1.4 原有的与 WPS Office 2003 或者 WPS Office 2002 结合的第三方程序，需 要做什么改动才能正常和 WPS Office2012 结合运行？

使用 VB 编写的程序，需要将引用库改为升级后的类型库。并将二者接口有差别
的部分进行修改，然后重新编译。

使用 VC 编写的并用到 CLSID 的程序，需要重写这些 CLSID。并将二者接口有差
别的部分进行修改，然后重新编译。

1.5 调用 WPS 接口 API 方法时要注意 Variant 参数类型的传递。

WPS 接口中的 Variant 参数类型，在使用时比较严格。建议无论使用何种语言，在传递 Variant 参数类型时，都要将其变量转换为 Variant 再进行传递。

VC 调用 ET 时，传递 Variant 参数的代码片断：

```
COleVariant varFileName("C:\\test.xls");
VARIANT varFALSE = {0};
varFALSE.vt = VT_BOOL;
//错误:调用时会出错
//varFALSE.boolVal = false;
//正确:正常调用
varFALSE.boolVal = VARIANT_FALSE;
VARIANT varEMPTY = {0};

kset::_Workbook.wb(wbs.Open(varFileName, varEMPTY, varFALSE, varNull,
varNull, varNull, varNull, varNull, varNull, varNull, varNull, varNull, varNull));
```

1.6 通过注册表修改宏代码的安全级别。

通过程序打开一些有宏代码或控件的文档，如果客户端的安全设置太高就会导致宏代码或控件无法加载，给应用带来很大麻烦。可以通过通过注册表来修改客户端的宏安全设置。

通过 VBS 脚本实现的代码如下：

```
Dim MyShell
Set Myshell = WScript.CreateObject("WScript.Shell") '创建 WScript 对象
Dim Level
Level = InputBox("请输入宏的安全级别，1 为低,2 为中,3 为高，4 为非常高","设置宏的安全级别",2)
If Level <> "" Then
    Myshell.RegWrite
"HKEY_CURRENT_USER\Software\Kingsoft\Office\6.0\wpp\Application
Settings\VbaSecurityLevel",Level '修改 WPS 演示模块的宏安全级别
    Myshell.RegWrite
"HKEY_CURRENT_USER\Software\Kingsoft\Office\6.0\wps\Application
Settings\VbaSecurityLevel",Level '修改 WPS 文字模块的宏安全级别
    Myshell.RegWrite
"HKEY_CURRENT_USER\Software\Kingsoft\Office\6.0\et\Application
Settings\KDESecurityLevel",Level '修改 WPS 表格模块的宏安全级别
    MsgBox "Done!"
End If
```

2 二次开发文字部分 FAQ

提示:

- 各对象、方法和属性的详细使用方法, 请参阅 **WPSAPI** 帮助。
- 对界面控件的操作都是在经典界面下完成的。由于 **WPS 2012** 界面使用 **Qt** 重构了界面代码, 所以一些涉及到界面控制的 **API** 暂时不能在 **WPS 2012** 界面中使用。
- 以下示例代码都用 **VB** 编写, 使用其它语言的开发者, 请自行做相应转换。

2.1 如何启动 **WPS** 文字模块?

如果 WPS 文字模块已经启动, 可以使用 `GetObject` 获得这个进程:

```
Set wpsApp = GetObject("WPS.Application")  
wpsApp.Visible = True
```

如果 WPS 尚未启动, 可以使用 `CreateObject` 来产生一个新进程:

```
Set wpsApp = CreateObject("WPS.Application")  
wpsApp.Visible = True
```

2.2 如何退出 **WPS** 文字模块?

```
wpsApp.Quit SaveChanges:=wpsSaveChanges, OriginalFormat:=wpsWPSDocument,  
RouteDocument:=True  
set wpsApp = Nothing
```

2.3 如何得到 **WPS** 中的工具栏数?

下面的例子显示应用程序所有工具栏的数目。

```
MsgBox wpsApp.Application.CommandBars.Count
```

下面的例子使工具栏的第一项不可见

```
wpsApp.CommandBars.Item(1).Visible = False
```

2.4 如何隐藏工具栏?

用 `CommandBars(Index)` 可以直接返回一个 `CommandBar` 对象。

使用下面的语句可以将第一个工具栏隐藏

```
Application.CommandBars.Item(1).Visible = false
```

2.5 如何使工具栏的按钮失效？

下面的例子切换第一条工具栏第一项按钮的有效性。

```
Dim bar As WPS.CommandBarControls
Set bar = wpsApp.CommandBars(1).Controls
bar(1).Enabled = Not bar(1).Enabled
```

2.6 如何获得一个文档？

下例通过索引号获取第一个文档。

```
Set wpsDoc = wpsApp.Documents(1)
```

2.7 如何打开 wps 文档？

在获得 WPS application 对象之后，就能对 WPS 进行控制，下面的例子中将以只读方式打开一个 WPS 文档，并激活此文档，如果文档不存在，将出示提示信息。

```
wpsApp.Documents.Open FileName:="E:\散文.wps", ReadOnly:=true
wpsApp.activatedocument. Activate
```

2.8 如何关闭一个文档？

关闭第一篇文档。

```
wpsApp.Documents(1).Close
```

2.9 如何保存一个文档？

如果活动文档在上次保存后进行了修改，下面的例子将保存活动文档。

```
If ActiveDocument.Saved = False Then
ActiveDocument.Save
End If
```

2.10 如何获得或修改当前用户名？

下面的例子设置应用程序的用户姓名(对应于菜单条"工具"-">"选项"-">"用户"-">"姓名"):

```
wpsApp.UserName = "wps user"
```

下面的例子显示应用程序的用户名

```
MsgBox wpsApp.UserName
```

2.11 如何在当前文档中添加一个表格？

下面的例子在文章开头添加一 4 行 4 列的表格，并在最后一个单元格里写上"需要写些什么" 字样

```
Dim rngTbl As WPS.Range
Dim wpsTable As WPS.Table
Set r = ActiveDocument.Range(0, 0)
Set wpsTable = ActiveDocument.Tables.Add(r, 4, 4)
wpsTable.cell(4, 4).Range.Text = "需要写些什么"
```

2.12 如何获取表格的行和列？

下面的例子将获得第一个表格的行和列

```
Dim mytable As WPS.Table
Set mytable = wpsApp.ActiveDocument.Tables(1)
MsgBox "第一个表有" & mytable.Columns.Count & "列" & mytable.Rows.Count & "行"
```

2.13 如何选中一个表？

下面的例子将当前活动文档的第一个表选中。

```
ActiveDocument.Tables(1).Select
```

2.14 如何修改一个表格的某个单元格的内容？

下面的例子对活动文档第一个表格的第一个单元格的内容进行显示及修改。

```
Dim mytable As WPS.Table
Set mytable = ActiveDocument.Tables(1)
MsgBox "原来 (1,1)单元格的内容是：" & mytable.cell(1, 1).range.text '缺省表示指定单元格内容
mytable.cell(1, 1).range.text = "金山文字"
MsgBox "修改后是：" & mytable.cell(1, 1).range.text
mytable.cell(1, 1).Range.Text = "WPS Office"
MsgBox "再次修改后是" & mytable.cell(1, 1).Range.Text
```

2.15 .如何设置一个表格的单元格的高度和宽度？

下面的例子将活动文档第一张表格的所有单元格的高度和宽度设置都设为 200*0.1mm。

```
Dim mytable As WPS.Table
```



```

Set mytable = wpsApp.ActiveDocument.Tables(1)
iCols = mytable.Columns.Count
iRows = mytable.Rows.Count
For i = 1 To iRows
mytable.cell(i, 1).Height = 200
Next
For j = 1 To iCols
mytable.cell(1, j).Width = 200
Next

```

下面的例子将活动文档的第一个表格的所有单元格的宽度设为 400*0.1mm。

```

Dim mytable As WPS.Table
Set mytable = ActiveDocument.Tables(1)
num = mytable.Columns.Count
For j = 1 To num
mytable.cell(1, j).Width = 400
Next j

```

2.16 表格增加一行之后，要获取增加表格的内容，必须重新给表格变量赋值

```

‘Tb 是一个 4 行 4 列的表
Dim Tb As Table
Set Tb = ActiveDocument.Tables.Item(1)

‘Tb 表格添加一行
Tb.Rows.Add

‘重新给表格变量赋值
Set Tb = ActiveDocument.Tables.Item(1)

‘再对新添加的行进行操作，否则会出错
Tb.Cell(5, 4).Range.Text = "5,4"

```

2.17 如何添加一个书签？

下面的例子将在当前文档中添加书签“我”，书签的内容为从第三个字符到第五个字符。

```

Dim r As WPS.Range
Set r = wpsApp.ActiveDocument.Range
r.SetRange Start:=3, End:=5
wpsApp.ActiveDocument.Bookmarks.Add Name:="我", Range:=r

```

2.18 如何替换一个书签的内容？

下面的例子判断名为"temp"的书签是否存在于活动文档中，如果存在则替换该书签的内容。

```
If ActiveDocument.Bookmarks.Exists("temp") = True Then
ActiveDocument.Bookmarks("temp").range.text="ReplaceText"
End If
```

2.19 如何获取和设置默认打印机？

本示例获取默认打印机的名字。

```
MsgBox "默认的打印机为:" & Application.ActivePrinter
```

本示例将 LPT1 端口的本地 HP LaserJet 6 打印机设置为活动打印机。

```
Application.ActivePrinter = "HP LaserJet 6 local on LPT1:"
```

2.20 如何打印一篇已打开的文档？

本示例以默认设置打印活动文档的所有页。

```
Application.ActiveDocument.PrintOut
```

本示例打印了活动文档中的 1 到 3 页。

```
Application.ActiveDocument.PrintOut From:="1", To:="3"
```

本示例按照活动文档的 75%大小进行打印。

```
Application.ActiveDocument.PrintOut PrintZoomPaperWidth:=0.75 * _
(8.5 * 1440), PrintZoomPaperHeight:=0.75 * (11 * 1440)
```

2.21 如何获取和设置某些文字的字体属性？

本示例获取和设置选中部分的字体名称和大小。

```
With Application.Selection.Range.Font
MsgBox "字体名称:" & .Name & vbCrLf & "字号:" & .Size
MsgBox "即将改变字体名称和字号!"
.Name = "华文行楷"
.Size = 20
MsgBox "改变后的内容如下:" & vbCrLf & "字体名称:" & .Name _
& vbCrLf & "字号:" & .Size
End With
```

2.22 如何获取和设置段落的格式？

本示例获取和设置活动文档首段左右各缩进 1 英寸。

```
With Application.ActiveDocument.Paragraphs(1)
    MsgBox “ 左 侧 缩 进 :” & .LeftIndent & VbCrLf & “ 右 侧 缩 进 :” & .RightIndent
    .LeftIndent = InchesToPoints(1)
    .RightIndent = InchesToPoints(1)
End With
```

本示例设置活动文档首段的行距为至少 12 磅。

```
With Application.ActiveDocument.Paragraphs(1)
    .LineSpacingRule = wpsLineSpaceAtLeast
    .LineSpacing = 12
End With
```

2.23 如何通过注册表获取 WPS 安装版本及安装路径？

在注册表中获取 WPS 的具体版本号

通过读取注册表

HKEY_LOCAL_MACHINE\SOFTWARE\Kingsoft\Office\6.0\Common\Version 的值便可获取 wps 的具体版本号。如：6.4.0.1855

在注册表中获取 WPS 程序的路径。

WPS 三大组件的程序都在同一目录，通过读取注册表

HKEY_LOCAL_MACHINE\SOFTWARE\Kingsoft\Office\6.0\common\InstallRoot 的值便可获取 wps 的安装路径。

VC 代码如下：

```
BOOL GetWPSRoot(LPSTR lpstrReturn)
{
    BOOL bRet = TRUE;
    HKEY hKey = 0;
    DWORD dwLength = 256;
    DWORD dwType = REG_SZ;
    if (RegOpenKeyEx(HKEY_LOCAL_MACHINE, "SOFTWARE\\Kingsoft\\Office\\6.0\\common", 0, KEY_READ, &hKey) == ERROR_SUCCESS)
    {
        if (RegQueryValueEx(hKey, "InstallRoot", NULL, &dwType, (unsigned char *)lpstrReturn, &dwLength) != ERROR_SUCCESS)
            bRet = FALSE;

        RegCloseKey(hKey);
    }
    else
        bRet = FALSE;
}
```

```

        return bRet;
    }
    //可以这样调用:
    char szRoot[256];
    GetWPSRoot(szRoot);
    AfxMessageBox(szRoot);

```

2.24 如何在插入对象里添加 **WPS** 文档对象？

增加注册表建值，如下：

```

Windows Registry Editor Version 5.00
[HKEY_CLASSES_ROOT\WPS.Document.6\Insertable]

```

2.25 在 **WPS** 文字中使用 **Find** 的用法

‘WPS 文字中 Document 对象中包含 Find 方法，需要将 Document 选中后再对 Selection 进行 Find 操作。

代码如下：

```

ActiveDocument.Content.Select
Selection.Find .....

```

2.26 去掉 **Find** 替换后出现的对话框

使用下面代码中 Execute 方法中的 Wrap 参数

wpsFindAsk ‘默认参数，表示 Find 后提示是否对其它区域进行 Find。

wpsFindContinue’表示 Find 后不提示，并继续对其它区域进行 Find。

wpsFindStop’表示 Find 后不提示，并不对其它区域进行 Find 操作。

代码示例：

```

With Selection.Find
    .ClearFormatting
    .Text = "[aaaa]"
    .Replacement.Text = "[ssss]"
    .Execute 'Replace:=wpsReplaceAll , Wrap:=wpsFindStop
End With

```

2.27 如何通过 **MIME** 设置实现 **WPS** 文字嵌入浏览器的方式打开服务器 **DOC** 或 **WPS** 文档

在服务器配置文件 web.xml 中增加下面的代码

```

<mime-mapping>

```

```

        <extension>wps</extension>
        <mime-type>application/kswps</mime-type>
    </mime-mapping>
    <mime-mapping>
        <extension>doc</extension>
        <mime-type>application/kswps</mime-type>
    </mime-mapping>

```

2.28 如何在不破坏书签的情况下设置内容？

```

Dim wpsRng As WPS.Range
Set wpsRng = ActiveDocument.Bookmarks.Item("bookmark1").Range
wpsRng.Text = "NewValue"
ActiveDocument.Bookmarks.Add "bookmark1", wpsRng

```

2.29 为什么在使用 **Range.Text** 为文档、书签，或用 **FormField.Result** 为窗体域设置文本内容时，回车符不见了？

请使用 Chr(13) & Chr(10) 作为回车换行符，尽量不要将 Chr(13) 或 Chr(10) 单独使用。

```

ActiveDocument.Content.Text = "a" & Chr(13) & Chr(10) & "b"
ActiveDocument.Bookmarks.Item(1).Range.Text = "a" & Chr(13) & Chr(10) & "b"
ActiveDocument.FormFields.Item(1).Result = "a" & Chr(13) & Chr(10) & "b"

```

2.30 如何判断文档的设计模式状态，如何退出设计模式？

在 WPS 文字中，需要通过判断“退出设计模式”按钮的状态来得知当前文档中嵌入的控件是否是设计模式，代码如下：

```

Dim btn As CommandBarButton '只有 CommandBarButton 才有 state 属性
Set btn = CommandBars("control tool box").Controls("退出设计模式")
If btn.State = -1 Then ' State = -1 是设计模式状态
    btn.State = 0
End If

```

2.31 如何隐藏表格的智能扩展标记？

WPS 的新版本增加了智能扩展标记，编辑表格会变得非常方便。但是如果 OA 系统里面的模板是通过隐藏表格来定制模板的，就会出现“表格隐藏了，标记还在”的现象，影响视觉效果为此 WPS 提供了 API 隐藏此标记，用法如下：

```

Application.Options.ShowTableAdjustLabel = False ' 隐藏表格智能调节标识按钮

```

2.32 如何通过 API 实现不显示表格虚线？

WPS 默认情况下是现实表格虚线的，如果 OA 系统里面的模板是通过隐藏表格来定制模板的，就会带来一些视觉影响，WPS 提供了 API 控制表格虚线的现实，用法如下：

```
Application.ActiveWindow.View.TableGridlines = False' True 为现实，False 不显示.
```

3 二次开发电子表格部分 FAQ

3.1 如何启动 WPS 电子表格模块？

如果 WPS 已经启动，可以使用 GetObject 获得这个进程：

```
Set etApp = GetObject(, "ET.Application")  
etApp.Visible = True
```

如果 WPS 电子表格模块尚未启动，可以使用 CreateObject 来产生一个新进程：

```
Set etApp = CreateObject("ET.Application")  
etApp.Visible = True
```

3.2 如何退出电子表格？

调用 Application 的 Quit 方法可以退出电子表格模块，
下面的例子先把所有打开的表格文档存盘，然后退出。

```
Dim objWB As ET.workbook  
For Each objWB In etApp.Workbooks  
    objWB.Save  
Next objWB  
etApp.Quit  
Set etApp = Nothing
```

3.3 如何打开一个电子表格文档？

下面的示例以只读方式打开了一个电子表格文档：

```
Application.Workbooks.Open FileName:="E:\Test.et", ReadOnly:=True
```

下面的示例打开了一个有密码的电子表格文档：

```
Application.Workbooks.Open FileName:="E:\Test.et", password:="et"
```

3.4 如何关闭一个电子表格文档？

本示例关闭了第一个工作簿，并放弃所有对该工作簿的更改。

```
Application.Workbooks(1).Close SaveChanges:=False
```

本示例关闭了所有工作簿，如果某个打开的工作簿有改变，将会询问是否保存所作的更改。

```
Application.Workbooks.Close
```

3.5 如何保存一个电子表格文档？

本示例演示保存所有已打开的工作簿，并退出金山表格。

```
For Each w In Application.Workbooks  
    w.Save  
Next w  
Application.Quit
```

本示例演示了将当前工作簿另存到其它位置。

```
Application.ActiveWorkbook.SaveAs "F:\temp.et"
```

3.6 如何设置一个单元格的大小？

本例演示了设置(2,2)单元格的所在行的高度为 20，所在列宽度为 100。

```
With Application.ActiveWorkbook.ActiveSheet  
    .Range("B2").RowHeight = 20  
    .Range("B2").ColumnWidth = 100  
End With
```

本例演示了设置当前工作表的所有行高度为 20，所有列宽度为 100。

```
With Application.ActiveWorkbook.ActiveSheet  
    .StandardHeight=20  
    .StandardWidth=100  
End With
```

3.7 如何获取单元格中的内容？

本示例获取了当前工作表中第(2,3)单元格中的内容。

```
MsgBox Application.ActiveWorkbook.ActiveSheet.Cells.Item(2,3).Formula
```

3.8 如何设置单元格中的内容？

本示例设置了当前工作表中第(2,3)单元格中的内容。

```
Application.ActiveWorkbook.ActiveSheet.Cells.Item(2, 3).Formula = "temp"
```

3.9 如何设置单元格中的公式？

本示例设置了当前工作表中第(2,3)单元格中的公式内容，该单元格将显示计算结果:2.5。

```
Application.ActiveWorkbook.ActiveSheet.Cells.Item(2, 3).Formula = "=2+3/6"
```

本例演示了统计从(2,1)到(3,4)这一区域的所有单元格内数字的平均值。

```
Application.ActiveWorkbook.ActiveSheet.Cells.Item(2, 3).Formula = "=AVERAGE(B1, C4) "
```

3.10 如何设置单元格的边框？

本例演示了将(2,2)单元格的边框粗细设置为 etMedium，线条样式设置为 etContinuous,颜色设置为 vbRed。

```
With Application.ActiveWorkbook.ActiveSheet
    .Range("b2").Borders.Weight = etMedium
    .Range("b2").Borders.LineStyle = etContinuous
    .Range("b2").Borders.Color = vbRed
End With
```

3.11 如何读取批注的内容

WPS 的电子表格模块用 Comment 对象代表一个批注。以下的示例，用于读取当前表单的第三行第四列单元格（编号为“D3”）中的批注信息。

```
Dim objCM As ET.Comment
Dim objRange As ET.Range
Dim objSheet As ET.Worksheet

Set objSheet = ETapp.ActiveSheet
Set objRange = objSheet.Cells.Item(3, 4)

Set objCM = objRange.Comment

If objCM Is Nothing Then
    MsgBox "未发现批注内容"
Else
    MsgBox objCM.Text
End If
```

3.12 如何添加批注？

WPS 的电子表格模块用 Comment 对象代表一个批注，通过 Range 的 AddComment 方

法可以添加批注。

下面的示例为当前表单的第三行第四列单元格（编号为“D3”），添加一个批注。

```
Dim objCM As ET.Comments
Dim objRange As ET.Range
Dim objSheet As ET.Worksheet

Set objSheet = ETapp.ActiveSheet
Set objRange = objSheet.Cells.Item(3, 4)
objRange.AddComment "这是一个批注"
```

3.13 如何设置单元格内字体的属性？

通过 Font 对象可以设置和读取字体的各种属性。

下面的例子设置当前单元格的字体设为粗体 45 号字

```
ETapp.ActiveCell.Font.Bold = True
ETapp.ActiveCell.Font.Size = 45
```

下面的例子把第 3 行，第 4 列的单元格中的文字，修改为“字体设置试验”。字体设为黑体，斜体，红色并加下滑线。

```
Dim objRange As ET.Range
Dim objSheet As ET.Worksheet
Set objSheet = ETapp.ActiveSheet
Set objRange = objSheet.Cells.Item(3, 4)

objRange.Font.Name = "黑体"
objRange.Font.Italic = True
objRange.Font.Underline = etUnderlineStyleDouble
objRange.Font.Color = 255
objRange = "字体设置试验"
```

3.14 如何打印电子表格文档？

通过 PrintOut 方法可以实现打印电子表格文档

以下示例实现打印，当前文档的第 1 到第 3 页。

```
Dim wb As ET.workbook
Set ETapp = GetObject("ET.Application")
Set wb = ETapp.Workbooks(1)
wb.PrintOut 1, 3, , False, ""
```

3.15 如何在插入对象里添加 WPS 表格对象？

如果在一些支持 OLE 嵌入的控件或产品中找不到 WPS 表格对象，可以通过增加注册表建值的方式来增加，注册表文件内容如下：

```
Windows Registry Editor Version 5.00
```

```
[HKEY_CLASSES_ROOT\ET.Workbook.6\Insertable]
```

3.16 WPS 表格中如何在默认位置插入图片

调用 Worksheet 的 Shapes.AddPicture 方法，用来向 ET 表格插入图片，可以通过，其设置其 left、top 参数中都为-1 时，图片会插入到 ET 的光标位置。

```
Dim etWs As Worksheet
Set etWs = Application.ActiveSheet
etWs.Shapes.AddPicture "c:\wps.jpg", ksoTrue, ksoTrue, -1, -1
```

3.17 WPS 表格中如何插入图表

用 VBA 代码插入图表：

```
Dim chr As chart
Dim chrobj As ChartObject
'添加图表
Set chrobj = Sheet1.ChartObjects.Add(4000, 3000, 8000, 6000) '注意度量单位
'显示图表
chrobj.Visible = True
'设置图表数据
Set chr = chrobj.chart
chr.ChartType = etColumnClustered
chr.SetSourceData Source:=Sheets("Sheet1").Range("C6:F8")
```

3.18 通过 MIME 设置 WPS 表格嵌入浏览器方式打开服务器的 et 或 xls 文档

在服务器配置文件 web.xml 中增加下面的代码

```
<mime-mapping>
  <extension>et</extension>
  <mime-type>application/kset</mime-type>
</mime-mapping>

<mime-mapping>
  <extension>xls</extension>
  <mime-type>application/kset</mime-type>
</mime-mapping>
```