

Python保留格式复制多个excel工作表到汇总表并生成目录(实例69)

工作中，经常需要将很多独立的excel表格按照顺序复制汇总到单个表格，并且要求保留原表格的格式，且要生成目录页，如下图酱紫的。使用excel自带的“移动或复制”可以实现，但是纯手工操作，需要一个一个打开并复制到新的表格。目录也要手动“复制粘贴”。可不可以用Python一键搞定呢？

多个工作表迁移到一个新工作表，且保留原格式，并生成目录页

名称	修改日期
测试1-ID信息.xlsx	2020/7/10 9:38
测试2-工程部领料明细.xls	2020/7/10 13:58
测试3-生产订单.xlsx	2020/7/2 16:54

	A	B	C	D	E	F	G
1	测试1-ID信息						
2	测试2-工程部领料明细						
3	测试3-生产订单						
4							

就绪

滚动

Python操作Office软件高效工作

```
首先，
xlrd
，
openpyxl
什么的基本考虑放弃。因为这些无法实现整个excel工作表带格式复制。本文使用
win32com
模块调用
Excel.Application
```

来操作。先锁定需要处理的文件。假设这些文件存在一个名叫“文件”的文件夹内，然后只读取excel文件，包括后缀为“xls”和“xlsx”的。将它们存入列表

```
files
```

。

```
import os #用于获取目标文件所在路径
path=os.getcwd()+"\\文件\\" # 文件夹绝对路径
files=[]
for file in os.listdir(path):
    if file.endswith(".xls") or file.endswith(".xlsx"): #只获取excel文件
        files.append(path+file)
files
>>
['C:\\Users\\python\\python_excel\\实例69_Python保留格式整体复制Excel工作表到汇总表\\文件\\测试1-ID信息.xlsx',
 'C:\\Users\\python\\python_excel\\实例69_Python保留格式整体复制Excel工作表到汇总表\\文件\\测试2-工程部领料明细.xls',
 'C:\\Users\\python\\python_excel\\实例69_Python保留格式整体复制Excel工作表到汇总表\\文件\\测试3-生产订单.xlsx']
```

然后就可以将子excel文件逐个复制到新的汇总表里了。

首先，导入相关库。然后调用excel应用，并且设定操作过程不显示。使用 `Workbooks.Add()` 新建一个excel工作簿，然后将其另存为 `汇总.xlsx` 文件。使用 `Worksheets(1)` 获取第一个表，并将其名称设置为“目录”。

汇总文件准备好后，就可以逐个打开子excel表，将内容复制过来了。使用老朋友 `for` 循环来遍历。从文件路径中提取出文件名，存入 `file_name` 以便后续使用。然后将子excel文件名写入目录表的第一列内。使用 `Workbooks.Open` 打开子文件，获取其活动工作表，存入 `ws_sub` （因为此处用于测试的子文件只有一个工作表，如果有多个，则要保证需要的表是活动的。在保存excel文件之前，先选中需要的那个表，那么那个表就是“活动”的了）。

然后使用 `Copy` 方法，将子excel工作表复制到汇总表。 `Copy(ws)` 里面的参数 `ws` 就来自于汇总表 `wb`，程序会将子工作表自动复制到汇总表的第一个工作表前面。复制完后，将文件名设定为工作表名，方便区分。依次这样操作，直到将所有子文件复制完毕。因为复制表的时候是默认从最前面开始摆放，所以“目录”工作表就被挤到最后一个了。我们希望它在最前面，所以使用 `Move` 方法将它移动到了第一个工作表的前面，而第一个表就是“测试1-ID信息”。

```
import os
import win32com

excel_app = win32com.client.Dispatch("Excel.Application") #调用Excel应用
excel_app.Visible = False # 不显示Excel文件
excel_app.DisplayAlerts = False #不显示警告信息

wb = excel_app.Workbooks.Add() #新建excel工作簿
wb.SaveAs(os.getcwd() + "\\汇总.xlsx") #保存新建的工作簿
ws = wb.Worksheets(1) #获取第一个表
ws.Name = "目录" #设置第一个表名称
for i in range(len(files)):
    file_name = files[i].split("\\")[-1].split(".")[0] #获取子excel文件的文件名
    ws.Range("A"+str(i+1)).Value = file_name #将文件名写入目录
    #读取子文件
    wb_sub = excel_app.Workbooks.Open(files[i]) #打开子excel文件
    ws_sub = wb_sub.ActiveSheet # 获取活动工作表

    ws_sub.Copy(ws) #复制工作表到汇总表
    wb.ActiveSheet.Name = file_name #更改工作表名为文件名
    wb_sub.Close() #关闭工作簿
    print(f"已复制文件 {file_name}")

#将“目录”工作表移动到最前面
first_sheet = files[0].split("\\")[-1].split(".")[0]
wb.Worksheets("目录").Move(wb.Worksheets(first_sheet)) #移动目录工作表到最前面

wb.Save() #保存
wb.Close() #关闭
excel_app.Quit() #退出应用
>>
已复制文件 测试1-ID信息
已复制文件 测试2-工程部领料明细
已复制文件 测试3-生产订单
```

这样就搞定了。原表的格式一点都没有变化，非常满意，上图看真相



。

	A	B	C	D	E	F
1	部门	产品编号	订单	文件	数量	日期
1055	GC	Q3SN161CXE-XEXE2	O029953	DOC-20-575	200	2020-6-22
1056	GC	Q3SN161CXE-XEXE2	O030001	DOC-SG-20200623-00	180	2020-6-23
1057	GC	59XE8-0665XE0	O029980	DOC-SG-1910001	92	2020-6-23
1058	MJ	Q4TS0608XE-XEXE1	O030002	DOC-20-608	35	2020-6-24
1059	GC	Q4TS0608XE-XEXE1	O030003	DOC-20-609	35	2020-6-24
1060	GC	Q2D-049-XE-XE020	O030004	DOC-20-612	12	2020-6-24
1061	SG	Q111715-01-XEXE6	O031694	DOC-20-606	26	2020-6-29
1062	GC	Q111745-01-XEXE3	O031695	DOC-20-607	26	2020-6-29
1063	GC	Q2D-057-XE-XE010	O030894	DOC-20-610	70	2020-6-29
1064	GC	Q3SN164-XE-XEXE1	O031472	DOC-20-590	368	2020-6-29
1065	GC	Q3SN164-XE-XEXE1	O031472	DOC-20-590	368	2020-6-29

Python操作Office软件高效工作

如果您有需要处理的问题，可发邮件到邮箱：

donyo@qq.com，一起探讨解决方案。

微信公众号输入“源文件”提取所有源文件及资料。

PythonOffice

专注于办公自动化，使用Python语言操作Excel、Word、PPT、PDF等Office软件，代替日常工作中的重复机械操作，千倍提升工作效率。



Python操作Office软件高效工作