#TXT文本文档的读写

def mergeTxt(txtFiles):#txtFiles是个文件名的列表

with open(‘c:\\result.txt', 'w') as fp:

with open(txtFiles[0]) as fp1, open(txtFiles[1]) as fp2:#open打开时默认是读文件‘r’

while True:

# 交替读取文件1和文件2中的行，写入结果文件

line1 = fp1.readline()

if line1:

fp.write(line1)

else: # 如果文件1先结束， 则flag = False，结束循环

flag = False

break # 这个break是跳出上面那个while True循环

line2 = fp2.readline()

if line2:

fp.write(line2)

else: # 读不到了，说明文件2结束，结束循环

flag = True #如果文件2先结束， 则flag = True，

break

# flag=false时，文件2长， fp3=fp2， flag=true时，文件1长， #fp3 = fp1 ，获取尚未结束的文件对象，总之就是获取#更长的那个文件

fp3 = fp1 if flag else fp2

for line in fp3: # 把剩余内容写入结果文件，注意这里是剩余内容，是接着刚才读的位置继续往后读，不是从开头重新开始读

fp.write(line)

txtFiles = [‘c:\\c1.txt’, ‘c:\\c2.txt’]#你可以自己建立两个txt文档

mergeTxt(txtFiles)

#json文件读写

import json

#一个列表，里面包含很多个字典

information = [{'小区名称': '小区A', '均价': 8000, '月交易量': 20},

{'小区名称': '小区B', '均价': 8500, '月交易量': 35},

{'小区名称': '小区C', '均价': 7800, '月交易量': 50},

{'小区名称': '小区D', '均价': 12000, '月交易量': 18}]

with open('房屋信息.json', 'w') as fp:#如果这个文件不存在，则创建它

json.dump(information, fp, indent=4, separators=[',', ':'])#用json.dump把要写入的数据#information给序列化，写入到fp这个文件， indent=4缩进4个空格，元素与元素之间分隔符用逗号‘,’，字典里键

#和值之间用冒号‘:’，房屋信息.json文件，里面汉字都是Unicode编码

with open('房屋信息.json') as fp:

information = json.load(fp)#这是读取，load跟dump两者是对应的

for info in information:#用for循环去遍历这个列表里的每一个元素，然后输出它

print(info)

#office文件的读入

from sys import argv #要用到命令行参数，所以导入这个

from os import listdir

from os.path import join, isfile, isdir#用来判断是否为一个文件，还是一个文件夹

from docx import Document #word文档操作需要先安装好扩展库python-docx，然后才能导入

from openpyxl import load\_workbook#EXCEL表格操作需要事先安装好扩展库openpyxl

from pptx import Presentation#PPT演示文稿操作需要事先安装好python-pptx扩展库

#检查word文档中是否包含一个字符串

def checkdocx(dstStr, fn):#这两个参数，前面是你要搜索的字符串，后面那个是文件路径

# 打开docx文档

document = Document(fn)

# 遍历所有段落文本，每一个文本段落都拿出来

for p in document.paragraphs:

if dstStr in p.text:

return True

# 遍历所有表格中的单元格文本，每一张表格都拿出来

for table in document.tables:

for row in table.rows:#遍历表格每一行

for cell in row.cells:#遍历一行内的所有cell

if dstStr in cell.text:

return True

return False

#检查EXCEL文档中是否包含一个字符串

def checkxlsx(dstStr, fn):

# 打开xlsx文件

wb = load\_workbook(fn)

# 遍历所有工作表的单元格

for ws in wb.worksheets:

for row in ws.rows:

for cell in row:

try:

if dstStr in cell.value:

return True

except:

pass

return False

def checkpptx(dstStr, fn):

presentation = Presentation(fn) # 打开pptx文档

# 遍历所有幻灯片

for slide in presentation.slides:

for shape in slide.shapes: # 遍历这个幻灯片里的所有形状，文本框，表格，图片都叫一个形状

if shape.shape\_type == 19: # 如果是19就表示是一个表格，

for row in shape.table.rows:# 遍历这个表格里所有的行

for cell in row.cells:# 遍历这个行里所有的单元格

if dstStr in cell.text\_frame.text:# 把单元格里的文本取出来看看它包含目标字符串吗

return True

elif shape.shape\_type == 14: # 如果是14就表示是一个文本框

try:

if dstStr in shape.text:#看看文本框里的文本是不是包含目标字符串

return True

except:

pass

return False

def main(dstStr, flag):

# 使用广度优先的方式遍历当前文件夹及其所有子文件夹

# 一个圆点表示当前文件夹

dirs = ['.']

while dirs:#当前文件夹路径非空

# 获取第一个尚未遍历的文件夹名称

currentDir = dirs.pop(0)#把列表中的第一项弹出来，这个列表记录的是我们尚未遍历的文件夹

for fn in listdir(currentDir):#把这个尚未遍历的文件夹目录下的所有文件和文件夹的清单通过一个循环逐个罗列出来进行遍历

path = join(currentDir, fn)#每一个拿来和它的父目录进行连接，如果连接起来是一个完整的文件路径

if isfile(path):#每一个拿来和它的父目录进行连接，如果连接起来的路径是一个文件

#看你的文件后缀名是啥，根据后缀名分别送给不同的函数，如果函数返回的结果是true，则表明这个文件中是包含dstStr的，我们就把这个path输出

if path.endswith(‘.docx’) and checkdocx(dstStr, path):

print(path)

elif path.endswith('.xlsx') and checkxlsx(dstStr, path):

print(path)

elif path.endswith('.pptx') and checkpptx(dstStr, path):

print(path)

# 广度优先遍历目录树，说明是个文件夹路径dir，不是一个文件file

elif flag and isdir(path):

dirs.append(path)#那就把这个文件夹路径dir追加到尚未遍历的文件夹名称列表里头，现在先不遍历，待会儿再遍历

main(dstStr, flag)