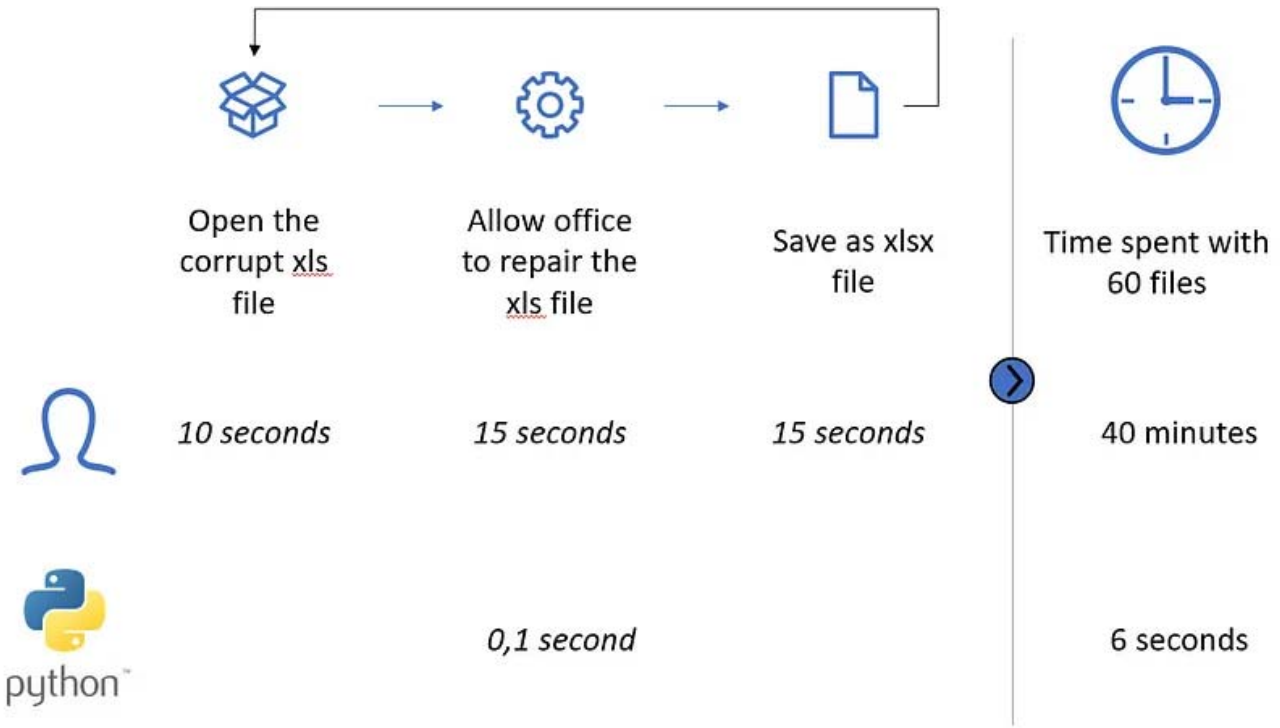
✦ Member-only story

Convert multiple corrupt XLS files to XLSX | Python in Finance #5

Edward Jones · Follow
Published in Geek Culture
2 min read · May 27, 2021

Listen Share

In this article you will learn how to convert a batch of corrupt XLS files to an XLSX file.



Trade-off of automating this process

In this article, you will learn how to automate the conversion process from (corrupt) xls files to normal xlsx files.

Data can be found: here

Packages

- pywin32: This package is basically vba for python. It allows us to interact and automate Windows applications with python.
- os: This package allows us to use the operating system.
- glob: This package allows us to create a list with the different file locations

```
import win32com.client
import os
import glob
```

Initialization

- You initialize the win32com and let it run an excel application.
- Set the object “o.Visible = False” in order to hide the excel application you have created.
- Set the input directory: This is the directory where all your corrupt xls files are located.
- Set the output directory: This is the directory where you want to store the converted xlsx files.
- Create a list of file paths of the all the files in the input directory by using the glob.glob function

```
o = win32com.client.Dispatch("Excel.Application")
o.Visible = False

input_dir = r"C:\input_directory"
output_dir = r"C:\output_directory"
files = glob.glob(input_dir + "/*.xls")
```

Loop

For each file, you perform the following manipulations:

- Assign the filename to the object “file”
- Set the ouput filename by concatenating the output directory and the output filename replacing the xls extension with the xlsx extension.
- Open the xls files by the excel application
- Save the xls file as an xlsx file using the output path as previously specified
- Close the file

```
for filename in files:
    file = os.path.basename(filename)
    output = output_dir + '/' + file.replace('.XLS','.xlsx')
    wb = o.Workbooks.Open(filename)
    wb.ActiveSheet.SaveAs(output,51)
    wb.Close(True)
```

Using these bits of code, you just quickly automated a task that would cost you normally 40 minutes.

Happy programming!

Cheers,

Ewoud

Python Automation Excel