

全部经验分类

Android (</lib/tag/Android>) iOS (</lib/tag/IOS>)

(</lib/list/all>) 

所有分类 (</lib/list/all>) > 开发语言与工具 (</lib/list/36>) > Python开发 (</lib/list/43>)

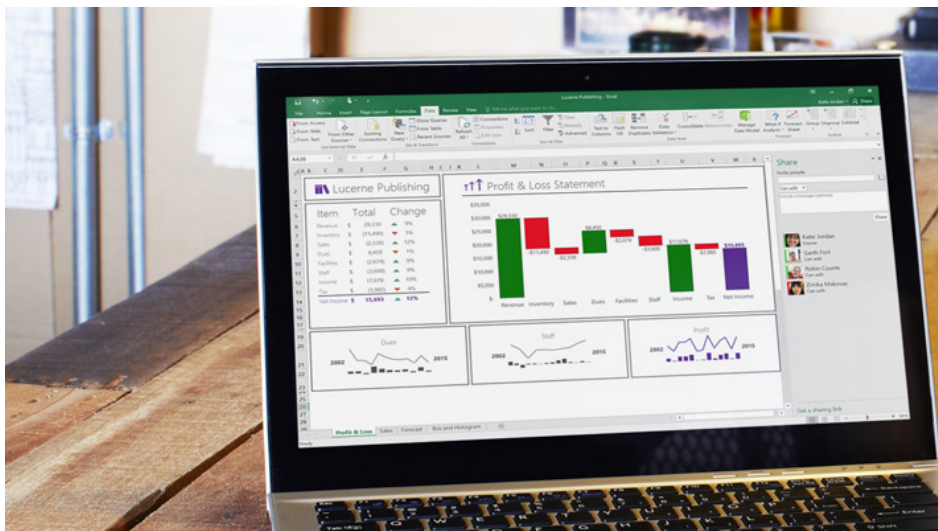
## Python 与 Excel 不得不说的

Python (</lib/tag/Python>) 2016-09-01 11:44:56 发布

您的评价: 5.0

收藏

2收藏



数据处理是 Python 的一大应用场景，而 Excel 则是最流行的数据处理软件。因此用 Python 进行数据相关的工作时，难免要和 Excel 打交道。

如果仅仅是要以表单形式保存数据，可以借助 CSV 格式（一种以逗号分隔的表格数据格式）进行处理，Excel 也支持此格式。但标准的 Excel 文件（xls/xlsx）具有较复杂的格式，并不方便像普通文本文件一样直接进行读写，需要借助第三方库来实现。

常用的库是 python-excel 系列：

**xlrd、xlwt、xlutils**

- xlrd - 读取 Excel 文件
- xlwt - 写入 Excel 文件
- xlutils - 操作 Excel 文件的实用工具，如复制、分割、筛选等

尽管这是目前被用得最多的 Excel 库，我还是很想吐槽为什么这三个包不能放在一个模块里.....另外它们有个缺陷，就是只能处理 xls 文件。如果你想使用新版本的 xlsx，可以考虑 openpyxl 和 xlsxwriter。

不过今天只说说这三个。

# 安装

安装的方法没啥特别的，只是得装三遍。可以下载安装包、下载代码压缩包、或者通过 pip 等。可参考 如何安装 Python 的第三方模块 - Crossin的编程教室 - 知乎专栏 (<https://zhuanlan.zhihu.com/p/20760860>)

如果安装过之前推荐的 anaconda，那么就已经有了 xlrd 和 xlwt，但 xlutils 没有附带在安装包中，使用时仍需另行安装。

# 读取

结合一段简单的代码来看：

```
import xlrd
# 打开 xls 文件
book = xlrd.open_workbook("test.xls")
print "表单数量:", book.nsheets
print "表单名称:", book.sheet_names()
# 获取第1个表单
sh = book.sheet_by_index(0)
print u"表单 %s 共 %d 行 %d 列" % (sh.name, sh.nrows, sh.ncols)
print "第二行第三列:", sh.cell_value(1, 2)
# 遍历所有表单
for s in book.sheets():
    for r in range(s.nrows):
        # 输出指定行
        print s.row(r)
```

测试文件：

	A	B	C
1	Rank	Team	Points
2		1 Brazil	15
3		2 Russia	12
4		3 South Korea	9
5		4 Japan	6
6		5 Argentina	5
7		6 Cameroon	1
8			
9			

◀ ▶

Group.A

Group.B

+

Crossin的编程教室

输出结果：

表单数量: 2

表单名称: [u'Group.A', u'Group.B']

表单 Group.A 共 7 行 3 列

第二行第三列: 15.0

[text:u'Rank', text:u'Team', text:u'Points']

[number:1.0, text:u'Brazil', number:15.0]

[number:2.0, text:u'Russia', number:12.0]

...

常用的方法：

- open\_workbook 打开文件
- sheet\_by\_index 获取某一个表单
- sheets 获取所有表单
- cell\_value 获取指定单元格的数据

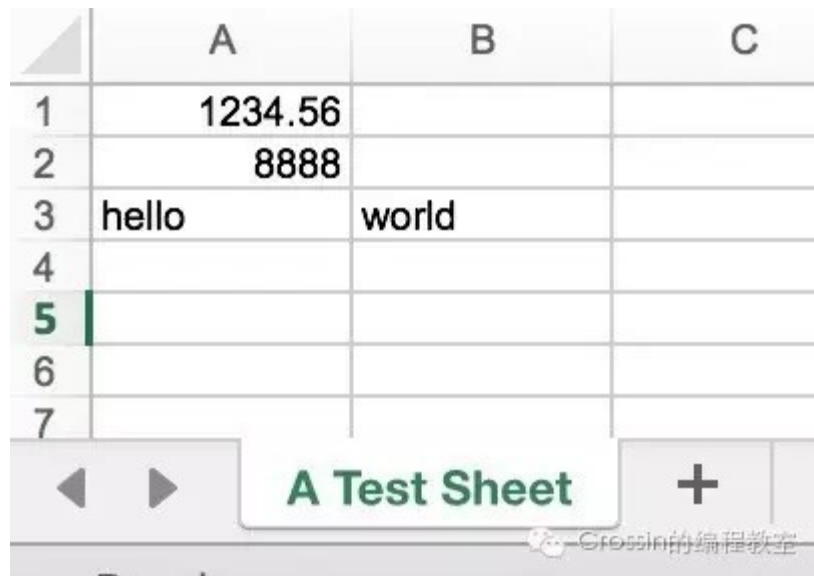
## 写入

还是看代码：

```
import xlwt
# 创建 xls 文件对象
wb = xlwt.Workbook()
# 新增一个表单
sh = wb.add_sheet('A Test Sheet')
# 按位置添加数据
sh.write(0, 0, 1234.56)
sh.write(1, 0, 8888)
sh.write(2, 0, 'hello')
sh.write(2, 1, 'world')
# 保存文件
wb.save('example.xls')
```

生成文件：





	A	B	C
1	1234.56		
2		8888	
3	hello	world	
4			
5			
6			
7			

常用的方法：

- Workbook 创建文件对象
- add\_sheet 新增一个表单
- write 在指定单元格写入数据

## 修改

很遗憾，并没有直接修改 xls 文件的方法。通常的做法是，读取出文件，复制一份数据，对其进行修改，再保存。

在复制时，需要用到 xlutils 中的方法。

```
from xlrd import open_workbook
from xlutils.copy import copy
# 打开文件
rb = open_workbook("example.xls")
# 复制
wb = copy(rb)
# 选取表单
s = wb.get_sheet(0)
# 写入数据
s.write(0, 1, 'new data')
# 保存
wb.save('example.xls')
```

修改后文件：

	A	B	C
1	1234.56	new data	
2	8888		
3	hello	world	
4			
5			

◀ ▶ A Test Sheet +

Ready Crossin的编程教室

特别要注意的是，选取读取表单时，要使用 `sheet_by_index`，而在选取写入表单时，则要用 `get_sheet`。不要问我为什么，我也很想知道这么设定的用意何在.....

## 时间转换

如果表单中有时间格式的数据，通过处理之后，你会发现时间数据出了差错。

	A	B	
1	1234.56	new data	
2	8888		8/31/16
3	hello	world	
4			
5			
6			

◀ ▶ A Test Sheet +

Ready Crossin的编程教室



	A	B	
1	1234.56	new data	
2	8888	42613	
3	hello	world	
4			
5			
6			

A Test Sheet

输出单元格内容：

```
[number:8888.0, xldate:42613.0]
```

因为这里 xldate 有自己的格式定义。如果要使用正确的格式，必须转换：

```
new_date = xlrd.xldate.xldate_as_datetime(date, book.datemode)
```

date 是对应单元格的数据，book 是打开的文件对象。

另外，在打开文件时，加上参数 `formatting_info=True`，可以保证在时间数据在 copy 时保持原样。

写入时间数据，则可通过此方法创建 excel 的时间对象：

```
xlrd.xldate.xldate_from_datetime_tuple
```

或者通过 `xlwt.easyxf` 指定时间格式：

```
style = xlwt.easyxf(num_format_str='D-MMM-YY')
ws.write(1, 0, datetime.now(), style)
```

具体细节及更多功能这里不展开说明。

以上便是 Python 操作 Excel 文件的一些基本方法。实际使用过程中遇到问题或者需要了解更多功能，永远记住两个词：

**RTFM**、**STFW**

参考资料：

<http://www.python-excel.org/> (<https://link.zhihu.com/?target=http%3A//www.python-excel.org/>)

<https://github.com/python-excel> (<https://link.zhihu.com/?>)

target=https%3A//github.com/python-excel)

<https://github.com/python-excel/tutorial/raw/master/python-excel.pdf>  
(<https://link.zhihu.com/?target=https%3A//github.com/python-excel/tutorial/raw/master/python-excel.pdf>)

来自：<https://zhuanlan.zhihu.com/p/22261597>

## 扩展阅读

命令行的艺术 (/lib/view/open1435670782060.html)  
构建基于Git为数据中心的CMS (/lib/view/open1448863736780.html)  
构建基于Git的静态全栈式框架 - CMS、移动应用、桌面编辑器 (/lib/view/open1448598449204.html)  
Python 的 Excel 操作库：xlwt (/lib/view/open1397098100184.html)  
用Python读取excel中的数据 (/lib/view/open1420378866734.html)

## 为您推荐

Python 与 Excel 不得不说的故事 (/lib/view/open1472701496085.html)  
让python在hadoop上跑起来 (/lib/view/open1453728831964.html)  
6.python模块（导入，内置，自定义，开源） (/lib/view/open1455236705292.html)  
构建基于Git为数据中心的CMS (/lib/view/open1448863736780.html)  
WEB前端规范 (/lib/view/open1449061887764.html)

## 更多

Python (<https://www.baidu.com/s?wd=site:open-open.com Python>)  
Python开发 (<https://www.baidu.com/s?wd=site:open-open.com Python开发>)

## 同类热门新闻



1. Linux命令行的艺术 (<http://www.open-open.com/news/view/1e0beac>)
2. Web开发人员有用的代码比较工具 (<http://www.open-open.com/news/view/8d518d>)
3. JavaScript，只有你想不到 (<http://www.open-open.com/news/view/afdb09>)
4. 【信息图】如何根据你想要的生活选择第一门编程语言 (<http://www.open-open.com/news/view/1ec0268>)

5. 学好 Python 的 11 个优秀资源 (<http://www.open-open.com/news/view/116c17f>)
6. Coursera 2015年最受欢迎的十门课，商业与程序相关最热门 (<http://www.open-open.com/news/view/6f0f8b>)

## 同类热门经验

---

1. 用Python做网页抓取与解析入门笔记 (/lib/view/open1346166586770.html)
2. python开发总结 (/lib/view/open1338942342094.html)
3. python开发总结二 (/lib/view/open1322892275249.html)
4. Python 数据分析包：pandas 基础 (/lib/view/open1402477162868.html)
5. 介绍Python程序员常用的IDE和其它开发工具 (/lib/view/open1324649712030.html)
6. 一个python的邮件发送脚本，自动，定时，可以附件发送，抄送，附有说明文件 (/lib/view/open1358769923989.html)





**相关文档** — 更多 (<http://www.open-open.com/doc>)

- Working with Excel files in Python.pdf (<http://www.open-open.com/doc/view/3a95b07d418a4cebb738c732626ef7ed>)
- Creating Excel files with Python and XlsxWriter.pdf (<http://www.open-open.com/doc/view/f59bbaf64ab148ebbd40a2c7f25a70c9>)
- 27篇python/小片段文章.pdf (<http://www.open-open.com/doc/view/1ace62ecdf0e49a3a3e3526f6d20df60>)
- Python灰帽子：黑客与逆向工程师的python编程之道英文版.pdf (<http://www.open-open.com/doc/view/4e8f5e0ada60409895bd77996fa50ea7>)
- Python灰帽子：黑客与逆向工程师的Python编程之道.pdf (<http://www.open-open.com/doc/view/de7961035b754cf4b46521d482d1ac89>)
- python灰帽子：黑客与逆向工程师的python编程之道.pdf (<http://www.open-open.com/doc/view/e0a42d0659cb4d4bacb56b58cd351b55>)
- Python灰帽子 - 黑客与逆向工程师的Python编程之道(中文版).pdf (<http://www.open-open.com/doc/view/d5892c37f2e74a04a1db6b8f81484a61>)
- Think.Stats-Python与数据分析.pdf (<http://www.open-open.com/doc/view/220a68aaf9454aa696777eb6028383b4>)
- 黑客与逆向工程师的Python编程之道(英文版).pdf (<http://www.open-open.com/doc/view/78fc9c3fc9c74fb7bc116c04ca9562d7>)
- python 3.X 与python 2.x的区别.doc (<http://www.open-open.com/doc/view/0d67d116a2df40da9554cb3c9045b6b2>)
- Perl 与 Python 之间的一些异同.pdf (<http://www.open-open.com/doc/view/7d95cef9a0c84f619fb4aeb8f2a871cc>)
- python并发与并行的新手指南.docx (<http://www.open-open.com/doc/view/f732e746eb7a40c1b0636f0dddaad1bd>)
- Dive into Python 3.pdf (<http://www.open-open.com/doc/view/2db9858b17ca4f758e64efbde36665bc>)
- 利用Python 与 Qt进行快速GUI编程.pdf (<http://www.open-open.com/doc/view/cd820d8ce820469da8a2bf4933a0c4b4>)
- 《与小卡特一起学Python》前言+目录+前两章.pdf (<http://www.open-open.com/doc/view/28306bdf36141949dc4d30d37ccfa6c>)
- Selenium2与Python自动化测试环境搭建.docx (<http://www.open-open.com/doc/view/cd21a4c162cd4ba98254725759e5fafd>)
- python与tkinter编程.pdf (<http://www.open-open.com/doc/view/6c892fc1ac8c42fc8bb78a9c94b00883>)
- Python Web 开发学习实录.pdf (<http://www.open-open.com/doc/view/813edf10dd3d4154a73724b58422361b>)
- Dive Into Python 中文版.chm (<http://www.open-open.com/doc/view/ad37a75522824d5abeaa3715c7717a58>)

**相关经验** — 更多

- 命令行的艺术 (<http://www.open-open.com/lib>)
- 构建基于Git为数据中心的CMS (<http://www.open-open.com/lib/view/open1435670782060.html>)
- 构建基于Git的静态全栈式框架 - CMS、移动应用、桌面编辑器 (<http://www.open-open.com/lib/view/open1448863736780.html>)
- Python 的 Excel 操作库：xlwt (<http://www.open-open.com/lib/view/open1397098100184.html>)
- 用Python读取excel中的数据 (<http://www.open-open.com/lib/view/open1420378866734.html>)
- Python框架、库和软件资源大全 【同僚】 (<http://www.open-open.com/lib/view/open1436275528583.html>)
- Python的Excel处理模块：XlsxWriter(<http://www.open-open.com/lib/view/open1374562003574.html>)
- PostgreSQL vs. MS SQL Server (<http://www.open-open.com/lib/view/open1417568649964.html>)
- Python、R 语言、SAS、SPSS 优缺点比较 (<http://www.open-open.com/lib/view/open1420785625937.html>)
- 加速的Excel XLSX生成的Python 库：PyExcelerate (<http://www.open-open.com/lib/view/open1420790232359.html>)
- Python中读取excel文件的利器：xlrd (<http://www.open-open.com/lib/view/open1420536000046.html>)
- python处理Excel (<http://www.open-open.com/lib/view/open1420378281046.html>)
- python使用xlrd处理Excel (<http://www.open-open.com/lib/view/open1433409709854.html>)
- python读取和生成excel文件 (<http://www.open-open.com/lib/view/open1433410333619.html>)



- Python介绍.ppt (<http://www.open-open.com/doc/view/5cbc725a8a4a435ea734b0be4d487c5d>)

---

©2006-2016 深度开源



(<http://www.open-open.com/>)

浙ICP备09019653号-31

(<http://www.miibeian.gov.cn/>)

站长统计

(<http://www.cnzz.com/stat/websit>

web\_id=1257892335)

