

# OpenPyxl

1. 從模組導入活頁簿

```
from openpyxl import Workbook
```

Capital

2. 建立活頁簿

```
wb = Workbook() # wb stand for workbook
```

3. 建立初始表單

```
ws = wb.active # ws stand for worksheet.
```

\* 該函數調用工作表的索引 \_active\_sheet\_index

除非重新賦值，此表單會恆為序列首。

4. 插入額外表單 (optional)

```
ws1 = wb.create_sheet() # 默認插在末位.
```

```
ws2 = wb.create_sheet(index=0) # 插入在工作簿首位.
```

5. 修改表單名稱和標籤顏色。

```
ws.title("New Title")
```

# RGB

```
ws.sheet_properties.tab_color = "EE27EB"
```

6. 通過 key 和方法 調用

```
ws3 = wb["New Title"]
```

```
ws4 = wb.get_sheet_by_name("New Title")
```

```
print(ws is ws3 is ws4)
```

# Result: True

## 7. 獲取工作表所有表單名稱

(a) 方法

`print(wb.get_sheet_by_name)` # 返回列表。

# Result: ["Sheet2", "NewTitle", "Sheet1"]

(b) for 循環。

`for sheet in wb:`

`print(sheet.title)`

## 8. 操作數據

透過 `key` 來調用索引。

`c = ws["A4"]`

調用，若不存在則創建，也可以直接賦值。

`ws["A4"] = 4`

透過方法

`c = ws.cell("A4")`

根據行(標) Column, 列 Row 值獲取單元格。

`d = ws.cell(row=4, column=2)`

注意：創建工作表後並不包含單元格。當單元格被取用後才會被創建。這種方式能有效減少記憶體的消耗。

警告：由於上述特性，即便僅索引了單元格而非使用，它們也會被在記憶體中被創建。

`for i in range(1, 10):`

`for j in range(1, 10):`

`ws.cell(row=i, column=j)`

遍 遍 即 片 使 用 複 數 單 元 格。

cell\_range = ws["A1": "C2"]

for i in cell\_range:

for j in i:

j.value = 10000

遍 遍 預 設 的 進 代 方 法 It's not working.  
for me.

9. 存 取 數 擟

# 加 入 日 期

import datetime as dt

datetime\_cell = ws["A12"]

datetime\_cell.value = dt.datetime.now()

(2) 最 大 列 級、最 小 行 數 應 用

words = [chr(i) for i in range(65, 91)]

current\_max\_row = ws.max\_row

print(current\_max\_row)

for i in range(0, len(words)):

foo = ws.cell(row=ws.max\_row,  
column=ws.min\_column)

foo.value = words[i]

# 11. 保存文件

wb = Workbook()

wb.save("File\_name.xlsx")

覆改時不會提示

File extension

預設為 False

可另存為模板。

wb.save("File\_name.xlsx", as\_template=True)

# 12. 导入活頁簿

lower case.

from openpyxl import load\_workbook

wb2 = load\_workbook("test.xlsx")

print(wb2.get\_sheet\_names())

sheet = wb2.get\_sheet\_by\_name("New Title")

# 14. 文字操作

Let text surround  
by pictures

(a) Set alignment inside the cell and text wrap.

Sheet.cell(row=1, column=1).alignment =

capital

lower case

openpyxl.styles.Alignment(horizontal="center",  
vertical="center",  
wrap\_text=True)

(b) To make font **Bold** or **Italic** (lower case)  
sheet.cell(row=1, column=1).font =

openpyxl.styles.Font(bold=True, italic=True)

(c) merge cells

sheet.merge\_cells("A1:C1")

	A	B	C	D	E
1					
2					
3					
4					
5					
6					

Start cell: A1

sheet.merge\_cells(start\_row=1, start\_column=1,  
end\_row=3, end\_column=3)

End cell: C3

(c) Remove sheet

wb.remove\_sheet(sheet)

\* initial always start with 1

(d) Insert row and column

ws.insert\_rows(7)



(e) Delete row and column [F:H]

ws.delete\_cols(6, 3)

start ↙ Step

(f) Moving ranges of cells

WS.move\_range("B3:D4",

	A	B	C	D	E	F
1						
2						
3						
4						
5						
6						

rows=-1, cols=2)

↑ 1 → 2

Sometime you need to move the relative references in Formula in the range by one row and one column.

WS.move\_range("B3:D4", rows=-1, cols=2,  
translate=True)

Refer:

Openpyxl Documentation

[https://openpyxl.readthedocs.io/en/stable/editing\\_worksheets.html](https://openpyxl.readthedocs.io/en/stable/editing_worksheets.html)

<https://www.jianshu.com/p/642456aa93e2>

Tarun Gupta

<https://hackernoon.com/working-with-spreadsheets-using-python-part-2-c8b4ea7e2b65>