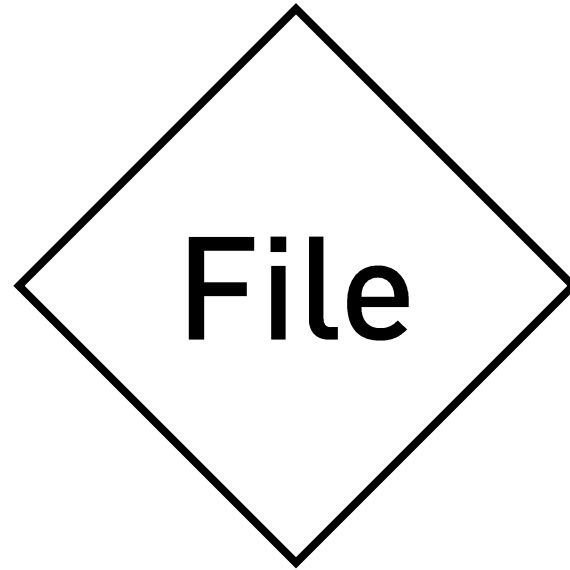


在開始前...

請到google classroom下載以下檔案  
detective

| 今天我們要學... |



Open / Close

Read / Write

為什麼我需要學 file ?

file

txt

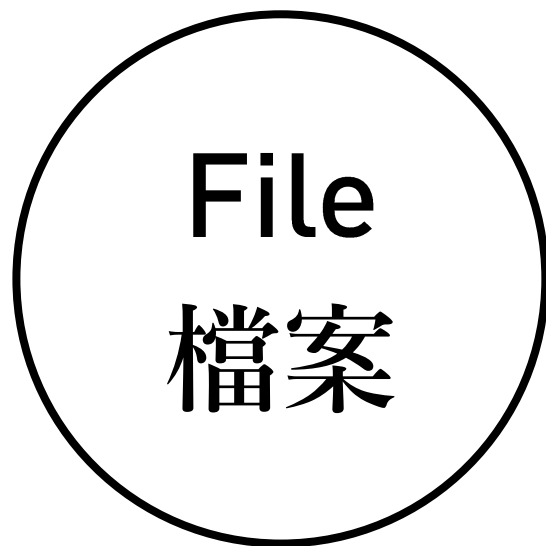
csv

json

| 請你先想想... |

– 不同的建築物 –

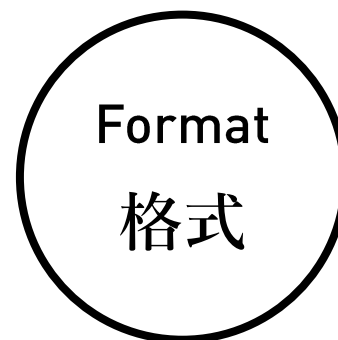
- 你家
- 百貨公司
- 學校



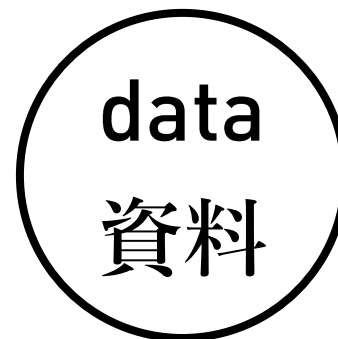
成功大學



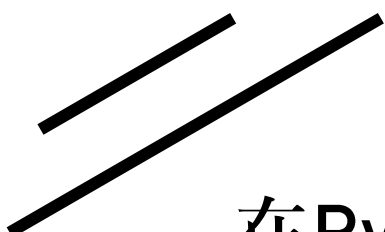
| 地址 |



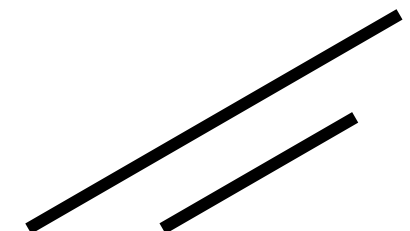
| 教育機構 |



| 學生、老師、教學... |



在Python中  
每個檔案也有自己的“地址”  
我們稱為 – 路徑



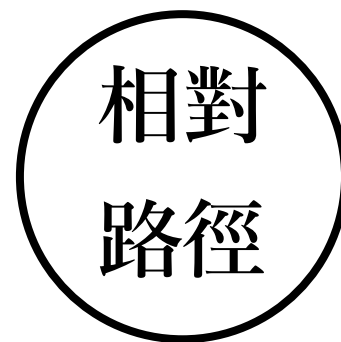




成功大學



| 台南市東區大學路1號 |



| 我家對面 |  
(相對於“我家”)

C:/

└─ user

└─ list.csv

└─ project/

    └─ 1\_part/

        └─ intro.md

        └─ program.txt

    └─ 2\_part.py

    └─ 3\_part/

        └─ stat.json

└─ Q&A

## 絕對路徑 從根目錄一路到檔案

我目前在 2\_part.py 這個檔案  
我該如何使用 stat.json ?

絕對路徑：呼叫檔名時從根目錄(最上層)往下找到你需要的檔案

stat.json的絕對路徑 → ex: C:/project/3\_part/stat.json

那要怎麼找program.txt這個檔案？

C:/

└─ user

└─ list.csv

└─ project/

└─ 1\_part/

└─ intro.md

└─ program.txt

└─ 2\_part.py

└─ test.txt

└─ 3\_part/

└─ stat.json

└─ Q&A

## 相對路徑 從我家去找鄰居

我們目前在 2\_part.py 這個檔案

相對路徑：根據現在檔案的位置，往上層、往下層，定位另一個檔案

“ / ”：到最上層(根目錄)

Ex: “ /list.csv ” 會到最上層 (C:) 找到 list.csv 檔

C:/

└─ user

└─ list.csv

└─ project/

└─ 1\_part/

└─ intro.md

└─ program.txt

└─ 2\_part.py

└─ test.txt

└─ 3\_part/

└─ stat.json

└─ Q&A

## 相對路徑 從我家去找鄰居

我們目前在 2\_part.py 這個檔案

相對路徑：根據現在檔案的位置，往上層、往下層，定位另一個檔案

“ ../ ”：到上一層

Ex: “../test.txt”會到上一層(project)再找到test.txt

C:/

└─ user

└─ list.csv

└─ project/

└─ 1\_part/

└─ intro.md

└─ program.txt

└─ 2\_part.py

└─ test.txt

└─ 3\_part/

└─ stat.json

└─ Q&A

## 相對路徑 從我家去找鄰居

我們目前在 2\_part.py 這個檔案

相對路徑：根據現在檔案的位置，往上層、往下層，定位另一個檔案

“ ./ ” 或者 “ 直接輸入資料夾/檔案的名字 ”：在目前這層

Ex: “ ./3\_part/stat.json ” 會先找到3\_part資料夾，接著在裡面找到stat.json檔

那要怎麼找program.txt這個檔案？

# | File的基本操作 |

1.打開檔案



open

2.對檔案做什麼事



模式

3.關閉檔案



close

```
f = open("test.txt", "r") #打開檔案
```

你要使用的檔案(絕對or相對路徑)開啟

模式

```
print(f.read())
```

```
f.close #關閉這個檔案
```



- open : 打開你要的檔案
- close : 對檔案做完動作後, 關閉檔案

- 切記 -

open一個檔案, 也要記得close檔案

open & close成對出現

Ex: open & close 檢查

我要怎麼檢視檔案的狀況?  
執行下面這段code看看

```
f = open(" test.txt ", "r")  
print(f.closed)  
f.close  
print(f.closed)
```

close() 是 **method**

當然也要加上括號!!

f.close()

f.close()

f.close()

那...

我如果不使用 `close()` ...  
會發生什麼事？

寫文件的時候，系統部會立刻把資料寫進硬碟儲存，而是放進暫存，等空閒時再一點一點存進硬碟，用`close()`可以保證系統把剩下的資料都寫進硬碟，否則系統存進一部分，剩下的可能丟失

# 接下來我們進入遊戲畫面

請大家用python打開house資料夾

1. 打開me.py, 這個檔案代表你現在位於玄關
2. 使用剛剛學到的絕對路徑/相對路徑進行操作

閱讀日記

with：更方便的開啟檔案方式，而且自動close()

**\*務必要會，接下來會常用到！！**

```
with open("__檔案名稱__", "r") as f:  
    #do something
```

```
with open("test.txt", "r") as f: #打開檔案
```



模式

# 模式-對檔案做什麼事

1. 功能
2. 若開的檔案存在/不存在，會怎麼處理
3. 開始處理的位置

	功能	開不存在的檔案	開存在的檔案	開始處理的位置
r	讀	Error	不改變內容	開頭
w	寫	創建新的檔案	清空內容	開頭
a	讀 + 寫	創建新的檔案	不改變內容	結尾
r+	讀 + 寫	Error	不改變內容	開頭
w+	讀 + 寫	創建新的檔案	清空內容	開頭
a+	讀 + 寫	創建新的檔案	不改變內容	結尾



# 模式-對檔案做什麼事

Mode (模式) :

原檔

0	1	2	3	4	5	6	7	8	9
L	e	a	r	n		f	i	l	e

r, r+

0	1	2	3	4	5	6	7	8	9
L	e	a	r	n		f	i	l	e



w, w+

0



a, a+

0	1	2	3	4	5	6	7	8	9	10
L	e	a	r	n		f	i	l	e	



Read / write

read / write

# File的基本操作-read

## Read 讀檔

1. `read()`：讀整份文件
2. `readline()`：每次讀一行
3. `readlines()`：將每一行的值以列表回傳
4. 補充：“rb”以二進位形式讀檔， ex:圖片

# 回到我們的房間

在Andrew的房間內  
打開me\_R\_A.py

閱讀日記

# File的基本操作-write

寫入內容

`f.write('要寫入的內容')`

`*( )`括號內的type必須是str

# File的基本操作-2個小技巧

`tell()`：返回(告訴你)目前的處理位置

ex:

```
with open("a.txt", "w") as f:  
    f.write('abcdef')  
    print(f.tell())
```

```
>>> 6
```

# File的基本操作-2個小技巧

`seek()`：回到指定的處理位置，括號內放指定的index

ex:

```
with open("a.txt", "w") as f:  
    f.write('abcdef')  
    print(f.tell())  
    f.seek(3)  
    f.write('5')
```

```
>>> 6
```

```
>>> abc5efg
```

# 偵。探

根據目前的線索接下來你要逐步完成動作  
請照著檔案內的指示做



# Hint

## 英文字母亂碼表

g \_ \_ l \_ \_ a

h \_ m m \_ r

a \_ r \_ w

c \_ u \_ d

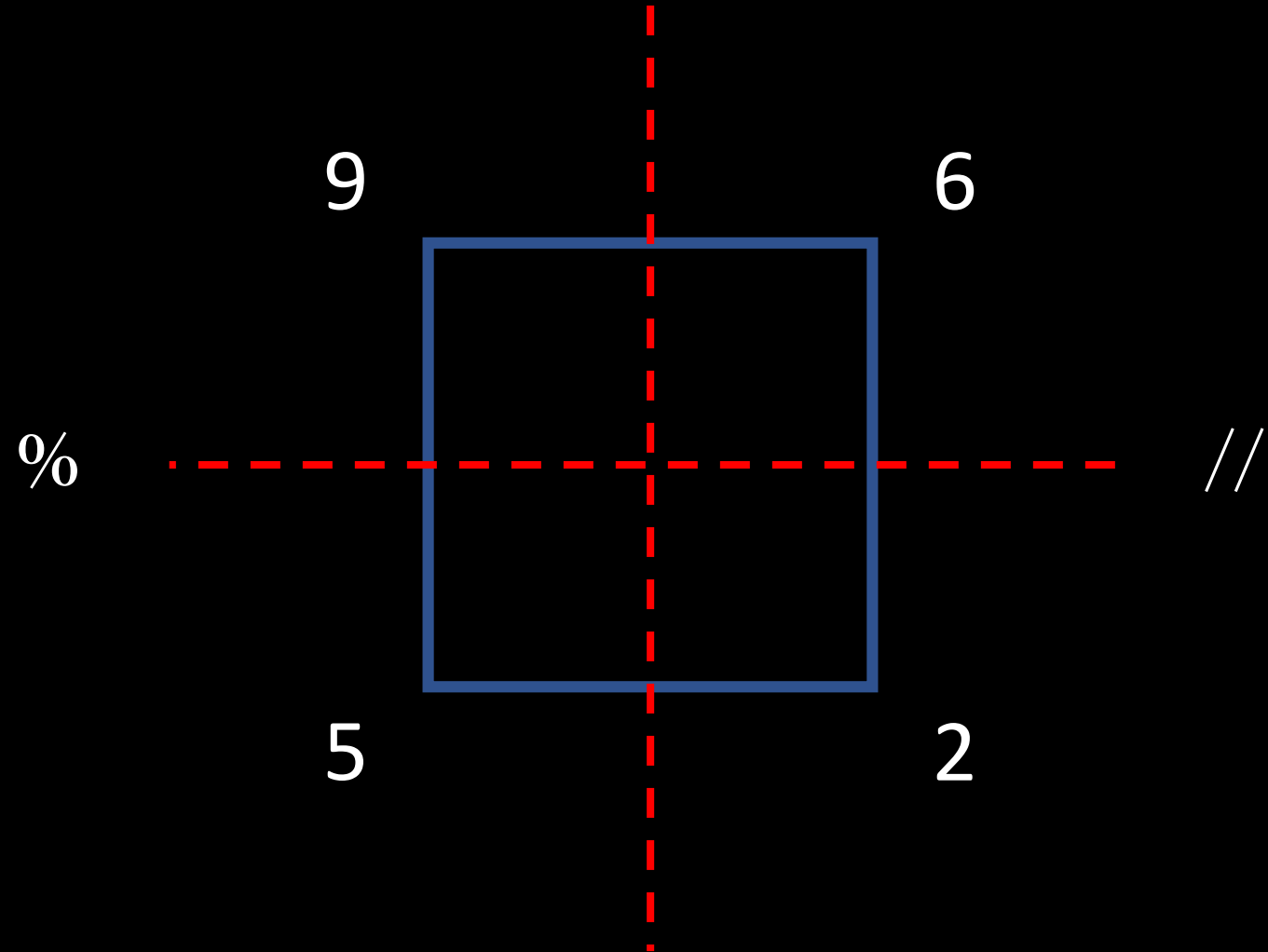
s \_ e \_ r

s \_ \_

b \_ \_ e

d \_ \_

# Hint



Hint

找出竊賊吧

✓   -   、   /   -

3 1 ...