Python
File (Simple)

臺北科技大學資訊工程系

開啟檔案

- <file> = open(<filename>, <mode>)
  - <file>：開啟檔案物件
  - <filename>：檔名
  - <mode>：開檔模式
    - r: 唯讀
    - w (write): 覆蓋寫入內容
    - a (append): 在後加入新內容

```
infile = open('hello.txt', 'r') # 以讀取模式開檔

outfile = open('hello.txt', 'w') # 以寫入模式開檔
```

符號	開檔模式	開檔模式
r	只讀取	
rb	只讀取(binary)	
r+	可讀取	可寫入
rb+	可讀取 binary	可寫入 binary
w		只寫入
w+	可讀取	可寫入
wb		只寫入binary
a		只新增
a+	可讀取	可新增

關閉檔案

- <file>.close() #將緩衝區內容寫入檔案，關閉檔案
  - 檔案讀寫可能產生IOError，而不會呼叫fp.close()造成關檔錯誤 。
  - 為保證是否出錯，都正確關閉檔案，可使用try ... finally

```
try:
  fp = open('hello.txt', 'r')
  print(fp.read())
  fp.write('test')
except FileNotFoundError:
  print("file not found")
except:
  print("Something wrong")
finally:
  fp.close()
```

關閉檔案

- 用 with 開檔，若錯誤，檔案仍會自動關閉，

```
with open('hello.txt', 'r') as fp:
```

- 錯誤情境：
  - 執行 return, continue, break 跳出 with 指令區塊
  - 發生例外 (Exception)

讀取檔案

- <file>.read()
  - 讀取全部或剩餘資料，回傳長字串

```
with open('filename.txt', 'r') as infile:
    data = infile.readline()    # 一次讀全部
    print(data)

infile.close()
```

讀取檔案

- <file>.readline()
  - 讀取下一行資料，回傳字串
  - 利用迴圈一次讀一行資料

```
with open('filename.txt', 'r') as infile:
    while True:                        # 無窮迴圈
        data = infile.readline()    # 一次讀一行資料
        print(data)
        if not data:                   # 所有資料讀取完畢
            print('end reading...')
            break

infile.close()
```

讀取檔案

- <file>.readlines()
  - 讀取全部或剩餘資料，回傳串列，每個元素都是一行資料

```
outfile = open('hello.txt', 'w')  # 以寫入模式開檔
outfile.write('test1甲\n')
outfile.write('test2乙\n')
outfile.write('test3丙\n')
outfile.close()

# Python 讀檔將每行資料存到串列中的元素
with open('hello.txt', 'r') as fp:
    data = fp.readlines()
    print(type(data))
    print(data)
```

```
<class 'list'>
['test1甲\n', 'test2乙\n', 'test3丙\n']
```

讀取檔案

- 使用 for 迴圈: 一行一行處理

```
with open('filename.txt', 'r') as infile:
    for line in infile.readlines():   # 一次讀取所有資料，再 for 迴圈 一行一行處理
        print(line, end=‘’)
```

```
test1甲
test2乙
test3丙
```

```
# 上述程式可簡化
with open('filename.txt', 'r') as infile:
    for line in infile:
        print(line, end='')
```

```
test1甲
test2乙
test3丙
```

寫檔 4 個方法

- write

```
# 開啟檔案
fp = open("out.txt", "a")

# 寫入 This is a testing! 到檔案緩衝區
fp.write("This is a testing!")
#將緩衝區寫入檔案，關閉檔案
fp.close()
```

```
This is a testing!
```

- writelines

```
# 開啟檔案
fp = open("filename.txt", "w")

# 將 lines 所有內容寫入到緩衝區
lines = ["One\n", "Two\n", "Three\n"]
fp.writelines(lines)
#將緩衝區寫入檔案，關閉檔案
fp.close()
```

```
One
Two
Three
```

寫檔 4 個方法

- print

```
# 開啟檔案
fp = open("out.txt", "a")

# 寫入 This is a testing! 到檔案緩衝區
print("This is a testing!\n", file = fp )
#將緩衝區寫入檔案，關閉檔案
fp.close()
```

```
This is a testing!
```

- with open

```
text = ["this is ","a book"]
with open("out.txt", 'a') as out_file:
    for line in text:
        out_file.write(line)
```

```
this is a book
```

Exercise

- 利用迴圈一次讀一行資料，將偶數行資料印出

```
with open('filename.txt', 'r') as infile:
    line_num=0
    for line in infile: # 讀取所有行
        line_num += 1
        if line_num % 2 == 0:
           print(line, end='')
```

```
bb
dd
```

filename.txt

```
aa
bb
cc
dd
```

Exercise

- 一次讀取、印出多行資料，
- 將第一行的第一個字與最後一個字印出

```
fp = open('hello.txt', 'w')
fp.write(“First line\n#Second line\n#Third line")
fp.close()

with open('hello.txt', 'r') as infile:
    data = infile.read()              # 一次讀取多行資料
    print(data)                       # 印出多行資料
    print(data[0], data[-1])          # 將第一個字元與最後一個字元印出
```

hello.txt

```
First line
#Second line
#Third line
```

```
First line
#Second line
#Third line
F e
```

讀寫檔案

- 顯示檔案所有行，忽略以#開頭的行

```
with open("hello.txt") as f:
    for line in f:
        if line.strip()[0] !=  "#":    # 忽略 # 開頭的行
            print(line)
```

```
First line
```

hello.txt

```
First line
#Second line
#Third line
```

```
strip()  # 刪除頭尾的空格

txt = ",,,,,rrttgg.....banana....rrr"
x = txt.strip(",.grt")
print(x)
```

```
banana
```

讀寫檔案

- ❑ 把 passwd 檔案中 'root' 字串用 'west' 替換，另存 tmp.txt 檔案

```
with open("passwd.txt") as f1:
    # 遍歷檔案的每一行內容;
    for line in f1:
        # 字串替換
        bline = line.replace("root", "west")
        with open("tmp.txt", "a+") as f2:
            # 寫入新檔案
            f2.write(bline)
```

passwd.txt

```
root        word
user        pass
```

tmp.txt

```
west        word
user        pass
```

Exercise

- 讀取 English.txt 檔案，將其中 x_i 字串以 y_i 替換，

I love cat and love dog, but I am afraid of tiger.

- X, Y 分別存在 translate.txt 檔案的第一列和第二列
  - 串列X = [x_i], 串列Y = [y_i]，
  - 串列X 是英文，串列Y 是中文翻譯

translate.txt

cat dog tiger
貓 狗 老虎

x_1	x_2	x_3
y_1	y_2	y_3

- 存檔 Chinese.txt

I love 貓 and love 狗, but I am afraid of 老虎.

Exercise

- code

```
def getHeader():   #讀取檔案第一列和第二行
    i = 0
    with open('translate.txt', 'r', encoding="utf-8") as infile:
        for line in infile:
            if i == 0:
                eng = line.split()
                print(eng)
            else:
                chi = line.split()
                print(chi)
            i = i + 1
    return eng, chi
```

```
['cat', 'dog', 'tiger']
['貓', '狗', '老虎']
cat 貓        I love 貓 and love dog, but I am afraid of tiger.
dog 狗        I love 貓 and love 狗, but I am afraid of tiger.
tiger 老虎  I love 貓 and love 狗, but I am afraid of 老虎.
I love 貓 and love 狗, but I am afraid of 老虎. ####
```

```
def convert(aFile, bFile, eng, chi):
    f1 = open(aFile, 'r', encoding="utf-8")
    f2 = open(bFile, 'w', encoding="utf-8")
    data = f1.read()
    # zip 將 eng, chi 打包成 tuple
    for e, c in zip(eng, chi):
        data = data.replace(e, c)
        print(e, c, data)
    print(data, '####')
    f2.write(data)
    f1.close()
    f2.close()

eng, chi = getHeader()
convert('English.txt', 'Chinese.txt', eng, chi)
```

讀取CSV檔案

- csv.reader() 讀取 csv 資料，一列一列印出

```
import csv
f= open('data.csv', encoding = 'utf-8') # csv 檔案是以 UTF-8 編碼
#f= open('data.csv')
readFile = csv.reader(f)          #<_csv.reader object at 0x0000025AE3EE2EC0>
print(readFile)
for row in readFile:
    print(row)
f.close()
```

data.csv

```
班級,學號,期中考成績
一甲,110591052,80
一甲,110591053,90
一乙,110591054,85
一丙,110591055,75
一丙,110591056,95
一丙,110591057,80
一丙,110591058,90
一乙,110591059,100
一乙,110591060,85
一乙,110591061,95
```

```
['班級', '學號', '期中考成績']
['一甲', '110591052', '80']
['一甲', '110591053', '90']
['一乙', '110591054', '85']
['一丙', '110591055', '75']
['一丙', '110591056', '95']
['一丙', '110591057', '80']
['一丙', '110591058', '90']
['一乙', '110591059', '100']
['一乙', '110591060', '85']
['一乙', '110591061', '95']
```

- 使用 with 開啟 csv 檔案
  - 加上 newline=''，為讓資料中包含的換行符號可正確解析

```
import csv
with open('data.csv', encoding = 'utf-8', newline = '') as csvfile:
    readFile = csv.reader(csvfile)
    for row in readFile:
        print(row)
```

newline=''

https://stackoverflow.com/questions/61861172/what-does-the-argument-newline-do-in-the-open-function

```
import csv

with open('csv_file.csv', 'w') as f:
    writer = csv.writer(f)
    writer.writerow(['row', 'one'])
    writer.writerow(['row', 'two'])

with open("csv_file.csv") as f:
    reader = csv.reader(f)
    for row in reader:
        print(row)
```

```
['row', 'one’]
[]
['row', 'two’]
[]
```

```
import csv

with open('csv_file.csv', 'w', newline='') as f:
    writer = csv.writer(f)
    writer.writerow(['row', 'one'])
    writer.writerow(['row', 'two'])

with open("csv_file.csv", newline='') as f:
    reader = csv.reader(f)
    for row in reader:
        print(row)
```

```
['row', 'one’]
['row', 'two’]
```

The reason why the csv documentation recommends opening a file with newline='' is because the csv writer terminates each line with a \r\n as mentioned in the official documentation. Now, on Windows, when writing to the stream using the open() function, if the value is None, at the ending of a line, the \r stays as it is while \n gets translated to \r\n according to the writing output to the stream section above. This means that a text like:

Line one
Line two

is in the form: 'Line one\r\r\nLine two\r\r\n’.

讀取CSV檔案

- 指定分隔符號
  - 資料欄位分隔符號非使用預設逗號，而是其他符號，讀取時要指定分隔符號

```
import csv

with open('data2.csv', encoding = 'utf-8', newline = '') as csvfile:
    readFile = csv.reader(csvfile, delimiter = ':')
    for row in readFile:
        print(row)
```

data.csv

```
班級:學號:期中考成績
一甲:110591052:80
一甲:110591053:90
一乙:110591054:85
一丙:110591055:75
一丙:110591056:95
一丙:110591057:80
一丙:110591058:90
一乙:110591059:100
一乙:110591060:85
一乙:110591061:95
```

讀取CSV檔案

- class.csv

```
班級,學號,期中考成績
一甲,110591052,80
一甲,110591053,90
一乙,110591054,85
一丙,110591055,75
一丙,110591056,95
一丙,110591057,80
一丙,110591058,90
一乙,110591059,100
一乙,110591060,85
一乙,110591061,95
```

```
<csv.DictReader object at 0x0000025CD174B140>
一甲 110591052 80
一甲 110591053 90
一乙 110591054 85
一丙 110591055 75
一丙 110591056 95
一丙 110591057 80
一丙 110591058 90
一乙 110591059 100
一乙 110591060 85
一乙 110591061 95
```

- 讀取csv 檔案內容後，轉為 dictionary 格式
  - csv.DictReader() 自動把第一列(row)當作欄位名稱，
  - 第二列後的每一列轉為 dictionary，如此可以使用欄位名稱存取資料

```
import csv
with open('class.csv', encoding = 'utf-8', newline = '') as csvfile:
    readFile = csv.DictReader(csvfile)
    print(readFile)          #印出 <csv.DictReader object at 0x0000025AE3F9F010>
    for row in readFile:
        print(row['班級'], row['學號'], row['期中考成績'])
```

寫入CSV檔案

- 一次寫入二維表格
  - 若資料是已整理好二維表格，可一次把整張表格寫進 csv 檔案

```
import csv

# 二維表格
table = [['班級', '學號', '成績'],
        ['資工一', '109590001', 90],
        ['資工一', '109590002', 85]]

with open('output.csv', 'w', encoding = 'utf-8', newline='') as csvfile:
    writer = csv.writer(csvfile)
    writer.writerows(table)    # 寫入二維表格
```

寫入CSV檔案

- ❑ 寫入 Dictionary
  - ❍ 資料格式是 dictionary，可使用 csv.DictWriter()寫入 csv 檔案中

```
import csv

with open('output.csv', 'w', encoding = 'utf-8', newline='') as csvfile:
    columns = ['班級', '學號', '成績']
    # 將 dictionary 寫入 CSV 檔
    writer = csv.DictWriter(csvfile, fieldnames=columns, delimiter=':')
    writer.writeheader()   # 寫入第一列的欄位名稱
    writer.writerow({'班級': '資工一', '學號': '109590003', '成績': 95})   # 寫入資料
    writer.writerow({'班級': '資工一', '學號': '109590004', '成績': 88})   # 寫入資料
```

delimiter=‘:’ is optional

output.csv

```
班級:學號:成績
資工一:109590003:95
資工一:109590004:88
```

Exercise

- 一行一行讀檔案 score.txt
  - 計算平均，將平均寫到最下面

Input file: score.txt

```
班級,學號,期中考成績,
資工一,109590001,88,
資工一,109590002,90,
資工一,109590003,92,
資工一,109590004,85,
資工一,109590005,87,
資工一,109590006,95,
資工一,109590007,80,
資工一,109590008,84,
資工一,109590009,86,
資工一,109590010,83,
```

Output file: avg_score.txt

```
Class,Student ID,Score,
資工一,109590001,88,
資工一,109590002,90,
資工一,109590003,92,
資工一,109590004,85,
資工一,109590005,87,
資工一,109590006,95,
資工一,109590007,80,
資工一,109590008,84,
資工一,109590009,86,
資工一,109590010,83,
平均,,87.0,
```

Exercise

- 製作一個 csv 檔 score.csv
  - 一行一行讀檔案 score.csv，製作成字典
  - 計算每位學生平均，寫在學生資料最後，
  - 計算全班平均，寫到最下面
- 輸出成 output.csv

Input file: score.csv

```
班級,學號,國文,數學,英文
資工一,109590001,80,80,80
資工一,109590002,90,90,90
資工一,109590003,70,70,70
資工一,109590004,60,60,60
```

Output file: output.csv

```
Class,Student ID,average,
資工一,109590001,80,
資工一,109590002,90,
資工一,109590003,70,
資工一,109590004,60,
75,75,75,75
```

Exercise

```
import csv
def trans(row):
    data = {}
    score = 0
    subject = ['國文','英文','數學']
    for key, value in row.items():
        print('=>', key, value)
        if key in subject:
           score = score + int(value)
    for key, value in row.items():
        if key=='班級' :
           data['Class'] = value
        elif key=='學號':
           data['Student Id'] = value
    data['average'] = score//3
    return data
```

```
with open('score.csv", encoding = 'utf-8',, newline='') as csvfile:
    readFile = csv.DictReader(csvfile)
    #print(readFile)
    inData = []
    for row in readFile:
        print(row)
        inData.append(trans(row))
```

```
{'班級': '資工一', '學號': '109590001', '國文': '80', '數學': '80', '英文': '80'}
{'班級': '資工一', '學號': '109590002', '國文': '90', '數學': '90', '英文': '90'}
{'班級': '資工一', '學號': '109590003', '國文': '70', '數學': '70', '英文': '70'}
{'班級': '資工一', '學號': '109590004', '國文': '60', '數學': '60', '英文': '60'}
```

```
print(inData)
```

```
[{'Class': '資工一', 'Student Id': '109590001', 'average': 80},
{'Class': '資工一', 'Student Id': '109590002', 'average': 90},
{'Class': '資工一', 'Student Id': '109590003', 'average': 70},
{'Class': '資工一', 'Student Id': '109590004', 'average': 60}]
```

```
with open('output.csv', 'w', newline='') as csvfile:
    #columns = ['班級', '學號','國文','數學','英文']
    columns = ['Class', 'Student Id','average']
    # 將 dictionary 寫入 CSV 檔
    writer = csv.DictWriter(csvfile, fieldnames=columns, delimiter=',')
    writer.writeheader()   # 寫入第一列的欄位名稱
    for data in inData:
        writer.writerow(data)   # 寫入資料
```

```
Class,Student Id,average
資工一,109590001,80
資工一,109590002,90
資工一,109590003,70
資工一,109590004,60
```

END