

Python60 系列 成績計算



</PLUS>

推動大學(程式設計)教學
MOE/Ministry of National Education

推動大學程式設計教學計畫。分項六：資料分析領域與學習評量推動團隊（德明財經科大資科系江政杰老師主編）

成績計算：課程簡述

▶ 問題情境：

- 學期末了，老師要計算學生的學期總分。
學生成績包含三次作業以及期中、期末考試。
 - 作業佔總成績 50%、期中、期末考試分別佔總成績 20% 及 30%。
- 老師決定放寬計分方式，三次作業只採計最高的兩次。
- 資料輸入與輸出皆以 CSV 檔案處理。

▶ 本課程將撰寫一個 17 行的程式， 實際課程時間 20-30 分鐘。

	A	B	C	D	E	F
1	姓名	作業一	作業二	作業三	期中考	期末考
2	賴一梅	100	90	100	80	85
3	陳二哥	90	90	100	90	60
4	張三豐	30	40	100	50	70
5	李四瑞	60	70	90	80	90
6	王五郎	99	99	80	99	80
7	錢六順	88	88	100	88	60
8	孫七天	100	100	100	0	100
9	朱八解	100	0	100	100	100

	A	B	C	D	E	F	G
1	姓名	作業一	作業二	作業三	期中考	期末考	總成績
2	賴一梅	100	90	100	80	85	91.5
3	陳二哥	90	90	100	90	60	83.5
4	張三豐	30	40	100	50	70	66
5	李四瑞	60	70	90	80	90	83
6	王五郎	99	99	80	99	80	93.3
7	錢六順	88	88	100	88	60	82.6
8	孫七天	100	100	100	0	100	80
9	朱八解	100	0	100	100	100	100

成績計算：學習內容

▶ 程式設計概念

- 列表處理
- 字串處理

▶ 資料分析概念

- 簡易統計
- 排序

成績計算

▶ 程式思考

```
# 開啟資料輸入檔
# 讀取輸入檔內容
# 針對每一筆的成績資料
#     取出各項成績分數
#     加總期中期末成績佔總分50%
#     計算作業成績計分：前兩高分成績平均
#     計算總平均

# 開啟要輸出的新成績檔案
# 針對每一筆的成績資料
#     寫入完整成績資料
```

寫程式前請先想想要做甚麼事!!

	A	B	C	D	E	F
1	姓名	作業一	作業二	作業三	期中考	期末考
2	賴一梅	100	90	100	80	85
3	陳二哥	90	90	100	90	60
4	張三豐	30	40	100	50	70
5	李四瑞	60	70	90	80	80
6	王五郎	99	99	80	99	80
7	錢六順	88	88	100	88	60
8	孫七天	100	100	100	0	100
9	朱八解	100	0	100	100	100



	A	B	C	D	E	F	G
1	姓名	作業一	作業二	作業三	期中考	期末考	總成績
2	賴一梅	100	90	100	80	85	91.5
3	陳二哥	90	90	100	90	60	83.5
4	張三豐	30	40	100	50	70	66
5	李四瑞	60	70	90	80	90	83
6	王五郎	99	99	80	99	80	93.3
7	錢六順	88	88	100	88	60	82.6
8	孫七天	100	100	100	0	100	80
9	朱八解	100	0	100	100	100	100

成績計算

完整程式展示

```
results = []
with open('學生成績.csv') as infile:
    data = infile.read().split()
    for e in data[1:]:
        stu = e.split(',')
        scores = [int(sc) for sc in stu[1:]]
        exams = scores[3]*0.2 + scores[4]*0.3
        hw = sum(sorted(scores[0:3])[1:])/2
        final = round(hw*0.5+exams, 2)
        print(stu[0], scores, '作業平均', hw, '總分', final)
        stu.append(final)
        results.append(stu)

with open('新學生成績.csv', 'w') as outfile:
    outfile.write(data[0]+'總成績\n')
    for stu in results:
        outfile.write(','.join([str(e) for e in stu])+'\n')
```

輸入檔

	A	B	C	D	E	F
1	姓名	作業一	作業二	作業三	期中考	期末考
2	賴一梅	100	90	100	80	85
3	陳二哥	90	90	100	90	60
4	張三豐	30	40	100	50	70
5	李四瑞	60	70	90	80	90
6	王五郎	99	99	80	99	80
7	錢六順	88	88	100	88	60
8	孫七天	100	100	100	0	100
9	朱八解	100	0	100	100	100

程式執行輸出畫面

```
賴一梅 [100, 90, 100, 80, 85] 作業平均 100.0 總分 91.5
陳二哥 [90, 90, 100, 90, 60] 作業平均 95.0 總分 83.5
張三豐 [30, 40, 100, 50, 70] 作業平均 70.0 總分 66.0
李四瑞 [60, 70, 90, 80, 90] 作業平均 80.0 總分 83.0
王五郎 [99, 99, 80, 99, 80] 作業平均 99.0 總分 93.3
錢六順 [88, 88, 100, 88, 60] 作業平均 94.0 總分 82.6
孫七天 [100, 100, 100, 0, 100] 作業平均 100.0 總分 80.0
朱八解 [100, 0, 100, 100, 100] 作業平均 100.0 總分 100.0
```

輸出檔

	A	B	C	D	E	F	G
1	姓名	作業一	作業二	作業三	期中考	期末考	總成績
2	賴一梅	100	90	100	80	85	91.5
3	陳二哥	90	90	100	90	60	83.5
4	張三豐	30	40	100	50	70	66
5	李四瑞	60	70	90	80	90	83
6	王五郎	99	99	80	99	80	93.3
7	錢六順	88	88	100	88	60	82.6
8	孫七天	100	100	100	0	100	80
9	朱八解	100	0	100	100	100	100

程式碼解說：讀取資料

```
01 results = []
02 with open('學生成績.csv') as infile:
03     data = infile.read().split()
04     for e in data[1:]:
05         stu = e.split(',')
06         scores = [int(sc) for sc in stu[1:]]
```

```
1 姓名,作業一,作業二,作業三,期中考,期末考
2 賴一梅,100,90,100,80,85
3 陳二哥,90,90,100,90,60
4 張三豐,30,40,100,50,70
5 李四瑞,60,70,90,80,90
6 王五郎,99,99,80,99,80
7 錢六順,88,88,100,88,60
8 孫七天,100,100,100,0,100
9 朱八解,100,0,100,100,100
```

- ▶ **CSV** 是常用的文字資料檔案格式，其以逗點「,」隔開資料項。程式中需要先開啟文字檔案後，透過指令將檔案內的資料讀入變數內，才能進一步處理。
 - 我們曾於第三單元講過以 `csv` 模組的函式來讀寫 **CSV** 格式檔案，本單元中我們則教大家如何不透過 `csv` 模組來處理。
- ▶ 第 **02** 行 `open()` 函式用來開啟檔案，我們需要將欲開啟的檔案名稱放在 `()` 裡面，並且用一對單引號（雙引號亦可）將名稱包起來，代表這是一個字串。這個檔案在此程式將會以變數 `infile` 來呈現使用。
- ▶ 第 **03** 行對 `infile` 呼叫 `read()` 函式，將檔案內所有資料全部讀出，同時透過函式 `split()` 將文字切斷，成為列表 `data`。
- ▶ 這兩行程式將學生成績 **CSV** 檔案內的資料，逐行讀入存成列表變數 `data`。

程式碼解說：迴圈控制

```
01 results = []
02 with open('學生成績.csv') as infile:
03     data = infile.read().split()
04     for e in data[1:]:
05         stu = e.split(',')
06         scores = [int(sc) for sc in stu[1:]]
```

```
1 姓名,作業一,作業二,作業三,期中考,期末考
2 賴一梅,100,90,100,80,85
3 陳二哥,90,90,100,90,60
4 張三豐,30,40,100,50,70
5 李四瑞,60,70,90,80,90
6 王五郎,99,99,80,99,80
7 錢六順,88,88,100,88,60
8 孫七天,100,100,100,0,100
9 朱八解,100,0,100,100,100
```

- ▶ 第 04 行宣告此為後面程式碼的迴圈設定，代表從列表 data 的第二筆，也就是 CSV 檔案內標題列除外，所有的資料中取出每筆資料。後面的迴圈中，變數 e 就是每一筆的成績資料。
- ▶ 第 05 行使用 split() 函式，將每一筆資料以逗點為分割點切割（因為「學生成績.csv」檔案內的資料分隔是逗點）。變數 stu 為包含一筆學生成績的列表資料。
- ▶ 第 06 行以下於後面課程說明。需要特別注意的是，第 06 行以下的指令都需要有相同的對齊縮排，這樣才會隸屬於相同的迴圈範圍。

程式概念：資料型態轉換

▶ 資料型態可分成兩大類

- 數字：123、456 可以加減乘除等計算，進一步又分成整數與小數。
- 字串：'123'、'456'、'abc' 各種符號各種文字。

▶ Python 提供多種字串處理的函式

- 成績是數字，但是在文字檔內讀取出來的時候，會以字串的方式記錄。
- 需要先將成績文字轉換成整數，才能進行數字運算。
- Ex: `int('123') → 123`，反之若想把數字轉換成字串則可以使用 `str(123) → '123'`

程式概念：列表生成

```
01 a = [10, 20, 30]
02 b = [e+1 for e in a]
03 c = [2*e+3 for e in a]
04 print(a, b, c)
```

[10, 20, 30] [11, 21, 31] [23, 43, 63]

- ▶ 列表生成式 (list comprehension) 是 Python 程式中常見用來建立列表的方法，簡單來說，是對一個舊列表內的每個元素進行指定運算，並以運算後的結果作為新列表的內容，語法像是

新列表 = [**運算式** for e in **舊列表**]

- 程式碼中的 e 可以取別的名字，但注意運算式中也要跟著使用新名字。
- ▶ 第 02、03 行我們用列表 a 的三個整數，對其作「加一」和「兩倍加三」的運算，然後生成了新的列表 b 和 c。

程式概念：列表生成

```
01 x = ['an', 'bo', 'ca']
02 y = [e+'t' for e in x]
03 print(x, y)
04
05 numstr = ['90', '95', '100']
06 num = [int(e) for e in numstr]
07 print(num)
```

```
['an', 'bo', 'ca'] ['ant', 'bot', 'cat']
[90, 95, 100]
```

- ▶ 列表生成式 (list comprehension) 是 Python 程式中常見用來建立列表的方法，簡單來說，是對一個舊列表內的每個元素進行指定運算，並以運算後的結果作為新列表的內容，語法像是

新列表 = [**運算式** for e in **舊列表**]

- 程式碼中的 e 可以取別的名字，但注意運算式中也要跟著使用新名字。
- ▶ 列表中的資料不一定是數字，也可能是文字。如上方程式中第 01 行和第 05 行的列表 x 及 numstr。
- ▶ 第 02 行我們對 x 中所有的字串在尾端加上 't'，然後以這些新字串建立新列表 y。
- ▶ 第 06 行我們將 numstr 中所有的字串轉換成整數，然後以這些新整數建立新列表 num。

程式碼解說：資料處理統計

	A	B	C	D	E	F
1	姓名	作業一	作業二	作業三	期中考	期末考
2	賴一梅	100	90	100	80	85
3	陳二哥	90	90	100	90	60
4	張三豐	30	40	100	50	70
5	李四瑞	60	70	90	80	90
6	王五郎	99	99	80	99	80
7	錢六順	88	88	100	88	60
8	孫七天	100	100	100	0	100
9	朱八解	100	0	100	100	100

```
01 results = []
02 with open('學生成績.csv') as infile:
03     data = infile.read().split()
04     for e in data[1:]:
05         stu = e.split(',')
06         scores = [int(sc) for sc in stu[1:]]
07         exams = scores[3]*0.2 + scores[4]*0.3
08         hw = sum(sorted(scores[0:3])[1:])/2
09         final = round(hw*0.5 + exams,2)
10         print(stu[0], scores, '作業平均', hw, '總分', final)
11         stu.append(final)
12     results.append(stu)
```

```
1 姓名,作業一,作業二,作業三,期中考,期末考
2 賴一梅,100,90,100,80,85
3 陳二哥,90,90,100,90,60
4 張三豐,30,40,100,50,70
5 李四瑞,60,70,90,80,90
6 王五郎,99,99,80,99,80
7 錢六順,88,88,100,88,60
8 孫七天,100,100,100,0,100
9 朱八解,100,0,100,100,100
```

- ▶ 學會列表生成式後，現在我們能理解第 06 行程式意義了。
- ▶ 假設 e 代表檔案第二行，即 '賴一梅,100,90,100,80,85'。
第 05 行對 e 作 split() 後，stu 代表的資料是 ['賴一梅','100','90','100','80','85']。
第 06 行取出 stu[1:]，即取出 ['100','90','100','80','85']；接著利用列表生成式，以原本的字串列表建立一個整數列表 [100,90,100,80,85]，以 scores 代表之。
- ▶ 第 07 行取出 scores[3] 和 scores[4]，就是第四、第五個分數，代表期中考和期末考分數，各佔 20% 和 30%。

程式概念：統計函式的運用

```
01 num = [4, 2, 2, 3, 7, 2]
02 if len(num)>0:
03     avg = sum(num)/len(num)
04 print(avg, round(avg,1), round(avg,2))
05
06 print(max(num), min(num), num.count(2))
07 snum = sorted(num)
08 print(num, snum)
```

```
3.3333333333333335 3.3 3.33
7 2 3
[4, 2, 2, 3, 7, 2] [2, 2, 2, 3, 4, 7]
```

- ▶ 對於數值資料，我們常計算的統計量包含平均、最大、最小和個數。Python 中都有內建的函式可供使用。
- ▶ 上方程式第 01 行建立一個包含六個整數的列表 num。第 02 行我們使用到函式 len() 去計算列表中的資料個數，此處為 6。
- ▶ 第 03 行我們以 sum() 函式計算 num 中數值的總和，然後除以 num 中數值的個數，即得平均。第 04 行我們示範了如何用 round() 函式取到小數點後的指定位數。
- ▶ 第 06 行我們示範 max() 和 min() 可以計算列表中最大和最小值，count() 則可以計算某個數值出現的次數。注意應用 count() 的語法和 max()/min() 不同。
- ▶ 第 07 行示範如何以 sorted() 函式得到一個由小到大排序的列表。

程式概念：統計函式的運用

```
01 data = ['賴一梅', 90, 80, 100]
02 snum = sorted(data[1:])
03 high2 = snum[1:]
04 avg = sum(high2)/2
05 print(data[1:], snum, high2, avg)
06
07 print(sum(sorted(data[1:])[1:])/2)
```

```
[90, 80, 100] [80, 90, 100] [90, 100] 95.0
95.0
```

- ▶ 本單元的其中一個核心任務是取出學生三個作業分數中較高的兩個來算平均，本頁我們作個簡易的範例。
- ▶ 第 01 行建立一個列表 `data`，裡面有學生姓名和三個分數。
- ▶ 第 02 行我們取出 `data[1:]`，即列表自第二個資料項開始的所有資料 `[90, 80, 100]`。接著以 `sorted()` 排序，令排序結果為列表 `snum`。因為 `sorted()` 是由小到大排序，`snum` 為 `[80, 90, 100]`。
- ▶ 第 03 行我們取出 `snum[1:]`，令其為 `high2`，代表列表 `[90, 100]`。第 04 行計算 `high2` 的總和再除以 2，便得到兩個最高分數的平均。
- ▶ 第 07 行為前面幾個步驟合併為一行的寫法。

程式碼解說：資料處理統計

	A	B	C	D	E	F
1	姓名	作業一	作業二	作業三	期中考	期末考
2	賴一梅	100	90	100	80	85
3	陳二哥	90	90	100	90	60
4	張三豐	30	40	100	50	70
5	李四瑞	60	70	90	80	90
6	王五郎	99	99	80	99	80
7	錢六順	88	88	100	88	60
8	孫七天	100	100	100	0	100
9	朱八解	100	0	100	100	100

```
01 results = []
02 with open('學生成績.csv') as infile:
03     data = infile.read().split()
04     for e in data[1:]:
05         stu = e.split(',')
06         scores = [int(sc) for sc in stu[1:]]
07         exams = scores[3]*0.2 + scores[4]*0.3
08         hw = sum(sorted(scores[0:3])[1:])/2
09         final = round(hw*0.5+exams, 2)
10         print(stu[0], scores, '作業平均', hw, '總分', final)
11         stu.append(final)
12     results.append(stu)
```

```
1 姓名,作業一,作業二,作業三,期中考,期末考
2 賴一梅,100,90,100,80,85
3 陳二哥,90,90,100,90,60
4 張三豐,30,40,100,50,70
5 李四瑞,60,70,90,80,90
6 王五郎,99,99,80,99,80
7 錢六順,88,88,100,88,60
8 孫七天,100,100,100,0,100
9 朱八解,100,0,100,100,100
```

- ▶ 假設 `e` 代表檔案第二行，即 '賴一梅,100,90,100,80,85'。
第 05 行對 `e` 作 `split()` 後，`stu` 代表的資料是 ['賴一梅','100','90','100','80','85']。
第 06 行取出 `stu[1:]`，即取出 ['100','90','100','80','85']；接著利用列表生成式，以原本的字串列表建立一個整數列表 [100,90,100,80,85]，以 `scores` 代表之。
- ▶ 現在我們了解第 08 行的意義是將 `scores[0:3]`（即三個作業分數）以 `sort()` 由小到大排序，然後取出後兩個分數（即最高的兩個分數），以 `sum()` 計算這兩個分數總和再除以二求平均。

程式概念：新增列表資料

```
01 a = []
02 a.append(10)
03 a.append(20)
04 print(a)
05
06 b = [66, 88, 99]
07 data = []
08 data.append(a)
09 data.append(b)
10 for e in data:
11     print(e)
```

```
[10, 20]
[10, 20]
[66, 88, 99]
```

- ▶ 除了使用列表生成式，另一種建立列表的方式是一筆一筆地將資料加入到列表中，`append()` 函式便是這樣的功能。
- ▶ 上方程式第 01 行我們建立一個空列表 `a`，第 02 行我們將 10 加入 `a`，第 03 行將 20 加入 `a`。從第 04 行的結果來看，我們知道 `append()` 會把新資料放在列表的最後。
- ▶ 列表內的資料本身也可以是列表型態。第 06-09 行我們示範把兩個列表 `a` 和 `b` 加入到列表 `data` 中。第 10-11 行我們將 `data` 中的資料逐一取出列印，可以看到其中的內容正是我們加入的列表 `a` 和 `b`。

程式概念：新增列表資料

```
01 lai = '賴一梅,90,80'  
02 chen = '陳二哥,70,100'  
03 data = [lai, chen]  
04 result = []  
05 for e in data:  
06     stu = e.split(',')  
07     stu.append(int(stu[2])+1)  
08     result.append(stu)  
09  
10 for stu in result:  
11     print(stu)
```

```
['賴一梅', '90', '80', 81]  
['陳二哥', '70', '100', 101]
```

- ▶ 我們再看一個複雜一點點的例子。
- ▶ 第 03 行建立列表 `data` 內含兩個字串，`result` 是個空列表。
- ▶ 第 05-07 行，我們將 `data` 中的資料逐一取出，令 `e` 代表之。我們對 `e` 作 `split()`，切分得到的列表稱為 `stu`。然後我們將 `stu[2]` 轉成整數的值加一，將此值加到 `stu` 中。
 - 若 `e` 代表 `data[0]`，也就是 `lai` 的內容 '賴一梅, 90, 80'。
`stu` 為 ['賴一梅', '90', '80']，`stu[2]` 為 '80'。
將 '80' 轉為整數得到 80，再加一的結果到 81。將此值加到 `stu` 後，`stu` 會變成 ['賴一梅', '90', '80', 81]。

程式碼解說：資料處理統計

```
01 results = []
02 with open('學生成績.csv') as infile:
03     data = infile.read().split()
04     for e in data[1:]:
05         stu = e.split(',')
06         scores = [int(sc) for sc in stu[1:]]
07         exams = scores[3]*0.2 + scores[4]*0.3
08         hw = sum(sorted(scores[0:3])[1:])/2
09         final = round(hw*0.5+exams, 2)
10         print(stu[0], scores, '作業平均', hw, '總分', final)
11         stu.append(final)
12     results.append(stu)
```

	A	B	C	D	E	F
1	姓名	作業一	作業二	作業三	期中考	期末考
2	賴一梅	100	90	100	80	85
3	陳二哥	90	90	100	90	60
4	張三豐	30	40	100	50	70
5	李四瑞	60	70	90	80	90
6	王五郎	99	99	80	99	80
7	錢六順	88	88	100	88	60
8	孫七天	100	100	100	0	100
9	朱八解	100	0	100	100	100

```
1 姓名,作業一,作業二,作業三,期中考,期末考
2 賴一梅,100,90,100,80,85
3 陳二哥,90,90,100,90,60
4 張三豐,30,40,100,50,70
5 李四瑞,60,70,90,80,90
6 王五郎,99,99,80,99,80
7 錢六順,88,88,100,88,60
8 孫七天,100,100,100,0,100
9 朱八解,100,0,100,100,100
```

- ▶ 列表 `stu` 是切割好的原始成績資料，包含姓名、三次作業成績、兩次考試成績。
- ▶ 第 11 行使用函式 `append()`，幫 `stu` 增加一個資料項 `final`，即總平均分數。第 10 行同樣使用函式 `append()`，將剛剛的 `stu` 整筆資料，加入列表 `results` 內成為新增的一筆資料。
- ▶ 這兩行程式同樣都使用列表的函式 `append()`，但放入的內容的資料型態不同：第 11 行放入的是一個數值，第 12 行放入的是一個列表。

程式碼解說：輸出檔案

```
01 results = []
02 with open('學生成績.csv') as infile:
03     data = infile.read().split()
04     for e in data[1:]:
05         stu = e.split(',')
           [省略]
11         stu.append(final)
12         results.append(stu)
13
14 with open('新學生成績.csv', 'w') as outfile:
15     outfile.write(data[0]+'總成績\n')
16     for stu in results:
17         outfile.write(','.join([str(e) for e in stu])+'\n')
```

1	姓名,作業一,作業二,作業三,期中考,期末考
2	賴一梅,100,90,100,80,85
3	陳二哥,90,90,100,90,60
4	張三豐,30,40,100,50,70
5	李四瑞,60,70,90,80,90
6	王五郎,99,99,80,99,80
7	錢六順,88,88,100,88,60
8	孫七天,100,100,100,0,100
9	朱八解,100,0,100,100,100

1	姓名,作業一,作業二,作業三,期中考,期末考,總成績
2	賴一梅,100,90,100,80,85,91.5
3	陳二哥,90,90,100,90,60,83.5
4	張三豐,30,40,100,50,70,66.0
5	李四瑞,60,70,90,80,90,83.0
6	王五郎,99,99,80,99,80,93.3
7	錢六順,88,88,100,88,60,82.6
8	孫七天,100,100,100,0,100,80.0
9	朱八解,100,0,100,100,100,100.0

- ▶ 最後我們要把加上總平均分數的結果 `results` 輸出到一個新檔案。
- ▶ 第 14 行使用 `open()` 函式開啟「新學生成績.csv」這個檔案，但是這個檔案目標是當作「輸出檔」，準備寫入資料用的，所以要加上 `'w'` 這個參數。我們使用 `outfile` 這個變數來代表這個檔案。
- ▶ 第 15 行對 `outfile` 呼叫函式 `write()`，以列表 `data` 內第一筆資料的鍵 (keys) 作為這個 CSV 輸出檔的第一行標題。
- ▶ 第 16-17 行將列表 `results` 內每一筆資料，逐項寫入 CSV 檔案內 (也就是 `outfile` 所代表的檔案)。

程式概念：字串合併

```
01 s = '10'  
02 t = '11'  
03 u = '12'  
04 x = s+'/' +t+'/' +u  
05 print(s, t, u, '|', x)  
06  
07 data = [s, t, u]  
08 y = '/'.join(data)  
09 print(y)  
10  
11 person = ['賴一梅', '90', '80', '85.0']  
12 z = ','.join(person)  
13 print(z)
```

```
10 11 12 | 10/11/12  
10/11/12  
賴一梅,90,80,85.0
```

- ▶ 前一個單元我們學過用運算 + 可以作字串的串接。
- ▶ 第 04 行我們將五個字串以 + 串接起來。(s, t, u 三個字串和兩個 ';' 字串，共五個。)
- ▶ 當我們想要把多個字串串接起來，且以一個特定字串作為間隔時，我們可以使用 join() 函式。第 08 行示範以 ',' 將 '10'、'11' 和 '12' 連起來，變成 '10,11,12'。
- ▶ 第 12 行示範將四個字串以 ',' 串接起來。要注意，使用 join() 時，列表內「所有」的資料都必須是字串型態。

程式碼解說：輸出檔案

```
01 results = []
02 with open('學生成績.csv') as infile:
03     data = infile.read().split()
04     for e in data[1:]:
05         stu = e.split(',')
           [省略]
11         stu.append(final)
12         results.append(stu)
13
14 with open('新學生成績.csv', 'w') as outfile:
15     outfile.write(data[0]+'總成績\n')
16     for stu in results:
17         outfile.write(','.join([str(e) for e in stu])+'\n')
```

1	姓名,作業一,作業二,作業三,期中考,期末考
2	賴一梅,100,90,100,80,85
3	陳二哥,90,90,100,90,60
4	張三豐,30,40,100,50,70
5	李四瑞,60,70,90,80,90
6	王五郎,99,99,80,99,80
7	錢六順,88,88,100,88,60
8	孫七天,100,100,100,0,100
9	朱八解,100,0,100,100,100

1	姓名,作業一,作業二,作業三,期中考,期末考,總成績
2	賴一梅,100,90,100,80,85,91.5
3	陳二哥,90,90,100,90,60,83.5
4	張三豐,30,40,100,50,70,66.0
5	李四瑞,60,70,90,80,90,83.0
6	王五郎,99,99,80,99,80,93.3
7	錢六順,88,88,100,88,60,82.6
8	孫七天,100,100,100,0,100,80.0
9	朱八解,100,0,100,100,100,100.0

▶ 第 16-17 行，我們逐一取出 `results` 中的每個學生資料列表，以列表生成式建立一個全數為字串的列表，然後以 `join()` 函式產生一個字串來寫入。

- 我們在第 11 行加入總平均成績 `final` 到 `stu` 列表中，注意這個 `final` 是個小數，不是字串。
- 因此我們在第 17 行需要一個列表生成式來建立一個全為字串的列表。假設 `stu` 原為 `['賴一梅', '100', '90', '100', '80', '85', 91.5]`，則列表生成式會產生 `['賴一梅', '100', '90', '100', '80', '85', 91.5]`，`join()` 後的結果為 `'賴一梅,100,90,100,80,85,91.5'`，最後加上一個換行，傳給 `write()` 函式寫入檔案。

成績計算

完整程式展示

```
results = []
with open('學生成績.csv') as infile:
    data = infile.read().split()
    for e in data[1:]:
        stu = e.split(',')
        scores = [int(sc) for sc in stu[1:]]
        exams = scores[3]*0.2 + scores[4]*0.3
        hw = sum(sorted(scores[0:3])[1:])/2
        final = round(hw*0.5+exams, 2)
        print(stu[0], scores, '作業平均', hw, '總分', final)
        stu.append(final)
        results.append(stu)

with open('新學生成績.csv', 'w') as outfile:
    outfile.write(data[0]+'總成績\n')
    for stu in results:
        outfile.write(','.join([str(e) for e in stu])+'\n')
```

輸入檔

	A	B	C	D	E	F
1	姓名	作業一	作業二	作業三	期中考	期末考
2	賴一梅	100	90	100	80	85
3	陳二哥	90	90	100	90	60
4	張三豐	30	40	100	50	70
5	李四瑞	60	70	90	80	90
6	王五郎	99	99	80	99	80
7	錢六順	88	88	100	88	60
8	孫七天	100	100	100	0	100
9	朱八解	100	0	100	100	100

程式執行輸出畫面

```
賴一梅 [100, 90, 100, 80, 85] 作業平均 100.0 總分 91.5
陳二哥 [90, 90, 100, 90, 60] 作業平均 95.0 總分 83.5
張三豐 [30, 40, 100, 50, 70] 作業平均 70.0 總分 66.0
李四瑞 [60, 70, 90, 80, 90] 作業平均 80.0 總分 83.0
王五郎 [99, 99, 80, 99, 80] 作業平均 99.0 總分 93.3
錢六順 [88, 88, 100, 88, 60] 作業平均 94.0 總分 82.6
孫七天 [100, 100, 100, 0, 100] 作業平均 100.0 總分 80.0
朱八解 [100, 0, 100, 100, 100] 作業平均 100.0 總分 100.0
```

輸出檔

	A	B	C	D	E	F	G
1	姓名	作業一	作業二	作業三	期中考	期末考	總成績
2	賴一梅	100	90	100	80	85	91.5
3	陳二哥	90	90	100	90	60	83.5
4	張三豐	30	40	100	50	70	66
5	李四瑞	60	70	90	80	90	83
6	王五郎	99	99	80	99	80	93.3
7	錢六順	88	88	100	88	60	82.6
8	孫七天	100	100	100	0	100	80
9	朱八解	100	0	100	100	100	100

挑戰時刻

- ▶ 挑戰一：若採計所有作業成績，結果如何？請你將兩種計算方式的總平均分數都寫入到新成績檔案，以便老師決定是否只採計兩次最高作業。
- ▶ 挑戰二：用試算表軟體開啟輸入檔，加入兩欄分數，使得每位學生共有五次作業分數。作業成績採計五次中最高的三次來計算學期總平均。請你修改程式以達成這樣的功能。
- ▶ 挑戰三：由使用者決定作業、期中考和期末考的比重。
 - 下面的程式碼示範如何讀取使用者輸入的比重並作計算。

```
hw = 90
exam = 80
hw_weight = float(input('請輸入作業比重'))
final = hw*hw_weight + exam*(1-hw_weight)
print(hw, exam, final)
```

```
請輸入作業比重0.4
90 80 84.0
```