

# 總結：實踐題目

## 請打開Thonny一起做

### Python 編程課程 第 6 課

- 成績計分和零用錢
- 水果清單
- 包剪揅
- 個人檔案讀取

# 練習一：成績計分和零用錢 (變量、條件判斷)

# 練習一：成績計分和零用錢

## 情境：

數學考試，85分或以上為A；70至84分為B；60至69分為C；60分以下不合格”Fail”；101分為SSR。

## 涉及知識：

- 變量 Variable
- 條件判斷 Conditionals

## 你現有\$100，家裡說：

如果你考獲A就給你加\$100；B就加\$50；C就不加不減；不合格就扣\$50；SSR就給你十倍。



# 練習一：成績計分和零用錢

請編寫程式，建立grade、money、score三個變量

分數	成績等級	後果	程式碼(你來做)
85-100	A	加\$100	grade = 'U' # 初始賦值 U=unknown未知 money = 100
70-84	B	加\$50	score = _____? _____ # 記得使用input()和int()
60-69	C	無	# if-elif-else 判斷score落入範圍 # 賦值給grade # 計算money
0-59	Fail	扣\$50	
101	SSR	加十倍	# 列印結果



# 練習一：成績計分和零用錢

**粉紅色斜體**: 題目

**黑色正常字體**: 提供的程式碼

**紅色底線**: 請填入程式碼

## 程式碼

### 下載:

[L6 - Grading \(variable, conditionals\).py](#)

```

1 # L6 - Grading (variable, conditionals)
2 # L6 - 成績計分和零用錢 (變量, 條件判斷)
3
4 grade = 'U'      # 初始賦值 U=unknown未知
5 money = 100
6 score = _____ # 使用input()和int()
7
8 # if-elif-else 判斷score落入範圍
9 # 賦值給grade
10 # 計算money
11
12 # 列印結果

```

## 輸出

You score:89  
 Your grade is: A  
 and you receive: \$ 200

<https://drive.google.com/file/d/1-MIv2ziQYB3Eut5fcuN-UFdGeoyF79WZ/view?usp=sharing>

# 練習二：水果清單 (函數、模組化設計、While循環)

# 練習二：水果清單

## 情境：

編寫三個小程序：

- 撰寫水果清單
- 修改庫存
- 顯示水果及其庫存量

## 涉及知識：

- 函數 Function
- 模組化設計 Modularization
- While循環

主程式	輸出
fruit_list()	Input fruit ("n" to quit):
fruit_count()	Input fruit count:
show_fruit()	We have __ <fruit_name>



# 練習二：水果清單

大程式	輸出
<pre> fruits = fruit_list() count = fruit_count(fruits) show_fruit(fruits, count) </pre>	<p>New fruit ("n" to quit): apple  New fruit ("n" to quit): banana  New fruit ("n" to quit): cherry  New fruit ("n" to quit): n  What is the count for apple?  Input: 2  What is the count for banana?  Input: 4  What is the count for cherry?  Input: 29  We have 2 apple.  We have 4 banana.  We have 29 cherry.</p>



# 練習二：水果清單

**粉紅色斜體**: 題目

**黑色正常字體**: 提供的程式碼

**紅色底線**: 請填入程式碼

## 程式碼

### 下載:

[L6 - Fruit list \(functions, modularization, while-loop\).py](#)

```

1 # L6 - Fruit list (functions, modularization, while-loop)
2 # L6 - 水果清單 (函數, 模組化設計, While循環)
3
4 def fruit_list():
5     # 定義水果清單fruits
6     # 接收第一個input()為新增水果
7     # 進入循環, 如果input不是"n"就繼續循環
8     # 新增新水果 (提示: .append())
9     # 接收下一個水果的input()
10    # 回傳水果清單
11
12 def fruit_count(fruits):
13     # 定義數量清單count
14     # 定義清單長度, 等同fruit (提示: len())
15     # 進入循環
16     # 方印日左些印T百M甲

```

## 輸出

New fruit ("n" to quit): apple

New fruit ("n" to quit): banana

New fruit ("n" to quit): n

What is the count for apple?

Input: 2

What is the count for banana?

Input: 4

We have 2 apple.

We have 4 banana.

[https://drive.google.com/file/d/1Z93lb0RODRZIsUVCrJQWCO4mHAvFaCir/view?usp=drive\\_link](https://drive.google.com/file/d/1Z93lb0RODRZIsUVCrJQWCO4mHAvFaCir/view?usp=drive_link)

# 練習三：包剪揀 (匯入模組、random、字典、模組化設計)

# 練習三：包剪揀

## 情境：

你和電腦玩包剪揀，你可以輸入1、2、3

- 1: Rock(揀／錘)
- 2: Paper(包／布)
- 3: Scissors(剪／剪刀)

## 涉及知識：

- 匯入模組 import library
- random
- 字典 Dictionary
- 模組化設計 Modularization

## 電腦：

電腦會隨機出招，可以使用random模組為電腦出招(1、2、3)



# 練習三：包剪揀

大程式	輸出
<pre># 定義變量you和cpu # 調用human_move()和cpu_move(), 直接用其回傳賦 值 you = human_move() cpu = cpu_move() # 調用決鬥函數 match(you, cpu)</pre>	<p>1: Rock      2: Paper      3: Scissors      Your move: 1      You go Rock, Computer goes Scissors      You win</p>



# 練習三：包剪揀

**粉紅色斜體**: 題目

**黑色正常字體**: 提供的程式碼

**紅色底線**: 請填入程式碼

## 程式碼

### 下載:

[L6 - Rock Paper Scissors \(import module, modularization\).py](#)

```

1 # L6 - Rock Paper Scissors (import module, random, dictionary, modularization)
2 # L6 - 包剪揀 (匯入模組, random, 字典, 模組化設計)
3
4
5 # 汇入 random 模組
6 _____ random
7
8
9 # 函數human_move(): 玩家輸入
10 def human_move():
11     print("1: Rock")
12     print("2: Paper")
13     print("3: Scissors")
14     # 使用input()接收玩家輸入, int()轉換成數字
15     move = _____
16     # 回傳move
17     return _____

```

## 輸出

1: Rock

2: Paper

3: Scissors

Your move: 1

You go Rock, Computer goes Scissors

You win

[https://drive.google.com/file/d/1\\_2Ezlj4YgydnIc6nER6YK4i\\_rM9YZskk/view?usp=drive\\_link](https://drive.google.com/file/d/1_2Ezlj4YgydnIc6nER6YK4i_rM9YZskk/view?usp=drive_link)

# 練習四：個人檔案讀取 (文字檔案處理、字串方法、冒泡排序法)

# 練習四：個人檔案讀取

## 情境：

你有些個人文字檔案，內含性別、年齡等信息。

每一行信息都有**欄位內容**和**信息內容**

請移除**欄位內容**

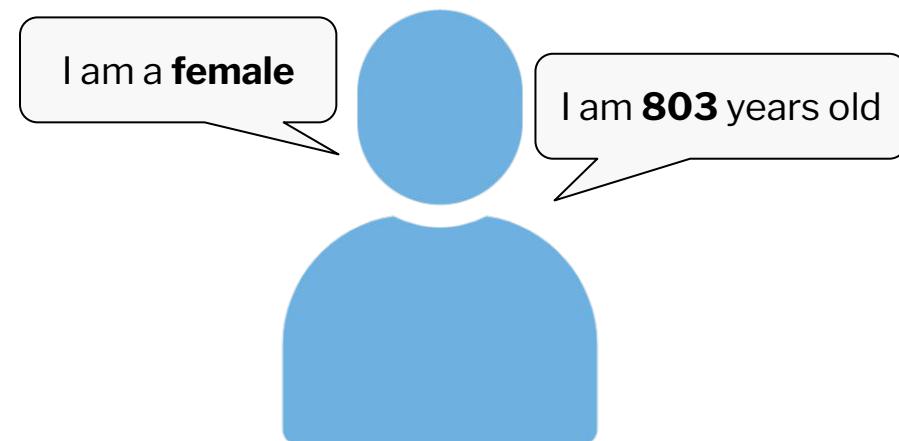
## 涉及知識：

- 文字檔案處理 File IO
- 字串方法 String method
- 冒泡排序法 Bubble sort

## 例子：

Daisy

- Gender: Female
- Age: 803



# 第一步:.txt 檔案

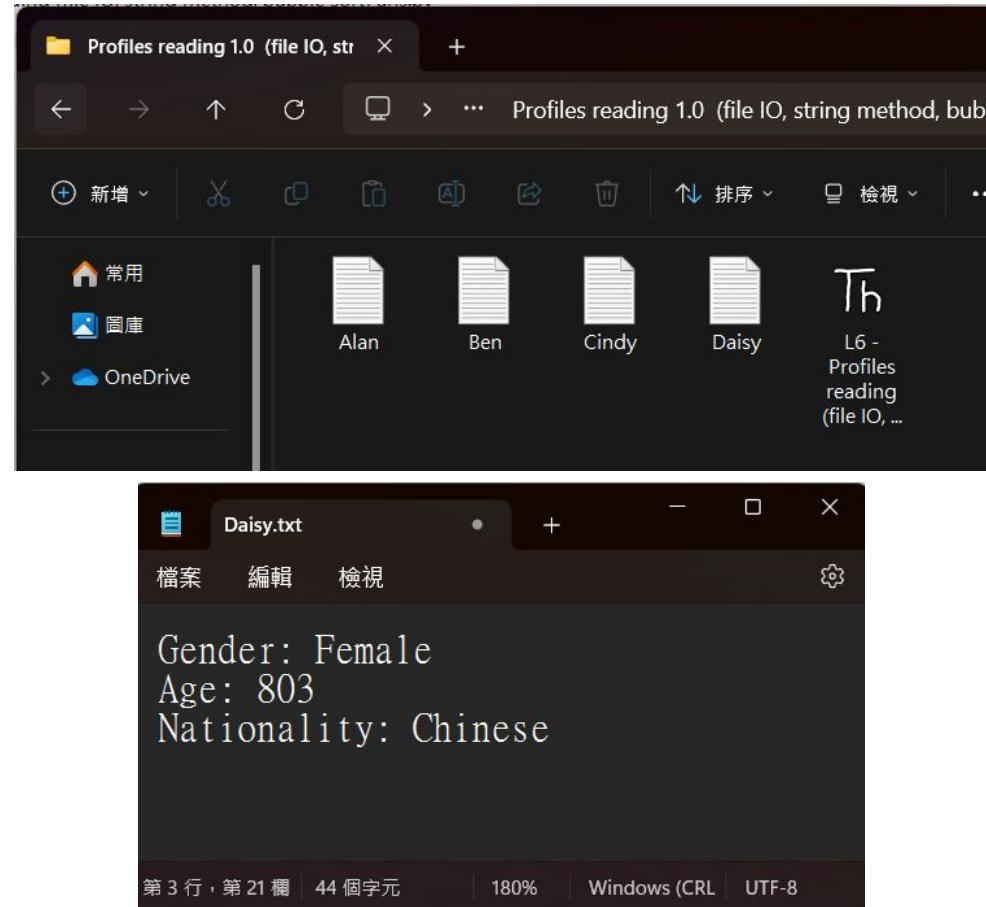
1. 下載文件夾 **Profiles reading 1.0 (file IO, string method, bubble sort)**

裡面應有：

- Python file (Thonny圖標)
- 個人文字檔案 (.txt)

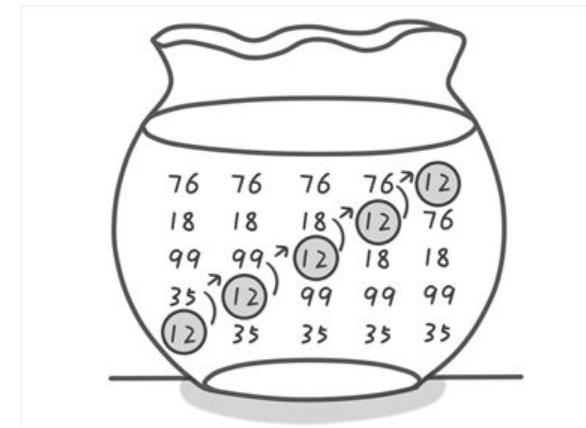
2. 打開其中一人的.txt, 應出現右邊內容

- 性別(Gender)
- 年齡(Age)
- 國籍(Nationality) (目前不用)



# 第二步: 按年齡由小至大排序

原排序	按年齡小至大排序
Original: ['Alan', 'Ben', 'Cindy', 'Daisy'] ['Male', 'Male', 'Female', 'Female'] [22, 92, 11, 803]	Sorted by age in ascending: ['Cindy', 'Alan', 'Ben', 'Daisy'] [11, 22, 92, 803] ['Female', 'Male', 'Male', 'Female']



# 練習四：個人檔案讀取

粉紅色斜體:題目

黑色正常字體:提供的程式碼

紅色底線:請填入程式碼

## 程式碼

打開:

L6 - Profiles reading (file IO, string method, bubble sort).py

```

1 # L6 - Profiles reading (file IO, string method, bubble sort)
2 # L6 - 個人檔案讀取 (文字檔案處理, 字串方法, 冒泡排序法)
3
4 # 初始化變量
5 names = ['Alan', 'Ben', 'Cindy', 'Daisy']
6 genders = []
7 ages = []
8
9
10 """1.0: 使用open()和.close()"""
11 # 讀取Alan的檔案
12 file = open('Alan.txt', 'r')
13 # 使用.readline()讀取性別內容
14 gender = file.readline()
15 # 使用.strip()刪除無關內容
16 gender = gender.strip('Gender: ')
17 gender = gender.strip('\n')
18 # 使用.append()添加gender到genders列表

```

## 輸出

Original:

['Alan', 'Ben', 'Cindy', 'Daisy']  
 ['Male', 'Male', 'Female', 'Female']  
 [22, 92, 11, 803]

Sorted by age in ascending:

['Cindy', 'Alan', 'Ben', 'Daisy']  
 [11, 22, 92, 803]  
 ['Female', 'Male', 'Male', 'Female']

<https://drive.google.com/drive/folders/1O2n9zALegdJ2YTn6RTZkC7jLZ4Rz3GEo?usp=sharing>

# 練習四：個人檔案讀取 2.0 (模組化設計)

# 練習四：個人檔案讀取 2.0

同一個情境：

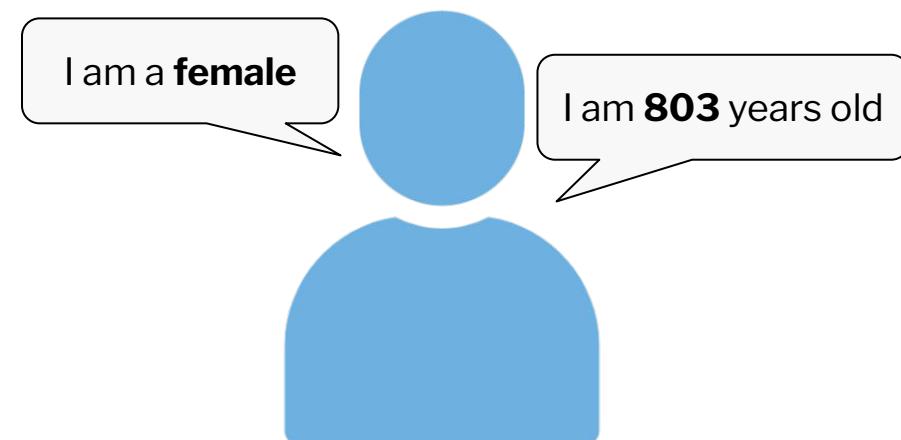
將不同程式碼 封裝成一個個函數

調用函數／使用回傳賦值

涉及知識：

- 模組化設計

函數	作用
trimG(gender)	刪去性別的欄位內容
trimA(age)	刪去年齡的欄位內容
swap(list, large, small)	在冒泡排序中交換一個列表中的前後兩個項目位置



# 練習四：個人檔案讀取 2.0

粉紅色斜體:題目

黑色正常字體:提供的程式碼

紅色底線:請填入程式碼

## 程式碼

打開:

L6 - Profiles reading (modularization).py

```

1 # L6 - Profiles reading 2.0 (modularization)
2 # L6 - 個人檔案讀取 2.0 (模組化設計)
3
4 # 初始化變量
5 names = ['Alan', 'Ben', 'Cindy', 'Daisy']
6 genders = []
7 ages = []
8
9 # trimG():將性別一欄無關的資訊刪掉
10 def trimG(gender):
11     # 使用.strip()移除不要的字符
12     gender = gender.strip('Gender: ')
13     gender = gender.___('\n')
14     return ___
15
16 # trimA():將年齡一欄無關的資訊刪掉+轉換成int()

```

## 輸出

Original:

['Alan', 'Ben', 'Cindy', 'Daisy']

['Male', 'Male', 'Female', 'Female']

[22, 92, 11, 803]

Sorted by age in ascending:

['Cindy', 'Alan', 'Ben', 'Daisy']

[11, 22, 92, 803]

['Female', 'Male', 'Male', 'Female']

[https://drive.google.com/drive/folders/16nBzAqwevl3Sl-jNuiWMGN0g\\_eOhLlyv?usp=sharing](https://drive.google.com/drive/folders/16nBzAqwevl3Sl-jNuiWMGN0g_eOhLlyv?usp=sharing)

# 總結

變量 Variable

(int, str, bool, list, dict)

條件判斷 Conditionals

(if - elif - else)

循環 Iteration

(for, while)

函數 Function

(def, return)

模組化設計 Modularization

(import <module>)

檢索算法 Searching

(Linear search, binary search)

排序算法 Sorting

(Bubble sort, insertion sort, selection sort)

文字檔案處理 File IO

(open(), close(), with open...as, read, write)



程式碼

## 關鍵字 import

## 模組 random

輸出

## 函數 randint()

程式碼

程式碼









