

## 12 計數迴圈



### 挑戰題目: 自我介紹

請用for迴圈，輸出六位參與者的自我介紹和最喜歡的甜點。

- 六位參與者，他們的名字分別是 Mario, Peach, Luigi, Daisy, Toad和Yoshi。
- 喜歡的甜點依次為Star Pudding, Peach Pie, Popsicles, Honey Cake, Cookies, Jelly Beans。

預期結果如下：

Hi, my name is Mario. My favorite dessert is Star Pudding.

Hi, my name is Peach. My favorite dessert is Peach Pie.

```
1  ## Input Data
2  people = ["Mario", "Peach", "Luigi", "Daisy", "Toad", "Yoshi"]
3  desserts = ["Star Pudding", "Peach Pie", "Popsicles", "Honey Cake", "Cookies", "Jelly Beans"]
4
5  ## Process
6  for index in range(len(people)): # range(0,6,1)
7      name = people[index]
8      dessert = desserts[index]
9      print(f"Hi! My name is {name}. My favorite dessert is {dessert}.")
```

```
Hi! My name is Mario. My favorite dessert is Star Pudding.
Hi! My name is Peach. My favorite dessert is Peach Pie.
Hi! My name is Luigi. My favorite dessert is Popsicles.
Hi! My name is Daisy. My favorite dessert is Honey Cake.
Hi! My name is Toad. My favorite dessert is Cookies.
Hi! My name is Yoshi. My favorite dessert is Jelly Beans.
```

從這個例子中，同學要瞭解的觀念有：

1. 串列可以儲存一系列的資料，不只是純量(單一的數字、文字、布林)。
2. 使用range()函數可以更方便地處理數字串列。
3. 學會for計數迴圈搭配串列的使用方法。
4. 學會for計數迴圈搭配range()函數的使用方法。

## 12-1 計數迴圈與資料容器

### 12-1-1. 資料容器 - 串列

1. 把串列想像成一系列火車，火車有一系列的車廂，每個車廂都可以裝東西(資料)。



## 2. 語法

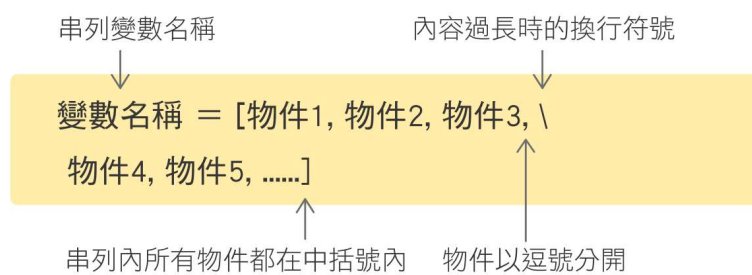


圖 4-2.2 串列的定義

- 串列將資料放在方括號裡面。
- 每個項目都是以逗號分隔。
- 串列項目的索引編號*i*從0開始。
- 索引編號為*i*的資料用 變數名稱[i] 取得。

### 範例1：購物清單

請用串列列出你最喜愛的五樣東西，當成購物清單(shopping\_list)！

```
1  ## 列出購物清單
2  shopping_list = ['牛奶', '蛋', '咖啡豆', '西瓜', '鳳梨']
3  print(shopping_list)
4  ## 取出購物清單上面第三樣物品，也就是索引編號為2的商品咖啡豆
5  shopping_list[2]
```

['牛奶', '蛋', '咖啡豆', '西瓜', '鳳梨']

'咖啡豆'

### 範例2：使用中括號建立串列資料

```
1  ## 建立串列資料
2  ## 數字清單
3  list1 = [1,2,3,4,5] # 數字資料
4  print(list1)
5
6  ## 文字清單
```

```

7 list2 = ["1", "2", "3", "4", "5"] # 文字資料
8 print(list2)
9
10 ## 混合資料型態
11 list3 = [1, 2.0, 'Python', True] # 串列可包含各種資料型態元素
12 print(list3)
13
14
15 ## 使用list()函數建立串列資料
16 list4 = list('python') # 文字字串轉串列清單，將每個字元看成一個單位
17 print(list4)
18
19 list5 = list(12345) # 數字無法轉串列清單，沒有更小的單位
20 print(list5)

```

## 12-1-2 for迴圈

1. for迴圈會逐次取出串列中每一個元素，再對每一個元素進行運算。

### 2. for迴圈語法

for 暫時變數 in 資料容器：  
     程式區塊 （暫時變數逐一取出、逐一處理）

- 每次從資料容器（串列）裡只拿一個東西
- 每個東西都會依序拿一次。
- 暫時變數 = 這次拿出來的東西。
- 執程式區塊
  - 第一行結尾必須加上冒號。
  - 迴圈內所有程式碼必須向右縮四個空格並且對齊。

### 範例3: 使用「for」迴圈讀取串列中的每一個元素

- 暫時變數輪流等於串列中的每個項目值
  - 透過索引值的變動
  - 由不同的索引值，取得串列對應索引項目的值

```

1 ## 1. 文字串列
2 names = ["Allen", "James", "Tom", "Jack"]
3 ## 計數迴圈：把names串列中的東西一個一個直接拿出來
4 for i in names: #每次拿出來的東西暫時叫它i
5     print('Welcome to Class!', i) #清單東西都拿完就結束
6
7 ## 2. 數字串列
8 ## 可用[]，也可用range(資料量大時，通常用range)
9
10 ## 方法(一)：for迴圈 + 資料容器(數字串列)
11 for i in [0,1,2,3,4]:
12     print(i)
13 ## 方法(二)：for迴圈 + range()函數
14 # for i in range(5): # range(5) = [0,1,2,3,4]
15 #     print(i)

```

```
Welcome to Class! Allen
Welcome to Class! James
Welcome to Class! Tom
Welcome to Class! Jack
0
1
2
3
4
```

#### 範例4: 文字串列的索引應用 - 購物清單

```
1 sheet = ['牛奶', '蛋', '咖啡豆']
2 index = [0,1,2] # index是索引值, sheet[index]是購物項目
3
4 ## 位置索引與項目(一)
5 for i in sheet: # 購物清單迴圈
6     print (i)
7
8 ## 位置索引與項目(二)
9 for i in index:
10     print (i, sheet[i]) #i是索引值 · sheet[i]是項目
11
12
13 ## 位置索引與項目(三)
14 for i in range(len(sheet)): # length()函數計算串列長度(項目數)
15     print (i, sheet[i])
```

```
牛奶
蛋
咖啡豆
0 牛奶
1 蛋
2 咖啡豆
0 牛奶
1 蛋
2 咖啡豆
```

#### 【隨堂練習1】好朋友名單

請寫一個for迴圈列印自己的好朋友名單，如下：

```
我有幾位好朋友：
Michael是我的好朋友
Tom是我的好朋友
Andy是我的好朋友
June是我的好朋友
Axel是我的好朋友
```

## 12-2. 計數迴圈與range()函數

### 12-2-1 range()函數

1. `range()`函數可以產生數字序列，給我們某一個範圍的數字清單。

## 2. 用法

```
...
```

# `range`函數的表示方法：

```
range ( start, stop, step )
```

```
...
```

### `range`函數的參數預設值

- 初值`start`的預設值是0
  - 若省略初值(`start`)，則變數從第0個位置開始。
- 遞增值`step`的預設值是1
  - 若省略遞增值(`step`)，則變數每次自動+1
  - 若`step = -1`時，則表示變數每次遞減1。
- 終值`stop`的預設結尾值，不包含終值、是到終值的前一個數字。
  - 終值沒有預設值，不能省略。
  - 變數等於終值時離開，所以程式執行的最後一個值是終值的前一個數字。

### `range()`函數的三種寫法

- `range(終值) => range(0, stop, 1)`
- `range(初值, 終值) => range(start, stop, 1)`
- `range(初值, 終值, 遞增值)`

=> 變數會依據初值到終值的變化，依序指定給左邊的變數名稱。

數字串列中的元素，可以依照索引(index)或切片(slice)取得。

- 變數索引可用 變數名稱[`i`] 的方式取得。
  - 索引編號`i`是從0開始。
  - `d[0]` 代表變數 `d` 的第 1 筆資料內容。
- 變數切片可用 變數名稱[`i:j`] 的方式取得。
  - `d[1:4]` 代表變數 `d` 第 2 筆資料內容到第4筆資料內容。

### 範例5：range()函數的使用

```
1  ## list1 = [0,1,2,3,4]
2  a = range(0,5,1) # a = range(0,5), a = range(5)
3  print(a)
4
5  ## 要用變數索引才知道用range()函數輸入的元素是什麼
6  print(a[0], a[1], a[2], a[3], a[4])
```

`range(0, 5)`

0 1 2 3 4

```
1  ## list2 = [2,3,4,5]
2  b = range(2, 6, 1) # b = range(2,6)
```

```
3 print(b[0], b[1], b[2], b[3]) # 四筆資料，index從0到3
```

2 3 4 5

```
1 ## list3 = [2,5,8]
2 c = range(2, 10, 3) # step為正，表示座標往右走、一次跳3個數
3 print(c[0], c[1], c[2]) # 三筆資料，index從0到2
```

2 5 8

```
1 ## list4 = [10,8,6,4]
2 d = range(10, 3, -2) # step為負，表示座標往左走、一次跳2個數
3 print(d[0], d[1], d[2], d[3]) # 四筆資料，index從0到3
```

10 8 6 4

【隨堂練習2】：請用range()函數建立下面的數字串列，並用索引將裡面的項目一一取出。

1. [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
2. [0, 2, 4, 6, 8]
3. [5, 7, 9]
4. [0, -1, -2, -3, -4, -5, -6, -7, -8, -9]

答案如下：

```
0 1 2 3 4 5 6 7 8 9
0 2 4 6 8
5 7 9
0 -1 -2 -3 -4 -5 -6 -7 -8 -9
```

## 12-2-2 for迴圈

### 1. for迴圈的語法：

通常for迴圈和range函數一起搭配使用。

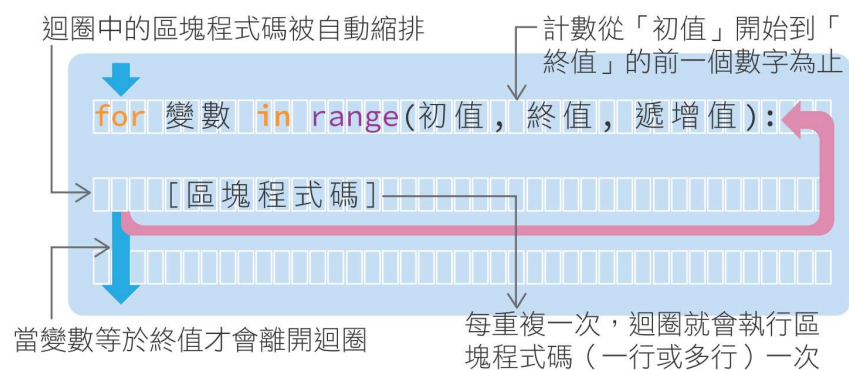


圖 4-3.4 for 迴圈語法說明

### for 迴圈

變數會依據初值到終值變化，若沒有指定遞增，預設為遞增 1，若設為負數表示遞減，反覆執行區塊程式碼，當變數等於終值才會離開迴圈，所以要記得變數等於終值是沒有執行區塊程式碼。

- for迴圈，透過range函數，指定迴圈變數的初值、終值與遞增(減)值。
  - 迴圈變數由初值到終值的前一個數字為止，不包括終值。

- 每重複執行一次，迴圈變數就依照遞增(減)值遞增或遞減，並執行迴圈內程式。
- for迴圈指令的最後面要加「冒號：」，代表底下區塊程式碼的敘述要開始。
- 區塊程式碼需要縮排，每行一律空四個空格。
  - Python用「縮排上下對齊」的方式，判斷是否在同一程式區塊。
  - 其他程式語言大多是利用大括號{}代表執行區塊。

一般而言，for迴圈使用的比while迴圈多，而for迴圈與range()函數的搭配用的又比for迴圈與資料容器串列多。下面我們整理for...in range()的三種常見用法，希望同學能夠瞭解並且熟悉這些方法。

## 12-2-3 應用

### 題型（一）：列印

範例6：文字列印，印出5個Hello。

~ 虛擬碼

for迴圈用i來數數，若i在數字範圍1~5內(range(0,5,1))，就列印Hello，超過就停止。

流程圖

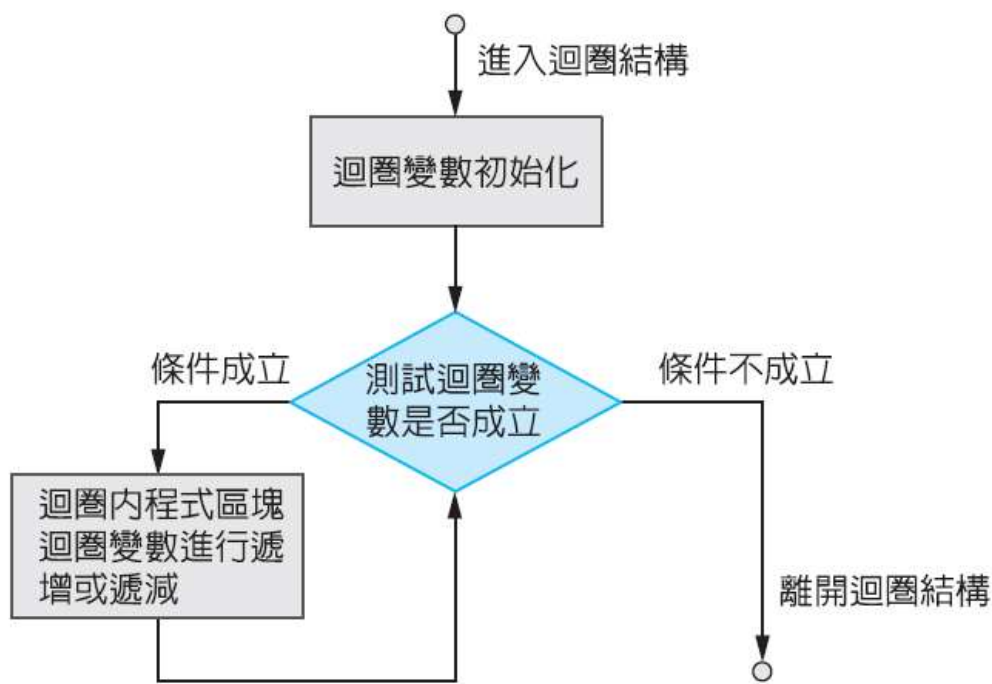


圖 A-11 流程圖

```

1  ## 印出5個Hello
2  for i in range(0,5,1): #[0,1,2,3,4]
3      print("Hello")
  
```

Hello  
Hello  
Hello  
Hello  
Hello

範例 7: 數字列印，印出range()函數值的範圍



```
1 ## [1,2,3,4,5]
2 for i in range(1, 6, 1):
3     print(i) # 預設為下一行 \n

1 ## [0,1,2,3,4]
2 for i in range(5): # range(0,5,1)
3     print(i) # 預設為下一行

1 ## [2,3,4]
2 for i in range(2,5): #range(2, 5, 1)
3     print(i, end = " ") # 橫著列印，中間空一格

1 ## [2,4,6,8]
2 for i in range(2,10,2): # range(2, 9, 2]
3     print(i, end = " ")

1 ## [100,97,94,91]
2 for i in range(100, 90, -3): #range(100, 90, -3)
3     print(i, end = " ")
```

使用方法	範例	執行結果
range( 終止值 ) range 函式指定「終止值」，數字串列會到「終止值」的前一個數字為止，沒有指定起始值，預設起始值為 0，沒有指定遞增值，預設為遞增 1。	for i in range(5): print(i)	0 1 2 3 4
range( 起始值, 終止值 ) range 函式指定「起始值」與「終止值」，數字串列由「起始值」開始到「終止值」的前一個數字為止，沒有指定遞增值，預設為遞增 1。	for i in range(2,6): print(i)	2 3 4 5

使用方法	範例	執行結果
range( 起始值, 終止值, 遞增 ( 減 ) 值 ) range 函式指定「起始值」、「終止值」與「遞增 ( 減 ) 值」，數字串列由「起始值」開始到「終止值」的前一個數字為止，每次遞增或遞減「遞增 ( 減 ) 值」。	for i in range(2,10,2): print(i)	2 4 6 8
	for i in range(100,90,-3): print(i)	100 97 94 91

題型 ( 二 )：計數



### 【隨堂練習3】：統計摩天輪旋轉次數

小芳在兒童樂園玩摩天輪，她很想知道玩一次摩天輪，到底能轉多少圈。可是她一上摩天輪就緊張，為了不出錯，每次旋轉到最高處，自己拍一張照片。回家後查看照片數，並做好記錄，經過查看照片共統計出摩天輪轉了9次。請問如何用Python語言寫出與下面用手寫做出的相同紀錄。

```

1 @
2 @@
3 @@@
4 @@@@
5 @@@@@
6 @@@@@@
7 @@@@@@
8 @@@@@@
9 @@@@@@

```

摩天輪總共轉了9圈。

### 題型（三）：加總

範例8：寫一個程式計算數字1到10之和。

#### ~ 解題想法

- 可以用for迴圈。
- 迴圈變數
  - 起始值為加總的起始值
  - 終止值為加總的終止值
  - 每執行一次迴圈，就會依照遞增(減)值變動。
- 加總過程
  - 迴圈內使用「 $\text{sum} = \text{sum} + \text{迴圈變數}$ 」進行數值加總。
  - 等號賦值。先計算等號右邊算式「 $\text{sum} + i$ 」，將結果回存到等號左邊變數(sum)。
  - sum要先給初值。

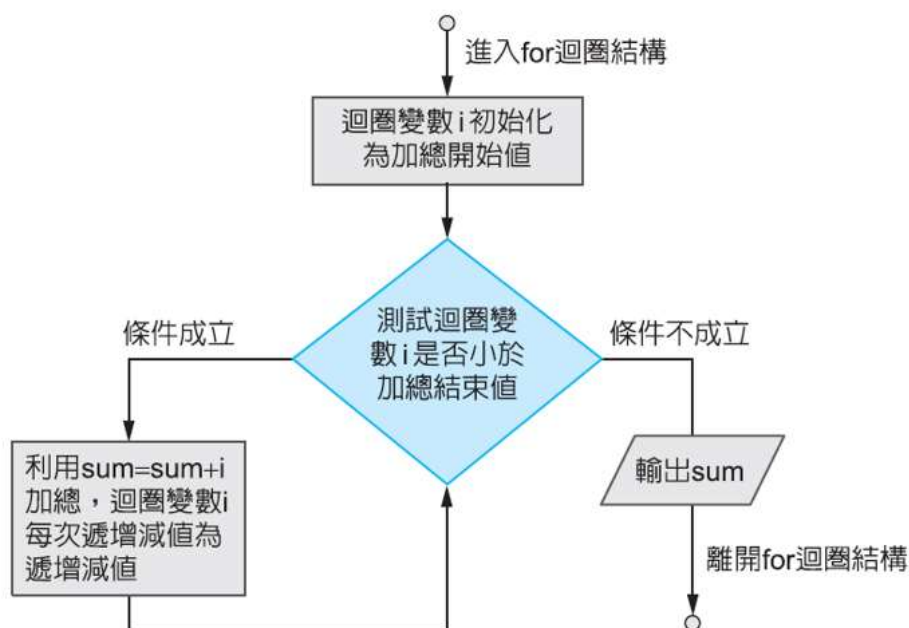


圖 A-13 流程圖

```
1 # 數字1到10之和。
2
3 sum = 0 # 起始條件
4 for i in range(1, 11, 1): # 數字串列[1,2,3,4,5,6,7,8,9,10]
5     sum = sum + i # 加總運算,變更條件
6 print("Total is", sum)
```

Total is 55

## 12-3. 進階迴圈

### 12-3-1 特殊指令的使用

迴圈在特殊需求下可以使用break、continue ( 與else ) 指令。

- 要跳出迴圈時，請使用break指令跳出迴圈。

break是讓迴圈終止不再執行的指令

```
1 # 對for迴圈執行break
2 for i in range(1,6):
3     print(i)
4     if i == 3:
5         break
```

1  
2  
3

```
1 # 對while迴圈執行break
2 i = 1
3 while (i<6):
4     print(i)
5     if i == 3:
6         break
7     i += 1
```

- 要跳過某一迴圈的執行，然後繼續迴圈，則請使用continue指令。

continue是讓迴圈跳過這一圈不做，但還是繼續做後面的迴圈。

```
1 # 對for迴圈執行continue
2 n = 0
3 for i in range(1,6):
4     if i%2 == 0:
5         continue
6     n += 1
7     print("執行的元素是", i)
```

執行的元素是 1

執行的元素是 3

執行的元素是 5

```

1  # 對while迴圈執行continue
2  i = 0
3  while (i<5):
4      i += 1
5      if i == 3:
6          continue
7      print(i)

```

- 迴圈正常結束後，可執行else 程式區塊，為可有可無的指令。

## 12-3-2 巢狀迴圈

巢狀迴圈是多層迴圈，為迴圈內又有迴圈的程式結構。從外層迴圈來看，內層迴圈只是外層迴圈內的動作，因此外層迴圈作用一次，內層迴圈全部都需要執行一次。

### 範例9：畫星星

方式一：正常的5列5行

```

* * * * *
* * * * *
* * * * *
* * * * *
* * * * *

```

方式二：左上角的直角三角形

```

* * * * *
* * * *
* * *
* *
*

```

方式三：左下角的直角三角形

```

*
* *
* * *
* * * *
* * * * *

```

```

1  # for 迴圈嵌套：內循環與外循環相互獨立
2  print("方式一：正常的5列5行")
3  for i in range(5):
4      for j in range(5):      # 第二層迴圈
5          print(" * ", end="")
6      print("")

```

方式一：正常的5列5行

```

* * * * *
* * * * *
* * * * *
* * * * *
* * * * *

```

```

1  # for 迴圈嵌套：內迴圈依賴外部循環變數
2  print("方式二：左上角的直角三角形")
3  for i in range(5,0,-1):
4      for j in range(i):          # 第二層迴圈
5          print(" * ", end="")
6      print("")

```

方式二：左上角的直角三角形

```

* * * * *
* * * *
* * *
* *
*

```

```

1  # for 迴圈嵌套：內迴圈依賴外部循環變數
2  print("方式三：左下角的直角三角形")
3  for i in range(5):
4      for j in range(i + 1):      # 第二層迴圈
5          print(" * ", end="")
6      print("")

```

方式三：左下角的直角三角形

```

*
* *
* * *
* * * *
* * * * *

```

### 加分練習題: 字串拼接成語接龍

用Python語言編制一個成語接龍遊戲。玩家輸入成語後，程序會把所有成語拼皆起來，顯示拼接的長龍。

```

In [18]: # 反覆輸入成語，把成語字串拼接出來。
...: saying = input("請輸入一個四字成語： ")
...: for i in range(3):
...:     next = input("接上一句成語，再輸入一個成語： ")
...:     saying += next
...: print(saying)

```

請輸入一個四字成語： 收放自如

接上一句成語，再輸入一個成語： 如花似玉

接上一句成語，再輸入一個成語： 玉樹臨風

接上一句成語，再輸入一個成語： 風景怡人

收放自如如花似玉玉樹臨風風景怡人

