



# 程式設計概論 Programming 101

## —list操作

授課老師：邱淑怡

Date: 10/19/2023

# 大綱

- list(串列)簡介
  - 序列型別(sequence type)
- list的資料型態下提供的function

# list (串列)

- 串列(list)是由一群資料排在一起形成的
  - 串列是由一連串資料所組成，有順序且可改變內容的序列
  - 定義時必須使用中括號，並在內容之間以逗點隔開
- 如何建立串列?
  - `list()` function可建立串列：`list1=list()` #建立空串列
  - (or) `list1_new=[]` #也是建立空串列
  - `list2=list([1,2,3])` #建立包含 1,2,3的串列
  - `list3=list(range(5))` #建立包含 0,1,2,3,4的串列
  - `list4=list(range(10, -10,-2))` #建立包含 10, 8, 6, 4, 2, 0, -2, -4, -6, -8的串列
  - `list5=list("ABCDE")` #建立包含'A','B','C','D','E'的串列

# list 操作

- 串列(list)使用時則是依照內容順序，由 0 開始數，位置即是索引值(index)
  - `a1=[9, 1, 3, 5.6, "python",-10]`
  - `print(a1[0])`
  - `print(a1[1])`
  - `print(a1[-1])`
  - `a1[2]=10`
  - `del a1[1]`
  - `print(a1)`

## list (cont.)

```
a1 = [12, 43, 56, 87, 9, 132, 456, 879, 1321, 4678, 13445]
print(132 in a1)
print(678 not in a1)
print(a1[4:7]) # a[4] 開始取到 a[7-1]，或者說 a[4] 開始取 7-4
個
print(a1[3:8:2]) # a[3] 開始取到 a[8-1]，每 2 個取 1 個
print(a1[-1]) # 倒數第一個
print(len(a1)) # list 長度
# 如果內容都是數字，可以使用以下的計算
print(min(a1))
print(max(a1))
print(sum(a1))
print(a1[4:11])
```

## list練習題

- $A=[1,3,6,-13,5,67,-8]$ 
  - 求list A 最大值和最小值
  - 取出list A 中第五個數值
  - 在list A 刪除第三個數值
  - 在list A 第四個數值變成 100

# 串列的運算

- 連接運算子: `[1,2,3]+["Taipei","Tokyo","Vienna"]`
- 重複運算子: `3*[1,2,3] = [1,2,3]*3`
- 比較運算子: `>, <, >=, <=, ==, !=`
  - `[1,2,3] != [1,2,3,4] # True`
  - `[1,"Hello","Python"] == ["Hello","Python",1] # False`
  - `['a','A'] < ['a','B'] # True`
- 索引運算子(`[]`):索引`0`表示串列第一個元素，索引`-1`表示串列最後一個元素
- 片段運算子: `[start:end]`指定索引範圍
  - 記得不包含`end`的數值(i.e., 也就是只包含`end`的前一個數值)

# 串列(list)資料型態下提供給該變數的function

```
x1=[0,1,2,3,4,5,3,-3,3.2,3]
```

```
x1.append(100) #新增一筆資料於list中最後一筆
```

```
x1.sort()
```

```
x1.count(3)
```

```
x1.reverse()
```

```
x1.index(1)
```

```
x1.insert(0,100)
```



# 練習題

- Q1: `x2=list(range(1,11))`
  - x2新增一筆資料在該串列第二個位置中該筆資料為**11111**
  - x2中刪除第一筆資料
  - 請告訴我**11111**該數值的位置是第幾筆資料

## 排序function: sort()

- 排序的方式使用自己定義的大小來排序
  - `a1 = ['Matlab', 'C', 'Ruby', 'Java', 'Python', 'R', 'JavaScript']`
  - `b1=sorted(a1)`
  - `print(b1)`
  - `c1=sorted(a1, key=len)`
  - `print(c1)`

## 二維串列(two-dimension list)

- 二維表格或矩陣，可用二維串列來存放

- Ex:儲存五個學生國英數成績

Grades=[[96,65,73],[88,76,82],[92,84,89],[82,73,64],[70,83,68]]

	國文	英文	數學
學生1	96	65	73
學生2	88	76	82
學生3	92	84	89
學生4	82	73	64
學生5	70	83	68

對應的  
索引值

	國文	英文	數學
學生1	[0][0]	[0][1]	[0][2]
學生2	[1][0]	[1][1]	[1][2]
學生3	[2][0]	[2][1]	[2][2]
學生4	[3][0]	[3][1]	[3][2]
學生5	[4][0]	[4][1]	[4][2]

# 如何運用?

- Ex: 儲存五個學生國英數成績

```
Grades=[[96,65,73],[88,76,82],[92,84,89],[82,73,64],[70,83,68]]
```

#取出某個學生的全部成績或某一科成績:

```
Grades[0]
```

```
Grades[1]
```

```
Grades[0][0]
```

```
Grades[1][2]
```

# 實例說明

- 印出每位學生的總分

```
Grades = [[96,65,73],[88,76,82],[92,84,89],[82,73,64],[70,83,68]]
for i in range(5):
    subtotal=0
    for j in range(3):
        subtotal = subtotal+Grades[i][j]
    Grades[i].append(subtotal)

for i in range(5):
    print("學生", i+1, "的總分為", Grades[i][3])
```

# 二維串列

- 二維串列可以用來存數學的矩陣(matrix)，下面有個4\*3的矩陣，請撰寫一行敘述定義一個名稱為mar1是4\*3的二維串列來存放該

矩陣 
$$\begin{bmatrix} 1 & 2 & 4 \\ 5 & 7 & 8 \\ 12 & 3 & 14 \\ 14 & 6 & 9 \end{bmatrix}$$

```
mar1=[[1,2,4],[5,7,8],[12,3,14],[14,6,9]]
```

# 練習題

# 練習題1

1. listA=[1,2,3,4,5,6,7,8,9,10,11]

- 請從listA印出[1,3,5,7,9,11]
- 請從listA印出最後三筆資料
- 請從listA印出第二筆資料到第五筆資料



## 練習題2

- 搜尋串列 $x = [1, 3, -7, 4, 9, -5, 4]$ 當中找出是否有負數，若有請印出它所在的索引值及該數值
- 請印出如下結果
  - `Index:2; value:-7`
  - `Index:5; value:-5`

## 練習題3

請讓使用者輸入一些數值並存於一個串列中，直到使用者輸入輸入no後才結束輸入(no不存到串列內)，最後印出該串列