



程式設計II 課程介紹

DavidCheng
2021/05/14

Python-串列資料應用

◆ 串列資料結構

◆ 串列的內建函式

◆ 串列的方法

◆ 串列的運算式

◆ 字串與串列轉換

◆ 串列的排序作業

◆ 二維串列

Python-串列資料 1/7

- 程式使用變數來儲存運算資料級及結果，當程式需要運算多筆資料時(5位學生的測驗成績)，若個別以變數宣告，必須宣告5個int整數變數來儲存這5個成績：

quiz1 = 71

quiz2 = 83

quiz3 = 67

quiz4 = 49

quiz5 = 59

測驗編號	成績
1	71
2	83
3	67
4	49
5	59

- 如果是50位學生的成績，我們需要50個變數；如此使得程式碼維護變得相當麻煩
- 觀察上述測驗成績的5個變數，其擁有的共同特性：
 - 變數有循序性，擁有順序的編號1~5
 - 變數的資料型態相同都是int

Python-串列資料 2/7

- 我們可以將5個成績變數集合起來，使用一個名稱`quizzes`代表：

```
quizzes = [71, 83, 67, 49, 59]
```

索引	
0	71
1	83
2	67
3	49
4	59

- 串列(List)如同是排成一列的箱子，每一個箱子可儲存一筆資料，稱為「元素 (Element)」，以此例`quizzes`串列內有5個元素，每個元素是單一值，稱為一維串列

Python-串列資料 3/7

- Python串列資料(list)類似其它程式語言的陣列資料(array)，但串列元素允許是不同資料型別

- 串列名稱 = [元素1, 元素2, 元素3, ...]

```
quizzes = [71, 83, 67, 49, 59]
```

#整數串列

```
fruit = ['appl', 'orange', 'melon']
```

#字串串列

```
price = ['apple', 17, 'orange', 23]
```

#不同資料型別組成之串列

```
mylist = []
```

#空串列

- 串列使用索引值(Index)來存取串列元素，索引值從0開始，不得超出串列範圍

```
print (quizzes[0])
```

#71

- 索引值可以是負數，-1表示最後一個元素，-2表示倒數第二個元素，一樣不能超過串列範圍

```
print (quizzes[-1])
```

#59

Python-串列資料 4/7

- 冒號運算子用來取得串列資料的子串列或元素(slicing語法)

- [start:end]
從start到end-1
- [start:end:increment]
從start到end-1 · 遞增取值
- [start:end:decrement]
從start到end+1 · 遞減取值

```
1 numlist = [1,2,3,4,5,6]
2 print(numlist[:])
3 print(numlist[1:3])
4 print(numlist[1:-2])    #-2表示倒數第2個元素
5 print(numlist[0:5:2])
6 print(numlist[5:0:-2])
```

```
[1, 2, 3, 4, 5, 6]
[2, 3]
[2, 3, 4]
[1, 3, 5]
[6, 4, 2]
```



Python-串列資料 5/7

- 串列的元素可以是另一個串列，形成多維串列

```
employee = [ ["joe", 23], ["mary", 21], ["david", 22]]
```

- 允許每一維元素為不規則

- 多維串列元素的存取是使用多個中括號索引組合

```
print(employee[1])           #["mary", 21]
print(employee[1][1])        #21
```

- `del list[n1:n2]`

刪除list串列n1到n2-1元素

- `del list[n1:n2:n3]`

刪除list串列n1到n2-1元素，刪除間隔為n3

Python-串列資料 6/7

- 字串資料可視為字元串列處理

```
1 alpha = 'abcdefghijklmnopqrstuvwxyz'  
2 print(alpha[3:15])  
3 print(alpha[3:15:3])  
4 print(alpha[15:3:-3])  
5 print(alpha[3:15:-1])  
6 print(alpha[::-3])
```

defghijklmno

dgjm

pmjg

zwtqnkheb

Python-串列資料 7/7

- 串列在建立時已存放有元素的初值，但如同變數一樣，允許更新元素內容：
 - ▣ 串列名稱[索引] = 資料

```
1 quizzes = [71, 83, 67, 49, 59]           #整數串列
2 print(quizzes[4])
3 quizzes[4] = 60
4 print(quizzes[4])
```

59

60

Python-串列的內建函式 1/2

- `len(list)`
取得串列元素數目
- `min(list)`
取得`list`串列中最小的元素值
- `max(list)`
取得`list`串列中最大的元素值
- `sum(list)`
計算`list`串列元素的總和
- `sorted(list)`
回傳一個由小到大排序的`list`，**並不會改變原來的串列資料**

Python-串列的内建函式 2/2

```
1 quizzes = [71, 83, 67, 49, 59] #整數串列
2 print(len(quizzes))
3 print(min(quizzes))
4 print(max(quizzes))
5 print(sum(quizzes)/len(quizzes))
6 print(sorted(quizzes))
7 print(quizzes)
```

5

49

83

65.8

[49, 59, 67, 71, 83]

[71, 83, 67, 49, 59]

Python-串列的方法 1/3

- `list.append(n1)`
將n1加到list串列最後，n1可以是元素或串列
- `list.clear()`
刪除串列所有的元素
- `list.count(n1)`
n1元素在list串列中出現的次數
- `list.extend(list2)`
將list2加到list串列最後
- `list.index(n1)`
list串列中第1個n1元素的索引值

Python-串列的方法 2/3

- `list.insert(n, n1)`
在list串列位置n插入n1元素
- `list.pop(n)`
從list串列取出位置n元素，並將n1自list串列中移除；省略參數n表示最後一個元素
- `list.remove(n1)`
從list串列移除第一個n1元素
- `list.reverse()`
反轉串列順序
- `list.sort()`
將串列由小到大排序

Python-串列的方法 3/3

```
1 score = []
2 total = inscore = 0
3 while(inscore != -1):
4     inscore = int(input("請輸入學生的成績："))
5     score.append(inscore)
6 score.remove(-1)
7 print("共有 %d 位學生" % (len(score)))
8 average = sum(score) / (len(score))
9 print("最高分：%d 分，最低分：%d 分，平均成績：%5.2f 分" % (max(score), min(score), average))
10
```

請輸入學生的成績：71

請輸入學生的成績：83

請輸入學生的成績：67

請輸入學生的成績：49

請輸入學生的成績：59

請輸入學生的成績：-1

共有 5 位學生

最高分：83 分，最低分：49 分，平均成績：65.80 分

Python-串列運算子 1/2

- `in`
判斷指定資料是否存在於串列中
- `not in`
判斷指定資料是否不存在於串列中
- `=`
複製串列
- `+`
連結兩個串列
- `*`
複製串列元素

Python-串列運算子 2/2

```
list1 = [10, 20, 30, 40]
print(10 in list1)
print(20 not in list1)
list2 = list1
print(list2)
list3 = list1 * 2
print(list3)
print(list1 + ['a', 'b', 'c', 'd'])
```

True

False

[10, 20, 30, 40]

[10, 20, 30, 40, 10, 20, 30, 40]

[10, 20, 30, 40, 'a', 'b', 'c', 'd']

Python-字串與串列轉換 1/2

- 字串資料可使用`split()`方法分割為字串串列

字串串列 = 字串.`split`(切割字元, 分割次數)

- 預設使用空字元(空格、換行`\n`或跳格`\t`)做為切割字元
- 分割次數, 預設為-1, 表示全部分割

```
1 url = "https://www.python.org/static/img/python-logo.png"
2 print(url.split('/'))
3 print(url.split('/', 1))
```

```
['https:', '', 'www.python.org', 'static', 'img', 'python-logo.png']
['https:', '/www.python.org/static/img/python-logo.png']
```


Python-字串與串列轉換 2/2

- 將字串串列的元素可使用join方法，連接成一個長字串

字串 = '連結字元'.join(字串串列)

```
1 pathlist = ['https:', '', 'www.python.org', 'static', 'img', 'python-logo.png']  
2 url = '/'.join(pathlist)  
3 print(url)
```

<https://www.python.org/static/img/python-logo.png>

Python-串列的排序 1/2

- 串列的元素，可以按資料值由小到大的排列方式，重新安排元素順序
 - 串列名稱.sort() #將串列元素由小到大排列
 - 串列名稱.sort(reverse = True) #將串列元素由大到小排列
- 串列的元素，可以按反方向重新排列元素的順序
 - 串列名稱.reverse()

```
1 quizzes = [71, 83, 67, 49, 59]
2 quizzes.sort()
3 print(quizzes)
4 quizzes.sort(reverse = True)
5 print(quizzes)
6 quizzes.reverse()
7 print(quizzes)
```

[49, 59, 67, 71, 83]

[83, 71, 67, 59, 49]

[49, 59, 67, 71, 83]

Python-串列的排序 2/2

- 使用`sort()`方法排序串列，是採就地排序方式，串列經排序後會失去原有的排列順序。若要有排序後的結果，又要保有排序前的原貌，就得使用`sorted()`函式來複製串列並排序

新串列 = `sorted(串列名稱, reverse = True|False)`

- ▣ `reverse = True`，進行由大到小排序；`reverse = False`，進行由小到大排序

```
1 quizzes = [71, 83, 67, 49, 59]
2 score1 = sorted(quizzes, reverse = True)
3 score2 = sorted(quizzes, reverse = False)
4 print(quizzes)
5 print(score1)
6 print(score2)
```

[71, 83, 67, 49, 59]

[83, 71, 67, 59, 49]

[49, 59, 67, 71, 83]

Python-課堂練習-加分

請設計出5/1-5/14缺席登入系統

條件1：輸入姓名、是否缺席(Andy、Bella、Calvin、Davis、Edison)

條件2：全部出席率、各別出席率、哪天出席率最高、那天出席率最低

出席率的標準

- A 同學每兩天缺席一天
- B 同學每三天缺席一天
- C 同學每5的倍數缺席
- D 同學每天都到
- E 同學每天缺席半天