06串列(list)

串列可以當作是儲存資料的容器，這有利於資料的存取。串列在其他程式語言如C、C++、Java則稱為陣列(array)。Python的串列可以存放不同型態的資料，與上述其他語言的陣列型態不相同，其餘使用方式與性質則相同。

串列在運用上，可同時以一個名稱，乘載多個變數內容，能簡化變數使用，能簡化變數運用。

串列語法

串列名稱 = [串列值1,串列值2,串列值3... ]

1. 如變數一般，先設定串列名稱，等號後面加上中括號，當中放入需要的內容，與變數不同，內容可以有多筆，以逗號區隔開來。
2. 中括號 [ ] 內容可以是空的，表示後面程式會在置入新的內容

串列存放模式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 編號 0 | 編號1 | 編號2 | 編號3 | 編號4 |
| 元素(資料) | 元素 | 元素 | 元素 | 元素 |

長度共有5格，索引碼(編號)為0~4

1. 串列建立

建立串列的時候，透過[ ] 存放內容。

list1 = []  #建立一個空串列

list2 = [1,2,3,4,5] #建立串列並賦予數值資料

list3 = ['apple','orange','banana'] #建立串列並賦予字串資料

list4 = [1,2,34,56,'word'] #建立串列賦予不同型別資料

list5 = list(('apple','banana','cherry')) #使用list指令將字串集創建為串列

list6 = [x for x in range(10)] # 創建0-9連續數字的串列

list7 = [0]\*10  #創建 10個 0的串列

1. 串列運作
   1. [ ] [start : end ] 檢索值

串列存放透過索引值檢索資料，在檢索資料的時候，[ ]中的數字代表檢索值，用來尋找特定位置的內容

print(list2[3])     #印出索引值(index) 3 的串列項目

print(list2[1:3])   #印出索引值 1~(3-1) 的串列項目

print(list2[0:5])   #印出索引值 0~4 的串列項目

print(list2[:4])    #印出索引值 3之前 的串列項目

print(list2[3:])    #印出索引值 3之後 的串列項目

print(list2[-1])    #印出索引值 由後往前算第一位的串列項目

print(list2[:-1])   #印出索引值 由後往前算所有的項目(注意不含-1)

* 1. len()：計算串列長度

透過len(串列名稱 ) 可以查出串列中一共有多少個值

x = len(list1)

print(x)

print(len(list1))   #取得串列"list1"的長度

print(len(list2))

print(len(list3))

print(len(list4))

輸出

0

0

6

3

5

* 1. append 與 insert

使用 append(value)的方法，將內容增加到串列尾端，利用

insert (index,value)將內容插入串列的索引值位置。

print(list1)

list1.append("Tim")  #在list1最末端加入一筆資料 "Tim"

print(list1)

list1.append("Tina") #在list1最末端加入一筆資料 "Tina"

print(list1)

list1.insert(1,"Sky") #在list1索引值1的位置插入一筆資料 "Sky"

print(list1)

list3.insert(5,'kiwi')

print(list3)

輸出

[]

[1]

[1, 2]

[1, 4, 2]

list1.append(input('輸入整數：'))

print(list1)

輸出

輸入內容5

['5']

練習：

* + 1. 使用上面範例list1，試著用迴圈存入1~10，印出最後結果
    2. 使用上面範例list2，輸入兩個值，第一個值是串列索引，第二個值是內容，插入指定索引位置
  1. pop與remove

pop()可用來刪除串列最後一個數，pop(index)可用來刪除index指定的索引位置資料，利用remove(value)刪除串列中，內容為value所指定的資料。

print(list2)

list2.pop()

list2.pop(1)

list2.append(3)

print(list2)

list2.remove(3)

print(list2)

輸出 (list2是用”1.串列建立”資料)

[1, 2, 3, 4, 5]

[1, 3, 4, 3]

[1, 4, 3]

* 1. count 與 index

count(value)用來計算某個資料(value)在串列中出現的次數，index(value) 立即回傳value這個內容所在位置。

print(list3)

list3.append('apple')

print(list3)

c = list3.count('apple')

print(c) #計算apple在串列中出現幾次

list3.insert(0,'kiwi')

print(list3)

print(list3.index('apple')) #搜尋apple第一次出現的位址(索引值)

print("banana在串列中%d的位置"%list3.index('banana'))

print(list5)

list3.extend(list5)     #將list5串列資料加入到list3 >> list3 += list5

print(list3)

print(f"apple共出現{list3.count('apple')}次，banana第一次出現在{list3.index('banana')}")

輸出

['apple', 'orange', 'banana']

['apple', 'orange', 'banana', 'apple']

2

['kiwi', 'apple', 'orange', 'banana', 'apple']

1

banana在串列中3的位置

['apple', 'banana', 'cherry']

['kiwi', 'apple', 'orange', 'banana', 'apple', 'apple', 'banana', 'cherry']

apple共出現3次，banana第一次出現在3

* 1. sort與reversr

利用sort()串列由小至大加以排序。而reverse()則用來將串列內容反轉排列。

list1 = [1,4,8,3,7,2]

print(list1)

list1.sort()    #排序

list1.sort(reverse = True)    #反向排序

print(list1)

list1.reverse() #反轉內容

print(list1)

輸出

[1, 4, 8, 3]

[1, 3, 4, 8]

[8, 4, 3, 1]

* 1. in 與not in

in與 not in 用來判斷特定資料是否存在於串列中，其使用語法

資料 in 串列 或 資料 not in 串列 ，傳回值為True 或False

list1 = [1,4,8,3]

a = 5 in list1

b = 4 in list1

if a == True:

    print("串列中有5")

else:

    print("找不到5這個數")

if b == True:

    print("串列中有4")

else:

    print("找不到4這個數")

print()

a = 5 not in list1

b = 4 not in list1

print(a)

if a == True:

    print("串列中沒有5")

else:

    print("5存在")

if b == True:

    print("串列中沒有4")

else:

    print("4存在")

輸出

找不到5這個數

串列中有4

串列中沒有5

4存在

* 1. sum 、 max 以及 min 函式

sum()函式可以加總串列中的所有數值和，利用max() 和min()，可以找出最大與最小的資料

a = sum(list2)

b = max(list2)

c = min(list2)

print(f"list內元素和為：{a}\n最大值為：{b}\n最小值為{c}")

輸出

list內元素和為：15

最大值為：5

最小值為1

* 1. + 與 \*

串列功能中，使用 + 號能將兩個串列連結在一起，使用 \* 號 可以複製串列內容。

print (list2,list3)

lst = list2 + list3

print(lst)

輸出

[1, 2, 3, 4, 5] ['apple', 'orange', 'banana']

[1, 2, 3, 4, 5, 'apple', 'orange', 'banana']

print(list2)

lst = list2 \* 2

print(lst)

或

print(list2)

lst = 2 \* list2

print(lst)

輸出相同

[1, 2, 3, 4, 5]

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

* 1. 使用for 迴圈列印

使用for迴圈配合串列，可解決大量資料反覆列印的問題，前面已經用過迴圈進行串列輸入，在輸出部分也容易利用

for i in range(len(list3)):

    print("list[%d] = %s"%(i,list3[i]))

輸出

list[0] = apple

list[1] = orange

list[2] = banana

python 提供串列專用的for迴圈方式

for i in 串列名稱:

這種模式i在迴圈進行時，會依序置入串列內容

for i in list3:

    print(i,end = ' ')

輸出

apple orange banana

這種方式會造成原來索引值i原來用作數值顯示功能，如上面範例list[i] 這時就無法使用，解決辦法再加一個變數

x = 0

for i in list2:

    print("list[%d]=%d"%(x,i))

    x +=1

輸出

list[0]=1

list[1]=2

list[2]=3

list[3]=4

list[4]=5

串列(容器)運作**函式**

|  |  |
| --- | --- |
| 函式 | 意義 |
| len() | 計算串列長度 |
| sum() | 加總串列每一個元素 |
| max() | 回傳串列最大值 |
| min() | 回傳串列最小值 |

串列運作**方法**

|  |  |
| --- | --- |
| 方法 | 意義 |
| append(value) | 附加value於串列尾端 |
| insert(index, value) | 在索引index處加入value |
| pop() | 刪除串列最後一個元素 |
| pop(index) | 刪除串列索引index所在的元素 |
| remove(value) | 刪除串列中的value，若有多個相同value，只刪除第一個 |
| count(value) | 串列中出現value的個數 |
| index(value) | value所在串列第一個出現的索引值 |
| sort() | 將串列元素由小至大排序 |
| reverse() | 將串列元素反轉 |

串列運作相關運算子

|  |  |
| --- | --- |
| 運算子 | 意義 |
| in | 檢視某一元素是否在串列中 |
| not in | 檢視某一元素是否不在串列中 |
| [index] | 印出串列index位置中某一元素 |
| [start:end] | 印出串列中從start 到end-1的元素 |
| \* | 複製多次串列元素 |
| + | 連結兩個串列元素 |

* 1. 練習
     1. 使用隨機方式產生樂透號碼 6個數字(1~49)，並存放入串列中，印出結果。
     2. 上述樂透程式，加上串列檢查，當檢查出元素有重複時，替換該元素
     3. 人工輸入12個正整數，存入串列，排序後輸出結果
     4. 隨機輸出五張撲克牌，不論花色，點數存放到串列中，計算點數。
     5. 使用者輸入十個數字存放串列中，接著由大到小順序顯示最大的三個數字。
     6. 使用者輸入十個數字，做為樣本數，輸出眾數(出現最多次數的數字
     7. 使用迴圈反覆輸入成績，存放到串列，直到-9999結束，輸出最大值、最小值，加總與平均
     8. 設定一個串列，建立5個部門員工姓名，透過迴圈輸入員工成績，接下來依照成績排序輸出評核等級

|  |  |
| --- | --- |
| 成績 | 等級 |
| 90~100 | A |
| 80~89 | B |
| 70~79 | C |
| 60~69 | D |
| 60(不含)以下 | E |