

List

List

- ลิสต์หรือรายการข้อมูล เป็นตัวช่วยรวมชุดข้อมูลจำนวนมากเข้าเป็นกลุ่ม เพื่อความสะดวกในการประมวลผลกับข้อมูลเหล่านั้นในรูปแบบเดียวกัน

หากเรามีข้อมูลตัวเลข 3 จำนวนแล้วต้องการหาค่าผลรวม ค่ามากที่สุด ค่าเฉลี่ยเราต้องทำอย่างไร

Score1 =

Score2 =

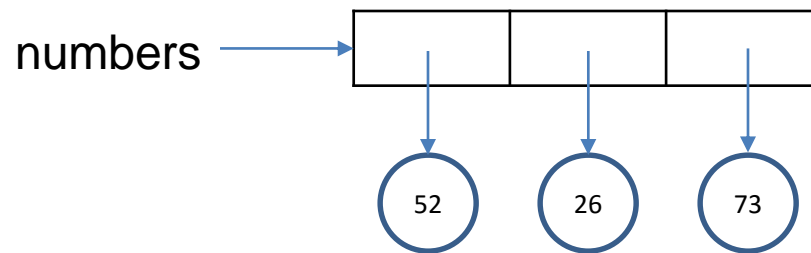
Score3 =

- หากใช้ลิสต์

```
numbers = [52,26,73]
```

```
print('Average =',sum(numbers)/len(numbers))
```

```
print('max = ',max(numbers))
```



การสร้างลิสต์

- การสร้างลิสต์มีรูปแบบดังนี้

[สมาชิกลิสต์ตัวที่1, สมาชิกลิสต์ตัวที่2,...]

สมาชิกลิสต์สามารถเป็นข้อมูลต่างชนิดกันได้ เป็น ลิสต์ว่างก็ได้
ส่วนมากจะกำหนดให้มีตัวแปรชี้ไปที่ลิสต์เพื่ออ้างอิงต่อไปเช่น

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

Student = ['Steven','Allen']

x = []

การประมวลผลข้อมูลในลิสต์

- ลิสต์มีฟังก์ชันให้ใช้ประมวลผลข้อมูลภายในซึ่งไม่ทำให้ข้อมูลต้นฉบับเปลี่ยนแปลงเช่น

ฟังก์ชัน	ผลลัพธ์	
len(L)	จำนวนข้อมูลภายใน L	Number = [1,2,3,4] print(len(Number))
sum(L)	ผลรวมของข้อมูลภายใน L เมื่อ L บรรจุตัวเลขเท่านั้น	print(sum(Number))
min(L)	ค่าต่ำสุดเมื่อ L บรรจุตัวเลขเท่านั้น	print(min(Number))
max(L)	ค่าสูงสุดเมื่อ L บรรจุตัวเลขเท่านั้น	print(max(Number))
L1+L2	ได้ลิสต์ใหม่ที่เกิดจาก L1 ต่อด้วย L2	Number1=[5,6,7] Number2=[1,2,3] Newlist=Number1+Number2
L*n	ได้ลิสต์ใหม่ที่เกิดจาก L ต่อกัน n ชุดเมื่อ n เป็นจำนวนเต็มบวก	Dup = Number1*5

การอ้างถึงข้อมูลในลิสต์

- การอ้างถึงข้อมูลในลิสต์ทำได้โดยการระบุดัชนีของข้อมูลแต่ละตัวครอบด้วยเครื่องหมาย [] ทั้งนี้ข้อมูลตัวแรกสุดของลิสต์จะมีดัชนีเป็น 0 และหากลิสต์มีข้อมูล n ตัว ดัชนีของข้อมูลตัวสุดท้ายจะเป็น $n-1$
- Months=['Jan','Feb','Mar','Apr','May','Jun','Jul','Aug','Sep','Oct','Nov','Dec']
- print(Months[0]) → Jan

หมายเหตุ string เทียบได้กับลิสต์ของอักขระ

การตรวจสอบการเป็นสมาชิกของลิสต์

- ใน Python มีคำสั่ง `in` เพื่อใช้ในการตรวจสอบการเป็นสมาชิกของลิสต์ โดยจำกัค่าเป็น `True` หรือ `False`
- `Months=['Jan','Feb','Mar','Apr','May','Jun','Jul','Aug','Sep','Oct','Nov','Dec']`
- `X='Jul'`
- `print(X in Months)`
- `Y='xxx'`
- `print(Y in Months)`

การเปลี่ยนแปลงข้อมูลในลิสต์

- เราสามารถใช้เครื่องหมาย = กับข้อมูลในลิสต์เพื่อกำหนดค่าหรือเปลี่ยนแปลงค่าในลิสต์ได้ ทั้งนี้ต้องระบุดัชนีให้ถูกต้องด้วย
- Months[1] = 'AAA'
- นอกจากนี้ลิสต์ยังมีการดำเนินการเพื่อช่วยให้สะดวกยิ่งขึ้น

คำสั่ง	การทำงาน	ตัวอย่าง
L.append(a)	เพิ่มข้อมูล a ไปต่อท้ายลิสต์ L	Numbers = [1,2,3] Numbers.append(10)
L.insert(i,a)	เพิ่มข้อมูล a ใน L ไปตำแหน่งก่อนหน้าดัชนี i	Numbers.insert(2,5)
L.sort()	หากข้อมูลเป็นตัวเลขจะเรียงจากน้อยไปมาก หากเป็นข้อความเรียงตามพจนานุกรม	Numbers.sort()
del L[i]	ลบข้อมูลใน L ตำแหน่งที่ i	del Numbers[1]

การเข้าถึงข้อมูลในลิสต์ด้วย for

- การเข้าถึงค่าสมาชิกในลิสต์ที่ละค่าตั้งแต่ต้นจนจบตามลำดับ เป็นสิ่งที่พบบ่อยในการประมวลผล
- for ตัวแปร in ลิสต์:
- กลุ่มคำสั่ง

ค่าตัวแปรจะในแต่ละรอบการทำงานจะเปลี่ยนไปตามลำดับในลิสต์

- Numbers= [1,2,3,4,5,6,7,8,9]
- for x in Numbers:
- print(x)

ฟังก์ชัน range()

- ใช้กำหนดค่าของข้อมูลที่เป็นช่วงลำดับในลิสต์ เช่น 1 2 3 4 หรือ 0 2 4 6 8 มีรูปแบบการใช้งาน 3 รูปแบบ
- `range(n)` สร้างข้อมูลที่แทนลิสต์ $[0, 1, \dots, n-1]$
- `range(a,b)` สร้างข้อมูลที่แทนลิสต์ $[a, a+1, \dots, b-1]$
- `range(a,b,c)` สร้างข้อมูลที่แทนลิสต์ $[a, a+c, a+2c, \dots, b']$ นั่นคือ เริ่มที่ a เพิ่มขึ้นทีละ c ทำไปเรื่อยๆ แต่ไม่เกิน b

การสร้างลิสต์ใหม่จากลิสต์เดิม

- ลิสต์สามารถถูกสร้างขึ้นใหม่โดยอาศัยข้อมูลจากลิสต์เดิมได้ในสองรูปแบบคือ list slicing และ list comprehension
- ลิสต์สไลซิง เป็นการระบุช่วงที่ต้องการ
 - ลิสต์เดิม[เริ่มต้น:สิ้นสุด] จะได้ข้อมูลตั้งแต่ดัชนีเริ่มต้นจนก่อนดัชนีสิ้นสุด
- ลิสต์คอมพรีเอนชัน เป็นการสร้างลิสต์ใหม่โดยระบุนิพจน์สำหรับการสร้างสมาชิกแต่ละตัว
 - [นิพจน์ for ตัวแปร in ลิสต์เดิม] เช่น `newl = [x**2 for x in oldl]`