Week 07 - Lists

Memo	
Week	7
課程單元	Python 基礎程式設計：Lists 清單

學習目標

- Python 基礎程式設計：Lists 清單 (Lesson 7)
- Sankify 範例解析：Lesson 7
  - Greater than neighbors
  - The number of distinct elements
  - The number of pairs of equal
  - Queens

參考資料

- Python 官方網站 (https://www.python.org/)
- Snakify - 線上學習網
- Zuvio - 大學師生互動平台

課程設計

Snakify 線上學習工具 Lesson 7 原理說明

Snakify - Python 3 Interactive Course

Most of programs work not only with variables. They also use lists of variables. For example, a program can handle an information about students in a class by reading the list of students from the keyboard or

https://snakify.org/en/lessons/lists/

https://snakify.org/en/lessons/lists/

Lists

> https://snakify.org/en/lessons/lists/

List Examples

```
L1 = [1,2,3,4,5]
L2 = ['a', 'b', 'c', 'd', 'e']
L3 = [1.2, 3.4, 5.6, 7.8]
L4 = [1, 2.3, 'STUST', 'a', 66]

print(f"L1 = {L1}")
print(f"L2 = {L2}")
print(f"L3 = {L3}")
print(f"L4 = {L4}")

for d in L4:
    print(d, end=" ")
```

```
L1 = [1, 2, 3, 4, 5]
L2 = ['a', 'b', 'c', 'd', 'e']
L3 = [1.2, 3.4, 5.6, 7.8]
L4 = [1, 2.3, 'STUST', 'a', 66]
1 2.3 STUST a 66
```

Access List Element

```
L1 = [1, 2, 3, 4, 5]

for index in range(len(L1)):
    print(f"L1[{index}] = {L1[index]}")
print()
print(f"L1[-1] = {L1[-1]}")
print(f"L1[1:3] = {L1[1:3]}")
print(f"L1[::-1] = {L1[::-1]}")
```

```
L1[0] = 1
L1[1] = 2
L1[2] = 3
L1[3] = 4
L1[4] = 5

L1[-1] = 5
L1[1:3] = [2, 3]
L1[::-1] = [5, 4, 3, 2, 1]
```

Mutable Data Structures - Lists

```
L1 = [1, 2, 3, 4, 5]

L1[3] = "Python"
print(f"L1 = {L1}")
del(L1[2])
print(f"L1 = {L1}")
L1.remove(1)
print(f"L1 = {L1}")
L1.insert(1,"STUST")
print(f"L1 = {L1}")
```

```
L1 = [1, 2, 3, 'Python', 5]
L1 = [1, 2, 'Python', 5]
L1 = [2, 'Python', 5]
L1 = [2, 'STUST', 'Python', 5]
```

Read input() line by line and append to the list L

```
N = int(input())
L = []
for i in range(N):
    L.append(input())
print(f"L = {L}")
```

```
 5
 1
 2.1
 Hello
 a
 3.6
L = ['1', '2.1', 'Hello', 'a', '3.6']
```

List Concatenation and Repetition

```
L1 = [1,2,3,4,5]
L2 = ['a', 'b', 'c', 'd', 'e']

print(f"L1 + L2 = {L1+L2}")
print(f"L1 * 2 = {L1 *2}")
```

```
L1 + L2 = [1, 2, 3, 4, 5, 'a', 'b', 'c', 'd', 'e']
L1 * 2 = [1, 2, 3, 4, 5, 1, 2, 3, 4, 5]
```

Extract digits from the list

```
s = 'ab12c59p7dq'
digits = []
for symbol in s:
    if '1234567890'.find(symbol) != -1:
        digits.append(int(symbol))
print(digits)
```

```
[1, 2, 5, 9, 7]
```

Split method

```
s1 = "1 2 3 4"
s2 = "a,b,c,d"
s3 = "a , b  , c ,  d"
print(f"s1.split() = {s1.split()}")
print(f"s2.split(',') = {s2.split(',')}")
print(f"s3.split(',') = {s3.split(',')}")
```

```
s1.split() = ['1', '2', '3', '4']
s2.split(',') = ['a', 'b', 'c', 'd']
s3.split(',') = ['a ', ' b  ', ' c ', '  d']
```

```
ns = input().split()
print(f"ns = {ns}")
sum = 0
for n in ns:
    sum += int(n)
print(sum)
```

```
 1 2 3 4 5 6 7 8 9 10
ns = ['1', '2', '3', '4', '5', '6', '7', '8', '9', '10']
55
```

Join method

```
L = ['a', 'b', 'c', 'd', 'e']
print(f"':'.join(L) = {':'.join(L)}")
print(f"'->'.join(L) = {'->'.join(L)}")
print(f"' + '.join(L) = {' + '.join(L)}")

I = [1, 2, 3, 4, 5]
L = [str(x) for x in I]
print(f"' + '.join(L) = {' + '.join(L)}")
```

```
':'.join(L) = a:b:c:d:e
'->'.join(L) = a->b->c->d->e
' + '.join(L) = a + b + c + d + e
' + '.join(L) = 1 + 2 + 3 + 4 + 5
```

Generators `[expression for variable in sequence]`

```
s1 = [0]*5
s2 = [0 for i in range(5)]
s3 = [6*a for a in range(1,10)]

print(f"s1 = {s1}")
print(f"s2 = {s2}")
print(f"s3 = {s3}")
```

```
s1 = [0, 0, 0, 0, 0]
s2 = [0, 0, 0, 0, 0]
s3 = [6, 12, 18, 24, 30, 36, 42, 48, 54]
```

```
ns = [int(x) for x in input().split()]
```

```
print(f"ns = {ns}")
sum = 0
for n in ns:
    sum += n
print(sum)
```

```
 1 2 3 4 5 6 7 8 9 10
ns = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
55
```

```
from random import randrange

rn5 = [randrange(1,6) for i in range(5)]
rn10 = [randrange(1,6) for i in range(10)]
rn20 = [randrange(1,6) for i in range(20)]
print(f"rn5  = {rn5}")
print(f"rn10 = {rn10}")
print(f"rn20 = {rn20}")
```

```
rn5 = [4, 3, 2, 1, 1]
rn10 = [5, 1, 2, 4, 3, 4, 2, 3, 5, 5]
rn20 = [2, 2, 5, 5, 3, 3, 1, 4, 1, 4, 2, 5, 1, 3, 5, 5, 2, 2, 1, 5]
```

```
a = [input() for x in range(int(input()))]
print(f"a = {a}\n")

b = [int(input()) for x in range(int(input()))]
print(f"nb = {b}")
print(f"b.count(0) = {b.count(0)}")
```

```
 3
 2
 4
 6
a = ['2', '4', '6']

 5
 1
 0
 2
 0
 100
nb = [1, 0, 2, 0, 100]
b.count(0) = 2
```

Snakify 範例解析 - Greater than neighbors

> https://snakify.org/en/lessons/lists/problems/greater_than_neighbours/

✏️ Given a list of numbers, determine and print the quantity of elements that are `greater than both of their neighbors`.

```
n = [int(x) for x in input().split()]
cnt = 0
for i in range(1,len(n)-1):
    if n[i-1]<n[i]>n[i+1]:
        cnt += 1
print(cnt)
```

Snakify 範例解析 - The number of distinct elements

> https://snakify.org/en/lessons/lists/problems/num_distinct/

✏️ Given a list of numbers with all of its elements `sorted in ascending order`, determine and print the quantity of `distinct` elements in it.

```
n = [int(x) for x in input().split()]
cnt = 1
a = n[0]
for b in n[1:]:
    if a!=b:
        cnt += 1
    a = b
print(cnt)
```

Snakify 範例解析 - The number of pairs of equal

> https://snakify.org/en/lessons/lists/problems/num_equal_pairs/

✏️ Given a list of numbers, `count how many element pairs have the same value (are equal)`. Any two elements that are equal to each other should `be counted exactly once`.

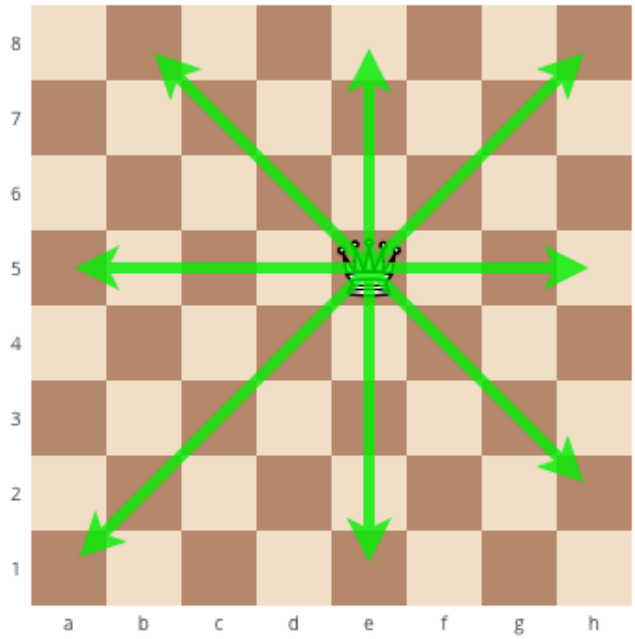
```
n = [int(x) for x in input().split()]
cnt = 0
for i in range(len(n)-1):
    for j in range(i+1,len(n)):
        if n[i]==n[j]:
            cnt += 1
print(cnt)
```

Snakify 範例解析 - Queens

> https://snakify.org/en/lessons/lists/problems/queens/

✏️ In chess it is known that it is possible to `place 8 queens on an 8×8 chess board` such

that none of them can attack another. Given a placement of 8 queens on the board, determine if there is a pair of queens that can attach each other on the next move. `Print the word NO if no queen can attack another`, `otherwise print YES.` The input consists of eight coordinate pairs, one pair per line, with each pair giving the position of a queen on a standard chess board with `rows and columns numbered starting at 1`.

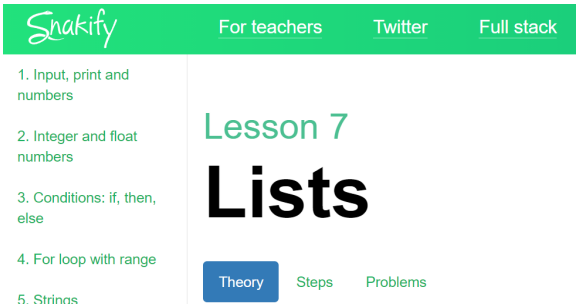



```
def attact(a, b):
    c1 = a[0]==b[0] or a[1]==b[1]
    c2 = abs(a[0]-b[0])==abs(a[1]-b[1])
    return c1 or c2

q = []
for i in range(8):
    q.append([int(x) for x in input().split()])

chk = False
for i in range(7):
    for j in range(i+1,8):
        if attact(q[i],q[j]):
            chk = True
            break
if chk:
    print("YES")
else:
    print("NO")
```

課堂作業

Homework 01 - Snakify Python 程式問題練習。 **驗收期限：期中考前一週 (10/31)。**

- 請完成 Lesson 7 中的所有題目。
- 請務必先註冊且登入你的 Snakify 帳號。



https://snakify.org/en/lessons/lists/