

# Python

## Loop (迴圈)

迴圈：讓電腦重複執行某一段程式

你可以設定重複的次數，以及每次重複時  
要做甚麼運算(計數、累加、輸出...)

# FOR LOOP的最基本語法

Counter (計數器)

設定迴圈重複的次數

- **for** **i** **in** **range(n)** :

```
python command1  
python command2  
...
```

這一段程式會被反覆執行  
重覆的次數由n設定

在**for...:**之後的幾行，只要開頭對齊(縮排情況一樣)，都會被當成同一個**loop**的內容

## FOR LOOP的例子(1): 一樣的指令做n次

```
print('hello world!')  
print('hello world!')  
print('hello world!')  
print('hello world!')  
print('hello world!')
```

```
for i in range(5):  
    print('hello world!')
```

改用For loop來寫



# 迴圈中的”計數器” (counter)

## Counter (計數器)

- **for** **i** in range(n) :

python command1  
python command2  
...

- 迴圈每重複一次，i的值會改變
- i值會根據range(...)的設定變化。
  - 以range(5)為例，  
迴圈執行第一次時，i=0，  
執行第二次時i=1，  
執行第三次時i=2，依此類推，  
執行第五次i=4 (n-1)後迴圈結束
- 可以在迴圈中輸出i值，或者利用i值來做累加、連乘等運算

## FOR LOOP的例子(2): 迴圈執行過程中counter的值

```
print('hello world!', 1)  
print('hello world!', 2)  
print('hello world!', 3)  
print('hello world!', 4)  
print('hello world!', 5)
```

```
for i in range(5):  
    print('hello world!', i+1)
```

改用For loop來寫



## FOR LOOP的例子(3): 利用迴圈與counter進行累加

```
A=1+2+3+4+5+6+7+8+9+10+11+....+99+100  
print(A)
```

$$A = \sum_{i=1}^{i=100} i$$

```
A=0  
for i in range(101):  
    A = A+i  
print(A)
```

改用For loop來寫

- 迴圈重覆時，i值會依序遞增，從0, 1, 2, 3...100
- 每次迴圈，把目前的i值與前一次迴圈的A值相加，用這個相加的結果來更新A變數的值  
→ i值不斷累加到A變數
- 可以在迴圈內輸出i與A的值來觀察變化 print(i,A)

## range(...) 還可有其他變化

- **range(...)** 經常用來控制迴圈的次數，以及counter數值的改變情況
- **range(n)**：從0開始，每次增加1，直到**n-1**
  - `for i in range(3):` → `i = 0, 1, 2`
- **range(na, nb)**：從na開始，每次增加1，直到**nb-1**
  - `for i in range(1,3):` → `i = 1, 2`
- **range(na, nb, intv)**：從na開始，以間隔intv遞增，直到**nb-1**  
(intv如果為負值，會從na遞減到**nb+1**)
  - `for i in range(1,8,2):` → `i=1, 3, 5, 7`
  - `for i in range(8,1,-2):` → `i=8, 6, 4, 2`



## 課堂練習(1):利用迴圈與counter進行”連乘”

$$A = 10!$$

```
B=1*2*3*4*....*9*10
```

```
B=1  
for i in range(1,11) :  
    B = ?????  
  
print(B)
```

改用For loop來寫

## 課堂練習(2): 利用迴圈與counter進行”連乘”的累加

$$A=1!+2!+3!+4!+\dots+9!+10!$$

$$A = \sum_{i=1}^{i=10} i!$$

改用For loop來寫

```
A=0
B=1
for i in range(1,11) :
    B = ???
    A = ???
print(A)
```

- 提示：讓每次迴圈計算出  $B = i!$  再把每次算出的B累加到A變數

## Counter還可以有更多變化

- 可以把for i in ...想成把某個清單中的數值在每次迴圈依序取出來，做為i變數的值
- 清單可以由range(...)函數產生，可以是一個字串變數，或這是list或陣列(array) (另外一份講義介紹)
  - 例如: 依序取出message字串變數裡的字元

```
message='hello!'
```

```
for A in message:  
    print (A)
```

## 多層迴圈

- 可以在迴圈中再寫入迴圈，形成「巢狀迴圈」結構(nested loop)
- 用不同數量的縮排區分每一層迴圈，越內層縮排越多

```
for j in range(1,6):
```

```
    print(j)
```

```
    for i in range(1,4):
```

```
        k=i*j
```

```
        print(j,'*',i,'=', k)
```

運算順序：

外層迴圈的counter j 先保持在1，  
讓內層迴圈counter i 由1到3繞完，

之後外層的j增加1 (j=2)，  
並且再進行一輪內層迴圈 (i=1,2,3)

直到外層迴圈也全部繞完 (j=5) 為止