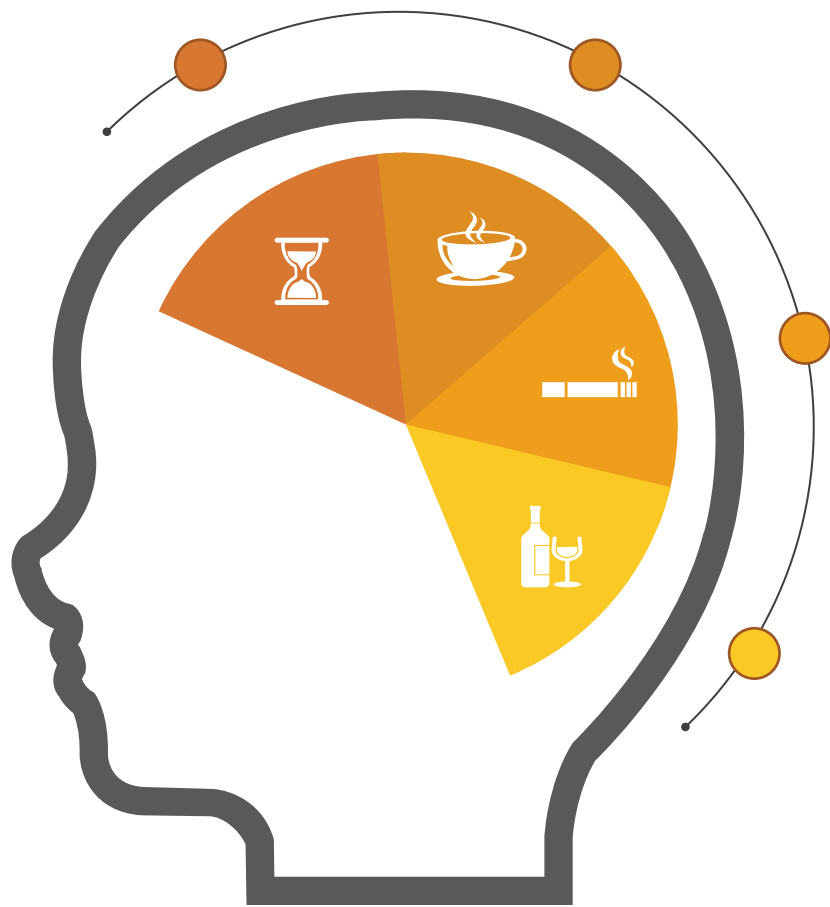

東吳大學 X 理律學堂

—— 法律科技學 - Fundamental (二) ——

吳貫綸助教

資料型態



01

數值型態: int, float, bool

02

字串型態: str, chr

03

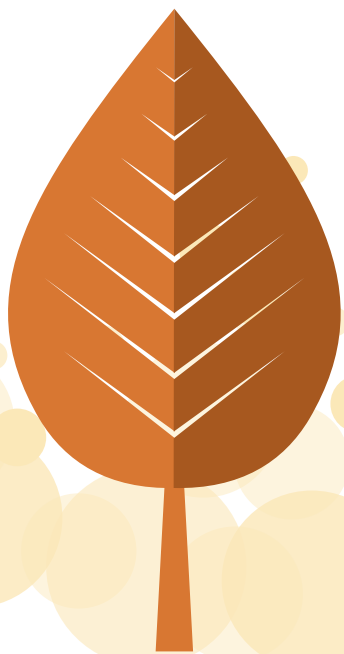
容器型態: list, dict, tuple

常用函式

Sum



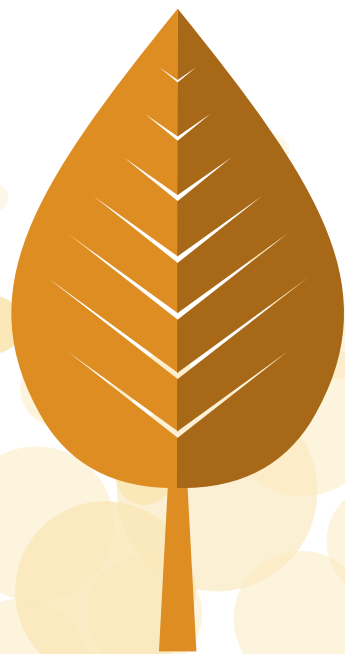
求和計算



Len



返回長度值



Max



回傳最大值



Min



回傳最小值

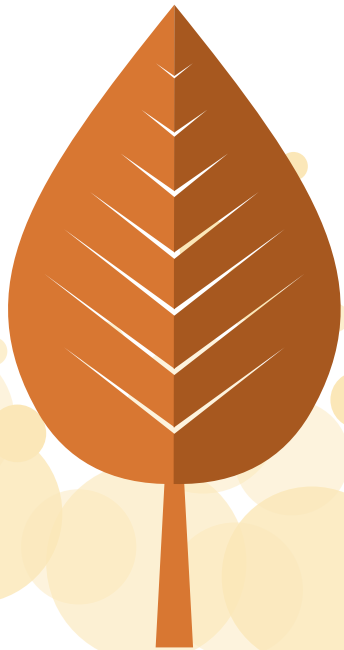


常用函式

Sorted



排序動作



Type



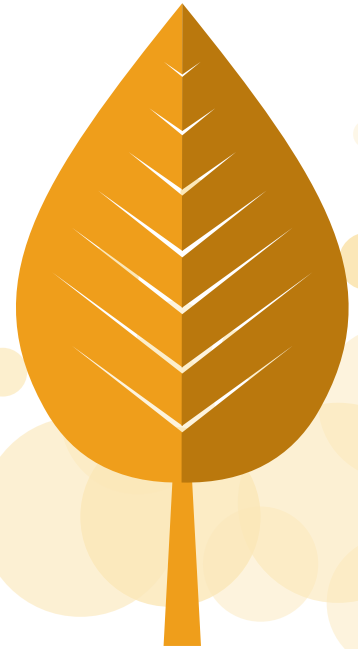
查看資料型態



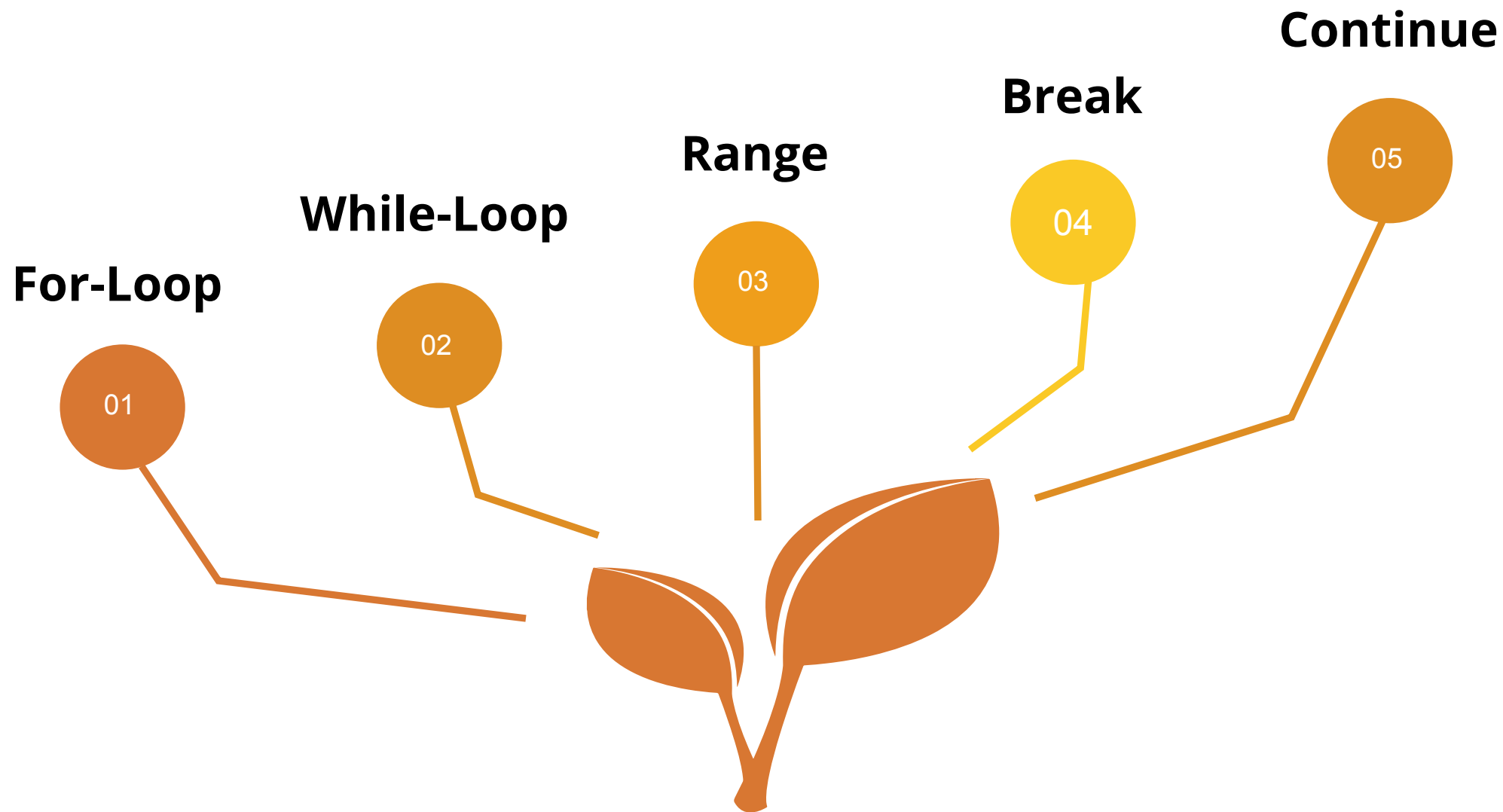
Enumerate



輸出索引與元素



迴圈指令



條件判斷

if (條件式 1) :

若條件1成立, 做以下動作...

elif (條件式 2) :

若條件2成立, 做以下動作...

elif (條件式 3) :

若條件3成立, 做以下動作...

else:

若上述條件都不成立, 做以下動作..

```
score = int(input('請輸入考試分數 (1-100): '))

if score > 100:
    print('請勿亂輸入!')
elif score >= 90 and score <= 100:
    print('成績: A+')
elif score >= 80 and score < 90:
    print('成績: B+')
elif score >= 70 and score < 80:
    print('成績: B')
else:
    print('成績: C')
```



For – 迴圈小學堂

- 設定變數及範圍
- 可搭配range/break/continue

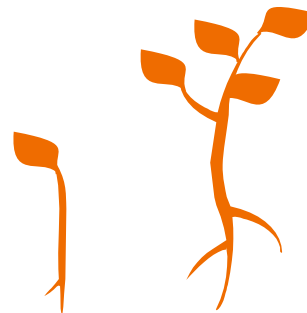
For 變數名稱 **in** 範圍：
重複執行程式碼

```
In [24]: #for-Loop練習  
number=[1, 2, 3]  
  
for i in number:  
    print(i)
```

```
1  
2  
3
```

```
In [6]: for letter in 'Python':  
        if letter == 'h':  
            break  
        print('當前字母 :', letter)
```

```
當前字母 : P  
當前字母 : y  
當前字母 : t
```



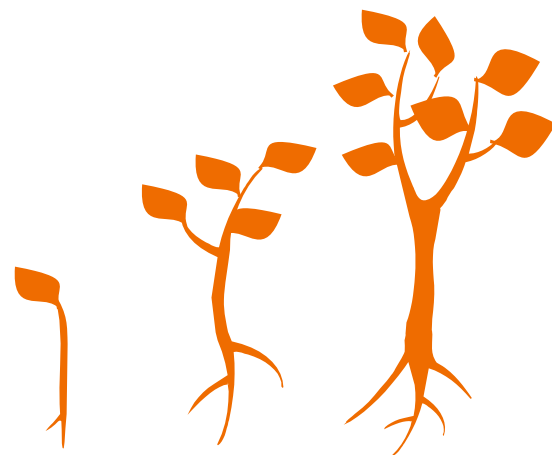
while – 迴圈小學堂

- 設定條件式
- 當條件為滿足，程式停止。
- 可搭配break/continue

while 判斷式：
重複執行程式碼

```
In [3]: #while-Loop  
count = 0  
while (count < 9):  
    print("The count is:", count)  
    count = count + 1  
print("Good bye!")
```

```
The count is: 0  
The count is: 1  
The count is: 2  
The count is: 3  
The count is: 4  
The count is: 5  
The count is: 6  
The count is: 7  
The count is: 8  
Good bye!
```



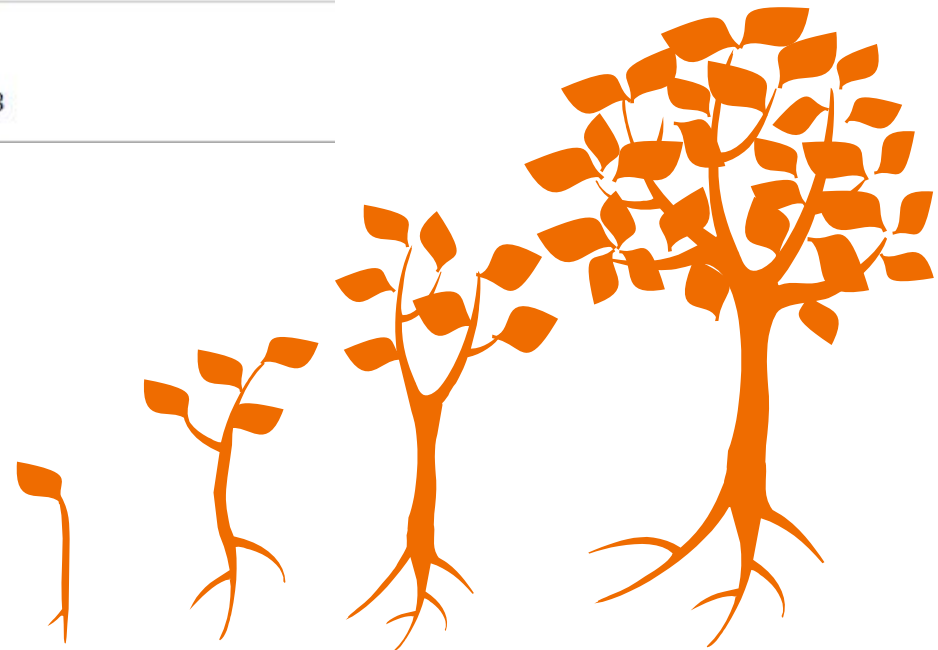
Python 包裝函式

- 程式的重複利用性
- 程式的易讀性
- 程式的易除錯性
- 程式的一致性

用法: **def** 函式名稱(參數):

語句...

```
In [7]: def BMI(h,w):  
        h= h / 100  
        bmi=w /(h**2)  
        return bmi  
  
        h=int(input("請輸入您的身高 (cm):"))  
        w=int(input("請輸入您的體重 (kg):"))  
  
        bmi=BMI(h,w)  
        print ("您的BMI為: ", (bmi))  
  
請輸入您的身高 (cm):150  
請輸入您的體重 (kg):40  
您的BMI為: 17.77777777777778
```



Python 匿名函式(lambda)

- 不用事先定義
- 可重複使用

```
In [2]: #匿名函式
add = lambda x, y: x + y
print(add(1, 2))
print(add(3, 5))

3
8
<function <lambda> at 0x00000212FC000798>
```

用法: **lambda** arg1 , arg2 : expression

