



# 程式設計概論 Programming 101

## 一 程式字串變數的操作 (string)

授課老師：邱淑怡

Date: 3/12/2024

# Outline

- String generation
- Basic operations on strings: addition and multiplication
- Reference: searching and manipulation strings
- Build-in function: len() function

# String(字符串)

- Strings: contain characters
- Strings are enclosed by two single quotes(') or two double quotes("")
- String addition: merge two strings
- String multiplication
- Select substring
- Calculate the length of a string: `len(string_name)`

# Example\_1

- `print("ABC"+"abc")`
- `print("ABC","abc")`
- `print("ABC"+"abc"+"+*-/")`
- `print("ABC"*3)`
- `print(2*"abc")`
- `print(len('abcde'))`

# Indexing & string slicing

5

- Accessing the individual characters in a string is with an index.
- Each character in a string has an index that specifies its position in the string.
- `x1='Python程式設計'`

index	0	1	2	3	4	5	6	7	8	9
內容	P	y	t	h	o	n	程	式	設	計
索引	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

```
print(x[0]) # out P
print(x[1]) # output y
print(x[6]) # output 程
print(x1[-1]) # output 計 (印出最後一個字元)
print(x1[-6]) # output o
print(len(x1)) # output 10 (印出該字串的長度)
```

Python uses 0 as the starting index.

# Indexing & string slicing (cont.)

`x1[a:b]`: 取出`x1`的索引值從`a`到`b`前一個字元的內容 (p.349)

- `x1='Python程式設計'`

index	0	1	2	3	4	5	6	7	8	9
內容	P	y	t	h	o	n	程	式	設	計
索引	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

```
print(x1[2:5])    # 'tho'
print(x1[3:7])    # 'hon程'
print(x1[6:-1])   # '程式設'
print(x1[:2])     # 印出索引0,1 (不包含2)
print(x1[2:])     # 印出索引2...9 (不包含10)
print(x1[9:10])  # 取出最後一個字元印出
```

```
print("Py" in "Python")
print("py" in "Python")
```

# Practice

- `string1='python programming'`
- Q1: print program substring from `string1`
- Q2: print python substring from `string1`
- Q3: print `python_gramming` substring from `string1`


# 字串的片段運算子(進階)

索引	0	1	2	3	4	5	6	7	8	9
內容	P	y	t	h	o	n	程	式	設	計

- `x1='Python程式設計'`
- `x1[a:b:c]=x1[開始索引值:(結束索引值的後一個):遞增量]`
  - a,b,c皆為正或負整數
  - (b-a)的數值=欲取出子字串的長度
  - c代表遞增量及方向
  - a:若沒有寫預設值0
  - b:若沒有寫預設值到最後一個數值
  - c:若沒有寫預設值1
- `print(x1[0:8:2])`
- `print(x1[::-2])`

`x1[-1:0:-2]`

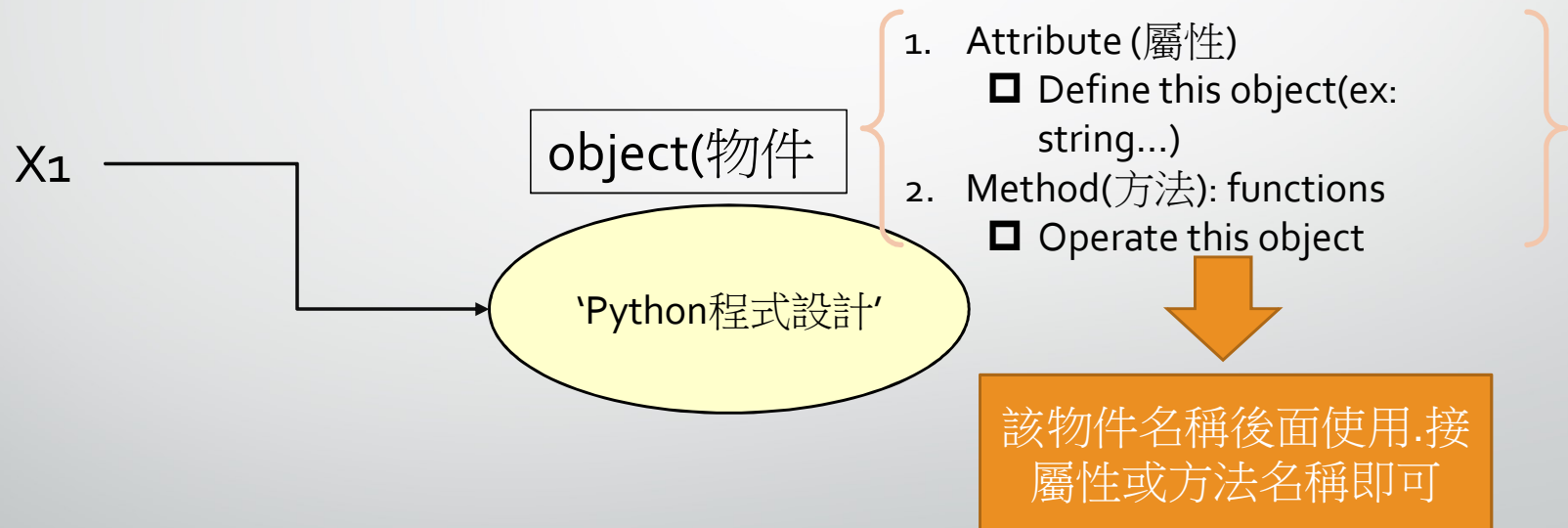




參考資料：  
字串進階 -  
使用字串類別(class)內提供的函式(function)

# Functions in the string class

X1='Python程式設計'



# String conversion method

- `x1='Python程式設計'`
- `print(x1.upper())`: 傳回字串`x1`的所有字元轉換成大寫的字串
- `print(x1.lower())`: 傳回字串`x1`的所有字元轉換成小寫的字串
- `print(x1.swapcase())`: 傳回字串`x1`大小寫互換的字串
- `print(x1.replace(old,new))`: 傳回字串`x1`的`old`字串取代為`new`的字串
- `print(x1.capitalize())`: 傳回字串`x1`的第一個字元轉換成大寫的字元
- `print(x1.title())`: 傳回字串`x1`的每個單字第一個字元轉換成大寫的字元

若 `old`沒出現在`x1`?

# 針對特定子字串進行操作

- `find()` method: 找出特定子字串，找出字串裡面尋找子字串第一次出現的位置，傳回的結果是第一次出現的索引值，若沒找到則傳回-1
- `rfind()` method: r是reverse，從字串尾段開始搜尋，並傳回第一次出現該子字串出現的索引值
- `str2='me myself and I'`
- `print(str2.find('and'))`
- `print(str2.find('you'))`
- `print(str2.find('e'))`
- `print(str2.rfind('e'))`

# 字串中搜尋子字串的方法

```
var1='PythonPythonPythonPython'  
print(var1.count('Python')) #var1字串出現'Python'的次數  
print(var1.startswith("Python")) #var1字串是否以'Python'開頭  
print(var1.endswith('Python')) #var1字串是否以'Python'結尾  
print(var1.find('Python')) #var1字串中出現'Python'的最小索引  
print(var1.rfind('Python')) #var1字串中出現'Python'的最大索引  
print(var1.startswith('Python', 0, len(var1)-5 ))  
print(var1.endswith('Python', 0, len(var1)-3 )) # 進階例子
```

# 字串內置方法(cont.)

- Python提供一系列的isxxx的功能，可判斷字串是否有xxx屬性
  - `var='python programming 123'`
  - `print(var.isdigit())` # 是否全為數字
  - `print(var.isalpha())` # 是否全為字母
  - `print(var.isalnum())` # 是否全為數字或字母
  - `print(var.islower())` # 是否全為小寫
  - `print(var.isupper())` # 是否全為大寫
  - `print(var.isspace())` # 是否全是空白，是傳回True
  - `print(var.istitle())` # 是否每個單字第一個字元都是大寫字元，是傳回True

傳回True/False

# Practice

- `s1='Today is Friday'`
  1. `s1`是否包含'day'?
  2. 'day'出現在`s1`的次數
  3. 利用`s1`建立一個新字串變數`new1`，令`new1`的值為"Today is Saturday"
  4. `s1`的每個單字的第一個字元都是大寫嗎?

# Summary

## Priority(優先順序表)

運算子	說明
[...], (...), {...}	list, set, tuple, dict
a[i], a[i:j]	index(索引)
a**b	Exponent(指數)
+a, -a	Positive/negative number (正, 負號)
a*b, a/b, a//b, a%b	Multiplication, division (乘法, 除法, 餘數...)
a+b, a-b	Addition, subtraction (加法, 減法)
<, >, <=, >=, ==, !=	Comparison(比較)
not a	Not logic
a and b	And logic
a or b	or logic

high

low

- Python uses **0** as the starting index.
  - string slicing
  - list (in the future)





# Student exercise 2

# Question 1

1. `str3="python language programming"`

- Q1: 請將`str3`轉換成大寫字母並印出結果
- Q2: 請利用`str3`印出"python"
- Q3: 請利用`str3`印出"programming"
- Q4: 請利用字串搜尋功能將`str3`中`programming`的'p'的索引值印出來
- Q5: 請問`str3`中有幾個'g'印出結果

## Question 2

2. 印出下列各問題的結果:

- `s1='HappyNewYear'`
- `s2='happynewyear'`
- `s3='new'`
- `s4='birthday'`
- Q1: `s1`的長度
- Q2: `s1`和`s2`長度是否相等，請印出False or True?
- Q3: `s3`是否存在於`s1`，請印出False or True?
- Q4: 如何利用上述的四個字串得到'Happy birthday'

# Review

- Textbook: chapter 9: 9.2