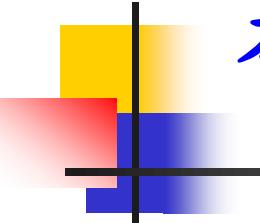


# 資料科學 進階函式與套件



國立宜蘭大學資訊工程系  
吳政瑋 助理教授  
蔡佳韋 課程助教

[wucw@niu.edu.tw](mailto:wucw@niu.edu.tw)



# 本堂教學重點

---

- 了解 Python 內建**數值函式**
- 了解 Python 內建**字串函式**
- 了解 Python 內建**亂數套件**
- 了解 Python 內建**時間套件**

# Python 內常見的數值函式

函式	主要功能	範例	結果
int(a)	將a轉成整數	int(94.87)	94 (整數)
float(a)	將a轉成浮點數	float("94")	94.0 (浮點數)
str(a)	將a轉成字串	str(94)	'94' (字串)
chr(a)	取得整數a的字元	chr(66)	'B' (字元)
hex(a)	取得a的16進位值	hex(94)	0x5e
oct(a)	取得a的8進位值	oct(94)	0o136
ord(a)	取得字元a的 Unicode 編碼值	ord("你")	20320

# Python 內常見的數值函式

函式	主要功能	範例	結果
<code>abs(v)</code>	取得v的絕對值	<code>abs(-87)</code>	87
<code>round(v)</code>	以4捨6入法取得v的近似值	<code>round(94.87)</code>	95
<code>pow(x, n)</code>	取得x的n次方	<code>pow(2, 3)</code>	8
<code>pow(x, n, m)</code>	取得x的n次方除以z的餘數	<code>pow(2, 3, 5)</code>	$2^3 = 8$ $8 \% 5 = 3$
<code>divmod(x, y)</code>	取得x除以y的商數及餘數的元組	<code>divmod(40, 5)</code>	$40/5 = 8 \dots 0$ (餘0) 回傳(8, 0)

# Python 內常見的數值函式

函式	主要功能	範例	結果
<code>len(L)</code>	取得串列L之長度	<code>len([1, 2, 3])</code>	3
<code>min(L)</code>	取得串列L之最小元素值	<code>min([1, 2, 3])</code>	1
<code>max(L)</code>	取得串列L之最大元素值	<code>max([1, 2, 3])</code>	3
<code>sorted(L)</code>	取得串列L之由小到大排序結果	<code>sorted([3, 1, 2])</code>	[1, 2, 3]
<code>sorted(L, reverse = true)</code>	取得串列L之由小到大之倒序結果	<code>sorted([3, 1, 2], reverse = true)</code>	[3, 2, 1]
<code>sum(L)</code>	取得串列L所有元素值總合	<code>sum([1, 2, 3])</code>	$1 + 2 + 3 = 6$
<code>sum(L, v)</code>	取得串列L所有元素值加v的總合	<code>sum([1, 2, 3], 4)</code>	$1 + 2 + 3 + 4 = 10$

# Python 內常見的字串函式

函式	主要功能	範例	結果
<code>len(s)</code>	取得字串s之長度	<code>len("dog")</code>	3
<code>s.find(w)</code>	搜尋字串w在字串s中的索引值	<code>"dog".find("d")</code> <code>"dog".find("e")</code>	0 -1
<code>s.startswith(w)</code>	判斷字串s是否以字串w開頭	<code>"dog".startswith("d")</code> <code>"dog".startswith("e")</code>	True False
<code>s.endswith(w)</code>	判斷字串s是否以字串w結尾	<code>"dog".endswith("g")</code> <code>"dog".endswith("e")</code>	True False
<code>s.islower()</code>	判斷字串s是否皆為小寫	<code>"dog".islower()</code> <code>"Dog".islower()</code>	True False
<code>s.isupper()</code>	判斷字串s是否皆為大寫	<code>"DOG".isupper()</code> <code>"Dog".isupper()</code>	True False
<code>s.upper()</code>	將字串s轉為大寫	<code>"dog".upper()</code>	DOG
<code>s.lower()</code>	將字串s轉為小寫	<code>"DOG".lower()</code>	dog

# Python 內常見的字串函式

函式	主要功能	範例	結果
s.center(n)	將字串s擴充為n個字元並置中	"dog".center(5) "dog".center(6)	□dog□ □dog□□
s.ljust(n)	將字串s擴充為n個字元並置左	"dog".ljust(5) "dog".ljust(6)	dog□□ dog□□□
s.rjust(n)	將字串s擴充為n個字元並置右	"dog".ljust(5) "dog".ljust(6)	□□dog □□□dog
s.lstrip()	移除字串s左方的空白字元	"□□dog".lstrip()	dog
s.rstrip()	移除字串s右方的空白字元	"dog□□".rstrip()	dog

# Python 內常見的字串函式

函式	主要功能	範例	結果
s.replace(a, b)	將字串s中的字串a 改為字串b	"dog".replace("o","e") "dog".replace("k","e")	deg dog
s.split(q)	依字串q切割字串s， 結果為串列	"1 3 5 8".split(" ")	["1", "3", "5", "8"] 串列中的元素仍為 字串
s.join(L)	依字串q合併L中的 元素，結果為字串	" ".join(["1","3","5","8"]) "#".join(["1","3","5","8"])	"1 3 5 8" "1#3#5#8"

# Python 套件匯入方式

- 在Python 中，使用套件需使用 **import** 指令

```
import 套件名稱
```

- 例如匯入**亂數**的套件，語法如下：

```
import random
```

# Python 套件函式

- 套件中通常有許多函式供使用者使用
- 使用套件函式的語法如下：

套件名稱.函式名稱

- 例如：

```
import random  
v = random.randint(1,10) #產生1(含)~10(含)之間的整數  
print(v)
```

# Python 套件函式

```
import random  
v = random.randint(1,10) #產生1(含)~10(含)之間的整數  
print(v)
```

- 以此種方式使用 **套件函式** 時，每次都要寫 **套件名稱**，相當地不方便。
- 有些套件名稱甚至很長，亦會造成輸入時的困擾。

# 匯入套件中的所有函式

- 使用下列語法後，使用 **套件函式** 時就不必再寫 **套件名稱**
- 匯入套件中的所有函式，其語法如下：

```
from 套件名稱 import *
```

- 例如：

```
from random import *
v = randint(1, 10)    #產生1(含)~10(含)之間的整數
print(v)
```

# 套件另取別名

```
from 套件名稱 import *
```

- 使用上述指令雖然方便，然而，**不同套件的函式名稱可能相同**，若此發生將導致程式發生錯誤。
- 為兼顧安全性及便利性，可使用下列語法，對套件名稱另取一個簡短的別名：

```
import 套件名稱 as 別名
```

# 舉例說明：套件另取別名

```
from random import *
v = randint(1, 10)    #產生1(含)~10(含) 之間的整數
print(v)
```

7

---

```
import random as r
v = r.randint(1, 10)    #產生1(含)~10(含) 之間的整數
print(v)
```

2

# 亂數套件內常見的函式

函式	主要功能	範例	結果
r.randint(a, b)	在a~b之間，隨機產生1個整數	r.randint(1, 10)	7
r.randrange(a,b,c)	在a~b之間每隔c中隨機產生1個整數	r.randrange(0,11,2)	10(偶數)
r.random()	由0~1之間，隨機產生1個浮點數	r.random()	0.893398
r.uniform(a,b)	在a~b之間，隨機產生1個浮點數	r.uniform(1,10)	6.351865
r.choice(s)	在字串s中，隨機挑出1個字元	r.choice("abcd")	'c'
r.sample(s, n)	在字串s中，隨機挑出n個字元	r.sample("abcd", 3)	['c', 'b', 'a']
r.shuffle(L)	產生串列L的洗牌結果	r.shuffle([1, 3, 4, 9])	[9, 4, 3, 1]

# r.shuffle() 套件函式

```
import random as r  
L1 = [1,5,6,9,13]  
L2 = r.shuffle(L1)  
print(L1)  
print(L2)
```

[1, 6, 5, 9, 13]

None

# Python 時間套件

函式	主要功能
t.time()	從1970年1月1日0時到現在經過幾秒
t.localtime([時間數值])	以傳入的[時間數值]取得[時間元組]
tctime([時間數值])	以傳入的[時間數值]取得[時間字串]
t.sleep(h)	程式停止 h 秒
t.clock()	取得程式執行時間

# 教師資訊



- 姓名:吳政瑋 (小吳老師)
- 現職:宜大資工專案助理教授
- 學歷:成功大學資工博士
- 研究興趣:資料探勘、人工智慧、AIoT應用
- 通訊方式
  - 電子信箱:wucw@niu.edu.tw
  - 校內電話: (03)9317331
  - Line: silvemoonfox
  - Office: 格致大樓E405室
  - 數位學習園區
- 實驗室:AI與資料科學實驗室
  - <https://sites.google.com/view/cwwuadslab/>

# 意見交流

歡迎提供意見與指導!!

您的寶貴意見將使本系更進步!!